

eMARS Check Writer Interface Files



FINAL

Prepared for
**Commonwealth of
Kentucky**
eMARS Project

Version 2.4

April 20, 2006

Version/Issue Date	Brief Summary of Changes From Prior Version
V2.0 – 10/05/2005	Initial version.
V2.1 – 10/27/2005	<ul style="list-style-type: none"> • Corrected the attribute name PYTMT_REMT_ADV in the Payment Component table of Appendix A to PYMT_REMT_ADV. • Changed the XML tag "AMS_DATAOBJECT_XML_IMPORT_FILE" to "AMS_DATAOBJECT_XML_EXPORT_FILE" where applicable in the whole document. • Specified allowable Event Types in the Accounting Component table of Appendix A, which are: CA01 (when coding Object on an accounting line) and CA02 (when coding Revenue Source on an accounting line). • Specified the three Miscellaneous Vendor Codes, which are. ZZMISCINDV for "individual", ZZMISCORP for "corporation" and ZZMISCPART for "partnership".
V2.3 – 03/06/2006	<ul style="list-style-type: none"> • Corrected the attribute name LGL-NM in the Payment Component table of Appendix A to LGL_NM • Changed the attribute name ACH_ACCT_NO in the Payment Component table to ACH_ACCT_NO_VIEW which will invoke a process to encrypt the account number provided in the check writer file. • Modified section 3.3.2 to note that a file containing only ACH pre-note transactions is not required to contain an accounting component. • Modified section 3.4 to present new method to construct CW_FILE_ID field.
V2.4 - 04/05/2006	<ul style="list-style-type: none"> • Corrected the Rover ID C31R188A for LRC Session Mileage Interface to C10R188A in Appendix B – Inventory of Check Writer Interface Files. • Corrected the Rover ID C31R1890 for LRC Travel Data Interface to C10R1890 in Appendix B – Inventory of Check Writer Interface Files. • Corrected the Rover ID C35R085D for Teachers Retirement – Retiree Payroll - Ancillary Interface to C33R085D in Appendix B – Inventory of Check Writer Interface Files. • Corrected the Rover ID C35R085C for Teachers Retirement – Retiree Payroll - Refund Interface to C33R085C in Appendix B – Inventory of Check Writer Interface Files. • Corrected the Rover ID C35R2100 for Treasury – for employee child support to C31R2100 in Appendix B – Inventory of Check Writer Interface Files. • Corrected the misspelling of weekly in the Frequency column of the Treasury – for employee child support interface information located in Appendix B - Inventory of Check Writer Interface Files. • Added CHFS MED KCHIP (KY Children’s Health Insurance Plan) Interface to Appendix B – Inventory of Check Writer Interface Files. • Added Revenue – Disbursements of Communication Tax Receipts Interface to Appendix B – Inventory of Check Writer Interface Files. • Added Revenue School Tax–Distribution of Tax Receipts to Schools Interface to Appendix B – Inventory of Check Writer Interface Files. • Corrected the Rover ID C31R046U for Kentucky Retirement – KERS Interface to C31R046A in Appendix B – Inventory of Check Writer Interface Files.

	<ul style="list-style-type: none"> • Corrected the Rover ID C31R046P for Kentucky Retirement – KERS – Employer 25C Refunds Interface to C31R046U in Appendix B – Inventory of Check Writer Interface Files • Corrected the Rover ID C31R218P for Kentucky Retirement – KERS – Member 25C Refunds Interface to C31R046P in Appendix B – Inventory of Check Writer Interface Files. • Corrected the Rover ID C31R046S for Kentucky Retirement – CERH – Employer 25C Refunds Interface to C31R046X in Appendix B – Inventory of Check Writer Interface Files. • Corrected the Rover ID C31R218Y for Kentucky Retirement – KERH – Employer 25C Refunds to C31R046Y in Appendix B – Inventory of Check Writer Interface Files. • Corrected the Rover ID C31R130C for LGEAF – Coal Severance/Mineral Severance Non CAPS, CITY Interface to C31R130D in Appendix B – Inventory of Check Writer Interface Files. • Added GOT – Communications – CMRS Interface to Appendix B – Inventory of Check Writer Interface Files. • Corrected the Rover ID C44F051L for Labor Cabinet – CWPf Pneumoconiosis interest – L Interface to C44R051L in Appendix B – Inventory of Check Writer Interface Files. • Corrected the Rover ID C48R16A Impact Plus CFC – New Version Interface to C48R016A in Appendix B – Inventory of Check Writer Interface Files. • Corrected the Rover ID C48R040G KAMES daily issuance kinship care start-up Interface to C48R040T in Appendix B – Inventory of Check Writer Interface Files.

Table of Contents

TABLE OF CONTENTS	2
1 INTRODUCTION	5
1.1 PURPOSE OF THE CHECK WRITER INTERFACE DESIGN	5
1.2 DOCUMENT DEFINITIONS AND CONVENTIONS	5
1.3 LOGICAL CHECK WRITER FILE COMPONENT STRUCTURE	6
1.4 EMARS GENERAL XML SPECIFICATIONS	7
1.5 EMARS ACCOUNTING CONCEPT	9
2 OVERVIEW OF CHECK WRITER APPROACH.....	11
2.1 BACKGROUND INFORMATION	11
2.2 KEEPING CURRENT WITH CHANGES.....	11
2.3 INTERFACE TESTING	11
2.4 ASSUMPTIONS.....	12
3 CHECK WRITER INTERFACE PROCESS	14

3.1 CHECK WRITER TABLES 14

 3.1.1 Check Writer Option Table 14

 3.1.2 Check Writer Pre-Assigned Check Numbers Table..... 14

 3.1.3 Check Writer Header Table 14

 3.1.4 Check Writer Accounting Table 15

 3.1.5 Check Writer Payment Table 15

 3.1.6 Check Writer ACH Addendum Table 16

 3.1.7 Check Writer Vendor Intercept Table 16

3.2 CHECK WRITER BATCH PROCESSES..... 16

 3.2.1 Check Writer Load Process..... 16

 3.2.2 Check Writer Pre-Edit Process 16

 3.2.3 Check Writer Clean-up Process 17

3.3 STRUCTURE OF THE AGENCY CHECK WRITER FILE 17

 3.3.1 Check Writer Header Component (R_AP_CW_HDR)..... 18

 3.3.2 Check Writer Accounting Component (R_AP_CW_ACTG)..... 18

 3.3.3 Check Writer Payment Component (R_AP_CW_PYMT) 18

 3.3.4 Check Writer ACH Addendum Component (R_AP_CW_ADNM) 18

 3.3.5 Check Writer in MARS vs. eMARS..... 19

3.4 CHECK WRITER FILE ID 20

3.5 LOADING CHECK WRITER FILES TO ADVANTAGE 3 APPLICATION..... 21

3.6 CERTIFYING CHECK WRITER FILES 22

3.7 PROCESSING CHECK WRITER FILES 22

3.8 CHECK STOCK AND NUMBERING 24

3.9 1099 POSTING..... 24

APPENDIX A - ADVANTAGE 3 CHECK WRITER XML INPUT FILE LAYOUT 26

APPENDIX B – INVENTORY OF CHECK WRITER INTERFACE FILES 46

APPENDIX C – EXAMPLE OF A CHECK WRITER INPUT XML FILE 61

1 Introduction

1.1 Purpose of the Check Writer Interface Design

The purpose of this design is to provide agencies of the Commonwealth of Kentucky the information needed to modify their current check writer systems to interface to the new eMARS Check Writer (CW) process and format. The Check Writer process will provide the capability to write checks outside the Automated Disbursement (AD) process and capture all the necessary information for 1099 reporting, check cancellation and check reconciliation. eMARS is an integrated software system that will require inbound (interfaced) data to be submitted in a XML (Extensible Markup Language) format. The goal of this document is to provide the specifications needed to submit a fully qualified XML check writer file into the eMARS system.

This document explains the mechanisms and guidelines for external systems to send information to eMARS. Agencies are responsible for developing and modifying their check writer systems to be submitted into the eMARS system.

1.2 Document Definitions and Conventions

Terms	Definitions	Examples
XML Tag	The tag used in XML file to define the attribute	DEPT_CD
Caption	Field name of the data element on respective table	
Description	Textual description of the data element	
R/C	Required/Conditional	<p>Required (R) indicates that the data element is required by the interface for transaction to successfully process.</p> <p>Conditionally Required (CR) means that specified conditions must be met by the interface for transaction to process successfully. When not required the field can be spaces.</p> <p>Optional (O) specifies that the data element is optional for the interface and can be</p>

Terms	Definitions	Examples
		populated with spaces.
Type	Data type of element	<p>Date – Date Format “yyyy-mm-dd”</p> <p>Number – A whole number whose value can not have either positive or negative sign and can not have a decimal position. An example of a number value is (4) = 9999.</p> <p>Decimal – A whole number whose value can be either positive or negative. An example of a decimal value with a specified decimal position is (4, 3) = 9999.999. Negative numbers must be prefixed with a negative sign.</p> <p>VARCHAR2 – Alphanumeric (can be a combination of letters, number and special characters) field with a <u>maximum</u> character size defined for each field.</p>
Max Size	The maximum numeric size of the data element	The numeric size of the data element
Comments	Further description of the field or specific comments that relate to the eMARS project.	

1.3 Logical Check Writer File Component Structure

The check writer input XML file layout is comprised of five components listed below. Refer to **Appendix C** for an example of a fully qualified Check Writer XML file.

XML Tag	Description	R/C
AMS_DATAOBJECT_XML_EXPORT_FILE	Unique Tag used to identify the beginning and ending of a check writer file.	R

XML Tag	Description	R/C
R_AP_CW_HDR	Unique Tag used to identify beginning and ending of a header component.	R
R_AP_CW_ACTG	Unique Tag used to identify beginning and ending of each accounting line within a check writer file.	R
R_AP_CW_PYMT	Unique Tag used to identify beginning and ending of each payment line within a check writer file.	R
R_AP_CW_ADNM	Unique Tag used to identify beginning and ending of each ACH Addendum line within a check writer file.	O

1.4 eMARS General XML Specifications

In general, an XML file is a series of beginning and ending tags. The examples below are meant to provide a general framework to illustrate the usage of XML tags. Furthermore, they will aid in understanding and constructing the eMARS Check Writer XML files. **Appendix C** contains examples of fully qualified eMARS Check Writer XML files.

The CDATA tag is an essential element in the creation of the XML document. The CDATA tag is embedded within the eMARS XML beginning and ending tags. CDATA tags hold the information that departments will submit into eMARS.

For example purposes only, let's assume a file will be submitted to eMARS which includes a first name and last name. Based on the discussion above, that XML file would look like the figure below. Subsequent to each XML tag, there is 'Attribute = "Y"' which declares the information that will be provided relating to that XML tag.

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<AMS_DATAOBJECT_XML_EXPORT_FILE>
  <R_AP_CW_HDR AMSDataObject="Y">
    <FIRST_NAME Attribute="Y"> <![CDATA[JOHN]]> </FIRST_NAME>
    <LAST_NAME Attribute="Y"> <![CDATA[DOE]]> </LAST_NAME>
  </R_AP_CW_HDR >
</AMS_DATAOBJECT_XML_EXPORT_FILE>
```

The following figure is meant to show the basic XML tag construction with emphasis on the CDATA tag for a CW file with a header, one accounting line, one payment line and one ACH Addendum line.

<pre> <?xml version="1.0" encoding="ISO-8859-1" ?> <AMS_DATAOBJECT_XML_EXPORT_FILE> <R_AP_CW_HDR AMSDataObject="Y"> <CW_FILE_ID Attribute="Y"> <![CDATA[123456]]> </CW_FILE_ID> </R_AP_CW_HDR> <R_AP_CW_ACTG AMSDataObject="Y"> </R_AP_CW_ACTG> <R_AP_CW_PYMT AMSDataObject="Y"> </R_AP_CW_PYMT> <R_AP_CW_ADNM AMSDataObject="Y"> </R_AP_CW_ADNM> </AMS_DATAOBJECT_XML_EXPORT_FILE> </pre>	<pre> CW File Beginning Tag CW Header Beginning Tag CW File ID Beginning Tag Data Tag (notice the required CDATA tag) CW File ID Ending Tag CW Header Ending Tag CW Accounting Beginning Tag CW Accounting Ending Tag CW Payment Beginning Tag CW Payment Ending Tag CW ACH Addendum Beginning Tag CW ACH Addendum Ending Tag CW File Ending Tag </pre>
--	---

The following are some technical notes with regard to the construction of XML files:

- There should not be any spaces after "<" or before ">".
- The CDATA must be preceded with "![[".
- The succeeding characters for the actual data in the CDATA section should be "]]".
- The special character "<" must be followed by "/" to indicate the end of a tag.
- If using a Document Type Definitions (DTD) to verify the CW XML file, then the attributes must be listed in the order shown within each data object of the CW template. Otherwise, the attributes can be listed in any order within a data object.
- Specified double quotes (") before and after provided values under the **Comments** column in the template should not be included in corresponding CDATA tags.
- The values "true" and "false" for attributes that can be set to only either value are not case sensitive (i.e. both lower and upper cases are allowed).
- Attributes that are defined as decimal types must include values for all places following the decimal point if the value is not zero. For example, a quantity of 50 with a max size of 15,5 should be represented as 50.00000 within the CDATA tags (i.e. CDATA[50.00000]).
- Data values contained in the CDATA tags should be trimmed on both sides (i.e. left and right) so that the CDATA tags include only the distinct values. Padding fields on the left or right to fill in the remainder of the field size is not necessary as the Check Writer Load process is tag oriented rather than fixed field length oriented. However, fields should not exceed the maximum size specified for each attribute in the templates. To further elaborate, let's assume the following:

- o DOC_NM is the XML tag for the Document Name that is defined as VARCHAR2 (40) and has the value "TEST DOCUMENT NAME".
- o PYMT_LN_NO is the XML tag for the Payment Line Number that is defined as NUMBER (5) and has the value "12".
- o DOC_AM is the XML tag for the Document Amount that is defined as DECIMAL (8,2) and has the value "150.91".

The CDATA tags for the above attributes will look like the following:

