NewVision
Official Records
System

Electronic Recording

Submitter Guidelines

For PRIA 2.4

NewVision Systems Corporation
50 Locust Avenue
New Canaan, CT 06850
(203) 323-4623

March 2013

About This Guide

This guide presents the basic information necessary to establish a relationship and submit documents for electronic recording when implementing the PRIA 2.4 standard. Older versions of eRecording, while still supported, are described in a different manual.

Who Should Use This Guide

This guide serves as a training manual for vendors of the system. It is assumed that the vendor has general programming knowledge, experience with Web services, and XML knowledge, as well as a detailed understanding of the concepts and standards as promulgated by PRIA.

Table of Contents

1.0 eRecording/Getting Started

There are several pieces of information required of the submitter for establishing a relationship with the Recorder for the submission of electronic recordings. This document describes those that are currently required and serves as a submitter's guide for the interactive process of eRecording.

Business Requirements:
In order to participate in electronic recording with the Recorder's office you will need to:

- Establish an ACH payment or escrow account with the Recorder.
- Provide basic contact information, including backup/alternative emails and telephone numbers.
- Acquire or develop a software package that utilizes and conforms to the PRIA2 xml DTD
- Acquire or develop a software package that interacts with a Web Service to request and receive recording information
- Receive from the Recorder's Office an account number to identify your documents.
- Receive from the Recorder's Office a list of the document types that are currently accepted.
- Successfully complete a testing process.
- Comply with the procedures and rules established by the Recorder's office.

Testing

Initial testing will be conducted to the satisfaction of both parties at a URL address sent with the start- up package.

Production

Upon successful completion of testing, the production URL for the Request and Receiving Web Services will be sent. All eRecording activity will be through these two services.

Information to be supplied by Submitter

- Agent Name
- Mailing Address
- Name of primary and secondary contact person
- Telephone number of primary and secondary contact person
- Fax Number
- email address of primary and secondary contact person
- ACH Account. If a submitter selects ACH payment, he must preauthorize the Recorder to collect ACH debits from his designated bank account for all fees associated with recording. He will be notified by the office when the transaction authorization is complete and will be supplied an Agent Account ID.

Submitter Elections

In addition to the above information, submitters will need to declare if they wish to have the original unstamped documents returned as well as the stamped documents.

2.0 eRecording Process Documentation

The following represents the basic information and the required information necessary to participate in the NewVision eRecording process.

Model 2 and Model 3 documents are accepted. Model 3 documents submitted as PDF will be decoded and translated into a TIF format. If the document is in a TIF format, it must be created at the minimum DPI specified by the Recorder.

The XML requirements of the NewVision eRecording software are based on the published PRIA V2.4 XML DTD and schema.

3.0 The eRecording Process

In the eRecording Process, there are two parties, the submitter and the recorder, communicating through the software components supplied by NewVision.

The Submitter is responsible for:

- Assembling batches of work that comply with the requirements of the NewVision Software and the Recorder
- Generating and tracking the unique "Packet ID" of each submission.
- Sending the batches of work to the Web Service interface designed by NewVision
- Ensuring adequate funds are available for the calculated recording fees.
- Querying the Response Web Service to receive completed recording information or to determine the status of the submitted batches

