**PRBC®** **Payment Reporting Builds Credit** sm

# Credim*e*trics

## *Real Time Processing Documentation*
Version 9
August, 2008

PRBCReports.com
Credimetrics.com

Technical Operations Centre
1935 Camino Vida Roble, Suite 260
Carlsbad, CA 92008
760.431.4300
760.431.4304 Fax

# Credimetrics/PRBC Reports System Online Order Processing

## OVERVIEW

The system provides an online, real-time response to requests for merged and unmerged credit data products. Using the 128 bit encryption protection of SSL, the system securely protects your request information and returned credit information over the world wide web.

The system performs the following functions:

- o Accepts your online request.
- o Validates your login account.
- o Validates your consumer identification information.
- o Validates your product options.
- o Retrieves previously acquired credit bureau files.
- o Accesses the national credit repositories using your subscription information and retrieves credit bureau files.
- o Optionally merges the bureau data.
- o Returns back merged XML or merged Report products.
- o Returns back decision results from your lending profile.

## PRODUCTS

The Credmetrics system provides a host of credit reporting products aimed at a variety of needs. Products range from simple credit files passed to the requestor directly from the national credit repositories, to merged credit files and online decisions based upon your credit policies.

The following is a list of the products currently provided:

| Product Code | Product Description |
| --- | --- |
| MRG_XML | Merged Credit file in an xml format. |
| MRG_XMLSTACK | Merged Credit file in an xml format where merged segments are followed by intra-bureau merged segments from each of the three national credit repositories |
| PRBC_XML | PRBC Bill Payment Score Card and Account history in XML Format. |
| PRBC_FICO_XML | FICO Expansion Score in XML format derived from PRBC data and passed input Cra segment. |
| MRG_RPTDTC | Merged Direct-to-consumer credit report in PDF Format. |
| MRG_RPTCOM | Merged Common, lending report. |
| MRG_DECISION | Automated lending decision. |
| OFACRPT | Suspicious persons report based upon entered name and address. |
| OFACXML | Suspicious persons XML file based upon entered name and address. |
| EFXPASSTHRU | Credit file passed directly from Equifax to requestor. |
| EFXCREDITCPU | Equifax cpu Credit file passed to requestor. |
| EFXCREDITTTY | Equifax tty Credit report passed to requestor. |
| EFXDTECTTY | Equifax tty DTEC report passed to requestor. |

| | |
|---|---|
| EXPPASSTHRU | Credit file passed directly from Experian to requestor. |
| EXPCREDITCPU | Experian cpu Credit file passed to requestor. |
| EXPCREDITTTY | Experian tty Credit report passed to requestor. |
| EXPSOCIALTTY | Experian tty Social report passed to requestor. |
| EXPCOLLTTY | Experian tty Collection report passed to requestor. |
| EXPEMPLTTY | Experian tty Employment report passed to requestor. |
| EXPBULSEYETTY | Experian tty BullsEye report passed to requestor. |
| | |
| TUCPASSTHRU | Credit file passed directly from TransUnion to requestor. |
| TUCCREDITCPU | TransUnion cpu credit file passed to requestor. |
| TUCCREDITTTY | TransUnion tty credit report passed to requestor. |
| TUCTRACETTY | TransUnion tty Trace report passed to requestor. |
| TUCTRPLUSTTY | TransUnion tty TracePlus report passed to requestor. |
| TUCFACTTTY | TransUnion tty Fact report passed to requestor. |
| | |
| PRBCPASSTHRU | Bill payment data passed directly from PRBC to requestor. |

Passing Product Requests to

provides an XML request layout that enables multiple product requests to be made. The product code is passed in the CODE attribute of the PRODUCT element within the CMPRODUCTS section of the CMREQUEST XML document.

There is also support for an optional OPTIONS attribute where custom processing options can be submitted. These options are comma delimited name value pairs and are provided by administration personnel.

**Notes on the MRG_XML Product**
The MRG_XML product returns an XML document containing credit history data on the consumer identified in the order. The XML document is contained in its entirety inside the CDATA section of the MRG_XML product response.

The data represents a merged view of any number of bureau or bill payment data sources requested. For example, if only an Equifax bureau is selected, the returned XML document represents a merge of Equifax data within itself, thus removing any duplicates that may be found within the credit file.

**Notes on the MRG_XMLSTACK Product**
The MRG_XMLSTACK product returns an XML document containing credit history data on the consumer identified in the order. The XML document is contained in its entirety inside the CDATA section of the MRG_XMLSTACK product response.

The data represents a stacked merged view of any number of bureau data sources requested. The merged element is followed by up to three child elements, one for each of the national credit repositories. The elements of the child are the same as the elements of the merged parent with the

addition of an INTRA child element containing a Bureau attribute. The possible values of the Bureau attribute are "EFX" for Equifax, "EXP" for Experian and "TUC" for TransUnion. The child element reflects information derived solely from the bureau identified by the Bureau attribute.

## Notes on the PRBC_XML Product
The PRBC_XML product returns an XML document containing non-traditional credit or bill payment history from the PRBC data source. A Bill payment score card is also included. This product requires a valid CMDATASOURCES/PRBC segment with either a profile name or valid PRBC subscriber information.

PRBC GETUPDATED "Y"

In any event where the CMDATASOURCES/PRBC/GETUPDATED element contains the "Y", a fresh PRBC consumer file is retrieved from the PRBC database. This is sometimes used during a re-access of an existing order. Under normal conditions, the re-order simply resends the last response. The GETUPDATED value of "Y" forces a fresh pull from the PRBC consumer database.

## Notes on the PRBC_FICO_XML Product
The PRBC_FICO_XML product returns an XML document containing the FICO Expansion Score designed specifically for PRBC data. It requires the PRBC_XML product to be ordered. The PRBC_XML product must appear before the PRBC_FICO_XML product. Additionally, a FICO defined 'Cra' element is required.

The returned product is the Transaction Segment Envelope from FICO's Expansion Score request. For definitions on the Cra Segment and the Transaction Segment, refer to FICO® Expansion™ Score Services *System-to-System* document from Fair Isaac Credit Services, Inc.

When the product code of PRBC_FICO_XML is requested, it is expected that it a valid CMDATASOURCES/PRBC segment is provided and that the PRBC_XML product precedes the PRBC_FICO_XML product code.

Once an access to PRBC is done on behalf of this credentialed user in the PRBC segment, PRBC evaluates the consumer file to determine if a FICO expansion score request is possible; and if FICO Credentials are on file for this consumer. The eligibility of a PRBC consumer file for a FICO Expansion score is predicated upon the existence of verified account information in that file.

If no verified accounts exists then the file is deemed ineligible for the FICO Expansion score and the Order Incomplete status is returned along with an

Error Segment with the code of 2504 "FICO No Score Returned" and a thin FICO response XML document.

PRBC GETUPDATED "Y"

In any event where the PRBC Datasource GETUPDATED element contains the "Y", a fresh PRBC consumer file is retrieved from the PRBC database.  This is sometimes used during a re-access of an existing order.   Under normal conditions, the re-order simply resends the last response.  The GETUPDATED value of "Y" forces a fresh pull from the PRBC consumer database.

When a fresh pull from the PRBC consumer database occurs, then it is presumed that any existing FICO Expansion Score previously delivered may be outdated; and a new FICO Expansion Score is ordered and delivered.

If the score is possible and those credentials exists then the FICO database is accessed and the Expansion score is returned in the PRBC_FICO_XML product envelope.

USING YOUR FICO SUBSCRIBER CODE

Normally, the FICO Expansion Score is ordered along with a PRBC Report. That PRBC Report usually contains the credentials to access the FICO System.  Optionally, the request can contain an overriding FICO Subscriber code to be used in the request to FICO.

To use this FICO Subscriber code, simply add a FICO DATASOURCE Entry within the CMDATASOURCES Element of the request XML data.   The only attribute necessary is the SUBCODE attribute.

