Data Broker
N-DEx, FBI NIBRS, and Texas NIBRS Reporting

Data Broker
Interface Control Document
with Web Services

Version 1.0

10/23/2020

Prepared by:

Crime Records Service (CRS)
Law Enforcement Support Division (LES)

Contents

1. Scope

This document provides instructions for submission of National Data Exchange (N-DEx), National Incident-Based Reporting System (NIBRS), and Texas NIBRS information to the Data Broker provided by the Texas Department of Public Safety. The document details how to connect to the Data Broker and the steps for approval to submit N-DEx, NIBRS, and Texas NIBRS data.

The Data Broker provides for the full interactions between law enforcement submitters, Texas's Texas Data Exchange (TDEx) program, the Federal Bureau of Investigation's (FBI's) N-DEx Program, and Texas's Uniform Crime Reporting (UCR) program.

The process for submission of data to the Data Broker for inclusion in the FBI's N-DEx and/or UCR programs and the Texas NIBRS specifications includes:

1. Law enforcement Entities originate N-DEx records;
2. The Data Broker accepts N-DEx records and NIBRS records;
3. The FBI-CJIS Division's N-DEx Program accepts N-DEx records, and
4. The Texas UCR Repository accepts federal and Texas NIBRS records.

The Data Broker processes are uni-directional, flowing from law enforcement Entities to the Data Broker to the FBI's N-DEx and to the Texas UCR repository.

2 Data Broker

The Data Broker provides web services to submit N-DEx and NIBRS data.

Data Broker Web Services	Description
SubmitNDExDocument	This process accepts N-DEx data via a NIEM N-DEx 4.0 IEPD formatted document. It returns a successfully submitted status.
SubmitNIBRSDocument	This process accepts NIBRS data via a NIEM NIBRS 2019.2 IEPD formatted document. It returns a successfully submitted status.
SubmitCombinedDocument	This process accepts a file containing both N-DEx and NIBRS data via an N-DEx 4.0 IEPD formatted document. It returns a successfully submitted status.

N-DEx and NIBRS data is submitted to the Data Broker Services via compliant National Information Exchange Model (NIEM) Information Exchange Package Documentation (IEPD) documents.

2.1 Interface Overview

Texas Entities interface with the Data Broker web services via the public internet using the Data Broker's URL – databroker.dps.texas.gov. The web service detail's full address is included in the document file distributed to vetted providers by DPS.

2.1.1 Availability

The Data Broker is available 24/7/365 days a year, excluding maintenance periods.

Contributing Entity contacts will be notified by DPS staff via email at least 72 hours in advance of scheduled maintenance. In the event that there needs to be unscheduled maintenance due to unforeseen circumstances, Contributing Entity contacts will be notified.

Contributing Entity contacts will be notified via the email addresses that are currently on file.

All transactions with the Data Broker are handled via HTTP REST protocol over the public internet. The messages are encrypted between the client and service.

2.1.2 Format

Data Broker report submissions must be structured to the NIEM IEPD markup standard. The NIEM IEPD documentation is available from the FBI N-DEx program or the FBI UCR program.

The state of Texas requires specific data elements to be included with certain types of NIBRS transactions. Those data elements conform to the IEPD standards. The rules governing those data elements and how they are incorporated into the NIBRS submission are described in the Texas Specific NIBRS Requirements document.

2.1.3 Data Broker Accounts

2.1.3.1 Establishing a Data Broker Account

The following describes the steps to establish a Data Broker account:

1. The Contributing Entity requests access to the Data Broker via email sent to databroker@dps.texas.gov.
2. The DPS representative creates a process account for the Contributing Entity.
3. The Contributing Entity's account receives an email with instructions to complete the Data Broker account setup.

2.1.3.2 Uploading Files

This section provides technical details on how to upload a document to the Data Broker, after the Contributing Entity has completed the account setup process.

The process to submit a file uses an Access ID and a Shared Secret. The Access ID is passed with every request. The Shared Secret is used to create an Authorization header that is used to retrieve a limited-use token. The token can then be used for a period of time to submit documents.