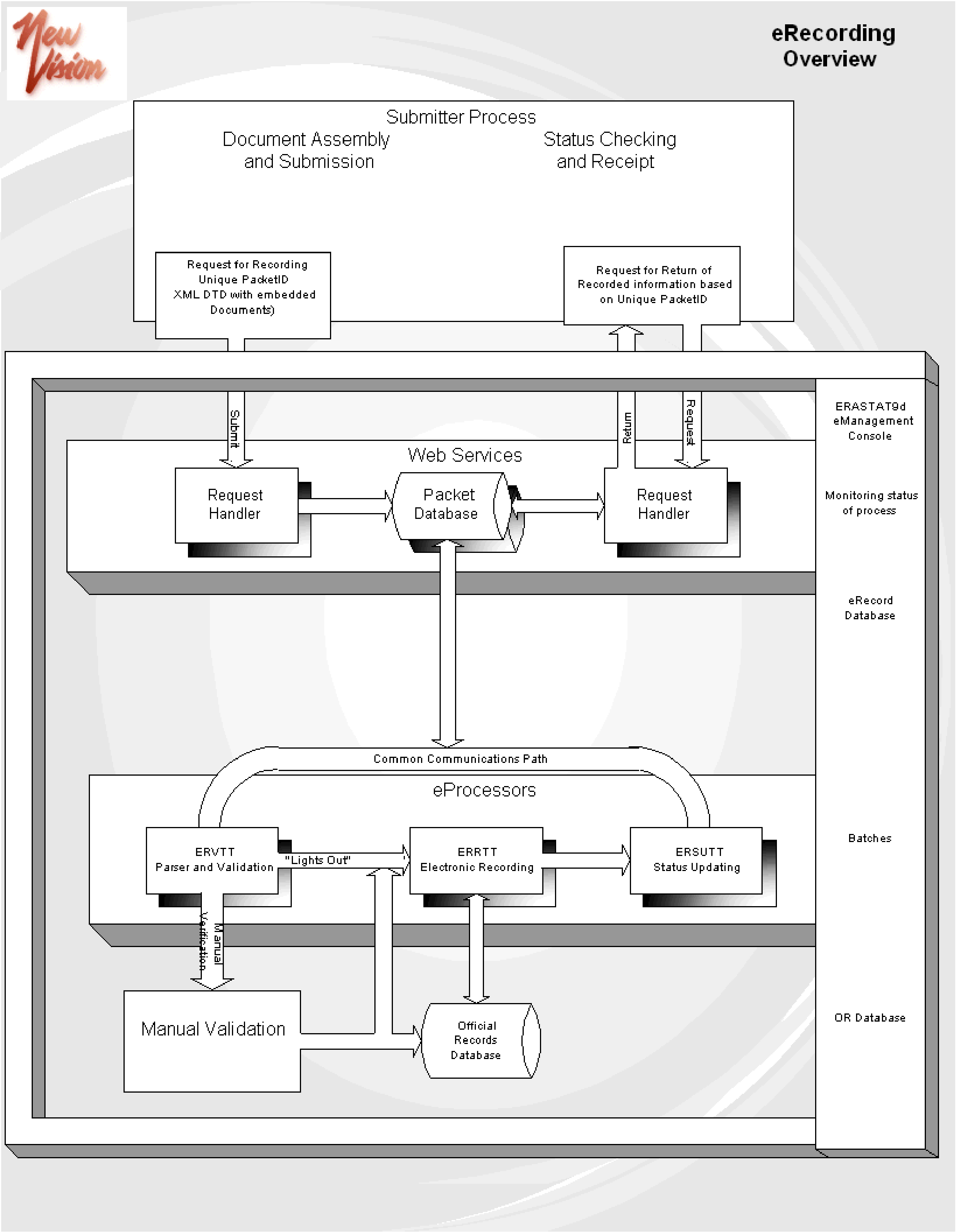
NewVision Components are responsible for:

- Providing Web services through which requests and receipts are queried
- Decoding the incoming batches of work and determining compliance with electronic validation
- Moving the batch along a recorder -determined work flow
- Completing the eRecording of the batch
- Assembling return messages and completed recording information

The Recorder is responsible for:

- Manual validation (if chosen)
- Overall control and monitoring of the process

The following chart provides a logical depiction of these segments and the data flow between them:


eRecording
Overview
Submitter Process
Document Assembly
and Submission
Status Checking
and Receipt
Request for Recording
Unique PacketID
XML DTD with embedded
Documents)
Request for Return of
Recorded information based
on Unique PacketID
Submit
Return
Request
ERASTAT9d
eManagement
Console
Monitoring status
of process
eRecord
Database
Batches
OR Database
Web Services
Request
Handler
Packet
Database
Request
Handler
Common Communications Path
eProcessors
ERVTT
Parser and Validation
"Lights Out"
ERRTT
Electronic Recording
ERSUTT
Status Updating
Manual
Verification
Manual Validation
Official
Records
Database

A. Submitter

The submitter is responsible for assembling all components of a batch into the format acceptable to the New Vision eRecording Software and the Recorders office. The packet may be assembled using either a model 2 or 3 process. The document images may be TIF or PDF. Documents submitted as PDF will be decoded and translated into a TIF format. If the PDF Intellidoc process is used to sign and secure the document, the submitter can elect to receive an unaltered version of the PDF back without the recording information.

1. Assembling the Batches

With an application tool:

Many application development tools support automatic generation of the HTTP requests to a Web service. In the .NET environment, for example, Visual Studio can generate either a Web service reference or a Web service proxy.

