Version 1.0

For Web Services on Devices

Date: November 9, 2006

Microsoft Corporation Non-Confidential Technical Documentation License Agreement (Non-Standard)

READ THIS! THIS IS A LEGAL AGREEMENT BETWEEN MICROSOFT CORPORATION ("MICROSOFT") AND THE RECIPIENT OF THESE MATERIALS, WHETHER AN INDIVIDUAL OR AN ENTITY ("YOU"). IF YOU HAVE ACCESSED THIS AGREEMENT IN THE PROCESS OF DOWNLOADING MATERIALS ("MATERIALS") FROM A MICROSOFT WEB SITE, BY CLICKING "I ACCEPT", DOWNLOADING, USING THE MATERIALS, YOU AGREE TO THESE TERMS. IF THIS AGREEMENT IS ATTACHED TO MATERIALS, BY ACCESSING OR USING THE ATTACHED MATERIALS, YOU AGREE TO THESE TERMS.

For good and valuable consideration, the receipt and sufficiency of which are acknowledged, You and Microsoft agree as follows:

1. You may review these Materials only as a reference to assist You in planning and designing Your product, service or technology ("Product") to interface with a Microsoft Product as described in these Materials. All other rights are retained by Microsoft; this agreement does not give You rights under any Microsoft patents. You may not (i) duplicate any part of these Materials, (ii) remove this agreement or any notices from these Materials, or (iii) give any part of these Materials, or assign or otherwise provide Your rights under this agreement, to anyone else.

2. The information contained in these Materials is also available through the Microsoft Windows Driver Kit (WDK). You will receive additional intellectual property rights by agreeing to the terms of the WDK.

3. All Materials are provided entirely "AS IS." To the extent permitted by law, MICROSOFT MAKES NO WARRANTY OF ANY KIND, DISCLAIMS ALL EXPRESS, IMPLIED AND STATUTORY WARRANTIES, AND ASSUMES NO LIABILITY TO YOU FOR ANY DAMAGES OF ANY TYPE IN CONNECTION WITH THESE MATERIALS OR ANY INTELLECTUAL PROPERTY IN THEM.

4. If You are an entity and (a) merge into another entity or (b) a controlling ownership interest in You changes, Your right to use these Materials automatically terminates and You must destroy them.

5. This agreement is governed by the laws of the State of Washington. Any dispute involving it must be brought in the federal or state superior courts located in King County, Washington, and You waive any defenses allowing the dispute to be litigated elsewhere. If there is litigation, the losing party must pay the other party’s reasonable attorneys’ fees, costs and other expenses. If any part of this agreement is unenforceable, it will be considered modified to the extent necessary to make it enforceable, and the remainder shall continue in effect. This agreement is the entire agreement between You and Microsoft concerning these Materials; it may be changed only by a written document signed by both You and Microsoft.

Contents

415

416 List of Tables

419 List of Figures

456

1. Overview and Scope

This service definition is compliant with the Devices Profile for Web Services [DEVICE].

This service-type enables the following functions:

- Scanning

This service template does not address:

- Printing
- Copying
- Faxing
  - Inbound
  - Outbound
- Multi-function Devices

2. Device Service Modeling Definitions

2.1. Device Service-Class

This Web Service for Devices (WSD) type maps to the WSDL “porttype” element. This combines operations, events and properties into a functional unit. A device service-class that is compliant with this template is identified with the following target namespace: “**http://schemas.microsoft.com/windows/2006/08/wdp/scan**”.

2.2. Terminology

This section defines terms that are used throughout this specification. These terms are always capitalized in order to indicate that they have the meaning defined in this section.

2.2.1. Conformance Terminology

The following terms have special meaning relating to conformance and so are always indicated in all capital letters:

a) MUST - This word, or the term "REQUIRED", mean that the definition is an absolute requirement of the specification.

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

c) SHOULD - This word, or the adjective "RECOMMENDED", mean 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.

d) SHOULD NOT - This phrase, or the phrase "NOT RECOMMENDED" 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.

e) MAY - This word, or the adjective "OPTIONAL", mean 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. An implementation which does include a particular option MUST be prepared to interoperate with another implementation which does not include the option

2.2.2. Other terminology

This document uses the terminology defined in the Devices Profile for Web Services [DEVICE], such as: operation, device description, and argument. This sub-section defines the following additional terms that are capitalized in order to indicate their specific meaning as defined in this section.

a) Control Point (CP) - the entity that initiates requests to the Scanner on behalf of the end user.

b) Scan Service (or Scanner) - the [DEVICE] entity that accepts operations from a CP (clients), returns responses including image data, and sends events.

2.2.3. Notation: use of quotation marks

Throughout this document, single quotes (‘) are used around literal string and integer values in running text, but not in Tables. The single quotes are not part of the values. Double quotes (“) are used around words in running text to indicate special English meanings. Element names, arguments names, and operation names are not quoted.

2.3. References

This section lists the references that this document refers to and the tag inside square brackets that is used for each such reference:

[DEVICE] – J. Schlimmer, et al, “Devices Profile for Web Services”, February 2006.
(See http://specs.xmlsoap.org/ws/2006/02/devprof/devicesprofile.pdf/.)

509 [MTOM] - N. Mendelsohn, et al, "SOAP Message Transmission Optimization Mechanism," January 2005.
510 (See http://www.w3.org/TR/2005/REC-soap12-mtom-20050125/.)

511 [ADDRESS] - D. Box, et al, "Web Services Addressing (WS-Addressing)," August 2004.
512 (See http://www.w3.org/Submission/2004/SUBM-ws-addressing-20040810/.)

513 [DISCOVERY] - J. Beatty, et al, "Web Services Dynamic Discovery (WS-Discovery)," April 2005.
514 (See http://schemas.xmlsoap.org/ws/2005/04/discovery.)

515 [WSDL] - E. Christensen, et al, "Web Services Description Language (WSDL) 1.1," March 2001.
516 (See http://www.w3.org/TR/2001/NOTE-wsdl-20010315.)

517 [WSDL Binding for SOAP 1.2] - K. Ballinger, et al, "WSDL Binding for SOAP 1.2," April 2002.
518 (See http://groups.yahoo.com/group/soapbuilders/files/soap12WSDL.htm.)

519 [EVENTING] - L. Cabrera, et al, "Web Services Eventing (WS-Eventing)," August 2004. (See
520 http://msdn.microsoft.com/ws/2004/08/ws-eventing/.)

521 [PNPX] – C. Brodeur, "PnP-X: Plug and Play Extensions for Windows", July 2006.
522 (See http://www.microsoft.com/whdc/Rally/pnpx-spec.mspx.)

523 [HTTP] - R. Fielding, et al, "Hypertext Transfer Protocol -- HTTP/1.1," June 1999. (See
524 http://www.ietf.org/rfc/rfc2616.txt.)

525 [URI] - T. Berners-Lee, et al, "Uniform Resource Identifiers (URI): Generic Syntax," January 2005.
526 (See ftp://ftp.rfc-editor.org/in-notes/rfc3986.txt.)

527 2.4. Purpose

528 The purpose of this document is to describe the operational model for network connected scanning devices. The Devices
529 Profile for Web Services [DEVICE] defines a peer-to-peer network connectivity model allowing intelligent applications to
530 discover and make use of network resident devices. This document describes how Scanners would operate in this
531 environment. This document focuses on the device control model that will be supported by scanning devices.

532 The scanning model described in this document focuses on walk-up scanning, where scanning is initiated by users at a
533 device. Devices that support this model are typically connected directly to a network and reside in a central location.
534 Scanning in this model is focused primarily on capturing a document and delivering it to a destination client or an
535 application.

536 This document describes how to accomplish walk-up scanning in small (home or workgroup) environments. Scanning in
537 the home is primarily driven from the PC although the user experience is very much a walk-up model. Although the
538 scanning model, as described, focuses on walk-up scanning it should be noted it is not limited to walk-up. The model
539 provides a framework in which all network scanning can be modeled.

540 This document outlines the functional components of Scanning services and describes their interactions with CLIENTs.

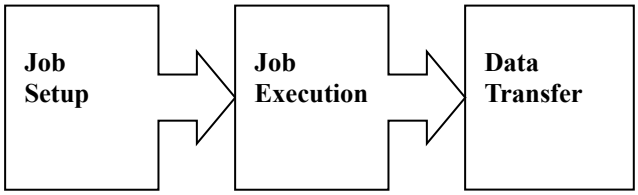
541 3. Scanning Functional Model

542 Operation of a device is modeled as a conversation between a Control Point (CP) and a Service exposed by a device. The
543 operation of scanning can be described as a combination of distinct functions. The design of the Scanner Functional Model
544 exposes these functions as distinct services and allows for different Control Points to control each function. In many
545 instances, a single CLIENT will be responsible for the complete set of interactions with the scanner device – but by
546 logically separating the CLIENTs we provide a foundation to support the many different Scanner usage patterns.

547 The scanning process consists of three logical functions:

548 • Job Setup
549 • Job Execution
550 • Data Transfer

551



552 **Figure 1 – Scan Workflow**

553 The Scanner Model defines a Scan Ticket as the means by which Job Elements are communicated down the line of
554 functions. At Job Setup the Scan Ticket is created based on user selections. At Job Execution a specific scan ticket is
555 submitted to the Scan Service by a CLIENT. During Job Execution the Scan Ticket is used by the device to determine the
556 appropriate settings to use during the scan. After the Job Execution has completed, the Scan Ticket may be used to generate
557 a log containing the actual scanning parameters used during execution as well as additional information.

558 3.1. Theory of Operation

559 3.1.1. Summary

560 Scanning breaks into three logical sets of interactions:

561 • **Job Setup:** The creation of a scan ticket based on user scanning preferences.

562 • **Job Execution:** The physical scan and generation of data. A scan ticket generated in the Job Setup function is
563 used to define the parameters of the Job Execution. Job Execution ends when the original document has been
564 scanned and all of the scanned data is transferred to an external destination or internal storage.

565 • **Data Transfer:** The movement of data from the scanner's buffer to a destination. Data transfer is contained within
566 the job execution function but may be asynchronous to the physical scan. A scanner will always push data to a
567 destination as the response to an operation.

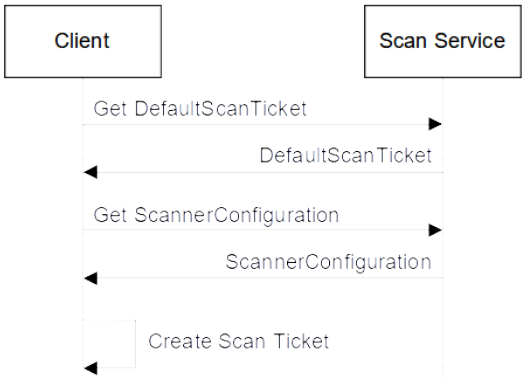
568 Each of the above sets of operations represents a series of interactions between a service component provided by a scanner
569 and a CLIENT.

570 3.1.2. Function Descriptions

571 This section describes the primary interactions that occur during each logical phase of the Scanning process.

572 3.1.2.1. Job Setup

573 The Job Setup function is provided by a CLIENT. The CLIENT providing Job Setup interacts with a Scan Service to
574 determine the capabilities of the device. The CLIENT provides an interface to allow users to choose scanning options such
575 as color mode, resolution, etc. based on the capabilities of the device. The CLIENT then creates a scan ticket that specifies
576 scanning parameters chosen by the user. The scan ticket will be used during Job Execution (See Section 3.1.2.2.).



577
578 **Figure 2 – Client Job Setup**

579 ### 3.1.2.2. Job Execution

580 Two different methods of Job Execution are supported:

581 • **Device Initiated: A** device can initiate execution as the result of a user interacting directly with the device.
582 • **CLIENT Initiated:** A CLIENT can initiate Job Execution by sending a request[2] to the scan device.
583
584 If a CLIENT is initiating the execution of a job, it may subscribe for events of interest beforehand. If a device will initiate
585 the execution, a client must register the intended destination of the scanned document.

586 ### 3.1.2.3. Data Transfer

587 • **Response Mode:** The CLIENT starts a scan based on a **ScanAvailableEvent** notification from the device. The
588 CLIENT sends down a **CreateScanJob** operation with the requested scan parameters (ScanTicket) and the device
589 processes the scan. The data from the scan is sent back to the CLIENT as a part of the response to the
590 **RetrieveImage** operation(s).

30 [2] See the **CreateScanJob** operation defined in the Scan Service Specification

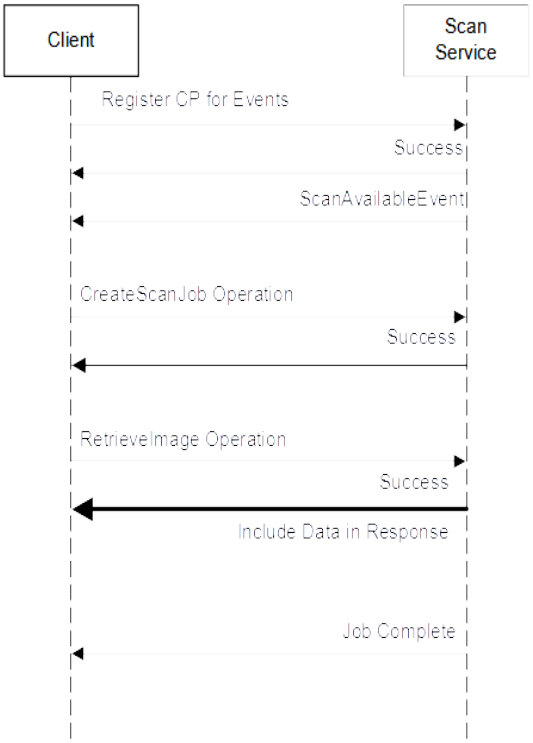


Figure 3 – Response Mode Data Transfer

The Data Transfer phase ends when all of the data has been successfully transferred or an unrecoverable error occurs.

3.2. Job Concurrency

It is possible that multiple jobs may exist at the same time. A scan for one job may be complete with data still transferring while another job is activated. The number of jobs transferring data is limited to the capacity of device resources.

The following diagram illustrates the number of allowable concurrent jobs during the Job Functions. Shown in the diagram are jobs X, Y and Z. Jobs X and Y must have completed scanning but may still be transferring data when job Z is activated. In order for another job to be activated, job Z must reach the completion of the physical scan. The Buffered Data Transfer portion of the diagram illustrates that multiple data transfers can be occurring at the same time.

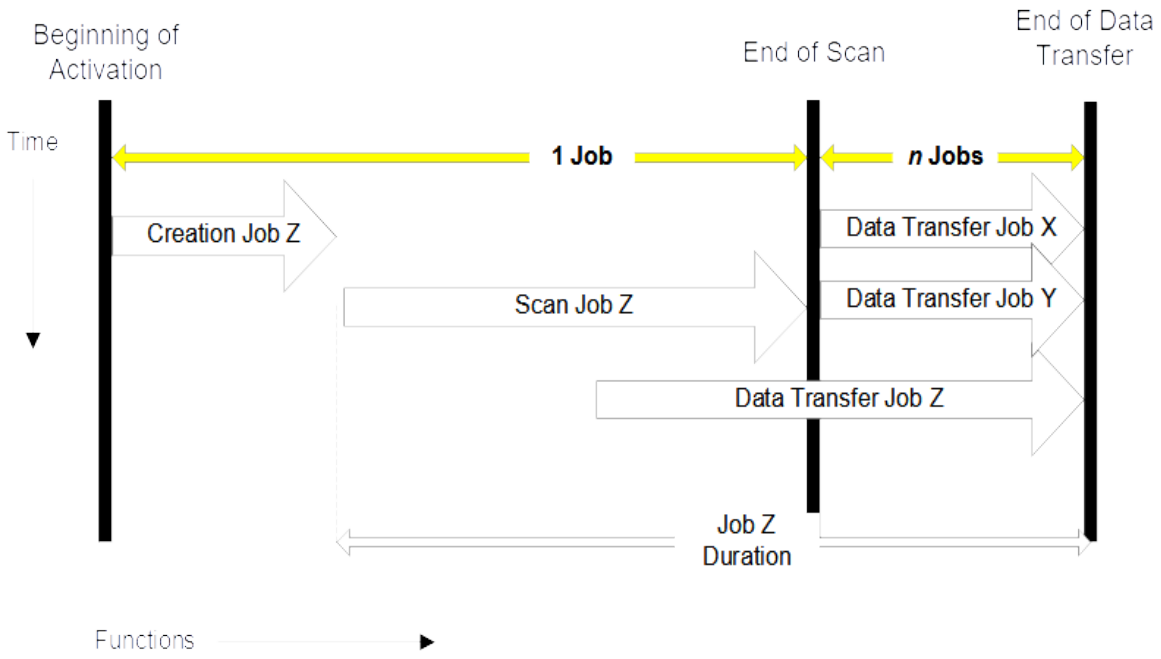


Figure 4 – Job Concurrency Example

3.3. Usage Scenarios

3.3.1. Scan to Workstation, Device Initiated

- Device Notification of waiting scan (event)
- CLIENT Initiated Execution, Response-based Data Transfer – Figure 3

A user wants to scan a document to his workstation. The user places their media on the platen/in the ADF. The user scrolls through the destinations on the scanner front panel. When the correct destination is found the user presses the SCAN button on the device. The scanner sends a **ScanAvailableEvent** to the specified destination (only). The CLIENT submits a **CreateScanJob** operation to the scanner using settings preconfigured by the user and then retrieves the data as part of the response to one or more **RetrieveImage** operations.

3.3.2. Scan to Workstation, Client Initiated

- CLIENT Job Setup – Figure 2
- CLIENT Initiated Execution, Response-based Data Transfer – Figure 3

A user wants to scan a document to his workstation. The UI displayed on or near the scanner is hosted by a CLIENT external to the scanner. The user selects scan settings using the UI. The user places the document on the platen/in the feeder and presses the Start button on the UI. The external CLIENT submits a **CreateScanJob** operation to the scanner and then retrieves the data as part of the response to one or more **RetrieveImage** operations.

3.4. Service Summary

3.4.1. Jobs

The Scan service's main task is to accept scan jobs from clients, queue them up (if the scanner is capable of handling more than one job at a time), scan them, and transfer the output data. A job is identified by an integer, the *JobId*, which is allocated by the device. The *JobId* is returned by the **CreateScanJob** operation.

The set of jobs the scanner has in its queue can be found by using the **GetActiveJobs** command.

3.4.2. Operations

The following operations are defined and MUST be supported by conforming ScannerService implementations:

- **CreateScanJob** – This operation is used to submit a job to the scanner. A scan ticket defining the scanning parameters is supplied with the request. An allocated *JobId* and the image information are returned in the response.
- **RetrieveImage** – This operation is used to retrieve the image data associated with a previously submitted scan job. The *JobId* and *JobToken* are supplied with the request. The image data are returned in the response.
- **CancelJob** – This can be used to cancel a job using the *JobId*.
- **ValidateScanTicket** – This can be used to verify the settings in a *ScanTicket* for a future scan operation.
- **GetScannerElements** – This operation can be used to query for scanner elements.
- **GetJobElements** – This operation can be used to query for job elements for a particular job.
- **GetActiveJobs** – This operation can be used to obtain a list of all the currently active Jobs on the scanner and a subset of each job's elements.
- **GetJobHistory** – This operation can be used to obtain a list of some of the recently completed Jobs on the scanner and a subset of each job's elements.

3.4.3. Events

Events are defined to inform a CP that a user has asked to scan, to tell about any configurations changes in the scanner and to tell the status of active and finished jobs. The basic event model is based on Web Service Eventing [EVENT]. When using events to allow CPs to synchronize access to scanned data with the device output a unique destination for each client is registered with the scan device. Then when the user requests to scan data to that destination an Event is generated to just the client associated with the requested destination.

3.4.4. Security

The Devices Profile for Web Services [DEVICE] defines how compliant clients and devices interact in a secure manner. This specification assumes all clients and scan devices comply with this security model.

3.4.5. Localization

A WSD Scanner is assumed to be operating within the locale of the user. No other localization mechanism is defined for the Scan Service. The CP (client) is expected to localize the well-known string values to the locale of its user. The CP (client) is expected to convert the enum integer values to human readable string values in the locale of the user.

4. Service Schema

A scanner contains zero or more Jobs. A Job contains processing instructions and one or more output Documents.

The Scanner, Job, and Document elements are grouped together into logical elements. These element groups collect similar elements together. *ScannerDescription* elements for Scanners contain descriptive information that is static per device or administratively set by a CP. "Description" elements for Jobs and Documents contain descriptive information that is administratively set or supplied by a CP during job submission. The *ScannerConfiguration* elements contain information that describes the current physical configuration of the scanner and its options and also all the allowed values for the various *DocumentParameters* elements. The Scanner and Job "Status" elements contain information primarily controlled by automata. These can be indirectly affected by a CP through operations such as **CancelJob**. The *DocumentParameters* elements specify features that are applied to the Documents in the Job (e.g. *Rotation*). Finally the *DefaultScanTicket* element provides the default values for all the elements used in Job creation. Figure 5 below illustrates the relationships between these elements. The member elements of Scanners, Jobs, and Documents are detailed in the following sections.

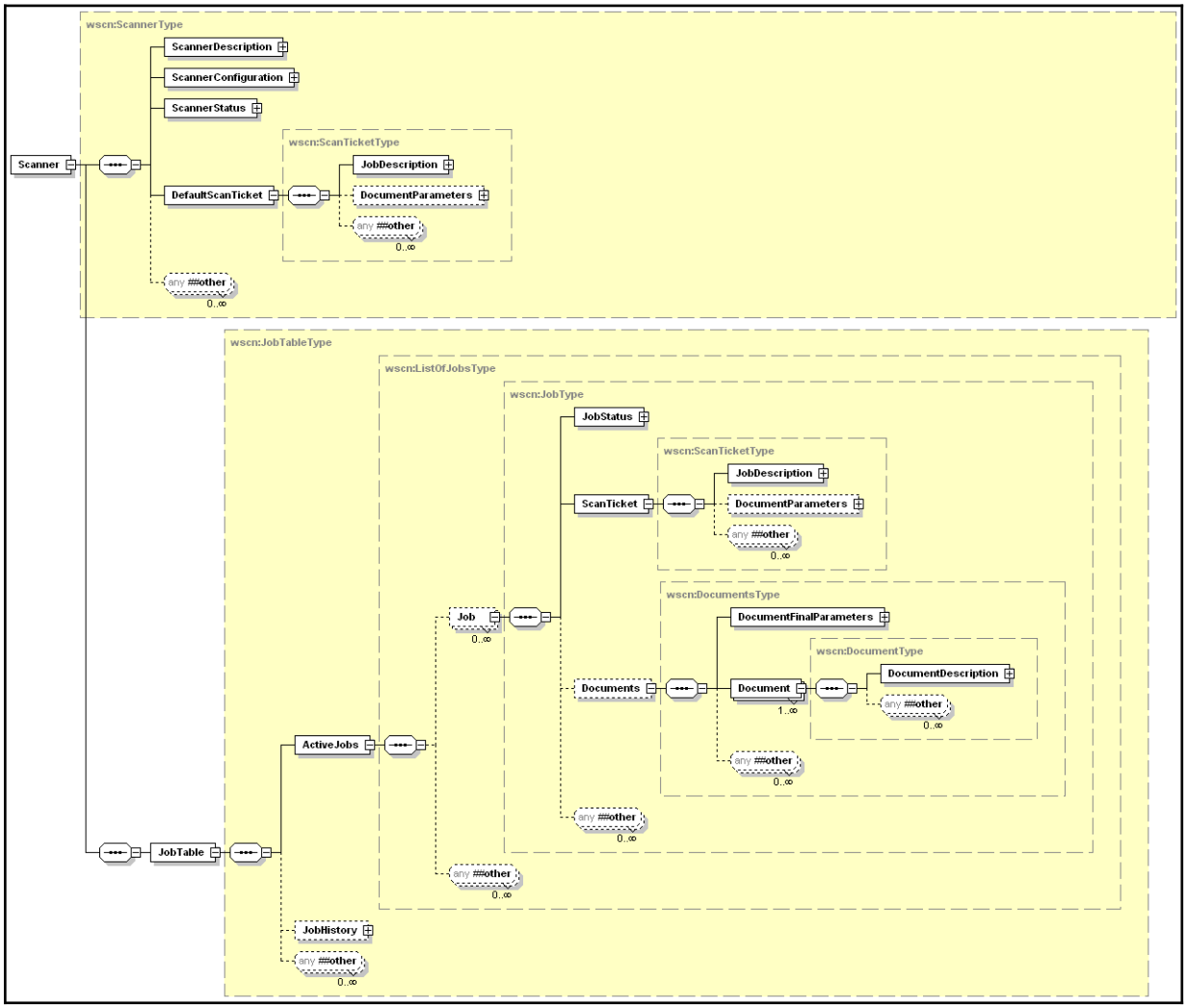


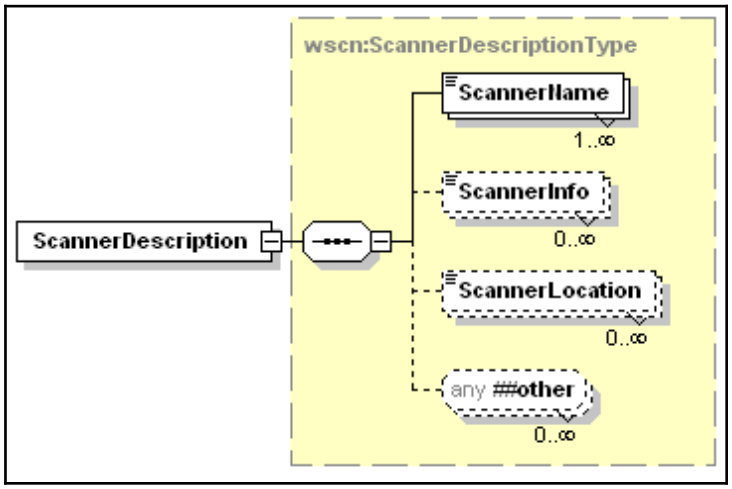
Figure 5 – Scanner Model

4.1. The Scanner's Elements

This section defines the elements that comprise the WSD Scanner Service.

Figure 6 shows the *ScannerDescription* elements for the Scanner. Figure 7 and Figure 8 show the *ScannerConfiguration* elements for the Scanner. Figure 9 shows the *ScannerStatus* elements for the Scanner. Figure 16 shows the *ScanTicket* elements which define the *DefaultScanTicket* for the scanner.

673 4.2. The ScannerDescription Elements



674

675 **Figure 6 - ScannerDescription Elements**

676 All of the Scanner elements contained within the *ScannerDescription* element are described in detail in the following
677 sections.

678 4.2.1. ScannerName

679 This element indicates the administratively assigned user-friendly name of the scanner. How the value for this element is
680 configured is implementation-specific, e.g., local console, Presentation Service (web access). If the Device Service has
681 only one hosted service, then the Device's <friendlyName> and *ScannerName* are recommended to have the same value.
682 However, if the Device contains several hosted services, the *ScannerName* identifies the scanner. A scan device can return
683 multiple version of this element to allow support for multiple localized languages by using the xml:lang attribute.

684 Allowed Values: any character string

685 4.2.2. ScannerInfo

686 This element is the descriptive information about this Scanner (example: "Out of courtesy for others, please scan only small
687 (1-5 page) jobs at this scanner"). How the value for this element is configured is implementation-specific, e.g., local
688 console, Presentation Service (web access). A scan device can return multiple version of this element to allow support for
689 multiple localized languages by using the xml:lang attribute.

690 Allowed Values: any character string

691 4.2.3. ScannerLocation

692 This element indicates the administratively assigned location of the device (for example, "Building 1"). How the value for
693 this element is configured is implementation-specific, e.g., local console, Presentation Service (web access). A scan device
694 can return multiple version of this element to allow support for multiple localized languages by using the xml:lang
695 attribute.

696 Allowed Values: any character string

4.3. The ScannerConfiguration Elements

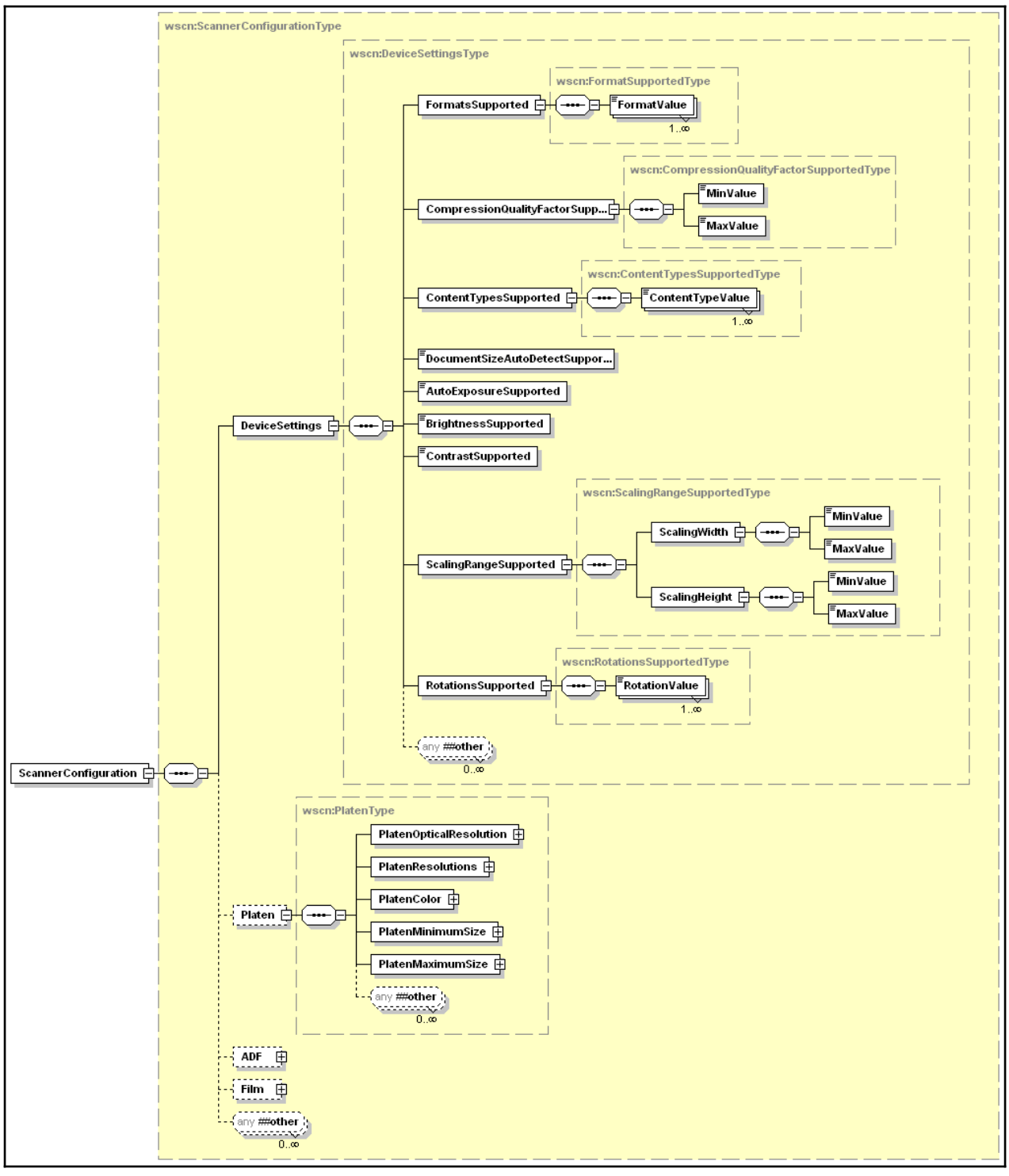


Figure 7 – ScannerConfiguration Elements – Part 1

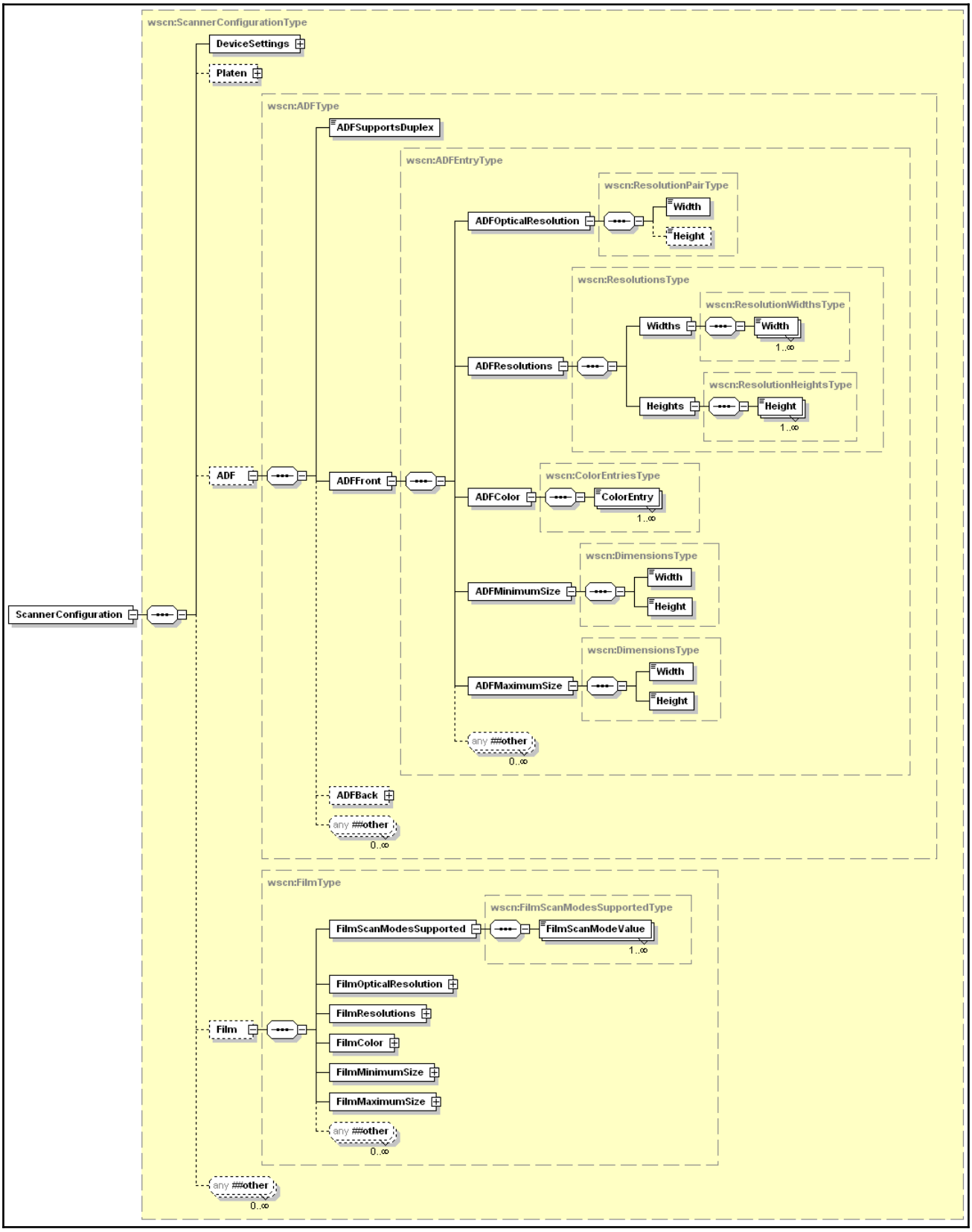


Figure 8 – ScannerConfiguration Elements – Part 2

All of the Scanner elements contained within the *ScannerConfiguration* element are described in detail in the following sections.

