

# **OpenFox™**

## **Interface Specification for OpenFox™ Markup Language (OFML™)**

### **Version 2.0**

June 18, 2010

Updated November 16, 2011



**Computer Projects of Illinois, Inc.**  
**475 - A Quadrangle Dr.**  
**Bolingbrook, IL 60440**  
**(630) 754-8820**

---

## Table of Contents

<b>1</b>	<b>OVERVIEW .....</b>	<b>1</b>
<b>2</b>	<b>GENERAL REQUIREMENTS .....</b>	<b>2</b>
<b>3</b>	<b>MESSAGE FORMATS .....</b>	<b>3</b>
3.1	Input Transactions.....	3
3.1.1	LOGON Transactions.....	4
3.1.2	Administrative Message .....	5
3.1.3	Tagged MFC Message.....	6
3.1.4	Free Form Message.....	7
3.1.5	OpenFox™ Command Message.....	8
3.2	Output Messages.....	8
3.2.1	Acknowledgement Responses .....	9
3.2.2	Error Responses .....	10
3.2.3	Hot Files Responses.....	11
3.2.4	NLETS GJXDM Responses .....	12
<b>4</b>	<b>ADDITIONAL EXAMPLES.....</b>	<b>13</b>
4.1	Client-Initiated Transaction and Responses.....	13
4.2	Client-Initiated Transaction to Another Client .....	18
4.3	Switch-Initiated Transaction .....	19

## 1 Overview

The Omnixx Force™ / OpenFox™ Markup Language™ (OFML™ ) is an extension of the eXtensible Markup Language (XML) for law enforcement interfaces. XML was developed by the World Wide Web Consortium (W3C) and is quickly becoming the standard for the exchange of information between applications. OFML™ 2.0 (based on OFML 1.1 jointly developed by CPI and Datamaxx Technologies, Inc.), defines an XML-based method for exchanging law enforcement information. It does not specify the format of the law enforcement information itself, only the structure for its exchange. The format of the law enforcement information itself will be a function of the agency's requirements and vary greatly among agencies.

Features of OFML™ include:

- 1) OFML™ structured messages are well-formed XML documents that focus on data and leave formatting to the applications. As XML documents, messages will be able to be formatted for display using standard XML techniques such as XSL style sheets and document processors.
- 2) OFML™ accommodates both unstructured XML documents with legacy text contents as well as fully tagged messages. Utilizing standard XML techniques such as XSL allows compatibility with GJXDM and NIEM formatted transactions as well as legacy text messages.
- 3) OFML™ creates a message that's easily interpreted by someone who is familiar with NCIC/NLETS messages without them having to obtain more than a cursory understanding of XML.

This document defines the standard OFML™ interface for use between the OpenFox™ message switch and any compliant OFML v2.0 system or client.

## 2 General Requirements

OFML™ messages consist of "well-formed" XML documents structured as defined in subsequent sections of this document. This is a very basic implementation of XML designed to meet the requirements of message switch interfaces. Each message contains a root (OFML) node, header element and either a transaction (TRN) or response (RSP) element. Specific attributes depend on the type of message and are defined in the remainder of this document.

```

<OFML>
  <HDR>
    <ID??> (Contents) </ID>
    <APP??> (Contents) </APP>
    <DAC??> (Contents) </DAC>
    <DAT??> (Contents) </DAT>
    <REF??> (Contents) </REF>
    <MKE TST?='Y'> (Contents) </MKE>
    <ENT??> (Contents) </ENT>
    <RQR??> (Contents) </RQR>
    <USR??> (Contents) </USR>
    <PWD??> (Contents) </PWD>
    <ORI??> (Contents) </ORI>
    <DST*> (Contents) </DST>
    <CTL??> (Contents) </CTL>
    <IDX??> (Contents) </IDX>
    <SUM??> (Contents) </SUM>
    <PASS> (Contents) </PASS>
  </HDR>
  <TRN|RSP>
    <MFC*> (Contents) </MFC>
  </TRN|RSP>
</OFML>

```

The following requirements apply to all OFML™ messages:

- 1) White space and line breaks between elements are meaningless and must not be used to convey relevant information. White-space is used in the message definitions and examples in this document for the sake of clarity only.
- 2) Comments may be included anywhere between elements and should be ignored during processing.
- 3) Optional attributes should only be included if necessary.
- 4) The <APP> element has specific values for given implementations and should always be left blank unless an express exception has been noted for the interface in question.
- 5) All messages must fit within the maximum message size for the interface and must not be arbitrarily truncated. Doing so would likely render the messages non-well-formed XML.