Submitters using a tool to generate the Web Service client will need to know the following information:

1. The URL for the e-recording submission Web service is:
   http://*server-ip*/erecord_request_2_4/erecord_request_service.asmx

2. The operation method for submission is "erecord_request", which takes three case sensitive string parameters. The first is the header XML:

   ```
   <XML_FORMAT>
      PRIA_2.4
   </XML_FORMAT>
   <PACKET_FROM>
   (Value supplied by the recorder that represents the Account Name from which funds will be debited.
   This will default to the XML_FORMAT Name if not supplied)
   </PACKET_FROM>
   <PACKET_ID>
      (Submitter-defined unique ID value, It is recommended that GUID can be used to guarantee
      uniqueness)
   </PACKET_ID>
   ```

   The second parameter is the formatted XML document based on the PRIA standards. In the Visual Studio .NET environment, this should be supplied as an XmlNode object (or a derived object such as XmlDocument). In other environments, the XML can be loaded into a string. Note: The Packet ID must also be embedded into the XML document under the element name of <TRANSACTION_IDENTIFIER=_VALUE="*PACKET_ID*"

3. The URL for the e-recording response Web service is:
   http://*server-ip*/erecord_response_2_4/erecord_response_service.asmx

4. The operation method for the response query is “erecord_response”, which takes a single case sensitive string parameter:

```
<XML_FORMAT>
   PRIA 2.4
</XML_FORMAT>
<PACKET_FROM>
   (Must be equal to the PACKET_FROM value submitted)
</PACKET_FROM>
<PACKET_ID>
   (Must be equal to the PACKET_ID value of the Request batch)
</PACKET_ID>
```

The values of PACKET_FROM and PACKET_ID are the key values which link the batch to the submitters. They are returned in all messages as originally sent, and must be duplicated to receive a response from the Request Web Service. **Note: If the Packet_From and Packet_ID value is not an identical match, you will receive the message; “PACKET NOT FOUND”.** It is the responsibility of the sending organization to uniquely identify the packet they are sending. It is recommended that the PACKET ID be a GUID to guarantee uniqueness. It is further suggested that the submitter include a request for status before sending a new packet ID. This is especially useful when there is some doubt as to the need to resend a packet.

Example of a Batch constructed and ready for submission:

```
   <?xml version="1.0" encoding="ISO-8859-1" ?>
 - <REQUEST_GROUP xmlns="http://www.pria.us" PRIAVersionIdentifier="2.4" >
   <REQUESTING_PARTY>
     <ELECTRONIC SIGNATURE />
   </REQUESTING PARTY>
   <RECEIVING_PARTY />
   <SUBMITTING_PARTY />
 - <REQUEST InternalAccountIdentifier="account number">
 - <PRIA_REQUEST>
 - <PACKAGE _Identifier=”submitted-defined-value”>
 - <PRIA_DOCUMENT _Code="D" _PRIAVersionIdentifier="2.4" >
   <GRANTOR _FirstName="SAM" _LastName="SELLER" _MiddleName=”T”
     _NonPersonEntityIndicator="N" />
   <GRANTEE _FirstName="BILL" _LastName="BUYER" " _MiddleName=”X”
     _NonPersonEntityIndicator="N" />
   <GRANTEE NonPersonEntityIndicator="N" _UnparsedName="NEGRE,
     MARTA”/>

 - <PROPERTY>
   <_LEGAL_DESCRIPTION _TextDescription="LOT 7 BLOCK 2" >
      <PLATTED_LAND PropertyLotIdentifier="7" PropertyBlockIdentifier="2" />
  </_LEGAL_DESCRIPTION>
    </PROPERTY>
 - <PARTIES>
 - <_RETURN_TO_PARTY>
   <NON_PERSON_ENTITY_DETAIL />
```

```
    <CONTACT_DETAIL />
      </_RETURN_TO_PARTY>
      </PARTIES>
    <EXECUTION />
    <MORTGAGE_CONSIDERATION />
    <CONSIDERATION _Amount="125000" />
  - <EMBEDDED_FILE _Description="DR219" _MIMEType="PDF"
       _EncodingType="BASE64" _NumberOfPages="1">
    <DOCUMENT> actual binary  document </DOCUMENT>
      </EMBEDDED_FILE>
  - <EMBEDDED_FILE File Description="DEED" MIMEType="PDF"
       FileEncodingType="BASE64" PageCount="1">
    <DOCUMENT>actual binary document DOCUMENT>
      </EMBEDDED_FILE>
      </PRIA_DOCUMENT>
      <PAYMENT MethodType="ACH" />
      </PACKAGE>
    <RECORDING_TRANSACTION_IDENTIFIER _Type="Erecorder"
                _TypeOtherDescription="ERECORD" _Value="packet_id" />
    </PRIA_REQUEST>
    </REQUEST>
    </REQUEST_GROUP>
```

Assembling batches without an application tool: Submitters choosing not to use any of the existing development tools must build a HTTP POST request with a SOAP 1.1 of SOAP 1.2 data message. Examples of this message construction are in appendix A.