```
<DOC_NM Attribute="Y"><![CDATA[TEST DOCUMENT NAME]]></DOC_NM>
<PYMT_LN_NO Attribute="Y"><![CDATA[12]]></PYMT_LN_NO>
<DOC_AM Attribute="Y"><![CDATA[150.91]]></DOC_AM>
```

- CDATA tags must not be submitted with ' ' (blank) values (i.e. CDATA[]).
- For attributes that are listed as being **Optional (O)**, there are two options for submitting (' ') BLANK information:
 - 1) Do not provide the tag at all, or
 - 2) See the table below for data type specifics:

Data Type	Blank Value
Varchar2	CDATA[null]
Number	CDATA[0]
Decimal	CDATA[0]
Date	CDATA[null]

Departments can choose either option to handle "blank" fields. However, it is recommended to not provide their tags since this will result in a smaller interface file.

- The header, accounting, payment and remittance advice records within a CW XML file must be constructed in the following order:
 - Header record (R_AP_CW_HDR)
 - Accounting Lines (R_AP_CW_ACTG)
 - Payment Lines (R_AP_CW_PYMT)
 - Remittance Advice (R_AP_CW_ADNM)

1.5 eMARS Accounting Concept

In Advantage 3, there is a field called Event Type which is entered on the accounting lines of documents to identify the type of financial activity for each accounting line. The primary purposes of the Event Type are the following:

- Ties directly to a set of posting codes that define the type of debits and credits updated to journals (e.g. Accounting Journal, Cash Journal, 1099 Journal ... etc).
- Determines rules for data entry concerning referenced documents, vendor/customer codes, and all defined chart of account elements.
- Drives certain table updates.

Additionally, users can create accounting templates which provide the ability to save frequently used data for use in subsequent document creation. Templates of chart of account elements are created with a unique name for identification called Accounting Template ID.

The Accounting Template ID's can be defined on the Accounting Template table to default fund (e.g. fund, department, unit, object, revenue source ... etc) and detail accounting information (e.g. function, activity, task, program ... etc) that can be entered on the accounting lines of documents.

Users can specify appropriate Accounting Template ID's on the accounting lines of documents and associated fund and detail accounting information will be populated from the Accounting Template table upon processing the documents. Also, users can specify several fund and/or detail accounting attributes along with the Accounting Template ID's on the accounting lines of documents. Those provided accounting attributes will not be overlaid with values from the Accounting Template upon processing the documents. Basically, the Accounting Template only defaults accounting attributes with blank values.

Generally, agencies that will be interfacing Check Writer files with eMARS should provide Accounting Templates ID's and Event Types and optionally applicable funding and/or detail accounting attributes.

For a full explanation of the eMARS Organizational, Central and Departmental Data Elements and a crosswalk of the Organizational, Central and Departmental Data Elements from MARS to eMARS, ***please refer to the eMARS Chart of Accounts Plan.***

2 Overview of Check Writer Approach

2.1 Background Information

Check writer files contain payment information submitted by agencies to the eMARS application in order to process payments. They originate from different agency systems and generate 90% of the state checks. A single check writer file may contain payments to thousands of vendors and a few (usually less than 10) accounting lines. All the payment requests in a single file must use the same accounting attributes and lines. A complete listing of identified Commonwealth check writer systems is found in **Appendix B**.

The requirements for processing check writer files through eMARS include:

- Payment validation to ensure that the required fields are provided.
- Accounting line validation to ensure that the accounting templates and attributes (if provided) are valid.
- A budgetary information check for the valid budget line and to ensure that sufficient funds are available to make the payments.
- Capture of appropriate information for check reconciliation and escheat.
- Capture of appropriate information for 1099 reporting.
- Handling of bank reconciliation for the check writer file in the same manner as Advantage disbursement checks
- Data location in the same table for all the checks generated by the Automated Disbursement and Check Writer processes.
- Provision of a unique identifier for each check writer file which will be based on the CW File ID Number and CW Department Code.
- An ability to print check writer payments on different check stocks, allowing each check stock to have its own check series.

2.2 Keeping Current with Changes

It must be recognized that the Check Writer process is currently under development in eMARS; therefore, changes may occur until this new process is tested successfully. Although the information that is provided in this document is current as of the date of publication, some information may change before the final check writer process is finally implemented. Agency Implementation Leads (AILs) will be informed of any updates.

2.3 Interface Testing

Each Agency is responsible for unit testing of its check writer input file. Due to the great risk an agency faces if a new check writer file is not completed, it is imperative that these files be validated and stabilized with eMARS in a timely fashion. During

the Integrated System Test, the Interface team will require the unit tested agency input files.

2.4 Assumptions

- Agencies will submit all check writer files in a standard format. The format of this file is specified in **Appendix A**.
- For check writer payments that do not have check numbers supplied by agencies, eMARS will assign check numbers. eMARS will also assign a tracking number for each ACH payment. ACH (Automated Clearing House) payments are also called EFT (Electronic Funds Transfer) and "Direct Deposit."
- Each check writer file will be paid from one bank account code.
- Some check writer files will be processed during the day while others will be processed during the nightly cycle.
- Agencies will not have the capability to flag their check writer files to bypass insufficient funds/cash or bypass vendor intercept.
- Check writer files will no longer have the concept of type "1" (i.e. disbursing checks/EFT's during the day) vs. type "2" (i.e. disbursing checks/EFT's during the night).
- Non-1099 and 1099 reportable payments must be submitted by agencies via separate check writer files.
- Agencies that can not have their check writer interfaces files designed, developed, unit tested and accepted within the given timeframe will use manual entry until the new interface is established and tested.
- Only one check page will be printed for each check payment or EFT payment with remittance advice.
- Check writer payments will not reference or liquidate any encumbrance transactions.
- Each bank account code will have its own check series.
- Payments within a check writer file will not be grouped with other payments, i.e. each payment will be considered as "single check".
- Payments will not be checked against any of the following adjustments that may impact the disbursed amount: discounts, taxes, penalties, interests, backup withholdings, and credit memos.
- Accounting lines will be prorated for each payment on the check writer file. For example, a check writer file has 2 accounting lines (\$16 and \$64) and 5 payments (\$10, \$20, \$30, \$10, \$10, totaling \$80). The accounting line amount is proportional for each payment based on the percentage of the accounting line amount. For the first payment, \$2 ($\$10 \times 16/80$) is paid out of accounting line 1 and \$8 ($\$10 \times 64/80$) is paid out of accounting line 2.
- Agencies can submit prenote EFT payments and \$0 check payments in the same check writer file.
- Agencies can include payments for miscellaneous and non-miscellaneous vendors on the same check writer file.

- Agencies must not submit check/EFT payments with negatives amounts in check writer files.
- Agencies must provide the Taxpayer Identification Number (TIN) and TIN Type for miscellaneous vendors if the accounting lines include a 1099 reportable Object code.
- Agencies must not include check payments with both pre-assigned check numbers and blank check numbers within a single check writer file.
- Vendor name and address information are limited to two name lines (Legal Name and Alias Name) and two street address lines.
- Agencies can choose to submit check payments with pre-assigned check numbers. The application will include a designated page to allow agencies to reserve the required numbers of checks for a check writer file.
- For check payments that do not have check numbers pre-assigned by submitting departments, the Advantage 3 application will assign check numbers. Also, the Advantage 3 application will assign ACH tracking numbers for EFT payments, including prenotes.
- Agencies must provide Accounting Template ID and Department Code on each Accounting line. If needed, they can provide additional funding attributes or overwrite some funding attributes that are pre-defined on the Accounting Template.
- Each agency submitting EFT payments to both miscellaneous and ineligible vendors for EFT will be responsible for maintaining and providing accurate bank account information.
- Payments within each submitted check writer file will be subject for intercept except those files exempted by the Office of Controller.
- Agencies can submit check and EFT payments in the same check writer file.
- Each non-miscellaneous vendor in eMARS will be established with a Taxpayer Identification Number (TIN)/TIN Type.
- For 1099 reporting purposes, there will be a miscellaneous vendor established for each type of applicable Organization Class (**ZZMISCINDV** for Individual, **ZZMISCCORP** for Incorporated, and **ZZMISCPART** for Partnership). Agencies submitting check writer files to miscellaneous vendors must select the appropriate vendor code based on organization class of their payee.

3 Check Writer Interface Process

The new eMARS check writer process will be developed to handle and process all Commonwealth's check writer files. The "Run Number" will no longer be generated; it will be replaced by the Check Writer File ID. Agencies must provide the CW Department Code, CW Unit Code, CW File ID within each check writer file. The CW File ID and the CW Department Code must be unique within the eMARS application.

Agencies will establish appropriate Accounting Templates on the Advantage 3 Accounting Template table in order to be provided along with the Department Codes on the CW Accounting component.

This chapter provides information needed to understand and develop a check writer interface to the eMARS Advantage 3 application. A detailed description follows.

3.1 Check Writer Tables

The eMARS Check Writer will include the tables described in this section.

3.1.1 Check Writer Option Table

This table will be keyed by the Fiscal Year. It will be used by eMARS System Administrator to manually establish system-wide and default options that are specific for the Commonwealth of Kentucky.

3.1.2 Check Writer Pre-Assigned Check Numbers Table

This table will be keyed by the CW Department Code and CW File ID. It will be used by authorized users within submitting departments to reserve a specific number of checks from applicable bank codes before generating check writer files with pre-assigned check numbers. The Starting and Ending Check Numbers will be determined based on the Next Check Number of the Bank Code on the BANK table and number of requested checks.

The HTML page of this table will be secured by the CW Department Code and CW Unit Code.

3.1.3 Check Writer Header Table

This is a reference table and will be updated by the CW Header component upon loading a check writer file. Also, it will be updated by check writer batch processes or manually by authorized users. Only one record will be inserted to this table for each uploaded check writer file.

Authorized users within agencies will use this table to certify their check writer files that pass pre-edit validations.

Authorized users within the Office of the Controller will use this table to certify check writer files that have been certified by their submitting agencies. Also, they can use this table to modify unprotected fields on this table, if needed, or put a check writer file on hold.

This table will include information to determine which disbursement format, category and handling will be used for a check writer file as well as the bank account that funds will be taken from. Various control fields will be present to validate the number of both accounting and payment lines within a check writer file and provide a control total payments amount. An override level will be used to override errors with overrideable messages.

Also, this table will include separate logging fields to record the User ID's and Processing Date/Time Stamp for departmental and central office certifications as well as each submitted Check Writer batch process against every check writer file.

The HTML page of this table will be secured by the CW Department Code and CW Unit Code.

3.1.4 Check Writer Accounting Table

This table is an inquiry table and will be updated by the CW Accounting component upon loading a check writer file. One or more accounting lines will be inserted to this table for each uploaded check writer file.

This table will contain the Accounting Template ID and funding attributes. Agencies must provide the Accounting Template ID and Department Code on each provided accounting line on the CW Accounting Component. Upon processing a check writer file, funding attributes with blank values will be inferred from associated Accounting that is setup on the Advantage 3 Accounting Template table.

The HTML page of this table will be secured by the CW Department Code and CW Unit Code.

3.1.5 Check Writer Payment Table

This table is an inquiry table will be updated by the CW Payment component upon loading a check writer file. One or more payment lines will be inserted to this table for each uploaded check writer file.

This table will contain payment types (i.e. check or EFT), payee related information and ACH payment related information for EFT's as well as optionally Remittance Advice information and pre-assigned check numbers for check payments. The Remittance Advice for EFT and check payments will be one long field with 1500 characters.

The HTML page of this table will be secured by the CW Department Code and CW Unit Code.

3.1.6 Check Writer ACH Addendum Table

This table is an inquiry table and will be updated by the CW ACH Addendum component, if provided, upon loading a check writer file. Each check writer EFT payment (excluding prenotes) can be associated with only one ACH Addendum record on this table.

This table will contain ACH payment related information that is posted on the Addenda Record (record type "7") when generating the ACH file.

The HTML page of this table will be secured by the CW Department Code and CW Unit Code.

3.1.7 Check Writer Vendor Intercept Table

This table is an inquiry table and will be updated when check writer payments are intercepted during processing. The number of inserted lines on this table for an intercepted payment will be based on the number of accounting lines provided on associated check writer files times the number of debts intercepted on the Advantage 3 Intercept Request table.

It will be used to provide on-line inquiry for payment intercept information for CW payments that have been intercepted. Information will include all details related to an individual claim against a payee such as claim reference information, claim control information, and claim recipient information.

The HTML page of this table will be secured by the CW Department Code and CW Unit Code.

3.2 Check Writer Batch Processes

The eMARS Check Writer will include several batch processes including the processes described in this section that will be submitted on-line by agency system administrators using the Job Manager of the Advantage 3 application.

3.2.1 Check Writer Load Process

The CW Load Process will perform a parsing of the Check Writer XML files received from agencies and load the parsed data into the respective Check Writer tables (CW Header, Accounting, Payment and ACH Addendum tables). Only data type validation will be performed by this process.

3.2.2 Check Writer Pre-Edit Process

This process will perform field inferences. It will infer values to the CW Header, Accounting, and Payment tables from the various related tables. These inferences are required to ensure that all the necessary fields are populated before any validations occur against the CW tables. The inferences will include populating

vendor name, address information and Taxpayer Identification Number (TIN)/TIN Type for non-miscellaneous vendors from the Advantage 3 Vendor Customer table.

Additionally, this process will perform validation for all information provided in the check writer files. These validations will include:

- Payment Validation - Each payment will be validated to ensure that all required fields are provided and the data is valid.
- Accounting Validation - Each accounting line will be validated to ensure that the accounting templates and funding attributes are valid within the Advantage 3 application.
- Budget/Allotment/Cash Validation - Budgetary information will be checked for the presence of budget lines and to ensure that sufficient funds are available to make the payments.
- 1099 Reportable Validation - Object codes will be checked for 1099 reportability.

This process will produce two reports (one for accounting validations and another for non-accounting validations) with all identified warning, overrideable and hard error messages. Only one check writer file can be edited per run.

Once a check writer file passes all edits, the submitting agency and the Office of the Controller must certify the check writer file on-line using the Advantage 3 Check Writer Header page for further processing (*refer to the section "Certifying Check Writer Files"*).

If a check writer file has errors and fixing those are not possible by other means (e.g. changing a value of an unprotected field on the CW Header page by the Office of the Controller), the submitting agency will run the CW Clean-up batch process to delete all associated records from CW tables (*refer to the section "Check Writer Clean-up Process"*).

3.2.3 Check Writer Clean-up Process

The CW Clean-Up process will allow agencies to delete all information loaded into the Check Writer tables from specific check writer input file(s) that failed the CW Pre-Edit process or the CW Load process. Upon the execution of this process, all associated records of the specified check writer file name(s) will be deleted from the Advantage 3 CW Header, Accounting, Payment and ACH Addendum (if applicable) tables. By deleting the failed file(s) information, the agency can correct the information, resend and re-process the corrected CW input file.

3.3 Structure of the Agency Check Writer File

An agency check writer input file is in an XML-based format comprised of multiple components: Header, Accounting, Payment and ACH Addendum (optional). Refer to **Appendix A** for the detailed input file layout. Check writer files must have a unique combination of CW File ID and CW Department Code, and must be provided with a CW Unit Code.

3.3.1 Check Writer Header Component (R_AP_CW_HDR)

Each check writer file must contain only one header record. The header record will provide information to determine the check stock/ACH format used for the file and the bank account that funds will be taken from. Various control fields are present to validate the check file including a control total check. Supported ACH formats in eMARS are CCD and PPD.

3.3.2 Check Writer Accounting Component (R_AP_CW_ACTG)

If a check writer file consists only of ACH pre-note transactions, each with a zero balance, an accounting component is not required. Otherwise, each check writer file must contain one or more accounting lines. The accounting line uses the Accounting Template and Chart of Account information provided to indicate the various accounting attributes that should be charged for this check file. The accounting attributes are comprised of both Statewide and Agency Chart of Accounts elements. If a check writer file includes more than one accounting line, the Accounting Line Number must be unique.

3.3.3 Check Writer Payment Component (R_AP_CW_PYMT)

Each check writer file must contain one or more payment lines. The Payment line will provide the detailed information needed to produce a check. Payment information includes payee name and address, and payment amount. If a check writer file includes more than one payment line, then the Payment Line Number must be unique.

Each payment line will include an optional single long field (1500 characters) to record remittance advice information (e.g. purchase order number, invoice number, invoice date) for check and EFT payments. The Remittance Advice field is defined on an agency basis to provide payment stub information. In MARS, the Remittance Advice Information field allows 150 bytes, but, by using it in conjunction with the Sequence Number field, up to 10 "lines" of the 150 byte fields can be used to provide remittance information.

Agencies that currently provide remittance advice information in MARS and plan to continue doing so in eMARS may decide to concatenate the 10 remittance advice lines into one field of 1500 bytes. Agencies that currently do not provide any remittance advice information in MARS but plan to do so in eMARS will need to contact the eMARS Interface Team and Treasury for guidance on formatting the remittance advice.

3.3.4 Check Writer ACH Addendum Component (R_AP_CW_ADNM)

The ACH Addendum record is optional and will contain information to be recorded on the Record Type "7" of the ACH file for EFT payments. An EFT payment (excluding prenotes) can be associated with only one ACH Addendum record.

3.3.5 Check Writer in MARS vs. eMARS

The following table provides a crosswalk for the check writer records between MARS and eMARS:

MARS		eMARS	
CW Record	Comments	CW Component	Comments
CW Header Record (Type 10)	Only one record required per file	CW Header Component	Only one component required per file; information will be displayed on the CW Header page for manual certification (departmental and central levels)
CW Accounting Record (Common Coding Strings, Type 21) CW Accounting Record (Funding Attributes, Type 22)	At least one Accounting Record (either 21 or 22) required per file	CW Accounting Component	At least one Accounting Component required per file; information will be displayed on the inquiry CW Accounting page
CW Payment Information Record (Type 30)	At least one Payment Record required per file	CW Payment Component	At least one Payment Component required per file; information will be displayed on the inquiry CW Payment page
CW Payment Remittance Advice Record (Type 31)	Optional for Check and EFT payments	CW Payment Component	The CW Payment Component will include an optional large field (1500 characters) to store the Remittance Advice information; information will be displayed on the inquiry CW Payment page
CW Payment ACH Information Record (Type 32)	Required for each EFT payment	CW Payment Component	The CW Payment will include ACH fields to store the ACH information (e.g. ABA Number, Bank Account Number) provided on the MARS Record Type "32"; only non-confidential information will be displayed on the inquiry CW Payment page
CW Payment ACH Addendum Record (Type 33)	Optional for EFT payments	CW ACH Addendum Component	Optional for EFT Payments; information will be displayed on the inquiry CW ACH Addendum page

3.4 Check Writer File ID

The following information will be used to uniquely identify each loaded check writer file in the Advantage 3 application:

- CW_DEPT_CD: CW Department Code - varchar(4)
- CW_FILE_ID: CW File ID - varchar(20)

Additionally, for security purposes, it is necessary that each department provides a CW_UNIT_CD (CW Unit Code) – varchar(4). Note that while the database table column is defined as varchar(20), the Commonwealth standard is that a thirteen character file id is to be provided.

In eMARS, check writer interface files have been structured like documents with each check writer file uniquely identified by a CW Department Code and CW File ID. The CW File ID will have a length of 13 characters and is to be created as follows:

Interface Indicator: W indicates that the ID is a check writer.

System Mnemonic: Four character description of a system from which the data is coming. It will be based on the Rover ID (i.e. Interface File ID) – *positions 5 through 8* - that is assigned to each check writer file

2-Digit Year: The last 2 digits of the year of the transaction date.

3-Digit Julian Day: Number of a day of the year of the transaction date

3-Digit Sequence: Sequentially assigned numbers for check writer files created that day.

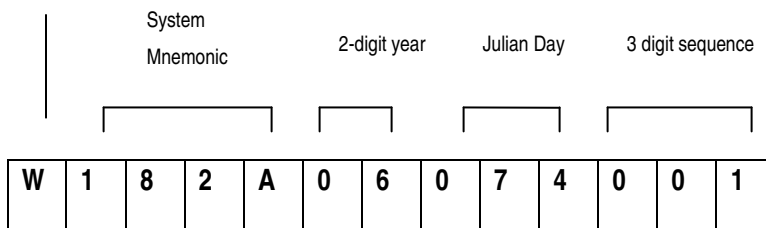
Following is the structure of the **CW File ID**, where the Rover Id is C35**182A**

2-DIGIT Current year is **06**

3-Digit Julian date is **074**

3-digit Sequence number is **001**

Interface Indicator



3.5 Loading Check Writer Files to Advantage 3 Application

Agencies generate check writer files in XML format based on the specification described in **Appendix A** on different platforms (e.g. mainframe, UNIX) according to the predetermined frequency and schedule. If the check writer file is generated on a platform other than an NT server, the agency will transfer the file to a designated folder on a NT server or workstation within their operational environment. Agencies will name their check writer files as described in **Appendix B**.

Authorized users within agencies will logon to the Advantage 3 application and open a designated page (the name of this page will be defined during the design stage) that is used to upload check writer files. On that page, the user will:

- Enter the CW Department Code, CW Unit Code and *CW File Name*,
- Click the "browse" button to navigate through the files on the user's computer or authorized network servers in order to select the appropriate interface file, then
- Click the "upload" button to load selected check writer file.

Departments will continue to name their files as they are currently being named, i.e. based on the Rover ID (e.g. C31R2130). The check writer interface file names are listed in **Appendix B** under the "**Rover ID**" column.

If the specified File Name was authorized for the entered Department and Unit Codes, and the user's security profile was authorized for the entered Department and Unit Codes, the selected check writer file will be loaded to a designated directory on the Advantage 3 Application Server and ready for processing. At the same time, a backup file of the uploaded check writer will be generated on another directory on the Advantage 3 Application Server. The name of the uploaded check writer file will be the selected "File Name" on that page with the extension "xml" (for example, C50R030A.xml). Whereas the name of the backup file will be composed of the File Name and the Server Date/Time Stamp with the extension "xml" (for example, C50R030A_08-01-2005_11:05:20AM.xml).

Once a check writer file is uploaded into the Advantage 3 Application Server, an authorized user within the respective agency will submit the CW Load batch process using the Advantage 3 Job Manager (it is a component within the Advantage 3 application that is used to submit jobs on-line). The CW Load batch process will import the Header, Accounting, Payment and ACH Addendum (if applicable) components of CW file into the Advantage 3 CW Header, Accounting Line, Payment Line and ACH Addendum tables, respectively.

3.6 Certifying Check Writer Files

The check writer tables will have HTML pages within the Advantage 3 application and can be accessed on-line by authorized users. These pages will be secured by the Department and Unit Codes. The CW Accounting, Payment and ACH Addendum pages can be used for inquiry purposes only. The CW Header page will contain two level of certification (department level and central office). Agencies will be granted authorization to certify/un-certify their check writer files. Whereas the Office of the Controller will be granted additional authorization privileges on the CW Header page to change unprotected fields as well as certify/un-certify check writer files.

3.7 Processing Check Writer Files

Due to Commonwealth payment requirements, some payments must be processed during the day (we refer to those in MARS as type "1" processing). Most payments however, will be processed during the nightly batch process (we refer to those in MARS as type "2" processing).

Each paying agency will be responsible for performing the following processes in a manual fashion to process its check writer files:

- Upload its files in the Advantage 3 Application Server through a secured/designated HTTPS based page after logging into the Advantage 3 application.
- Execute the CW Load batch process through the Advantage 3 application to import its uploaded files from the Advantage 3 Application Server to the Check Writer tables (CW Header, Accounting, Payment, and ACH Addendum tables).
- Execute the CW Pre-Edit batch process against its loaded check writer files to validate provided information. This process will produce two reports (accounting and non-accounting reports) for each processed file where. Each produced report will include generated messages along with appropriate message types (i.e. warning, error or overrideable).
- Certify check writer files that pass the validations (or pass appropriate overrides) by checking the Department Certification flag on the CW Header page through the Advantage 3 application.

Periodically during the day, the Office of the Controller will use the Check Writer Header page to search for certified files by submitting agencies and are pending the central office certification. The Office of the Controller will manually flag applicable check writer files on the CW Header table to bypass Vendor Intercept process, whether those files are processed during the day or scheduled for nightly processing. Also, the Office of the Controller may manually change some unprotected fields on the CW Header page such as the Disbursement Category before certifying a CW file.

After completing necessary changes to the CW Header table, the Office of the Controller will run the CW Pre-Edit process to ensure that changes will not be rejected during further processing. Based on overrideable error messages issued during the Pre-Edit process, the Office of the Controller will determine whether to continue processing the file or reject it. To continue processing the file, the Office of

the Controller will certify the file that passes the validations (or pass appropriate overrides) by checking the Central Office Certification flag on the CW Header. If the Office of the Controller decided to reject the file due to issued overrideable errors, for example related to insufficient cash and/or budget, they will not certify the file and will notify the submitting agency through email to fix those issued errors.

For completely certified CW files, additional check writer batch processes will either be manually submitted by the Office of the Controller to process the file during the day, or executed during the eMARS Nightly Cycle.

A single Check Writer Accounting (CWA) document will be generated for each processed CW file to record the financial events based on the accounting lines provided on the CW Accounting component. The CWA document will be posted and submitted into the Advantage 3 Document Catalog table (which is similar to the SUSF table, in Advantage 2). The CWA document will consist of a document header and one or more accounting lines, and will be named based on provided CW Department Code, CW Unit Code, and CW File ID.

Additionally, a Check Writer Intercept (CWI) document will be created for each intercepted payment due to an outstanding debt whether its check writer file was processed during the day or during the nightly cycle.

If the payment is reduced due to intercept, then a notation will appear on the stub information accompanying the check. If the payment was originally submitted as an EFT, then the EFT payment will be disbursed as a check with the payment portion remaining after offset. If there is no balance remaining, a check for \$0.00 will be printed.

Check numbers will be assigned (if not already provided) for check payments and tracking numbers will be assigned for EFT payments, including prenotes, based on provided Bank Account Code (refer to the section "*Check Stock and Numbering*").

All check and EFT payment (excluding prenotes) for processed check writer files will be posted to the Check Reconciliation table where the Document Number is set as follows:

- Document Code: "CW" for check payments or "CE" for EFT payments
- Document Department Code: CW Department Code
- Document ID: CW File ID

Payment information will be posted to the 1099 Journal and the 1099 Reporting Information will be updated (whenever applicable) if a 1099 reportable Object code is provided on a check writer accounting line.

All updates to applicable tables and journals associated with processing check writer files during the day will occur real-time.

Generated payment files (Check, Remittance Advice and ACH) for check writer files completely processed during the day will be sent to the Treasury for printing during the day. Whereas, generated payment files for check writer files processed during the nightly cycle will be sent to the Treasury for printing during next business day.

A Check Writer Register report will be produced for each processed CW file during the day or nightly cycle which lists disbursed check and EFT payments.

3.8 Check Stock and Numbering

In eMARS, the Check Format and Disbursement Category are optional on the CW Header component of the check writer input file. The default value would be inferred from the CW Options table upon processing the CW Pre-Edit batch process if either code was left blank.

The Check Format will be used to specify the type of the check stock (e.g. payroll, z-fold ... etc). The Disbursement Category will be used to specify whether the check is Treasury mailed, Treasury hold, or agency mailed.

The Next Check Number on the Bank table will be used to assign Check Numbers for check payments that do not have a check number provided by agencies. If check numbers were provided by the agency on the input check writer file, they will be validated against the CW Pre-Assigned Check Numbers table.

Additionally, the Next Available EFT Number on the Bank table will be used to assign ACH Tracking Numbers for EFT payments. Assigned ACH Tracking Numbers will be populated on the Identification Number of the ACH Record Type "6" (i.e. payment record). Also, the agency-supplied 16-character Receiving Names will be populated on the ACH Receiving Name of the ACH Record Type "6".

3.9 1099 Posting

For 1099 reporting purposes, there will be a miscellaneous vendor established for each type for applicable Organization Class (**ZZMISCINDV** for Individual, **ZZMISCCORP** for Incorporated, and **ZZMISCPART** for Partnership) that agencies can use for payment processing. Also, every non-miscellaneous vendor will be associated with a TIN/TIN Type (Social Security Number (SSN) or Employer Identification Number (EIN)). Agencies that submit check writer files with 1099 Reportable Object codes and miscellaneous vendors must provide appropriate Miscellaneous Vendor Customer Code (i.e. Individual, Incorporated or Partnership) along with the TIN, TIN Type, payee name and address to ensure 1099 reporting accuracy.

Advantage 3 has a 1099 table called "1099 Reporting Information (1099I)". This table is used to define valid TIN and TIN Type combinations and tax related information for the 1099 Reporting process. Upon processing check writer files with miscellaneous vendors and 1099 reportable object codes, payee name and address information will either be inserted to the 1099I table if a record for the TIN/TIN Type does not exist or replaced on this table if a corresponding record is found. In this way, the 1099I table will always have the most recent address information for miscellaneous vendors.

The 1099I table includes two Name lines and only one Address line. The name and address lines on the 1099I table for miscellaneous vendors will be updated as follows:

- Name on 1099I will store the Vendor Legal Name on CW Payment Line
- Name(s) on 1099I table will store the Vendor Alias Name on CW Payment Line.

- Street Address on 10991I table will store the second Vendor Address Line on CW Payment Line if it is not blank, otherwise, it will store the first Vendor Address Line on CW Payment Line.

Appendix A - Advantage 3 Check Writer XML Input File Layout

Header Component

XML Tag	Caption on CW Header Table	Description	R/C	Type	Max Size	Comments
CW_DEPT_CD	Department	The Department Code of the department submitting the CW file.	R	Varchar2	4	Must be valid on the Department Fiscal Year Controls table. This Code along with the Check Writer File ID must be unique in eMARS.
CW_FILE_ID	Check Writer File ID	Check Writer File ID assigned by the submitting departments.	R	Varchar2	20	This ID along with the Check Writer Department Code must be unique in eMARS.
CW_UNIT_CD	Unit	The Unit Code of the department submitting the CW file.	R	Varchar2	4	Must be valid on the Unit table.
BANK_ACCT_CD	Bank Account Code	Used to assign check numbers for check payments, if check numbers are not already pre-assigned and tracking numbers for EFT payments.	R	Varchar2	4	Must be a valid on the Bank table. The MARS bank account code will continue in eMARS.
PYMT_DT	Payment Date	The date to be printed on each check, or used to determine the Settlement Date for EFT payments.	R	Date	10	Must be a valid date with the format "yyyy-mm-dd". The difference between the provided Payment Date and the processing date must not exceed 45 calendar days.