### 3 Message Formats

#### 3.1 Input Transactions

Client-initiated transactions are structured as follows:

Element/Attribute	Required	Length	Specification
OFML	<b>Required</b>	<b>N/A</b>	OFML™ element
OFML.HDR	<b>Required</b>	<b>N/A</b>	Header element
OFML.HDR.ID	Optional	10	Message identifier used to identify a message Defaults to OFML.HDR.REF if blank
OFML.HDR.APP	Optional	15	Application Identifier Should always be left blank unless required by the particular implementation
OFML.HDR.DAC	Optional	11	Device address code Used when the source of a message can't be determined directly from the communications interface
OFML.HDR.DAT	Optional	14	Date/time message generated Format: YYYYMMDDHHMMSS
OFML.HDR.REF	Optional	10	Transaction reference number
OFML.HDR.MKE	<b>Required</b>	7	Message key Per specific transaction Optional TST attribute if set to 'Y' indicates a test transaction
OFML.HDR.ENT	Optional	10	Entering agency Only used when entering agency is included in each transaction.
OFML.HDR.RQR	Optional	30	Requestor Only used when requestor is included in each transaction.
OFML.HDR.USR	Optional	30	User ID Only used when user ID is included in each transaction. Standard logon transactions will include the user ID as an MFC element.
OFML.HDR.PWD	Optional	15	Password Only used when user ID and password is included in each transaction. Standard logon transactions will include the password as an MFC element.
OFML.HDR.ORI	Optional	9	Source ORI
OFML.HDR.DST	Optional (1-5)	11	One or more destination elements ORI, state code, device mnemonic or broadcast code
OFML.HDR.CTL	Optional	10	NLETS control field Does not include asterisk

Element/Attribute	Required	Length	Specification
OFML.HDR.PASS	Optional	10	Pass through identifier Used to indicate direct message pass through. Value and pass through results are implementation specific.
OFML.HDR.SUM	Optional	80	Transaction summary Text to summarize contents of message
OFML.TRN	<b>Required</b>	<b>N/A</b>	Transaction element
OFML.TRN DATA ELEMENTS	<b>Required</b>	<b>N/A</b>	Defined in following sections

All input transactions will be of one of the following types

- 1) Administrative Message
- 2) Tagged MFC Message
- 3) Free form message
- 4) OpenFox™ command message

This type will be determined by the MKE present in the OFML.HDR.MKE element. This will also dictate the data elements and their formatting in the OFML.TRN element. Each type of message is defined below.

### 3.1.1 LOGON Transactions

Clients such as OpenFox™ Desktop/Messenger are configured with the ‘signon’ flag turned on. As such, before any input operations can take place or any received messages can be viewed from the client, a user must logon and be validated by the system. Clients using the “FTREMO” interface and exchanging data using OFML standards marked with the signon flag may also be required to submit a logon based upon specific customer requirements.

```

<OFML>
  <HDR>
    <APP>DESKTOP</APP>           * leave blank for FTREMO *
    <ID></ID>                     * generated by remote system *
    <ORI>KSKBI0000</ORI>         * Required - [station ORI] *
    <MKE TYP="U">LOGON</MKE>     * Required = LOGON *
    <SUM>LOGON: TESTER</SUM>     * Summary line (see explanation) *
    <DAT>20111118080545</DAT>   * format = YYYYMMDDHHMMss
  </HDR>
  <TRN>
    <USR>TESTER</USR>           * Assigned User ID
    <PWD>dsfgeytuyuk</PWD>     * PIN+passcode *
  </TRN>
</OFML>

```

The <PWD> field must contain the PIN and passcode concatenated. PIN is 4 digit and passcode is 6 digit as generated by the RSA token.