1. An authorized Entity user signs into the Entity Data Broker service account and visits the API Key Management page found under the Data Broker menu or the Account menu. On this page, you can see information about the keys that have been generated and generate a new key. It also contains template scripts for Windows and Unix to demonstrate how to submit a file. When a new key is generated, the shared Secret is shown one-time to be copied. It will never be shown again, so if the Secret is lost, a new one must be generated. Keys can also be deactivated from this page. Multiple keys can be active at a time to enable an entity to transition from one key to another without interruption.

2. When the Entity is ready to submit a file, they must first sign a request to generate a limited use token. They will do this by signing the Shared Secret, the Access ID, and the current date down to the minute using SHA 256.

   For example, if the secret is ABCDEFGHIJKLMNOP and the Acess ID is 1234567890 and the time is 2021-04-08 21:10, the "raw" key would be:

   ABCDEFGHIJKLMNOP1234567890202104082110

   The SHA 256 algorithm would be run on that raw key, resulting in a signed key of:

53a77048d15b62a43ac12e0b18ff84a1b9b4fe2334e0d2fbd1bd46e83b497082

It is critical that there be no new lines or carriage returns in the raw key and that the date be given in UTC.

An HTTP POST request for a token must be sent to the token endpoint, passing the Access ID in a query parameter called “accessId” and a header called Authorization with a value of the signed key. The response will be a token that is used to submit a document.

To submit a document, an HTTP POST request is sent to the submission endpoint, again passing the Access IDin a query parameter called “accessId” and an Authorization header, this time with a value of “bearer “ followed by the token retrieved above. The submit request should also have a Content-Type header set to text/xml and the body of the POST should be the XML document being submitted.

The templates below illustrate the above process for Unix and Windows, but most programming languages should be able to implement the algorithm.

There are two endpoints for each environment (UAT and Production), differing only in the hostname:

Production:

https://databroker.dps.texas.gov/api/token - retrieve the limited use token described below

https://databroker.dps.texas.gov/api/databroker/submit - submit a document

UAT:

https://databrokeruat.dps.texas.gov/api/token - retrieve the limited use token described below

https://databrokeruat.dps.texas.gov/api/databroker/submit - submit a document

Example:

We have included examples for the bash shell on Linux/Unix/OS X and Powershell on Windows. Other programming languages can be used as long as they generate the correct data. **In particular, the key that is hashed below must not include any carriage returns or line feeds and the date must be in UTC.**

```
Linux/Unix:
  # Set up the variables that will be used for every submission.
  # These will be unique to your agency.
  secret="<the secret you obtain in step 1>"
  accessId="<the access ID you obtain in step 2>"
  ori="<your ORI such as TXDPS0000>"

  date=`date +'%Y%m%d'`
  time=`date +'%H%M'`

  # The document should be named according to the N-DEx/NIBRS specifications.
  sequence="0001"    # a sequence number to make each document unique
  # The document would have already been stored in this file name:
  documentId="${ori}_${date}_${time}_${sequence}.xml"

  host="databrokeruat.dps.texas.gov"   # leave off the uat in production
  tokenUrl="https://${host/api/token?accessId=${accessId}"

submitUrl=http://${host}/api/databroker/submit?accessId=${accessId}&documentId=${documentId}

  rawKey="${secret}${accessId}`date +'%Y%m%d%H%M'`".  # No carriage returns or line feeds
  hashedKey=$(echo -n $rawKey|shasum -a 512|cut -d" " -f1)

Windows Powershell
  $secret="<the secret you obtain in step 1>"
  $accessId="<the access ID you obtain in step 2>"
  $ori="<your ORI such as TXDPS0000>"
  $date = (get-date -UFormat '%Y%m%d')
  $time = (get-date -UFormat '%H%M')
  # The document should be named according to the N-DEx/NIBRS specifications.
  $sequence = "0001" # a sequence number to make each document unique
  # The document should be named according to the N-DEx/NIBRS specifications.
  $documentId = "${ori}_${date}_${time}_${sequence}.xml"

  $host = "databrokeruat.dps.texas.gov" # leave off the uat in production
  $tokenUrl = "http://${host}/api/token?accessId=$accessId"
  $submitUrl =
http://${host}/api/databroker/submit?accessId=$accessId&documentId=$documentId

  $rawKey = "$secret$accessId$date$time". # No carriage returns or line feeds
  $stream = [System.IO.MemoryStream]::new()
  $writer = [System.IO.StreamWriter]::new($stream)
  $writer.write($rawKey)
  $writer.Flush()
  $stream.Position = 0
  $hashedKey = (Get-FileHash -InputStream $stream -Algorithm SHA512).hash
```

3. The hash the agency generates will be used along with the Access Key ID to generate an upload API key. The upload API key will be valid for 20 minutes from the time it is requested. The agency generates the API key using an endpoint that will be provided in step 1.

