PWG October 22, 2008 MFD Working Group Meeting
At Lexmark, Lexington, KY
Meeting Minutes

Minutes Taker: Nancy Chen.

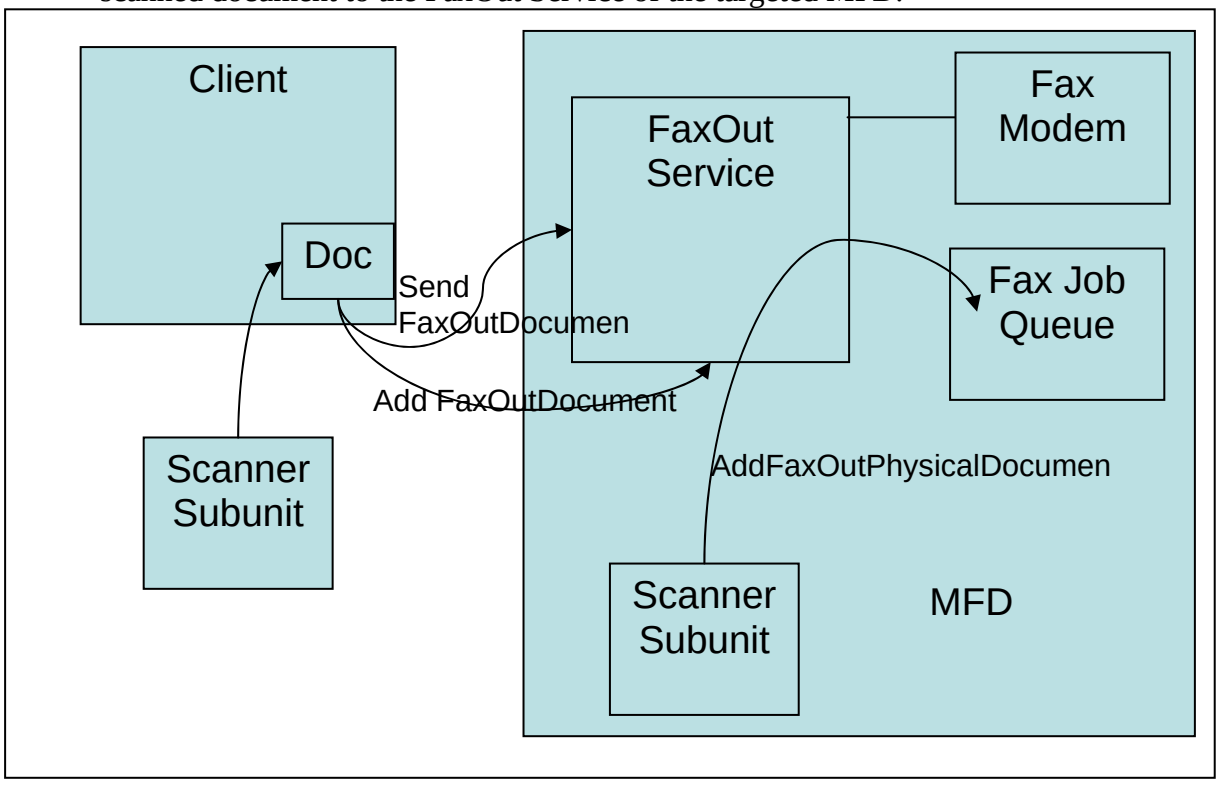
1. **Attendees**:
   Ron Bergman, Ricoh
   Shah Bhatti, Samsung
   Nancy Chen, Okidata
   Lee Ferrell, Canon
   Grant Gilmore, 366 Software
   Ira McDonald, High North (on phone)
   Glen Petrie, Epson
   Andrey Savov, Toshiba
   Ole Skov, MPI Tech
   Jerry Thrasher, Lexmark
   Bill Wagner, TIC
   Dave Whitehead, Lexmark
   Craig Whittle, Sharp
   Peter Zehler, Xerox