A rejected logon will appear similar to the following. If the logon is rejected, no transactions will be allowed from the device.

```

<OFML>
  <HDR>
    <APP>DESKTOP</APP>
    <ID>LNF1GGxCQ=</ID>
    <DAC>TESTSTA</DAC>
    <SRC>SWITCH</SRC>
    <DAT>20111117165408</DAT>
    <MKE>ERR</MKE>
    <SUM>LOGON: TESTER</SUM>
  </HDR>
  <RSP>
    <TXT>LOGON REJECTED - LOGON REQUEST TIMED OUT - PLEASE TRY
AGAIN</TXT>
  </RSP>
</OFML>

```

An accepted logon will appear as follows for Messenger and will be slightly different for FTREMO devices

```

<OFML>
  <HDR>
    <APP>DESKTOP</APP>
    <ID>LNF1GHERM=</ID>
    <DAC>TESTSTA</DAC>
    <SRC>SWITCH</SRC>
    <DAT>20111117165634</DAT>
    <MKE>LOGON</MKE>
    <SUM>LOGON: TESTER</SUM>
  </HDR>
  <RSP>
    <TXT>LOGON ACCEPTED</TXT>
    <USR>
      <UID>TESTER</UID>
      <FNA>TEST</FNA>
      <LNA>USER</LNA>
      <ROLES>
        <ROLE name="SYSGEN" version="4EBD3B42"/>
        <ROLE name="ARCHIVE ACCESS ALL" version="4E95E5CA"/>
      </ROLES>
      <AGY>TPKKB</AGY>
      <STATUS></STATUS>
      <AUTH_METHOD>PASSWORD</AUTH_METHOD>
    </USR>
  </RSP>
</OFML>

```

### 3.1.2 Administrative Message

This input format will be used whenever the MKE present in the OFML header is a Admin type Message Key, for example ‘AM’. Administrative messages allow free format, non-tagged text as their data component. This text is contained within a single <TXT> element directly underneath the OFML.TRN element. It should be the only element present within the OFML.TRN element. The text is free length, however the total message size will still be dictated by the overall message length limitations present on the interface which the message is transmitted over, or the system

set maximum message length, whichever is less. While the text is free format, any reserved XML characters must be escaped or contained within a CDATA section to ensure the document remains a well formed XML document. An example input administrative message is presented below:

```
<OFML>
  <HDR>
    <ID>1234509878</ID>
    <DAT>20010824073123</DAT>
    <REF>12345ABCDE</REF>
    <MKE>AM</MKE>
    <DST>SPT0</DST>
    <SUM>AM TO SPT0</SUM>
  </HDR>
  <TRN>
    <TXT>
      ADMIN MESSAGE TEXT GOES HERE
      HELLO WORLD
    </TXT>
  </TRN>
</OFML>
```

### 3.1.3 Tagged MFC Message

This input format will be used whenever the MKE present in the OFML header is a type that requires MFC data. This would include entry, modify, inquiry, locate, cancel and clear transactions. The OFML representation of MFCs is achieved by including each individual MFC as a child element of the OFML.TRN element, using the MFC name as the XML tag and the MFC data as the contents of the element.

Additional requirements for MFC elements:

- 1) MFC elements will be arranged much like the current NCIC and NLETS standard for formatted messages with the following exceptions:
  - a) The data will be tagged as XML elements. For example, the current NCIC/NLETS standard would represent the LIC MFC and data as: LIC/ABC123, while in the XML format this would be: <LIC>ABC123</LIC>.
  - b) The data will always be tagged with identifiers. Unlike an NCIC entry transaction where the fields are present without the MFC/ code to identify them, in XML formatted messages the element tag will always be the MFC code.
  - c) Blank MFCs, even on entry transactions need not be present. Since all MFCs are tagged there is no need to include any MFC that does not contain data.
- 2) MFC order is irrelevant with the following exceptions:
  - a) NCIC entries: same as NCIC formats
  - b) NCIC modifies: record identifiers must be the first two MFCs followed by the modifiers
  - c) Any element, including non-MFC elements, can include an optional error identifier (EID) attribute, which will be used for identifying errors. Where included, it indicates



the value to be used in the ERR response to identify the field in error when it is not the same as the tag.

- 3) If using OFML over DMPP2020, images may be included as DSEO2020 objects only.
- 4) If using OFML over FoxTalk™, images are indicated as standard HTML objects. Refer to the FoxTalk specification from Computer Projects of Illinois for additional information.