Example:

```
Linux/Unix:
 apiKey=`curl -X POST -H "Authorization: ${hashedKey}" "${tokenUrl}"`

Windows Powershell
  $headers = @{"Authorization" = "$hashedKey"; }
  $apiKey = Invoke-WebRequest -Uri $tokenUrl -Method 'POST' -Headers $headers
```

4. The API key is used to upload a document. The same API key can be used to submit multiple documents, provided they are submitted within the 20-minute expiration

window. After the key expires, steps 2 and 3 must be repeated to obtain a new API key. The exact endpoint to upload a document can be confirmed in step 1.

Example:

```
Linux/Unix:
  curl -X POST -H "Content-Type: text/xml" -H "Authorization: bearer ${apiKey}" -d @${documentId}
"${submitUrl}"

Windows Powershell
  $headers = @{"Content-Type" = "text/xml"; "Authorization" = "bearer $apiKey"; }
  Invoke-WebRequest -Uri "$submitUrl" -Method 'POST' -Headers $headers -Infile $pathToDocument
```

An HTTP response code of 200 from the upload end point can be used to determine if the document was uploaded successfully. If the code is anything else, the file was not submitted successfully. Only the HTTP code is necessary to determine if the upload was successful. However, the response text will be an XML string that indicates the reason for the failure, if possible.

The response text for a successful upload will be:

```
<response>
  <id>…the document ID from the submission…</id>
  <reference>… an internal Data Broker number</reference>
  <time>…the timestamp (ISO8601)</time>
</response>
```

For a failure, it will be:

```
<response>
  <error>
    <code>400</code>
    <text>reason</text>
  </error>
  <time>…the timestamp (ISO8601)</time>
</response>
```

Inbound data validation occurs when a NIEM document is submitted to the Data Broker process. Validation is not done at the time of submission. The response of the submission only indicates if the Data Broker successfully received the document.

2.1.4 Validation

The Data Broker will validate inbound NIEM IEPD submissions specific to the published IEPD standard available from the FBI N-DEx or the FBI UCR program.

In the event of a Data Broker system failure, the Contributing Entity may receive an error with a message stating to contact DPS.

