

WebSphere Data Interchange v3.2 Data Transformation SetElementAttribute() Implementation Guide

Data Transformation SetElementAttribute() CSD Overview

WebSphere Data Interchange(WDI) development has created this CSD in response to the immediate demand to support significant zeros in optional EDI data elements with Data Transformation processing. This document is not a formal IBM publication. It is a technical document written by WDI development to help customers implement the Data Transformation Map's new SetElementAttribute() mapping command using this CSD. This CSD requires WDI Client fix pack 14.

There is a particular need for customers to control zero values for an EDI optional Data Element in Data Transformation mapping. This was handled with the &ZEROSIG mapping command for Send mapping. To support this requirement, WDI has added a new Data Transformation mapping command called SetElementAttribute().

SetElementAttribute() Mapping Command:

The purpose of the new mapping command is to provide some control of the automatic formatting of simple elements that occurs when the element is output. WDI currently does automatic formatting of values when a simple element is output. For example, leading and trailing zeros will be removed from EDI numeric data elements when they are output. If the value is zero and the data element is optional, the value is suppressed in the EDI output. With the SetElementAttribute() mapping command customers can control certain types of formatting when a simple element is output, such as how zeros, leading and trailing blanks, left and right justification, and empty elements are handled when they are output. The SetElementAttribute() command does not affect the values of simple elements during translation itself. It only affects the formatting of simple elements when they are written to the output document.

The output formatting provided by the SetElementAttribute() command takes place after validation of data that will be output to an EDI document. However, the validation of the EDI data does consider the formatting that will occur because of the SetElementAttribute() command.

The command syntax, attribute names, and attribute values along with the current default setting for each document syntax type are explained in a later section of this document.

Scope of the SetElementAttribute() Command

The SetElementAttribute() command can be placed anywhere in a Data Transformation Map. There is a path specification in the command identifying a simple element or compound element in the target document definition. The target path identifies the scope of the command execution. The scope can be for the entire document, specific compound elements (including loops, segments and records, and composite elements and structures), and specific simple elements. This allows customers to apply the formatting control at specific levels in the mapping without the need of a mapping command for all simple elements involved.

It is important to understand the scope of the command. Excessive use of the SetElementAttribute() command will increase WDI memory usage during map execution. An explanation of the scope and how this command can be used follows:

The **SuppressZeroValues** element attribute has a default value of 'Y' for target EDI Standards documents. This means that optional simple elements with a value of zero will not have their zero value output to the target document. The default handling of zero values can be changed for the entire map execution using the following command:

```
SetElementAttribute ( TargetRoot, "SuppressZeroValues", 'N' )
```

Since the target path is the document root, the scope is the entire target document. This will result in the zero value of all optional simple elements with zero values being written to the target EDI Standards document. Additional SetElementAttribute() commands can be specified to override the updated default value.

In the same map, if a particular compound element, such as a segment, should suppress zero values on all simple elements within the compound element, the compound element can be specified in the SetElementAttribute() to override the default attribute value for all simple elements contained within the compound element. For example:

```
SetElementAttribute ( \Table 1\230 O O8\, "SuppressZeroValues", "Y" )
```

In this example scope of the SetElementAttribute() command is identified by the compound element path "\Table 1\230 O O8\". This is EDI Segment O8 at position 230 in Table 1. This means that optional numeric EDI Data Elements with zero values within EDI Segment O8 will not have their zero value written to the output document. Their zero values will be *suppressed*. EDI Data Elements with zero values within EDI Segments before and after the O8 EDI Segment will not suppress the zero values because the scope of this SetElementAttribute() command is for the specific EDI Segment identified in the command. The previous SetElementAttribute() command set the default element attribute that will be used for all other EDI Segments and EDI Data Elements within the target document.

Consider the scope of the SetElementAttribute() command when mapping. For example, if you want to suppress zero values most of the time and only a few elements should have zero values, set the element attribute for the entire document to suppress zero values and set the element attribute for the particular elements to stop the zero suppression. If you find that all elements in a particular compound element should not suppress zero values, set the attribute for the entire compound element as opposed to setting the attribute on each element in the compound element. Also consider the default formatting for each document syntax type.

