



VARS and VRXML Implementation Guide

Version: 1.2
Date: 8/4/10

1.0 INTRODUCTION..... 3

1.1 INTRODUCTION TO VARs 3

1.2 INTRODUCTION TO VRXML 3

1.3 REPORT SUBMISSION..... 4

2.0 VARs 1.3 IMPLEMENTATION..... 5

2.1 ENHANCEMENTS FROM VARs 1.0/1.1 TO VARs 1.3 5

2.2 CME GROUP REPORTING REQUIREMENTS FOR VARs 1.3 5

2.3 UNPROCESSED FIELDS AND FIELD VALUES 6

2.4 UNPROCESSED RECORD TYPES 7

3.0 CME GROUP VRXML 1.1 IMPLEMENTATION..... 8

3.1 CME GROUP REPORTING REQUIREMENTS FOR VRXML 1.1 8

3.2 UNPROCESSED ELEMENTS FOR VRXML 1.1..... 8

3.3 SAMPLE VRXML 1.1 FILE LAYOUT..... 11

4.0 REVISION HISTORY..... 14

CME®, iLink® EOS Trader™, Galax-C®, the Globex logo, and CME Globex® are registered trademarks of Chicago Mercantile Exchange Inc.
 CME Group is a trademark of CME Group Inc.
 Copyright © 2009 CME Group. All rights reserved.

NYSE is a registered trademark of the New York Stock Exchange, Inc.

TCB Data Systems is a registered trademark of TCB Data Systems, Inc.

1.0 Introduction

VARs and VRXML files are two accepted file types for submitting monthly Market Data reports to CME Group. The requirements outlined below apply to all VARs or VRXML files submitted to CME Group, regardless of how they are submitted.

Note: CME Group Supports VARs 1.3, and VRXML 1.1. Not all features of VARs 1.3 and VRXML 1.1 will be supported.

For any policy questions, please contact marketdata@cmegroup.com.

1.1 Introduction to VARs

The preferred file format for reporting monthly market data is VRXML, but CME Group also supports the VARs format on a transitional basis. VARs is a file format initially developed by TCB Data Systems Incorporated for Market Data vendors to store and report Market Data inventory. To report, the file must be submitted to CME Group for processing.

Each VARs file contains five segments of data, referred to as records. The available record types are: Header Record (Type-0), Customer Data Record (Type-1), Transaction Record (Type-2), Trailer Record (Type-8), and End of File Record (Type-9). Each record is made up of a specific collection of fields populated with values. The sequence of record types in the VARs file is as important as the values within the individual fields.

1.2 Introduction to VRXML

VRXML (Vendor Reporting eXtensible Markup Language) is the preferred file format for Market Data vendors to store and report Market Data inventory. To report, the file must be submitted to CME Group for processing.

VRXML was initially developed by the New York Stock Exchange (NYSE), and is based on the XML protocol. It uses a particular schema developed for use in reporting Market Data inventory. Each VRXML file contains layers of elements, or segments of data, that are embedded within higher level elements. All layers are contained within the VRXML root element.

1.3 Report Submission

Please contact marketdata@cmegroup.com regarding submitting market data reporting files to CME Group.

2.0 VARs 1.3 Implementation

The VARs 1.0 and 1.1 specifications are being deprecated and replaced by the VARs 1.3 specification, which is for CME Group customers only.

Note: All VARs files submitted to CME Group must have the extension .vrf.

2.1 Enhancements from VARs 1.0/1.1 to VARs 1.3

VARs 1.3 includes the following enhancements:

- In the Header Record (Type-0), files of VARs versions later than 1.0 are now identified by “1.1” or “1.3” in the Report Version field.
- In the Transaction Record (Type-2), for files of VARs versions later than 1.0, the Usage Quantity and Balance Quantity fields have been increased to 10-digit quantities.
- The Vendor ID fields in the Header Record (Type-0) and the End of File Designator (Type-9) record have been expanded to 6 characters.

2.2 CME Group Reporting Requirements for VARs 1.3

- For the Header Record (Type-0):
 - For Online submission, the vendor id is validated against the vendor id in the file. If they do not match, an error will be reported.
 - For FTP submission, the vendor id is validated against the vendor id on the filename. If they do not match, an error will be reported.
- For the Customer Data Record (Type-1):
 - The Customer Data information required for reporting reflects standard CME Group Market Data reporting policy. Please contact marketdata@cmegroup.com with any reporting policy questions.
- For the Transaction Record (Type-2):

- The inventory reporting dates (Effective Date) found in the VARs file should be within the period for which you are reporting. Files with dates for more than one month will be rejected. Please note that the due date for reporting and policies related to late reporting reflect standard CME Group Market Data reporting policy. Please contact marketdata@cmegroup.com with any reporting policy questions.
- The Effective Date field is required for all Transaction Records (Type-2) on file. The effective date(s) must fall within the same reporting month and year. If the above conditions are not satisfied, an error will be reported.
- If multiple Type-2 records are reported for a subscriber for the same product, the usage will be calculated as the sum of the quantities reported on the different lines.