For example, the legacy format for a NLETS Registration Query (RQ) by plate is as follows:

```
RQ.IL1234567.IN.LIC/ABC123.LIY/2010.LIT/PC
```

This would translate into the following OFML Message:

```
<OFML>
  <HDR>
    <ID>1234509878</ID>
    <DAT>20010824073123</DAT>
    <REF>12345ABCDE</REF>
    <MKE>RQ</MKE>
    <ORI>IL1234567</ORI>
    <DST>IN</DST>
    <SUM>RQ: LIC/ABC123</SUM>
  </HDR>
  <TRN>
    <LIC>ABC123</LIC>
    <LIY>2010</LIY>
    <LIT>PC</LIT>
  </TRN>
</OFML>
```

### 3.1.4 Free Form Message

This input format is used to supply free form, command line style legacy input messages within the OFML envelope. This input style is specified by setting the contents of the OFML.HDR.MKE element to 'FREE'. The data message itself will be contained in a <FRE> element directly underneath the OFML.TRN element. The <FRE> element should be the only child of OFML.TRN present, and it must contain a valid legacy input formatted message (ie NLETS or NCIC dot delimited message) supported at the current installation. These formats vary from state to state and as such can't be defined in their entirety here. These legacy input formats typically include positions for fields that are also present in the OFML.HDR element such as destination, ORI, and control field. In these cases the information present in the <FRE> element will override any data present in the OFML.HDR, and as such for clarity the fields in the OFML.HDR should be left blank.

For example, the legacy format for a NLETS Registration Query (RQ) by plate is as follows:

```
RQ.IL1234567.IN.LIC/ABC123.LIY/2010.LIT/PC
```

This would translate into the following OFML "FREE" Message:

```
<OFML>
  <HDR>
    <ID>1234509878</ID>
    <DAT>20010824073123</DAT>
    <REF>12345ABCDE</REF>
    <MKE>FREE</MKE>
    <SUM>FREE FORMAT RQ</SUM>
```

```

</HDR>
<TRN>
  <FRE>RQ.IL1234567.IN.LIC/ABC123.LIY/2010.LIT/PC</FRE>
</TRN>
</OFML>

```

### 3.1.5 OpenFox™ Command Message

This input format is used to supply commands to the OpenFox™ Message Switch. This input style is specified by setting the contents of the OFML.HDR.MKE element to 'CMND'. The data message itself will be contained in a <CMD> element directly underneath the OFML.TRN element. The <CMD> element should be the only child of OFML.TRN present, and it must contain a valid OpenFox™ command. These command messages are defined in the Relavent OpenFox™ Message Switch documentation. For example, to execute a Display Station Activity (DSA) report, the input message format would be as follows:

```

<OFML>
  <HDR>
    <ID>1234509878</ID>
    <DAT>20010824073123</DAT>
    <REF>12345ABCDE</REF>
    <MKE>CMND</MKE>
    <SUM>OPENFOX COMMAND: DSA</SUM>
  </HDR>
  <TRN>
    <CMD>/DSA</CMD>
  </TRN>
</OFML>

```

## 3.2 Output Messages

Output OFML messages delivered to devices are structured as follows:

Element/Attribute	Specification
OFML	Same as client-initiated transactions
OFML.HDR	Same as client-initiated transactions
OFML.HDR.ID	Returns the data supplied in the original input transaction or the value "UNKNOWN" if none supplied or unable to determine original value.
OFML.HDR.DAC	Device address code Identifies the receiving station
OFML.HDR.SRC	Set to mnemonic associated with source of response (e.g., "NCIC", "NLETS", "DMV", "CCH").
OFML.HDR.ENT	Returns the data supplied in the original input transaction – only included if present in original transaction
OFML.HDR.RQR	Returns the data supplied in the original input transaction – only included if present in original transaction
OFML.HDR.DAT	Date/time message sent to output device Format: YYYYMMDDHHMMSS
OFML.HDR.REF	Returns the data supplied in the original input transaction or the value "UNKNOWN" if none supplied or unable to determine original value.