Also note that if you set an element attribute on an element before the element is created, then an empty target element will be created. This is effectively like doing an implicit "Create()" command on the target element.

SetElementAttribute() Command Syntax:

The **SetElementAttribute()** command is used to alter the default formatting of a simple element when it is output to a document. The **SetElementAttribute()** command has the following format:

SetElementAttribute (*targetPath*, *attributeName*, *attributeValue*)

Where:

targetPath – is a path in the target document definition that identifies a simple or compound element. The reserved word **TargetRoot** can be specified to indicate that the attribute value will be the default for the element attribute for all simple elements within the document. The keyword **This** can be used to indicate the command is being applied to the parent element.

attributeName – is the name of the element attribute you want to set. See Element Attributes below for additional information about element attributes and a complete list of attributes that you can set. This parameter is typically a string constant, but can be a variable, a path identifying a simple element in the source document definition, or an expression that evaluates to a character string.

attributeValue – is the value the element attribute should be set to. This can be a numeric or string constant, variable, path identifying a simple element in the source document definition, or an expression.

Comments:

Use this command to alter the formatting characteristics applied to a simple element when it is output. Currently formatting characteristics include indicating whether elements with a value of an empty string or zero should be suppressed.

Every simple element has certain default formatting characteristics that are applied when the element is written to an output document. The **SetElementAttribute()** command can be used to change the formatting characteristics for a simple element, for all simple elements within a compound element, or for all simple elements within a document.

The **SetElementAttribute()** command can appear anywhere in a map.

Each use of the **SetElementAttribute()** command will increase memory usage so it is best to minimize the use of the command. If many elements require a common setting for an element attribute, consider specifying the path of a parent element or even the entire document in a single **SetElementAttribute()** command.

Certain compound node types should not be specified as the target path. These include table nodes found in EDI Standards, XML sequence nodes, XML choice nodes, and XML attribute lists.

When a compound element is specified as *targetPath*, the command has the effect of setting the default value for the attribute for all simple elements within the compound element. A default attribute value can

be overridden for a simple element by specifying the simple element in a SetElementAttribute() command.

The SetElementAttribute() command is used in Data Transformation Maps and Functional Acknowledgment Maps. It is not used in Validation Maps.

A SetElementAttribute() command is created by doing a right click on a node where you want to insert the command. In the pop up menu that is displayed, select an available **Insert** action, then select **Command**, and then **SetElementAttribute**. Doing this opens the Mapping Command Editor, where you will need to specify the *targetPath*, name of the element attribute, and the value of the element attribute. You can use **Values** action in the pop up menus displayed by right clicking on the *attributeName* and *attributeValue* parameters to set their values. Once the parameters have been entered, press the OK button. This will insert the SetElementAttribute() command where requested.

Example:

The following command ensure that the Data Element 353 in position 1 of the BEG Segment will be written even when it has a value of zero.

```
SetElementAttribute ( \Table 1\20 M BEG\1 M 353\\, "SuppressZeroValues", "N" )
```

SetElementAttribute() Command Element Attributes :

SuppressZeroValues (Y | N) – Indicates whether or not to output the value of a simple element declared to be of a numeric type, if it has a value of zero. Valid values are:

Y – Suppress zero values. Do not write the value of the simple element to output document if it has a value of zero. Compound elements with this value for the SuppressZeroValues element attribute will not be written to the output document if all of the elements within the compound element contain zero values.

N – Do not suppress zero values. Write the value of the simple element to the output document even when it has a value of zero.

Formatting notes for each document syntax type:

- **XML** – The default setting for this element attribute is “N” (do not suppress zero values). A value is considered to be numeric if the value contains all numbers or numbers with a decimal. The evaluation to determine if an XML value is numeric is performed after any optional operations to remove leading or trailing blanks. By default, trailing blanks are automatically removed for XML values while leading blanks are not automatically removed. For the following formatting examples, assume XML element “M803” contains an attribute element named “desc” and an element containing the value associated with the “M803” element. It is assumed that the SuppressEmptyElements element attribute is set to “N” for the “M803” element.