Example:

```
<CMDATASOURCES>
            <PRBC SUBCODE="123" PASSWORD="xyz" PROFILE="" PURPOSE="RP"
                PURPOSEUSER="JOEs MORTGAGE">
                <GETUPDATED>Y<GETUPDATED>
            </PRBC>
            <FICO SUBCODE="123456' />
```

CANNED PRBC/FICO TEST FILES

Currently there exist two test files that contain both PRBC data and FICO data.  Requests can be made using these test file names that circumvent accessing both the PRBC and FICO databases.  These files are handy for demonstrations and downstream testing of the combined PRBC and FICO responses.

Test Case Names
SURNAME="CMTESTSAMPLE"  FIRSTNAME="ONE"
SURNAME="CMTESTSAMPLE"  FIRSTNAME="TWO"

In order to use these samples, a FICO DATASOURCE must be present in the CMDATASOURCES Entry within the CMDATASOURCES Element of the request XML data.  No attributes are necessary, simply the presence of the FICO Element is sufficient.

Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<!—Copyright 2004 Luxor Technologies, Inc.-->
<CMREQUEST REFERENCENUMBER="" ORDERKEY="">
        <CMREQUESTOR LOGINNAME="123" PASSWORD="xyz"/>
        <CMDATASOURCES>
                <PRBC SUBCODE="123" PASSWORD="xyz" PROFILE="" PURPOSE="RP"
                        PURPOSEUSER="JOEs MORTGAGE">
                        <GETUPDATED>Y<GETUPDATED>
                </PRBC>
                <FICO/>
        </CMDATASOURCES>
        <CMCONSUMERS ECOA="I">
                <SUBJECT SURNAME="CMTESTSample" FIRSTNAME="One" MIDNAME="" GENERATION=""
                        SSN="123456789" AGE="01/14/1956"/>
                <ADDRESS TYPE="CURRENT" ADDRESS="123 Main" CITY="Baton Rouge" STATE="LA"
                        POSTALCODE="70805"/>
        </CMCONSUMERS>
        <CMPRODUCTS>
                <PRODUCT CODE="PRBC_XML" OPTIONS=""/>
                <PRODUCT CODE="PRBC_FICO_XML" OPTIONS=" FORCETEST"/>
        </CMPRODUCTS>
</CMREQUEST>
```

Options

FORCETEST

i.e <PRODUCT CODE="PRBC_FICO_XML" OPTIONS="FORCETEST"/>

When setting up a new account; or testing a connection it is handy to order a test file.  PRBC and FICO both have test subjects, however, FICO only provides test subjects on their test system.  When testing, the FORCETEST option can be added to the PRBC_FICO_XML product option segment.  This causes the system to direct FICO requests to their test system.

Example of a PRBC_FICO_XML Request

The following example depicts a request for a PRBC_FICO Expansion Score. The PRBC Datasource depicts the PRBC login credentials. Optionally, those credentials can exists on the  host and be referred to by a PROFILE name.

```
<?xml version="1.0" encoding="UTF-8"?>
<!—edited with XMLSPY v5 rel. 4 U (http://www.xmlspy.com) by Robb Murdock (Luxor Technolgies, Inc.) →
<!—Copyright 2004 Luxor Technologies, Inc.-->
<CMREQUEST REFERENCENUMBER="" ORDERKEY="">
        <CMREQUESTOR LOGINNAME="123" PASSWORD="xyz"/>
        <CMDATASOURCES>
                <PRBC SUBCODE="123" PASSWORD="xyz" PROFILE="" PURPOSE="RP"
                        PURPOSEUSER="JOEs MORTGAGE">
                        <GETUPDATED>Y<GETUPDATED>
                </PRBC>
        </CMDATASOURCES>
        <CMCONSUMERS ECOA="I">
                <SUBJECT SURNAME="Pleasant" FIRSTNAME="Wilbert" MIDNAME="" GENERATION=""
                        SSN="435969633" AGE="01/14/1956"/>

<!—     NOTE CRA SEGMENT ONLY REQUIRED IF PRBC_FICO_XML PRODUCT BEING ORDERED
        CRA REQUIRED FOR EACH APPLICANT (SUBJECT OR SPOUSE)
        SEE FAIR ISSAC EXPANSION SCORE DOCUMENTATION FOR DETAILS.

        NOTE:  ALTHOUGH FICO EXPECTS MIXED CASE FOR ITS XML,  REQUIRES ALL
        XML ELEMENT AND ATTRIBUTE NAMES TO BE UPPER CASE.  THUS THE CRA SEGMENT IS EXPECTED
        TO BE IN ALL UPPER CASE.
→

                        <CRA CRACODE="2" CREDITOBLIGATIONSREPORTEDCOUNT="12"
                        INQUIRIESCOUNT="3" MONTHSSINCEMOSTRECENTINQUIRY="6"
                        MONTHSSINCEMOSTRECENTADVERSEPUBLICRECORD=""
                        MONTHSSINCEMOSTRECENTCOLLECTIONASSIGNED=""
                        MONTHSSINCEOLDESTTRADELINEOPENED="45"
                        MAXIMUMDELINQUENCYEVERCODE="U"
                        MONTHSSINCEMOSTRECENTDELINQUENCY=""
                        CREDITOBLIGATIONSWITHBALANCECOUNT="3"
                        TOTALBALANCESONCREDITREPORTAMOUNT="6544" />


                <SPOUSE SURNAME="" FIRSTNAME="" MIDNAME="" GENERATION="" SSN="" AGE=""/>
                <ADDRESS TYPE="CURRENT" ADDRESS="4437 Clayton Dr" CITY="Baton Rouge" STATE="LA"
                POSTALCODE="70805"/>
        </CMCONSUMERS>
        <CMPRODUCTS>
                <PRODUCT CODE="PRBC_XML" OPTIONS=""/>
                <PRODUCT CODE="PRBC_FICO_XML" OPTIONS=" FORCETEST"/>
        </CMPRODUCTS>
</CMREQUEST>
```

## Example of a PRBC_FICO_XML Product Response.

Within the  response envelope exists the CMDATASETS segment.  Within the CMDATASETS segment exists a CMDATA segment for each returned product.

```
<CMRESPONSE  REFERENCENUMBER=""  ORDERKEY=""  ORDERCOMPLETEDDATE=""  ORDERSTATUS=""
ORDERMESSAGE="">
        <CMERRORS>
                <ERROR CODE="" MESSAGE=""/>
        </CMERRORS>
        <CMDATASETS>
                <CMDATA PRODUCTCODE="" STATUSCODE="" STATUSMESSAGE=""/>
        </CMDATASETS>
```

</CMRESPONSE>

The CMDATA segment for PRODUCTCODE="PRBC_FICO_XML" will contain the Base 64 encoded response from FICO.

Example:

```xml
<DecisionEngine xmlns="">
        <Transaction SubscriberId="12334455" UserName="" TransactionCrossReferenceId="RO0801200700000001"
                TransactionType="Standard" MarketType="ExpansionScore" CreditType="Mortgage"
                ChannelType="SystemToSystem" ApplicationDate="2008-03-07"
                Timestamp="2008-03-07T19:58:41Z"
                TransactionId="JF4FQGF7OHDE5NFLNE">
                <MessageList StatusCode="0" StatusDescription="Successful" />
        </Transaction>
        <ApplicantList>
                <Applicant>
                        <Personal Ssn="435969633" LastName=" Pleasant " FirstName=" Wilbert "
                                MaternalSurname="" MiddleName="" Suffix="">
                                <CurrentAddress City="Baton Rouge" State="LA" Zip="70805" ZipPlus4=""
                                        UnparsedStreetAddress="4437 Clayton Dr " />
                        </Personal>
                        <StandardScoring Score="520">
                                <ReasonCodeList>
                                        <ReasonCode ReasonCodeExplanation="Supplied with app: amount
                                        owed onCRA accounts is too high">IO</ReasonCode>
                                        <ReasonCode ReasonCodeExplanation="Supplied with app: length of
                                         time CRA accounts have been established">MP</ReasonCode>
                                        <ReasonCode ReasonCodeExplanation="Supplied with app: lack of
                                         recent CRA account information">ON</ReasonCode>
                                        <ReasonCode ReasonCodeExplanation="No property records
                                         reported or identified">QG</ReasonCode>
                                </ReasonCodeList>
                        </StandardScoring>
                </Applicant>
        </ApplicantList>
 </DecisionEngine>
```

### Notes on the MRG_RPTCOM Product
The MRG_RPTCOM product returns a TTY report that is 100 characters wide and 60 lines per page; and containing credit history data on the consumer identified in the order.  The TTY report is contained in its entirety inside the CDATA section of the MRG_ RPTCOM product response.

The data represents a merged view of any number of bureau data or bill payment sources requested.  For example, if only an Equifax bureau is selected, the returned TTY document represents a merge of Equifax data within itself, thus removing any duplicates that may be found within the credit file.

### Notes on the MRG_RPTDTC Product
The MRG_RPTDTC product returns a direct-to-consumer cred report that is formatted in the Adobe Acrobat PDF format; and containing credit history data on the consumer identified in the order.  The PDF report is contained in its entirety inside the CDATA section of the MRG_ RPTDTC product response.

The data represents a merged view of any number of bureau data sources requested. For example, if only an Equifax bureau is selected, the returned PDF document represents a merge of Equifax data within itself, thus removing any duplicates that may be found within the credit file.

### Notes on the MRG_DECISION Product

The MRG_DECISION product returns a set of decision results from a customized decision profile stored in the system. This product may need additional information in order to process the script such as 'Loan Amount' or 'Stated Income' or other such variables not normally found in a credit data set.

To convey these application-oriented variables; and to specify the decision profile name to execute, a section exists in the CMREQUEST XML document. In the CMREQUEST section, the decision profile name is set in the PROFILENAME element; and the optional parameters are set in multiple PARAMETER elements using the NAME and VALUE attributes. The NAME represents the name that the script will use to refer to the VALUE.

### Notes on the OFACTTY/OFACXML Products

The OFACTTY and OFACXML products perform a search of the Suspicious Persons data provided by the United States Treasury Department. This product requires no DATASOURCE to be selected as the data is completely located within the system. The data is updated daily from a feed from the Treasury Department.

The search is based upon the entered Subject Name and/or Current Address. Both products can be returned on a single request just as other products are ordered in the same transaction. Each product is encased in its own CDATA section.

The OFACTTY is a 100 character wide TTY report with no page breaks. It lists the date, time and subject on which the search is conducted; and lists each possible match along with aliases and addresses that match the suspicious person listed.

The OFACXML product is a direct equivalent of the TTY product and is the basis upon which the TTY product is built.

### Notes on the EFXPASSTHRU, EXPPASSTHRU and TUCPASSTHRU Products

The 'passthru' products provided by are intended to serve those requests where the original bureau data is required. Using these product codes may be considered illogical when ordering merged products unless the 'passthru'

product is a credit file in cpu format. This is because the merged products depend on cpu format credit files to operate.

When one of the 'passthru' product codes is submitted, the desired product from each of the bureaus to be passed is set in the PASSTHRUPRODUCT field for the particular bureau in the CMDATASOURCES section of the CMREQUEST XML document.

While it is possible to receive multiple products, only a single credit file from each of the repositories can be obtained through  via a single request. Hence, a EFXPASSTHRU of a EFXDTECTTY report cannot be made in the same transaction that requests a three bureau merged credit report.

## *REALTIME PROCESSING*

The system is engineered to accept a single, individual request for merged or unmerged credit data via a published Web Service. This service expects a request XML stream identifying the customer, the customer's repository subscription information, the consumer's identification information and general request information identifying a customer reference number, individual or joint indications, which bureaus to process and what products to return. Multiple products can be returned on a single request.

When the system receives the request XML via the Web Service, it begins processing the request; and returns a responseXML when the product is completed. Hence, the process is REALTIME.

During this processing, telecommunication tasks are conducted with the national credit and bill payment repositories. When the telecommunications processes are concluded, the products are fulfilled and returned.

It is possible for one or more of the national repositories to return a No Record; or have some outage. The outage can be temporary or an outage that is caused by some problem with the requesting information such as the customer's bureau subscriber code data.

In any case, the system continues and produces the requested products based on what credit data it has. An indication is made in the response that identifies the product as complete or incomplete and states the reason why it would be incomplete.

In the response is also a unique order number. Using this order number in a subsequent order will re-use any existing, good credit data; and will re-order any missing credit data.

How Realtime Processing Works
The realtime ordering process consists of:

1. Receiving the customer's order in a Request XML file via a Web Service.
2. Validating it.
3. Returning a response XML immediately if validation fails.
4. Retrieving any existing, good credit data based upon the inclusion of a valid, existing OrderKey.
5. Ordering any new or missing data based upon the bureau's being requested.
6. Fulfilling the requested products.
7. Returning the products back in a response XML file.

Placing the Real Time Order

Currently the only means of placing a Real Time Order is via the SOAP Listener. This SOAP Listener is a Web Service that accepts an XMLRequest and returns an XMLResponse.

## Merge SOAP Listener

The Credimeterics Merge SOAP Listener is a published Web Service. The interface definition is found at:

https://webservice..com/merge/CMMergeListener.wsdl

This Web Service Definition Language file is shown below:

```xml
<?xml version="1.0" encoding="UTF-8" ?>
  <!—
 Generated 12/02/02 by Microsoft SOAP Toolkit WSDL File Generator, Version 1.02.813.0
  
- <definitions name="CMMergeListener" targetNamespace="http://tempuri.org/wsdl/"
    xmlns:wsdlns="http://tempuri.org/wsdl/" xmlns:typens="http://tempuri.org/type"
    xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:stk="http://schemas.microsoft.com/soap-toolkit/wsdl-extension"
    xmlns="http://schemas.xmlsoap.org/wsdl/">
- <types>
  <schema targetNamespace="http://tempuri.org/type"
    xmlns="http://www.w3.org/2001/XMLSchema" xmlns:SOAP-
    ENC="http://schemas.xmlsoap.org/soap/encoding/"
    xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" elementFormDefault="qualified" />
    </types>
- <message name="Order.Place">
  <part name="RequestXML" type="xsd:string" />
  <part name="ResponseXML" type="xsd:base64Binary" />
    </message>
- <message name="Order.PlaceResponse">
  <part name="ResponseXML" type="xsd:base64Binary" />
    </message>
- <portType name="OrderSoapPort">
- <operation name="Place" parameterOrder="RequestXML ResponseXML">
  <input message="wsdlns:Order.Place" />
  <output message="wsdlns:Order.PlaceResponse" />
    </operation>
    </portType>
- <binding name="OrderSoapBinding" type="wsdlns:OrderSoapPort">
```

```
<stk:binding preferredEncoding="UTF-8" />
<soap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http" />
- <operation name="Place">
<soap:operation soapAction="http://tempuri.org/action/Order.Place" />
- <input>
<soap:body use="encoded" namespace="http://tempuri.org/message/"
    encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
    </input>
- <output>
<soap:body use="encoded" namespace="http://tempuri.org/message/"
    encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
    </output>
    </operation>
    </binding>
- <service name="CMMergeListener">
- <port name="OrderSoapPort" binding="wsdlns:OrderSoapBinding">
<soap:address location="https://webservice..com/CMMergeListener.WSDL" />
    </port>
    </service>
    </definitions>
```

## RequestXML

The request sent to  Merge contains the following information:

- o Order information – Customer reference number, Previous OrderKey, Which Bureaus to Use, Individual or Joint accessing, What Products to return.

- o Customer Information – Customer login id and password.

- o Data Source Information – Subscriber codes for each of the national credit and bill payment repositories, or a profile name of a pre-defined subscription profile, or a stream of raw bureau data in lieu of accessing the national repositories.  Additionally, the name of a pass-thru product to be returned directly from the repository.

- o Consumer's identification information – Subject Name, Spouse Name, Current and Former Address information.

- o Product request information – A set of product codes reflecting the products requested.

- o Decision parameters – The decision profile name and a set of parameter names and values.

The RequestXML schema can be found at:
https://www...com/merge/Schemas/CMRequest.xsd

The RequestXML schema is shown below:

```xml
<?xml version="1.0" encoding="UTF-8" ?>
- <!--
  edited with XMLSPY v5 rel. 4 U (http://www.xmlspy.com) by Robb Murdock (Luxor Technologies, Inc.)
  W3C Schema generated by XMLSPY v5 rel. 4 U (http://www.xmlspy.com)
  -->
- <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
- <xs:element name="ADDON">
- <xs:complexType>
  <xs:attribute name="MODELID" type="xs:string" />
    </xs:complexType>
    </xs:element>
- <xs:element name="ADDRESS">
- <xs:complexType>
  <xs:attribute name="TYPE" type="xs:string" use="required" />
  <xs:attribute name="ADDRESS" type="xs:string" use="required" />
  <xs:attribute name="CITY" type="xs:string" use="required" />
  <xs:attribute name="STATE" type="xs:string" use="required" />
  <xs:attribute name="POSTALCODE" type="xs:string" use="required" />
    </xs:complexType>
    </xs:element>
- <xs:element name="CMCONSUMERS">
- <xs:complexType>
- <xs:sequence>
  <xs:element ref="SUBJECT" />
  <xs:element ref="SPOUSE" />
  <xs:element ref="ADDRESS" />
    </xs:sequence>
  <xs:attribute name="ECOA" type="xs:string" use="required" />
    </xs:complexType>
    </xs:element>
- <xs:element name="CMDATASOURCES">
- <xs:complexType>
- <xs:sequence>
  <xs:element ref="EQUIFAX" />
  <xs:element ref="EXPERIAN" />
  <xs:element ref="TRANSUNION" />
  <xs:element ref="PRBC" />
    </xs:sequence>
    </xs:complexType>
    </xs:element>
- <xs:element name="CMDECISION">
- <xs:complexType>
- <xs:sequence>
  <xs:element ref="PARAMETER" />
    </xs:sequence>
  <xs:attribute name="PROFILENAME" type="xs:string" use="required" />
    </xs:complexType>
    </xs:element>
- <xs:element name="CMPRODUCTS">
- <xs:complexType>
- <xs:sequence>
  <xs:element ref="PRODUCT" />
    </xs:sequence>
    </xs:complexType>
    </xs:element>
- <xs:element name="CMREQUEST">
- <xs:complexType>
- <xs:sequence>
  <xs:element ref="CMREQUESTOR" />
  <xs:element ref="CMDATASOURCES" />
```

```
      <xs:element ref="CMCONSUMERS" />
      <xs:element ref="CMPRODUCTS" />
      <xs:element ref="CMDECISION" />
          </xs:sequence>
      <xs:attribute name="REFERENCENUMBER" type="xs:string" use="required" />
      <xs:attribute name="ORDERKEY" type="xs:string" use="required" />
          </xs:complexType>
          </xs:element>
-   <xs:element name="CMREQUESTOR">
-   <xs:complexType>
      <xs:attribute name="LOGINNAME" type="xs:string" use="required" />
      <xs:attribute name="PASSWORD" type="xs:string" use="required" />
          </xs:complexType>
          </xs:element>
-   <xs:element name="CRA">
      <xs:complexType />
          </xs:element>
-   <xs:element name="EQUIFAX">
-   <xs:complexType>
-   <xs:sequence>
      <xs:element ref="SCORE" />
      <xs:element ref="ADDON" />
          </xs:sequence>
      <xs:attribute name="PREAMBLE" type="xs:string" use="required" />
      <xs:attribute name="SUBCODE" type="xs:string" use="required" />
      <xs:attribute name="PASSWORD" type="xs:string" use="required" />
      <xs:attribute name="PROFILE" type="xs:string" use="required" />
      <xs:attribute name="PURPOSE" type="xs:string" use="required" />
      <xs:attribute name="PURPOSEUSER" type="xs:string" use="required" />
          </xs:complexType>
          </xs:element>
-   <xs:element name="EXPERIAN">
-   <xs:complexType>
-   <xs:sequence>
      <xs:element ref="SCORE" />
      <xs:element ref="ADDON" />
          </xs:sequence>
      <xs:attribute name="PREAMBLE" type="xs:string" use="required" />
      <xs:attribute name="SUBCODE" type="xs:string" use="required" />
      <xs:attribute name="PASSWORD" type="xs:string" use="required" />
      <xs:attribute name="PROFILE" type="xs:string" use="required" />
      <xs:attribute name="PURPOSE" type="xs:string" use="required" />
      <xs:attribute name="PURPOSEUSER" type="xs:string" use="required" />
          </xs:complexType>
          </xs:element>
-   <xs:element name="GETUPDATED">
      <xs:complexType />
          </xs:element>
-   <xs:element name="PARAMETER">
-   <xs:complexType>
      <xs:attribute name="NAME" type="xs:string" use="required" />
      <xs:attribute name="VALUE" type="xs:string" use="required" />
          </xs:complexType>
          </xs:element>
-   <xs:element name="PRBC">
-   <xs:complexType>
-   <xs:sequence>
      <xs:element ref="GETUPDATED" />
          </xs:sequence>
      <xs:attribute name="SUBCODE" type="xs:string" use="required" />
      <xs:attribute name="PASSWORD" type="xs:string" use="required" />
      <xs:attribute name="PROFILE" type="xs:string" use="required" />
      <xs:attribute name="PURPOSE" type="xs:string" use="required" />
      <xs:attribute name="PURPOSEUSER" type="xs:string" use="required" />
          </xs:complexType>
          </xs:element>
-   <xs:element name="PRODUCT">
-   <xs:complexType>
```

```
    <xs:attribute name="CODE" type="xs:string" use="required" />
    <xs:attribute name="OPTIONS" type="xs:string" use="required" />
        </xs:complexType>
        </xs:element>
-   <xs:element name="SCORE">
-   <xs:complexType>
    <xs:attribute name="MODELID" type="xs:string" />
        </xs:complexType>
        </xs:element>
-   <xs:element name="SPOUSE">
-   <xs:complexType>
-   <xs:sequence>
    <xs:element ref="CRA" />
        </xs:sequence>
    <xs:attribute name="SURNAME" type="xs:string" use="required" />
    <xs:attribute name="FIRSTNAME" type="xs:string" use="required" />
    <xs:attribute name="MIDNAME" type="xs:string" use="required" />
    <xs:attribute name="GENERATION" type="xs:string" use="required" />
    <xs:attribute name="SSN" type="xs:string" use="required" />
    <xs:attribute name="AGE" type="xs:string" use="required" />
        </xs:complexType>
        </xs:element>
-   <xs:element name="SUBJECT">
-   <xs:complexType>
-   <xs:sequence>
    <xs:element ref="CRA" />
        </xs:sequence>
    <xs:attribute name="SURNAME" type="xs:string" use="required" />
    <xs:attribute name="FIRSTNAME" type="xs:string" use="required" />
    <xs:attribute name="MIDNAME" type="xs:string" use="required" />
    <xs:attribute name="GENERATION" type="xs:string" use="required" />
    <xs:attribute name="SSN" type="xs:string" use="required" />
    <xs:attribute name="AGE" type="xs:string" use="required" />
        </xs:complexType>
        </xs:element>
-   <xs:element name="TRANSUNION">
-   <xs:complexType>
-   <xs:sequence>
    <xs:element ref="SCORE" />
    <xs:element ref="ADDON" />
        </xs:sequence>
    <xs:attribute name="MARKET" type="xs:string" use="required" />
    <xs:attribute name="SUBMARKET" type="xs:string" use="required" />
    <xs:attribute name="SUBCODE" type="xs:string" use="required" />
    <xs:attribute name="PASSWORD" type="xs:string" use="required" />
    <xs:attribute name="INDUSTRYCODE" type="xs:string" use="required" />
    <xs:attribute name="PROFILE" type="xs:string" use="required" />
    <xs:attribute name="PURPOSE" type="xs:string" use="required" />
    <xs:attribute name="PURPOSEUSER" type="xs:string" use="required" />
        </xs:complexType>
        </xs:element>
        </xs:schema>
```

## Request XML

A sample RequestXML xml file can be found at:

https://www..com/Merge/Schemas/CMRequest.xml

The RequestXML sample XML file is shown below:

```
    <?xml version="1.0" encoding="UTF-8" ?>
-   <!--
    edited with XMLSPY v5 rel. 4 U (http://www.xmlspy.com) by Robb Murdock (Luxor Technologies, Inc.)
    Copyright 2004 Luxor Technologies, Inc.
```

```
-->
- <CMREQUEST REFERENCENUMBER="" ORDERKEY="">
  <CMREQUESTOR LOGINNAME="" PASSWORD="" />
- <CMDATASOURCES>
- <EQUIFAX PREAMBLE="" SUBCODE="" PASSWORD="" PROFILE="" PURPOSE="" PURPOSEUSER="">
  <SCORE />
  <ADDON MODELID="" />
    </EQUIFAX>
- <EXPERIAN PREAMBLE="" SUBCODE="" PASSWORD="" PROFILE="" PURPOSE="" PURPOSEUSER="">
  <SCORE MODELID="" />
  <ADDON />
    </EXPERIAN>
- <TRANSUNION MARKET="" SUBMARKET="" SUBCODE="" PASSWORD="" INDUSTRYCODE="" PROFILE=""
      PURPOSE="" PURPOSEUSER="">
  <SCORE MODELID="" />
  <ADDON />
    </TRANSUNION>
- <PRBC SUBCODE="" PASSWORD="" PROFILE="" PURPOSE="" PURPOSEUSER="">
  <GETUPDATED />
    </PRBC>
    </CMDATASOURCES>
- <CMCONSUMERS ECOA="">
- <SUBJECT SURNAME="" FIRSTNAME="" MIDNAME="" GENERATION="" SSN="" AGE="">
  <CRA />
    </SUBJECT>
- <SPOUSE SURNAME="" FIRSTNAME="" MIDNAME="" GENERATION="" SSN="" AGE="">
  <CRA />
    </SPOUSE>
  <ADDRESS TYPE="" ADDRESS="" CITY="" STATE="" POSTALCODE="" />
    </CMCONSUMERS>
- <CMPRODUCTS>
  <PRODUCT CODE="" OPTIONS="" />
    </CMPRODUCTS>
- <CMDECISION PROFILENAME="">
  <PARAMETER NAME="" VALUE="" />
    </CMDECISION>
    </CMREQUEST>
```

Notes on XMLRequest

The following is a partial data dictionary for the RequestXML file

| Node | Attribute | Notes |
|---|---|---|
| CMREQUEST | REFERENCENUMBER | Optional. Value is extracted and used in CMResponse to sync the response to the request. |
| CMREQUEST | ORDERKEY | Optional. Used to reference an existing order's bureau data. Useful for re-ordering from a previously incomplete order. |
| CMREQUEST | ECOA | Optional. If used must contain I for Individual or J for Joint. I is default. |
| CMCONSUMER/SUBJECT | SURNAME,FIRSTNAME & SSN | **Required.** SSN must be 9 digit numeric. |
| CMCONSUMER/SPOUSE | SURNAME,FIRSTNAME & SSN | Optional but required if ECOA is J. |
| CMCONSUMER/SUBJECT OR SPOUSE | GENERATION | Optional. Valid values are: J,S,JR,SR,1-6,I- VI |
| CMPRODUCTS/PRODUCT | CODE | **Required**. **MRG_XML** – Merged credit data suitable for normal merged applications. **MRG_XMLSTACK** – Merged credit data suitable for normal or stacked |

| | | |
|---|---|---|
| | | merged applications.<br>**PRBC_XML** – Non traditional bill payment data with Bill Payment Score Card.<br><br>**PRBC_FICO_XML** – FICO Expansion Score XML document derived from PRBC file plus input CRA Segment.<br><br>**MRG_RPTCOM** – Formatted consumer lending credit report.<br>**MRG_RPTDTC** – Formatted direct to consumer report in PDF format.<br>**MRG_DECISION** – Decision results based upon passed parms and customer decision script profile.<br>**OFACTTY** – Suspicious persons report in TTY format.<br>**OFACXML** – Suspicious persons report in XML format.<br><br>**NOTE:** For other products, see Product Codes in the Products section of this document. |
| | OPTIONS | Optional. Contains comma delimited string of name, value option pairs. See Credmetrics administration personnel for values. |
| CMDECISION | PROFILENAME | <span style="color:red">**Required if MRG_DECISION product code is specified**</span>. Name of a customized decision profile script to be used to provide decision. |
| CMDECISION/PARAMETER | NAME/VALUE | Optional. Set of name value pairs representing specific parameters to be passed to the script for processing. |
| CMDATASOURCES/bureau | Various elements | Subscriber code information for each of the requested bureaus including Permissible Purpose code and the End User name associated with the purpose code. Alternatively, all subscriber code info can be stored at Credmetrics and referenced by the PROFILE name. |

## Merge ResponseXML

The response returned from Credmetrics Merge contains the following information:

- o Order information – Customer reference number, Assigned OrderKey, CompletionStatus, CompletionDate and CompletionMessage.

- o Error Information – If any errors or found a set of errors are returned containing an ErrorCode and ErrorMessage.

o Data – Returned products containing Type of data and the returned product in a CDATA section.

## ResponseXML

The ResponseXML schema can be found at:
https://webservice.credmetrics.com/Schemas/CMResponse.xsd

The ResponseXML schema is shown below:

```xml
<?xml version="1.0" encoding="UTF-8" ?>
- <!--
 edited with XMLSPY v5 rel. 2 U (http://www.xmlspy.com) by Robb Murdock (Luxor Technologies,
   Inc.)
  -->
- <!--
W3C Schema generated by XMLSPY v5 rel. 2 U (http://www.xmlspy.com)
  -->
- <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
     elementFormDefault="qualified">
  - <xs:element name="CMData">
    - <xs:complexType mixed="true">
        <xs:attribute name="ProductCode" type="xs:string" use="required" />
        <xs:attribute name="StatusCode" type="xs:string" use="required" />
        <xs:attribute name="StatusMessage" type="xs:string" use="required" />
      </xs:complexType>
    </xs:element>
  - <xs:element name="CMDataSets">
    - <xs:complexType>
      - <xs:sequence>
          <xs:element ref="CMData" />
        </xs:sequence>
      </xs:complexType>
    </xs:element>
  - <xs:element name="CMErrors">
    - <xs:complexType>
      - <xs:sequence>
          <xs:element ref="Error" />
        </xs:sequence>
      </xs:complexType>
    </xs:element>
  - <xs:element name="CMResponse">
    - <xs:complexType>
      - <xs:sequence>
          <xs:element ref="CMErrors" />
          <xs:element ref="CMDataSets" />
        </xs:sequence>
        <xs:attribute name="ReferenceNumber" type="xs:string" use="required"
           />
        <xs:attribute name="OrderKey" type="xs:string" use="required" />
        <xs:attribute name="OrderCompletedDate" type="xs:string"
           use="required" />
        <xs:attribute name="OrderStatus" type="xs:string" use="required" />
        <xs:attribute name="OrderMessage" type="xs:string" use="required" />
      </xs:complexType>
    </xs:element>
```