XML Tag	Caption on CW Header Table	Description	R/C	Type	Max Size	Comments
PYMT_DSCR	Payment Description	User provided information that describes the check writer file.	O	Varchar2	60	
CNTAC_CD	Contact	The contact code of the department that its address will appear as the "Return Address" on the check.	R	Number	19	Must be valid on the Contact table. Valid values are not yet determined and will be provided at a later date.
BFY	Budget Fiscal Year	The budget fiscal year associated with the check writer file.	CR	Number	4	Leave blank unless it is July and the document needs to be posted in the prior year. If left blank, the system will populate it with the Fiscal Year of the CWA document Record Date on associated document. If posting to a prior fiscal year, provide the budget fiscal year. If provided, must be valid on the Fiscal Year table and follow the format "YYYY".
FY	Fiscal Year	The fiscal year associated with the check writer file.	CR	Number	4	Leave blank unless it is July and the document needs to be posted in the prior year. If left blank, the system will populate it with the Fiscal Year of the CWA document Record Date on associated document. If posted to the 13th accounting period in the prior fiscal year,

XML Tag	Caption on CW Header Table	Description	R/C	Type	Max Size	Comments
						the Fiscal Year must be entered. If provided, must be valid on the Fiscal Year table and follow the format "YYYY".
PER	Period	The accounting period associated with the check writer file.	CR	Number	2	<p>Leave blank unless it is July and the document needs to be posted in the prior year.</p> <p>If left blank, the system will populate it with the Fiscal Period of the CWA document Record Date on associated document.</p> <p>If posted to the 13th accounting period in the prior fiscal year, the Fiscal Period must be entered.</p> <p>If provided, the valid value is 13.</p>
EFT_FRMT	EFT Format	It indicates the format that will be used to generate the ACH file. It will be used if a CW file included EFT payments.	O	Varchar2	4	<p>Valid values are: CCD and PPD.</p> <p>If left blank, it will be inferred from the Default ACH Format on the Check Writer Options table which will be set to "CCD".</p>
CHK_FRMT	Check Format	It indicates the format of the check stock that will be used for printing checks or remittance advice for EFT payments. It will be used if a CW file included check payments or EFT payments with remittance advice.	O	Varchar2	4	<p>Valid values are stored on the Disbursement Format table where the Disbursement Type is set to "check".</p> <p>If left blank, it will be inferred from the Default Check Format on the Check Writer Options</p>

XML Tag	Caption on CW Header Table	Description	R/C	Type	Max Size	Comments
		Check Format codes will be like the Check Types on the Check Category table in MARS (e.g. Employment Retirement, Payroll, Child Support, Unemployment Insurance and Teacher's Retirement).				table which will be set to "GENZ" (i.e. z-fold checks). Other valid values on the Disbursement Format table are: <ul style="list-style-type: none"> • GENC: Generic C, • KERS: Employee's Retirement, • FINC: Finance, • TRES: Treasury, • PYRL: Payroll, • CSUP: Child Support, • KTRS: Teacher's Retirement, • UINS: Unemployment Insurance.
DISB_CAT	Disbursement Category	Disbursement Category codes will be like the Check/Seal Indicators on the Check Category table in MARS (e.g. Unsealed Treasury Hold, Unsealed Treasury Mailed, Unsealed Agency Mailed, Sealed Treasury Hold, Sealed Treasury Mailed, and Sealed Agency Mailed).	0	Varchar2	4	Valid values are stored on the Disbursement Category table. If left blank, it will be inferred from the Default Disbursement Category on the Check Writer Options table which will be set to "STM" (i.e. sealed/Treasury Mailed). Other valid values on the Disbursement Category table are: <ul style="list-style-type: none"> - UTH (Unsealed Treasury Hold), - UAM (Unsealed Agency Mailed), - STH (Sealed Treasury Hold),

XML Tag	Caption on CW Header Table	Description	R/C	Type	Max Size	Comments
						- SAM (Sealed Agency Mailed)
TOT_PYMT_AM	Total Payment Amount	Total amount of the payment lines in a check writer file.	R	Decimal (14,2)	15	Must match the sum of accounting line amounts as well the sum of payment amounts. It cannot be a negative amount. It can only be zero if the file includes only payments with \$0 amounts. The format is 999999999999.99.
NO_ACTG_LN	Number of Accounting Lines	Total number of the accounting lines in a check writer file.	R	Number	3	Must match number of accounting lines in the file.
NO_PYMT_LN	Number of Payment Lines	Total number of payment lines in a check writer file.	R	Number	6	Must match number of payment lines in the file.
PERMIT_NO	Permit Number	Agencies can provide a Postal Permit Number to mail material without affixing postage.	O	Varchar2	10	

Accounting Component

XML Tag	Caption on CW Accounting Table	Description	R/C	Type	Max Size	eMARS Comments
LN_NO	Line Number	The accounting line number that uniquely identifies each accounting line within a CW file.	R	Number	5	Must be unique within a check writer file.
EVNT_TYP	Event Type	An event type identifies the type of financial activity for an accounting line. The activity may or may not have an accounting impact on the journals.	R	Varchar2	4	Must be valid on the Event Type table. Valid values are: <ul style="list-style-type: none"> • CA01 (when Object is coded) • CA02 (when Revenue Source is coded)
ACTG_TMPL_ID	Accounting Template ID	Accounting templates are used to bring in chart of account element codes for ease of data entry on documents.	R	Varchar2	6	Must be valid on the Accounting Template table.
FUND_CD	Fund	The identification code associated with the fund.	O	Varchar2	4	If entered, must be valid on the Fund table.
SFUND_CD	Sub Fund	The identification code associated with the sub-fund.	O	Varchar2	4	If entered, must be valid on the Sub Fund table. To enter a Sub Fund Code, the Fund Code must be entered.
DEPT_CD	Department	An identification code assigned to a central organizational level element.	R	Varchar2	4	Must be valid on Department Fiscal Year Controls table.

XML Tag	Caption on CW Accounting Table	Description	R/C	Type	Max Size	eMARS Comments
UNIT_CD	Unit	Unit is the lowest organizational level in the main organizational structure. Only sub unit is lower, but it is not necessarily required as the unit is. At the unit level, all the organizational elements it reports to are defined so that they are inferred to accounting documents.	O	Varchar2	4	If entered, must be valid on the Unit table.
SUNIT_CD	Sub Unit	Sub Unit is a means of breaking down a unit code into smaller measurements. It is the lowest level of organizational structure.	O	Varchar2	4	If entered, must be valid on the Sub Unit table. To enter a Sub Unit Code, the Unit Code must be entered.
APPR_CD	Appr Unit	The identification code assigned to a single appropriation unit. An appropriation is a budgeting Chart of Accounts element.	O	Varchar2	9	If entered, must be valid on the Appropriation table.
OBJ_CD	Object	The identification code associated with an object of expenditure. Object is a fund accounting Chart of Accounts element.	O	Varchar2	4	If entered, must be valid on the Object table.
SOBJ_CD	Sub Object	The identification code associated with the sub-object. Sub-object is a fund accounting Chart of Accounts element.	O	Varchar2	4	If entered, must be valid on the Sub Object table. To enter a Sub Object Code, the Object Code must be entered.
RSRC_CD	Revenue Source	Revenue sources are used to track individual sources of revenue taken in.	O	Varchar2	4	If entered, must be valid on the Revenue Source table.

XML Tag	Caption on CW Accounting Table	Description	R/C	Type	Max Size	eMARS Comments
SRSRC_CD	Sub Revenue Source	The identification code associated with the sub-revenue source.	0	Varchar2	4	If entered, must be valid on the Sub Revenue Source table. To enter a Sub Revenue Source Code, the Revenue Source Code must be entered.
DOBJ_CD	Dept Object	The identification code assigned to the Department object.	0	Varchar2	4	If entered, must be valid on the Department Object table.
DRSRC_CD	Dept Revenue Source	The identification code assigned to the Department Revenue Source.	0	Varchar2	4	If entered, must be valid on the Department Revenue Source table.
ACTV_CD	Activity	The identification code associated with the activity. An activity is an internal program that you want to define for budgeting and/or reporting purposes.	0	Varchar2	4	If entered, must be valid on the Activity table.
SACTV_CD	Sub Activity	The identification code associated with the sub-activity.	0	Varchar2	4	If entered, must be valid on the Sub Activity table. To enter a Sub Activity Code, the Activity Code must be entered.
FUNC_CD	Function	The identification code associated with the function. The function code defines broad operational objectives, such as instruction, support services, and facilities acquisition.	0	Varchar2	10	If entered, must be valid on the Function table.
SFUNC_CD	Sub Function	The identification code associated with the sub-function.	0	Varchar2	4	If entered, must be valid on the Sub Function table. To enter a Sub Function Code, the Function Code must be entered.

XML Tag	Caption on CW Accounting Table	Description	R/C	Type	Max Size	eMARS Comments
BSA_CD	BSA	The identification code assigned to the balance sheet account.	O	Varchar2	4	If entered, must be valid on the BSA table.
SBSA_CD	Sub BSA	The identification code assigned to the Sub Balance Sheet Account.	O	Varchar2	4	If entered, must be valid on the Sub BSA table. To enter a Sub BSA Code, the BSA Code must be entered.
RPT_CD	Reporting	The reporting code element is a programmatic element that can be used in many different ways for measurement and tracking. It is even used as an informal cost accounting entity at times. It can be used on all accounting documents and is stored in journals and optionally in ledgers.	O	Varchar2	10	If entered, must be valid on the Reporting Code table.
SRPT_CD	Sub Reporting	Sub reporting codes are used to further define a specific reporting code.	O	Varchar2	4	If entered, must be valid on the Sub Reporting table. To enter a Sub Reporting Code, the Reporting Code must be entered.
LOC_CD	Location	The identification code assigned to the location. Location is a programmatic Chart of Accounts element.	O	Varchar2	4	If entered, must be valid on the Location table.
SLOC_CD	Sub Location	The identification code associated with the sub-location.	O	Varchar2	4	If entered, must be valid on the Sub Location table. To enter a Sub Location Code, the Location Code must be entered.
TASK_CD	Task	The unique identification code assigned to the task.	O	Varchar2	4	If entered, must be valid on the Task table.

XML Tag	Caption on CW Accounting Table	Description	R/C	Type	Max Size	eMARS Comments
STASK_CD	Sub Task	The sub task is a means of breaking down tasks into smaller components.	O	Varchar2	4	If entered must be valid on Sub Task table. To enter a Sub Task Code, the Task Code must be entered.
TASK_ORD_CD	Task Order	The unique identification code assigned to the task order.	O	Varchar2	6	If entered, must be valid on the Task Order table.
PROG_CD	Program	The identification code assigned to the program. Program is a cost accounting Chart of Accounts element.	O	Varchar2	10	If entered, must be valid on the Program Setup table.
PHASE_CD	Phase	The identification code assigned to the phase. Phase is a cost accounting Chart of Accounts element representing a phase of a program.	O	Varchar2	6	If entered, must be valid on the Program Phase table. To enter a Phase Code, the Program Code must be entered.
PPC_CD	Program Period	The identification code assigned to the program period.	O	Varchar2	6	If entered, must be valid on the Program Period table. To enter a Program Period Code, the Program Code must be entered (to infer the Major Program).
LN_AM	Line Amount	Amount charged to the accounting line.	R	Decimal (14,2)	15	It can only be zero or greater, but cannot be negative amount. The format is 999999999999.99

Payment Component

XML Tag	Caption on CW Payment Table	Description	R/C	Type	Max Size	Comments
LN_NO	Line Number	The payment line number that uniquely identifies each payment line within a CW file.	R	Number	5	Must be unique with in a check writer file.
VEND_CUST_CD	Vendor Customer	The unique identifier assigned to the vendor/customer and defined on the Vendor Customer table.	R	Varchar2	20	Must be valid the Vendor/Customer table (VCUST) and can be a miscellaneous vendor. Valid miscellaneous vendor codes are: - ZZMISCINDV for individual - ZZMISCCORP for Incorporated - ZZMISCPART for Partnership
TIN	Taxpayer ID Number	The taxpayer identification number (TIN).	CR	Varchar2	9	Required if the vendor code is miscellaneous on the Vendor Customer Table and at least one accounting line includes a 1099 reportable Object code.
TIN_TYP	TIN Type	The type associated with the taxpayer identification number.	CR	Varchar2	1	Required if the TIN was required. Valid values are: - "1" for EIN, - "2" for SSN/ITIN/ATIN

XML Tag	Caption on CW Payment Table	Description	R/C	Type	Max Size	Comments
AD_ID	Address ID	The Vendor Address ID on the Vendor/Customer table.	CR	Varchar2	20	For miscellaneous vendors, it must be blank. For non-miscellaneous vendor codes, it will be required for vendor codes with <u>no active default</u> Payment Address ID. If entered, must be a valid Vendor Address ID for the Vendor/Customer Code on the VCUST table.
CNTAC_ID	Contact Code	The Vendor Contact ID on the Vendor/Customer table.	O	Varchar2	20	For miscellaneous vendors, it must be blank. For non-miscellaneous vendor codes it is optional. If entered, must be a valid Vendor Contact ID for the Vendor/Customer Code on the VCUST table.
LGL_NM	Legal Name	The legally defined name of the company or individual to be printed on the check.	CR	Varchar2	60	Required for miscellaneous vendors. For non-miscellaneous vendors, it will always be inferred / overwritten from corresponding Legal Name on the Vendor Customer table.
ALIAS_NM	Alias/DBA	The alternate name of the company or individual to be printed on the check.	O	Varchar2	60	For non-miscellaneous vendors, if it exists on the Vendor Customer table, then it will be inferred / overwritten from corresponding Alias Name.

XML Tag	Caption on CW Payment Table	Description	R/C	Type	Max Size	Comments
AD_LN_1	Address Line 1	The first line of the mailing address.	CR	Varchar2	75	Required for miscellaneous vendors. For non-miscellaneous vendors, it will always be inferred / overwritten from corresponding Vendor Address Line 1 on the Vendor Customer table based on the provided Vendor Address ID or the active default Payment Address ID (if the Vendor Address ID is not active for the provided vendor code).
AD_LN_2	Address Line 2	The second line of the mailing address.	O	Varchar2	75	For non-miscellaneous vendors, if it exists on the Vendor Customer table, it will be inferred / overwritten from corresponding Vendor Address Line 2 based on the provided Vendor Address ID or the active default Payment Address ID (if the Vendor Address ID is not active for the provided vendor code).

XML Tag	Caption on CW Payment Table	Description	R/C	Type	Max Size	Comments
CITY	City	The city name associated with the address.	CR	Varchar2	60	Required for miscellaneous vendors. For non-miscellaneous vendors, it will always be inferred / overwritten from corresponding City on the Vendor Customer table based on the provided Vendor Address ID or the active default Payment Address ID (if the Vendor Address ID is not active for the provided vendor code).
ST	State/Province	The state or province associated with the address.	CR	Varchar2	2	Required if miscellaneous vendor and Country Code is either "USA" or Blank. For non-miscellaneous vendors, it will always be inferred / overwritten from corresponding State/Province Code on the provided Vendor Customer table based on the Vendor Address ID or the active default Payment Address ID (if the Vendor Address ID is not active for the provided vendor code).

XML Tag	Caption on CW Payment Table	Description	R/C	Type	Max Size	Comments
ZIP	Zip	The zip code or postal code associated with the address.	CR	Varchar2	10	<p>Required if miscellaneous vendor and Country Code is either "USA" or blank.</p> <p>For non-miscellaneous vendors, it will always be inferred / overwritten from corresponding Zip Code on the provided Vendor Customer table based on the Vendor Address ID or the active default Payment Address ID (if the Vendor Address ID is not active for the provided vendor code).</p> <p>For US based addresses, the format is either 99999 or 99999-9999.</p>
CTRY	Country	The unique identification code associated with the country.	O	Varchar2	3	<p>For non-miscellaneous vendors, it will always be inferred / overwritten from corresponding Country Code on the provided Vendor Customer table based on the Vendor Address ID or the active default Payment Address ID (if the Vendor Address ID is not active for the provided vendor code).</p> <p>For miscellaneous vendors, if not entered, it will be defaulted to "USA". Otherwise, it must be valid on the Country table.</p>

XML Tag	Caption on CW Payment Table	Description	R/C	Type	Max Size	Comments
DLVR_PT	Delivery Point	Agencies may provide a delivery point for Post Office bar code sort.	O	Varchar2	10	
PYMT_AM	Payment Amount	Payment amount due to vendor.	R	Decimal (14,2)	15	It can be zero or greater, but cannot be negative amount. For EFT payments, the ACH Transaction Code must be set as a prenote if the payment amount is \$0. For EFT payments, this amount can not be greater than \$99999999.99.
CMNT	Comments	User provided data associated with a payment which will appear on the check face.	O	Varchar2	60	
PYMT_CD	Payment Code	It indicates if the payment to be disbursed as a check or an EFT.	R	Varchar2	2	Valid values are: - "CW" for check payments - "CE" for EFT payments
CHK_EFT_NO	Check/EFT Number	For CW files with check payments, this field may include a pre-assigned check number (left justified and leading spaces need to be filled with zeros). If check numbers are not supplied for Check payments by submitting agencies, they will be assigned by the application.	O	Varchar2	15	If provided, it must be numeric and cannot be all zeros. It applies for Check payments only.

XML Tag	Caption on CW Payment Table	Description	R/C	Type	Max Size	Comments
ACH_TRAN_CD	ACH Transaction Code	It indicates whether the vendor's Bank Account Number is a checking or savings account. Also, it indicates whether an EFT payment is a true payment or prenote.	CR	Number	2	<p>Required if EFT payment and the:</p> <ul style="list-style-type: none"> - Vendor is miscellaneous, - Vendor is non-miscellaneous and inactive, or - Vendor is non-miscellaneous, active, and not eligible for EFT. <p>Valid values are: <u>Checking Account:</u> 22 EF payment 23 Prenote (payment amount must be \$0) <u>Savings Account:</u> 32 EF Payment 33 Prenote (payment amount must be \$0)</p> <p>For EFT payments and under other conditions for non-miscellaneous vendors, this field will be overwritten / populated based on the setup of EFT information for the vendor on the VCUST table.</p>

XML Tag	Caption on CW Payment Table	Description	R/C	Type	Max Size	Comments
ACH_RECV_ID	Hidden	Routing Number or ABA Number of the vendor's Bank Account Number.	CR	Number	9	Required if EFT payment and the: - Vendor is miscellaneous, - Vendor is not miscellaneous and inactive, or - Vendor is not miscellaneous, active, and not eligible for EFT. For EFT payments and under other conditions for non-miscellaneous vendors, this field will be overwritten / populated from the ABA Number of the vendor on the VCUST table.
ACH_ACCT_NO_VI EW	Hidden	Bank Account Number of the vendor (left justified).	CR	Varchar2	17	Required if EFT payment and the: - Vendor is miscellaneous, - Vendor is not miscellaneous and inactive, or - Vendor is not miscellaneous, active, and not eligible for EFT. For EFT payments and under other conditions for non-miscellaneous vendors, this field will be overwritten / populated from the Account Number of the vendor on the VCUST table.

XML Tag	Caption on CW Payment Table	Description	R/C	Type	Max Size	Comments
ACH_RECV_NM	ACH Receiving Name	Supplied by the submitting department for tracking purposes. If desired, submitting departments can use this field to uniquely identify their CW EFT payments at the bank.	CR	Varchar2	16	Required for EFT payments. Will be posted to ACH Payment record (i.e. record type "6").
DSCRE_DATA	Discretionary Data	Submitting agencies may include codes of significance only to them. There is no standardized interpretation for the value of this field.	O	Varchar2	2	Can be provided for EFT payments.
TOT_ADNM_LN	Total Number of ACH Addendum Lines	It indicates if an EFT payment is associated with an ACH Addendum record.	O	Number	4	For EFT payments, valid values are "0" (if no Addendum record is associated) or "1" (if an Addendum record is associated).
PYMT_REMT_ADV	Remittance Advice Information	Payment-related information that can be provided by submitting agencies for check and EFT payments.	O	Varchar2	1500	Provided information will be printed on the check or RA stub. Refer to section 3.3.3 for further information.

ACH Addendum Payment Section

XML Tag	Caption on ACH Addendum Table	Description	R/C	Type	Max Size	eMARS Comments
LN_NO	Line Number	The payment line number that uniquely identifies each payment line within a CW file.	R	Number	5	Must match the Payment Line Number of the corresponding record on the CW Payment Component.
ACH_ADD_SEQ_NO	ACH Addendum Sequence Number	The ACH addendum sequence number that uniquely identifies an ACH Addendum line within an EFT payment.	R	Number	4	This value can only be "1" since the ACH formats CCD and PPD allow only one addendum record per EFT payment.
ACH_PYMT_INFO	ACH Payment Related Info	Payment-related information to be provided by submitting department.	R	Varchar2	80	This information will appear on the Record Type "7" of the ACH file.

Appendix B – Inventory of Check Writer Interface Files

Interface Name	Interface Description	Frequency	Rover ID (Interface ID) FINA.AFN1.SEQ.CW.xxxxxxxx
Business Tax Refund (Business Tax)	Check payments	Weekly	C30R1280
Cab 35 / Cab 48-49 Check EFT	Check and electronic payments	On-request	C47R063D
Cab 35 / Cab 48-49 Checks	Check payments	On-request	C47R063C
CFC - Children's Trust Fund 1099	Electronic payments	Daily	C48R232B
CFC - Children's Trust Fund Non-1099	Check and electronic payments	Daily	C48R232A
CFC/FAD Family Assistance Diversion	Check payments	Daily	C48R171B
CFC/FAD Family Assistance Diversion FAD	Check payments	Daily	C48R171A
CFC/FAD Family Assistance Diversion RT	Check payments	Daily	C48R171C
CFC/FST Food Stamps Training Program	Check payments	Daily	C48R1720
CFC/JAS Jobs Training (STEP) Kentucky Works	Check payments	Daily	C48R173A

Interface Name	Interface Description	Frequency	Rover ID (Interface ID) FINA.AFN1.SEQ.CW.xxxxxxxxxx
CFC/JAS Jobs Training (STEP) Kentucky Works	Check payments	Daily	C48R173B
CFC/JAS Jobs Training (STEP) Kentucky Works - type 38 non-TANF non-1099	Check payments	Daily	C48R173C
CFC/JAS Jobs Training (STEP) Kentucky Works non-TANF 1099	Check payments	Daily	C48R173D
CHFS MED KCHIP (Kentucky Children's Health Insurance Program)		Monthly	C53R2400
CHS Foster Grandparent Program	Check payments	Twice a month	C49R0190
Delinquent Property	Check payments	Quarterly	C30R2150
Direct deposit (Individual Income Refunds)	Electronic payments	Daily	C30R0750
Disability Determinations (DDS) 1099 RPT	Check payments	Weekly	C48R190A
Disability Determinations (DDS) non 1099 RPT	Check payments	Weekly	C48R190B
Error refunds (Individual Income Refunds) - sealed	Check payments	Daily	C30R177B
Error refunds (Individual Income Refunds) - unsealed	Check payments	Daily	C30R177A

Interface Name	Interface Description	Frequency	Rover ID (Interface ID) FINA.AFN1.SEQ.CW.xxxxxxxxx
GOT – Telecommunications – CMRS			C31R2230
HCTC Health insurance payment	Check payments	Weekly	C51R2350
Impact Plus CFC	Check payments	Weekly	C48R0160
Impact Plus CFC - New Version	Check payments	Weekly	C48R016A
Impact Plus CHS - New Version	Check payments	Weekly	C49R017A
KAMES - daily DEFRA	Check payments	Daily	C48R040A
KAMES - daily issuance education bonus	Check payments	Daily	C48R040K
KAMES - daily issuance employment retention	Check payments	Daily	C48R040L
KAMES - daily issuance kinship care	Check payments	Daily	C48R040N
KAMES - daily issuance kinship care start-up	Check payments	Daily	C48R040M
KAMES - daily issuance non ER checks	Check payments	Daily	C48R040J

Interface Name	Interface Description	Frequency	Rover ID (Interface ID) FINA.AFN1.SEQ.CW.xxxxxxxxx
KAMES - daily State Supplementation	Check payments	Daily	C48R040B
KAMES - daily TANF	Check payments	Daily	C48R040C
KAMES - first issuance kinship care	Check payments	Monthly	C48R040P
KAMES - first issuance kinship care ach	Electronic payments	Monthly	C48R040Q
KAMES - first issuance State Supplementation	Check payments	Monthly	C48R040F
KAMES - first issuance TANF checks	Check payments	Monthly	C48R040E
KAMES - first issuance TANF direct deposit	Electronic payments	Monthly	C48R040D
KAMES - second issuance kinship care	Check payments	Monthly	C48R040R
KAMES - second issuance kinship care ach	Electronic payments	Monthly	C48R040S
KAMES - second issuance State Supplementation	Check payments	Monthly	C48R040I
KAMES - second issuance TANF checks	Check payments	Monthly	C48R040H

Interface Name	Interface Description	Frequency	Rover ID (Interface ID) FINA.AFN1.SEQ.CW.xxxxxxxx
KAMES - second issuance TANF direct deposit	Electronic payments	Monthly	C48R040G
KAMES daily issuance kinship care start-up 1099	Check payments	Monthly	C48R040T
KASES / Child Support system	Check payments	Daily	C48R182A
KASES / Child Support system - EFT	Electronic payments	Daily	C48R182B
KASES / Child Support system - EFT - monthly	Electronic payments	Daily	C48R182D
KASES / Child Support system - EFT- Saturday Processing	Electronic payments	Daily	C48R182F
KASES / Child Support system - monthly	Check payments	Daily	C48R182C
KASES / Child Support system- Saturday Processing	Check payments	Daily	C48R182E
KDSS (Fish and Wildlife)- Agent Refunds	No records found	On-request	C43R166A
KDSS (Fish and Wildlife)- Security Deposit Refunds	Check payments	Bi-Weekly	C43R166C
Kentucky Child Care Management System KCCMS	Check payments	Weekly	C48R0490

Interface Name	Interface Description	Frequency	Rover ID (Interface ID) FINA.AFN1.SEQ.CW.xxxxxxxx
Kentucky Early Intervention System (KEIS)	Check payments	Twice a Month	C49R500A
Kentucky Retirement – CERH	Check payments	Bi-Monthly	C31R046D
Kentucky Retirement – CERH - EFT	Electronic payments	Bi-Monthly	C31R046I
Kentucky Retirement – CERH - EFT - Excess Benefits	No records found		C31R218N
Kentucky Retirement – CERH - Employer 25C Refunds	Check payments	Bi-Monthly	C31R046X
Kentucky Retirement – CERH - Excess Benefits	No records found		C31R218I
Kentucky Retirement – CERH - Insurance	Check payments		C31R218D
Kentucky Retirement – CERH - Member 25C Refunds	Check payments	Bi-Monthly	C31R046S
Kentucky Retirement – CERH - Member Refunds	Check payments	Bi-Monthly	C31R046N
Kentucky Retirement – CERS	Check payments	Bi-Monthly	C31R046B
Kentucky Retirement - CERS - EFT	Electronic payments	Bi-Monthly	C31R046G

Interface Name	Interface Description	Frequency	Rover ID (Interface ID) FINA.AFN1.SEQ.CW.xxxxxxxx
Kentucky Retirement - CERS - EFT - Excess Benefits	No records found		C31R218L
Kentucky Retirement - CERS - Employer 25C Refunds	Check payments	Bi-Monthly	C31R046V
Kentucky Retirement - CERS - Excess Benefits	No records found		C31R218G
Kentucky Retirement - CERS - Insurance	Check payments		C31R218B
Kentucky Retirement - CERS - Member 25C Refunds	Check payments	Bi-Monthly	C31R046Q
Kentucky Retirement - CERS - Member Refunds	Check payments	Bi-Monthly	C31R046L
Kentucky Retirement - KERH	Check payments	Bi-Monthly	C31R046E
Kentucky Retirement - KERH - EFT	Electronic payments	Bi-Monthly	C31R046J
Kentucky Retirement - KERH - EFT - Excess Benefits	No records found		C31R218O
Kentucky Retirement - KERH - Employer 25C Refunds	Check payments	Bi-Monthly	C31R046Y
Kentucky Retirement - KERH - Excess Benefits	No records found		C31R218J

Interface Name	Interface Description	Frequency	Rover ID (Interface ID) FINA.AFN1.SEQ.CW.xxxxxxxx
Kentucky Retirement - KERH - Insurance	Check payments		C31R218E
Kentucky Retirement - KERH - Member 25C Refunds	Check payments	Bi-Monthly	C31R046T
Kentucky Retirement - KERH - Member Refunds	Check payments	Bi-Monthly	C31R046O
Kentucky Retirement - KERS	Check payments	Bi-Monthly	C31R046A
Kentucky Retirement - KERS - EFT	Electronic payments	Bi-Monthly	C31R046F
Kentucky Retirement - KERS - EFT - Excess Benefits	No records found		C31R218K
Kentucky Retirement - KERS - Employer 25C Refunds	Check payments	Bi-Monthly	C31R046U
Kentucky Retirement - KERS - Excess Benefits	No records found		C31R218F
Kentucky Retirement - KERS - Insurance	Check payments		C31R218A
Kentucky Retirement - KERS - Member 25C Refunds	Check payments	Bi-Monthly	C31R046P
Kentucky Retirement - KERS - Member Refunds	Check payments	Bi-Monthly	C31R046K

Interface Name	Interface Description	Frequency	Rover ID (Interface ID) FINA.AFN1.SEQ.CW.xxxxxxxx
Kentucky Retirement – SPRS	Check payments	Bi-Monthly	C31R046C
Kentucky Retirement - SPRS - EFT	Electronic payments	Bi-Monthly	C31R046H
Kentucky Retirement - SPRS - EFT - Excess Benefits	No records found		C31R218M
Kentucky Retirement - SPRS - Employer 25C Refunds	Check payments	Bi-Monthly	C31R046W
Kentucky Retirement - SPRS - Excess Benefits	No records found		C31R218H
Kentucky Retirement - SPRS - Insurance	Check payments		C31R218C
Kentucky Retirement - SPRS - Member 25C Refunds	Check payments	Bi-Monthly	C31R046R
Kentucky Retirement - SPRS - Member Refunds	No records found	Bi-Monthly	C31R046M
KRS – Payroll	Check and electronic payments	Twice a Month	C31R2310
Labor Cabinet - CWPf Coal Fund - unsealed - N	Check and electronic payments	On-request	C44R051N
Labor Cabinet – CWPf Pneumoconiosis attorney fees - M	Check and electronic payments	On-request	C44R051M

Interface Name	Interface Description	Frequency	Rover ID (Interface ID) FINA.AFN1.SEQ.CW.xxxxxxxx
Labor Cabinet – CWPf Pneumoconiosis interest - L	Check and electronic payments	On-request	C44R051L
Labor Cabinet – CWPf Pneumoconiosis- K	Check and electronic payments	On-request	C44R051K
Labor Cabinet - Green Coal - E	Check and electronic payments	On-request	C44R051E
Labor Cabinet - Green Coal attorney fees - D	Check and electronic payments	On-request	C44R051G
Labor Cabinet - Green Coal interest – F	Check and electronic payments	On-request	C44R051F
Labor Cabinet - Southeast Coal - H	Check and electronic payments	On-request	C44R051H
Labor Cabinet - Southeast Coal attorney fees -J	Check and electronic payments	On-request	C44R051J
Labor Cabinet - Southeast Coal interest - I	Check and electronic payments	On-request	C44R051I
Labor Cabinet - Special Fund - A- sealed	Check and electronic payments	Weekly	C44R051A
Labor Cabinet - Special Fund - B- unsealed	Check and electronic payments	On-request	C44R051B

Interface Name	Interface Description	Frequency	Rover ID (Interface ID) FINA.AFN1.SEQ.CW.xxxxxxxx
Labor Cabinet - Special Fund attorney fees - D	Check and electronic payments	On-request	C44R051D
Labor Cabinet - Special Fund interest - C	Check and electronic payments	On-request	C44R051C
LGEAF - Coal Severance/Mineral Severance Non CAPS, CITY	Check payments	Monthly	C31R130C
LGEAF - Coal Severance/Mineral Severance Non CAPS, CITY	Check payments	Monthly	C31R130D
LGEAF - Coal Severance/Mineral Severance Non CAPS, County	Check payments	Monthly	C31R130A
LGEAF - Coal Severance/Mineral Severance Non CAPS, County	Check payments	Monthly	C31R130B
Local Government - CDBG - EFT	Electronic payments	Bi-Monthly	C31R0520
Local Government - Road Aid Tax, City	Check payments	Monthly	C31R053B
Local Government - Road Aid Tax, County	Check payments	Monthly	C31R053A
LRC Daily Exp	Check payments	Bi-Monthly	C10R1880
LRC Session Mileage	Check payments	Bi-Monthly	C10R188A

Interface Name	Interface Description	Frequency	Rover ID (Interface ID) FINA.AFN1.SEQ.CW.xxxxxxxx
LRC Travel Data	Check payments	Monthly	C10R1890
Motor Carrier IRP Licensing System	Check payments	Bi-Weekly	C35R165B
Motor Carrier IRP Settlement	Check payments	Bi-Weekly	C35R165A
Motor Carrier Tax Refunds	Check payments	Bi-Weekly	C35R1650
Motor Fuel Refund (Business Tax)	Check payments	Monthly	C30R0790
Non-error refunds (Individual Income Refunds)	Check payments	Daily	C30R1780
Personnel (UPPS)	Check payments	On-request	C47R063A
Personnel (UPPS) EFT	Check and electronic payments	On-request	C47R063B
Public Assistance System - PAS - First regular	Check payments	Daily	C48R161A
Public Assistance System - PAS - Second regular checks	Check payments		C48R161C
Public Assistance System - PAS - State Supplementation monthly	Check payments		C48R161B

Interface Name	Interface Description	Frequency	Rover ID (Interface ID) FINA.AFN1.SEQ.CW.xxxxxxxxx
REOPT204	Check payments	Quarterly	C30R1860
Revenue – Disbursements of Communication Tax Receipts	Check and Electronic Payments	Monthly	C30R2390
Revenue School Tax - Distribution of Tax Receipts to Schools	Electronic Payments	Monthly	C39R2370
REVRCS65	Check payments	Monthly	C30R1850
Sec of State - UCC Refunds	Check payments	Weekly	C31R2250
State Fair Board	No records found	On-request	C43R0830
Teachers Retirement - Medical Insurance Checks	Check payments	Quarterly	C33R085F
Teachers Retirement - Retiree Payroll	Check and electronic payments	Monthly	C33R085A
Teachers Retirement - Retiree Payroll - Ancillary	Check payments		C33R085D
Teachers Retirement - Retiree Payroll - Refunds	Check payments		C33R085C
Teachers Retirement Payroll	Check and electronic payments	3 Times a Month	C33R085B

Interface Name	Interface Description	Frequency	Rover ID (Interface ID) FINA.AFN1.SEQ.CW.xxxxxxxx
Treasury -for employee child support	Check payments	Bi-Weekly	C31R2100
Treasury-Unclaimed Property being programmed by Wagers & Assoc.	Check payments	Bi-Weekly	C31R2190
TWIST CFC	Check payments	3,14&25	C48R105A
TWIST CFC	Check payments	3,14&25	C48R105B
TWIST CFC EFT 1099	Electronic payments	3,14&25	C48R105D
TWIST CFC EFT NON-1099	Electronic payments	3,14&25	C48R105C
Unemployment Insurance Claims and Benefits - Additional day 1	Check payments	As needed	C46R163B
Unemployment Insurance Claims and Benefits - Additional day 2	Check payments	As needed	C46R163C
Unemployment Insurance Claims and Benefits - Additional day 3	Check payments	As needed	C46R163D
Unemployment Insurance Claims and Benefits - Daily	Check payments	As needed	C46R163A
Unisys (EDS) Medicaid (Medicaid Management Information System.MMIS)	Check payments	Weekly	C49R109B

Interface Name	Interface Description	Frequency	Rover ID (Interface ID) FINA.AFN1.SEQ.CW.xxxxxxxxx
Unisys (EDS) Medicaid (Medicaid Management Information System.MMIS) - 1099 Reportable	Check payments	Weekly	C49R109A
Unisys (EDS) Medicaid (Medicaid Management Information System.MMIS) EFT- 1099 Reportable	Check payments	Weekly	C49R109C
Upload of Property Tax	Check payments	Quarterly	C30R1870
Kentucky Child Care Management System KCCMS	EFT payments		C48r2360

Appendix C – Example of a Check Writer Input XML File

The following example includes the following check writer components:

- 1 Header Line
- 2 Accounting Lines
- 4 Payment Lines: Payment Lines 2 and 3 are check payments, Payment Lines 1 and 4 are EFT payments
- Payment Line 1 is associated with 1 Addendum Line
- Payment Line 4 is associated with 1 Addendum Line

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<AMS_DATAOBJECT_XML_EXPORT_FILE>
  <R_AP_CW_HDR AMSDataObject="Y">
    <CW_DEPT_CD Attribute="Y"><![CDATA[500]]></CW_DEPT_CD>
    <CW_FILE_ID Attribute="Y"><![CDATA[123456789]]></CW_FILE_ID>
    <CW_UNIT_CD Attribute="Y"><![CDATA[1234]]></CW_UNIT_CD>
    <BANK_ACCT_CD Attribute="Y"><![CDATA[G1]]></BANK_ACCT_CD>
    <PYMT_DT Attribute="Y"><![CDATA[2005-09-10]]></PYMT_DT>
    <PYMT_DSCR Attribute="Y"><![CDATA[FOOD STAMP EMPLOYEE]]></PYMT_DSCR>
    <CNTAC_CD Attribute="Y"><![CDATA[5555]]></CNTAC_CD>
    <BFY Attribute="Y"><![CDATA[2005]]></BFY>
    <FY Attribute="Y"><![CDATA[2005]]></FY>
    <PER Attribute="Y"><![CDATA[6]]></PER>
    <EFT_FRMT Attribute="Y"><![CDATA[CCD]]></EFT_FRMT>
    <CHK_FRMT Attribute="Y"><![CDATA[REG]]></CHK_FRMT>
    <DISB_CAT Attribute="Y"><![CDATA[GA]]></DISB_CAT>
    <TOT_PYMT_AM Attribute="Y"><![CDATA[5000.00]]></TOT_PYMT_AM>
    <NO_ACTG_LN Attribute="Y"><![CDATA[2]]></NO_ACTG_LN>
    <NO_PYMT_LN Attribute="Y"><![CDATA[4]]></NO_PYMT_LN>
    <PERMIT_NO Attribute="Y"><![CDATA[null]]></PERMIT_NO>
  </R_AP_CW_HDR>
  <R_AP_CW_ACTG AMSDataObject="Y">
```

```

<LN_NO Attribute="Y"><![CDATA[1]]></LN_NO>
<EVNT_TYP Attribute="Y"><![CDATA[EV01]]></EVNT_TYP>
<ACTG_TMPL_ID Attribute="Y"><![CDATA[TEST1]]></ACTG_TMPL_ID>
<FUND_CD Attribute="Y"><![CDATA[1200]]></FUND_CD>
<SFUND_CD Attribute="Y"><![CDATA[null]]></SFUND_CD>
<DEPT_CD Attribute="Y"><![CDATA[500]]></DEPT_CD>
<UNIT_CD Attribute="Y"><![CDATA[UNIT]]></UNIT_CD>
<SUNIT_CD Attribute="Y"><![CDATA[null]]></SUNIT_CD>
<APPR_CD Attribute="Y"><![CDATA[AAAA]]></APPR_CD>
<OBJ_CD Attribute="Y"><![CDATA[E119]]></OBJ_CD>
<SOBJ_CD Attribute="Y"><![CDATA[null]]></SOBJ_CD>
<RSRC_CD Attribute="Y"><![CDATA[null]]></RSRC_CD>
<SRSRC_CD Attribute="Y"><![CDATA[null]]></SRSRC_CD>
<DOBJ_CD Attribute="Y"><![CDATA[null]]></DOBJ_CD>
<DRSRC_CD Attribute="Y"><![CDATA[null]]></DRSRC_CD>
<ACTV_CD Attribute="Y"><![CDATA[null]]></ACTV_CD>
<SACTV_CD Attribute="Y"><![CDATA[null]]></SACTV_CD>
<FUNC_CD Attribute="Y"><![CDATA[null]]></FUNC_CD>
<SFUNC_CD Attribute="Y"><![CDATA[null]]></SFUNC_CD>
<BSA_CD Attribute="Y"><![CDATA[null]]></BSA_CD>
<SBSA_CD Attribute="Y"><![CDATA[null]]></SBSA_CD>
<RPT_CD Attribute="Y"><![CDATA[null]]></RPT_CD>
<SRPT_CD Attribute="Y"><![CDATA[null]]></SRPT_CD>
<LOC_CD Attribute="Y"><![CDATA[null]]></LOC_CD>
<SLOC_CD Attribute="Y"><![CDATA[null]]></SLOC_CD>
<TASK_CD Attribute="Y"><![CDATA[null]]></TASK_CD>
<STASK_CD Attribute="Y"><![CDATA[null]]></STASK_CD>
<TASK_ORD_CD Attribute="Y"><![CDATA[null]]></TASK_ORD_CD>
<PROG_CD Attribute="Y"><![CDATA[null]]></PROG_CD>
<PHASE_CD Attribute="Y"><![CDATA[null]]></PHASE_CD>
<PPC_CD Attribute="Y"><![CDATA[null]]></PPC_CD>
<LN_AM Attribute="Y"><![CDATA[2000.00]]></LN_AM>
</R_AP_CW_ACTG>
<R_AP_CW_ACTG AMSDataObject="Y">
  <LN_NO Attribute="Y"><![CDATA[2]]></LN_NO>