2. Sending the Batches to the Web Service

After the batch is assembled, the submitter is responsible for forwarding the batch to the appropriate Request Web service (Test or Production).

The time required to complete the process is largely dependent on one factor; "Are the submitted documents authorized for automatic processing?" This determination is at the discretion of the recording office. Non automatic documents must be evaluated by a clerk to determine if the document should be recorded or returned. Until this is done, a request for status on these documents can only result in an *eValidated* or *eRejected* response.

3. Querying the Response Web Service

Additionally, after a reasonable processing time has elapsed, the submitter must send a request to the appropriate Response Web service (Test or Production) to retrieve the completed document or determine the current status.

eRejected (automatic) and Rejected (manual) batches are considered finalized by the eProcessing system. The cause for the rejection is included in the response message. It is the responsibility of the submitter to correct the problem and submit the batch as a new item.

B. New Vision's eRecording Software

This process is a multipart interactive process. The Web services provide the necessary portal to the outside world of the submitter and function as an intermediary between the submitter and the eProcessing modules. The eProcessing modules provide the cashiering and indexing functions and are the interface to the official records database and the recorder.

1. Web Services: Two similar copies of these modules are installed in the recording office, one for production and one for testing. The URL's to both the testing and production version are supplied by the recording office along with other recorder specific information. Example of what to expect are:

Test: Request service (used for submission)
http://*server-IP*/erecord_request_test_2_4/erecord_request_service.asmx

Response service (used to solicit the status and returned recorded images)
http://*server-IP*/erecord_response_test_2_4/erecord_response_service.asmx

Production:

Request service (used for submission)
http://*server-IP*/erecord_request_2_4/erecord_request_service.asmx

Response service (used to solicit the status and returned recorded images)
http://*server-I*P/erecord_response_2_4/erecord_response_service.asmx

The actual replacement string for the server-IP is the responsibility of the recording office. They may choose to substitute a registered name for the actual IP address. The actual Web services used for test and production are identical except for name and service address.

As new features or standards are adopted, new Web services will be added. The intent is to permit the use of the old while the new is being evaluated and tested. In the examples above, the service names with a suffix of 2_4 represent their version number.

As implied by the forgoing conventions, it is the intent of NewVision to permit all submitters to continue using their initial process while they prepare for, test, and move to the new. However, this opportunity is under the control of the recording office and no length of time guarantees are made.

2. Decoding Batches and Determining Compliance: Batches sent (Request service) are subjected to an extensive list of validation requirements. The batches may be composed of multiple documents as in the case of a mortgage closing. Typical mortgage closings consist of a Satisfaction of Mortgage, a Deed and a new Mortgage. If all three documents are prepared and ready for recording at the same session, they can be included in the same batch. However, ALL documents in a batch must be acceptable or all will be rejected.

Reporting on Submitted Batches. There are two types of messages returned for any response requested (Response Service):

1. Successful Recording - represents a successfully completed and recorded batch. This message contains a PACKAGE in which the recorded information and image is returned.
2. Current Status. The current and possibly final status of a submitted batch. This message always contains a Status element and has an empty PACKAGE.