Element/Attribute	Specification
OFML.HDR.APP	Returns the data supplied in the original input transaction or the value "UNKNOWN" if none supplied or unable to determine original value.
OFML.HDR.MKE	Message key or function code TYP attribute identifies the MKE type (ie "A" – Admin, "R" – Response etc...)
OFML.HDR.ORI	Source ORI – If supplied by source or blank if none present
OFML.HDR.DST	Destination – Each entered message destination will be present in an individual <DST> element. To determine actual destination for the receiving device see the <DAC> element.
OFML.HDR.CTL	NLETS control field from transaction For solicited responses this field returns the data supplied in the original input transaction – only included if present in original transaction. For unsolicited messages contains the NLETS control field sent by the originating device if included. Asterisk not included
OFML.HDR.SUM	Returns the data supplied in the original input transaction – only included if present in original transaction
OFML.RSP	Response element – contents defined below

Additional requirements for RSP element:

- 1) For unstructured responses, the text will be included in a single child XML element with the "TXT" tag.
- 2) Response can contain multiple records within the RSP element, each with its own MFC elements. Such elements can have any tag but must contain a TYP="REC" (record) attribute.
- 3) If using OFML over DMPP2020, images may be included as DSEO2020 objects only.
- 4) If using OFML over FoxTalk™, images are indicated as standard HTML objects. Refer to the FoxTalk specification from Computer Projects of Illinois for additional information.

Output messages can also be acknowledgement responses or error messages. These messages are further defined below.

### 3.2.1 Acknowledgement Responses

Acknowledgements messages are sent when a message is received & successfully processed by the message switch. The text of the response message contained in the OFML.RSP.TXT element will contain information pertinent to the message as specified by the current state or agency installation settings, and as such will vary. This information will typically include a list of destination databases and / or the system generated MRI number for the acknowledged input message. All OFML fields will follow the standard output message format with the following specifics for the below listed fields

Element/Attribute	Specification
OFML.HDR.SRC	Set to "SWITCH"
OFML.HDR.MKE	Set to "ACK"

Element/Attribute	Specification
OFML.RSP	Response element
OFML.RSP.TXT	Acknowledgement text element (optional) <TXT> <i>Acknowledgement text</i> </TXT>

### 3.2.2 Error Responses

Error messages are sent when a message is received but is not successfully processed by the message switch. The text of the response message contained in the OFML.RSP.TXT element will contain information pertinent to the processing error as specified by the current state or agency installation settings, and as such will vary. This information will typically include a user readable error message as well as the system generated MRI number for the input message responsible for the returned error. The OFML.RSP.FLD element will identify the specific OFML field responsible for the error when determinable, and the OFML.RSP.COD element will contain a numeric value specific to the generated error message. All OFML fields will follow the standard output message format with the following specifics for the below listed fields

Element/Attribute	Specification
OFML.HDR.SRC	Set to "SWITCH"
OFML.HDR.MKE	Set to "ERR"
OFML.RSP	Response element
OFML.RSP.COD	Error code element – Contains numeric value that specifies the error condition encountered
OFML.RSP.FLD	Field identifier element Tag of OFML element containing the error, unless that OFML element has an EID attribute, in which case that value of the EID attribute will be used instead. This field will only be included if the error is specific to a single identifiable OFML element.
OFML.RSP.TXT	Free text description of error

As noted in the table above, the <FLD> element will identify, when able, the specific OFML element that was responsible for the generated error. For example, if a <DST> (OFML.HDR.DST) field contained an invalid destination code, the <FLD> (OFML.RSP.FLD) element would contain the text "DST". If the OFML element responsible for the error contained an EID attribute, then that value will be substituted for the element tag when identifying the element. To continue the example if the destination field that generated the error were entered as such: <DST EID="DST2">, then the field identifier element in the error would contain "DST2" as such: <FLD>DST2</FLD>. This is a useful technique for situations where there are multiple elements either in the <HDR> or <TRN> element that all use the same XML tag, to uniquely identify the actual field responsible for causing the error.

### 3.2.3 Hot Files Responses

Hot Files response messages are received in response to a query to an OpenFox™ Hot Files database system. These response messages contain a special OFML.RSP.RTYPE element which will identify whether the response message is a positive hit response, a no-hit response, or an error message. The values for the OFML.RSP.RTYPE element are shown in the below table.

OFML.RSP.RTYPE Value	Description
HIT	Positive Hit Response
NOHIT	Negative No-Hit Response
ERROR	Error Message Response

In the case of a positive hit response message, the OFML response element will contain a positive number of RECORDX elements. Each RECORDX element contains the information for one hot files record. The data fields within each RECORDX element may be styled to a user presentation format via a standardized method.