4.3.1. DeviceSettings

This section describes the basic capabilities of the device as a whole. This section contains the supported values for many of the imaging options settable in the *ScanTicket* for a scan operation. A CP can use the values returned in this section to create valid *ScanTicket* elements.

4.3.1.1. FormatsSupported

This element is a collection of keywords that describe the *Document* file formats supported by the scanner. Each file format describes a combination of file type and compression type.

4.3.1.1.1. FormatValue

This element indicates a single supported file format/compression type value.

Values:

- **`dib`** – Windows Device Independent Bitmap
- **`exif`** – Exchangeable Image File Format Version 2.x
- **`jbig`** – ISO/IEC 11544:1993 Standard - Coded representation of picture and audio information -- Progressive bi-level image compression
- **`jfif`** – JPEG File Interchange Format 1.x
- **`jpeg2k`** – JPEG 2000 standards based file format and compression
- **`pdf-a`** – PDF/A format: Standard based on ISO/CD 19005-1
- **`png`** – Portable Networks Graphics (PNG) format. This format only supports PNG compression type
- **`tiff-single-uncompressed`** – single page TIFF file with no compression type
- **`tiff-single-g4`** – single page TIFF file with g4 compression type
- **`tiff-single-g3mh`** – single page TIFF file with g3mh compression type
- **`tiff-single-jpeg-tn2`** – single page TIFF file with jpeg compression type as described in Technical Note 2.
- **`tiff-multi-uncompressed`** – multiple page TIFF file with no compression type
- **`tiff-multi-g4`** – multiple page TIFF file with g4 compression type
- **`tiff-multi-g3mh`** – multiple page TIFF file with g3mh compression type
- **`tiff-multi-jpeg-tn2`** – multiple page TIFF file with jpeg compression type as described in Technical Note 2.
- **`xps`** – XML Paper Specification

Vendors MAY extend the allowed values for this element
Vendors MAY subset the allowed values for this element

4.3.1.2. CompressionQualityFactorSupported

This element is an integer value used by any lossy compression type to determine the amount of acceptable image loss. The higher the requested fidelity the larger the resulting file size. A value of `100` means the device should use the least amount of compression it supports to produce the highest image quality. Currently JPEG compression is the only supported lossy compression type.

4.3.1.2.1. MinValue

This element indicates the minimum *CompressionQualityFactor* value supported by the scan device.

Values: `0` – `100`

4.3.1.2.2. MaxValue

This element indicates the maximum *CompressionQualityFactor* value supported by the scan device.

Values: `0` – `100`

4.3.1.3. ContentTypesSupported

This element is a collection of keywords that describe the different document content types supported by the scanner. Each *ContentType* describes the main characteristics of the original document.

4.3.1.3.1. ContentTypeValue

This element indicates a single supported *ContentType* value

Values:

`Auto`: The device will automatically detect the original type

`Text`: The original is mainly composed of distinct text that contrasts strongly with the background.

`Photo`: The original is mainly composed of photographic images, where shades change gradually and edges are not distinct.

`Halftone`: The original is mainly composed of halftoned images.

`Mixed`: A multipage document with characteristics of more than one specific *ContentType*.

Vendors MAY extend the allowed values for this element

Vendors MAY subset the allowed values for this element

4.3.1.4. DocumentSizeAutoDetectSupported

This element specifies whether the device can detect the size of the original media.

Values: `0`, `1`, `true`, `false`

4.3.1.5. AutoExposureSupported

This element specifies whether the device supports automatic adjustment of the various Exposure settings.

Values: `0`, `1`, `true`, `false`

4.3.1.6. BrightnessSupported

This element specifies whether the device supports user control of the scan brightness setting.

Values: `0`, `1`, `true`, `false`

4.3.1.7. ContrastSupported

This element specifies whether the device supports user control of the scan contrast setting.

Values: `0`, `1`, `true`, `false`

4.3.1.8. ScalingRangeSupported

This section describes the range of values this scan device supports for scaling the output document. A value of `100` specifies that no adjustments are made to the scanned image.

4.3.1.8.1. ScalingWidth

This element contains the minimum and maximum allowable values for scaling the width of the output document.

4.3.1.8.1.1. MinValue

This element indicates the minimum *ScalingWidth* value supported by the scan device.

Values: `1` – `1000`

4.3.1.8.1.2. MaxValue

This element indicates the maximum *ScalingWidth* value supported by the scan device.

Values: `1` – `1000`

4.3.1.8.2. ScalingHeight

This element contains the minimum and maximum allowable values for scaling the height of the output document.

4.3.1.8.2.1. MinValue

This element indicates the minimum *ScalingHeight* value supported by the scan device.

Values: `1` – `1000`

4.3.1.8.2.2. MaxValue

This element indicates the maximum *ScalingHeight* value supported by the scan device.

Values: `1 – 1000`

4.3.1.9. RotationsSupported

This element is a collection of keywords that describe the values supported by the scanner for rotation of each image of a scanned document. All requested *Rotation* values are applied to the scan data after acquisition and will be applied in the clockwise direction.

4.3.1.9.1. RotationValue

This element indicates a single supported *Rotation* value that tells the scanner the amount to rotate each image of a scanned document.

Values:

`0`
`90`
`180`
`270`

All WSD Scanners MUST support the `0` element value.
Vendors MAY subset the allowed values for this element
Vendors MAY extend the allowed values for this element

4.3.2. Platen

This element describes the capabilities of the flatbed scanning platen available on the scanner.

4.3.2.1. PlatenOpticalResolution

This element describes the maximum optical resolution at which the *Platen* is capable of scanning.
Resolution is specified in pixels per inch.

4.3.2.1.1. Width

This element indicates the maximum optical resolution the platen supports in the fast scan direction for this WidthxHeight pair.

Values: `1 – 2147483647`

4.3.2.1.2. Height

This element indicates the maximum optical resolution the platen supports in the slow scan direction for this WidthxHeight pair.

Values: `1 – 2147483647`

4.3.2.2. PlatenResolutions

This element is a collection that describes the resolutions at which the *Platen* is capable of scanning. *Width* and *Height* values are independent and most devices will support them being paired in any combination for a *ScanTicket*

Resolutions are specified in pixels per inch.

4.3.2.2.1. Widths

This element describes a list of *Width*(s) that the scanner is capable of scanning images.

4.3.2.2.1.1. Width

This element indicates a resolution the *Platen* supports in the fast scan direction.

Values: `1 – 2147483647`

4.3.2.2.2. Heights

This element describes a list of *Height*(s) that the scanner is capable of scanning images.

4.3.2.2.2.1. Height

This element indicates a resolution the *Platen* supports in the slow scan direction.

Values: `1` – `2147483647`

4.3.2.3. PlatenColor

This element is a collection of keywords that describe the *ColorProcessing* capabilities of the platen. The *PlatenColor* section contains the information needed to determine what type of color processing/acquisition the flatbed platen supports. The amount of information needed to describe each pixel depends on the specific *ColorEntry* keyword. Black and White images require only 1 bit per pixel whereas Grayscale and color images require significantly more information, the exact amount of information is driven by the color space and the technical capabilities of the scan device.

Another important aspect of the retuned scan data is the Photemetric Interpretation of the aquired data. All image data returned is required to be black on white, where black is represented by `0` and white is represented by `1`.

4.3.2.3.1. ColorEntry

This element describes a single color processing mode supported by the *Platen*. Each keyword describes the data type/encoding, bit depth, and bits per channel. The table below illustrates how the keywords map to the color processing properties.

ColorEntry Keyword	Pixel Bit Depth	Bits Per Channel
`BlackandWhite1`	1	1
`Grayscale4`	4	{4}
`Grayscale8`	8	{8}
`Grayscale16`	16	{16}
`RGB24`	24	{8,8,8}
`RGB48`	48	{16,16,16}
`RGBa32`	32	{8,8,8,8}
`RGBa64`	64	{16,16,16,16}

Table 1 – Color Processing Elements

Values:

- **`BlackAndWhite1`** – Black and White images, 1 bit per pixel and a single channel
- **`Grayscale4`** – Grayscale images, 4 bits per pixel and a single channel
- **`Grayscale8`** – Grayscale images, 8 bits per pixel and a single channel
- **`Grayscale16`** – Grayscale images, 16 bits per pixel and a single channel
- **`RGB24`** – RBG encoded color images, 24 bits per pixel divided between 3 channels of 8 bits each
- **`RGB48`** – RBG encoded color images, 48 bits per pixel divided between 3 channels of 16 bits each
- **`RGBa32`** – RBG encoded color images with an alpha channel, 32 bits per pixel divided between 4 channels of 8 bits each
- **`RGBa64`** – RBG encoded color images with an alpha channel, 64 bits per pixel divided between 4 channels of 16 bits each

Vendors MAY extend the allowed values for this element
Vendors MAY subset the allowed values for this element

4.3.2.4. PlatenMinimumSize

This element specifies the smallest size original that can be scanned on the *Platen*. All media dimensions are measured in 1/1000^{ths} of an inch.

4.3.2.4.1. Width

This element indicates the minimum size of media the *Platen* supports in the fast scan direction.

Values: `1` – `2147483647`

4.3.2.4.2. Height

This element indicates the minimum size of media the *Platen* supports in the slow scan direction.

Values: `1` – `2147483647`

4.3.2.5. PlatenMaximumSize

This element specifies the largest size original that can be scanned on the *Platen*. This element is made up of a pair of elements that describe the maximum media size. All media dimensions are measured in 1/1000^{ths} of an inch.

4.3.2.5.1. Width

This element indicates the maximum size of media the *Platen* supports in the fast scan direction.

Values: `1` – `2147483647`

4.3.2.5.2. Height

This element indicates the maximum size of media the *Platen* supports in the slow scan direction.

Values: `1` – `2147483647`

4.3.3. ADF

This element describes the capabilities of the Automatic Document Feeder (ADF) attached to the scanner.

4.3.3.1. ADFSupportsDuplex

This element specifies whether the attached *ADF* supports scanning both sides of the media.

Allowed Values: `0`, `1`, `true`, `false`

4.3.3.2. ADFFront

This section describes the capabilities of the front side of the Automatic Document Feeder attached to the scanner.

4.3.3.2.1. ADFOpticalResolution

This element describes the maximum optical resolution at which the *ADF* Front side is capable of scanning. See Section 4.3.2.1. for a detailed description of this data element.

4.3.3.2.2. ADFResolutions

This element is a collection that describes the resolutions at which the *ADF* Front side is capable of scanning. See Section 4.3.2.2. for a detailed description of this data element.

4.3.3.2.3. ADFColor

This element is a collection of keywords that describe the *ColorProcessing* capabilities of the *ADF* Front side. See Section 4.3.2.2. for a detailed description of this data element.

4.3.3.2.4. ADFMinimumSize

This element specifies the smallest size original that can be scanned with the *ADF* Front side. See Section 4.3.2.4. for a detailed description of this data element.

4.3.3.2.5. ADFMaximumSize

This element specifies the largest size original that can be scanned with the *ADF* Front side. See Section 4.3.2.5. for a detailed description of this data element.

4.3.3.3. ADFBack

This optional element describes the capabilities of the back side of a duplex Automatic Document Feeder attached to the scanner. If the scanner has a simplex ADF installed this element will not be present in the *ScannerConfiguration*.

4.3.3.3.1. ADFOpticalResolution

This element describes the maximum optical resolution at which the *ADF* Back side is capable of scanning. See Section 4.3.2.1. for a detailed description of this data element.

4.3.3.3.2. ADFResolutions

This element is a collection that describes the resolutions at which the *ADF* Back side is capable of scanning. See Section 4.3.2.2. for a detailed description of this data element.

4.3.3.3.3. ADFColor

This element is a collection of keywords that describe the *ColorProcessing* capabilities of the *ADF* Back side. See Section 4.3.2.2. for a detailed description of this data element.

4.3.3.3.4. ADFMinimumSize

This element specifies the smallest size original that can be scanned with the *ADF* Back side. See Section 4.3.2.4. for a detailed description of this data element.

4.3.3.3.5. ADFMaximumSize

This element specifies the largest size original that can be scanned with the *ADF* Back side. See Section 4.3.2.5. for a detailed description of this data element.

4.3.4. Film

This element describes the capabilities of the Film scanning option attached to the scanner.

4.3.4.1. FilmScanModesSupported

This element is a collection of keywords that describe the values supported for film exposure types by the *Film* scanning option.

4.3.4.1.1. FilmScanModeValue

This element indicates a single supported *FilmScanMode* value that specifies the exposure type of the film to be scanned.

Values:

- **`NotApplicable`** – Only valid in a *DefaultScanTicket*. The default scan source is no longer the *Film* option, thus the *FilmScanMode* is no longer an applicable value for the *DefaultScanTicket*
- **`ColorSlideFilm`** – Film images are in the normal color space captured
- **`ColorNegativeFilm`** – Film images are negatives of the normal color space captured.
- **`BlackandWhiteNegativeFilm`** – Film images are black and white negatives of the images captured

Vendors MAY extend the allowed values for this element
Vendors MAY subset the allowed values for this element

4.3.4.2. FilmOpticalResolution

This element describes the maximum optical resolution at which the *Film* scanning option is capable of scanning. See Section 4.3.2.1. for a detailed description of this data element.

4.3.4.3. FilmResolutions

This element is a collection that describes the resolutions at which the *Film* scanning option is capable of scanning. See Section 4.3.2.2. for a detailed description of this data element.

4.3.4.4. FilmColor

This element is a collection of keywords that describe the *ColorProcessing* capabilities of the *Film* scanning option. See Section 4.3.2.2. for a detailed description of this data element.

4.3.4.5. FilmMinimumSize

This element specifies the smallest size original that can be scanned with the *Film* scanning option. See Section 4.3.2.4. for a detailed description of this data element.

4.3.4.6. FilmMaximumSize

This element specifies the largest size original that can be scanned with the *Film* scanning option. See Section 4.3.2.5. for a detailed description of this data element.

4.4. The ScannerStatus Elements

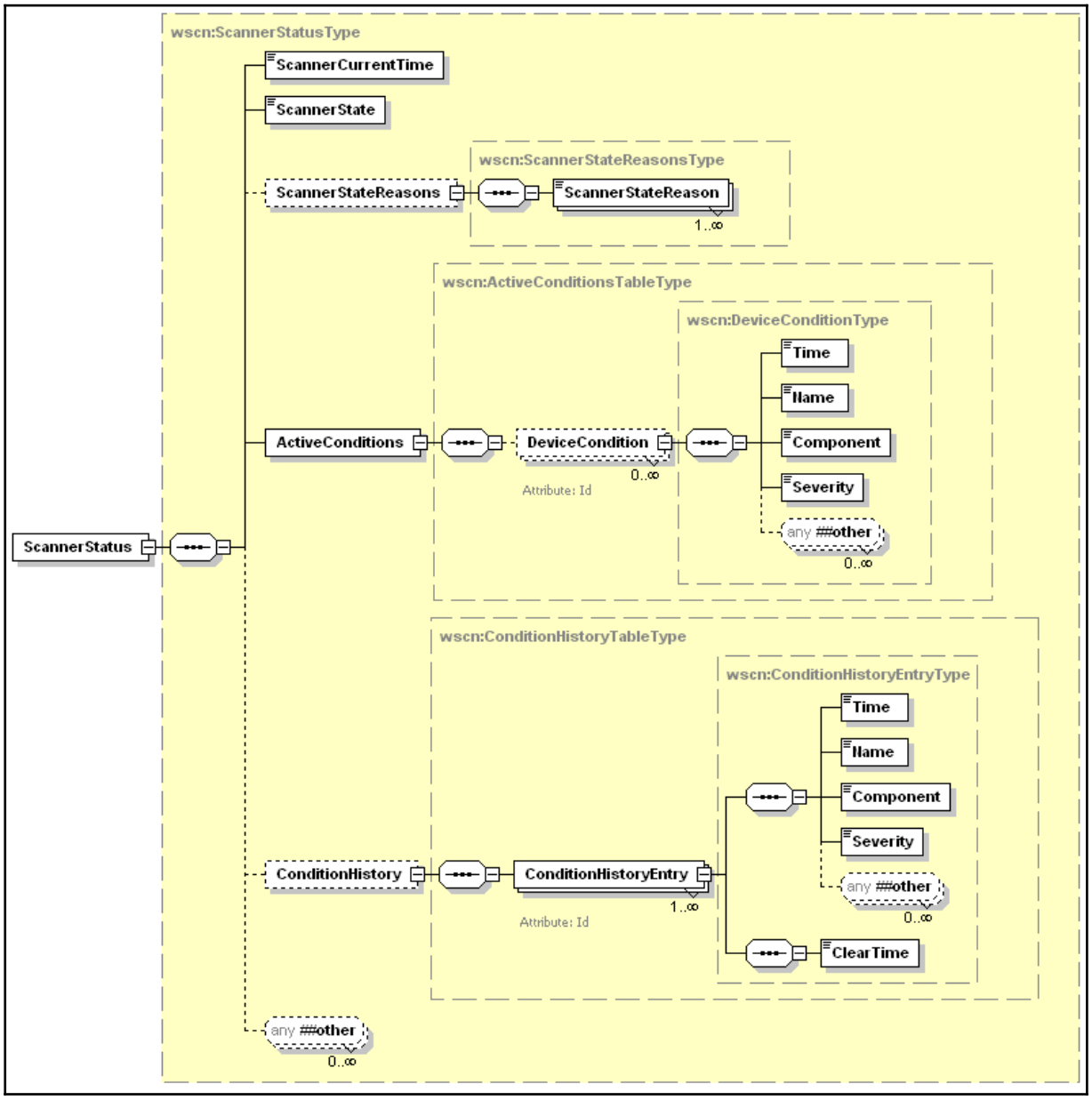


Figure 9 - ScannerStatus Elements

All of the Scanner elements contained within the *ScannerStatus* element are described in detail in the following sections.

4.4.1. ScannerCurrentTime

This element indicates the current date and time according the scanner's internal clock. This is not required to be a realtime clock, it can start at zero (`0001-01-01T00:00:00Z`) and count up when the device is powered on. All times are based on the time at startup so duration and relative time can be calculated by reading the *ScannerCurrentTime* and comparing it to the previous time value.

Values: Any valid dateTime value

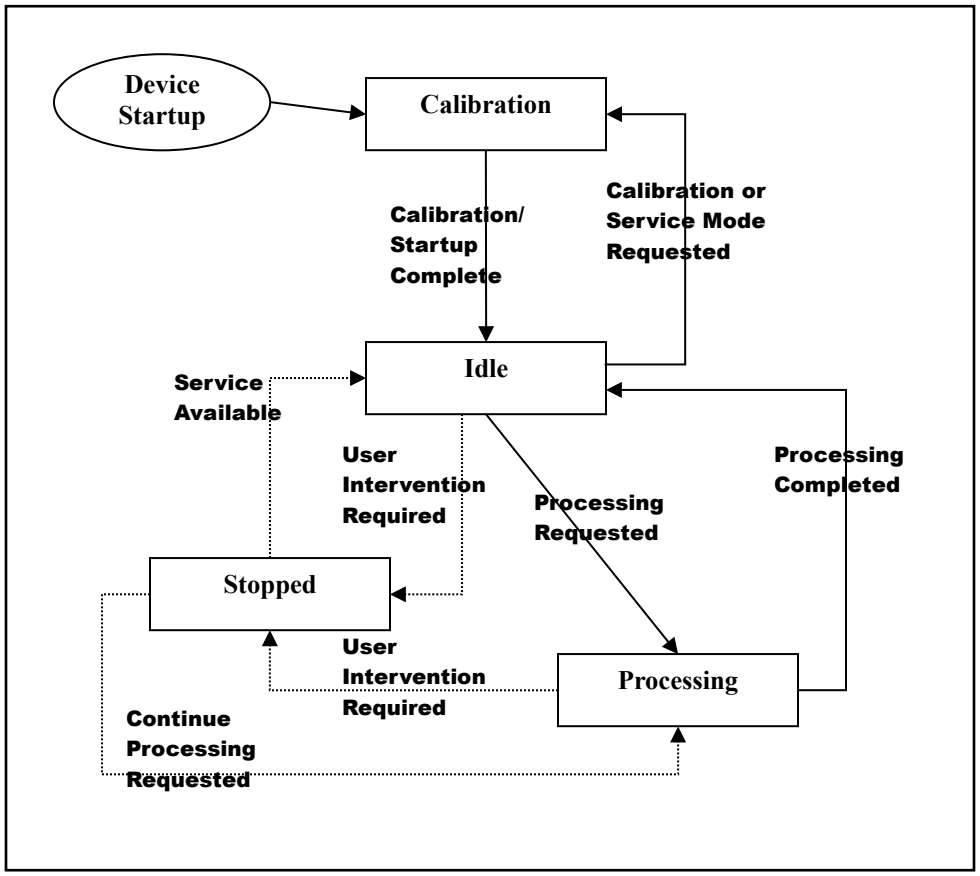
4.4.2. ScannerState

This element identifies the current state of scanning portion of the device.

Values:

958 **Idle** - Scanner is available and can start processing a new job.
959 **Processing** – The scanner is currently processing jobs.
960 **Stopped** - No jobs can be processed and intervention is needed.

961 Vendors MAY extend the allowed values for this element
962 Vendors MAY subset the allowed values for this element
963
964



965 **Figure 10 - Scanner State Diagram**

966

967 4.4.3. ScannerStateReasons

968 This element is a collection that describes all of the additional information about why the Scanner is in its current state.

969 4.4.3.1. ScannerStateReason

970 This element indicates a single additional piece of information about why the scanner is in its current state

971 Note: some of these reasons describe state of the scanner that cannot be entered on the basis of the currently defined WSD
972 operations set. For example the scanner can be `Paused`; there is no PauseScanner operation. The reason these states are
973 presented is because some other protocol (or console action) can have caused the scanner to enter that state.

974 Values:
975 **AttentionRequired** - User intervention is required before the device can continue.
976 **Calibrating** – The scan device is calibrating its internal components in preparation to acquire images
977 **CoverOpen** - One or more covers on the device are open.
978 **InputTrayEmpty** – The ADF input has no media
979 **InterlockOpen**
980 **InternalStorageFull**
981 **LampError** – The scanner lamp is failing and image acquisition can not proceed

LampWarming – The scanner lamp is warming in preparation to acquire images
MediaJam – Media is jammed in one of the input options causing image acquisition to fail
MultipleFeedError – The *ADF* option picked more than one piece of media
None - Indicates that there are no current state reasons
Paused - Someone has paused the scanner and the *ScannerState* is `Stopped`. In this state, a scanner will not produce Scanned output.

Vendors MUST support the values that represent conditions that are detectable in their implementation. Therefore, vendors MAY subset allowed values if specific *ScannerStateReasons* are undetectable in their implementation.
Vendors MAY extend allowed values. However, scanner vendors need to understand the implications of extending this list on a CP. The CP usually localizes the *ScannerStateReasons* value (as with other string variable values) to the human language of the user. However, such a scanner vendor extension value will not be recognized by the CP. As a Fallback presentation, the CP MAY display the value received as is, which should be in English and therefore, might not be understandable by the user. Alternatively, the vendor might use the general *ScannerStateReasons* value: `AttentionRequired` and then explain the problem on the scanner console which the user would see when they are at the scanner.

4.4.4. ActiveConditions

This element is a collection of detailed descriptions that describe in detail all of the currently active conditions/errors on the device. These conditions could vary in severity from `Informational` to `Critical`.

4.4.4.1. DeviceCondition

This is an element that describes the details about one of the currently active conditions.

4.4.4.1.1. Id

This attribute specifies the unique *Id* of the current *DeviceCondition* entry or *ConditionHistoryEntry* entry. This *Id* will be used by the CP to determine if an error condition is new, or has gone away. This *Id* is an integer value and MUST not be re-used over a reasonable duration of time to allow CPs to keep track of individual *DeviceCondition*(s).

REQUIRED attribute.
Values: `1 – 2147483647`

4.4.4.1.2. Time

This element specifies the time a condition occurred. This time is according to the internal clock of the scanner.

Values: Any valid dateTime value

4.4.4.1.3. Name

This element specifies the name of the current *DeviceConditon* entry. This name describes the type of the current error condition. There are different error names for each component.

Values:

Calibrating
CoverOpen
InputTrayEmpty
InterlockOpen
InternalStorageFull
MediaJam
LampError
LampWarming
MultipleFeedError

Vendors MAY extend the allowed values for this element.
Vendors MAY subset the allowed values for this element

4.4.4.1.4. Component

This element specifies the component that is described by the current *DeviceCondition* entry.

Values:

ADF
Film
MediaPath
Platen

Vendors MAY extend the allowed values for this element.
Vendors MAY subset the allowed values for this element

4.4.4.1.5. Severity

This element specifies the level of severity of the current *DeviceCondition* entry. The scanner determines the severity level assigned to error condition.

Values:

Informational – This condition is purely for user information and has no noticeable affect on the image acquisition process
Warning – This condition is not affecting processing currently, but if not attended to it could become `Critical`
Critical – The device can not continue processing until this condition is resolved

Vendors MAY extend the allowed values for this element.
Vendors MAY subset the allowed values for this element

4.4.5. ConditionHistory

This element is a collection of detailed descriptions that describe a set of the most recent conditions/errors on the device. These conditions could vary in severity from `Informational` to `Critical`.

4.4.5.1. ConditionHistoryEntry

This is an element that describes the details about one of the past conditions.

4.4.5.1.1. Id

This data element is described in Section 4.4.4.1.1.

4.4.5.1.2. Time

This data element is described in Section 4.4.4.1.2.

4.4.5.1.3. Name

This data element is described in Section 4.4.4.1.3.

4.4.5.1.4. Component

This data element is described in Section. 4.4.4.1.4.

4.4.5.1.5. Severity

This data element is described in Section 4.4.4.1.5.

4.4.5.1.6. ClearTime

This element specifies the time a condition was cleared. This time is according to the internal clock of the scanner.

Values: Any valid dateTime value

4.5. The Job's Elements

This section defines the elements of the WSD Scan Job object. Figure 11 shows all the elements for the Job and their grouping. Jobs may contain one or more Documents. As indicated in the figure below the processing instructions for both the Job and the Document are done at the Job level. The Document object is covered in detail in section 4.6..