```

```

<EVNT_TYP Attribute="Y"><![CDATA[EV02]]></EVNT_TYP>
<ACTG_TMPL_ID Attribute="Y"><![CDATA[TEST2]]></ACTG_TMPL_ID>
<FUND_CD Attribute="Y"><![CDATA[null]]></FUND_CD>
<SFUND_CD Attribute="Y"><![CDATA[null]]></SFUND_CD>
<DEPT_CD Attribute="Y"><![CDATA[null]]></DEPT_CD>
<UNIT_CD Attribute="Y"><![CDATA[UNIT]]></UNIT_CD>
<SUNIT_CD Attribute="Y"><![CDATA[null]]></SUNIT_CD>
<APPR_CD Attribute="Y"><![CDATA[null]]></APPR_CD>
<OBJ_CD Attribute="Y"><![CDATA[null]]></OBJ_CD>
<SOBJ_CD Attribute="Y"><![CDATA[null]]></SOBJ_CD>
<RSRC_CD Attribute="Y"><![CDATA[null]]></RSRC_CD>
<SRSRC_CD Attribute="Y"><![CDATA[null]]></SRSRC_CD>
<DOBJ_CD Attribute="Y"><![CDATA[null]]></DOBJ_CD>
<DRSRC_CD Attribute="Y"><![CDATA[null]]></DRSRC_CD>
<ACTV_CD Attribute="Y"><![CDATA[null]]></ACTV_CD>
<SACTV_CD Attribute="Y"><![CDATA[null]]></SACTV_CD>
<FUNC_CD Attribute="Y"><![CDATA[null]]></FUNC_CD>
<SFUNC_CD Attribute="Y"><![CDATA[null]]></SFUNC_CD>
<BSA_CD Attribute="Y"><![CDATA[null]]></BSA_CD>
<SBSA_CD Attribute="Y"><![CDATA[null]]></SBSA_CD>
<RPT_CD Attribute="Y"><![CDATA[null]]></RPT_CD>
<SRPT_CD Attribute="Y"><![CDATA[null]]></SRPT_CD>
<LOC_CD Attribute="Y"><![CDATA[null]]></LOC_CD>
<SLOC_CD Attribute="Y"><![CDATA[null]]></SLOC_CD>
<TASK_CD Attribute="Y"><![CDATA[null]]></TASK_CD>
<STASK_CD Attribute="Y"><![CDATA[null]]></STASK_CD>
<TASK_ORD_CD Attribute="Y"><![CDATA[null]]></TASK_ORD_CD>
<PROG_CD Attribute="Y"><![CDATA[null]]></PROG_CD>
<PHASE_CD Attribute="Y"><![CDATA[null]]></PHASE_CD>
<PPC_CD Attribute="Y"><![CDATA[null]]></PPC_CD>
<LN_AM Attribute="Y"><![CDATA[3000.00]]></LN_AM>
</R_AP_CW_ACTG>
<R_AP_CW_PYMT AMSDataObject="Y">
  <LN_NO Attribute="Y"><![CDATA[1]]></LN_NO>
  <VEND_CUST_CD Attribute="Y"><![CDATA[NON-MISC EFT VEND]]></VEND_CUST_CD>