3. Return Messages

Successful Recording All documents in the batch have been accepted and recorded. This message is based on the PRIA standard DTD which has no Status codes or reporting. However, it is easily recognized because it is the one response that contains a PACKAGE. The package body returns the images of the recorded document and recording data. A sample of this response follows:

```
  <?xml version="1.0" encoding="UTF-8" ?>
- <RESPONSE_GROUP xmlns="http://www.pria.us"
    PRIAVersionIdentifier="2.4">
    <RESPOND_TO_PARTY />
    <RESPONDING_PARTY />
  - <RESPONSE ResponseDateTime="yyyy-mm-ddThh:mm:ss"
        InternalAccountIdentifier="90001">
  - <PRIA_RESPONSE>
    - <PACKAGE _Identifier="7">
      - <PRIA_DOCUMENT _Code="" _Type="D"
            _UniqueIdentifier="14010662"
            PriaVersionIdentifier="2.4">
          <GRANTOR _NonPersonEntityIndicator="N"
            _FirstName="SAM" _MiddleName=""
            _LastName="SELLER" _NameSuffix="" />
          <GRANTEE _NonPersonEntityIndicator="N"
            _FirstName="BILL" _MiddleName=""
            _LastName="BUYER" _NameSuffix="" />
         <GRANTEE NonPersonEntityIndicator="N"
         _UnparsedName="NEGRE, MARTA"/>

        - <PARTIES>
          - <_RETURN_TO_PARTY>
              <NON_PERSON_ENTITY_DETAIL />
              <CONTACT_DETAIL />
            </_RETURN_TO_PARTY>
          </PARTIES>
          <EXECUTION />
        - <RECORDING_ENDORSEMENT
            _InstrumentNumberIdentifier="20050321"
            _PagesCount="1"
      _RecordedDateTime="2006-01-
      24T11:31:26 AM" _Volume="32">
     <_VOLUME_PAGE
        _NumberIdentifier="213" />
   - <_FEES _TotalAmount="760.00">
       <_RECORDING_FEE
          RecordingEndorsementFeeSequence
          Identifier="1"
          RecordingEndorsementFeeDescriptio
          n="DOCUMENT STAMP TAX"
          _Amount="0" />
       <_RECORDING_FEE
          RecordingEndorsementFeeSequence
          Identifier="2"
          RecordingEndorsementFeeDescriptio
          n="DEED DOCUMENT TAX"
          _Amount="750.00" />
       <_RECORDING_FEE
          RecordingEndorsementFeeSequence
          Identifier="3"
          RecordingEndorsementFeeDescriptio
          n="INTANGIBLE TAX"
          _Amount="0" />
       <_RECORDING_FEE
          RecordingEndorsementFeeSequence
          Identifier="4"
          RecordingEndorsementFeeDescriptio
          n="RECORDING FEE"
          _Amount="5.00" />
       <_RECORDING_FEE
          RecordingEndorsementFeeSequence
          Identifier="5"
          RecordingEndorsementFeeDescriptio
          n="TRUST FEE" _Amount="1.00"
          />
       <_RECORDING _FEE
          RecordingEndorsementFeeSequence
          Identifier="6"
          RecordingEndorsementFeeDescriptio
          n="INDEX FEE" _Amount="0" />
       <_RECORDING _FEE
          RecordingEndorsementFeeSequence
          Identifier="7"
          RecordingEndorsementFeeDescriptio
          n="EXTRA FEES" _Amount="4.00"
          />
     </_FEES>
  </RECORDING_ENDORSEMENT>
- <EMBEDDED_FILE FileEncodingType="BASE64"
     MIMEType="TIFF" PageCount="1">
     <DOCUMENT> (Actual Bininary TIF
        document)=
                </DOCUMENT>
              </EMBEDDED_FILE>
            </PRIA_DOCUMENT>
            <PAYMENT _MethodType="ACH"
              _ReferenceIdentifier="receipt number" />
          </PACKAGE>
          <RECORDING_TRANSACTION_IDENTIFIER
            _Value="packet_id" />
        <ORIGINATING_RECORDING_REQUEST
            _Identifier="packet_id"/>>
          <STATUS Code= ‘P’  Name=’Verified and Recorded’>
        </PRIA_RESPONSE>
      </RESPONSE>
    </RESPONSE_GROUP>
```

Status Reporting: All submitted batches that are recognized are tracked through the eRecording process with a status code that represents the location of the batch in the workflow. In many cases the status will only exist for extremely short periods. In others, like those that depend on manual intervention they may exist for longer periods. At any point after a submitted request is recognized, a request to the Receipt Web service will return a response appropriate to the real time batch status.

The basic format of the receipt response message is:

```
<?xml version='1.0' encoding='UTF-8'?>
<RESPONSE_GROUP xmlns='http://www.pria.us' PRIAVersionIdentifier='2.4'>
<RESPOND_TO_PARTY/><RESPONDING_PARTY/>
<RESPONSE>
<PRIA_RESPONSE>
<PACKAGE/>
<STATUS _Code=”s” _Name=”Short message” _Description=’Extended description of
the status”></STATUS>
</PRIA_RESPONSE>
<ELECTRONIC_SIGNATURE/>
</RESPONSE>
</RESPONSE_GROUP>
```

Possible Status Messages: These messages are returned for batches that are currently engaged in one of the steps of processing. Actual responses in BLUE BOLD, the associated variables in red with their system meaning follow:

<STATUS _Code='null' _Name='Received’ _Description= ‘Batch received and pending parsing and electronic validation.’ </STATUS>

Full description: The Request Web Service has acknowledged the receipt of the batch. This status should be of short duration and will change as soon as the eValidater finishes processing the batch. Possible causes for this persisting are:

- Web server is running at full capacity and has a backlog of requests
- eValidation process hasn’t been started, is running at full capacity or has paused.