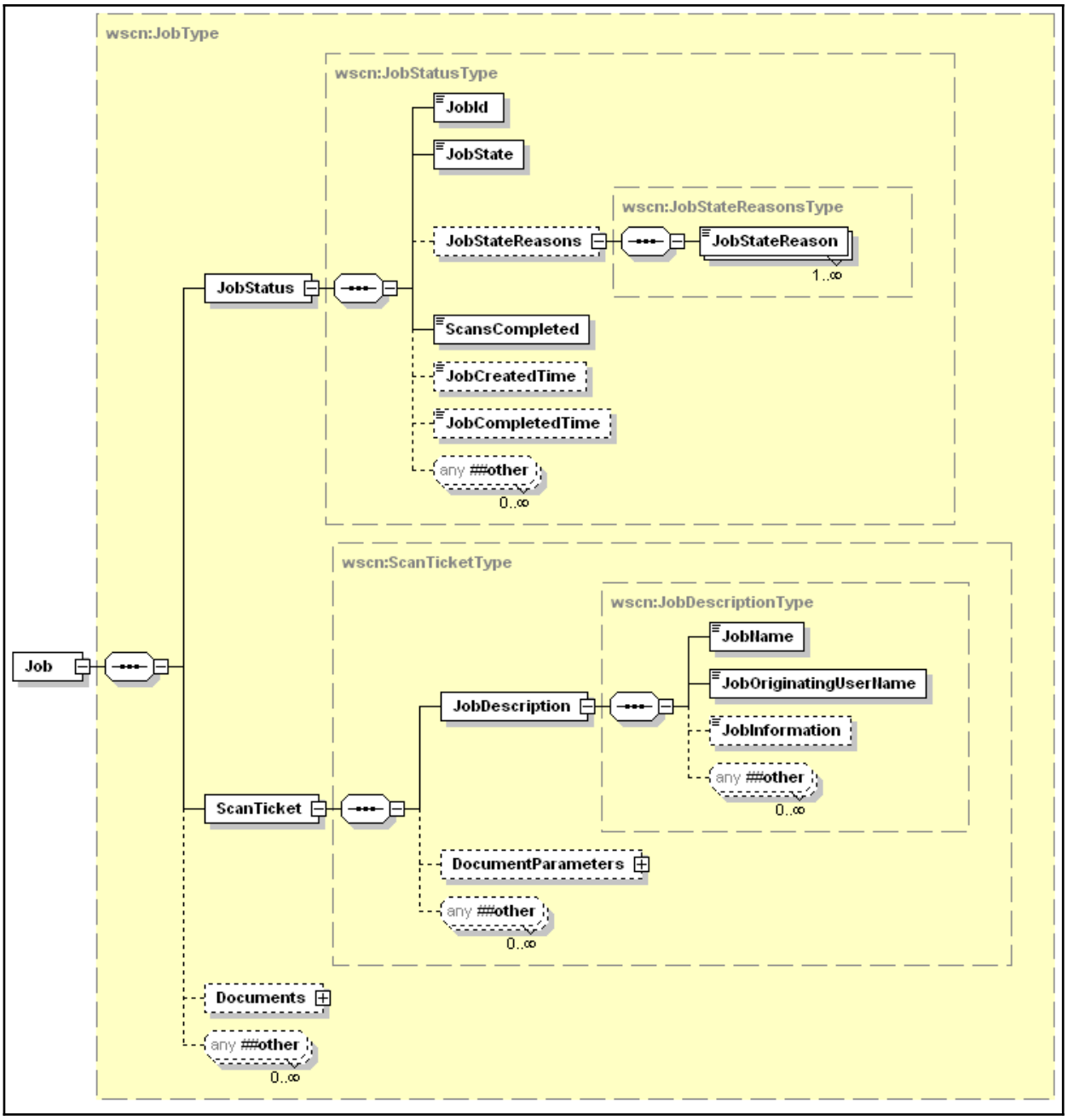


Figure 11 - Job Elements (Part 1)

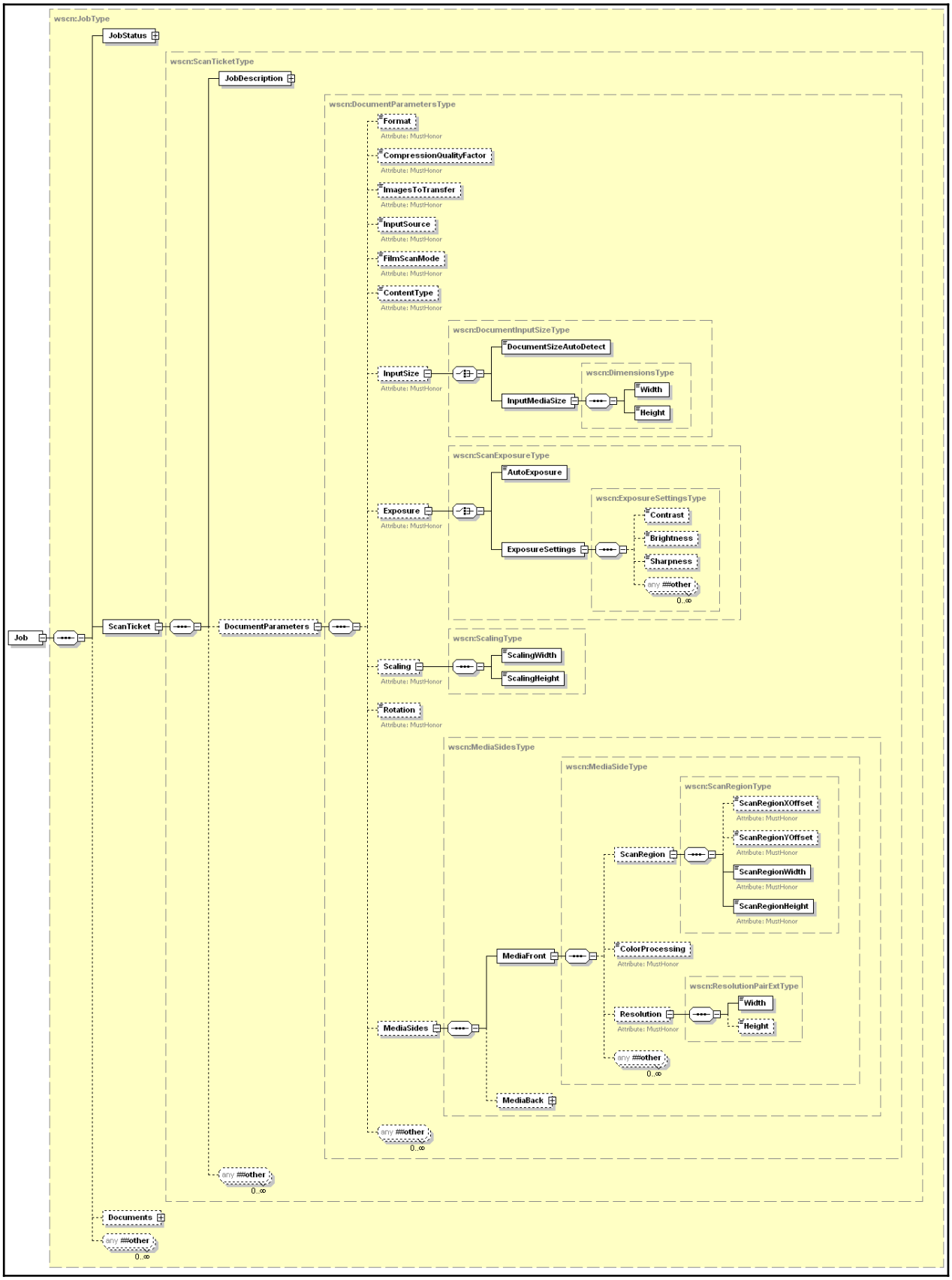


Figure 12 - Job Elements (Part 2)

All of the Scanner elements contained within a scan job are described in detail in the following sections.

The *DocumentParameters* elements of a Job are populated from corresponding elements explicitly supplied in a Job creation request by a CP. If the CP does not supply a parameter or processing element it indicates that the CP wants the scanner to use its default value. How the scanner's default and allowed values for processing elements are set is implementation-specific, e.g., local console, Presentation Service (web access).

The *JobDescription* elements are populated from corresponding elements supplied in a Job creation request. These values may be changed by the scanner if more reliable information can be obtained. The *JobStatus* elements are maintained by automata. The values may be indirectly affected by a CP operation (e.g. **CancelJob**).

4.5.1. JobStatus

This section describes all of the elements that make up the *JobStatus* type in the scan job.

4.5.1.1. JobId

This element uniquely identifies a Job within a scanner. This number is not globally unique.

Values: `1 – 2147483647`

4.5.1.2. JobState

This element specifies the current state of the Job. When contained in the **JobEndStateEvent** or a *JobHistory* entry it represents the completion state of the Job.

Values:

- **`Aborted`** – The Job was aborted by the system.
- **`Canceled`** – The Job was canceled either by a CP using the **CancelJob** operation or by means outside the scope of WSD
- **`Completed`** – The Job is finished processing and all of the image data has been sent to the client.
- **`Creating`** – The Job is being initialized.
- **`Pending`** – The Job has been initialized and is waiting to be processed.
- **`Pending-Held`** – The Job is waiting to be processed but is unavailable for scheduling. This state can only be reached by methods outside the scope of WSD.
- **`Processing`** – The Job data is being digitized, transformed, or transferred.
- **`Started`** – The scan device has begun processing the Job. This is a transient state, and will usually be seen only within a **JobStatusEvent**.
- **`Terminating`** – The Job was canceled either by a CP using the **CancelJob** operation or aborted by means outside the scope of WSD.

Vendors MAY extend the allowed values for this element.
Vendors MAY subset the allowed values for this element.

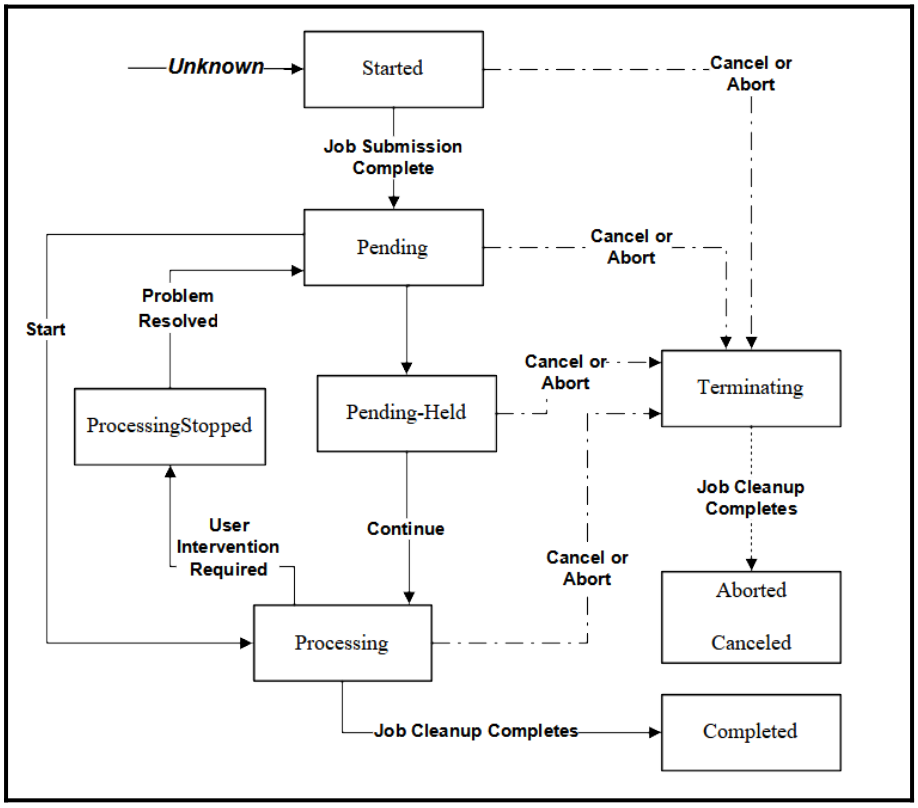


Figure 13 - Job States

4.5.1.3. JobStateReasons

This element is a collection that describes all of the additional information about why the Job is in its current state.

4.5.1.3.1. JobStateReason

This element is a single piece of additional information about why the Job is in its current state

Values:

- **`InvalidScanTicket`** – The Job was rejected because the ScanTicket could not be processed.
- **`DocumentFormatError`** – The requested Document format is not supported by the scan service
- **`ImageTransferError`** – The data transfer of an image in the job failed. If this occurs the job is aborted.
- **`JobCanceledAtDevice`** – The current scan job was canceled at the device front panel
- **`JobCompletedSuccessfully`** – The Job is complete without any warnings or errors
- **`JobCompletedWithErrors`** – The Job completed with at least one error.
- **`JobCompletedWithWarnings`** – The Job completed with at least one warning. The Job data is expected to be successfully transferred. The warning may indicate that alterations to the job ticket occurred in order to process the Job.
- **`JobScanning`** – The scanner is digitizing the Job data.
- **`JobScanningAndTransferring`** – The scanner is digitizing the Job data and data is being transferred to the client.
- **`JobTimedOut`** – The Job completed after no **RetrieveImage** operations followed the **CreateScanJob** operation in a timely fashion.
- **`JobTransferring`** – The Job data is being transferred to the client.
- **`None`** – The Job has no additional information on the Job State.
- **`ScannerStopped`** – The scan device is stopped due to an active condition and the job can not continue until the condition is corrected

Vendors MUST support the values that represent conditions that are detectable in their implementation. Therefore, vendors MAY subset allowed values if specific *JobStateReasons* are undetectable in their implementation.
Vendors MAY extend allowed values. However, scanner vendors need to understand the implications of extending this list on a CP. The CP usually localizes the *JobStateReason* value (as with other string variable values) to the human language of the user. However, such a scanner vendor extension value will not be recognized by the CP. As a Fallback presentation, the CP MAY display the value received as is, which should be in English and therefore, might not be understandable by the user.

4.5.1.4. JobCreatedTime

This element contains the local time when the job was created. Job creation is the point in time where the job is submitted to the system.

Values: Any valid dateTime value

4.5.1.5. JobCompletedTime

This element specifies the scanner time at completion of the scan job. Completion is when all processing has completed, either because of successful completion of scanning and document transfer or because a fatal error was encountered.

Values: Any valid dateTime value

4.5.1.6. ScansCompleted

This element specifies the number of images scanned. A sheet of media may be scanned multiple times. In duplex scanning each side of the sheet is scanned, generating two scans in the *ScansCompleted* count.

(Note: This information may not be known until the scanner has completed processing the Job. The scanner MUST update this element when more exact information is available)

Values: `0 – 2147483647`

4.5.2. ScanTicket

This section of the schema defines all the information that pertains to the description and processing parameters of the currently identified Job. This is the *ScanTicket* definition that was supplied with the **CreateScanJob** operation which started the Job.

4.5.2.1. JobDescription

This section describes all of the elements that make up the *JobDescription* type in the scan job.

4.5.2.1.1. JobName

This element is the user-friendly name of the job supplied by the CP. It is RECOMMENDED that the CP supply a value to help a user easily distinguish between the jobs that he/she has submitted.

Values: any character string

4.5.2.1.2. JobOriginatingUserName

This element is the name of the user that submitted the job. Either supplied by the CP or by the security infrastructure, if any. It is RECOMMENDED that the CP supply a value to help a user easily distinguish between the jobs that he/she has submitted and jobs that others have submitted.

Values: any character string

4.5.2.1.3. JobInformation

This element describes the intended use of the job. This value is useful if the job ticket used to create this job will be reused.

Values: any character string

4.5.2.2. DocumentParameters

This section describes all of the elements that make up the *DocumentParameters* type in the scan job. These parameters apply to all documents within the current scan job.

4.5.2.2.1. Format

This element specifies the document format in which the image data should be rendered. See section 4.3.1.1.1. for a list of values.

Vendors MAY extend the allowed values for this element.
Vendors MAY subset the allowed values for this element.

4.5.2.2.2. CompressionQualityFactor

This element specifies an integer value used by lossy compression types to determine the amount of acceptable image loss. See section 4.3.1.2. for a list of values. If the requested compression type is lossless then this element can be ignored if it is specified and the service should use a value of `100`.

Values: `0 – 100`
Vendors MAY subset the allowed values for this element

4.5.2.2.3. ImagesToTransfer

This element tells the scan device how many images to scan for the current job. This is useful if the document feeder could contain more pages of media than the current job. A value of `0` means scan as many pages as are currently available for the selected *InputSource*. If the *InputSource* is the `Platen` or `Film` then a value of `0` means a single image acquisition. If the *InputSource* is the `ADF` or `ADFDuplex` then a value of `0` means acquire images from the feeder until it is empty.

When acquiring images from `ADFDuplex` each side of the media represents a single image. To get both sides of the media the *ImagesToTransfer* value needs to be an even value of at least `2`, sending an odd value, such as `1`, would only acquire the front of the last sheet of media and the device should discard any data aquired from the back side of the media.

Values: `0 – 2147483647`

4.5.2.2.4. InputSource

This element specifies the source of the original document:

Values:

`ADF:` Document being scanned is being delivered by a document feeding device scanning only the front side.
`ADFDuplex:` Document being scanned is being delivered by a document feeding device scanning both sides.
`Film:` Document is to be scanned using the Film scanning option.
`Platen:` Document is to be scanned from the platen

Vendors MAY extend the allowed values for this element.
Vendors MAY subset the allowed values for this element.

4.5.2.2.5. FilmScanMode

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 `Film`. See section 4.3.4.1.1. for a list of values for this element.

4.5.2.2.6. ContentType

This element specifies the main characteristics of the original document. See section 4.3.1.3.1. for a list of values for this element.

4.5.2.2.7. InputSize

This section describes the scan-able media size and how to detect it. This section is defined as a choice entry, which means only one of the sub-elements is valid at once. A CP can either ask for auto-size detection or specify a size.

4.5.2.2.7.1. DocumentSizeAutoDetect

This element indicates the scanner will do its best to determine the size of the original scan media. When/how this occurs is completely device dependent. If this element is specified along with a *ScanRegion* element, the Scan Region will be ignored if it falls outside of the media size detected by the device.

Values: `0`, `1`, `true`, `false`

4.5.2.2.7.2. InputMediaSize

This element describes the size of the media to be scanned for the current job. This element is made up of a pair of elements that describe the Width and Height of the media.

4.5.2.2.7.2.1. Width

This element indicates the width of the original media in the fast scan direction.

Values: `1` – `2147483647`

4.5.2.2.7.2.2. Height

This element indicates the height of the original media in the slow scan direction.

Values: `1` – `2147483647`

4.5.2.2.8. Exposure

This section indicates whether the scan service should automatically employ image processing techniques to reduce the background of the document to a white image or adjust the image processing by the values supplied. This section is defined as a choice entry, which means only one of the sub-elements is valid at once. A CP can either ask for auto-exposure adjustments or specify specific image processing adjustment values.

4.5.2.2.8.1. AutoExposure

This element indicates whether the scan service should automatically employ image processing techniques to reduce the background of the document to a white image. When set to `1` or `true` automatic background reduction will be performed on the original document.

Values: `0`, `1`, `true`, `false`

4.5.2.2.8.2. ExposureSettings

This element contains individual adjustment values which the scan service should apply to the image data after acquisition.

4.5.2.2.8.2.1. Contrast

This element indicates the relative amount to enhance or reduce the contrast of the scanned image. A value of `0` means to make no adjustments to the scanned contrast.

Values: `-1000 – 1000`
All WSD scanners MUST support the all values between and including `-1000` and `1000`, internally mapping as needed these values to the actual brightness/contrast values supported by the device.
Vendors MAY subset the allowed values for this element.

4.5.2.2.8.2.2. Brightness

This element indicates the relative amount to enhance or reduce the brightness of the scanned image. A value of `0` means to make no adjustments to the scanned brightness.

Values: `-1000 – 1000`
All WSD scanners MUST support the all values between and including `-1000` and `1000`, internally mapping as needed these values to the actual brightness/contrast values supported by the device.
Vendors MAY subset the allowed values for this element.

4.5.2.2.8.2.3. Sharpness

This element indicates the relative amount to enhance or reduce the sharpness of the scanned image. A value of `0` means to make no adjustments to the scanned sharpness.

Values: -100 – 100
All WSD scanners MUST support at least the 0 value.
Vendors MAY subset the allowed values for this element.

4.5.2.2.9. Scaling

This section specifies the scaling for both the slow and fast scan directions.

Both values must be specified. Isomorphic scaling is accomplished by supplying the same values for both width and height.

4.5.2.2.9.1. ScalingWidth

This element indicates the scaling factor to apply 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: 1 – 1000
All WSD scanners MUST support at least the 100 value.
Vendors MAY subset the allowed values for this element.

4.5.2.2.9.2. ScalingHeight

This element indicates the scaling factor to apply in the slow scan direction. A value of 100 specifies that no adjustments are made to the scanned image. Scaling is expressed in 1 percent increments.

Values: 1 – 1000
All WSD scanners MUST support at least the 100 value.
Vendors MAY subset the allowed values for this element.

4.5.2.2.10. Rotation

This element indicates the amount to rotate each image of a scanned document. See section 4.3.1.9.1. for a list of values for this element. All devices must support a value of 0 (no rotation) for this element.

4.5.2.2.11. MediaSides

This element contains the parameters which are unique to each physical side of the media to be scanned. Many duplex capable scanners allow setting different scan regions, color processing and resolutions for each physical side of the media to be scanned. This element contains separate data for the Front side and Back side of the physical media. Every scan job can have parameters for the media front, however parameters for the media back are only valid if the *InputSource* specified equals ADFDuplex. If *InputSource* is ADFDuplex and the *MediaBack* element is missing, any parameters specified in *MediaFront* will be applied to the back side scanning as well.

4.5.2.2.11.1. MediaFront

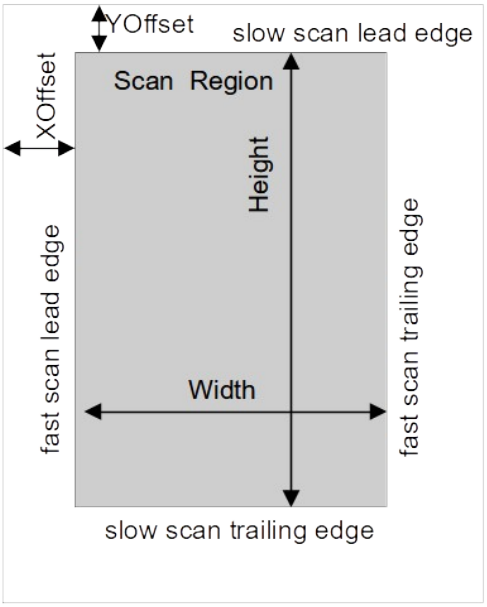
This element contains all of the parameters specific to scanning the Front side of the physical media.

4.5.2.2.11.1.1. ScanRegion

This element specifies the area within the input document boundaries to scan. All *ScanRegion* values are in 1/1000ths of an inch.

If *ScanRegion* is not specified, the device should use 0 as the offsets and the width and height of the *InputMediaSize*, if given. If *ScanRegion* is not specified and *InputMediaSize* is not specified or cannot be determined by the device, the implementation is up to the hardware vendor.

The step size for all *ScanRegion* values in 1/1000th of an inch. If the requested scan region would fall completely outside of the supported acquisition area of the scanner the scan operation should be rejected. The scanner can also adjust the scan region requested if it can't fulfill the exact dimensions. Any changes to the scan region should be reported in the *DocumentFinalParameters* elements in the scan job.



1299

1300 *ScanRegionWidth* corresponds to the fast scan direction and *ScanRegionHeight* corresponds to the slow scan direction.

1301 *ScanRegionXOffset* + *ScanRegionWidth* must be equal to or less than *InputSize* width.

1302 *ScanRegionYOffset* + *ScanRegionLength* must be equal to or less than *InputSize* length

1303 #### 4.5.2.2.11.1.1.1. ScanRegionXOffset

1304 This element indicates how far from the fast scan lead edge to begin the scanning for this document. This value must be
1305 less than the *InputSize* width.

1306 Values: `0` – *InputSize* width

1307 #### 4.5.2.2.11.1.1.2. ScanRegionYOffset

1308 This element indicates how far from the slow scan lead edge to begin the scanning for this document. This value must be
1309 less than the *InputSize* height.

1310 Values: `0` – *InputSize* height

1311 #### 4.5.2.2.11.1.1.3. ScanRegionWidth

1312 This element indicates how far from the fast scan lead edge plus the *ScanRegionXOffset* to end the scanning for this
1313 document. This value plus *ScanRegionXOffset* must be less than the *InputSize* width.

1314 Values: `1` – *InputSize* width

1315 #### 4.5.2.2.11.1.1.4. ScanRegionHeight

1316 This element indicates how far from the slow scan lead edge plus the *ScanRegionYOffset* to end the scanning for this
1317 document. This value plus *ScanRegionYOffset* must be less than the *InputSize* height.

1318 Values: `1` – *InputSize* height

1319 ### 4.5.2.2.11.1.2. ColorProcessing

1320 This element contains the information needed to determine how color should be handled for the scanned image(s). See
1321 section 4.3.2.3.1. for a detailed description of this data element.

4.5.2.2.11.1.3. Resolution

This element specifies the resolution at which to capture the image. This element contains a single WidthxHeight pair that describes the desired capture resolution. If the *Height* element is missing, it is assumed this resolution pair (WidthxHeight) is a square resolution (i.e. 300x300) based on the *Width*.

Resolution is specified in pixels per inch.

4.5.2.2.11.1.3.1. Width

This element indicates the resolution to capture the image in the fast scan direction for this WidthxHeight pair.

4.5.2.2.11.1.3.2. Height

This element indicates the resolution to capture the image in the slow scan direction for this WidthxHeight pair.

4.5.2.2.11.2. MediaBack

This optional element contains all of the parameters specific to scanning the Back side of the physical media. These parameters are only valid when the scanner supports duplex scanning and the current scan job specifies `ADFDuplex` as the input source. See Section 4.5.2.2.11.1. for a detailed description of this data element.

If the current scan job specifies `ADFDuplex` as the input source and this element is not present, all parameters in the *MediaFront* element will be applied to the backside of the duplex scan as well.

4.6. The Document's Elements

This section defines the elements of the WSD Document Object. Figure 14 shows all the elements for the Document and their grouping. Jobs contain one or more Documents.

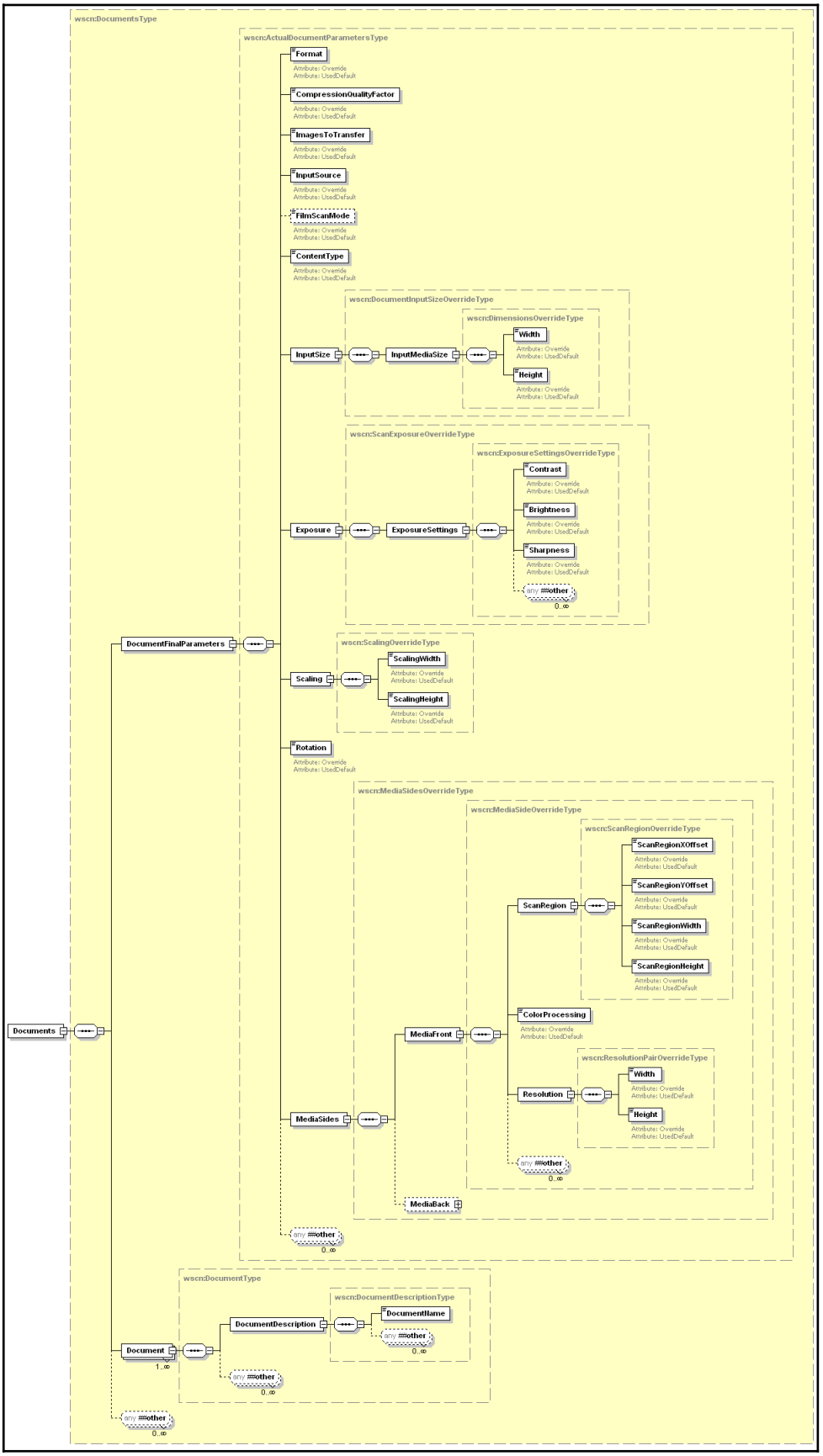


Figure 14 – Documents & Document Elements

All of the Scanner elements contained within a scan document are described in detail in the following sections.

The *DocumentDescription* elements are populated from corresponding elements supplied in a Job creation request. These values may be changed by the scanner if more reliable information can be obtained.

4.6.1. Documents

This element contains the actual scan characteristics used during image acquisition and also a collection of all the Document elements contained in the scan job.

4.6.1.1. DocumentFinalParameters

This section of the schema contains the actual *DocumentParameters* used by the scan device for image acquisition. These values may differ from the values requested in the job *ScanTicket* for any number of reasons. Each parameter is represented in this section, and must be filled in once the values are known. Each data element in this section also has two possible attributes associated with it. These attributes can be used by the client to determine whether the data value used by the scan device was the value sent in the *ScanTicket*, the scan device default value, or an override of the value requested by the *ScanTicket*.

4.6.1.1.1. Override

This attribute specifies that the current element was overridden by the scan device when the image data was aquired. The value of the element this attribute is associated with is the actual value used by the scan device during image acquisition.

OPTIONAL attribute.
Values: `0`, `1`, `true`, `false`

4.6.1.1.2. UsedDefault

This attribute specifies that the current element was not specified in the *ScanTicket* and the scan device used the default value when the image data was aquired. The value of the element this attribute is associated with is the actual value used by the scan device during image acquisition.

OPTIONAL attribute.
Values: `0`, `1`, `true`, `false`

4.6.1.2. Document

This is an element that describes the details about one of the documents scanned during the current scan job.

4.6.1.2.1. DocumentDescription

This section of the schema defines all the description attributes that pertain to the basic creation information of the currently identified Document.

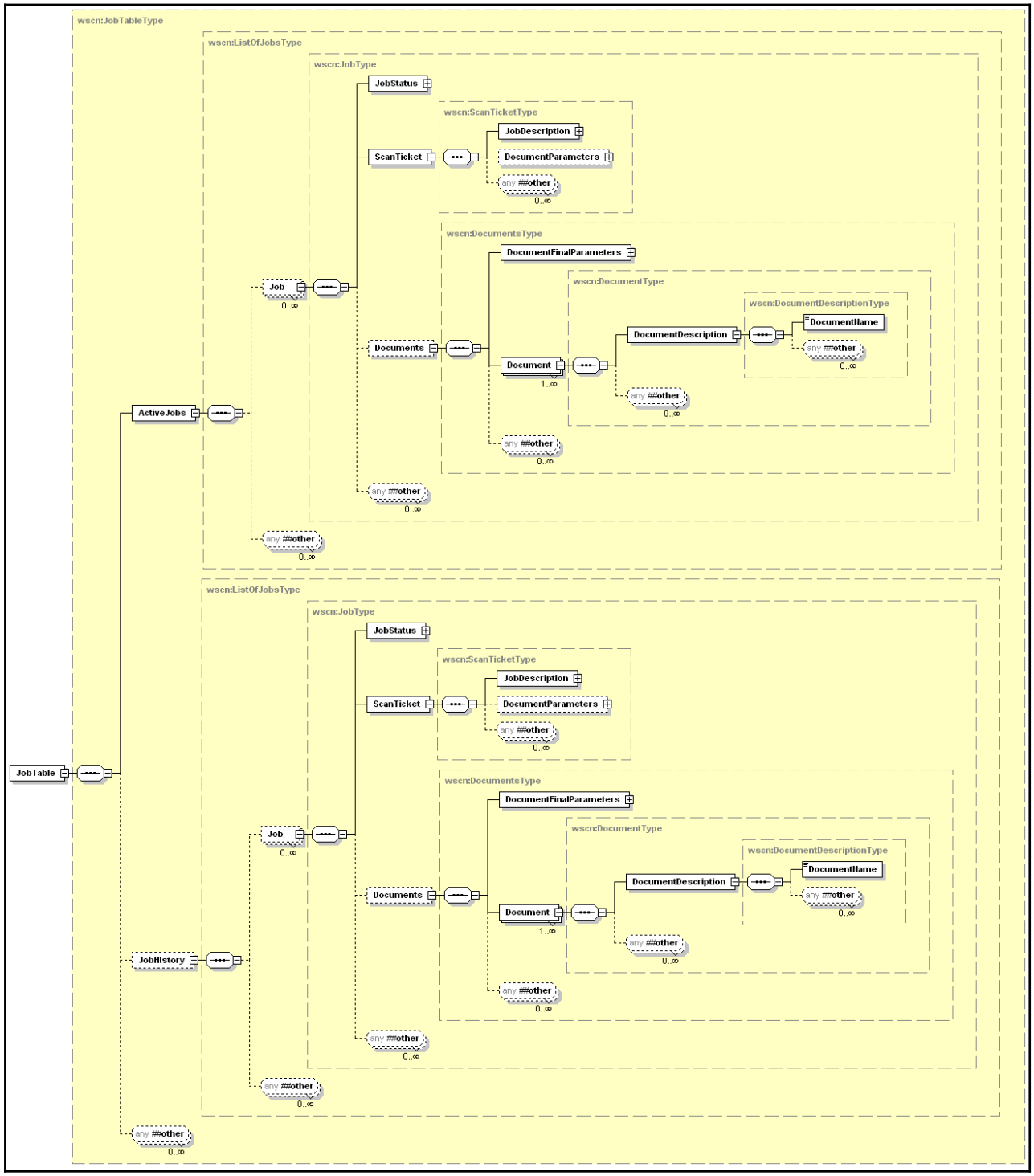
4.6.1.2.1.1. DocumentName

This element specifies the name of the document supplied by the CP. The CP MUST supply a value to be used to store the document on the client.