```

```

<TIN Attribute="Y"><![CDATA[null]]></TIN>
<TIN_TYP Attribute="Y"><![CDATA[null]]></TIN_TYP>
<AD_ID Attribute="Y"><![CDATA[AD001]]></AD_ID>
<CNTAC_ID Attribute="Y"><![CDATA[null]]></CNTAC_ID>
<LGL_NM Attribute="Y"><![CDATA[null]]></LGL_NM>
<ALIAS_NM Attribute="Y"><![CDATA[null]]></ALIAS_NM>
<AD_LN_1 Attribute="Y"><![CDATA[null]]></AD_LN_1>
<AD_LN_2 Attribute="Y"><![CDATA[null]]></AD_LN_2>
<CITY Attribute="Y"><![CDATA[null]]></CITY>
<ST Attribute="Y"><![CDATA[null]]></ST>
<ZIP Attribute="Y"><![CDATA[null]]></ZIP>
<CTRY Attribute="Y"><![CDATA[null]]></CTRY>
<DLVR_PT Attribute="Y"><![CDATA[null]]></DLVR_PT>
<PYMT_AM Attribute="Y"><![CDATA[750.00]]></PYMT_AM>
<CMNT Attribute="Y"><![CDATA[null]]></CMNT>
<PYMT_CD Attribute="Y"><![CDATA[CE]]></PYMT_CD>
<CHK_EFT_NO Attribute="Y"><![CDATA[null]]></CHK_EFT_NO>
<ACH_TRAN_CD Attribute="Y"><![CDATA[null]]></ACH_TRAN_CD>
<ACH_RECV_ID Attribute="Y"><![CDATA[null]]></ACH_RECV_ID>
<ACH_ACCT_NO_VIEW Attribute="Y"><![CDATA[null]]></ACH_ACCT_NO_VIEW>
<ACH_RECV_NM Attribute="Y"><![CDATA[12345]]></ACH_RECV_NM>
<DSCRE_DATA Attribute="Y"><![CDATA[null]]></DSCRE_DATA>
<TOT_ADNM_LN Attribute="Y"><![CDATA[1]]></TOT_ADNM_LN>
<PYMT_REMT_ADV Attribute="Y"><![CDATA[INVOICE# A1123 - $200.00]]></PYMT_REMT_ADV>
</R_AP_CW_PYMT>
<R_AP_CW_PYMT AMSDataObject="Y">
  <LN_NO Attribute="Y"><![CDATA[2]]></LN_NO>
  <VEND_CUST_CD Attribute="Y"><![CDATA[NON-MISC CHK VEND]]></VEND_CUST_CD>
  <TIN Attribute="Y"><![CDATA[null]]></TIN>
  <TIN_TYP Attribute="Y"><![CDATA[null]]></TIN_TYP>
  <AD_ID Attribute="Y"><![CDATA[AD001]]></AD_ID>
  <CNTAC_ID Attribute="Y"><![CDATA[null]]></CNTAC_ID>
  <LGL_NM Attribute="Y"><![CDATA[null]]></LGL_NM>
  <ALIAS_NM Attribute="Y"><![CDATA[null]]></ALIAS_NM>
  <AD_LN_1 Attribute="Y"><![CDATA[null]]></AD_LN_1>