<STATUS _Code='V' _Name='eValidated’ _Description= ‘Batch has been parsed and is in compliance with automated validation rules.’ </STATUS>

Full Description: The automated parsing program has confirmed compliance with the XML DTD standards and has determined that the submitter and the batch content comply with establish recording processing requirements. This status would be of short duration for automatically recorded documents, but can be expected to persist for documents that are required to undergo manual verification. Possible causes for this status persisting are:

- Document type marked for manual validation and operator has not yet reviewed the document.
- Automatic recording hasn’t been started, is running at full capacity or has paused.

<STATUS _Code=A' _Name='Verified’ _Description= ‘Batch has been verified by a clerk and is pending recording ’ </STATUS>

Full Description: A human operator has reviewed the batch and found all documents to be acceptable for recording. This Status should exist for an extremely short period of time. Possible causes for this persisting are:

- The eRecording process has not been started, is running at full capacity or has paused.
- The Recording Database is running at full capacity

< STATUS _Code='D’ _Name='Processing Suspended’ _Description= ‘DO NOT resubmit this batch. A system related problem has been identified and processing has been suspended until the cause has been found and a solution enacted.’ </STATUS>

Full Description: This batch has passed through the received status and possibly others. However, a system condition surfaced that caused processing to be halted pending manual intervention. Once an operator determines the cause and affects a solution, this batch will be re-evaluated. Duration of this status is undetermined. Possible causes include:

- Database Errors
- Insufficient space remaining for processing
- Network connectivity issues.

Final Status: These messages are returned for batches that have been rejected. Batches with this response must be corrected and processing must be again requested. Actual responses and their meaning with the replacement string in BOLD are:

<STATUS _Code='R' _Name='eRejected’ _Description= ‘Batch is not incompliance with established rules.”>
<RECORDING_ERROR _Type=”Other” _TypeOtherDescription=’error reason’ />
</STATUS>

Full Description: There are numerous reasons for a batch to be considered out of compliance. The error reasons are intended to supply the underlying cause of the problem. It should be noted that automated validation stops at the first detected error. It is therefore possible to have additional errors embedded that would be detected on a second pass. Typically, errors from this stage of processing are only encountered during initial testing of an agent and the document types. Typical error reasons include:

- **No PACKET ID or the XML format not entered**
- **Incorrectly formatted XML DTD**
- **Agent Account Not found**
- **Agent Account doesn’t match Agent Name**
- **Agent Not permitted to eRecord**
- **Agent Not permitted to send this document type**
- **Insufficient parties submitted for document type**
- **(In Florida) Deed submitted without a DR219**
- **If an Escrow request, Insufficient Funds for fees calculated.**
- **If ACH requested, daily fees exceed permitted level.**

```
<STATUS _Code='X' _Name='Rejected’ _Description= ‘Batch has been rejected by a
clerk for the reason shown”>
<RECORDING_ERROR _Type=”Other” _TypeOtherDescription=’error reason’
/>
</STATUS>
```

Full Description: Documents that were not designated for automated recording go to a manual queue where an experienced clerk reviews the summation. Documents that are in this status have been found by the clerk to be unacceptable for recording. The error reason is manually entered by the clerk and is expected to provide adequate information for correction. Some typical reasons are:

- Model 2 submission and the document image doesn’t match the XML data.
- (In Florida) document image of the DR219 doesn’t match the deed.

C. Recorder

The Recording office defines the rules and requirements for all submitters. They may choose to permit only specific documents types prepared by specific agents to be processed automatically. In this case, submitted and conforming batches will normally be processed within minutes of submission. The alternative is a default to a manual review and verification of the document.

1. Manual Validation

If manual validation is chosen, conforming documents will not be recorded until an operator reviews the document and indicates that the document meets the essential requirements. If the operator decides that the batch is non-conforming, the operator rejects the batch. The submitter will get a Rejected status when the Web service is polled for the batch.

2. Overall Control and Monitoring

The recorder will also be using the NewVision eRecording Management application which provides an overall view of all batches in the process. This application permits them to see all submitted work, the current processing state similar to the Status Codes presented above, the essential details of the batch, the formatted XML, the raw XML and an error log associated with each batch.

With the use of the Response Web Service and the eRecording Management Application, the recorder may expect that all reasonable questions should be answerable without need for human interaction. One possible exception is when a batch is reported to be in a *Processing Suspended* status. This status was created and is used in the event that some system event is preventing the completion of the evaluation and recording for this batch. In this event, the recorder is required to take three steps:

1) Determine the cause of the problem from the error messages
2) Affect an appropriate remedy
3) Attempt reprocessing of the batch.

Alternatively, the recorder could determine that the problem is with the source batch and choose to have it rejected. In this case, the recorder is expected to enter additional message text that explains the reason for his action.