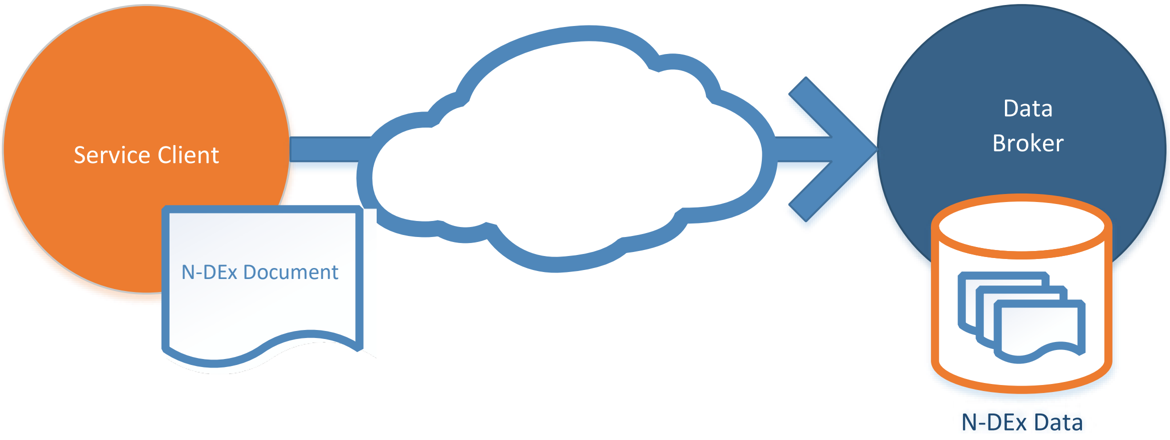
2.2 Workflow

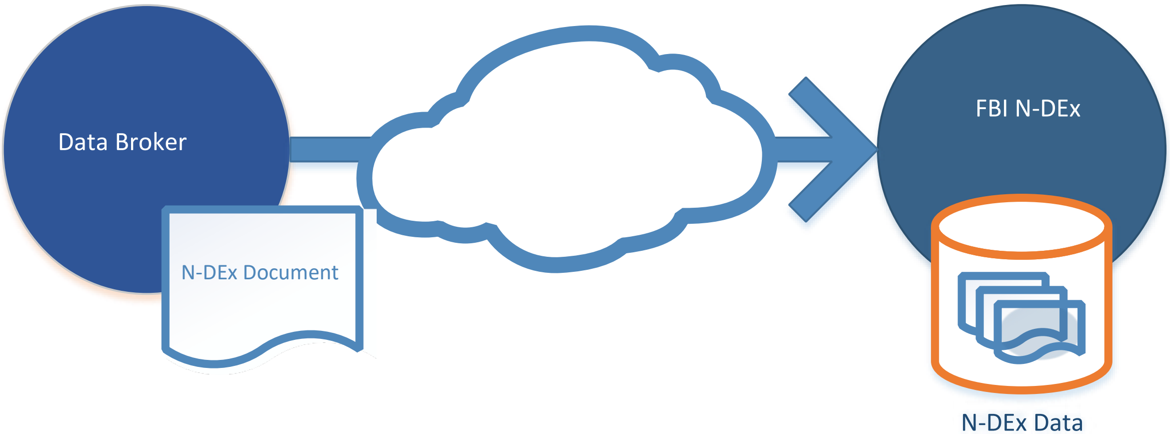
Contributing Entities will use one of the three processes listed below to submit NIEM IEPD documents to the Data Broker via an HTTP Post web service request. The HTTP response will indicate whether or not the document was successfully ingested. The following status codes will be returned:

- 200 (OK) – the document was successfully ingested. Note: this does not mean the document was determined to be valid, only that it was received.
- 400 (Bad Request) – this indicates that the web service did not understand the request, such as if a parameter was missing.
- 403 (Forbidden) – this indicates that the client was not authorized to submit the document or that the authorization information sent with the request was invalid.
- 500 (Internal Error) – this indicates a problem with the Data Broker. The file may be re-submitted, but if the error continues, DPS should be notified at databroker@dps.texas.gov.

2.2.1 Data Broker Data Submission Workflow for N-DEx documents

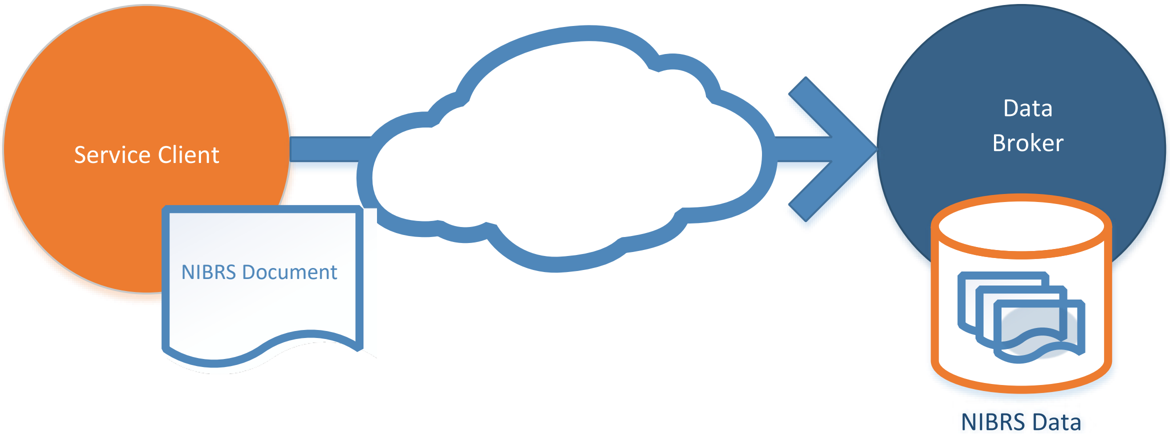
Contributing Entities utilize the SubmitNDExDocument process to submit NIEM N-DEx IEPD documents to the Data Broker. Each document is initially validated using XSD Schema validation against the N-DEx IEPD. Following successful validation, the Data Broker transmits the document to the FBI N-DEx system.