Allowed Values: any character string

4.7. Job Table

This section describes how the Scanner keeps track of all the current and finished jobs submitted to the Scanner service. Figure 15 below shows how the *JobTable* fits into the Scanner service. The *JobTable* has two lists of jobs. The first list (*ActiveJobs*) holds all the jobs which have not yet completed processing. The state of active jobs could be scanning, pending, or stopped. The key is they are not completed yet. The second optional list (*JobHistory*) contains a subset of the most recent jobs that have finished processing. These jobs could have scanned, been aborted, or failed for other reasons. The maximum number of jobs in this list is dependent upon the device.



1382
1383 **Figure 15 - JobTable Elements**

1384 4.8. ScanTicket

1385 The *ScanTicket* is an instance document conforming to a proper subset of the scan job schema supported by the scanner.
1386 The *ScanTicket* contains the values that the CP has selected for the scanner settings for the current job. The *ScanTicket* is
1387 composed of these subsections:

1388 *JobDescription* – This section of the *ScanTicket* contains general job information.

DocumentParameters– This section specifies the image processing functions and their values that will be applied against the job/document.

Figure 16 below illustrates the member elements of the *ScanTicket* element. Note that the member elements directly map to an instance of a Job and they are exactly what are required to be sent in a Job Creation operation.

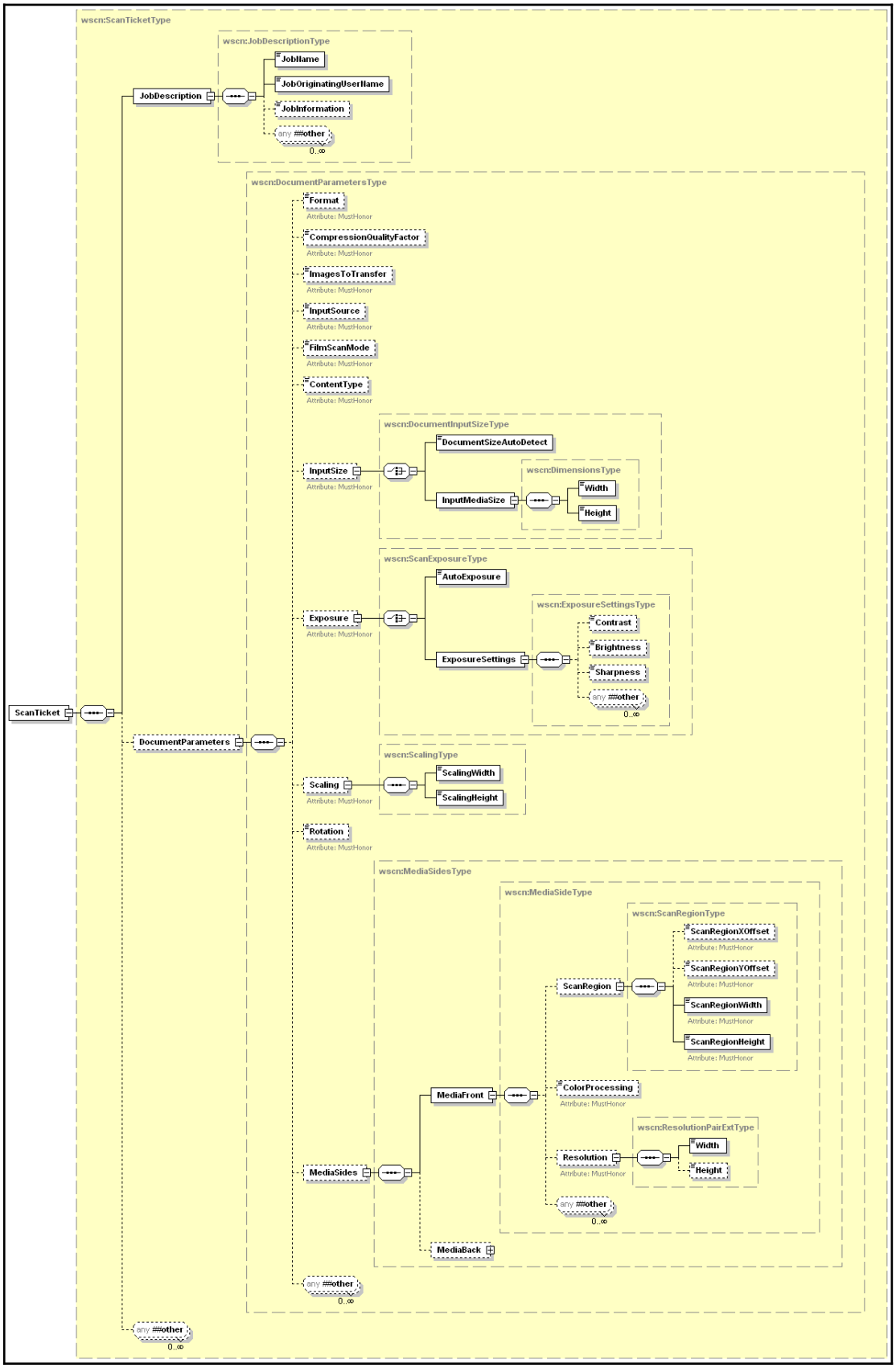


Figure 16 – ScanTicket Elements

4.8.1. Example ScanTicket

```
<wscn:ScanTicket>
    <wscn:JobDescription>
        <wscn:JobName>ScanJob for Den PC</wscn:JobName>
        <wscn:JobOriginatingUserName>Dad</wscn:JobOriginatingUserName>
        <wscn:JobInformation>Job created at Front Panel</wscn:JobInformation>
    </wscn:JobDescription>
    <wscn:DocumentParameters>
        <wscn:Format>jfif</wscn:Format>
        <wscn:CompressionQualityFactor>35</wscn:CompressionQualityFactor>
        <wscn:ImagesToTransfer>1</wscn:ImagesToTransfer>
        <wscn:InputSource>Platen</wscn:InputSource>
        <wscn:InputSize>
            <wscn:DocumentSizeAutoDetect>true</wscn:DocumentSizeAutoDetect>
        </wscn:InputSize>
        <wscn:Exposure>
            <wscn:AutoExposure>true</wscn:AutoExposure>
        </wscn:Exposure>
        <wscn:MediaSides>
            <wscn:MediaFront>
                <wscn:ScanRegion>
                    <wscn:ScanRegionXOffset>250</wscn:ScanRegionXOffset>
                    <wscn:ScanRegionYOffset>250</wscn:ScanRegionYOffset>
                    <wscn:ScanRegionWidth>4000</wscn:ScanRegionWidth>
                    <wscn:ScanRegionHeight>6000</wscn:ScanRegionHeight>
                </wscn:ScanRegion>
                <wscn:ColorProcessing>RGB24</wscn:ColorProcessing>
                <wscn:Resolution>
                    <wscn:Width>1200</wscn:Width>
                </wscn:Resolution>
            </wscn:MediaFront>
        </wscn:MediaSides>
    </wscn:DocumentParameters>
</wscn:ScanTicket>
```

4.9. Default Values and Allowed Values for Job Submission

The CP that submits a Scan Job may need to know the possible values and default values for the *DocumentParameters* elements that are included in a Job Creation operation. The Scanner has a *ScannerConfiguration* element and a *DefaultScanTicket* element to allow CPs to retrieve this information. The *DefaultScanTicket* element describes the current set of default values the scan service will apply when a job is created without specifying specific job creation elements. The *ScannerConfiguration* element contains information that will allow the CP to determine all of the allowed values for the Job and Document elements which can be specified to affect the image creation/acquisition process.

4.9.1. DefaultScanTicket

The *DefaultScanTicket* element is an instance of a *ScanTicket* element filled in with all the current default values for Job and Document creation. A CP can request the *DefaultScanTicket* and then use it as part of a **CreateScanJob** operation without error. The *DefaultScanTicket* will contain values for all *ScanTicket* options the device supports.

4.9.2. Example DefaultScanTicket

```
<wscn:DefaultScanTicket>
    <wscn:JobDescription>
        <wscn:JobName>Scan Job</wscn:JobName>
        <wscn:JobOriginatingUserName></wscn:JobOriginatingUserName>
        <wscn:JobInformation>User Selected Scan Job</wscn:JobInformation>
    </wscn:JobDescription>
    <wscn:DocumentParameters>
        <wscn:Format>tiff-multi-uncompressed</wscn:Format>
        <wscn:CompressionQualityFactor>100</wscn:CompressionQualityFactor>
        <wscn:ImagesToTransfer>0</wscn:ImagesToTransfer>
        <wscn:InputSource>Platen</wscn:InputSource>
        <wscn:FilmScanMode>NotApplicable</wscn:FilmScanMode>
        <wscn:ContentType>Auto</wscn:ContentType>
        <wscn:InputSize>
            <wscn:InputMediaSize>
                <wscn:Width>8500</wscn:Width>
                <wscn:Height>11000</wscn:Height>
```

```
            </wscn:InputMediaSize>
        </wscn:InputSize>
        <wscn:Exposure>
            <wscn:AutoExposure>true</wscn:AutoExposure>
        </wscn:Exposure>
        <wscn:Scaling>
            <wscn:ScalingWidth>100</wscn:ScalingWidth>
            <wscn:ScalingHeight>100</wscn:ScalingHeight>
        </wscn:Scaling>
        <wscn:Rotation>0</wscn:Rotation>
        <wscn:MediaSides>
            <wscn:MediaFront>
                <wscn:ScanRegion>
                    <wscn:ScanRegionXOffset>0</wscn:ScanRegionXOffset>
                    <wscn:ScanRegionYOffset>0</wscn:ScanRegionYOffset>
                    <wscn:ScanRegionWidth>8500</wscn:ScanRegionWidth>
                    <wscn:ScanRegionHeight>11000</wscn:ScanRegionHeight>
                </wscn:ScanRegion>
                <wscn:ColorProcessing>RGB24</wscn:ColorProcessing>
                <wscn:Resolution>
                    <wscn:Width>600</wscn:Width>
                    <wscn:Height>600</wscn:Height>
                </wscn:Resolution>
            </wscn:MediaFront>
        </wscn:MediaSides>
    </wscn:DocumentParameters>
</wscn:DefaultScanTicket>
```

5. Eventing

The scanner service will implement eventing as defined by the WS-Eventing [EVENT] specification. The scanner service will be an extension of the WS-Eventing porttype. This will add the operations defined in that spec to the scan service for use in creating and managing Event subscriptions.

5.1. Event Model

The eventing model for the scan service serves three main purposes: First is to inform the CP when the Configuration changes on the scan device. This could include changes in the ADF capabilities or changes in storage available, etc. Second is to inform the CP when there is a change in condition of the scan device. Examples: the scanner becomes idle or a document jam occurs. The *ScannerState*, *ScannerStateReasons* and *DeviceConditon* elements provide this information. Third is for job tracking; events inform a CP when a job is submitted, completed or removed from the job queue. The *JobStatus* and **JobEndStateEvent** elements provide this information. **JobEndStateEvent** indicates the final status of each job. It lets control points know whether it completed successfully or was canceled or aborted. *ScansCompleted* in the *JobStatus* is an example of information that updates an interested CP on the number of images scanned for the current job.

5.2. ScanAvailableEvent

This event is defined to inform the CP that a scan device for which the CP is subscribed is ready to scan a job. To keep from notifying every CP when a user presses the scan button, the **Subscribe** request will contain one or more destinations described by extensions that the scanner will use to filter down to a single CP per **ScanAvailableEvent** notification. These elements are defined in the scanner service namespace, and then added to the *Subscribe* request body.

5.2.1. Subscribe Extensions

Each of these extended elements is required to be sent in the *Subscribe* element of the [EVENT] **Subscribe** operation. It is important to note that only one *ScanDestinations* extension element should be included in the *Subscribe* element. Any subsequent *ScanDestinations* elements will be ignored. This also implies that only one *DestinationResponses* extension element will be returned in the *SubscribeResponse* element. An example subscribe follows the element definitions.

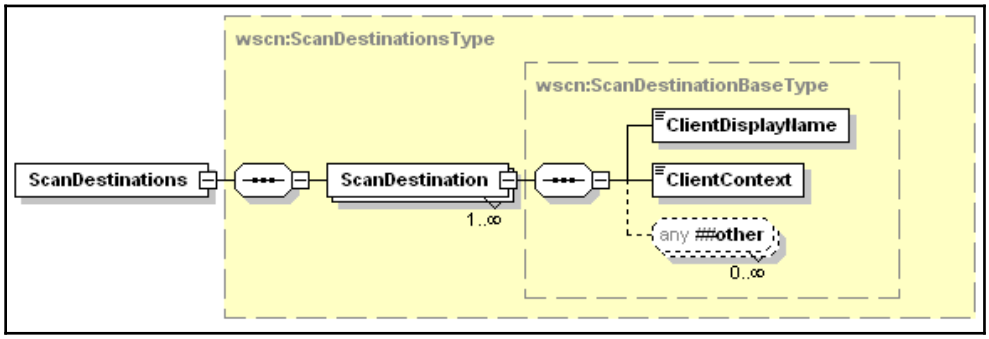


Figure 17 - ScanDestinations Elements

5.2.1.1. ScanDestinations

This element is a collection of all the Scan destinations the client wishes to register with the scan device. In most cases there will be a single scan destination, but this allows the client the flexibility to register multiple unique destinations at once.

5.2.1.1.1. ScanDestination

This element is a single scan destination on the CP. The elements that make up this destination registration will be used by the scan device to create appropriate **ScanAvailableEvent**(s) and are described fully below.

5.2.1.1.1.1. ClientDisplayName

This element specifies a string the scanner should display in its user-interface to allow the user to select the requesting client as a scan destination. When the user picks this display name and presses the scan button, the scanner will send a **ScanAvailableEvent** to the endpoint contained in this subscription.

Allowed Values: any character string

5.2.1.1.1.2. ClientContext

This element specifies a client specific string the scanner will send in the **ScanAvailableEvent**. This string will allow the client to associate the **ScanAvailableEvent** with the correct scanner device/service.

Allowed Values: any character string

5.2.2. Example ScanAvailableEvent Subscribe

```
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wse="http://schemas.xmlsoap.org/ws/2004/08/eventing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
        <wsa:To>AddressofScannerService</wsa:To>
        <wsa:Action>
            http://schemas.xmlsoap.org/ws/2004/08/eventing/Subscribe
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
    </soap:Header>
    <soap:Body>
        <wse:Subscribe>
            <wse:Delivery>
                <wse:NotifyTo>
                    <wsa:Address>
                        http://www.example.com/MyEventSink/OnScanAvailableForMe
                    </wsa:Address>
                </wse:NotifyTo>
            </wse:Delivery>
            <wse:Expires>P0Y0M0DT30H0M0S</wse:Expires>
            <wse:Filter Dialect="http://schemas.xmlsoap.org/ws/2006/02/devprof/Action">
                http://schemas.microsoft.com/windows/2006/08/wdp/scan/ScanAvailableEvent
            </wse:Filter>
            <wscn:ScanDestinations>
```

```
                <wscn:ScanDestination>
                    <wscn:ClientDisplayString>Den Computer</wscn:ClientDisplayString>
                    <wscn:ClientContext>App1ScanID2345</wscn:ClientContext>
                </wscn:ScanDestination>
            </wscn:ScanDestinations>
        </wse:Subscribe>
    </soap:Body
</soap:Envelope>
```

5.2.3. SubscribeResponse Extensions

There is also an extended element in the **SubscribeResponse** body. This element helps tie the subscription back to the device that accepted it.

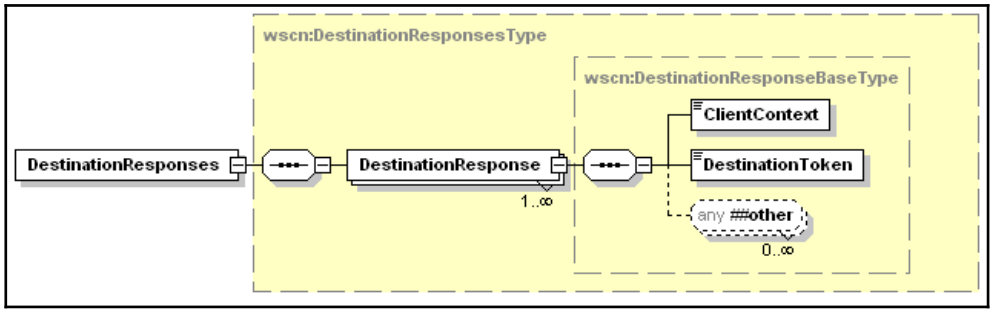


Figure 18 - DestinationResponses Elements

5.2.3.1. DestinationResponses

This element is a collection of all the responses to the scan destination requests. There will be one *DestinationResponse* element for each *ScanDestination* element in the **Subscribe** request.

5.2.3.1.1. DestinationResponse

This element is the response information for a single *ScanDestination* registration. The element contains the *ClientContext* from the *ScanDestination* to identify the response and a *DestinationToken* element for use in all **CreateScanJob** operations from this destination.

5.2.3.1.1.1. ClientContext

This element is a copy of the data sent in a *ScanDestination* element in the **Subscribe** operation. There is a one-to-one match of *ClientContext*(s) between a *ScanDestination* element and a *DestinationResponse* element. See Section 5.2.1.1.1.2. for a detailed description of this data element.

5.2.3.1.1.2. DestinationToken

This element specifies a device specific string the scanner will assign to this client destination. When the client sends the **CreateScanJob** operation after a **ScanAvailableEvent** it will include this token. This string will allow the device to double check the correct client is requesting the scan.

Allowed Values: any character string

5.2.4. Example ScanAvailableEvent SubscribeResponse

```
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wse="http://schemas.xmlsoap.org/ws/2004/08/eventing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”
        soap:encodingStyle='http://www.w3.org/2002/12/soap-encoding' >
    <soap:Header>
        <wsa:To>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa:To>
        <wsa:Action>
            http://schemas.xmlsoap.org/ws/2004/08/eventing/SubscribeResponse
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
        <wsa:RelatesTo>uuid:MsgIdOfTheSubscribe</wsa:RelatesTo>
```

```
    </soap:Header>
    <soap:Body>
        <wse:SubscribeResponse>
            <wse:SubscriptionManager>
                <!-- Elements removed for clarity  -->
            </wse:SubscriptionManager>
            <wse:Expires>P0Y0M0DT30H0M0S</wse:Expires>
            <wscn:DestinationResponses>
                <wscn:DestinationResponse>
                    <wscn:ClientContext>App1ScanID2345</wscn:ClientContext>
                    <wscn:DestinationToken>Client3478</wscn:DestinationToken>
                </wscn:DestinationResponse>
            </wscn:DestinationResponses>
        </wse:SubscribeResponse>
    </soap:Body
</soap:Envelope>
```

5.2.5. Event Elements

The body of the **ScanAvailableEvent** consists of information that will allow the CP to determine which scanner to send the **CreateScanJob** operation.

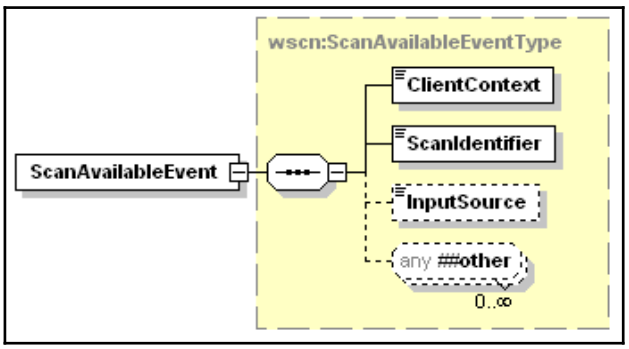


Figure 19 – ScanAvailableEvent Elements

5.2.5.1. ClientContext

This element contains a client specific string the scanner received as part of the **ScanAvalibleEvent** subscription request.. See section 5.2.1.1.1.2. for more details about this element.

5.2.5.2. ScanIdentifier

This element specifies a device specific string the scanner sends in the **ScanAvailableEvent**. This identifier will be sent as part of the resulting **CreateScanJob** operation from the client. It allows the scanner to make sure the correct client is requesting the scan after a user has selected the destination. This value should be unique per instance of the **ScanAvailableEvent**.

Allowed Values: any character string

5.2.5.3. InputSource

This optional element contains the *InputSource* the user has placed the original images in on the scan device. The scan client should use this value for the *InputSource* element of the *ScanTicket* in the successive **CreateScanJob** operation. See section 4.5.2.2.4. for more details about this element.

5.2.6. Example ScanAvailableEvent

```
<soap:Envelope
       xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
       xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
       xmlns:wse="http://schemas.xmlsoap.org/ws/2004/08/eventing"
       xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
       <wsa:To>AddressofEventSink</wsa:To>
       <wsa:Action>
           http://schemas.microsoft.com/windows/2006/08/wdp/scan/ScanAvailableEvent
       </wsa:Action>
```

```
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
    </soap:Header>
    <soap:Body>
        <wscn:ScanAvailableEvent>
            <wscn:ClientContext>App1ScanID2345</wscn:ClientContext>
            <wscn:ScanIdentifier>AnyUniqueIdentifierSuchAsAGUID</wscn:ScanIdentifier>
            <wscn:InputSource>ADF</wscn:InputSource>
        </wscn:ScanAvailableEvent>
    </soap:Body
</soap:Envelope>
```

5.3. ScannerElementsChangeEvent

This event is defined to inform the CP that something has changed in the *ScannerDescription* element, the *ScannerConfiguration* element, the *DefaultScanTicket* element, or an IHV extension in the Scanner. The body of the **ScannerElementsChangeEvent** consists of the complete XML for the updated element. If an optional element is missing from returned XML the implication is that that element is no longer supported by the scanner service. This could be caused by a removal of a film scan option or a duplex scanning mode. The CP is then responsible for comparing the incoming element against previous data to determine which values have changed.

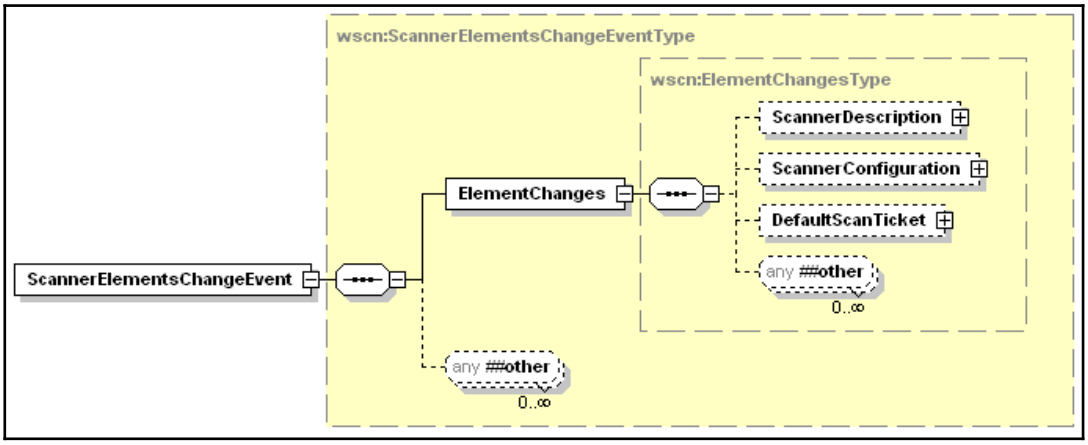


Figure 20 - ScannerElementsChangeEvent Elements

5.3.1. ElementChanges

The information returned in this data element should be entire element in the Scanner schema which contains changed values. This could be because an ADF was installed, a value changed in the *DefaultScanTicket* or *DeviceSettings,* or an IHV extension has changed value. In each case the complete *ScannerDescription*, *ScannerConfiguration*, *DefaultScanTicket* or IHV extension would be returned in this element.

5.3.2. Example ScannerElementsChangeEvent

In this example the device is reporting updated device configuration due to the installation of a *Film* scanning option.

```
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wse="http://schemas.xmlsoap.org/ws/2004/08/eventing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
        <wsa:To>AddressofEventSink</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/ScannerElementsChangeEvent
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
    </soap:Header>
    <soap:Body>
        <wscn:ScannerElementsChangeEvent>
            <wscn:ElementChanges>
                <wscn:ScannerConfiguration>
                    <wscn:DeviceSettings>
                        <wscn:FormatsSupported>
```

```
                        <wscn:FormatValue>dib</wscn:FormatValue>
                        <wscn:FormatValue>exif</wscn:FormatValue>
                        <wscn:FormatValue>jpeg2k</wscn:FormatValue>
                        <wscn:FormatValue>pdf-a</wscn:FormatValue>
                        <wscn:FormatValue>png</wscn:FormatValue>
                        <wscn:FormatValue>tiff-single-uncompressed</wscn:FormatValue>
                        <wscn:FormatValue>tiff-single-g4</wscn:FormatValue>
                        <wscn:FormatValue>tiff-multi-uncompressed</wscn:FormatValue>
                        <wscn:FormatValue>tiff-multi-g4</wscn:FormatValue>
                        <wscn:FormatValue>xps</wscn:FormatValue>
                    </wscn:FormatsSupported>
                    <wscn:CompressionQualityFactorSupported>
                        <wscn:MinValue>15</wscn:MinValue>
                        <wscn:MaxValue>100</wscn:MaxValue>
                    </wscn:CompressionQualityFactorSupported>
                    <wscn:ContentTypesSupported>
                        <wscn:ContentTypeValue>Auto</wscn:ContentTypeValue>
                        <wscn:ContentTypeValue>Text</wscn:ContentTypeValue>
                        <wscn:ContentTypeValue>Photo</wscn:ContentTypeValue>
                        <wscn:ContentTypeValue>Halftone </wscn:ContentTypeValue>
                        <wscn:ContentTypeValue>Mixed</wscn:ContentTypeValue>
                    </wscn:ContentTypesSupported>
                    <wscn:DocumentSizeAutoDetectSupported>
                        true
                    </wscn:DocumentSizeAutoDetectSupported>
                    <wscn:AutoExposureSupported>true</wscn:AutoExposureSupported>
                    <wscn:BrightnessSupported>true</wscn:BrightnessSupported>
                    <wscn:ContrastSupported>true</wscn:ContrastSupported>
                    <wscn:ScalingRangeSupported>
                        <wscn:ScalingWidth>
                            <wscn:MinValue>50</wscn:MinValue>
                            <wscn:MaxValue>500</wscn:MaxValue>
                        </wscn:ScalingWidth>
                        <wscn:ScalingHeight>
                            <wscn:MinValue>50</wscn:MinValue>
                            <wscn:MaxValue>500</wscn:MaxValue>
                        </wscn:ScalingHeight>
                    </wscn:ScalingRangeSupported>
                    <wscn:RotationsSupported>
                        <wscn:RotationValue>0</wscn:RotationValue>
                        <wscn:RotationValue>90</wscn:RotationValue>
                        <wscn:RotationValue>180</wscn:RotationValue>
                        <wscn:RotationValue>270</wscn:RotationValue>
                    </wscn:RotationsSupported>
                </wscn:DeviceSettings>
                <wscn:Platen>
                    <wscn:PlatenOpticalResolution>
                        <wscn:Width>1200</wscn:Width>
                        <wscn:Height>1200</wscn:Height>
                    </wscn:PlatenOpticalResolution>
                    <wscn:PlatenResolutions>
                        <wscn:Widths>
                            <wscn:Width>150</wscn:Width>
                            <wscn:Width>204</wscn:Width>
                            <wscn:Width>300</wscn:Width>
                            <wscn:Width>600</wscn:Width>
                            <wscn:Width>1200</wscn:Width>
                        </wscn:Widths>
                        <wscn:Heights>
                            <wscn:Height>96</wscn:Height>
                            <wscn:Height>150</wscn:Height>
                            <wscn:Height>204</wscn:Height>
                            <wscn:Height>300</wscn:Height>
                            <wscn:Height>600</wscn:Height>
                            <wscn:Height>900</wscn:Height>
                            <wscn:Height>1200</wscn:Height>
                        </wscn:Heights>
                    </wscn:PlatenResolutions>
                    <wscn:PlatenColor>
                        <wscn:ColorEntry>BlackAndWhite1</wscn:ColorEntry>
                        <wscn:ColorEntry>Grayscale4</wscn:ColorEntry>
                        <wscn:ColorEntry>Grayscale8</wscn:ColorEntry>
                        <wscn:ColorEntry>RGB24</wscn:ColorEntry>
```

```
                        <wscn:ColorEntry>RGB48</wscn:ColorEntry>
                        <wscn:ColorEntry>RGBa32</wscn:ColorEntry>
                        <wscn:ColorEntry>RGBa64</wscn:ColorEntry>
                    </wscn:PlatenColor>
                    <wscn:PlatenMinimumSize>
                        <wscn:Width>250</wscn:Width>
                        <wscn:Height>250</wscn:Height>
                    </wscn:PlatenMinimumSize>
                    <wscn:PlatenMaximumSize>
                        <wscn:Width>11000</wscn:Width>
                        <wscn:Height>14000</wscn:Height>
                    </wscn:PlatenMaximumSize>
                </wscn:Platen>
                <wscn:ADF>
                    <wscn:ADFSupportsDuplex>false</wscn:ADFSupportsDuplex>
                    <wscn:ADFFront>
                        <wscn:ADFOpticalResolution>
                            <wscn:Width>600</wscn:Width>
                            <wscn:Height>600</wscn:Height>
                        </wscn:ADFOpticalResolution>
                        <wscn:ADFResolutions>
                            <wscn:Widths>
                                <wscn:Width>150</wscn:Width>
                                <wscn:Width>204</wscn:Width>
                                <wscn:Width>300</wscn:Width>
                                <wscn:Width>600</wscn:Width>
                            </wscn:Widths>
                            <wscn:Heights>
                                <wscn:Height>96</wscn:Height>
                                <wscn:Height>150</wscn:Height>
                                <wscn:Height>204</wscn:Height>
                                <wscn:Height>300</wscn:Height>
                                <wscn:Height>600</wscn:Height>
                            </wscn:Heights>
                        </wscn:ADFResolutions>
                        <wscn:ADFColor>
                            <wscn:ColorEntry>BlackAndWhite1</wscn:ColorEntry>
                            <wscn:ColorEntry>Grayscale4</wscn:ColorEntry>
                            <wscn:ColorEntry>RGB24</wscn:ColorEntry>
                        </wscn:ADFColor>
                        <wscn:ADFMinimumSize>
                            <wscn:Width>4000</wscn:Width>
                            <wscn:Height>6000</wscn:Height>
                        </wscn:ADFMinimumSize>
                        <wscn:ADFMaximumSize>
                            <wscn:Width>8500</wscn:Width>
                            <wscn:Height>11000</wscn:Height>
                        </wscn:ADFMaximumSize>
                    </wscn:ADFFront>
                </wscn:ADF>
                <wscn:Film>
                    <wscn:FilmScanModesSupported>
                        <wscn:FilmScanModeValue>
                            ColorSlideFilm
                        </wscn:FilmScanModeValue>
                        <wscn:FilmScanModeValue>
                            ColorNegativeFilm
                        </wscn:FilmScanModeValue>
                        <wscn:FilmScanModeValue>
                            BlackandWhiteNegativeFilm
                        </wscn:FilmScanModeValue>
                    </wscn:FilmScanModesSupported>
                    <wscn:FilmOpticalResolution>
                        <wscn:Width>600</wscn:Width>
                        <wscn:Height>600</wscn:Height>
                    </wscn:FilmOpticalResolution>
                    <wscn:FilmResolutions>
                        <wscn:Widths>
                            <wscn:Width>150</wscn:Width>
                            <wscn:Width>300</wscn:Width>
                            <wscn:Width>600</wscn:Width>
                        </wscn:Widths>
                        <wscn:Heights>
```

```
                                <wscn:Height>150</wscn:Height>
                                <wscn:Height>300</wscn:Height>
                                <wscn:Height>600</wscn:Height>
                            </wscn:Heights>
                        </wscn:FilmResolutions>
                        <wscn:FilmColor>
                            <wscn:ColorEntry>BlackAndWhite1</wscn:ColorEntry>
                            <wscn:ColorEntry>Grayscale4</wscn:ColorEntry>
                            <wscn:ColorEntry>RGB24</wscn:ColorEntry>
                            <wscn:ColorEntry>RGBa32</wscn:ColorEntry>
                        </wscn:FilmColor>
                        <wscn:FilmMinimumSize>
                            <wscn:Width>1378</wscn:Width>
                            <wscn:Height>1378</wscn:Height>
                        </wscn:FilmMinimumSize>
                        <wscn:FilmMaximumSize>
                            <wscn:Width>2756</wscn:Width>
                            <wscn:Height>10000</wscn:Height>
                        </wscn:FilmMaximumSize>
                    </wscn:Film>
                </wscn:ScannerConfiguration>
            </wscn:ElementChanges>
        </wscn:ScannerElementsChangeEvent>
    </soap:Body
</soap:Envelope>
```

5.4. ScannerStatusSummaryEvent

This event is defined to inform the CP that the device status has changed. The body of the **ScannerStatusSummaryEvent** consists of the *StatusSummary* element.