Example		Attribute element	Value element	Output Result
1	Value	0	0	<M803 desc="0">0</M803>
	SuppressZeroValues	N	N	
2	Value	0	0	<M803 desc="0"/>
	SuppressZeroValues	N	Y	
3	Value	0	0	<M803>0</M803>
	SuppressZeroValues	Y	N	
4	Value	0	0	<M803/>
	SuppressZeroValues	Y	Y	
6	Value	12	0	<M803 desc="12"/>
	SuppressZeroValues	Y	Y	
7	Value	0	82	<M803>82</M803>
	SuppressZeroValues	Y	Y	

- **EDI Standard** – The default setting for this element attribute is “Y” (suppress zero values). Mandatory Data Elements with a zero value will be written to the output document regardless of the setting for this element attribute. A Data Element is considered to be numeric if its data type is real or numeric

- **Data Format** – The default setting for this element attribute is “N” (do not suppress zero values). A Data Format Field is considered to be numeric if its data type is real, numeric, integer, binary, hexadecimal, packed decimal, or zoned decimal.

This element attribute has the effect of moving no value for fixed position Data Format output files. This means that numeric Fields will have blank values in the output record. This element attribute will affect the output of a Field for a comma separated values Data Format output file. If a Record contains four fields, RECORDID, TYPE, QUANTITY, and DESCRIPTION with the values “RECID”, “ABC”, 0, and “My Item Order” respectively, then the comma separated values Record will appear as follows when the SuppressZeroValues element attribute is set to “N” for the third Field:

“RECID”, “ABC”, 0, “My Item Order”

The Record will appear as follows when the SuppressZeroValues element attribute is set to “Y” for the third Field:

“RECID”, “ABC” ,, “My Item Order”

If the suppressed zero value is the last Field in the Data Format Record then the terminating comma for the last Field will not be written.

Command Example:

The following command alters the default setting for the SuppressZeroValues element attribute for all simple elements within the target document. The command indicates that simple elements will be written to the output document even when they contain a zero value unless otherwise specified.

SetElementAttribute (TargetRoot, “SuppressZeroValues”, “N”)

The following command alters the setting for the SuppressZeroValues element attribute for the M8 compound element within the target document. The command indicates that the M8 compound element will not be written to the output document if all child simple elements contain a zero value.

SetElementAttribute (\ADTBXML\M8\, “SuppressZeroValues”, “Y”)

SuppressEmptyElements (Y | N) – Indicates whether or not to output the value of a simple element if it has no value. A value is considered to have no value when it has a length of zero or it contains only white space. White space includes blanks, carriage returns, line feeds and tab characters. Valid values are:

Y – Suppress empty values. When applied to a simple element this value means that the simple element should not be written to the output document if it has no value. Compound elements with this value for the SuppressEmptyElements element attribute will not be written to the output document if no elements within the compound element contain a value.

N – Do not suppress empty values. The element will be written to the output document even if it has no value or all the elements contained within it have no value.

Formatting notes for each document syntax type:

- **XML** – The default setting for this element attribute is “N” (do not suppress empty elements). For the following formatting examples, assume XML element “M803” contains an attribute element named “desc” and an element containing the value associated with the “M803” element. It is assumed that the SuppressEmptyElements element attribute is set to “N” for the “M803” element.

Example		Attribute element	Value element	M803	Output Result
1	Value	“ ”	“ ”	-	<M803 desc=“ ”> </M803>
	SuppressEmptyElements	N	N	N	
2	Value	“ ”	“ ”	-	<M803 desc=“ ”/>
	SuppressEmptyElements	N	Y	N	
3	Value	“ ”	“ ”	-	<M803> </M803>
	SuppressEmptyElements	Y	N	N	
4	Value	“ ”	“ ”	-	<M803/>
	SuppressEmptyElements	Y	Y	N	
5	Value	“ABC”	“ ”	-	<M803 desc=“ABC”/>
	SuppressEmptyElements	Y	Y	N	
6	Value	“ ”	“DEF”	-	<M803>DEF</M803>
	SuppressEmptyElements	Y	Y	N	
7	Value	“ABC”	“ ”	-	<M803 desc=“ABC”/>
	SuppressEmptyElements	Y	Y	Y	
8	Element value	“ ”	“ ”	-	”M803” XML element is not written.
	SuppressEmptyElement	Y	Y	Y	

- **EDI Standard** – The default setting for this element attribute is “Y” (suppress empty elements). Mandatory Composite Data Elements and mandatory Data Elements will be written to the output document regardless of the setting for this element attribute. When the setting is “Y” for:
 - a Segment (mandatory or optional) then the Segment will not be written to the output document if all Data Elements within the Segment are empty.
 - an optional Composite Data Element then the Composite Data Element will not be written to the output document if all Data Elements within the Composite Data Element are empty. Only the required delimiters will be written.