```



```

<AD_LN_2 Attribute="Y"><![CDATA[null]]></AD_LN_2>
<CITY Attribute="Y"><![CDATA[null]]></CITY>
<ST Attribute="Y"><![CDATA[null]]></ST>
<ZIP Attribute="Y"><![CDATA[null]]></ZIP>
<CTRY Attribute="Y"><![CDATA[null]]></CTRY>
<DLVR_PT Attribute="Y"><![CDATA[null]]></DLVR_PT>
<PYMT_AM Attribute="Y"><![CDATA[1500.00]]></PYMT_AM>
<CMNT Attribute="Y"><![CDATA[null]]></CMNT>
<PYMT_CD Attribute="Y"><![CDATA[CW]]></PYMT_CD>
<CHK_EFT_NO Attribute="Y"><![CDATA[0000005]]></CHK_EFT_NO>
<ACH_TRAN_CD Attribute="Y"><![CDATA[null]]></ACH_TRAN_CD>
<ACH_RECV_ID Attribute="Y"><![CDATA[null]]></ACH_RECV_ID>
<ACH_ACCT_NO_VIEW Attribute="Y"><![CDATA[null]]></ACH_ACCT_NO_VIEW>
<ACH_RECV_NM Attribute="Y"><![CDATA[12345]]></ACH_RECV_NM>
<DSCRE_DATA Attribute="Y"><![CDATA[null]]></DSCRE_DATA>
<TOT_ADNM_LN Attribute="Y"><![CDATA[0]]></TOT_ADNM_LN>
<PYMT_REMT_ADV Attribute="Y"><![CDATA[PO#: 666 AMT: $1,500.00 DESC: STATIONARY]]></PYMT_REMT_ADV>
</R_AP_CW_PYMT>
<R_AP_CW_PYMT AMSDataObject="Y">
  <LN_NO Attribute="Y"><![CDATA[3]]></LN_NO>
  <VEND_CUST_CD Attribute="Y"><![CDATA[MISC CHK VEND]]></VEND_CUST_CD>
  <TIN Attribute="Y"><![CDATA[223344556]]></TIN>
  <TIN_TYP Attribute="Y"><![CDATA[2]]></TIN_TYP>
  <AD_ID Attribute="Y"><![CDATA[null]]></AD_ID>
  <CNTAC_ID Attribute="Y"><![CDATA[null]]></CNTAC_ID>
  <LGL_NM Attribute="Y"><![CDATA[JOE SMITH, JR]]></LGL_NM>
  <ALIAS_NM Attribute="Y"><![CDATA[null]]></ALIAS_NM>
  <AD_LN_1 Attribute="Y"><![CDATA[500 MAIN STREET, SUITE # 5]]></AD_LN_1>
  <AD_LN_2 Attribute="Y"><![CDATA[null]]></AD_LN_2>
  <CITY Attribute="Y"><![CDATA[FRANKFORT]]></CITY>
  <ST Attribute="Y"><![CDATA[KY]]></ST>
  <ZIP Attribute="Y"><![CDATA[40601-1234]]></ZIP>
  <CTRY Attribute="Y"><![CDATA[null]]></CTRY>
  <DLVR_PT Attribute="Y"><![CDATA[A1]]></DLVR_PT>
  <PYMT_AM Attribute="Y"><![CDATA[1100.00]]></PYMT_AM>