2. **Meeting Agenda**:
   9:00am-9:15am : Introductions, Assign Minute Taker(s)
   9:15am-10:15am : Presentation of FaxOut and Model/interface whiteboard discussion
   10:15am-10:30am : Break
   10:30am-11:30am: Continuation of FaxOut discussion
   11:30am-1:00pm : Lunch
   1:00pm-1:30pm: Review of Scan Service Last Call issues
   1:30pm-3:00pm: Detailed review of Resource Service spec, record and resolve issues
   3:00pm-3:15pm: Break
   3:15pm-5:30pm: Continue Resource Service, Next steps
3. **The Straw-man FaxOut Model and Interface Schema** was reviewed.
   - The XML schema (viewable using XMLSpy) for the straw-man FaxOut Service is: ftp://ftp.pwg.org/pub/pwg/mfd/schemas/PWG-SM2-Latest.zip
   - An MFD hosts 0~n FaxOut services, each contain the following elements: DefaultFaxOutTicket, FaxOutServiceCapabilities, FaxOutServiceConfiguration, FaxOutServiceDescription, FaxOutServiceStatus, and a JobTable that contains ActiveJobs and JobHistory.
4. **FaxOut Model and Interface whiteboard discussion**
   - The FaxOut service defined here only faxes digital document. It does not take a physical document as an input.

A new AddFaxOutPhysicalDocument () operation will be added to enable FaxOut of Hardcopy Documents (See below). This operation will enable "traditional" FaxOut which is faxing out a document placed on the MFD.

The service does not stream data out until the entire digital document is available.

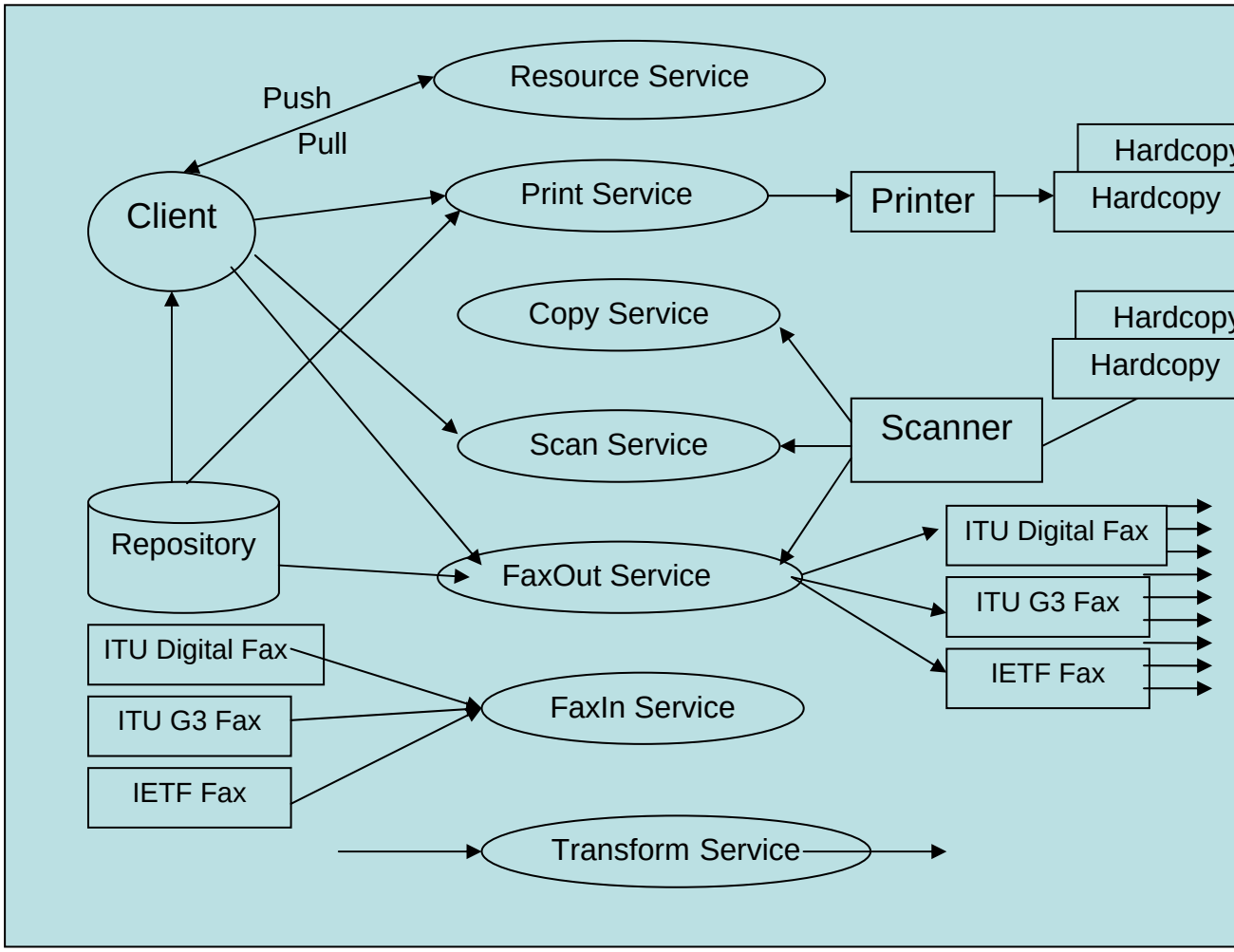
- The diagram below illustrates the top-level concept of a FaxOut Service.
  - A Client can remotely fax out a document via SendFaxOutDocument interface to FaxOut Service. The document can be a reference to the digital document.
  - A FaxOut Service can use the MFD local scanner subunit to scan a physical document and use the AddFaxOutPhysicalDocument interface to add a physically scanned digital document to the Fax Job Queue.
  - A Client can use its locally attached scanner to scan a physical document then remotely use AddFaxOutDocument interface to add the scanned document to the FaxOut Service of the targeted MFD.