- an optional Data Element then the Data Element will not be written to the output document. A delimiter will be written if required by the EDI Standard.

If the value is “N” on a Data Element, then the Data Element will be written to the output document even when it contains no value. This means that at a minimum the terminator for the Data Element preceding the Data Element with the “N” value for its SuppressEmptyElements element attribute will be written.

For the following formatting examples, assume EDI Mandatory Segment “M8” contains five elements:

- M801, data type alphanumeric (AN), optional, value of a single blank.
- M802, data type alphanumeric (AN), optional, value of a single blank.
- M803, data type real with zero significant decimal positions (R0), minimum and maximum length of 8, mandatory, value is zero.
- M804, data type real with four significant decimal positions (R4), minimum length of 8, maximum length of 12, mandatory, value is zero.
- M805, data type alphanumeric (AN), optional, value of a single blank.

It is assumed that the SuppressZeroValues element attribute is set to “Y” for the target document for this example.

Example	SuppressEmptyElements	Target Path	Output Result
1	N	M8	M8* * *00000000*000.0000 !
2	Y	M802	M8* **00000000*000.0000 !
3	Y	M8	M8***00000000*000.0000!

For the following formatting examples, assume EDI Mandatory Segment “M8” contains five elements:

- M801, data type alphanumeric (AN), optional, value of a single blank.
- M802, data type alphanumeric (AN), optional, value of a single blank.
- M803, data type real with zero significant decimal positions (R0), minimum and maximum length of 8, optional, value is zero.
- M804, data type real with four significant decimal positions (R4), minimum length of 8, maximum length of 12, optional, value is zero.
- M805, data type alphanumeric (AN), optional, value of a single blank.

It is assumed that the SuppressZeroValues element attribute is set to “Y” for the target document for this example.

Example	SuppressEmptyElements	Target Path	Output Result
1	N	M8	M8* * * * !
2	Y	M802	M8* * * * !
3	Y	M8	No Segment Produced

- **Data Format** – The default setting for this element attribute is “N” (do not suppress empty elements). If the SuppressEmptyElements element attribute is specified on a Data Format Record with a value of “Y” then the Data Format Record will not be written to the output document if no

Fields within the Record contain a value. If the SuppressEmptyElements element attribute is specified on a Data Format Loop with a value of “Y”, then each the Data Format Record within the Loop will not be written to the output document unless one or more Fields within the Record contain a value. This assumes that no Data Format Records within the Loop the SuppressEmptyElements element attribute have their value set to “N”.

The SuppressEmptyElements element attribute has no effect for a fixed position Data Format Field. It will affect the output of a Field in a comma separated values Data Format output file. If a Record contains four fields, RECORDID, TYPE, DESCRIPTION, and QUANTITY with the values “RECID”, “ABC”, “” and 0 respectively, then the comma separated values record will appear as follows when the SuppressEmptyElements element attribute is set to “N” for the third Field:

```
“RECID”,”ABC”,””,0
```

The record will appear as follows when the SuppressZeroValues element attribute is set to “Y” for the third Field:

```
“RECID”,”ABC”,,0
```

If the suppressed empty value is the last Field in the Data Format Record then the terminating comma for the last Field will not be written.

Command Example:

The following command alters the setting for the SuppressEmptyElements element attribute for the LOOPX02 compound element within the target document. The command indicates that the LOOPX02 compound element will not be written to the output document if all simple elements within it contain no significant value.

```
SetElementAttribute (\DATHDR_LOOP2\LOOPX02\, "SuppressEmptyElements", “Y”)
```

Notes:

- **XML** – A value is considered to be numeric if the value contains all numbers or numbers with a decimal. The evaluation to determine if an XML value is numeric is performed after any optional operations to remove leading or trailing blanks. By default, trailing blanks are automatically removed for XML values while leading blanks are not automatically removed.
- **EDI Standard** – A Data Element is considered to be numeric if its data type is real or numeric.
- **Data Format** – A Data Format Field is considered to be numeric if its data type is real, numeric, integer, binary, hexadecimal, packed decimal, or zoned decimal.