```

```

<CMNT Attribute="Y"><![CDATA[REIMBURSEMENT FOR OCTOBER EXPENSES]]></CMNT>
<PYMT_CD Attribute="Y"><![CDATA[CW]]></PYMT_CD>
<CHK_EFT_NO Attribute="Y"><![CDATA[0000006]]></CHK_EFT_NO>
<ACH_TRAN_CD Attribute="Y"><![CDATA[null]]></ACH_TRAN_CD>
<ACH_RECV_ID Attribute="Y"><![CDATA[null]]></ACH_RECV_ID>
<ACH_ACCT_NO_VIEW Attribute="Y"><![CDATA[null]]></ACH_ACCT_NO_VIEW>
<ACH_RECV_NM Attribute="Y"><![CDATA[12345]]></ACH_RECV_NM>
<DSCRE_DATA Attribute="Y"><![CDATA[null]]></DSCRE_DATA>
<TOT_ADNM_LN Attribute="Y"><![CDATA[0]]></TOT_ADNM_LN>
<PYMT_REMT_ADV Attribute="Y"><![CDATA[null]]></PYMT_REMT_ADV>
</R_AP_CW_PYMT>
<R_AP_CW_PYMT AMSDataObject="Y">
  <LN_NO Attribute="Y"><![CDATA[4]]></LN_NO>
  <VEND_CUST_CD Attribute="Y"><![CDATA[MISC EFT VEND]]></VEND_CUST_CD>
  <TIN Attribute="Y"><![CDATA[1231231239]]></TIN>
  <TIN_TYP Attribute="Y"><![CDATA[1]]></TIN_TYP>
  <AD_ID Attribute="Y"><![CDATA[null]]></AD_ID>
  <CNTAC_ID Attribute="Y"><![CDATA[null]]></CNTAC_ID>
  <LGL_NM Attribute="Y"><![CDATA[SAM WOOD & SONS]]></LGL_NM>
  <ALIAS_NM Attribute="Y"><![CDATA[null]]></ALIAS_NM>
  <AD_LN_1 Attribute="Y"><![CDATA[500 HIGH STREET]]></AD_LN_1>
  <AD_LN_2 Attribute="Y"><![CDATA[null]]></AD_LN_2>
  <CITY Attribute="Y"><![CDATA[LEXINGTON]]></CITY>
  <ST Attribute="Y"><![CDATA[KY]]></ST>
  <ZIP Attribute="Y"><![CDATA[40512-6789]]></ZIP>
  <CTRY Attribute="Y"><![CDATA[null]]></CTRY>
  <DLVR_PT Attribute="Y"><![CDATA[null]]></DLVR_PT>
  <PYMT_AM Attribute="Y"><![CDATA[1650.00]]></PYMT_AM>
  <CMNT Attribute="Y"><![CDATA[null]]></CMNT>
  <PYMT_CD Attribute="Y"><![CDATA[CE]]></PYMT_CD>
  <CHK_EFT_NO Attribute="Y"><![CDATA[null]]></CHK_EFT_NO>
  <ACH_TRAN_CD Attribute="Y"><![CDATA[22]]></ACH_TRAN_CD>
  <ACH_RECV_ID Attribute="Y"><![CDATA[12345678]]></ACH_RECV_ID>
  <ACH_ACCT_NO_VIEW Attribute="Y"><![CDATA[666555]]></ACH_ACCT_NO_VIEW>
  <ACH_RECV_NM Attribute="Y"><![CDATA[VI#12345]]></ACH_RECV_NM>