In all cases, a submitter with a batch in a *Processing Suspended* status should NOT resubmit the batch. Doing so, may result in duplicate recordings and duplicate charges.

4.0 Recorders eRecording Requirements

While the PRIA DTDs define the standard data definitions, until PRIA moves to schema, additional documentation is necessary to define characteristics and requirements. There are currently two areas where this has been done: DR219 requirements for deeds that are specific to Florida, and the validation rules through which the incoming XML will be tested:

DR219: This form must be submitted with each deed that is to be recorded. It must be included in the deed package with a DOC_TYPE tag of DR219. If it is not present, the recording will be rejected. The DR219 form is not returned in the response message.

Validation rules: A spreadsheet which defines the standards to which the XML data types are subjected is attached as Appendix B

5.0_Appendices

Appendix A: eRecord Request

The post command is:

```
POST /erecord_request_2_4/erecord_request_service.asmx HTTP/1.1
Host: server-ip-address
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://newvisionsystems.com/erecording_2_1/erecord_request"
```

The **length** value above is the total length of the following SOAP message, which is sent as the following request data:

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <erecord_request xmlns="http://newvisionsystems.com/erecording_2_1">
      <input_header>string</input_header>
      <input_xml>xml</input_xml>
    </erecord_request>
  </soap:Body>
</soap:Envelope>
```

In the above message, substitute the following for **string:**

```
<XML_FORMAT>
PRIA_2.4
</XML_FORMAT>
<PACKET_FROM>
```

(Value supplied by the recorder that represents the Account Name from which funds will be debited. This will default to the XML_FORMAT Name if not supplied)

```
</PACKET_FROM>
<PACKET_ID>
submitter-defined value, e.g. 12222
</PACKET_ID>
```

Substitute the PRIA-formatted XML document for the **xml** value above.

This request will add the document sent to our email_queue table and send a response with a return code 0 if the data was loaded successfully or an error code and error message if there was an error. The HTTP response message appears as follows:

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <erecord_requestResponse
xmlns="http://newvisionsystems.com/erecording_2_1">
      <erecord_requestResult>string</erecord_requestResult>
    </erecord_requestResponse>
  </soap:Body>
</soap:Envelope>
```

The value of **string** in this response is the following XML:
<RESPONSE>
   <ERROR_CODE>**n**</ERROR_CODE>
   <MESSAGE>**text**</MESSAGE>
</RESPONSE>

The value of **n** is numeric, **0** indicates success. The value of **text** is a message such as **DATA UPLOADED SUCCESSFULLY.**

Appendix B: Requesting Status of a Submission Request

The following HTTP POST command will return the current status of the package submitted:

```
POST /erecord_response_2_4/erecord_response_service.asmx HTTP/1.1
Host: server-ip-address
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://newvisionsystems.com/erecording_2_1/erecord_response"
```

The **length** value above is the total length of the following SOAP message, which is sent as the following request data:

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
```

```
    <erecord_response xmlns="http://newvisionsystems.com/erecording_2_1">
      <input_xml>string</input_xml>
    </erecord_response>
  </soap:Body>
</soap:Envelope>
```

In the above message, substitute the following for **string:**

```
<XML_FORMAT>
PRIA_2.4
</XML_FORMAT>
<PACKET_FROM>
submitter-defined value from the request’s <XML_FORMAT> tag, e.g. ACS_1.0
</PACKET_FROM>
<PACKET_ID>
submitter-defined value from the request’s <PACKED_ID> tag, e.g. 12222
</PACKET_ID>
```

This command will access the e-recording Oracle database here and return an appropriate message to the sender, in the following response:

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <erecord_responseResponse
xmlns="http://newvisionsystems.com/erecording_2_1">
      <erecord_responseResult>string</erecord_responseResult>
    </erecord_responseResponse>
  </soap:Body>
</soap:Envelope>
```

Appendix C: Additional New Vision XML Tags

The following extra XML TAGS are for submitted Mortgage and Deed Documents:

- NV County - Hard coded to county name
  <NV_COUNTY County=”Erie” />

- NV Common Towns / Locality – Integer translated to a lookup table
  <NV_COMMON_TOWNS Town_Code=”123456” />

- NV Deed RP Type – Integer translated to a lookup table
  <NV_DEED_RP_TYPE Deed_RP_Code=”1” />

code	rp_type
1	1-3 FAMILY (A,B)
2	AGRICULTURAL (E)
3	CONDO (G+8)
4	PROP CLASS 100-299
5	SR ELIGIBLE

- NV Deed Exempt – Integer translated to a lookup table
  <NV_DEED_EXEMPT Deed_Exempt_Code=”2” />

code	exempt_type
1	TP584 III a
2	TP584 III b-l
3	SR CITIZEN
4	RE-RECORD
5	GRANTOR EXEMPT TP