Mapping Tasks:

Outputting a Zero Value

A simple element with the value of zero may or may not be written to the target document. There are cases where it is desirable not to output a simple element with a value of zero. For instance, the size of the output document can be reduced by omitting elements with a zero value. Sometimes a requirement exists to prevent the output of elements with zero values except for a specific element.

Whether or not an element with a zero value is written depends on the document syntax type and whether the default formatting behavior has been overridden using the `SetElementAttribute()` command with the `SuppressZeroValues` element attribute. The `SetElementAttribute()` command is used to alter the default formatting characteristics of elements when they are written to the output document. The command has no effect on elements during translation. The `SuppressZeroValues` element attribute indicates whether a value of zero should be written to the output document.

Additional output formatting can be overridden using the `SuppressEmptyElements` element attribute. The `SuppressEmptyElements` element attribute can be used to prevent the output of elements containing no value and can also prevent the output of compound elements, such as EDI Segments and Data Format Records.

The `SuppressZeroValues` element attribute can have a value of “Y” or “N”. The “Y” value indicates that an element with a value of zero should not be written to the output document. “N” indicates that an element with the value zero should be written to the output document.

Each syntax type has the following default behavior for the `SuppressZeroValues` element attribute:

- **XML** – The default setting for the `SuppressZeroValues` element attribute is “N” (do not suppress zero values).
- **EDI Standard** – The default setting for the `SuppressZeroValues` element attribute is “Y” (suppress zero values). Mandatory Data Elements with a zero value will be written to the output document regardless of the setting for this element attribute.
- **Data Format** – The default setting for the `SuppressZeroValues` element attribute is “N” (do not suppress zero values). This element attribute has the effect of moving no value for fixed position Data Format output files. This means that numeric fields would have blank values in the output record. The element attribute will affect the output of a Field for a comma separated values Data Format output file.

Alter the default formatting behavior of an element using the `SetElementAttribute()` command. The default value for an element attribute can be changed for the entire target document by specifying the path as “TargetRoot”. Specifying a compound element as the path on the `SetElementAttribute()` command causes the default value for the specified element attribute to be changed for all elements within the compound element. Elements not within the compound element will not be affected by the `SetElementAttribute()` command. Finally, specify the simple element itself in the path of the `SetElementAttribute()` command to just change the formatting behavior for a specific element. If your

map is a target based map you can specify the keyword “This” as the path to indicate the command applies to the current element (the element that is the parent node of the command).

It is important to use as few SetElementAttribute() commands as you can to conserve computer memory. If most elements need to have their default formatting behavior overridden, then specify the SetElementAttributes() command once using “TargetRoot” as the path. If only a few elements need the default behavior changed then put specific SetElementAttribute() commands in just for those few elements.

Examples:

The following command alters the default setting for the SuppressZeroValues element attribute for all simple elements within the target document. The command indicates that simple elements will be written to the output document even when they contain a zero value unless otherwise specified.

SetElementAttribute (TargetRoot, “SuppressZeroValues”, “N”)

The following command alters the setting for the SuppressZeroValues element attribute for the M8 simple element within the target document. The command indicates that the M8 simple element will not be written to the output document if it contains a zero value.

SetElementAttribute (\ADTBXML\M8\, “SuppressZeroValues”, “Y”)

- **XML** – For the following formatting examples, assume XML element “M803” contains an attribute element named “desc” and an element containing the value associated with the “M803” element. It is assumed that the SuppressEmptyElements element attribute is set to “N” for the “M803” element.