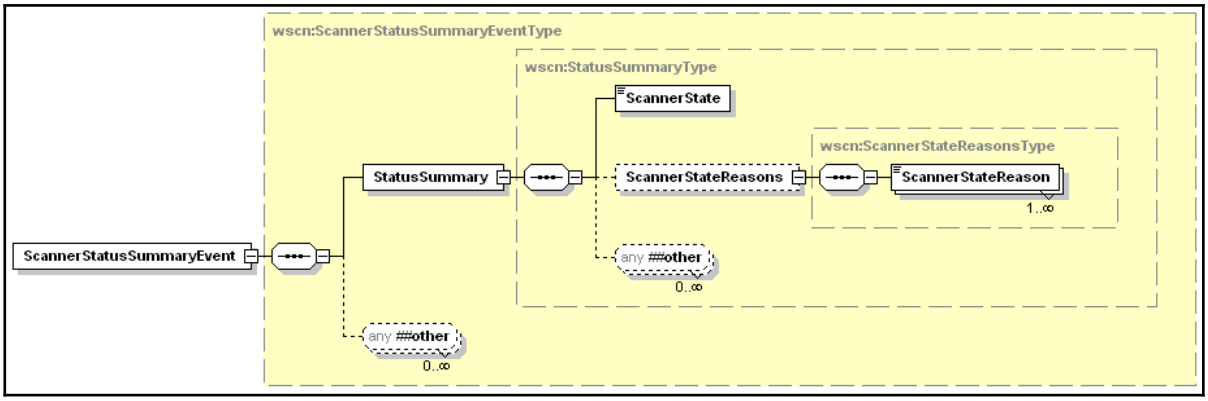


Figure 21 - ScannerStatusSummaryEvent Elements

5.4.1. StatusSummary

This data element contains the current summary of the scanner status.

5.4.1.1. ScannerState

This data element is described in Section 4.4.2.

5.4.1.2. ScannerStateReasons

This data element is described in Section 4.4.3.

5.4.1.2.1. ScannerStateReason

This data element is described in Section 4.4.3.1..

5.4.2. Example ScannerStatusSummaryEvent

In this example the device is stopped because of a jam in the media feed path.

```
<soap:Envelope
```

```
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wse="http://schemas.xmlsoap.org/ws/2004/08/eventing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
        <wsa:To>AddressofEventSink</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/ScannerStatusSummaryEvent
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
    </soap:Header>
    <soap:Body>
        <wscn:ScannerStatusSummaryEvent>
            <wscn:StatusSummary>
                <wscn:ScannerState>Stopped</wscn:ScannerState>
                <wscn:ScannerStateReasons>
                    <wscn:ScannerStateReason>MediaJam</wscn:ScannerStateReason>
                </wscn:ScannerStateReasons>
            </wscn:StatusSummary>
        </wscn:ScannerStatusSummaryEvent>
    </soap:Body
</soap:Envelope>
```

5.5. ScannerStatusConditionEvent

This event is defined to inform the CP detailed information about a status change in the device. The body of the **ScannerStatusConditionEvent** event consists of the *DeviceConditon* element for the status change. This element contains the following data:

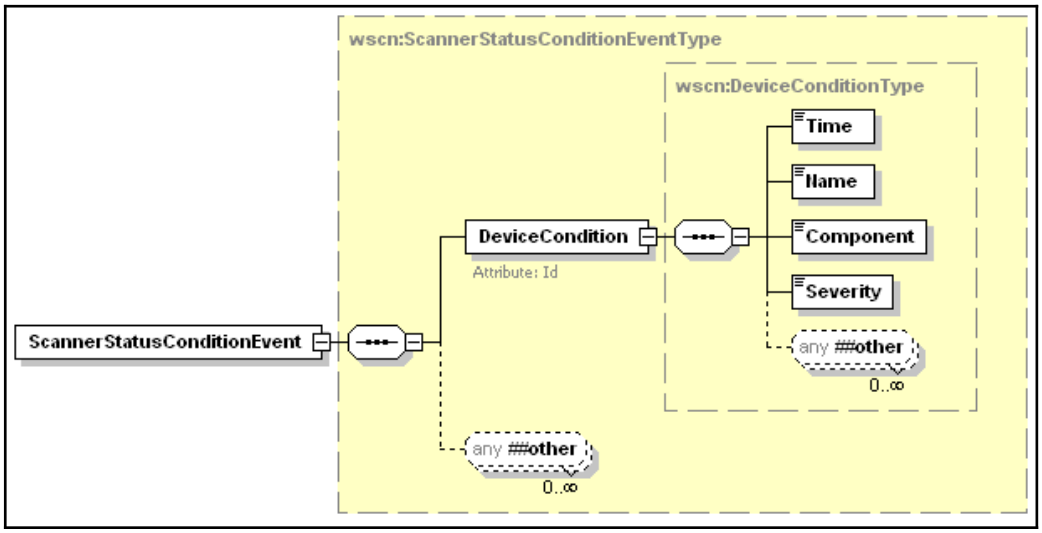


Figure 22 - ScannerStatusConditon Event

5.5.1. DeviceCondition

This is an element that describes the details about one of the currently active conditions. This data element is described in Section 4.4.4.1..

5.5.2. Example ScannerStatusConditionEvent

In this example the device is notifying the CP of a scan Lamp failure.

```
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wse="http://schemas.xmlsoap.org/ws/2004/08/eventing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
        <wsa:To>AddressofEventSink</wsa:To>
        <wsa:Action>
```

```
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/ScannerStatusConditionEvent
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
    </soap:Header>
    <soap:Body>
        <wscn:ScannerStatusConditionEvent>
            <wscn:DeviceCondition Id="1543">
                <wscn:Time>2006-01-21T17:22:27.5242689Z</wscn:Time>
                <wscn:Name>LampError</wscn:Name>
                <wscn:Component>Platen</wscn:Component>
                <wscn:Severity>Critical</wscn:Severity>
            </wscn:DeviceCondition>
        </wscn:ScannerStatusConditionEvent>
    </soap:Body
</soap:Envelope>
```

5.6. ScannerStatusConditionClearedEvent

This event is defined to inform the CP that a previously reported *DeviceCondition* has been cleared. The body of the **ScannerStatusConditionClearedEvent** message consists of the *DeviceConditonId* element for the condition that has been cleared, and an element representing the time the condition was cleared. This event contains the following data:

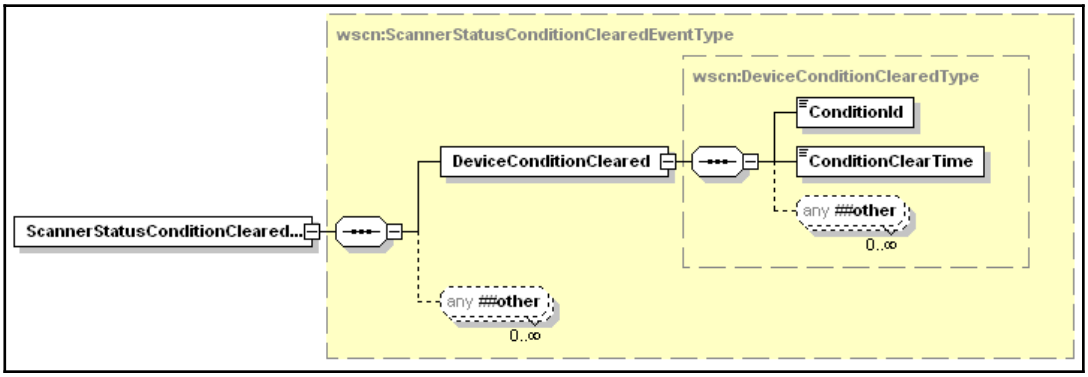


Figure 23 - ScannerStatusConditionCleared Event

5.6.1. DeviceConditonCleared

This data element contains the *Id* of the condition that cleared and the time the condition was cleared.

5.6.1.1. ConditonId

This data element is equivalent to the *Id* attribute of a *DeviceCondition* entry. See Section 4.4.4.1.1. for an explanation of the format.

5.6.1.2. ConditionClearTime

This data element is a dateTime element that describes when the condition was cleared. See Section 4.4.5.1.6. for an explanation of the format.

5.6.2. Example ScannerStatusConditionClearedEvent

In this example the device is notifying the CP that the previous condition identified by Id 1543 has cleared.

```
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wse="http://schemas.xmlsoap.org/ws/2004/08/eventing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan">
    <soap:Header>
        <wsa:To>AddressofEventSink</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/ScannerStatusConditionClearedEvent
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
    </soap:Header>
```

```
    <soap:Body>
        <wscn:ScannerStatusConditionClearedEvent>
            <wscn:DeviceConditionCleared>
                <wscn:ConditionId>1543</wscn:ConditionId>
                <wscn:ConditionClearTime>2006-01-21T17:22:35.8345Z</wscn:ConditionClearTime>
            </wscn:DeviceConditionCleared>
        </wscn:ScannerStatusConditionClearedEvent>
    </soap:Body
</soap:Envelope>
```

5.7. JobStatusEvent

This event is defined to inform the CP that a job's status has changed. The first **JobStatusEvent** message sent will usually have the *JobId* and a *JobState* of `Started`. The body of the **JobStatusEvent** message consists of the following data elements:

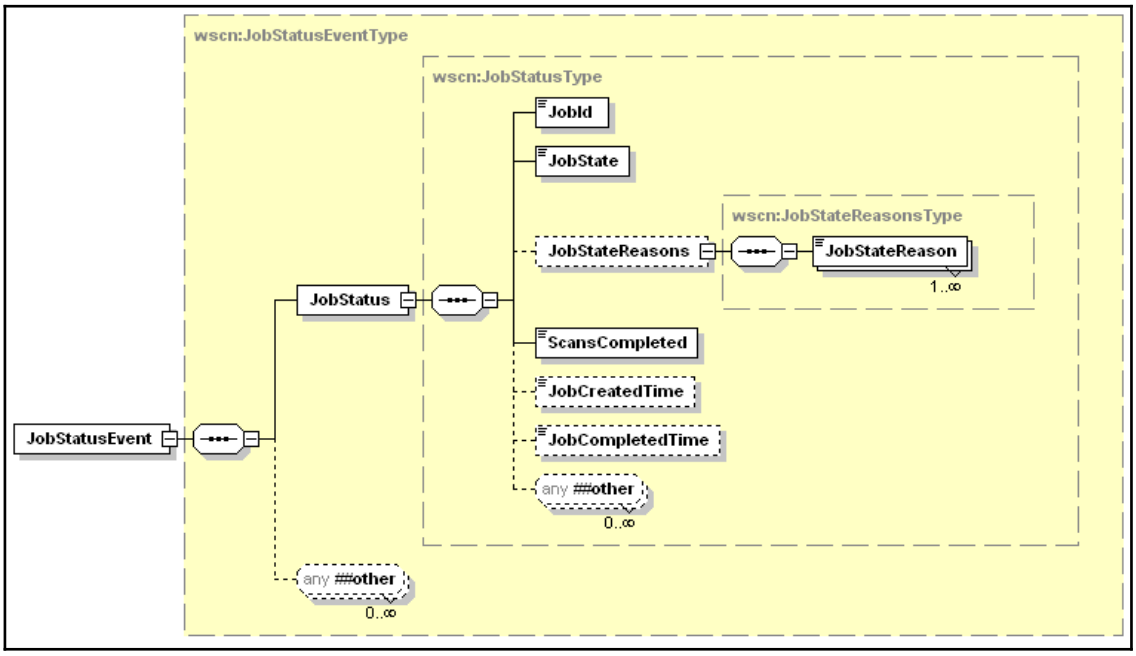


Figure 24 - JobStatus Event

5.7.1. JobStatus

This data element contains the current status of a Job. This data element is described in Section 4.5.1.

5.7.2. Example JobStatusEvent

In this example the device is notifying the CP of the current state of Job 253.

```
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wse="http://schemas.xmlsoap.org/ws/2004/08/eventing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
        <wsa:To>AddressofEventSink</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/JobStatusEvent
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
    </soap:Header>
    <soap:Body>
        <wscn:JobStatusEvent>
            <wscn:JobStatus>
                <wscn:JobId>253</wscn:JobId>
                <wscn:JobState>Processing</wscn:JobState>
                <wscn:JobStateReasons>
```

```
                    <wscn:JobStateReason>JobScanning</wscn:JobStateReason>
                </wscn:JobStateReasons>
                <wscn:ScansCompleted>4</wscn:ScansCompleted>
                <wscn:JobCreatedTime>2006-01-24T11:34:35.8345Z</wscn:JobCreatedTime>
            <wscn:JobStatus>
        </wscn:JobStatusEvent>
    </soap:Body
</soap:Envelope>
```

5.8. JobEndStateEvent

This event is defined to inform the CP that a Job has finished processing.

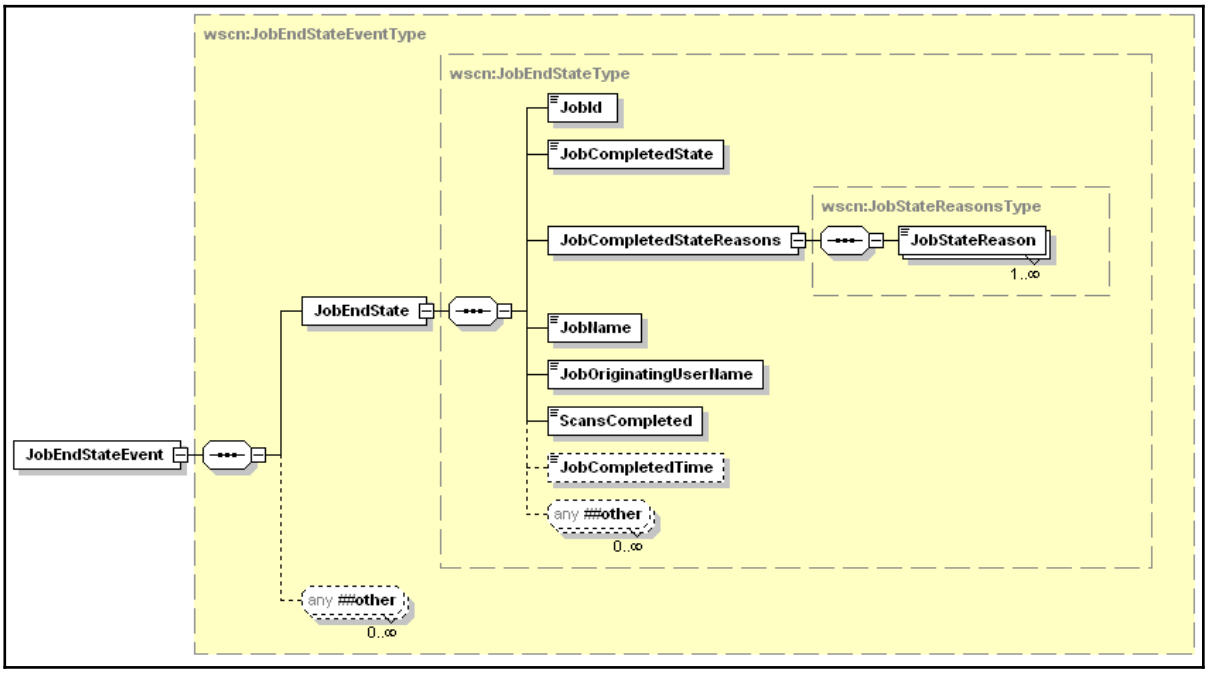


Figure 25 - JobEndState Event

5.8.1. JobEndState

This data element contains the final state of the Job status elements.

5.8.1.1. JobId

This data element is described in Section 4.5.1.1.

5.8.1.2. JobCompletedState

This data element is the final *JobState* for the job and has the same semantics and values as the *JobState* element from *JobStatus*. The *JobState* element is described in Section 4.5.1.2.

5.8.1.2.1. JobCompletedStateReasons

This element is a collection that describes all of the additional information about how/why the Job completed.

5.8.1.2.1.1. JobStateReason

This elements indicates additional information about how/why the Job completed and has the same semantics and values as the *JobStateReason* element from *JobStatus.* The *JobStateReason* element is described in Section 4.5.1.3.1.

5.8.1.3. JobName

This data element is described in Section 4.5.2.1.1.

5.8.1.4. JobOriginatingUser

This data element is described in Section 4.5.2.1.2.

5.8.1.5. ScansCompleted

This data element is described in Section 4.5.1.6.

5.8.1.6. JobCompletedTime

This data element is described in Section 4.5.1.5.

5.8.2. Example JobEndStateEvent

In this example the device is notifying the CP of the final state/status of Job 253.

```
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wse="http://schemas.xmlsoap.org/ws/2004/08/eventing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
        <wsa:To>AddressofEventSink</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/JobEndStateEvent
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
    </soap:Header>
    <soap:Body>
        <wscn:JobEndStateEvent>
            <wscn:JobEndState>
                <wscn:JobId>253</wscn:JobId>
                <wscn:JobCompletedState>Completed</wscn:JobCompletedState>
                <wscn:JobCompletedStateReasons>
                    <wscn:JobStateReason>JobCompletedWithWarnings</wscn:JobStateReason>
                </wscn:JobCompletedStateReasons>
                <wscn:JobName>Scan from Imaging App</wscn:JobName>
                <wscn:JobOriginatingUserName>User</wscn:JobOriginatingUserName>
                <wscn:ScansCompleted>7</wscn:ScansCompleted>
                <wscn:JobCompletedTime>2006-01-24T11:37:05.673Z</</wscn:JobCompletedTime>
            </wscn:JobEndState>
        </wscn:JobEndStateEvent>
    </soap:Body
</soap:Envelope>
```

6. Operations

Immediately following this table is detailed information about these operations, including short descriptions of the operations, the effects of the operations on state, and error codes defined by the operations.

Table 2 - Operations

Name	Req. or Opt. [1]
CreateScanJob	R
RetrieveImage	R
CancelJob	R
ValidateScanTicket	R
GetScannerElements	R
GetJobElements	R
GetActiveJobs	R
GetJobHistory	R
Non-standard operations implemented by a WSD vendor go here.	X

[1] R = REQUIRED, O = Optional, X = Non-standard.

6.1. Operation Error Reporting

Error codes are returned in the <soap:Fault> element. A vendor MAY subset or extend these error codes by supporting private error subcodes. All fault messages defined in this specification MUST be sent according to the rules described in [ADDRESS] section 4 and [DEVICE]. They are sent to the [fault endpoint], if present and valid. Otherwise they are sent to the [reply endpoint] if present. If neither is present faults may be sent to the [source endpoint].

Endpoints compliant with this specification MUST include required message information headers on all fault messages. Fault messages are correlated as replies using the [relationship] property as defined in WS-Addressing. The [action] property below designates fault messages:

`http://schemas.xmlsoap.org/ws/2004/08/addressing/fault`

The definitions of faults use the following properties:

[Code]	The fault code.
[Subcode]	The fault subcode.
[Reason]	The English language reason element.
[Detail]	The detail element. If absent, no detail element is defined for the fault.

The properties above bind to a SOAP 1.2 fault as follows:

```
<S:Envelope>
    <S:Header>
        <wsa:Action>http://schemas.xmlsoap.org/ws/2004/08/addressing/fault</wsa:Action>
        <!-- Headers excluded for clarity -->
    </S:Header>
    <S:Body>
        <S:Fault>
            <S:Code>
                <S:Value>[Code]</S:Value>
                <S:Subcode>
                    <S:Value>[Subcode]</S:Value>
                </S:Subcode>
            </S:Code>
            <S:Reason>
                <S:Text xml:lang="en">[Reason]</S:Text>
            </S:Reason>
            <S:Detail> [Detail] </S:Detail>
        </S:Fault>
    </S:Body>
</S:Envelope>
```

Example SOAP Fault:

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soapelope"
        xmlns:xml=”http://www.w3.org/XML/1998/namespace”
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/08/wdp/scan">
    <soap:Header>
        <wsa:Action>http://schemas.xmlsoap.org/ws/2004/08/addressing/fault</wsa:Action>
        <!-- Headers excluded for brevity -->
    </soap:Header>
    <soap:Body>
        <soap:Fault>
            <soap:Code>
                <soap:Value>soap:Receiver</soap:Value>
                <soap:Subcode>
                    <soap:Value>wscn:OperationFailed</soap:Value>
                </soap:Subcode>
            </soap:Code>
            <soap:Reason>
                <soap:Text xml:lang="en">Service can not perform the requested Operation</soap:Text>
            </soap:Reason>
```

```
        </soap:Fault>
    </soap:Body>
</soap:Envelope>
```

6.1.1. Common Operation Faults

The following table lists SOAP Faults common to all operations for this service type. If an operation results in multiple errors, the most specific fault SHOULD be returned.

6.1.1.1. wsa:ActionNotSupported

This fault is sent when a Client requests an operation that is not supported by the current service.

[Code]	soap:Sender
[Subcode]	wsa:ActionNotSupported
[Reason]	The [wsa:action] cannot be processed at the receiver
[Detail]	*The invalid operation name*

6.1.1.2. InvalidArgs

This fault is sent when a Client sends an invalid argument as part of an operation. The invalid argument could be any of the following: not enough in args, too many in args, no in arg by that name, one or more in args are of the wrong data type.

[Code]	soap:Sender
[Subcode]	wscn:InvalidArgs
[Reason]	At least one input argument is invalid
[Detail]	*The invalid argument*

6.1.1.3. OperationFailed

This fault may be returned if the current state of the service prevents invoking that action.

[Code]	soap:Receiver
[Subcode]	wscn:OperationFailed
[Reason]	Service can not perform the requested operation
[Detail]	*None*

6.1.1.4. ServerErrorTemporaryError

This fault is sent when the server experiences a temporary error that occurs while the Scanner processes the operation. The client MAY try the unmodified request again at some later point in time with an expectation that the temporary internal error condition MAY have been cleared. If there is a more specific error defined that applies to a temporary error, such as disk full, that code SHOULD be used.

[Code]	soap:Receiver

[Subcode]	wscn:ServerErrorTemporaryError
[Reason]	The service had an unexpected error
[Detail]	*None*

6.1.1.5. ServerErrorInternalError

This fault is sent when the Scanner encounters an unexpected condition that prevented it from fulfilling the request. This error differs from *ServerErrorTemporaryError* in that it implies a more permanent type of internal error and resending the operation will return the same fault.

[Code]	soap:Receiver
[Subcode]	wscn:ServerErrorInternalError
[Reason]	The service had an unexpected error
[Detail]	*None*

6.2. CreateScanJob

This operation is the main mechanism to prepare a scan device to scan the images available to it. This operation can be initiated in two different ways – either the User selects a destination and pushes the scan button at the device or the User starts an application on the client and acquires an image. Each method will send different arguments in the operation request.

If the scan is initiated from the device via a **ScanAvailableEvent** the operation contains the *ScanIdentifier* from the **ScanAvailableEvent** and also the *DestinationToken* returned during the **Subscribe** operation for the **ScanAvailableEvent**. The request also contains a scan ticket to control the processing of the scan. The values in the *ScanTicket* are the defaults set at the client before the User went to the device to initiate the scan.

If the scan is initiated from an application on the client the request contains a *ScanTicket* to control the processing of the scan. The *ScanIdentifier* and *DestinationToken* are omitted.

The allowed values returned in a **GetScannerElements(***ScannerConfiguration)* (see section 4.3.) indicate the values of the arguments that the Scan Service instance (scanner) supports (see section 4.9.) for a Job creation operation. The scanner performs the following validation in the indicated order:

- If the *Format* is not supported, the Scanner MUST reject the request and return the `ClientErrorDocumentFormatNotSupported` error code.
- If the client (CP) supplies elements that are unsupported or their values are unsupported (except *Format*) the Scanner
  - MUST reject the Job if the *MustHonor* attribute with a value of `1` or `true` is in effect.
  - MUST ignore or substitute with supported values, respectively, if the *MustHonor* attribute with a value of `0` or `false` is in effect
  - The *MustHonor* attribute is in effect if the attribute is supplied at the element level. Note that if the *MustHonor* attribute is not explicitly specified in the element, the default is `0` or `false`.

- If a client (CP) supplies a conflicting combination of elements in the arguments (such as *InputSource* and *Resolution*), the Scanner MUST reject the Job if the *MustHonor* attribute with a value of `1` or `true` is in effect.

The scanner returns a unique *JobId* to identify the job for this service. The scanner generates the *JobId* in an implementation-defined manner. The scanner MUST return values in the range 1 to 2^{31-1}; 0 and negative values are invalid values to be returned as a result of a **CreateScanJob** operation. Furthermore, the scanner MUST NOT re-use values recently assigned, since CPs would confuse such jobs with older jobs. The scanner also returns a unique identifier to be used in subsequent **RetrieveImage** operations associated with this scan job. The device MUST respond to the **CreateScanJob** request as quickly as possible. The scan device should not wait for scanning to begin before sending the **CreateScanJobResponse.**

The client (CP) MUST retrieve the image data from the scan service using one or more **RetrieveImage** operations. Once the device has responded to the **CreateScanJob** request, the client has 60 seconds to send the **RetrieveImage** operation. If the device does not receive a **RetrieveImage** operation within this time, it should abort the job with a JobStateReason of `JobTimedOut`**.** If the job consists of multiple documents this timeout applies between each successive **RetrieveImage** operation also.

6.2.1. Request Elements

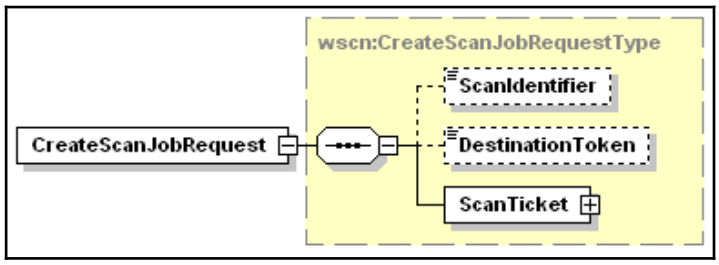


Figure 26 - CreateScanJobRequest Elements

6.2.1.1. ScanIdentifier

This optional element specifies a device specific string the scanner sent in the **ScanAvailableEvent**. This identifier will allow the scanner to make sure the correct client is requesting the scan after a user selected a destination. See section 5.2.5.2. for more details on this element.

6.2.1.2. DestinationToken

This optional element specifies a device specific string the scanner will assign to this client. This string was returned as a part of the *SubscribeResponse* element when the client subscribed to the **ScanAvailableEvent** for the current destination. See section 5.2.3.1. for more details on this element.

6.2.1.3. ScanTicket

This element contains all the settings the client wishes to affect the scan operation. The scan ticket may contain only the processing elements that the client wishes to override in the scanner, or it may contain every possible element supported in the scan service. See section 4.5.2. for more details on the definition of the *ScanTicket*.

6.2.2. Response Elements

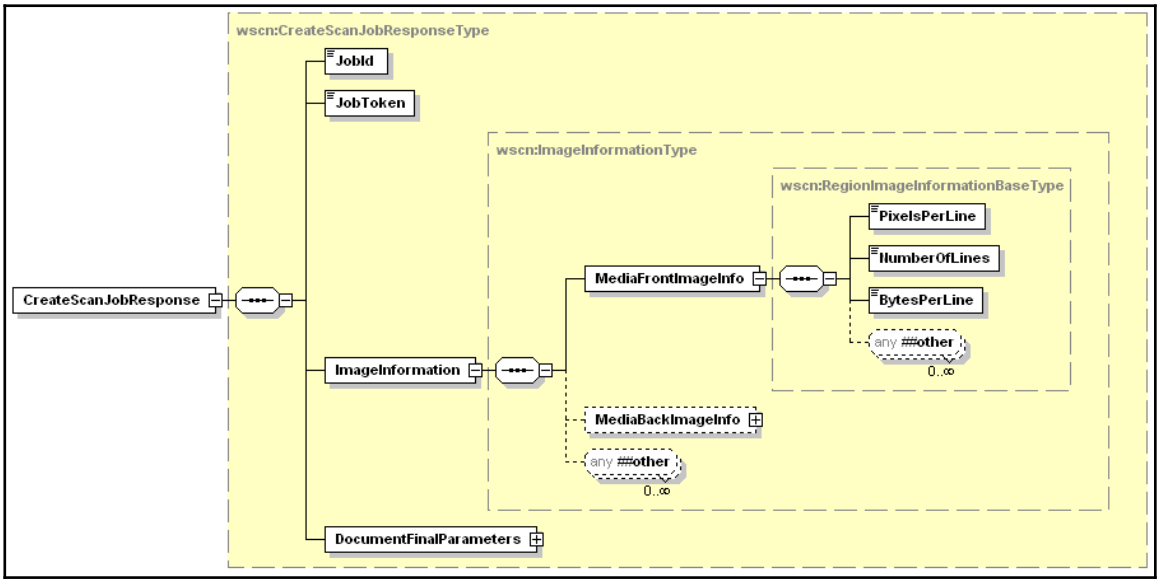


Figure 27 - CreateScanJobResponse Elements

6.2.2.1. JobId

This element is the device created identifier for the new scan job. This Id can be used in subsequent operations like **CancelJob** and **GetJobElements** to modify the job or retrieve job information. This data element is described in Section 4.5.1.1.

6.2.2.2. JobToken

This element is the device created token for the new scan job. This element is paired with the *JobId* to uniquely represent the scan job. It will be passed to the device in the **RetrieveImage** command to allow the scan device to verify the requester actually created the scan job.

6.2.2.3. ImageInformation

This element contains information about the resulting image data from a scan made with the *ScanTicket* that is currently being validated. This data is valuable to scan applications for decoding the image within an image file.

6.2.2.3.1. MediaFrontImageInfo

This element contains the *ImageInformation* for a simplex scan, or the front side of an `ADFDuplex` scan.

6.2.2.3.1.1. PixelsPerLine

This element describes the exact width, in pixels, of the final image that would be scanned from this document side using the current *ScanTicket* settings that are being validated. This width should include rotation and any adjustment done by the scanner on the final images transferred to the client.

Values: `1 – 2147483647`

6.2.2.3.1.2. NumberOfLines

This element describes the exact height, in pixels (or: number of lines) of the final output image that would be generated for the current *ScanTicket*, including rotation and any adjustment the scanner may perform on the scanned image before to transfer it to the client.

Values: `1 – 2147483647`

6.2.2.3.1.3. BytesPerLine

This element indicates how many bytes each scan line will use in the resultant image file. This includes the data pixels and any padding the scanner will add to each scan line. This element is only valid if the requested *Format* is an uncompressed file format. If the file format indicates compression then this element should have a value of `0`.

Values: `0` – `2147483647`

6.2.2.3.2. MediaBackImageInfo

This element contains the *ImageInformation* for the back side of an `ADFDuplex` scan.

6.2.2.4. DocumentFinalParameters

This section of the schema contains the actual *DocumentParameters* which were used by the scan device for this scan job. This data element is described in Section 4.6.1.1.

6.2.3. Example Request – Device Initiated

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
       xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
       xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
       xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
       <wsa:To>AddressofScannerService</wsa:To>
       <wsa:Action>
           http://schemas.microsoft.com/windows/2006/08/wdp/scan/CreateScanJob
       </wsa:Action>
       <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
    </soap:Header>
    <soap:Body>
       <wscn:CreateScanJobRequest>
           <wscn:ScanIdentifier>uuid:12e7a983-1034-5428-d298-0016f11097fa</wscn:ScanIdentifier>
           <wscn:DestinationToken>Dest1234TokenString</wscn:DestinationToken>
           <wscn:ScanTicket>
               <wscn:JobDescription>
                   <wscn:JobName>Photo Scan</wscn:JobName>
                   <wscn:JobOriginatingUserName>RogerSmith</JobOriginatingUserName>
               </wscn:JobDescription>
               <wscn:DocumentParameters>
                   <wscn:Format>jfif</wscn:Format>
                   <wscn:CompressionQualityFactor>45</wscn:CompressionQualityFactor>
                   <wscn:InputSource>Platen</wscn:InputSource>
                   <wscn:ContentType>Auto</wscn:ContentType>
                   <wscn:InputSize>
                       <wscn:DocumentSizeAutoDetect>true</wscn:DocumentSizeAutoDetect>
                   </wscn:InputSize>
                   <wscn:Scaling wscn:MustHonor=”1”>
                       <wscn:ScalingWidth>125</wscn:ScalingWidth>
                       <wscn:ScalingHeight>125</wscn:ScalingHeight>
                   </wscn:Scaling>
                   <wscn:MediaSides>
                       <wscn:MediaFront>
                           <wscn:Resolution wscn:MustHonor=”1”>
                               <wscn:Width>300</wscn:Width>
                               <wscn:Height>300</wscn:Height>
                           </wscn:Resolution>
                       </wscn:MediaFront>
                   </wscn:MediaSides>
               </wscn:DocumentParameters>
           </wscn:ScanTicket>
       </wscn:CreateScanJobRequest>
    </soap:Body>
</soap:Envelope>
```