2.3 Unprocessed Fields and Field Values

Beginning with VARs 1.3, CME Group does not process the fields and field values indicated below. The data found in these fields will be obtained from other fields.

Header Record (Type-0)

For the Header Record (Type-0):

- ‘T’ (Net change transaction) is not a supported value. Only the Report Type value ‘D’ (Full Inventory) is supported.
- The Date File Created field is not processed
- The Time File Created field is not processed

Customer Data Record (Type-1)

The following fields will not be processed:

- Contact Name
- Contact Telephone
- Comment

Transaction Record (Type-2)

For the Transaction Record (Type-2):

- The Subscriber ID field is not processed, but if provided, it must match the Subscriber ID field on Customer Data Record (Type-1). If the Subscriber IDs do not match, an error will be reported.
- For the Action Code field, 'D' (Delete) is not a supported value. 'A' (Add) is the only supported value.

2.4 Unprocessed Record Types

The following record types are required but will not be processed or validated:

- Trailer Record (Type-8)
- End of File Record (Type-9)

3.0 CME Group VRXML 1.1 Implementation

This section contains important information that is necessary to ensure proper processing of VRXML files when reporting Market Data inventory to CME Group.

Note: All VRXML files submitted to CME Group must have the extension .xml

3.1 CME Group Reporting Requirements for VRXML 1.1

1. The inventory reporting dates (effectiveDate) found in the VRXML file should be within the period for which you are reporting. Files with dates for more than one month will be rejected. Please note that the due date for reporting and policies related to late reporting reflect standard CME Group Market Data reporting policy. Please contact marketdata@cmegroup.com with any reporting policy questions.
2. The inventory reporting dates (effectiveDate) does not have to be the same on every inventory entry in the submission file, but must fall within the same reporting month and year.
3. If there are multiple inventory entries in a file for a single product, the sum of the quantities will be calculated. The sum of the entries will be considered the inventory entry for that product. An additional inventoryEntry element with a Product Action Code of UPD will NOT overwrite a previous inventory entry in the file for a product.
4. All VRXML files will be validated against the NYSE XSD (XML Schema Document) for accuracy of data submitted. If the file does not pass this validation, an error will be reported. For more information on VRXML, refer to the NYSE VRXML 1.1 specification or contact: marketdata@cmegroup.com.

3.2 Unprocessed Elements for VRXML 1.1

CME Group will not process all elements found in the VRXML 1.1 files. For more information on VRXML, refer to the NYSE VRXML 1.1 specification or contact marketdata@cmegroup.com.

CME Group will ignore the following elements of VRXML 1.1 during processing, even though some/all of the tags/elements are required for the validation process. The data found in these elements will be obtained from other elements.

1. VRXML Element

- a. CME Group will not verify that the total inventory you report is the same as the sum of all inventories for the products that are listed in the file. You must calculate the total inventory and report it.

2. 1st Layer Elements

- a. Vendor Entry (none are processed)
 - i. Vendor Name changes
 - ii. Billing address changes
 - iii. Contact changes (adding, deleting and updating)
- b. Subscriber New Entry
 - i. Contact information
 - ii. Billing address changes
 - iii. Company
- c. Subscriber Change Entry
 - i. Subscriber's contact changes
 - ii. Billing address changes
 - iii. Company
 - iv. Subscriber's new locations

3. 2nd Layer Elements

- a. Location new
 - i. Contact information for the new location
 - ii. Comment
 - iii. Inventory (found in first layer)
- b. Location Change

- i. Changes to the location's contact information
 - ii. Comment
- c. Contact (none are processed)
 - i. Deletion of contacts
 - ii. Addition of contacts
 - iii. Person information for contact
 - iv. Address information for contact

4. Core Data Elements

- a. Company (not processed)
 - i. Company-related information changes
- b. Person (not processed)
 - i. Person-related information changes

CME Group will ignore the **codeOwner** element of VRXML 1.1 files during processing, but the element must be included to ensure proper processing of the files.