TXT elements are styled by simply presenting the user with the value of the TXT element. A blank TXT element (i.e. <TXT></TXT> or <TXT/>) indicates a new line.

All other elements under the RECORDX element have two optional attributes: “caption” and “eol”. If an element contains the “caption” attribute, then the element’s value is preceded by the value of the caption attribute and a forward slash (/) character. If there is no “caption” attribute, then the value of the element is preceded by the element’s tag and a forward slash (/) character. If the element has a “eol” attribute, and the “eol” attribute has a value of ‘y’, then a new line follows the element’s value. The following table provides some examples.

RECORDX child	Styled Output
<TXT>WANTED PERSON</TXT>	WANTED PERSON
<TXT></TXT>	( <i>newline</i> )
<DOB>19500101</DOB>	DOB/19500101
<DOB eol="y">19500101</DOB>	DOB/19500101( <i>newline</i> )
<DOB caption="DATE OF BIRTH">19500101</DOB>	DATE OF BIRTH/19500101

### 3.2.4 NLETS GJXDM Responses

OFML response messages may deliver GJXDM and/or NEIM data returned from the NLETS network. Currently, these instances are limited to the interstate sharing of driver's license photos. The presence of GJXDM data in an OFML message is indicated via an OFML.RSP.GJXDM element. The OFML.RSP.GJXDM element has a "type" attribute which indicates the type of GJXDM data contained under the element.

At the time of this document, the only valid value for the "type" attribute is "driver" which indicates a GJXDM compatible NLETS formatted driver response. The "driver" type indicates that the GJXDM element may contain one of three child elements. This information is presented in the below table.

Type	Contained Data	Description
<GJXDM type="driver">	n:NLETSResponseData/n:ResponseText	Text formatted driver information
<GJXDM type="driver">	n:NLETSResponseData/n:DriverStatus/n:StandardResponse	NLETS formatted driver status response
<GJXDM type="driver">	n:NLETSResponseData/j2:PersonDigitalImage	NLETS formatted image

## 4 Additional Examples

### 4.1 Client-Initiated Transaction and Responses

DMV query from client assuming the switch spawns DQ to NLETS and QV to NCIC:

```
<OFML>
  <HDR>
    <ID>0204002453</ID>
    <DAC>DEVA</DAC>
    <DAT>20010824073119</DAT>
    <REF>12345ABCDE</REF>
    <MKE>DQ</MKE>
    <ORI>IN1234567</ORI>
    <DST EID="DRI">VA</DST>
    <CTL>ABC1234567</CTL>
    <SUM>DQ: SMITH, GEORGE</SUM>
  </HDR>
  <TRN>
    <NAM>SMITH, GEORGE</NAM>
    <DOB>19511205</DOB>
    <SEX>M</SEX>
  </TRN>
</OFML>
```

ACK from switch:

```
<OFML>
  <HDR>
    <ID>1234509878</ID>
    <SRC>SWITCH</SRC>
    <DAC>DEVA</DAC>
    <DAT>20010824073123</DAT>
    <REF>12345ABCDE</REF>
    <MKE>ACK</MKE>
    <SUM>DQ: SMITH, GEORGE</SUM>
  </HDR>
  <RSP>
    <TXT>
      DQ TO NLETS
      QV TO NCIC
    </TXT>
  </RSP>
</OFML>
```

**Error unrelated to specific field:**

```
<OFML>
  <HDR>
    <ID>1234509878</ID>
    <SRC>SWITCH</SRC>
    <DAC>DEVA</DAC>
    <DAT>20010824073123</DAT>
    <REF>12345ABCDE</REF>
    <MKE>ERR</MKE>
    <SUM>DQ: SMITH, GEORGE</SUM>
  </HDR>
  <RSP>
    <COD>123</COD>
    <TXT>NAM OR OLN REQUIRED</TXT>
  </RSP>
</OFML>
```