6.2.4. Example Response

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
       xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
       xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
       xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
```

```
    <soap:Header>
        <wsa:To>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/CreateScanJobResponse
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
        <wsa:RelatesTo>uuid:MsgIdOfTheCreateScanJobRequest</wsa:RelatesTo>
    </soap:Header>
    <soap:Body>
        <wscn:CreateScanJobResponse>
            <wscn:JobId>1</wscn:JobId>
            <wscn:JobToken>Job9876TokenString</wscn:JobToken>
            <wscn:ImageInformation>
                <wscn:MediaFrontImageInfo>
                    <wscn:PixelsPerLine>900</wscn:PixelsPerLine>
                    <wscn:NumberOfLines>1500</wscn:NumberOfLines>
                    <wscn:BytesPerLine>113</wscn:BytesPerLine>
                </wscn:MediaFrontImageInfo>
            </wscn:ImageInformation>
            <wscn:DocumentFinalParamters>
                <wscn:Format>jfif</wscn:Format>
                <wscn:CompressionQualityFactor>45</wscn:CompressionQualityFactor>
                <wscn:ImagesToTransfer>0</wscn:ImagesToTransfer>
                <wscn:InputSource>Platen</wscn:InputSource>
                <wscn:ContentType>Auto</wscn:ContentType>
                <wscn:InputSize>
                    <wscn:InputMediaSize>
                        <wscn:Width Override=”true”>8500</wscn:Width>
                        <wscn:Height Override=”true”>11000</wscn:Height>
                    </wscn:InputMediaSize>
                </wscn:InputSize>
                <wscn:Exposure>
                    <wscn:ExposureSettings>
                        <wscn:Contrast UsedDefault=”true”>0</wscn:Contrast>
                        <wscn:Brightness UsedDefault=”true”>0</wscn:Brightness>
                        <wscn:Sharpness UsedDefault=”true”>0</wscn:Sharpness>
                    </wscn:ExposureSettings>
                </wscn:Exposure>
                <wscn:Scaling>
                    <wscn:ScalingWidth>125</wscn:ScalingWidth>
                    <wscn:ScalingHeight>125</wscn:ScalingHeight>
                </wscn:Scaling>
                <wscn:Rotation UsedDefault=”true”>0</wscn:Rotation>
                <wscn:MediaSides>
                    <wscn:MediaFront>
                        <wscn:ScanRegion>
                            <wscn:ScanRegionXOffset UsedDefault=”true”>0</wscn:ScanRegionXOffset>
                            <wscn:ScanRegionYOffset UsedDefault=”true”>0</wscn:ScanRegionYOffset>
                            <wscn:ScanRegionWidth UsedDefault=”true”>8500</wscn:ScanRegionWidth>
                            <wscn:ScanRegionHeight UsedDefault=”true”>11000</wscn:ScanRegionHeight>
                        </wscn:ScanRegion>
                        <wscn:ColorProcessing UsedDefault=”true”>RGB24</wscn:ColorProcessing>
                        <wscn:Resolution>
                            <wscn:Width>300</wscn:Width>
                            <wscn:Height>300</wscn:Height>
                        </wscn:Resolution>
                    </wscn:MediaFront>
                </wscn:MediaSides>
            </wscn:DocumentFinalParamters>
        </wscn:CreateScanJobResponse>
    </soap:Body>
</soap:Envelope>
```

6.2.5. **Example Request – Client Initiated**

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
        <wsa:To>AddressofScannerService</wsa:To>
        <wsa:Action>
```

```
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/CreateScanJob
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
    </soap:Header>
    <soap:Body>
        <wscn:CreateScanJobRequest>
            <wscn:ScanTicket>
                <wscn:JobDescription>
                    <wscn:JobName>Application Scan</wscn:JobName>
                    <wscn:JobOriginatingUserName>RogerSmith</JobOriginatingUserName>
                </wscn:JobDescription>
                <wscn:DocumentParameters>
                    <wscn:Format>xps</wscn:Format>
                    <wscn:ImagesToTransfer>0</wscn:ImagesToTransfer>
                    <wscn:InputSource>ADF</wscn:InputSource>
                    <wscn:ContentType>Auto</wscn:ContentType>
                    <wscn:InputSize>
                        <wscn:DocumentSizeAutoDetect>true</wscn:DocumentSizeAutoDetect>
                    </wscn:InputSize>
                    <wscn:MediaSides>
                        <wscn:MediaFront>
                            <wscn:ColorProcessing>RGB48</wscn:ColorProcessing>
                            <wscn:Resolution>
                                <wscn:Width>1200</wscn:Width>
                            </wscn:Resolution>
                        </wscn:MediaFront>
                    </wscn:MediaSides>
                </wscn:DocumentParameters>
            </wscn:ScanTicket>
        </wscn:CreateScanJobRequest>
    </soap:Body>
</soap:Envelope>
```

6.2.6. Errors

All the Codes described in section 6.1.1. - Common Operation Error Codes could be returned from this operation. The following errors could also be returned from this operation.

6.2.6.1. ServerErrorNotAcceptingJobs

This fault is sent when the server can't accept a new scan job. This could occur because the scanner has been put into service mode, or there is a user intervention condition and all the memory buffers have been exhausted. The client MAY try the unmodified request again at some later point in time with an expectation that the server has become unblocked and the scanner is accepting jobs again.

[Code]	soap:Receiver
[Subcode]	wscn:ServerErrorNotAcceptingJobs
[Reason]	The service is temporarily blocked and can't accept new job or document requests.
[Detail]	*None*

6.2.6.2. ClientErrorFormatNotSupported

This fault is sent when the supplied *Format* value is not supported by the Scanner object.

[Code]	soap:Sender
[Subcode]	wscn:ClientErrorFormatNotSupported

[Reason]	*Format* parameter value not supported
[Detail]	*Optional: Return a list of supported formats. The data in this element should be of type wscn:FormatSupportedType.*

6.2.6.3. ClientErrorInvalidScanIdentifier

This fault is sent when the supplied *ScanIdentifier* value is not currently valid within the scan device.

[Code]	soap:Sender
[Subcode]	Wscn:ClientErrorInvalidScanIdentifier
[Reason]	*ScanIdentifier* parameter value not currently valid
[Detail]	*None*

6.2.6.4. ClientErrorInvalidDestinationToken

This fault is sent when the supplied *DestinationToken* value is not valid for this scan device.

[Code]	soap:Sender
[Subcode]	Wscn:ClientErrorInvalidDestinationToken
[Reason]	*DestinationToken* parameter value not currently valid
[Detail]	*None*

6.2.6.5. ClientErrorConflictingRequiredParameters

This fault is sent when the there is a conflict between 2 or more *DocumentParameters* elements which each have the *MustHonor* attribute set to `true`. Using all of the settings supplied with *MustHonor* attributes set to `true` will cause a conflict in the scanner mechanism. This conflict cannot be resolved so the *ScanTicket* is invalid.

[Code]	soap:Sender
[Subcode]	wscn: ClientErrorConflictingRequiredParameters
[Reason]	Multiple elements in the *DocumentParameters* element have *MustHonor* set to `true` but applying all settings causes a conflict in the scanner device.
[Detail]	*None*

6.3. RetrieveImage

This operation is used to actually retrieve the scan data from the device after a scan job has been created.

The scan data is returned as a binary attachment along with the *RetrieveImageResponse* packet. The response will be packaged as a MIME Multipart/Related content type and make use of the new SOAP Message Transmission Optimization Mechanism [MTOM] to efficiently send the binary image data. The example below indicates how to create a valid response.

The number of images returned in the resultant file depends on the combination of the *ImagesToTransfer* element of the *ScanTicket* and the image file *Format* element. If the file *Format* is a single image format then the returned file will always contain a single image, if the file *Format* is a multi-page format then the resultant file will contain as many images as the input source can scan up to the value of *ImagesToTransfer*. In the case of a single image file *Format* and an *ImagesToTransfer* value of 0 or >1, the client will send repeated **RetrieveImage** commands until either the devices replies with a `ClientErrorNoImagesAvailable` fault to one of the operations, or the *ImagesToTransfer* value is met.

The scan device sends the headers and XML part of the **RetrieveImageResponse** as soon as it can. This should be before the timeout of 60 seconds occurs and can be well before the scanner starts writing the binary *ScanData* as the data becomes available, without an imposed timeout restriction. This is the benefit of using Multipart/Related in the MTOM processing. The first part containing the XML for the **RetrieveImageResponse**, including the reference to the 2nd part (binary data block), can be sent back immediately and the http connection will remain open without timing out waiting for the completion of the data transfer. When the complete file is scanned and all *ScanData* for this file is written the current **RetrieveImage** operation ends.

If there is a communication failure during the transfer of the image data the device will abort the job with a *JobStateReason* of `ImageTransferError`.

6.3.1. Request Elements

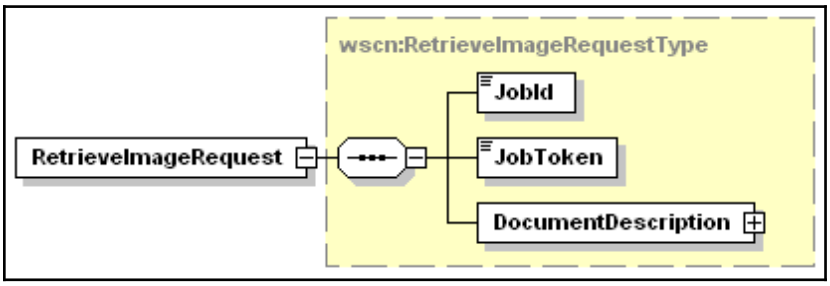


Figure 28 - RetrieveImageRequest Elements

6.3.1.1. JobId

This data element contains the *JobId* with which the current document is associated. This data element is described in Section 4.5.1.1.

6.3.1.2. JobToken

This element is the device created token for the scan job. This data element is described in Section 6.2.2.2..

6.3.1.3. DocumentDescription

This data element contains the description attributes that pertain to the basic creation information of the currently identified *Document*. This data element is described in Section 4.6.1.

6.3.2. Response Elements

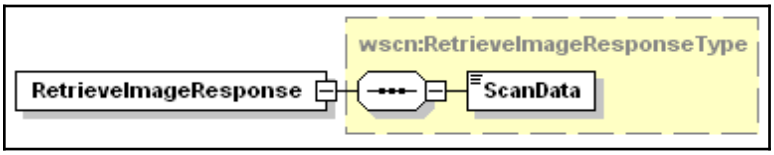


Figure 29 - RetrieveImageResponse Elements

6.3.2.1. ScanData

The data element is the binary data that represents the scanned image is part of the response as a binary attachment to the SOAP Envelope/Body.

6.3.3. Example Request

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
       xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
       xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
       xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
        <wsa:To>AddressofScannerService</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/RetrieveImage
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
    </soap:Header>
    <soap:Body>
        <wscn:RetrieveImageRequest>
            <wscn:JobId>1</wscn:JobId>
            <wscn:JobToken>Job9876TokenString</wscn:JobToken>
            <wscn:DocumentDescription>
                <wscn:DocumentName>Scan001.jpg</DocumentName>
            </wscn:DocumentDescription>
        </wscn:RetrieveImageRequest>
    </soap:Body>
</soap:Envelope>
```

6.3.4. Example Response

```
mime-version: 1.0
Content-Type: Multipart/Related;
    boundary=4aa7d814-adc1-47a2-8e1c-07585b9892a4;
    type=application/xop+xml;
    start=”<14629f74-2047-436c-8046-5cac76d280fc@uuid>”;
    startinfo=”application/soap+xml”

--4aa7d814-adc1-47a2-8e1c-07585b9892a4
Content-Type: application/xop+xml; type="application/soap+xml"
                  charset=UTF-8
Content-Transfer-Encoding: binary
Content-ID: <14629f74-2047-436c-8046-5cac76d280fc@uuid>

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
       xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
       xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
       xmlns:xop="http://www.w3.org/2003/12/xop/include"
       xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
        <wsa:To>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/RetrieveImageResponse
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
        <wsa:RelatesTo>uuid:MsgIdOfTheRetrieveImageRequest</wsa:RelatesTo>
    </soap:Header>
    <soap:Body>
        <wscn:RetrieveImageResponse>
            <wscn:ScanData>
                <xop:Include href=”cid:1c696bd7-005a-48d9-9ee9-9adca11f8892@uuid” />
            </wscn:ScanData>
        </wscn:RetrieveImageResponse>
    </soap:Body>
</soap:Envelope>

--4aa7d814-adc1-47a2-8e1c-07585b9892a4
Content-Type: image/jpeg;
Content-Transfer-Encoding: binary
Content-ID: <1c696bd7-005a-48d9-9ee9-9adca11f8892@uuid>
```

```
Binary Scan Data
--4aa7d814-adc1-47a2-8e1c-07585b9892a4--
```

6.3.5. Errors

All the Codes described in section 6.1.1. - Common Operation Error Codes could be returned from this operation. The following errors could also be returned from this operation.

6.3.5.1. ClientErrorJobIdNotFound

This fault is sent when the scanner can not find a job matching the *JobId* argument (including when the argument is not in the range: 1 to 2^31-1).

[Code]	soap:Sender
[Subcode]	wprt:ClientErrorJobIdNotFound
[Reason]	Specified *JobId* not found
[Detail]	Jobid:*Incorrect JobId*

6.3.5.2. ClientErrorNoImagesAvailable

This fault is sent when the scanner does not have any more images available for the client to retrieve.

[Code]	soap:Sender
[Subcode]	Wscn:ClientErrorNoImagesAvailable
[Reason]	The server has no images available to acquire.
[Detail]	*None*

6.3.5.3. ClientErrorInvalidJobToken

This fault is sent when the supplied *JobToken* value is not valid for this scan *JobId*.

[Code]	soap:Sender
[Subcode]	Wscn:ClientErrorInvalidJobToken
[Reason]	*JobToken* parameter value not valid with *JobId* parameter
[Detail]	*None*

6.3.5.4. ClientErrorJobCancelled

This fault is sent when the current scan job has been cancelled.

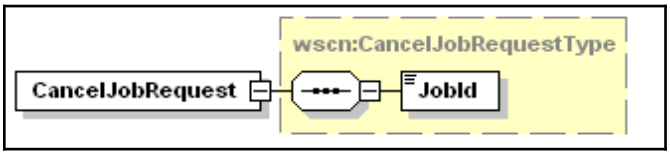
[Code]	soap:Sender
[Subcode	wscn:ClientErrorJobCancelled

]	
[Reason]	The current Scan job has been cancelled.
[Detail]	*None*

2517 6.4. CancelJob

2518 This operation allows a client to cancel a Scan job from the time the job is created up to the time it is completed, canceled
2519 or aborted.

2520 6.4.1. Request Elements



2521

2522 6.4.1.1. JobId

2523 This data element contains the *JobId* which the client is trying to cancel. This data element is described in Section 4.5.1.

2524 6.4.2. Request

```
2525 <?xml version="1.0" encoding="utf-8"?>
2526 <soap:Envelope
2527         xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
2528         xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
2529         xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
2530     <soap:Header>
2531         <wsa:To>AddressofScannerService</wsa:To>
2532         <wsa:Action>
2533             http://schemas.microsoft.com/windows/2006/08/wdp/scan/CancelJob
2534         </wsa:Action>
2535         <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
2536     </soap:Header>
2537     <soap:Body>
2538         <wscn:CancelJobRequest>
2539             <wscn:JobId>1</ wscn::JobId>
2540         </wscn:CancelJobRequest>
2541     </soap:Body>
2542 </soap:Envelope>
```

2543 6.4.3. Response

```
2544 <?xml version="1.0" encoding="utf-8"?>
2545 <soap:Envelope
2546         xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
2547         xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
2548         xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
2549     <soap:Header>
2550         <wsa:To>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa:To>
2551         <wsa:Action>
2552             http://schemas.microsoft.com/windows/2006/08/wdp/scan/CancelJobResponse
2553         </wsa:Action>
2554         <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
2555         <wsa:RelatesTo>uuid:MsgIdOfTheCancelJobRequest</wsa:RelatesTo>
2556     </soap:Header>
2557     <soap:Body>
2558         <wscn:CancelJobResponse/>
2559     </soap:Body>
2560 </soap:Envelope>
```

6.4.4. Errors

All the Codes described in section 6.1.1. - Common Operation Error Codes could be returned from this operation. The following error could also be returned from this operation.

6.4.4.1. ClientErrorJobIdNotFound

This fault is sent when the scanner can not find a job matching the *JobId* argument (including when the argument is not in the range: 1 to 2^31-1).

[Code]	soap:Sender
[Subcode]	wscn:ClientErrorJobIdNotFound
[Reason]	Specified *JobId* not found
[Detail]	Jobid:*Incorrect JobId*

6.4.5. Effect on State

The specified job moves to the `Canceled` state if the Job was `Pending` or `Processing`. It is an error to attempt to cancel a completed or canceled job or to try and cancel any Job the CP does not have rights to.

6.5. ValidateScanTicket

This operation will allow a client to ask the scan device if the settings for a future scan operation are valid. This can be used to validate changes/combinations of settings.

6.5.1. Request Elements

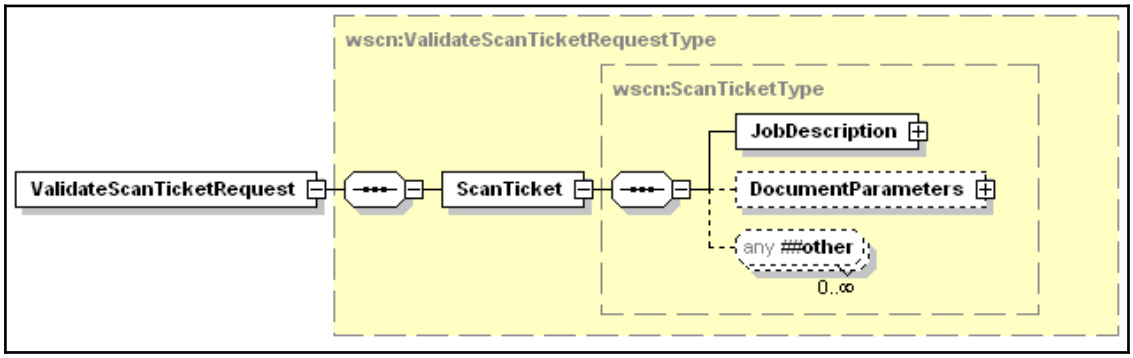


Figure 30 - ValidateScanTicketRequest Elements

6.5.1.1. ScanTicket

This element contains all the settings the client wishes to submit in a future scan operation. The scan ticket may contain only the processing elements that the client wishes to override in the scanner, or it may contain every possible element supported in the scan service. See section 4.8. for more details on the definition of the *ScanTicket*.

6.5.2. Response Elements

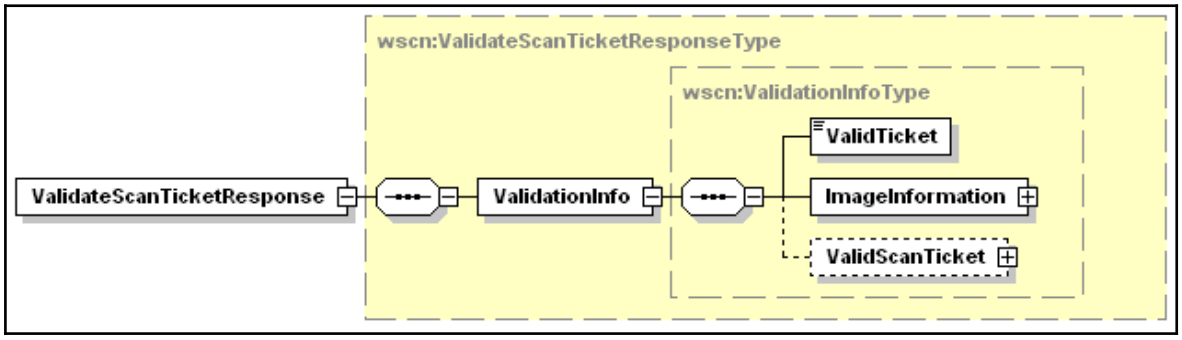


Figure 31 - ValidationInfo elements

6.5.2.1. ValidationInfo

This element defines whether the *ScanTicket* was valid and if not what data was changed to make the ticket valid.

6.5.2.1.1. ValidTicket

This element indicates whether the *ScanTicket* the device received contains all valid settings.

Values: `0`, `1`, `true`, `false`

6.5.2.1.2. ImageInformation

This element contains information about the resulting image data from a scan made with the *ScanTicket* that is currently being validated. This data is valuable to scan applications for decoding the image within an image file. This data element is described in Section 6.2.2.3..

6.5.2.1.3. ValidScanTicket

If the initial *ScanTicket* had some invalid settings and the scan device could fix the problems; this element will contain a new *ScanTicket* with any invalid options changed to valid data. See section 4.8. for more details on the definition of the *ScanTicket*.

6.5.3. Example Request – Valid Ticket

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
        <wsa:To>AddressofScannerService</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/ValidateScanTicket
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
    </soap:Header>
    <soap:Body>
        <wscn:ValidateScanTicketRequest>
            <wscn:ScanTicket>
                <wscn:JobDescription>
                    <wscn:JobName>Photo Scan</wscn:JobName>
                    <wscn:JobOriginatingUserName>RogerSmith</JobOriginatingUserName>
                </wscn:JobDescription>
                <wscn:DocumentParameters>
                    <wscn:Format>dib</wscn:Format>
                    <wscn:InputSource>Platen</wscn:InputSource>
                    <wscn:ContentType>Auto</wscn:ContentType>
                    <wscn:InputSize>
                        <wscn:InputMediaSize>
                            <wscn:Width>3000</wscn:Width>
                            <wscn:Height>5000</wscn:Height>
                        </wscn:InputMediaSize>
                    </wscn:InputSize>
```

```
                    <wscn:Scaling>
                        <wscn:ScalingWidth>125</wscn:ScalingWidth>
                        <wscn:ScalingHeight>125</wscn:ScalingHeight>
                    </wscn:Scaling>
                    <wscn:MediaSides>
                        <wscn:MediaFront>
                            <wscn:ColorProcessing>GrayScale4</wscn:ColorProcessing>
                            <wscn:Resolution>
                                <wscn:Width>300</wscn:Width>
                                <wscn:Height>300</wscn:Height>
                            </wscn:Resolution>
                        <wscn:MediaFront>
                    <wscn:MediaSides>
                </wscn:DocumentParameters>
            </wscn:ScanTicket>
        </wscn:ValidateScanTicketRequest>
    </soap:Body>
</soap:Envelope>
```

6.5.4. **Example Response – Valid Ticket**

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan" >
    <soap:Header>
        <wsa:To>
            http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous
        </wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/ValidateScanTicketResponse
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
        <wsa:RelatesTo>uuid:MsgIdOfTheValidateScanTicketRequest</wsa:RelatesTo>
    </soap:Header>
    <soap:Body>
        <wscn:ValidateScanTicketResponse>
            <wscn:ValidationInfo>
                <wscn:ValidTicket>true</wscn:ScanIdentifier>
                <wscn:ImageInformation>
                    <wscn:MediaFrontImageInfo>
                        <wscn:PixelsPerLine>900</wscn:PixelsPerLine>
                        <wscn:NumberOfLines>1500</wscn:NumberOfLines>
                        <wscn:BytesPerLine>113</wscn:BytesPerLine>
                    </wscn:MediaFrontImageInfo>
                </wscn:ImageInformation>
            </wscn:ValidationInfo>
        </wscn:ValidateScanTicketResponse>
    </soap:Body>
</soap:Envelope>
```

6.5.5. **Example Request – Invalid Ticket**

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan">
    <soap:Header>
        <wsa:To>AddressofScannerService</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/ValidateScanTicket
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
    </soap:Header>
    <soap:Body>
        <wscn:ValidateScanTicketRequest>
            <wscn:ScanTicket>
                <wscn:JobDescription>
                    <wscn:JobName>Photo Scan</wscn:JobName>
                    <wscn:JobOriginatingUserName>RogerSmith</JobOriginatingUserName>
                </wscn:JobDescription>
```

```
                <wscn:DocumentParameters>
                    <wscn:Format>jfif</wscn:Format>
                    <wscn:InputSource>Platen</wscn:InputSource>
                    <wscn:ContentType>Auto</wscn:ContentType>
                    <wscn:InputSize>
                        <wscn:DocumentSizeAutoDetect>true</wscn:DocumentSizeAutoDetect>
                    </wscn:InputSize>
                    <wscn:Scaling>
                        <wscn:ScalingWidth>1250</wscn:ScalingWidth>
                        <wscn:ScalingHeight>1250</wscn:ScalingHeight>
                    </wscn:Scaling>
                    <wscn:MediaSides>
                        <wscn:MediaFront>
                            <wscn:Resolution>
                                <wscn:Width>350</wscn:Width>
                                <wscn:Height>350</wscn:Height>
                            </wscn:Resolution>
                        <wscn:MediaFront>
                    <wscn:MediaSides>
                </wscn:DocumentParameters>
            </wscn:ScanTicket>
        </wscn:ValidateScanTicketRequest>
    </soap:Body>
</soap:Envelope>
```

6.5.6. **Example Response – Invalid Ticket**

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
        <wsa:To>
            http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous
        </wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/ValidateScanTicketResponse
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
        <wsa:RelatesTo>uuid:MsgIdOfTheValidateScanTicketRequest</wsa:RelatesTo>
    </soap:Header>
    <soap:Body>
        <wscn:ValidateScanTicketResponse>
            <wscn:ValidationInfo>
                <wscn:ValidTicket>false</wscn:ValidTicket>
                <wscn:ImageInformation>
                    <wscn:MediaFrontImageInfo>
                        <wscn:PixelsPerLine>0</wscn:PixelsPerLine>
                        <wscn:NumberOfLines>0</wscn:NumberOfLines>
                        <wscn:BytesPerLine>0</wscn:BytesPerLine>
                    </wscn:MediaFrontImageInfo>
                </wscn:ImageInformation>
                <wscn:ValidScanTicket>
                    <wscn:JobDescription>
                        <wscn:JobName>Photo Scan</wscn:JobName>
                        <wscn:JobOriginatingUserName>RogerSmith</JobOriginatingUserName>
                    </wscn:JobDescription>
                    <wscn:DocumentParameters>
                        <wscn:Format>jfif</wscn:Format>
                        <wscn:InputSource>Platen</wscn:InputSource>
                        <wscn:ContentType>Auto</wscn:ContentType>
                        <wscn:InputSize>
                            <wscn:DocumentSizeAutoDetect>true</wscn:DocumentSizeAutoDetect>
                        </wscn:InputSize>
                        <wscn:Scaling>
                            <wscn:ScalingWidth>1000</wscn:ScalingWidth>
                            <wscn:ScalingHeight>1000</wscn:ScalingHeight>
                        </wscn:Scaling>
                        <wscn:MediaSides>
                            <wscn:MediaFront>
                                <wscn:Resolution>
                                    <wscn:Width>300</wscn:Width>
```