3.3 Sample VRXML 1.1 File Layout

Line	Tag Example
1	<?xml version="1.0" encoding="UTF-8" ?/>
2	<tns:vrxml xmlns:tns="http://mds.nyseddata.com/vrxml/2004/1.1" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://mds.nyseddata.com/vrxml/2004/1.1 http://mds.nyseddata.com/vrxml/2004/1.1/VRXML.xsd" >
3	<tns:fileDescriptor tns:vendorId=12345>
4	<tns:dateFileCreated>2009-05-14T15:16:24</tns:dateFileCreated>
5	<tns:entryMetaData tns:entry="inventoryEntry">
6	<tns:count>2</tns:count>
7	</tns:entryMetaData>
8	<tns:entryMetaData tns:entry="subscriberNewEntry">
9	<tns:count>1</tns:count>
10	</tns:entryMetaData>
11	<tns:entryMetaData tns:entry="subscriberChangeEntry">
12	<tns:count>1</tns:count>
13	</tns:entryMetaData>
14	</tns:fileDescriptor>
15	<tns:inventoryEntry tns:VAN="COMPANYA">
16	<tns:product tns:actionCode="ADD">
17	<tns:productCode tns:codeOwner="VEN" tns:code="BOTACC"/>
18	<tns:inventory>
19	<tns:effectiveDate>2009-04-02</tns:effectiveDate >
20	<tns:quantity>2</tns:quantity>
21	<tns:comments />
22	</tns:inventory>
23	<tns:effectiveDate>2009-04-02</tns:effectiveDate />
24	</tns:product>
25	</tns:inventoryEntry>
26	<tns:inventoryEntry tns:VAN="COMPANYB"/>
27	<tns:product tns:actionCode="ADD"/>
28	<tns:productCode tns:codeOwner="VEN" tns:code="BOTDSV" />
29	<tns:inventory>
30	<tns:effectiveDate>2009-04-02</tns:effectiveDate />
31	<tns:quantity>3</tns:quantity />
32	<tns:comments />
33	</tns:inventory>
34	<tns:effectiveDate>2009-04-02</tns:effectiveDate >
35	</tns:product>

Line	Tag Example
36	</tns:inventoryEntry>
37	<tns:subscriberNewEntry>
38	<tns:company>
39	<tns:name1>FIRM2</tns:name1 >
40	<tns:name2 / >
41	</tns:company>
42	<tns:locationNew tns:VAN="COMPANYB">
43	<tns:locals>
44	<tns:address>
45	<tns:address1>345 Main St</tns:address1>
46	<tns:address2>Ste. 100</tns:address2>
47	<tns:address3 />
48	<tns:city>Chicago</tns:city>
49	<tns:stateProvince>IL</tns:stateProvince=IL>
50	<tns:postalCode>23456</tns:postalCode>
51	<tns:country>US</tns:country=US>
52	</tns:address>
53	<tns:company>
54	<tns:name1>FIRM2</tns:name1>
55	<tns:name2 />
56	</tns:company>
57	</tns:locals>
58	<tns:inventoryEntry tns:VAN="COMANYB">
59	<tns:product tns:actionCode="ADD">
60	<tns:productCode tns:codeOwner="VEN" tns:code="BOTDSV" />
61	<tns:inventory>
62	<tns:effectiveDate>2009-04-02</tns:effectiveDate>
63	<tns:quantity>33</tns:quantity>
64	<tns:comments />
65	</tns:inventory>
66	<tns:effectiveDate>2009-04-02</tns:effectiveDate>
67	</tns:product>
68	</tns:inventoryEntry>
69	</locationNew>
70	</tns:subscriberNewEntry>
71	<tns:subscriberChangeEntry tns:refVAN="COMPANYA">
72	<tns:locationChange tns:VAN="COMPANYA">
73	<tns:locals>
74	<tns:address>

Line	Tag Example
75	<tns:address1>123 Main St.</tns:address1>
76	<tns:address2>Ste 800</tns:address2>
77	<tns:address3></tns:address3>
78	<tns:city>Chicago</tns:city>
79	<tns:stateProvince>IL</tns:stateProvince >
80	<tns:postalCode>12345</tns:postalCode
81	<tns:country>US</tns:country>
82	</tns:address>
83	<tns:company>
84	<tns:name1>COMPANYA</tns:name1>
85	<tns:name2 />
86	</tns:company>
87	</tns:locals>
88	</tns:locationChange>
89	</tns:subscriberChangeEntry>
90	</tns:vrxml>

4.0 Revision History

Initial Release	Version	Last Update	Author	Description
7/14/09	1.0	N/A	DT	Initial release.
7/14/09	1.1	9/15/09	DT	<ul style="list-style-type: none"> • added section 2.2 “Product Code Translation” • Added information on product translation codes to section 4.2: “Unprocessed Elements for VRXML 1.1.” • Updated 4.3: “Sample VRXML 1.1 File Layout” with codeOwner=”VEN”
7/14/09	1.2	8/4/10	DT	<ul style="list-style-type: none"> • Removed content related to previous COWS functionality. • Updated for new policy changes.