```
- <xs:element name="Error">
    - <xs:complexType>
        <xs:attribute name="Code" type="xs:string" use="required" />
        <xs:attribute name="Message" type="xs:string" use="required" />
      </xs:complexType>
  </xs:element>
</xs:schema>
```

A sample ResponseXML xml file can be found at:

https://webservice..com/ Schemas/CMResponse.xml

The ResponseXML sample XML file is shown below:

```
<CMRESPONSE REFERENCENUMBER="" ORDERKEY="" ORDERCOMPLETEDDATE="" ORDERSTATUS=""
ORDERMESSAGE="" XMLNS:XSI="HTTP://WWW.W3.ORG/2001/XMLSCHEMA-INSTANCE">
    <CMERRORS>
        <ERROR CODE="" MESSAGE=""/>
    </CMERRORS>
    <CMDATASETS>
        <CMDATA PRODUCTCODE="" STATUSCODE="" STATUSMESSAGE=""/>
    </CMDATASETS>
</CMRESPONSE>
```

Notes on XMLResponse

The following is a partial data dictionary for the ResponseXML file

| Node | Attribute | Notes |
|---|---|---|
| CMRESPONSE | REFERENCENUMBER | Optional.  Will be returned and contain same value as found in CMRequest/ReferenceNumber |
| CMRESPONSE | ORDERKEY | Optional.   Will contain either new or existing assigned OrderKey if a product is returned.   Will not be returned if input transaction could not be processed. |
| CMRESPONSE | ORDERCOMPLETEDATE | Optional.   Will be set if the order is deemed complete.   That is, the OrderStatus is set to Complete. |
| CMRESPONSE | ORDERSTATUS, ORDERMESSAGE | **Constant**. Status Codes and English text to indicate status of the order. <br><br> CODE / DESCRIPTION <br> 0 / Order Completed <br> 1 / Order Incomplete <br> 2 / Order Not processed <br><br> The OrderMessage may contain extended verbiage to expound upon status. |
| CMRESPONSE/ERRORS | | Optional.  Will contain Error Nodes if any input or processing errors were encountered. |
| ERROR | CODE, MESSAGE | Constant if Error Node is present. Codes and Messages indicating any form of error. |

| | | CODE | DESCRIPTION |
|---|---|---|---|
| | | 0001 | Communications failure. *(Returned from the (CMMergeClient.Request object only).* |
| | | 1001 | Customer invalid |
| | | 1002 | Order invalid |
| | | 1003 | Decision processing error |
| | | 1004 | Invalid XML request data |
| | | 1005 | Login processing error |
| | | 1006 | Product processing error |
| | | 1007 | Data sources processing error |
| | | 1008 | Bureau server processing error |
| | | 1009 | Product code parsing error |
| | | 1010 | Order completion processing error. |
| | | 2001 | Bureau communications Error |
| | | 2002 | Bureau data incomplete |
| | | 2101 | Equifax timed out |
| | | 2102 | Equifax communications Error |
| | | 2103 | Equifax returned error. |
| | | 2201 | Experian timed out |
| | | 2202 | Experian communications Error |
| | | 2203 | Experian returned error. |
| | | 2301 | TransUnion timed out |
| | | 2302 | TransUnion communications Error |
| | | 2303 | TransUnion returned error. |
| | | 2401 | PRBC timed out |
| | | 2402 | PRBC communications Error |
| | | 2403 | PRBC returned error. |
| | | 2404 | PRBC No record. |
| | | 2501 | FICO timed out. |
| | | 2502 | FICO communications Error. |
| | | 2503 | FICO Error in Decision File |
| | | 2504 | FICO No score returned. |
| | | 3000 | Merge processing Error |
| | | 4000 | Decision error. |

Message may contain extended verbiage to expand upon error.

| | | |
|---|---|---|
| CMRESPONSE/DATASETS/DATA | PRODUCTCODE | Defines type of data to follow. |

| TYPE | DESCRIPTION |
|---|---|
| MRG_XML | Merged credit data in XML format |
| PRBC_XML | PRBC data in XML Format |
| PRBC_FICO_XML | FICO Transaction |

| | | response in XML format |
|---|---|---|
| | MRG_RPTCOM | Merged credit report in TTY format |
| | MRG_RPTDTC | Merged credit Report in PDF format. |
| | MRG_DECISION | Decision results in XML format |
| | OFACRPT | Suspicious persons report in TTY format |
| | OFACXML | Suspicious persons report in XML format |
| | | |
| | For other products see the Product Codes in the Product Section of this document. | |
| CMRESPONSE/DATASETS/DATA | STATUSCODE | 0 = Complete 1 = Incomplete |
| CMRESPONSE/DATASETS/DATA | | Base64 Encoded Product Response. |

## MRG_XML/MRG_XMLSTACK Credit Data

The MRG_XML and MRG_XMLSTACK products are XML data sets and are found in the CDATA section of the CMResponse.XML structure when the MRG_XML or MRG_XMLSTACK products are ordered.

The MRG_XML and MRG_XMLSTACK data sets are organized as 15 Segments:

IDENTIFICATION
ADDRESS
EMPLOYMENT
INQUIRY
CONSUMER_STATEMENT
CREDITOR
AKA
CONSUMER_REFERRAL
BANKING
SCORE
MISCELLANEOUS
PUBLIC_RECORD
BUREAU_ERROR
TRADELINE

## Merged Bureau Distinction

These segments represent a merged view of credit data from any number of repositories. Based upon a SOURCE field, the segment can be identified as representing a combination of Equifax, Experian or TransUnion.

Similarly, not all of the bureaus may consider an item as derogatory. As such credit measuring segments such as TRADELINE and PUBLIC_RECORD contain a DEROGIND field which indicates which bureaus consider the data derogatory.

SOURCE and DEROGIND EXAMPLE:

| 1 | Equifax |
|---|---|
| 2 | Experian |
| 3 | Equifax, Experian |
| 4 | TransUnion |
| 5 | Equifax,TransUnion |
| 6 | Experian,TransUnion |
| 7 | Equifax,Experian,TransUnion |

**MRG_XMLSTACK Stacked Merge Data**

While the 15 top level segments adequately represent a merged data set, often it is desirous to also list a stacked merge data set. In this fashion, a cross bureau merged picture can be shown as well as a picture of each of the bureaus.

With the exception of the BUREAU_ERROR Segment, all segments contain a sub node named INTRA. The INTRA node contains the same field nodes as its parent node; and represents the un-merged credit data from each of the bureaus. There will be up to three INTRA nodes within each segment. Each INTRA node represents one credit repository as indicated by a Bureau attribute.

EXAMPLE:

INTRA Bureau="EFX"     Identifying this sub-segment as originating from Equifax.
INTRA Bureau="EXP"     Identifying this sub-segment as originating from Experian.
INTRA Bureau="TUC"     Identifying this sub-segment as originating from TransUnion

A schema for the XML_MERGE product can be found at
https://webservice..com/schemas/mergeresponse.xsd,
as well as a sample xml file at
https://webservice.credmetrics.comschemas/mergeresponse.xml.

## USING THE CLIENT OBJECT

Customer systems operating in the Microsoft environment can take advantage of the Credmetrics Merge Client Object. This object accepts parameters from the customer's system, creates the appropriate XML document and interacts with the Credmetrics SOAP listener at the Credmetrics host.

Customers using this object will typically be web-based ASP (Active Server Page) applications, enterprise developed applications using Visual Basic or any other Microsoft Windows application that can interface with an ActiveX DLL.

NOTE: THE CLIENT OBJECT CURRENTLY DOES NOT SUPPORT THE PRBC_FICO_XML PRODUCT.

CMMERGE CLIENT METHODS
The CMMergeClient Object's REQUEST class provides the following methods:

### CreateXMLRequestString()
Creates an XML document from the current settings. Useful for applications that wish to perform their own SOAP dialogue with ; and simply need a way to create the XML document

### GetResponse(*ByVal ProductCode As String, ByRef StatusCode As String, ByRef StatusMessage As String*) As String
Extracts the product data for the specified ProductCode from the returned XML document from . Additionally, any status code or message related to that particular product code is returned in the parameters.

### PlaceOrder()
Sends the XML request document to the SOAP Listener and waits for the response.

### SetRequestor(*ByVal LoginName As String, ByVal Password As String*)
Sets the customer login name and password for authentication with .

### WriteResponseToDisk(*ByVal ProductCode, ByVal ReportPath As String, _*
*ByRef StatusCode As String, ByRef StatusMessage As String*) As
Integer
Similar to the GetResponse method, the WriteResponseToDisk method extracts the product data for the specified ProductCode from the returned XML document from and writes that data to a local disk file. Additionally, any status code or message related to that particular product code is returned in the parameters.

CMMERGE CLIENT PROPERTIES
TheCMMergeClient Object's REQUEST class provides the following properties:

### CompletionDate  – returns the date and time from  that the order was completed.

**OrderKey** – The order key assigned by  when the product is returned; or the order key assigned by Credmetrics to be referenced on a re-access.

**OrderStatus** – The status code of the current order.
**OrderMessage** – The English equivalent of the status code.

CMMERGE CLIENT CLASSES
The CMMergeClient Object provides the following classes:

**CONSUMER** - A class that contains information regarding the consumer on which the credit data is being ordered.

Properties:
**ECOA** Valid values are "I" for Individual or "J" for Joint access.

Methods:
**SetSubject**(*ByVal Surname As String, ByVal FirstName As String, _*
*ByVal MidName As String, ByVal Generation As String, _*
*ByVal SSN As String, ByVal Age As String*)

**SetSubjectCRA**(*ByVal CRASegment As String*)
Used to add a fully formatted CRA segment as defined by the FICO Expansion Score specifications.  To conform with ' edit rules, however, the XML tags MUST be upper case.

**SetSpouse**(*ByVal Surname As String, ByVal FirstName As String, _*
*ByVal MidName As String, ByVal Generation As String, _*
*ByVal SSN As String, ByVal Age As String*)
Used to add a fully formatted CRA segment as defined by the FICO Expansion Score specifications.  To conform with ' edit rules, however, the XML tags MUST be upper case.

**SetSpouseCRA**(*ByVal CRASegment As String*)

**SetCurrentAddress**(*ByVal Address As String, ByVal City As String, ByVal State As String, ByVal ZipCode As String*)

**SetFormerAddress**(*ByVal Address As String, ByVal City As String, ByVal State As String, ByVal ZipCode As String*)

**DATASOURCES** - A class containing repository subcodes, profiles and pass through product information

Properties:

**UseEquifax** as boolean. TRUE if requesting Equifax data
**UseExperian** as boolean. TRUE if requesting Experian data
**UseTransUnion** as boolean. TRUE if requesting TransUnion data
**UsePRBC** as Boolean. TRUE if requesting PRBC data
**PRBCGetUpdated** as Boolean. TRUE if requesting an updated PRBC file after an original PRBC file has been ordered for this OrderKey.  NOTE: this only affects the behavior of a re-ordered product.  Normally, re-ordered products use the same PRBC data as from a previous order.  Setting this value to TRUE causes a new PRBC data file to be used for this re-ordered product.


Methods:

**SetEquifax**(*ByVal Preamble As String, ByVal SubCode As String, _*
*ByVal Password As String, ByVal Purpose As String, _*
*ByVal EndUser as string, ByVal Profile as String, _*
*Optional ByVal PassThruProduct as string, _*
*Optional ByVal RawData As String, _*
*Optional  ByVal RawDataFile As String*)


**SetEquifaxScoreModels**(*ByVal ModelIds as string*)
Comma delimited string of score model ids.
**SetEquifaxAddOns**(*ByVal AddOns as string*)
Comma delimited string of add on features.


**SetExperian**(*ByVal Preamble As String, ByVal SubCode As String, _*
*ByVal Password As String, ByVal Purpose As String, _*
*ByVal EndUser as string, ByVal Profile as string, _*
*Optional ByVal PassThruProduct as String, _*
*Optional  ByVal RawData As String,_*
*Optional  ByVal RawDataFile As String*)


**SetExperianScoreModels**(*ByVal ModelIds as string*)
Comma delimited string of score model ids.
**SetExperianAddOns**(*ByVal AddOns as string*)
Comma delimited string of add on features.


**SetTransUnion**(*ByVal Market As String, ByVal SubMarket As String, ByVal*
*IndustryCode As String,  ByVal SubCode As String, _*
*ByVal Password As String, ByVal Purpose As String, _*
*ByVal EndUser as string, ByVal Profile as string, _*
*Optional ByVal PassThruProduct as String, _*
*Optional ByVal RawData As String, _*
*Optional ByVal RawDataFile As String*)

**SetTransUnionScoreModels**(*ByVal ModelIds as string*)
    Comma delimited string of score model ids.
**SetTransUnionAddOns**(*ByVal AddOns as string*)
    Comma delimited string of add on features.


**SetPRBC**(*ByVal SubCode As String, _
        ByVal Password As String, ByVal Purpose As String, _
        ByVal EndUser as string, ByVal Profile as string, _
        Optional ByVal RawData As String, _
        Optional ByVal RawDataFile As String*)


**DECISIONS**        - A class containing the name of the decision profile to be run and any optional decision parameters required by the profile.

Properties:
**ProfileName** as string.   The name of the decision profile being executed.
**ResultsNameList** as string array. An array of names of results parameters returned by the decision.
**XML** as string.      The XML product data returned by the REQUEST class is passed to the DECISION class via this property for processing.
**PRBCGetUpdated** as Boolean.  True forces  to obtain a fresh copy of a PRBC data file even though one already has been accessed from PRBC.

Methods:
**AddParameters**(*ByVal ParmName As String, ByVal ParmValue As String*)
Adds a decision request parameter by name and value to an internal collection of parameters used by the decision.

**GetResultsByName**(*ByVal ParmName As String*)
Retrieves the value for a given decision result parameter name.

**PRODUCTS**        - A class containing product codes to be ordered in this request.

Properties:
**ProductsList** as Collection. A Visual Basic Collection containing the list of all the products being returned.

Methods:
**AddProduct**(*ByVal ProductCode As String, Optional ByVal Options as string*)
Adds a product code to the request XML document.

**RemoveProduct**(*ByVal ProductCode As String*)
Removes a product code from the request XML document.

**RemoveAllProducts**()
Removes all product codes from the request XML document.

---

**REQUEST** - The main class of the CMMergeClient containing references to other classes and basic properties and methods. (See the CMMERGE CLIENT METHODS for details.)

## CLIENT OBJECT SAMPLES
The code herein represents a sample of various ordering scenarios.

### ORDERING MERGED PRODUCTS
This example orders all of the merged products in a single order.