```
                                    <wscn:Height>300</wscn:Height>
                                </wscn:Resolution>
                            <wscn:MediaFront>
                        <wscn:MediaSides>
                    </wscn:DocumentParameters>
                </wscn:ValidScanTicket>
            </wscn:ValidationInfo>
        </wscn:ValidateScanTicketResponse>
    </soap:Body>
</soap:Envelope>
```

6.5.7. Errors

All the Codes described in section 6.1.1. - Common Operation Error Codes could be returned from this operation. The following error could also be returned from this operation.

6.5.7.1. ClientErrorConflictingRequiredParameters

This fault is sent when the there is a conflict between 2 or more *DocumentParameters* elements which each have the *MustHonor* attribute set to `true` and using all of the settings supplied with *MustHonor* attributes set to `true` will cause a conflict in the scanner mechanism. This conflict cannot be resolved so the ScanTicket is invalid.

[Code]	soap:Sender
[Subcode]	wscn: ClientErrorConflictingRequiredParameters
[Reason]	Multiple elements in the *DocumentParameters* element have MustHonor set to `true` but applying all settings causes a conflict in the scanner device.
[Detail]	*None*

6.6. GetScannerElements

The **GetScannerElements** operation allows a client (CP) to request information about the scanner. This info includes any part of scanner data accessible at the device root level. Examples of this are the configuration, status, default *ScanTicket*, or IHV extensions to the Scanner schema.

The **GetScannerElements** operation allows a client to discover standard and vendor extended elements.

6.6.1. Request Elements

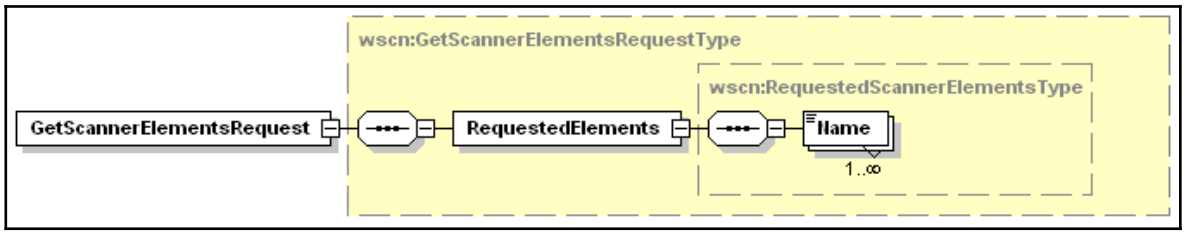


Figure 32 - GetScannerElementsRequest Elements

6.6.1.1. RequestedElements

This data element is a collection of elements that describes the data in which the CP is interested.

6.6.1.2. Name

This data element is a QName that represents a location within the Scanner schema. The keyword represents a top level section of the scanner schema such as the *ScannerConfiguration* or *DefaultScanTicket.*

Allowed Values:

wscn:ScannerDescription– Get all the descriptive information for the scan device.

wscn:ScannerConfiguration– Get all the configuration information for the scan device.

wscn:ScannerStatus – Entire status section including all ActiveConditons and the ConditionHistory

wscn:DefaultScanTicket – The current default values for Job creation and processing

xmlns:VendorSection– A Vendor defined extension to the Scanner Schema

6.6.2. Response Elements

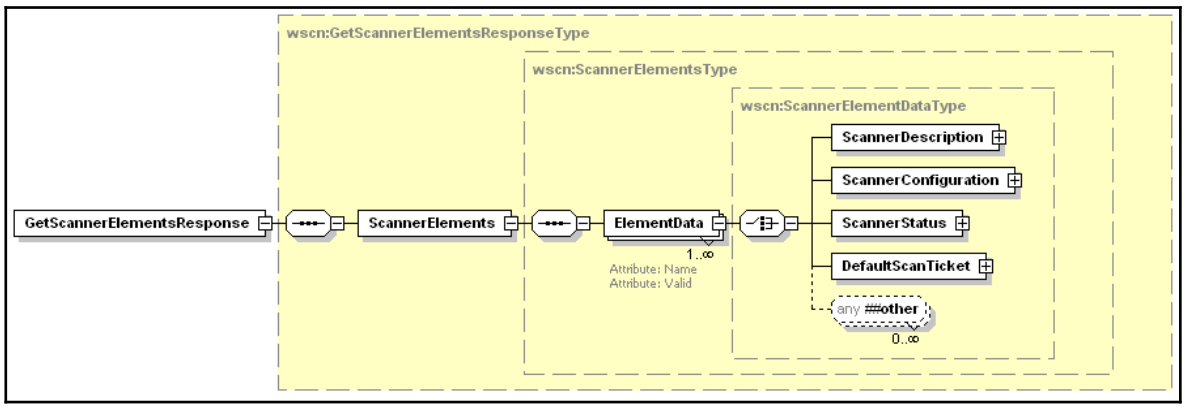


Figure 33 - GetScannerElementsResponse Elements

6.6.2.1. ScannerElements

This data element is a collection of all the responses generated by the scan device for the schema queries in the **GetScannerElements** request.

6.6.2.2. ElementData

This data element contains the data returned for one of the schema requests. There will be the same number of *ElementData* elements in the response as there were *Name* elements in the request.

6.6.2.3. Name

This attribute contains the schema QName used to create the data that is returned in the element.

REQUIRED Attribute

6.6.2.4. Valid

This attribute indicates whether the schema query value was for a valid or invalid schema keyword within the device schema. This attribute will be `false` if the requested schema keyword does not map to a valid schema section in the Scanner schema supported by the device.

REQUIRED Attribute

6.6.3. Request - ScannerDescription

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
        <wsa:To>AddressofScannerService</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/GetScannerElements
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
    </soap:Header>
    <soap:Body>
        <wscn:GetScannerElementsRequest>
```

```
            <wscn:RequestedElements>
                <wscn:Name>wscn:ScannerDescription</wscn:Name>
            </wscn:RequestedElements>
        </wscn:GetScannerElementsRequest>
    </soap:Body>
</soap:Envelope>
```

6.6.4. **Response – ScannerDescription**

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
        <wsa:To>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/GetScannerElementsResponse
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
        <wsa:RelatesTo>uuid:MsgIdOfTheGetScannerElementsRequest</wsa:RelatesTo>
    </soap:Header>
    <soap:Body>
        <wscn:GetScannerElementsResponse>
            <wscn:ScannerElements>
                <wscn:ElementData Name=”wscn:ScannerDescription” Valid=”true”>
                    <wscn:ScannerDescription>
                        <wscn:ScannerName xml:lang="en-AU, en-CA, en-GB, en-US">
                            Accounting Scanner in Copy Room 2
                        </wscn:ScannerName>
                        <wscn:ScannerInfo xml:lang="en-AU, en-CA, en-GB, en-US">
                            Scanner for use of Accounting only
                        </wscn:ScannerInfo>
                        <wscn:ScannerLocation xml:lang="en-AU, en-CA, en-GB, en-US">
                            LA Campus – Building 3
                        </wscn:ScannerLocation>
                    </wscn:ScannerDescription>
                </wscn:ElementData>
            </wscn:ScannerElements>
        </wscn:GetScannerElementsResponse>
    </soap:Body>
</soap:Envelope>
```

6.6.5. **Request - ScannerStatus**

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”
        soap:encodingStyle='http://www.w3.org/2002/12/soap-encoding'>
    <soap:Header>
        <wsa:To>AddressofScannerService</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/GetScannerElements
    </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
    </soap:Header>
    <soap:Body>
        <wscn:GetScannerElementsRequest>
            <wscn:RequestedElements>
                <wscn:Name>wscn:ScannerStatus</wscn:Name>
            </wscn:RequestedElements>
        </wscn:GetScannerElementsRequest>
    </soap:Body>
</soap:Envelope>
```

6.6.6. **Response – ScannerStatus**

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
```

```
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
        <wsa:To>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/GetScannerElementsResponse
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
        <wsa:RelatesTo>uuid:MsgIdOfTheGetScannerElementsRequest</wsa:RelatesTo>
    </soap:Header>
    <soap:Body>
        <wscn:GetScannerElementsResponse>
            <wscn:ScannerElements>
                <wscn:ElementData Name=”wscn:ScannerStatus” Valid=”true”>
                    <wscn:ScannerStatus>
                        <wscn:ScannerCurrentTime>2006-01-26T11:17:00Z</wscn:ScannerCurrentTime>
                        <wscn:ScannerState>Stopped</wscn:ScannerState>
                        <wscn:ScannerStateReasons>
                            <wscn:ScannerStateReason>MediaJam</wscn:ScannerStateReason>
                            <wscn:ScannerStateReason>LampError</wscn:ScannerStateReason>
                        </wscn:ScannerStateReasons>
                        <wscn:ActiveConditions>
                            <wscn:DeviceCondition Id="1384">
                                <wscn:Time>2005-01-26T11:07:00Z</wscn:Time>
                                <wscn:Name>MediaJam</wscn:Name>
                                <wscn:Component>MediaPath</wscn:Component>
                                <wscn:Severity>Critical</wscn:Severity>
                            </wscn:DeviceCondition>
                            <wscn:DeviceCondition Id="534">
                                <wscn:Time>2005-01-26T11:09:12Z</wscn:Time>
                                <wscn:Name>LampError</wscn:Name>
                                <wscn:Component>Platen</wscn:Component>
                                <wscn:Severity>Warning</wscn:Severity>
                            </wscn:DeviceCondition>
                        </wscn:ActiveConditions>
                    </wscn:ScannerStatus>
                </wscn:ElementData>
            </wscn:ScannerElements>
        </wscn:GetScannerElementsResponse>
    </soap:Body>
</soap:Envelope>
```

6.6.7. **Request – ScannerConfiguration and Invalid entry**

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/08/wdp/scan"
        xmlns:ihv=”http://www.example.com/extention”>
    <soap:Header>
        <wsa:To>AddressofScannerService</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/GetScannerElements
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
    </soap:Header>
    <soap:Body>
        <wscn:GetScannerElementsRequest>
            <wscn:RequestedElements>
                <wscn:Name>wscn:ScannerConfiguration</wscn:Name>
                <wscn:Name>ihv:InvalidRequestEntry</wscn:Name>
            </wscn:RequestedElements>
        </wscn:GetScannerElementsRequest>
    </soap:Body>
</soap:Envelope>
```

6.6.8. **Response – ScannerConfiguration and Invalid entry**

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/08/wdp/scan"
```

```
        xmlns:ihv=”http://www.example.com/extention”>
    <soap:Header>
        <wsa:To>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/GetScannerElementsResponse
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
        <wsa:RelatesTo>uuid:MsgIdOfTheGetScannerElementsRequest</wsa:RelatesTo>
    </soap:Header>
    <soap:Body>
        <wscn:GetScannerElementsResponse>
            <wscn:ScannerElements>
                <wscn:ElementData Name=”wscn:ScannerConfiguration” Valid=”true”>
                    <wscn:ScannerConfiguration>
                        <wscn:DeviceSettings>
                            <wscn:FormatsSupported>
                                <wscn:FormatValue>dib</wscn:FormatValue>
                                <wscn:FormatValue>exif</wscn:FormatValue>
                                <wscn:FormatValue>jpeg2k</wscn:FormatValue>
                                <wscn:FormatValue>pdf-a</wscn:FormatValue>
                                <wscn:FormatValue>png</wscn:FormatValue>
                                <wscn:FormatValue>tiff-single-uncompressed</wscn:FormatValue>
                                <wscn:FormatValue>tiff-single-g4</wscn:FormatValue>
                                <wscn:FormatValue>tiff-multi-uncompressed</wscn:FormatValue>
                                <wscn:FormatValue>tiff-multi-g4</wscn:FormatValue>
                                <wscn:FormatValue>xps</wscn:FormatValue>
                            </wscn:FormatsSupported>
                            <wscn:CompressionQualityFactorSupported>
                                <wscn:MinValue>15</wscn:MinValue>
                                <wscn:MaxValue>100</wscn:MaxValue>
                            </wscn:CompressionQualityFactorSupported>
                            <wscn:ContentTypesSupported>
                                <wscn:ContentTypeValue>Auto</wscn:ContentTypeValue>
                                <wscn:ContentTypeValue>Text</wscn:ContentTypeValue>
                                <wscn:ContentTypeValue>Photo</wscn:ContentTypeValue>
                                <wscn:ContentTypeValue>Halftone </wscn:ContentTypeValue>
                                <wscn:ContentTypeValue>Mixed</wscn:ContentTypeValue>
                            </wscn:ContentTypesSupported>
                            <wscn:DocumentSizeAutoDetectSupported>
                                true
                            </wscn:DocumentSizeAutoDetectSupported>
                            <wscn:AutoExposureSupported>true</wscn:AutoExposureSupported>
                            <wscn:BrightnessSupported>true</wscn:BrightnessSupported>
                            <wscn:ContrastSupported>true</wscn:ContrastSupported>
                            <wscn:ScalingRangeSupported>
                                <wscn:ScalingWidth>
                                    <wscn:MinValue>50</wscn:MinValue>
                                    <wscn:MaxValue>500</wscn:MaxValue>
                                </wscn:ScalingWidth>
                                <wscn:ScalingHeight>
                                    <wscn:MinValue>50</wscn:MinValue>
                                    <wscn:MaxValue>500</wscn:MaxValue>
                                </wscn:ScalingHeight>
                            </wscn:ScalingRangeSupported>
                            <wscn:RotationsSupported>
                                <wscn:RotationValue>0</wscn:RotationValue>
                                <wscn:RotationValue>90</wscn:RotationValue>
                                <wscn:RotationValue>180</wscn:RotationValue>
                                <wscn:RotationValue>270</wscn:RotationValue>
                            </wscn:RotationsSupported>
                        </wscn:DeviceSettings>
                        <wscn:Platen>
                            <wscn:PlatenOpticalResolution>
                                <wscn:Width>1200</wscn:Width>
                                <wscn:Height>1200</wscn:Height>
                            </wscn:PlatenOpticalResolution>
                            <wscn:PlatenResolutions>
                                <wscn:Widths>
                                    <wscn:Width>150</wscn:Width>
                                    <wscn:Width>204</wscn:Width>
                                    <wscn:Width>300</wscn:Width>
                                    <wscn:Width>600</wscn:Width>
                                    <wscn:Width>1200</wscn:Width>
```

```
                            </wscn:Widths>
                            <wscn:Heights>
                                <wscn:Height>96</wscn:Height>
                                <wscn:Height>150</wscn:Height>
                                <wscn:Height>204</wscn:Height>
                                <wscn:Height>300</wscn:Height>
                                <wscn:Height>600</wscn:Height>
                                <wscn:Height>900</wscn:Height>
                                <wscn:Height>1200</wscn:Height>
                            </wscn:Heights>
                        </wscn:PlatenResolutions>
                        <wscn:PlatenColor>
                            <wscn:ColorEntry>BlackAndWhite1</wscn:ColorEntry>
                            <wscn:ColorEntry>Grayscale4</wscn:ColorEntry>
                            <wscn:ColorEntry>Grayscale8</wscn:ColorEntry>
                            <wscn:ColorEntry>RGB24</wscn:ColorEntry>
                            <wscn:ColorEntry>RGB48</wscn:ColorEntry>
                            <wscn:ColorEntry>RGBa32</wscn:ColorEntry>
                            <wscn:ColorEntry>RGBa64</wscn:ColorEntry>
                        </wscn:PlatenColor>
                        <wscn:PlatenMinimumSize>
                            <wscn:Width>250</wscn:Width>
                            <wscn:Height>250</wscn:Height>
                        </wscn:PlatenMinimumSize>
                        <wscn:PlatenMaximumSize>
                            <wscn:Width>11000</wscn:Width>
                            <wscn:Height>14000</wscn:Height>
                        </wscn:PlatenMaximumSize>
                    </wscn:Platen>
                    <wscn:ADF>
                        <wscn:ADFSupportsDuplex>false</wscn:ADFSupportsDuplex>
                        <wscn:ADFFront>
                            <wscn:ADFOpticalResolution>
                                <wscn:Width>600</wscn:Width>
                                <wscn:Height>600</wscn:Height>
                            </wscn:ADFOpticalResolution>
                            <wscn:ADFResolutions>
                                <wscn:Widths>
                                    <wscn:Width>150</wscn:Width>
                                    <wscn:Width>204</wscn:Width>
                                    <wscn:Width>300</wscn:Width>
                                    <wscn:Width>600</wscn:Width>
                                </wscn:Widths>
                                <wscn:Heights>
                                    <wscn:Height>96</wscn:Height>
                                    <wscn:Height>150</wscn:Height>
                                    <wscn:Height>204</wscn:Height>
                                    <wscn:Height>300</wscn:Height>
                                    <wscn:Height>600</wscn:Height>
                                </wscn:Heights>
                            </wscn:ADFResolutions>
                            <wscn:ADFColor>
                                <wscn:ColorEntry>BlackAndWhite1</wscn:ColorEntry>
                                <wscn:ColorEntry>Grayscale4</wscn:ColorEntry>
                                <wscn:ColorEntry>RGB24</wscn:ColorEntry>
                            </wscn:ADFColor>
                            <wscn:ADFMinimumSize>
                                <wscn:Width>4000</wscn:Width>
                                <wscn:Height>6000</wscn:Height>
                            </wscn:ADFMinimumSize>
                            <wscn:ADFMaximumSize>
                                <wscn:Width>8500</wscn:Width>
                                <wscn:Height>11000</wscn:Height>
                            </wscn:ADFMaximumSize>
                        </wscn:ADFFront>
                    </wscn:ADF>
                    <wscn:Film>
                        <wscn:FilmScanModesSupported>
                            <wscn:FilmScanModeValue>
                                ColorSlideFilm
                            </wscn:FilmScanModeValue>
                            <wscn:FilmScanModeValue>
                                ColorNegativeFilm
```

```
                                </wscn:FilmScanModeValue>
                                <wscn:FilmScanModeValue>
                                    BlackandWhiteNegativeFilm
                                </wscn:FilmScanModeValue>
                            </wscn:FilmScanModesSupported>
                            <wscn:FilmOpticalResolution>
                                <wscn:Width>600</wscn:Width>
                                <wscn:Height>600</wscn:Height>
                            </wscn:FilmOpticalResolution>
                            <wscn:FilmResolutions>
                                <wscn:Widths>
                                    <wscn:Width>150</wscn:Width>
                                    <wscn:Width>300</wscn:Width>
                                    <wscn:Width>600</wscn:Width>
                                </wscn:Widths>
                                <wscn:Heights>
                                    <wscn:Height>150</wscn:Height>
                                    <wscn:Height>300</wscn:Height>
                                    <wscn:Height>600</wscn:Height>
                                </wscn:Heights>
                            </wscn:FilmResolutions>
                            <wscn:FilmColor>
                                <wscn:ColorEntry>BlackAndWhite1</wscn:ColorEntry>
                                <wscn:ColorEntry>Grayscale4</wscn:ColorEntry>
                                <wscn:ColorEntry>RGB24</wscn:ColorEntry>
                                <wscn:ColorEntry>RGBa32</wscn:ColorEntry>
                            </wscn:FilmColor>
                            <wscn:FilmMinimumSize>
                                <wscn:Width>1378</wscn:Width>
                                <wscn:Height>1378</wscn:Height>
                            </wscn:FilmMinimumSize>
                            <wscn:FilmMaximumSize>
                                <wscn:Width>2756</wscn:Width>
                                <wscn:Height>10000</wscn:Height>
                            </wscn:FilmMaximumSize>
                        </wscn:Film>
                    </wscn:ScannerConfiguration>
                </wscn:ElementData>
                <wscn:ElementData Name=”ihv:InvalidRequestEntry” Valid=”false”/>
            </wscn:ScannerElements>
        </wscn:GetScannerElementsResponse>
    </soap:Body>
</soap:Envelope>
```

6.6.9. **Errors**

All the Codes described in section 6.1.1. - Common Operation Error Codes could be returned from this operation.

6.7. GetJobElements

The **GetJobElements** operation allows a client (CP) to access job-related information of the specified job with a *JobId* from 1 to $2^{\wedge 31}$-1. The information returned complies with the full scan job schema.

If the specified job is not found, the fault *ClientErrorJobIdNotFound* is returned. Any job not found either never existed or has been purged from the system and is no longer known to the Scan Service.

The **GetJobElements** operation allows a client to discover standard and vendor extended elements.

6.7.1. Request Elements

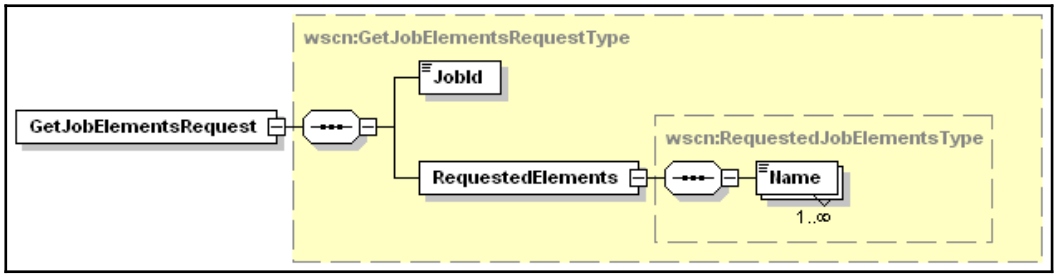


Figure 34 - GetJobElementsRequest Elements

6.7.1.1. JobId

This data element contains the *JobId* of the Job about which the client is request information. This data element is described in Section 4.5.1.1.

6.7.1.2. RequestedElements

This data element is a collection of elements that describes the data in which the CP is interested.

6.7.1.3. Name

This data element is a QName that represents a location within the Job schema. The keyword represents a top level section of the Job schema such as the *JobStatus* or *ScanTicket.*

Allowed Values:

wscn:JobStatus – Get the current *JobStatus* element for the specified Job.
wscn:ScanTicket – Get the *ScanTicket* element for the specified Job
wscn:Documents - Get the *Documents* element for the specified Job
xmlns:*VendorSection*– A Vendor defined extension to the Job Schema

6.7.2. Response Elements

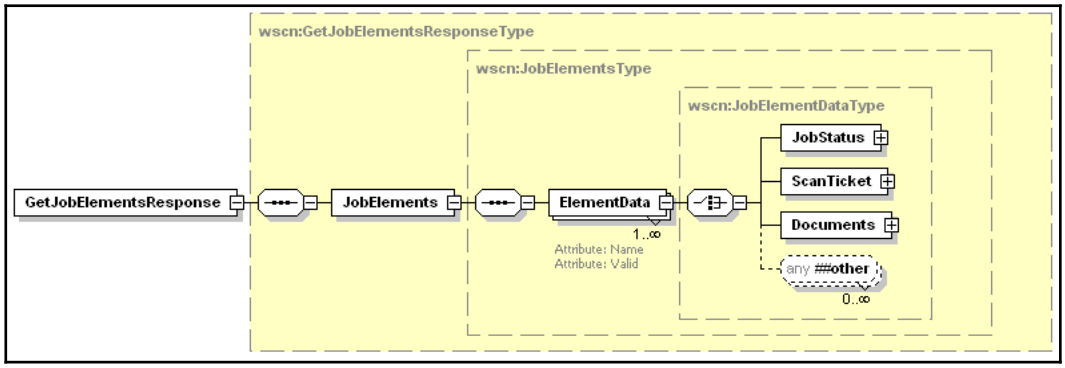


Figure 35 - GetJobElementsResponse Elements

6.7.2.1. JobElements

This data element is a collection of all the responses generated by the scan device for the schema queries in the **GetJobElements** request.

6.7.2.2. ElementData

This data element contains the data returned for one of the schema requests. There will be the same number of *ElementData* elements in the response as there were *Name* elements in the request.

6.7.2.3. Name

This attribute contains the schema QName used to create the data that is returned in the element.

6.7.2.4. Valid

This attribute indicates whether the schema query value was for a valid or invalid schema keyword within the device schema. This attribute will be `false` if the requested schema keyword does not map to a valid schema section in the Job schema supported by the device.

6.7.3. Request

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
        <wsa:To>AddressofScannerService</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/GetJobElements
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
    </soap:Header>
    <soap:Body>
        <wscn:GetJobElements>
            <wscn:JobId>1</wscn:JobId>
            <wscn:RequestedElements>
                <wscn:Name>wscn:JobStatus</wscn:Name>
            </wscn:RequestedElements>
        </wscn:GetJobElements>
    </soap:Body>
</soap:Envelope>
```

6.7.4. Response

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
        <wsa:To>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/GetJobElementsResponse
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
        <wsa:RelatesTo>uuid:MsgIdOfTheGetJobElementsRequest</wsa:RelatesTo>
    </soap:Header>
    <soap:Body>
        <wscn:GetJobElementsResponse>
            <wscn:JobElements>
                <wscn:ElementData Name=”wscn:JobStatus” Valid=”true”>
                    <wscn:JobStatus>
                        <wscn:JobId>1</wscn:JobId>
                        <wscn:JobState>Completed</wscn:JobState>
                        <wscn:ScansCompleted>1</wscn:ScansCompleted>
                    </wscn:JobStatus>
                </wscn:ElementData>
            <wscn:JobElements>
        </wscn:GetJobElementsResponse >
    </soap:Body>
</soap:Envelope>
```

6.7.5. Errors

All the Codes described in section 6.1.1. - Common Operation Error Codes could be returned from this operation. The following error could also be returned from this operation.

6.7.5.1. ClientErrorJobIdNotFound

This fault is sent when the scanner can not find a job matching the *JobId* argument (including when the argument is not in the range: 1 to 2^{31}-1).

[Code]	soap:Sender
[Subcode]	wscn:ClientErrorJobIdNotFound
[Reason]	Specified *JobId* not found
[Detail]	Jobid:*Incorrect JobId*

6.8. GetActiveJobs

The **GetActiveJobs** operation allows a client (CP) to retrieve a list containing a summary of the currently active jobs. The list of Jobs returned contains information such as the *JobId*, *JobOriginatingUserName* and the *JobName*. To retrieve more detailed information on a Job use the **GetJobElements** operation (section 6.7.)

6.8.1. Response Elements

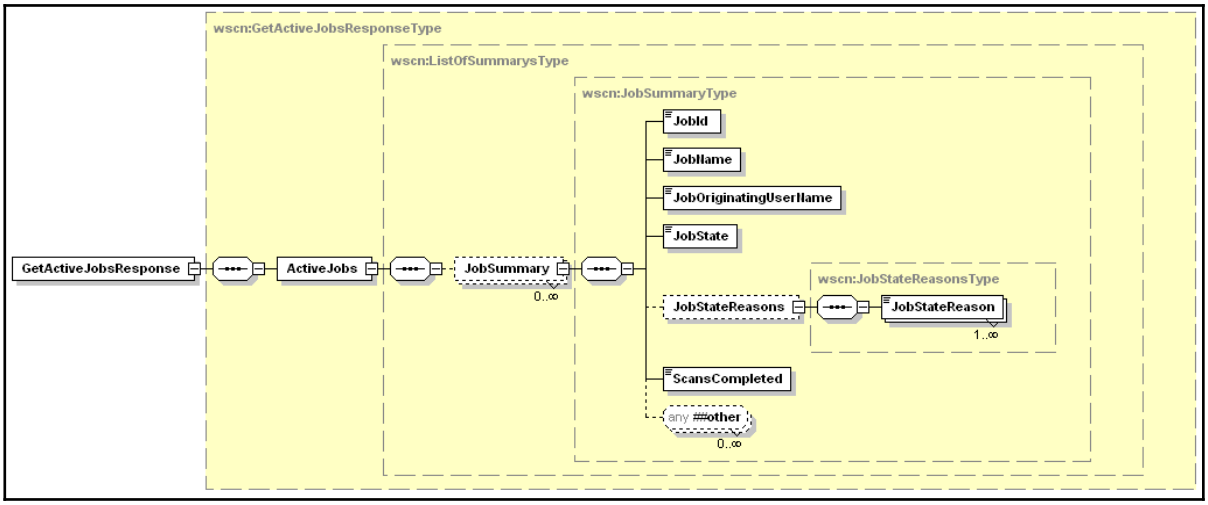


Figure 36 - GetActiveJobsResponse Elements

6.8.1.1. ActiveJobs

This data element is a collection of *JobSummary* elements that describe all the currently active jobs in the scan device.

6.8.1.1.1. JobSummary

This data element collects a subset of information about a Job currently active within the scan device. The individual elements are detailed in the following sections.

6.8.1.1.1.1. JobId

This data element is the *JobId* of the current job entry. This data element is described in Section 4.5.1.1.

6.8.1.1.1.2. JobName

This data element is described in Section 4.5.2.1.1.

6.8.1.1.1.3. JobOriginatingUserName

This data element is described in Section 4.5.2.1.2.

6.8.1.1.1.4. JobState

This data element is described in Section 4.5.1.2.

6.8.1.1.1.5. JobStateReasons

This data element is described in Section 4.5.1.3.

6.8.1.1.1.5.1. JobStateReason

This data element is described in Section 4.5.1.3.1.

6.8.1.1.1.6. ScansCompleted

This data element is described in Section 4.5.1.6.

6.8.2. **Example Request**

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
        <wsa:To>AddressofScannerService</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/GetActiveJobs
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
    </soap:Header>
    <soap:Body>
        <wscn:GetActiveJobsRequest/>
    </soap:Body>
</soap:Envelope>
```

6.8.3. **Example Response – No Active Jobs**

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
        <wsa:To>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/GetActiveJobsResponse
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
        <wsa:RelatesTo>uuid:MsgIdOfTheGetActiveJobsRequest</wsa:RelatesTo>
    </soap:Header>
    <soap:Body>
        <wscn:GetActiveJobsResponse>
            <wscn:ActiveJobs/>
        </wscn:GetActiveJobsResponse>
    </soap:Body>
</soap:Envelope>
```

6.8.4. **Example Response – Two Active Jobs**

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
        <wsa:To>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/GetActiveJobsResponse
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
        <wsa:RelatesTo>uuid:MsgIdOfTheGetActiveJobsRequest</wsa:RelatesTo>
    </soap:Header>
    <soap:Body>
        <wscn:GetActiveJobsResponse>
            <wscn:ActiveJobs>
```

```
                <wscn:JobSummary>
                    <wscn:JobId>1</wscn:JobId>
                    <wscn:JobName>SampleJob 1</wscn:JobName>
                    <wscn:JobOriginatingUserName>Joe.Smith</wscn:JobOriginatingUserName>
                    <wscn:JobState>Processing</wscn:JobState>
                    <wscn:JobStateReasons>
                        <wscn:JobStateReason>JobScanning</wscn:JobStateReason>
                    </wscn:JobStateReasons>
                    <wscn:ScansCompleted>2</wscn:ScansCompleted>
                </wscn:JobSummary>
                <wscn:JobSummary>
                    <wscn:JobId>2</wscn:JobId>
                    <wscn:JobName>Sample Job 2</wscn:JobName>
                    <wscn:JobOriginatingUserName>JaneSmith</wscn:JobOriginatingUserName>
                    <wscn:JobState>Pending</wscn:JobState>
                    <wscn:JobStateReasons>
                        <wscn:JobStateReason>None</wscn:JobStateReason>
                    </wscn:JobStateReasons>
                    <wscn:ScansCompleted>0</wscn:ScansCompleted>
                </wscn:JobSummary>
            </wscn:ActiveJobs>
        </wscn:GetActiveJobsResponse>
    </soap:Body>
</soap:Envelope>
```

6.8.5. Errors

All the Codes described in section 6.1.1. - Common Operation Error Codes could be returned from this operation.

6.9. GetJobHistory

The **GetJobHistory** operation allows a client (CP) to retrieve a list containing a summary of some of the recently previously completed jobs. The list of Jobs returned contains information such as *JobId*, *JobOriginatingUserName* and the *JobName*. To retrieve more detailed information on a Job use the **GetJobElements** operation (section 6.7.)

How much Job history the device keeps is completely device specific. Each device can keep as much or as little history as it deems useful/practical.

6.9.1. Response Elements

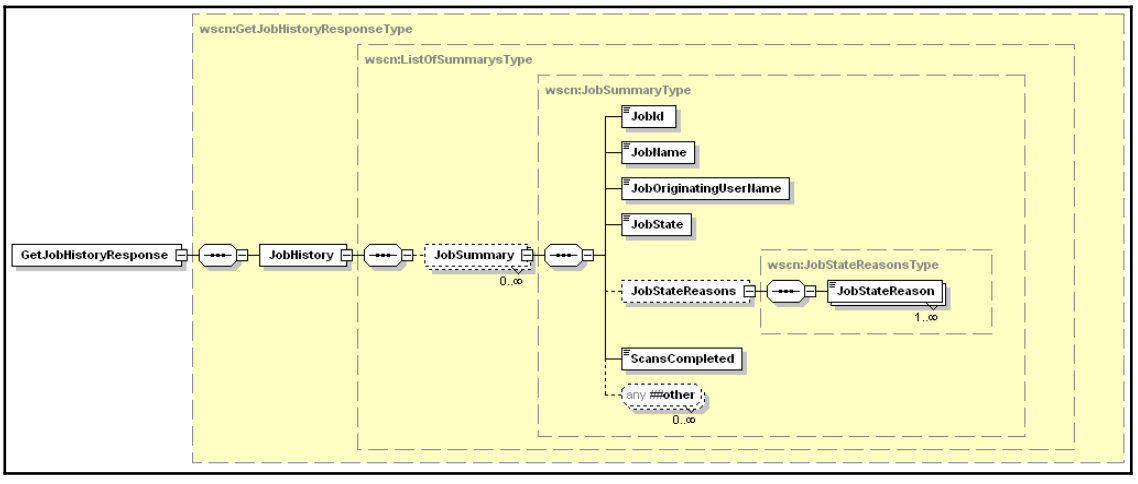


Figure 37 - GetJobHistoryResponse Elements

6.9.1.1. JobHistory

This data element is a collection of *JobSummary* elements that describes the most recently completed jobs in the scan device.