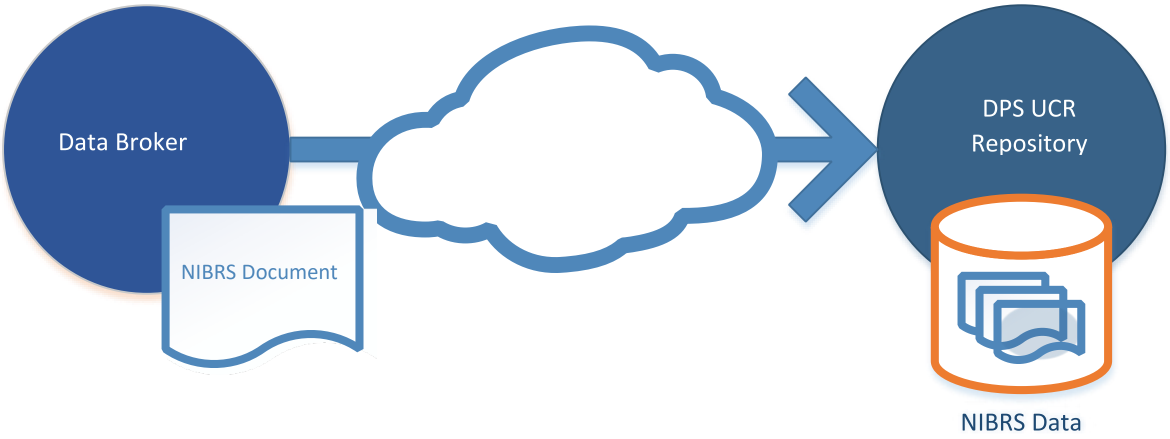

Service Client
N-DEx Document
Data
Broker
N-DEx Data


Data Broker
N-DEx Document
FBI N-DEx
N-DEx Data

2.2.2 Data Broker Data Submission Workflow for NIBRS documents

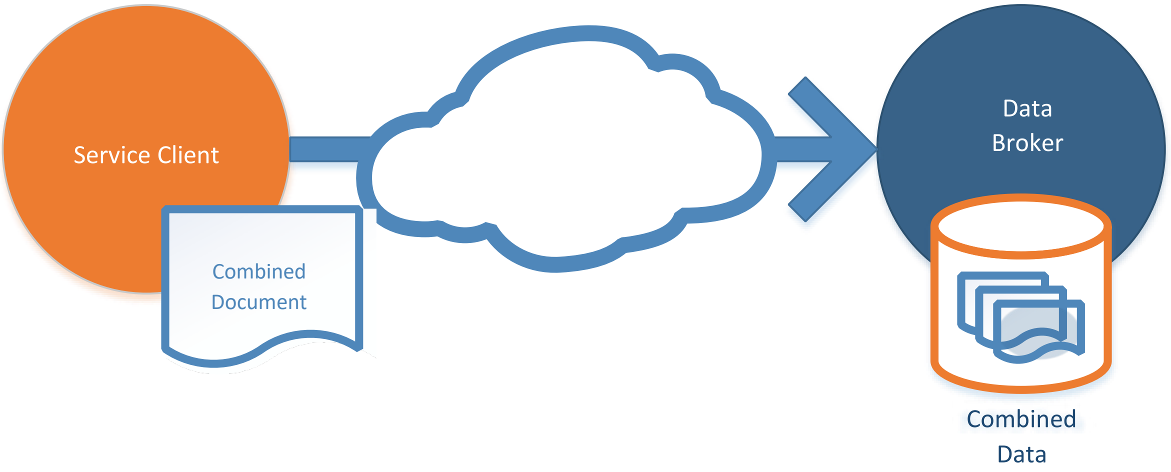
Service clients utilize the SubmitNIBRSDocument process to submit NIEM NIBRS IEPD documents to the Data Broker. Each document is initially validated using XSD Schema validation. Once a document successfully passes schema validation, a document is then validated using the FBI XCOTA business rules. Following successful submission, the Data Broker transmits the document to the DPS UCR Repository.