```
Dim oCMMergeClient As New CMMergeClient.Request
Dim i As Integer
Dim MyStatusCode as string
Dim MyStatusMessage as string
'************************************************************************
'* SET THE LOGIN INFO
'************************************************************************
oCMMergeClient.SetRequestor txtLoginName, txtPassword
'************************************************************************
'* SET THE CONSUMER INFO
'************************************************************************
oCMMergeClient.Consumer.ECOA = "I"
oCMMergeClient.Consumer.SetCurrentAddress "111 AVERY RD", "ROME", "GA", "30361"
oCMMergeClient.Consumer.SetSubject "CRAWFORD", "ROBERT", "M", "JR", "143777890", "28"
'************************************************************************
'* SET THE BUREAU DATA SOURCE INFO
'************************************************************************
' USE MY OWN PRIVATE EQUIFAX SUBCODE AND PASSWORD INFO
oCMMergeClient.Datasources.SetEquifax MyPreamble, MySubCode, MyPassword, "", "", ""
oCMMergeClient.Datasources.UseEquifax = True


' REFERENCE THE EXPERIAN PROFILE ON FILE AT CREDMETRICS
oCMMergeClient.Datasources.SetExperian "", "", "","","", "MYEXPPROFILE"
oCMMergeClient.Datasources.UseExperian = True