6.9.1.2. JobSummary

This data element collects a subset of information about a recently completed Job within the scan device. This data element is described in Section 6.8.1.1.1..

6.9.2. Example Request

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
        <wsa:To>AddressofScannerService</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/GetJobHistory
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
    </soap:Header>
    <soap:Body>
        <wscn:GetJobHistoryRequest/>
    </soap:Body>
</soap:Envelope>
```

6.9.3. Example Response – No Job History

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
        <wsa:To>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/GetJobHistoryResponse
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
        <wsa:RelatesTo>uuid:MsgIdOfTheGetJobHistoryRequest</wsa:RelatesTo>
    </soap:Header>
    <soap:Body>
        <wscn:GetJobHistoryResponse>
            <wscn:JobHistory/>
        </wscn:GetJobHistoryResponse>
    </soap:Body>
</soap:Envelope>
```

6.9.4. Example Response – 2 Completed Jobs

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
        xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        xmlns:wscn="http://schemas.microsoft.com/windows/2006/01/wdp/scan”>
    <soap:Header>
        <wsa:To>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa:To>
        <wsa:Action>
            http://schemas.microsoft.com/windows/2006/08/wdp/scan/GetJobHistoryResponse
        </wsa:Action>
        <wsa:MessageID>uuid:UniqueMsgId</wsa:MessageID>
        <wsa:RelatesTo>uuid:MsgIdOfTheGetJobHistoryRequest</wsa:RelatesTo>
    </soap:Header>
    <soap:Body>
        <wscn:GetJobHistoryResponse>
            <wscn:JobHistory>
                <wscn:JobSummary>
                    <wscn:JobId>1</wscn:JobId>
                    <wscn:JobName>SampleJob 1</wscn:JobName>
                    <wscn:JobOriginatingUserName>Joe.Smith</wscn:JobOriginatingUserName>
                    <wscn:JobState>Completed</wscn:JobState>
                    <wscn:JobStateReasons>
                        <wscn:JobStateReason>JobCompletedSuccessfully</wscn:JobStateReason>
                    </wscn:JobStateReasons>
                    <wscn:ScansCompleted>4</wscn:ScansCompleted>
```

```
                </wscn:JobSummary>
                <wscn:JobSummary>
                    <wscn:JobId>2</wscn:JobId>
                    <wscn:JobName>Sample Job 2</wscn:JobName>
                    <wscn:JobOriginatingUserName>JaneRogers</wscn:JobOriginatingUserName>
                    <wscn:JobState>Canceled</wscn:JobState>
                    <wscn:JobStateReasons>
                        <wscn:JobStateReason>JobCanceledAtDevice</wscn:JobStateReason>
                    </wscn:JobStateReasons>
                    <wscn:ScansCompleted>1</wscn:ScansCompleted>
                </wscn:JobSummary>
            </wscn:JobHistory>
        </wscn:GetJobHistoryResponse>
    </soap:Body>
</soap:Envelope>
```

6.9.5. **Errors**

All the Codes described in section 6.1.1. - Common Operation Error Codes could be returned from this operation.

6.10. Non-Standard Operations Implemented by a WSD Vendor

To facilitate certification, non-standard operations implemented by WSD vendors MUST be part of a new porttype that extends the porttype defined in this specification.

3463 Appendix A. WSDL Service Description

3464 Scan Device WSDL

```
3465 <wsdl:definitions xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
3466              xmlns:wsoap12="http://schemas.xmlsoap.org/wsdl/soap12/"
3467              xmlns:wsdp="http://schemas.xmlsoap.org/ws/2006/02/devprof"
3468              xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
3469              xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
3470 1.0.xsd"
3471              xmlns:wscn="http://schemas.microsoft.com/windows/2006/08/wdp/scan"
3472              targetNamespace="http://schemas.microsoft.com/windows/2006/08/wdp/scan"
3473              name="WSDScanDevice">
3474
3475     <wsp:Policy wsu:Id="DevicePolicy">
3476         <wsdp:Profile />
3477     </wsp:Policy>
3478
3479     <wsdl:portType name="ScanDeviceType" />
3480
3481     <wsdl:binding name="ScannerSoapBinding" type="wscn:ScanDeviceType" >
3482         <wsoap12:binding style="document" transport="http://schemas.xmlsoap.org/soap/http" />
3483         <wsp:PolicyReference URI="#DevicePolicy" wsdl:required="true" />
3484     </wsdl:binding>
3485
3486 </wsdl:definitions>
```

3487 Scanner Service WSDL

```
3488 <definitions xmlns="http://schemas.xmlsoap.org/wsdl/"
3489              xmlns:wsoap12="http://schemas.xmlsoap.org/wsdl/soap12/"
3490              xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
3491              xmlns:xs="http://www.w3.org/2001/XMLSchema"
3492              xmlns:wsdp="http://schemas.xmlsoap.org/ws/2006/02/devprof"
3493              xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
3494              xmlns:wsd="http://schemas.xmlsoap.org/ws/2005/04/discovery"
3495              xmlns:wse="http://schemas.xmlsoap.org/ws/2004/08/eventing"
3496              xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
3497              xmlns:wscn="http://schemas.microsoft.com/windows/2006/08/wdp/scan"
3498              xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
3499 1.0.xsd"
3500              targetNamespace="http://schemas.microsoft.com/windows/2006/08/wdp/scan"
3501              name="WSDScannerService">
3502
3503     <xs:annotation>
3504         <xs:documentation>
3505             WSD Scanner Service Web Service Description (WSDL) file
3506             Copyright 2005-2006 Microsoft Corporation. All rights reserved
3507         </xs:documentation>
3508     </xs:annotation>
3509
3510     <wsp:Policy wsu:Id="ServicePolicy">
3511         <wsdp:Profile />
3512     </wsp:Policy>
3513
3514     <types>
3515         <xs:schema targetNamespace="http://schemas.microsoft.com/windows/2006/08/wdp/scan">
3516             <xs:include schemaLocation="WDPScan.xsd"/>
3517         </xs:schema>
3518     </types>
3519
3520     <message name="CreateScanJobRequestMsg">
3521         <part name="body" element="wscn:CreateScanJobRequest"/>
3522     </message>
3523     <message name="CreateScanJobResponseMsg">
3524         <part name="body" element="wscn:CreateScanJobResponse"/>
3525     </message>
3526     <message name="RetrieveImageRequestMsg">
3527         <part name="body" element="wscn:RetrieveImageRequest"/>
3528     </message>
3529     <message name="RetrieveImageResponseMsg">
3530         <part name="body" element="wscn:RetrieveImageResponse"/>
3531     </message>
```

```
    <message name="CancelJobRequestMsg">
        <part name="body" element="wscn:CancelJobRequest"/>
    </message>
    <message name="CancelJobResponseMsg">
        <part name="body" element="wscn:CancelJobResponse"/>
    </message>
    <message name="ValidateScanTicketRequestMsg">
        <part name="body" element="wscn:ValidateScanTicketRequest"/>
    </message>
    <message name="ValidateScanTicketResponseMsg">
        <part name="body" element="wscn:ValidateScanTicketResponse"/>
    </message>
    <message name="GetScannerElementsRequestMsg">
        <part name="body" element="wscn:GetScannerElementsRequest"/>
    </message>
    <message name="GetScannerElementsResponseMsg">
        <part name="body" element="wscn:GetScannerElementsResponse"/>
    </message>
    <message name="GetJobElementsRequestMsg">
        <part name="body" element="wscn:GetJobElementsRequest"/>
    </message>
    <message name="GetJobElementsResponseMsg">
        <part name="body" element="wscn:GetJobElementsResponse"/>
    </message>
    <message name="GetActiveJobsRequestMsg">
        <part name="body" element="wscn:GetActiveJobsRequest"/>
    </message>
    <message name="GetActiveJobsResponseMsg">
        <part name="body" element="wscn:GetActiveJobsResponse"/>
    </message>
    <message name="GetJobHistoryRequestMsg">
        <part name="body" element="wscn:GetJobHistoryRequest"/>
    </message>
    <message name="GetJobHistoryResponseMsg">
        <part name="body" element="wscn:GetJobHistoryResponse"/>
    </message>

    <message name="ScanAvailableEventMsg">
        <part name="body" element="wscn:ScanAvailableEvent"/>
    </message>
    <message name="ScannerElementsChangeEventMsg">
        <part name="body" element="wscn:ScannerElementsChangeEvent"/>
    </message>
    <message name="ScannerStatusSummaryEventMsg">
        <part name="body" element="wscn:ScannerStatusSummaryEvent"/>
    </message>
    <message name="ScannerStatusConditionEventMsg">
        <part name="body" element="wscn:ScannerStatusConditionEvent"/>
    </message>
    <message name="ScannerStatusConditionClearedEventMsg">
        <part name="body" element="wscn:ScannerStatusConditionClearedEvent"/>
    </message>
    <message name="JobStatusEventMsg">
        <part name="body" element="wscn:JobStatusEvent"/>
    </message>
    <message name="JobEndStateEventMsg">
        <part name="body" element="wscn:JobEndStateEvent"/>
    </message>

    <portType name="ScannerServiceType" wse:EventSource="true">
        <operation name="CreateScanJob">
            <input message="wscn:CreateScanJobRequestMsg"
                   wsa:Action="http://schemas.microsoft.com/windows/2006/08/wdp/scan/CreateScanJob" />
            <output message="wscn:CreateScanJobResponseMsg"
                   wsa:Action="http://schemas.microsoft.com/windows/2006/08/wdp/scan/
CreateScanJobResponse" />
        </operation>
        <operation name="RetrieveImage">
            <input message="wscn:RetrieveImageRequestMsg"
                   wsa:Action="http://schemas.microsoft.com/windows/2006/08/wdp/scan/RetrieveImage" />
            <output message="wscn:RetrieveImageResponseMsg"
                   wsa:Action="http://schemas.microsoft.com/windows/2006/08/wdp/scan/
RetrieveImageResponse" />
```

```
        </operation>
        <operation name="CancelJob">
            <input message="wscn:CancelJobRequestMsg"
                   wsa:Action="http://schemas.microsoft.com/windows/2006/08/wdp/scan/CancelJob" />
            <output message="wscn:CancelJobResponseMsg"
                   wsa:Action="http://schemas.microsoft.com/windows/2006/08/wdp/scan/CancelJobResponse"
/>
        </operation>
        <operation name="ValidateScanTicket">
            <input message="wscn:ValidateScanTicketRequestMsg"
                   wsa:Action="http://schemas.microsoft.com/windows/2006/08/wdp/scan/
ValidateScanTicket" />
            <output message="wscn:ValidateScanTicketResponseMsg"
                   wsa:Action="http://schemas.microsoft.com/windows/2006/08/wdp/scan/
ValidateScanTicketResponse" />
        </operation>
        <operation name="GetScannerElements">
            <input message="wscn:GetScannerElementsRequestMsg"
                   wsa:Action="http://schemas.microsoft.com/windows/2006/08/wdp/scan/
GetScannerElements" />
            <output message="wscn:GetScannerElementsResponseMsg"
                   wsa:Action="http://schemas.microsoft.com/windows/2006/08/wdp/scan/
GetScannerElementsResponse" />
        </operation>
        <operation name="GetJobElements">
            <input message="wscn:GetJobElementsRequestMsg"
                   wsa:Action="http://schemas.microsoft.com/windows/2006/08/wdp/scan/GetJobElements" />
            <output message="wscn:GetJobElementsResponseMsg"
                   wsa:Action="http://schemas.microsoft.com/windows/2006/08/wdp/scan/
GetJobElementsResponse" />
        </operation>
        <operation name="GetActiveJobs">
            <input message="wscn:GetActiveJobsRequestMsg"
                   wsa:Action="http://schemas.microsoft.com/windows/2006/08/wdp/scan/GetActiveJobs" />
            <output message="wscn:GetActiveJobsResponseMsg"
                   wsa:Action="http://schemas.microsoft.com/windows/2006/08/wdp/scan/
GetActiveJobsResponse" />
        </operation>
        <operation name="GetJobHistory">
            <input message="wscn:GetJobHistoryRequestMsg"
                   wsa:Action="http://schemas.microsoft.com/windows/2006/08/wdp/scan/GetJobHistory" />
            <output message="wscn:GetJobHistoryResponseMsg"
                   wsa:Action="http://schemas.microsoft.com/windows/2006/08/wdp/scan/
GetJobHistoryResponse" />
        </operation>
        <operation name="ScanAvailableEvent">
            <output message="wscn:ScanAvailableEventMsg"
                   wsa:Action="http://schemas.microsoft.com/windows/2006/08/wdp/scan/
ScanAvailableEvent" />
        </operation>
        <operation name="ScannerElementsChangeEvent">
            <output message="wscn:ScannerElementsChangeEventMsg"
                   wsa:Action="http://schemas.microsoft.com/windows/2006/08/wdp/scan/
ScannerElementsChangeEvent" />
        </operation>
        <operation name="ScannerStatusSummaryEvent">
            <output message="wscn:ScannerStatusSummaryEventMsg"
                   wsa:Action="http://schemas.microsoft.com/windows/2006/08/wdp/scan/
ScannerStatusSummaryEvent" />
        </operation>
        <operation name="ScannerStatusConditionEvent">
            <output message="wscn:ScannerStatusConditionEventMsg"
                   wsa:Action="http://schemas.microsoft.com/windows/2006/08/wdp/scan/
ScannerStatusConditionEvent" />
        </operation>
        <operation name="ScannerStatusConditionClearedEvent">
            <output message="wscn:ScannerStatusConditionClearedEventMsg"
                   wsa:Action="http://schemas.microsoft.com/windows/2006/08/wdp/scan/
ScannerStatusConditionClearedEvent" />
        </operation>
        <operation name="JobStatusEvent">
            <output message="wscn:JobStatusEventMsg"
                   wsa:Action="http://schemas.microsoft.com/windows/2006/08/wdp/scan/JobStatusEvent" />
```

```
        </operation>
        <operation name="JobEndStateEvent">
            <output message="wscn:JobEndStateEventMsg"
                    wsa:Action="http://schemas.microsoft.com/windows/2006/08/wdp/scan/
JobEndStateEvent" />
        </operation>
    </portType>
    <binding name="ScannerServiceBinding" type="wscn:ScannerServiceType">
        <wsoap12:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
        <wsp:PolicyReference URI="#ServicePolicy" />
        <operation name="CreateScanJob">
            <wsoap12:operation
                soapAction="http://schemas.microsoft.com/windows/2006/08/wdp/scan/CreateScanJob"
                soapActionRequired="true" />
            <input>
                <wsoap12:body use="literal" />
            </input>
            <output>
                <wsoap12:body use="literal" />
            </output>
        </operation>
        <operation name="RetrieveImage">
            <wsoap12:operation
                soapAction="http://schemas.microsoft.com/windows/2006/08/wdp/scan/RetrieveImage"
                soapActionRequired="true" />
            <input>
                <wsoap12:body use="literal" />
            </input>
            <output>
                <wsoap12:body use="literal" />
            </output>
        </operation>
        <operation name="CancelJob">
            <wsoap12:operation
                soapAction="http://schemas.microsoft.com/windows/2006/08/wdp/scan/CancelJob"
                soapActionRequired="true" />
            <input>
                <wsoap12:body use="literal" />
            </input>
            <output>
                <wsoap12:body use="literal" />
            </output>
        </operation>
        <operation name="ValidateScanTicket">
            <wsoap12:operation
                soapAction="http://schemas.microsoft.com/windows/2006/08/wdp/scan/ValidateScanTicket"
                soapActionRequired="true" />
            <input>
                <wsoap12:body use="literal" />
            </input>
            <output>
                <wsoap12:body use="literal" />
            </output>
        </operation>
        <operation name="GetScannerElements">
            <wsoap12:operation
                soapAction="http://schemas.microsoft.com/windows/2006/08/wdp/scan/GetScannerElements"
                soapActionRequired="true" />
            <input>
                <wsoap12:body use="literal" />
            </input>
            <output>
                <wsoap12:body use="literal" />
            </output>
        </operation>
        <operation name="GetJobElements">
            <wsoap12:operation
                soapAction="http://schemas.microsoft.com/windows/2006/08/wdp/scan/GetJobElements"
                soapActionRequired="true" />
            <input>
                <wsoap12:body use="literal" />
            </input>
            <output>
```

```
                <wsoap12:body use="literal" />
            </output>
        </operation>
        <operation name="GetActiveJobs">
            <wsoap12:operation
                soapAction="http://schemas.microsoft.com/windows/2006/08/wdp/scan/GetActiveJobs"
                soapActionRequired="true" />
            <input>
                <wsoap12:body use="literal" />
            </input>
            <output>
                <wsoap12:body use="literal" />
            </output>
        </operation>
        <operation name="GetJobHistory">
            <wsoap12:operation
                soapAction="http://schemas.microsoft.com/windows/2006/08/wdp/scan/GetJobHistory"
                soapActionRequired="true" />
            <input>
                <wsoap12:body use="literal" />
            </input>
            <output>
                <wsoap12:body use="literal" />
            </output>
        </operation>
        <operation name="ScanAvailableEvent">
            <wsoap12:operation
                soapAction="http://schemas.microsoft.com/windows/2006/08/wdp/scan/ScanAvailableEvent"
                soapActionRequired="true" />
            <output>
                <wsoap12:body use="literal" />
            </output>
        </operation>
        <operation name="ScannerElementsChangeEvent">
            <wsoap12:operation
                soapAction="http://schemas.microsoft.com/windows/2006/08/wdp/scan/
ScannerElementsChangeEvent"
                soapActionRequired="true" />
            <output>
                <wsoap12:body use="literal" />
            </output>
        </operation>
        <operation name="ScannerStatusSummaryEvent">
            <wsoap12:operation
                soapAction="http://schemas.microsoft.com/windows/2006/08/wdp/scan/
ScannerStatusSummaryEvent"
                soapActionRequired="true" />
            <output>
                <wsoap12:body use="literal" />
            </output>
        </operation>
        <operation name="ScannerStatusConditionEvent">
            <wsoap12:operation
                soapAction="http://schemas.microsoft.com/windows/2006/08/wdp/scan/
ScannerStatusConditionEvent"
                soapActionRequired="true" />
            <output>
                <wsoap12:body use="literal" />
            </output>
        </operation>
        <operation name="ScannerStatusConditionClearedEvent">
            <wsoap12:operation
                soapAction="http://schemas.microsoft.com/windows/2006/08/wdp/scan/
ScannerStatusConditionClearedEvent"
                soapActionRequired="true" />
            <output>
                <wsoap12:body use="literal" />
            </output>
        </operation>
        <operation name="JobStatusEvent">
            <wsoap12:operation
                soapAction="http://schemas.microsoft.com/windows/2006/08/wdp/scan/JobStatusEvent"
                soapActionRequired="true" />
```

```
            <output>
                <wsoap12:body use="literal" />
            </output>
        </operation>
        <operation name="JobEndStateEvent">
            <wsoap12:operation
                soapAction="http://schemas.microsoft.com/windows/2006/08/wdp/scan/JobEndStateEvent"
                soapActionRequired="true" />
            <output>
                <wsoap12:body use="literal" />
            </output>
        </operation>
    </binding>
    <service name="ScannerService">
        <port name="ScannerPort" binding="wscn:ScannerServiceBinding">
            <wsoap12:address location="http://localhost/ScannerService/"/>
        </port>
    </service>
</definitions>
```

Appendix B. Windows Vista™ Support Requirements

1. PnP-X installation support

Installation of WSD Scan devices on Windows Vista™ takes advantage of a new feature called PNP-X. This enables networked devices to take part in the normal PnP experience just like locally connected devices. To enable this support a device that complies with [DEVICE] needs to add a specific XML element to their *Relationship* section of the Device Metadata. This element will allow the inbox WSD Scan device driver to be loaded and kick off the install process for the WSD Scan device and the correct scanner driver.

Within the extensiability area of the *wsdp:Relationship/wsdp:Hosted* element the metadata should include a *CompatibleId* element as defined in [PNPX].

1.1. PnP-X Namespace

The current required namespace for all PNP-X related elements is:
http://schemas.microsoft.com/windows/pnpx/2005/10

1.2. PnP-X CompatibleId definition and value

The current definition of the CompatibleId element is:

```
<xs:element name="CompatibleId" type="tns:CompatibleIdType" >
    <xs:annotation>
        <xs:documentation xml:lang="en" >
            Used as the Compatible ID for INF file matching for devices.  The length
            of this string must not exceed 196 wide characters (392 bytes).  To specify
            more than one CompatibleID, separate them with a space character.  For example:
            PNPX_SampleService_CPID_1 PNPX_SampleService_CPID_2 PNPX_SampleService1_CPID_3
        </xs:documentation>
    </xs:annotation>
</xs:element>
<xs:simpleType name="CompatibleIdType" >
    <xs:list itemType="xs:string" />
</xs:simpleType>
```

The value of this field in a supported Metadata section is:
`http://schemas.microsoft.com/windows/2006/08/wdp/scan/ScannerServiceType`

1.3. Sample Device Metadata response

```
<mex:MetadataSection Dialect="http://schemas.xmlsoap.org/ws/2006/02/devprof/relationship">
    <wsdp:Relationship Type="http://schemas.xmlsoap.org/ws/2006/02/devprof/host">
        <wsdp:Hosted>
            <wsa:EndpointReference>
                .
                .
                .
            </wsa:EndpointReference>
            <wsdp:Types xmlns:wscn="http://schemas.microsoft.com/windows/2006/08/wdp/scan">
                wscn:ScannerServiceType
            </wsdp:Types>
            <wsdp:ServiceId>
                .
            </wsdp:ServiceId>
            <pnpx:CompatibleId xmlns:pnpx="http://schemas.microsoft.com/windows/pnpx/2005/10">
                http://schemas.microsoft.com/windows/2006/08/wdp/scan/ScannerServiceType
            </pnpx:CompatibleId>
        </wsdp:Hosted>
    </wsdp:Relationship>
</mex:MetadataSection>
```

2. Network Explorer category support

Another new feature of Windows Vista™ is the Network Explorer. This is a folder that can be used to discover and install network connected devices such as WSD capable printers & scanners. To improve the usability of the Network Explorer every device that is discovered can have belong to one or more Category(s). The way to indicate what category(s) a device belongs to is by adding one or more *DeviceCategory* elements to the device *wsdp:ThisModel* metadata section. The primary category for the device is equal to the first *DeviceCategory* element found. Each successive *DeviceCategory* element will be considered a secondary category. For more information on the definition of the device categories see [PNPX].

Each device should include the *DeviceCategory* element(s) as defined in [PNPX] within the extensiability area of the *wsdp:ThisModel* metadata section

2.1. PnP-X Namespace

The current required namespace for all PNP-X related elements is:
http://schemas.microsoft.com/windows/pnpx/2005/10

2.2. PnP-X Category definition and value

The current definition of the *DeviceCategory* element is:

```
<xs:element name="DeviceCategory" type="tns:DeviceCategoryType" >
    <xs:annotation>
        <xs:documentation>
            Used to identify the category to which the device belongs.  The device
            categories are strings which are defined in the DeviceCategory section
            of [PNPX].  To specify more than one device category, separate them with a
            space character. For example: Scanners Storage
            identifies a device with a primary category of Scanners and a secondary
            category of Storage.  Devices can also specify a device subcategory for
            a more descriptive device category.
                For example: Displays.Television MediaDevices.DVR
            identifies a device which is a Television and a Digital Video recorder.
            The primary device category for this device would be Displays.
        </xs:documentation>
    </xs:annotation>
</xs:element>
<xs:simpleType name="DeviceCategoryType" >
    <xs:list itemType="xs:string" />
</xs:simpleType>
```

2.3. Sample Device Metadata response

```
<mex:MetadataSection Dialect="http://schemas.xmlsoap.org/ws/2006/02/devprof/ThisModel">
    <wsdp:ThisModel>
        <wsdp:Manufacturer xml:lang="en">..</wsdp:Manufacturer>
        <wsdp:ManufacturerUrl>..</wsdp:ManufacturerUrl>
        <wsdp:ModelName xml:lang="en">..</wsdp:ModelName>
        <wsdp:ModelNumber>..</wsdp:ModelNumber>
        <pnpx:DeviceCategory>Scanners</pnpx:DeviceCategory>
    </wsdp:ThisModel>
</mex:MetadataSection>
```

3. Scan Device discoverability

A common user scenario is to search for a particular type of device on the network. Searching for local devices takes advantage of the Multicast capabilities of [DISCOVERY] while the discovery of a distant device used the Unicast, or directed, discovery capabilities. To advertise a service with devices that implement [DISCOVERY] a defined porttype must be exposed by the high level device for each hosted service that wishes to advertise. This porttype contains no operations but implies the device hosts at least one of the advertised service types.

Requirements to support each discovery method are explained in detail in the following sections.

3.1. Local subnet Discovery support

To advertise support for a particular service type a device includes a Scan Device porttype in the wsdisco:*Types* element of the wsdisco:*Hello* and wsdisco:*ProbeMatches* messages. The porttype is defined in the same namespace as the WSD Scanner Service.

3.1.1. WSD Scan Namespace

The current namespace for WSD Scan related elements is:
`http://schemas.microsoft.com/windows/2006/08/wdp/scan`

3.1.2. WSD Scan Device porttype

The Device Type is defined as: "**ScanDeviceType**".

3.1.3. Device Hello example

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
    xmlns:wsdisco="http://schemas.xmlsoap.org/ws/2005/04/discovery"
    xmlns:wsdp="http://schemas.xmlsoap.org/ws/2006/02/devprof"
    xmlns:wscn="http://schemas.microsoft.com/windows/2006/08/wdp/scan">
    <soap:Header>
        <wsa:MessageID>urn:uuid:ac8523ee-6813-498c-8c1b-6272a22353f7</wsa:MessageID>
        <wsa:Action>
            http://schemas.xmlsoap.org/ws/2005/04/discovery/Hello
        </wsa:Action>
        <wsa:To>urn:schemas-xmlsoap-org:ws:2005:04:discovery</wsa:To>
        <wsdisco:AppSequence InstanceId="3094203799" MessageNumber="1" />
    </soap:Header>
    <soap:Body>
        <wsdisco:Hello>
            <wsa:EndpointReference>
                <wsa:Address>uuid:D10000D1-ca45-5fee-a376-112233445555</wsa:Address>
            </wsa:EndpointReference>
            <wsdisco:Types>wsdp:Device wscn:ScanDeviceType</wsdisco:Types>
            <wsdisco:XAddrs>http://IPAddress:80/TestDevice</wsdisco:XAddrs>
            <wsdisco:MetadataVersion>1</wsdisco:MetadataVersion>
        </wsdisco:Hello>
    </soap:Body>
</soap:Envelope>
```

3.1.4. Device ProbeMatches example

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
    xmlns:wsdisco="http://schemas.xmlsoap.org/ws/2005/04/discovery"
    xmlns:wsdp="http://schemas.xmlsoap.org/ws/2006/02/devprof"
    xmlns:wscn="http://schemas.microsoft.com/windows/2006/08/wdp/scan">
    <soap:Header>
        <wsa:MessageID>urn:uuid:147823ee-83f7-498c-8c1b-6272a22353f7</wsa:MessageID>
        <wsa:RelatesTo>urn:uuid:0a6dc791-2be6-4991-9af1-454778a1917a</wsa:RelatesTo>
        <wsa:Action>
            http://schemas.xmlsoap.org/ws/2005/04/discovery/ProbeMatches
        </wsa:Action>
        <wsa:To>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa:To>
        <wsdisco:AppSequence InstanceId="3094203799" MessageNumber="2" />
    </soap:Header>
    <soap:Body>
        <wsdisco:ProbeMatches>
            <wsdisco:ProbeMatch>
                <wsa:EndpointReference>
                    <wsa:Address>uuid:D10000D1-ca45-5fee-a376-112233445555</wsa:Address>
                </wsa:EndpointReference>
                <wsdisco:Types>wsdp:Device wscn:ScanDeviceType</wsdisco:Types>
                <wsdisco:XAddrs>http://IPAddress:80/TestDevice</wsdisco:XAddrs>
                <wsdisco:MetadataVersion>1</wsdisco:MetadataVersion>
            </wsdisco:ProbeMatch>
        </wsdisco:ProbeMatches>
```

```
    </soap:Body>
</soap:Envelope>
```

3.2. Directed Discovery support

A stable discovery URL is needed to enable a Windows Vista™ admin to easily discover a device in an enterprise environment. Having a defined URL allows the client to simply ask for the Hostname or IP Address of the device and then create the stable URL which a device will be listening at for a Probe message via HTTP. This also allows the device to listening at a single URL for discovery messages over HTTP.

3.2.1. Directed Discovery URL

http://*IPAddressOrHostname*:80/StableWSDiscoveryEndpoint/schemas-xmlsoap-org_ws_2005_04_discovery

3.2.2. Secure Directed Discovery URL

https://*IPAddressOrHostname*:443/StableWSDiscoveryEndpoint/schemas-xmlsoap-org_ws_2005_04_discovery

4. WSD Scan WIA Driver required capabilities

The WSD Scanner Service describes a number of input sources, file formats/compression technologies, and color capabilities. Each device and client can decide which of the described capabilities best meet the needs. However to be supported fully by the Windows Vista™ WSD Scan WIA Driver the following minimum requirements must be met.

4.1. InputSource values supported

Each device must support either the *Platen* or *ADF* input source.

Additionally, the device must go through a power-up cycle whenever an *InputSource* is added or removed.

4.2. ColorEntry values required

Each device must support at least one of the following *ColorEntry* values for their *Platen* or *ADF* input source.

- **`BlackAndWhite1`**
- **`Grayscale4`**
- **`Grayscale8`**
- **`RGB24`**

4.3. Document file formats required

Each device must support at least one of the following file formats:

- **`png`**
- **`exif`** – only for `RGB24` data (**)
- **`tiff-single-g4`** – only for `BlackAndWhite1` data (**)
- **`tiff-single-uncompressed`**
- **`dib`**

All supported *ColorEntry* values must be transferable in this file format except where noted.

** - If this is the only required file format supported by the scanner all other color modes that may be supported by the scanner are ignored by the driver.

If the scanner does not support the **`dib`** format and a WIA application client (including the TWAIN Compatibility Layer) requests transferring images in the **`dib`** format the WSD Scan WIA Driver tries to scan in one of the following file formats, in order, if supported by the scanner. The driver then converts the images to the **`dib`** format before transferring them to the application:

1. **`png`**

2. **exif** (if the current color mode is **RGB24**)
3. **tiff-single-g4** (if the current color mode is **BlackAndWhite1**)
4. **tiff-single-uncompressed**

The file formats `xps` and `pdf-a` are used as multi-page image file formats with no specified internal compression for image data. WSD scanners are allowed to choose the compression mode appropriate with the scanned images. The WSD Scan WIA Driver will treat only WIA_COMPRESSION_NONE as supported for these formats, meaning the application cannot change the default compression mode selected by the scanner.

4.4. Duplex scanning support

The WSD Scan WIA Driver will always set the same image acquisition parameters for both sides of a duplex scan using only the *MediaFront* element.