  - The FaxOut Service is another user of the MFD local scanner subunit, but not the user of the entire MFD Scan Service. It only uses a subset of Scanner controls, producing G3 fax resolution only for faxing out the document, does not utilized the whole Scan Service.
  - FaxOut Service is not a composition of Scan Service and a Fax service.
  - A MFD Copy Service is very similar to FaxOut Service.

- Below is the diagram showing the system view of the FaxOut Service and all other MFD services and relationships with the subunits.

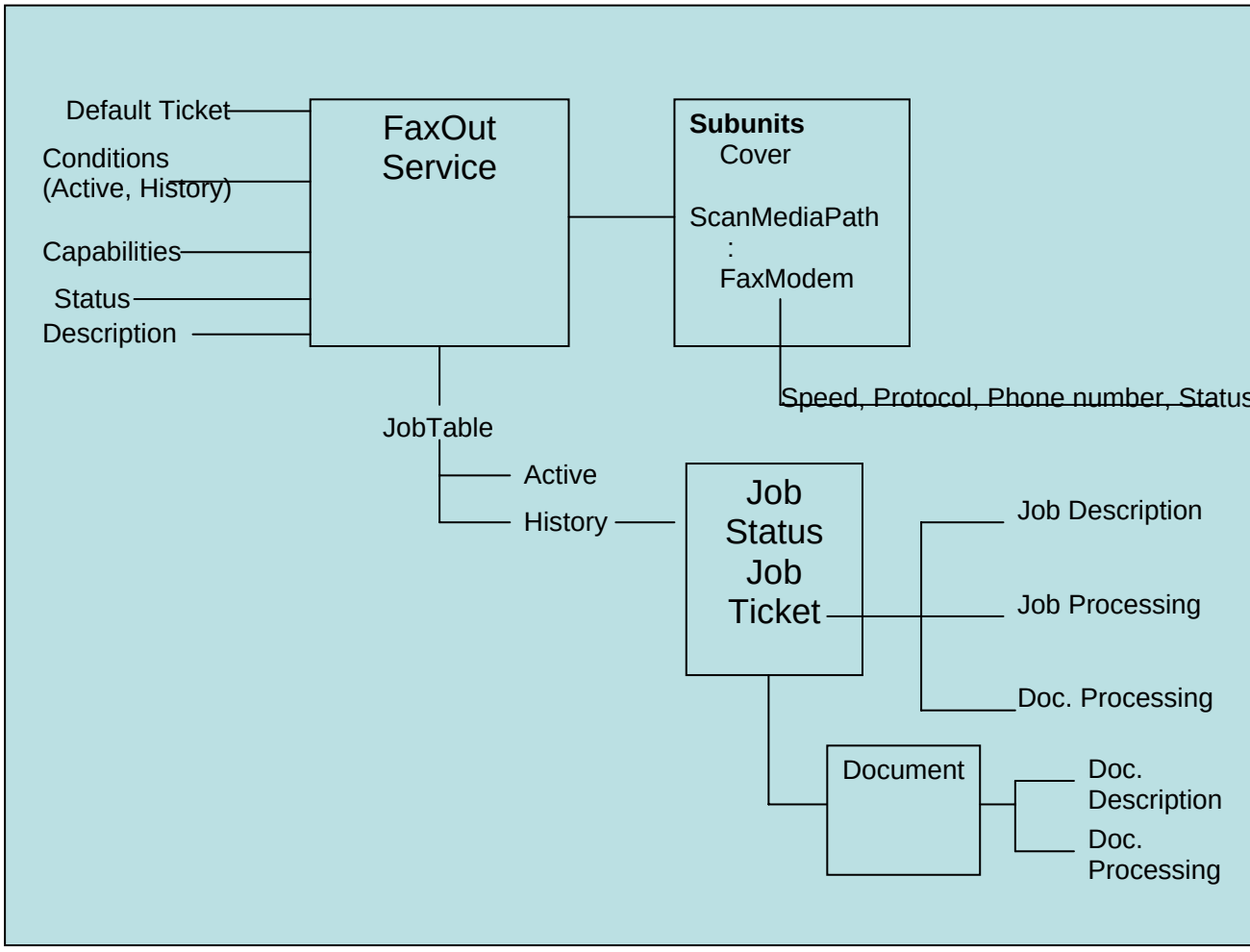
  - We eliminated NetFaxIn and NetFaxOut services which was to service digital fax when ITU did not have digital fax yet, there were only ITU G3 fax and IETF fax. NetFaxIn and NetFaxOut are now included in FaxIn and FaxOut Services respectively, using generic source and destination URIs. Each of the ITU digital fax, ITU G3 fax, and IETF fax subunit can have multiple destination URIs (phone numbers).
  - We discussed on what should be included as basic services – such as e-filing. We concluded that the model does not prevent a vendor to add additional services, keeping only the basic services defined so far is sufficient.



- Below is the top level diagram of the straw-man FaxOut Service.
  - FaxOut Service has a default ticket, Conditions (service specific view of alerts from subunits) for ActiveJobs and JobHistory, service capabilities(document processing, description, job processing capabilities), status, and descriptions, a JobTable that contain

ActiveJobs and JobHistory. In addition to the subunits covered in Scan Service, Fax service has FaxModem subunit. The properties of the FaxModem include speed, protocol, phone number (associated with each job), status. There may be multiple FaxModems available to a FaxOut Service, each modem can dial up a set of phone numbers. It is implementer's decision on which FaxModem should be used for dialing up which phone numbers for each job. There is no way the standard can capture the specific FaxModem selection rule that each vendor implements. FaxModem is associated with FaxOut Service, but each job is not associated with a specific FaxModem. The Description element of FaxOut Service should have a "default FaxModem" element that is used whenever there is a malfunctioning modem, no FaxModem is selected or "auto select" on FaxModem is choosed. The default modem is choosed by the Administrator. There is a status element for every subunit.

- Ira McDonald will investigate any RFC exists for FaxModem and what properties of FaxModem should be included from the RFC.
- Address book should be treated as a resource that is handled by Resource Service, can be used by a FaxOut Client.