'DON'T USE ANY TRANS UNION DATA IN THIS REQUEST
oCMMergeClient.Datasources.UseTransUnion = False

' ADD PRBC DATA TO THIS REQUEST
oCMMergeClient.DataSources.SetPRBC  "", "","","","MYPRBCPROFILE"
oCMMergeClient.DataSources.UsePRBC = True


'************************************************************************
'* SET THE PRODUCT REQUEST INFO
'************************************************************************
oCMMergeClient.Products.AddProduct "MRG_XML"
oCMMergeClient.Products.AddProduct "MRG_RPTCOM"
oCMMergeClient.Products.AddProduct "MRG_RPTDTC"
oCMMergeClient.Products.AddProduct "MRG_DECISION"
oCMMergeClient.Products.AddProduct "OFACXML"
'************************************************************************
'* SINCE DECISION WAS REQUESTED, SET THE ADDITIONAL DECISION INFO REQUIRED
'* BY THE 'MYDECISIONPROFILE' SCRIPT ON FILE AT CREDMETRICS
'************************************************************************
oCMMergeClient.Decisions.AddParameters "INCOME", "50000"
oCMMergeClient.Decisions.AddParameters "LOANAMOUNT", "39544"
oCMMergeClient.Decisions.ProfileName = "MYDECISIONPROFILE"

Me.lstErrors.Clear
oCMMergeClient.PlaceOrder

txtCompletionDate = oCMMergeClient.CompletedDate
txtOrderKey = oCMMergeClient.OrderKey
txtOrderMessage = oCMMergeClient.OrderMessage
txtOrderStatus = oCMMergeClient.OrderStatus
If oCMMergeClient.OrderStatus = "0" Then
    MyTextBox = oCMMergeClient.GetResponse ("MRG_RPTCOM", MyStatusCode, MyStatusMessage)
    oCMMergeClient.WriteResponseToDisk "OFACXML", "C:\ofacxml.xml", "", ""
```

```
    oCMMergeClient.WriteResponseToDisk "MRG_RPTCOM", "C:\mrg_rptcom.txt", "", ""
    oCMMergeClient.WriteResponseToDisk "MRG_RPTDTC", "C:\mrg_rptdtc.pdf", "", ""
    oCMMergeClient.Decisions.XML = oCMMergeClient.GetResponse("MRG_DECISION", "", "")
    MyForm.txtPassFail = oCMMergeClient.Decisions.GetResultsByName("PASSFAIL")
    MyForm.txtReason = oCMMergeClient.Decisions.GetResultsByName("REASON")
Else
    For i = 1 To oCMMergeClient.Errors.Count
        MyForm.lstErrors.AddItem oCMMergeClient.Errors(i)("CODE") & "-" &
oCMMergeClient.Errors(i)("MESSAGE")
    Next i
End If
```

## ORDERING PASS THROUGH PRODUCTS

## This example orders an Equifax DTEC TTY report from Equifax.

```
Dim oCMMergeClient As New CMMergeClient.Request
Dim i As Integer
'****************************************************************************
'* SET THE LOGIN INFO
'****************************************************************************
oCMMergeClient.SetRequestor MyForm.txtLoginName, MyForm.txtPassword
'****************************************************************************
'* SET THE CONSUMER INFO
'****************************************************************************
oCMMergeClient.Consumer.SetSubject "", "", "", "", "143777890", ""
'****************************************************************************
'* SET THE BUREAU DATA SOURCE INFO
'****************************************************************************
oCMMergeClient.Datasources.SetEquifax txtPreamble, txtSubCode, txtPassword, _
                "", "", "", " EFXDTECTTY"
oCMMergeClient.Datasources.UseEquifax = True
oCMMergeClient.Datasources.UseExperian = false
oCMMergeClient.Datasources.UseTransUnion = False
'****************************************************************************
'* SET THE PRODUCT REQUEST INFO
'****************************************************************************
oCMMergeClient.Products.AddProduct " EFXPASSTHRU"

Me.lstErrors.Clear
oCMMergeClient.PlaceOrder

MyForm.txtCompletionDate = oCMMergeClient.CompletedDate
MyForm.txtOrderKey = oCMMergeClient.OrderKey
MyForm.txtOrderMessage = oCMMergeClient.OrderMessage
MyForm.txtOrderStatus = oCMMergeClient.OrderStatus
If oCMMergeClient.OrderStatus = "0" Then
    oCMMergeClient.WriteResponseToDisk " EFXPASSTHRU ", "C:\efxdtecreport.txt", "", ""
Else
    For i = 1 To oCMMergeClient.Errors.Count
        MyForm.lstErrors.AddItem oCMMergeClient.Errors(i)("CODE") & "-" &
oCMMergeClient.Errors(i)("MESSAGE")
    Next i
End If
```

## ORDERING PRBC XML AND FICO EXPANSION SCORE

This example orders a PRBC XML file and a FICO Expansion Score.
Note: The FICO Expansion score expects a CRA segments formatted to their specifications to be attached to each borrower (Subject and/or Spouse).

```
Dim oCMMergeClient As New CMMergeClient.Request
Dim i As Integer
'*******************************************************************************
'* SET THE LOGIN INFO
'*******************************************************************************
oCMMergeClient.SetRequestor MyForm.txtLoginName, MyForm.txtPassword
'*******************************************************************************
'* SET THE CONSUMER INFO (SUBJECT)
'*******************************************************************************
oCMMergeClient.Consumer.SetSubject MyForm.txtSubLastName, _
        MyForm.txtSubFirstName, MyForm.txtSubMidName, _
        MyForm.txtSubSSN, MyForm.txtSubDOB

oCMMergeClient.Consumer.SetSubjectCRA myFormattedSubjectCRASegment


'*******************************************************************************
'* SET THE CONSUMER INFO (SPOUSE)
'*******************************************************************************
oCMMergeClient.Consumer.SetSpouse MyForm.txtSpoLastName, _
        MyForm.txtSpoFirstName, MyForm.txtSpoMidName, _
MyForm.txtSpoSSN, MyForm.txtSpoDOB

oCMMergeClient.Consumer.SetSubjectCRA myFormattedSpouseCRASegment


'*******************************************************************************
'* SET THE BUREAU DATA SOURCE INFO
'*******************************************************************************
oCMMergeClient.Datasources.SetPRBC MyForm.txtPRBCSubcode, _
        MyForm.txtPRBCSubCode, MyForm.txtPRBCPassword, _
        MyForm.txtPRBCPurpose, MyForm.txtPRBCEndUser, _
        MyForm.txtPRBCProfile
oCMMergeClient.Datasources.UsePRBC = True
'*******************************************************************************
'* SET THE PRODUCT REQUEST INFO
'* NOTE: PRBC_XML MUST PRECEDE PRBC_FICO_XML
'*******************************************************************************
oCMMergeClient.Products.AddProduct "PRBC_XML"
oCMMergeClient.Products.AddProduct "PRBC_FICO_XML"

Me.lstErrors.Clear
oCMMergeClient.PlaceOrder

MyForm.txtCompletionDate = oCMMergeClient.CompletedDate
MyForm.txtOrderKey = oCMMergeClient.OrderKey
MyForm.txtOrderMessage = oCMMergeClient.OrderMessage
MyForm.txtOrderStatus = oCMMergeClient.OrderStatus
If oCMMergeClient.OrderStatus = "0" Then
    oCMMergeClient.WriteResponseToDisk " PRBC_XML ", "C:\prbc.xml", "", ""
    oCMMergeClient.WriteResponseToDisk " PRBC_FICO_XML ", "C:\prbc_fico.xml", "", ""
Else
    For i = 1 To oCMMergeClient.Errors.Count
        MyForm.lstErrors.AddItem oCMMergeClient.Errors(i)("CODE") & "-" &
oCMMergeClient.Errors(i)("MESSAGE")
    Next i
End If
```