**Error related to specific field:**

```
<OFML>
  <HDR>
    <ID>1234509878</ID>
    <SRC>SWITCH</SRC>
    <DAC>DEVA</DAC>
    <DAT>20010824073123</DAT>
    <REF>12345ABCDE</REF>
    <MKE>ERR</MKE>
    <SUM>DQ: SMITH, GEORGE</SUM>
  </HDR>
  <RSP>
    <COD>321</COD>
    <FLD>LIC</FLD>
    <TXT>MAXIMUM FIELD LENGTH OF 30 CHARACTERS EXCEEDED</TXT>
  </RSP>
</OFML>
```



## NLETS response sent to client:

```
<OFML>
  <HDR>
    <ID>1234509876</ID>
    <SRC>NLETS</SRC>
    <DAC>DEVA</DAC>
    <DAT>20010824073123</DAT>
    <REF>12345ABCDE</REF>
    <MKE>DR</MKE>
    <ORI>VA</ORI>
    <DST>IN1234567</DST>
    <CTL>ABC1234567</CTL>
    <SUM>DQ: SMITH, GEORGE</SUM>
  </HDR>
  <RSP>
    <TXT>
      DR.VA
      07:31 08/24/01 17070
      07:31 08/24/01 18545 IN1234567
      TXT
      NAM/SMITH, GEORGE.DOB/19511205.SEX/M
      NO RECORD ON FILE
    </TXT>
  </RSP>
</OFML>
```

## NCIC response sent to client:

```
<OFML>
  <HDR>
    <ID>1234509877</ID>
    <SRC>NCIC</SRC>
    <DAC>DEVA</DAC>
    <DAT>20010824073123</DAT>
    <REF>12345ABCDE</REF>
    <MKE>QV</MKE>
    <DST>IN1234567</DST>
    <CTL>ABC1234567</CTL>
    <SUM>DQ: SMITH, GEORGE</SUM>
  </HDR>
  <RSP>
    <TXT>
      1L01XXXXXXXXXXXXXXXXXX
      IN1234567
      NO RECORD NAM/SMITH, GEORGE DOB/19511205 SEX/M
    </TXT>
  </RSP>
</OFML>
```

Hot File response sent to client containing two records:

```

<OFML>
  <HDR>
    <ID>1234509876</ID>
    <SRC>HFS</SRC>
    <DAC>DEVA</DAC>
    <DAT>20010824073123</DAT>
    <REF>12345ABCDE</REF>
    <MKE>HFR</MKE>
    <DST>IN1234567</DST>
    <CTL>ABC1234567</CTL>
    <SUM>QW: SMITH, GEORGE</SUM>
  </HDR>
  <RSP>
    <RTYPE>HIT</RTYPE>
    <RECORDX TYP="generic">
      <TXT>***** IN STATE WANTED PERSON RECORD *****</TXT>
      <TXT></TXT>
      <LST caption="LAST">SMITH</LST>
      <FST caption="FIRST">GEORGE</FST>
      <MID caption="INITIAL" eol="y">T</MID>
      <SEX>M</SEX>
      <DOB>19601228</DOB>
      <HGT caption="HEIGHT">510</HGT>
      <WGT caption="WEIGHT">150</WGT>
      <EYE eol="y">BRN</EYE>
      <TXT></TXT>
      <ST1 caption="">2521 MERMAID AV</ST1>
      <CIT caption="">WANTAGH</CIT>
      <TXT>, </TXT>
      <STA caption="">GA</STA>
      <ZIP caption="">30101</ZIP>
      <TXT></TXT>
    </RECORDX>
    <RECORDX TYP="generic">
      <TXT>***** IN STATE WANTED PERSON RECORD *****</TXT>
      <TXT></TXT>
      <LST caption="LAST">SMITH</LST>
      <FST caption="FIRST">GEORGE</FST>
      <MID caption="INITIAL" eol="y">F</MID>
      <SEX>F</SEX>
      <DOB>19611016</DOB>
      <HGT caption="HEIGHT">510</HGT>
      <WGT caption="WEIGHT">135</WGT>
      <EYE eol="y">HAZ</EYE>
      <TXT></TXT>
      <ST1 caption="">6325 WOODLAWN DRIVE</ST1>
      <TXT></TXT>
      <CIT caption="">ACWORTH</CIT>
      <TXT>, </TXT>
      <STA caption="">GA</STA>
      <ZIP caption="">30101</ZIP>
      <TXT></TXT>
    </RECORDX>
  </RSP>
</OFML>