- JobHistory has Job Status and Job Ticket which contains Job Description, Job Processing, Document Processing properties. Associated with each job is a document which has Document Description, and Document Processing elements.
- FaxOut Job ticket should have a destination list and a status associated with each destination phone number. We need an extension for Job Status to include FaxOut destination list extended with status for each destination phone number.
- RFC 4734 is Modem Fax and Fax Telephony Event (for features of Fax subunit). There is also a Modem MIB RFC 1696 standard.
- Fax header is a function of the FaxModem. This is the cover sheet for FaxOut. Change header to CoverSheetInfo. This should include "Message" as one element. There should also be a "DateTime". "Request for Acknowledge" is included in URI scheme. We should investigate RFC to see whether we should externalize the "Request for Acknowledge".
- Fax destination phone number will be URI to include NetFaxOut service.
- JobOriginatingPhoneNumber is the fax modem number (PSTN fax). This will be changed to JobOriginatingURI, for accommodating NetFax, this is an email address, or a phone number for PSTN fax. This should be a Status element (because it is selected by automata).
- Samsung proposed to add "JobAccountingSheet" which provide information such as the medium used for printing the sheet, when to print the sheet, and log information on the sheet. It should have a JobAccountingType which is "normal", "detail", … This is a log sheet. We don't want to programmatically control the content of the log. In other services, Job History is optional. We need to make Job History as mandatory in FaxOut Service. The Job History should be presented in log record. There should be a flag indicating when a Job History can be legally deleted after the result of the job has been logged. How long a job history should be kept should not be controlled by client, should be a service implementation decision according to what is legally required. When the log flag is set, the Job History is then allowed to be deleted, for legal requirement, so that the Job History is not lost. There should be a retention policy for job history for any service, represented in the service. One idea is to provide a list of tokens for end user to specify job policy settings. This proposal was left open, we will further investigate how to best deal with this issue.
- We need to add a ConfirmationSheet in Job property.
- We need to allow remote access to Job History/log.

5. Scan Service Last Call Comment Review

- The Scan Service specification for the Last Call is: ftp://ftp.pwg.org/pub/pwg/mfd/wd/lcrc-mfdscanmodel10-20080911.pdf
- The Last Call comments and proposed resolutions are documented in this file:

ftp://ftp.pwg.org/pub/pwg/mfd/wd/MFD-Scan-LastCallResolutionComments-20081016.pdf

- GetScanDocumentElements operation is removed from "REQUIRED" operation, due to the fact that a WS-Scan compliant MFD will not have conformant Scan Document elements as they are defined in the PWG Scan Service.
- Vendor may extend the model resulting in a new PWG namespace or as an alternative, add a porttype to extend the existing operations.
- Resolution for AccessMode now reads: "this element corresponds to the access mode property of POSIX file that controls the basic access control policy of the Scan Service object set by the owner. The AccessMode takes precedence of any external access control policy such as ACL as an example.
- For element names different from what used in WS-Scan, but having the same semantics, there will be explanation of why keeping them different in the appendix.
- Another major difference between PWG Scan Service and WS-Scan is that PWG Scan Service is a "push" only scan model, WS-Scan is a "pull" only scan model. The information expert for the destination of scan document is not the scan service, but rather the subscription client. When the client submit a scan job, it register scan destination with the subscription of scan service event so that the client can get notification to pull the document when it's ready for retrieval. For PWG Scan Service it is possible to mimic that behavior by extending the scan destination and implement a WS-Scan specific operation. In job creation, specify the same destination currently defined by WS-Scan.
- International Considerations: Keyword, String (service generated, e.g. state messages), values supplied by administrator or client.
- Enumerated values that define values that are keywords. The value of element that are part of enumeration that represents keyword.
- Can not have a working draft as a normative reference – Production Print. Delete the references in text or move to informative reference.

- The fact that only few comments came back for the Last Call draft but in fact the draft has a whole section for which the entire content was missing, the group thinks there is a need to make sure there are enough members reviewed the draft. The Chairman Peter Zehler will announce on the PWG list to ask people respond that they have reviewed and have no comment.

6. Resource Service Working Draft Review

- The draft version reviewed is: ftp://ftp.pwg.org/pub/pwg/mfd/wd/wd-mfdresourcemodel10-20081012.pdf
- Reviewed the XML Schema of Resource Service model
  - General concept of Resource Service
    - The service performs resource store and retrieval requests and responses. These requests are not the normal sense of jobs in other services such as Scan/Print which go through

a long job processing after a job is created. The life cycle of resource request is very short.