```

```
<DSCRE_DATA Attribute="Y"><![CDATA[null]]></DSCRE_DATA>
<TOT_ADNM_LN Attribute="Y"><![CDATA[1]]></TOT_ADNM_LN>
<PYMT_REMT_ADV Attribute="Y"><![CDATA[null]]></PYMT_REMT_ADV>
</R_AP_CW_PYMT>
<R_AP_CW_ADNM AMSDataObject="Y">
  <LN_NO Attribute="Y"><![CDATA[1]]></LN_NO>
  <ACH_ADD_SEQ_NO Attribute="Y"><![CDATA[1]]></ACH_ADD_SEQ_NO>
  <ACH_PYMT_INFO Attribute="Y"><![CDATA[RMT*IV*99933*500.00\DTM*003*041230\]]></ACH_PYMT_INFO>
</R_AP_CW_ADNM>
<R_AP_CW_ADNM AMSDataObject="Y">
  <LN_NO Attribute="Y"><![CDATA[4]]></LN_NO>
  <ACH_ADD_SEQ_NO Attribute="Y"><![CDATA[1]]></ACH_ADD_SEQ_NO>
  <ACH_PYMT_INFO Attribute="Y"><![CDATA[RMT*IV*61122336*1650.00\DTM*003*041225\]]></ACH_PYMT_INFO>
</R_AP_CW_ADNM>
</AMS_DATAOBJECT_XML_EXPORT_FILE>
```