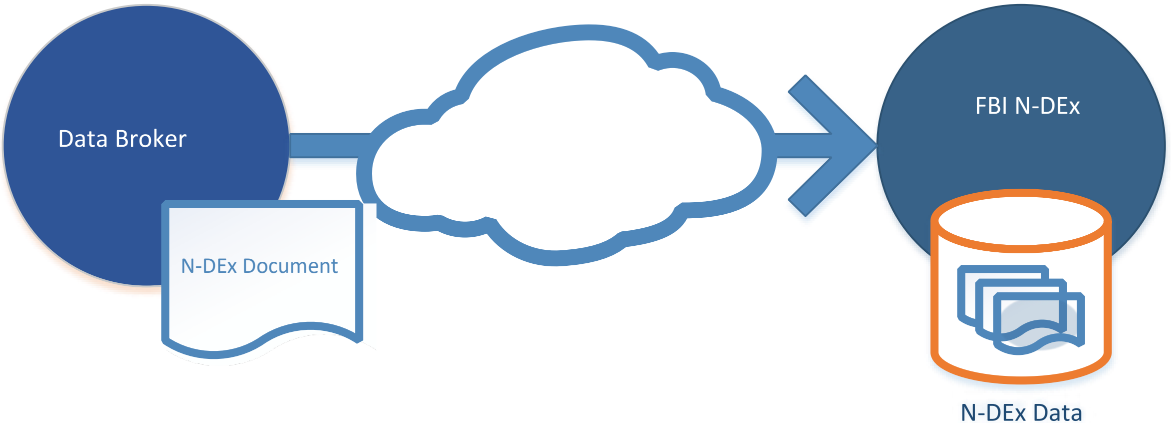


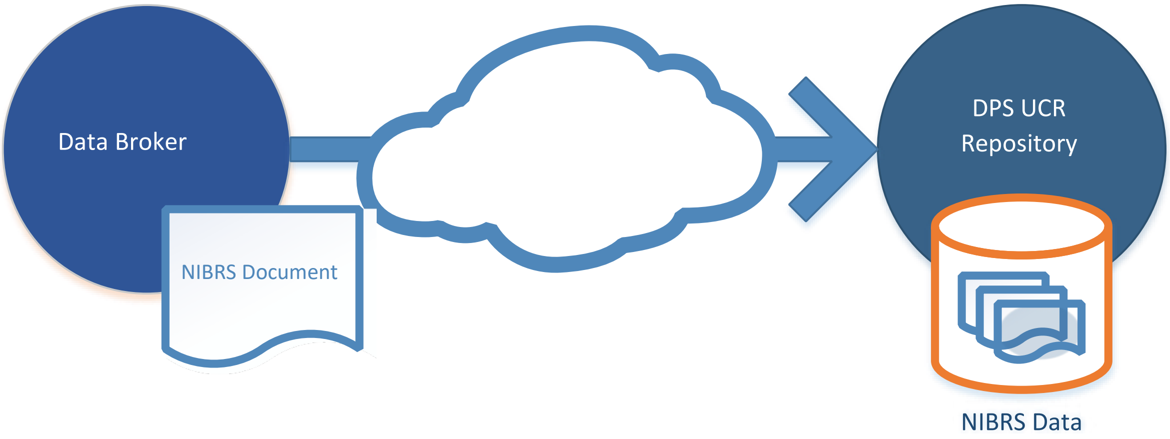


2.2.3 Data Broker Data Submission Workflow for N-DEx and NIBRS documents

Service clients utilize the SubmitCombinedDocument process to submit NIEM N-DEx IEPD documents to the Data Broker. Each document is initially validated using XSD Schema validation against the N-DEx IEPD. Following successful submission, the Data Broker transmits the document to the FBI N-DEx system and NIBRS information is transmitted to the DPS UCR system for further processing.






Data Broker
NIBRS Document
DPS UCR Repository
NIBRS Data

2.3 Testing and Certification

Submissions from each Contributing Entity will be certified using a test environment prior to being forwarded for testing to N-DEx or DPS UCR.