- NV Lender Type – “BANK” “PRIVATE”
  <NV_LENDER_TYPE TYPE=”Bank” />

- NV Dwelling Type – List of hardcoded values
  “1-2 Family” , “3-6 Family”,”PTComm/PTRes”, “Comm/Vacant”
  <NV_DWELLING_TYPE TYPE=”1-2 Family” / >

- NV Exempt Mtg – List of hardcoded values
  <NV_EXEMPT_MTG TYPE=“252 Affidavit” />

Exempt_MTG	
“252 Affidavit”	-exempts all mortgage taxes
“253 Affidavit”	-exempts NFTA Special Additional tax
“255 Affidavit”	-exempts all mortgage taxes
“255Aff New Money”	-Pays all mortgage taxes
“Gov Exempt”	-exempts all mortgage taxes
“Re-Record”	-exempts all mortgage taxes
“Tax Paid”	

Example Town Codes

town_code	town_name
140200	BUFFALO
140900	LACKAWANNA
141600	TONAWANDA CITY
142001	ALDEN VLG
142089	ALDEN TOWN
142201	AMH-WMSVL VLG
142289	AMHERST
142401	AUR-E AURORA VL
142489	AURORA
142600	BOSTON
142801	BRT-FARNHAM VLG
142889	BRANT 456
143001	CKG-SLOAN VLG
143003	CKG-DEPEW VLG
143005	CKG-WMSVL VLG
143089	CHEEKTOWAGA
143200	CLARENCE
143400	COLDEN
143601	COL-GOWANDA VLG
143689	COLLINS
143801	CON-SPRINGVL VL
143889	CONCORD
144000	EDEN
144200	ELMA
144401	EVA-ANGOLA VLG
144489	EVANS
144600	GRAND ISLAND
144801	HMB-BLASDELL VL
144803	HAMBURG VLG
144889	HAMBURG TOWN
145000	HOLLAND
145201	LAN-DEPEW VLG
145203	LANCASTER VLG
145289	LANCASTER TOWN
145400	MARILLA
145601	NST-AKRON VLG
338888	MISCELLANEOUS
339999	HOLD
999999	NULL

DISCLAIMER: THIS WORK IS PROVIDED "AS IS." PRIA, THE COPYRIGHT HOLDER, THE AUTHORS OF THIS WORK AND ANY STANDARD-SETTING BODY CONTRIBUTORS TO THIS WORK MAKE NO REPRESENTATIONS OR WARRANTIES (i) EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE OR NON-INFRINGEMENT; (ii) THAT THE CONTENTS OF SUCH WORK ARE FREE FROM ERROR OR SUITABLE FOR ANY PURPOSE; NOR THAT IMPLEMENTATION OF SUCH CONTENTS WILL NOT INFRINGE ANY THIRD-PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS. IN NO EVENT WILL PRIA, THE COPYRIGHT HOLDER, ANY AUTHOR OF THIS WORK, OR THE STANDARD-SETTING BODY CONTRIBUTORS TO THIS WORK BE LIABLE TO ANY PARTY FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES FOR ANY USE OF THIS WORK, INCLUDING, WITHOUT LIMITATION, ANY LOST PROFITS, BUSINESS INTERRUPTION, LOSS OF PROGRAMS OR OTHER DATA ON YOUR INFORMATION HANDLING SYSTEM OR OTHERWISE, EVEN IF PRIA, THE COPYRIGHT HOLDER AND/OR ANY AUTHORS AND/OR ANY STANDARD-SETTING BODY CONTRIBUTORS TO THIS WORK ARE EXPRESSLY ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Portions of this document are reproduced from the Mortgage Industry Maintenance Organization (MISMO) "MISMO "All Process Area" Logical Data Dictionary (LDD)" and subject to the disclaimer and license relating to that document.

Starter package Information

Agent Name:
Agent Number: ________-_____
Test Request Server Address: ___________________________
Test Response Server Address :___________________________

Production Request Server Address: _____________________________________
Production Response Server Address: ____________________________________

Replacement String Values:
PRIA 2.4 Packet value:

```
<REQUEST InternalAccountIdentifier=above agent number
```

Header String values:

```
<XML_FORMAT>
    PRIA_2.4
</XML_FORMAT>
<PACKET_FROM>
    (above agent name or default to the XML_Format name.)
</PACKET_FROM>
```