```

The records in the response on the previous page would be presented as shown below:

\*\*\*\*\* IN STATE WANTED PERSON RECORD \*\*\*\*\*  
LAST/SMITH FIRST/GEORGE INITIAL/T  
SEX/M DOB/19601228 HEIGHT/510 WEIGHT/150 EYE/BRN

2521 MERMAID AV  
WANTAGH, GA 30101

\*\*\*\*\* IN STATE WANTED PERSON RECORD \*\*\*\*\*  
LAST/SMITH FIRST/GEORGE INITIAL/F  
SEX/F DOB/19611016 HEIGHT/510 WEIGHT/135 EYE/HAZ

6325 WOODLAWN DRIVE  
ACWORTH, GA 30101

## 4.2 Client-Initiated Transaction to Another Client

AM from initiator:

```
<OFML>
  <HDR>
    <ID>0204002454</ID>
    <SRC>DEVA</SRC>
    <DAT>20010824073156</DAT>
    <REF>12345ABCDF</REF>
    <MKE>AM</MKE>
    <ORI>IN1234567</ORI>
    <DST EID="DRI">IN7654321</DST>
    <DST EID="DRI1">IN7654322</DST>
    <SUM>RECORD REQUEST</SUM>
  </HDR>
  <TRN>
    <TXT>
      PLEASE SEND LOCAL RECORDS FOR FOLLOWING SUBJECT:
      NAM/ANDERSON, RAYMOND   OLN/12345678   DOB/010101
      RAC/W   SEX/M
      OFFICER: SCHWARTZ
      AGENCY : CPD
      ADDRESS: 1144 FOSTER AVE, CHARLOTTE
    </TXT>
  </TRN>
</OFML>
```

AM to destination IN7654321:

```
<OFML>
  <HDR>
    <ID>1234509878</ID>
    <SRC>USER</SRC>
    <DAC>DEVB</DAC>
    <DAT>20010824073157</DAT>
    <REF>UNKNOWN</REF>
    <MKE>AM</MKE>
    <ORI>IN1234567</ORI>
    <DST>IN7654321</DST>
    <DST>IN7654321</DST>
    <SUM>RECORD REQUEST</SUM>
  </HDR>
  <TRN>
    <TXT>
      PLEASE SEND LOCAL RECORDS FOR FOLLOWING SUBJECT:
      NAM/ANDERSON, RAYMOND   OLN/12345678   DOB/010101
      RAC/W   SEX/M
      OFFICER: SCHWARTZ
      AGENCY : CPD
      ADDRESS: 1144 FOSTER AVE, CHARLOTTE
    </TXT>
  </TRN>
</OFML>
```

### 4.3 Switch-Initiated Transaction

NCIC dollar message sent to client:

```
<OFML>
  <HDR>
    <ID>1234509878</ID>
    <SRC>NCIC</SRC>
    <DAC>DEVA</DAC>
    <DAT>20010824073123</DAT>
    <MKE>$NICS</MKE>
  </HDR>
  <TRN>
    <TXT>
      $.NICS.DOWN.
      NICS GOING DOWN.
      NICS WILL BE UNAVAILABLE STARTING AT 0110 EST
    </TXT>
  </TRN>
</OFML>
```

NLETS AM sent to client:

```
<OFML>
  <HDR>
    <ID>1234509878</ID>
    <SRC>NLETS</SRC>
    <DAC>DEVA</DAC>
    <DAT>20010824073123</DAT>
    <REF>UNKNOWN</REF>
    <MKE>AM</MKE>
    <ORI>VA</ORI>
    <DST>IN1234567</DST>
  </HDR>
  <TRN>
    <TXT>
      AM.VA1234567
      07:31 08/24/01 17070
      07:31 08/24/01 18545 IN1234567
      TXT
      PLEASE SEND CERTIFIED DRIVING HISTORY ON FOLLOWING SUBJECT:
      NAM/ANDERSON, RAYMOND   OLN/12345678   DOB/010101
      RAC/W   SEX/M
      DRIVERS LICENSE STATUS/REVOKED
      OFFICER: SCHWARTZ
      AGENCY : TBI
      ADDRESS: 1144 FOSTER AVE, NASH
    </TXT>
  </TRN>
</OFML>
```