2.3.1 N-DEx Testing and Certification

The certification process for XML documents containing N-DEx data and formatted using the N-DEx IEPD is as follows:

1. The Contributing Entity creates sample documents for each of the defined N-DEx record types. Once sample XML documents are created, it is recommended that they be validated against the IEPD and against the CJIS XML Conformance Testing Assistant (XCOTA) tool. These validations are used by the Data Broker and by N-DEx.
2. The Contributing Entity submits the documents to the Data Broker test environment using the upload process described above.
3. The Data Broker validates the documents against the N-DEx IEPD xml schemas.
4. The Data Broker validates the documents using the CJIS XCOTA tool.
5. If there are errors, the DPS representative will contact the Entity with details.
6. When each type of document is determined to be valid, DPS will release the documents to N-DEx for further validation.
7. When N-DEx confirms that the sample submissions are valid, the Contributing Entity will be marked as certified and provided access to the production Data Broker environment.
8. The Contributing Entity begins submitting production documents.

2.3.2 NIBRS Testing and Certification

The certification process for XML documents containing NIBRS data and formatted using either the NIBRS IEPD or the N-DEx IEPD is as follows:

1. The Contributing Entity creates sample documents for each of the defined NIBRS record types. Once sample XML documents are created, it is recommended that they be validated against the IEPD and against the CJIS XML Conformance Testing Assistant (XCOTA) tool. These validations are used by the Data Broker.
2. The Contributing Entity submits the documents to the Data Broker test environment using the upload process described above.
3. The Data Broker validates the documents against the NIBRS or N-DEx IEPD xml schemas.
4. The Data Broker forwards the documents to the Texas UCR system.
5. If there are errors, the DPS representative will contact the Entity with details.

6. When the Texas UCR staff confirms that the sample submissions are valid, the Contributing Entity will be marked as certified and provided access to the production Data Broker environment.
7. The Contributing Entity begins submitting production documents.

3 Examples

This section provides examples for utilizing the Data Broker Services. Where applicable, data standards should be deferred to their respective current documentation.

3.1. Data Broker Reporting Data Submission Workflow for N-DEx documents

3.1.1. SubmitNDExDocument Data Assembly Characteristics

```
<?xml version="1.0" encoding="UTF-8"?>

<ndex:Submission xsi:schemaLocation="http://fbi.gov/cjis/ndex/4.0 ../base-xsd/ndex/4.0/ndex.xsd...>

 <cjis:MessageMetadata>

  …

  </cjis:MessageMetadata>

  <ndex:IncidentReport>…</ndex:IncidentReport>

  <ndex:Arrest>…</ndex:Arrest>

  … other ndex elements

</ndex:Submission>
```

3.2 Data Broker Reporting Data Submission Workflow for NIBRS documents

3.2.1 SubmitNIBRSDocument Data Assembly Characteristics

```
<?xml version="1.0" encoding="UTF-8"?>
<nibrs:Submission xmlns:nibrs="http://fbi.gov/cjis/nibrs/2019.2" …>
  <cjis:MessageMetadata>
  …
  </cjisMessageMetadata>
  <nibrs:Report>
    <nibrs:ReportHeader>
      <!-- Submission Type -->
      <nibrs:NIBRSReportCategoryCode>GROUP A INCIDENT
REPORT</nibrs:NIBRSReportCategoryCode>
      <!-- Submission Action Type -->
      <nibrs:ReportActionCategoryCode>I</nibrs:ReportActionCategoryCode>
      …
    <nibrs:ReportHeader>
    <nc:Incident>…see NIBRS 2019.2 IPED for details</nc:Incident>
</nibrs:Submission>
```

3.3 Data Broker Reporting Data Submission Workflow for Combined N-DEx and NIBRS Documents

3.3.1 SubmitCombinedDocument Data Assembly Characteristics

```
<?xml version="1.0" encoding="UTF-8"?>
<ndex:Submission xsi:schemaLocation="http://fbi.gov/cjis/ndex/4.0 ../base-
xsd/ndex/4.0/ndex.xsd...>
  <cjis:MessageMetadata>
  …
  </cjis:MessageMetadata>
  <ndex:IncidentReport>
   …
   <ndex:NIBRSReportCategoryCode>GROUP B ARREST
REPORT</ndex:NIBRSReportCategoryCode>
   <ndex:NIBRSReportActionCategoryCode>
     A
    </ndex:NIBRSReportActionCategoryCode>
  </ndex:IncidentReport>
  <ndex:Arrest>…</ndex:Arrest>
  … other ndex elements
</ndex:Submission>
```