- The elements/properties of the Resource Service were reviewed which have been documented in the working draft and the schema.
- Any time when a get resource request is sent, there is no guarantee that the resource has not been deleted.
- ListResource could get a list of resources, but there maybe resources already been deleted, or resources that the user has no privilege to retrieve the resource data.
- No resource dependency between resources is modeled in the service. Services have complex dependency among resources. Font is dependent of the embedded PDL and the version of the PDL, executable resource is dependent on environment. The representation of execution environment for various resources was rejected previously.
- A user can guess a ResourceID in order to get a resource data, however, unless the user has the permission to access the data, the resource data will not be returned.
- The client of resource service could be print / scan service, …, etc. that requesting template resource for example. An external client such as a template manager could be a client of resource service that requests for storing a pre-configured template for later use by a print/scan… client to submit a job. The resources served by Resource Service are those useful for MFDs, not general resources. The request and response to/from Resource Service is not a type of transactional request and response – no locking mechanism required from request to response. A resource got by one request is not guaranteed to be "gotten" by the next request. Similar to the behavior of a directory, an item in the directory could be moved from one request to the next.
- Pause and ResumeResourceService operations are deleted because the semantics of these operations are only applicable to jobs/transactions, not resources.
- Toshiba proposed to provide get and put multiple resource operations. This will require a sequence of multiple ResourceDescription elements in the request, and it requires correct binding of each ResourceDescription with the correct resource data. We need to be able to specify how this can be done. One use case for these operations is to put/get a font family, all need to be done in one transaction, have success/failure for the one request, not individual resource in the family done separately.
- Action Item: Andrey Savov will modify the Get/PutResourceRequest schema to represent the desired metadata structure for get/put multiple resources in one transactional request; the structure must maintain proper relationship between the metadata and actual resource data.

        - There was a desire to further type different images: watermark,…
        - ResourceInfo can be used for information on how to use the resource after it's stored.
    - Review of Resource Service Operations
        - Toshiba requested to filter the response of ListResources based on ResourceName. ResourceName will be added to the ListResourcesRequest as one of the parameters.
        - PutResource should use MTOM to transport the binary resource data.
        - The response of PutResource and ReplaceResource should return a success status with UnsupportedElements if there is any resource element or the value of an element (e.g. vendor's extension) is unsupported by the service.
        - We need to add the list of supported description elements in the service capabilities.
        - An observation was made that "ReplaceResource" is a very dangerous operation because the resource could be replaced with a completely different Resourcetype as a mistake. The Resource Client MUST supply at least two elements in the ResourceDescription which are ResourceType and ResourceCategory. The service SHALL verify that the replacing resource has the same ResourceType and ResourceCategory as the one currently stored in the repository. For maintaining consistency, the service SHALL never update the CreatorUserName, ResourceCategory, and ResourceType elements of the ResourceDescription of the existing resource.
        - SetResourceElementRequest updates the metadata of the resource; should use ResourceId instead of ResourceStatus. The CreatorUserName, ResourceType and ResourceCategory shall never be updated.
    - Review other parts of Resource Service working draft is postponed, until most participants have fully read the document.
    - We discussed whether it's good idea to include phone book as a resource, allowing browsing / accessing /extracting certain parts of t he phone book. One concern for allowing this is the resource service only retrieve and store binary data, not complex lookup / browsing, extracting the internal data of a resource. We concluded that it's appropriate if the entire address book/phone book is a resource that can be retrieved /stored by the Resource Service. It requires another service to provide browsing/ accessing/ extracting certain parts of the address book/ phone book. It is not appropriate for Scan/Fax Service destination to be an entry in the address /phone book that requires lookup in the book. The model does not preclude a vendor to use an URI to represent an entry in address/phone book using an unregistered vendor specific URI scheme such as addressbook://… which must be declared in URISupported element.

- No operations shall be performed on any stored resource itself other than Get/Put/ListResource. Only when a PhoneBook Service is implemented, then the destination of Scan/Fax can be an entry in the phone book as an URI.
- Action Item: As a follow-up, all should look at defining a AddressBook service as one of the core MFD services.