Example		Attribute element	Value element	Output Result
1	Element value	0	0	<M803 desc="0">0</M803>
	SuppressZeroValues	N	N	
2	Element value	0	0	<M803 desc="0"/>
	SuppressZeroValues	N	Y	
3	Element value	0	0	<M803>0</M803>
	SuppressZeroValues	Y	N	
4	Element value	0	0	<M803/>
	SuppressZeroValues	Y	Y	
6	Element value	12	0	<M803 desc="12"/>
	SuppressZeroValues	Y	Y	
7	Element value	0	82	<M803>82</M803>
	SuppressZeroValues	Y	Y	

- **Data Format** – If a Record contains four fields, RECORDID, TYPE, QUANTITY, and DESCRIPTION with the values “RECID”, “ABC”, 0, and “My Item Order” respectively, then the

comma separated values Record will appear as follows when the SuppressZeroValues element attribute is set to “N” for the third Field:

```
“RECID”,“ABC”,0,“My Item Order”
```

The Record will appear as follows when the SuppressZeroValues element attribute is set to “Y” for the third Field:

```
“RECID”,“ABC”,,“My Item Order”
```

If the suppressed zero value is the last Field in the Data Format Record then the terminating comma for the last Field will not be written

.

Outputting a Blank or Empty Value

An element with no value may or may not be written to the target document. A simple element is considered to have “no value” when it has a length of zero or it contains only white space. White space includes blanks, carriage returns, line feeds and tab characters. A compound element, such as an EDI Segment or Data Format Record, is considered to have no value when it contains only elements with no value. When the output of compound elements is to be minimized, it is common to also specify the SuppressZeroValues element attribute with a value of “Y” so numeric elements that would otherwise be written with a zero value are omitted.

There are cases where it is desirable not to output a simple element with no value. For instance, the size of the output document can be reduced by omitting elements with no value. Sometimes a requirement exists to prevent the output of elements with no values except for a specific element.

Whether or not empty elements are written to the output document depends on the document syntax type and whether the default formatting behavior has been overridden using the SetElementAttribute() command with the SuppressEmptyElements element attribute. The SetElementAttribute() command is used to alter the default formatting characteristics of elements when they are written to the output document. The command has no effect on elements during translation. The SuppressEmptyElements element attribute indicates whether a:

- simple element with no value should be written to the output document.
- compound element that contains only elements with no value should be written to the output document.

The SuppressEmptyElements element attribute can have a value of “Y” or “N”. The “Y” value indicates that an empty element value should not be written to the output document. “N” indicates that an empty element should be written to the output document.

Each syntax type has the following default behavior for the SuppressEmptyElements element attribute:

- **XML** – The default setting for the SuppressEmptyElements element attribute is “N” (do not suppress empty elements).
- **EDI Standard** – The default setting for this element attribute is “Y” (suppress empty elements). Mandatory Composite Data Elements and mandatory Data Elements will be written to the output document regardless of the setting for this element attribute. When the setting is “Y” for:
 - a Segment (mandatory or optional) then the Segment will not be written to the output document if all Data Elements within the Segment are empty.
 - an optional Composite Data Element then the Composite Data Element will not be written to the output document if all Data Elements within the Composite Data Element are empty. Only the required delimiters will be written.
 - an optional Data Element then the Data Element will not be written to the output document. A delimiter will be written if required by the EDI Standard.

If the value is “N” on a Data Element, then the Data Element will be written to the output document even when it contains no value. This means that at a minimum the terminator for the Data Element

preceding the Data Element with the “N” value for its SuppressEmptyElements element attribute will be written.

- **Data Format** – The default setting for this element attribute is “N” (do not suppress empty elements). If the SuppressEmptyElements element attribute is specified on a Data Format Record with a value of “Y” then the Data Format Record will not be written to the output document if no Fields within the Record contain a value. If the SuppressEmptyElements element attribute is specified on a Data Format Loop with a value of “Y”, then each the Data Format Record within the Loop will not be written to the output document unless one or more Fields within the Record contain a value. This assumes that no Data Format Records within the Loop the SuppressEmptyElements element attribute have their value set to “N”.

The SuppressEmptyElements element attribute has no effect for a fixed position Data Format Field. It will affect the output of a Field in a comma separated values Data Format output file.

Alter the default formatting behavior of an element using the SetElementAttribute() command. The default value for an element attribute can be changed for the entire target document by specifying the path as “TargetRoot”. Specifying a compound element as the path on the SetElementAttribute() command causes the default value for the specified element attribute to be changed for all elements within the compound element. Elements not within the compound element will not be affected by the SetElementAttribute() command. Finally, specify the simple element itself in the path of the SetElementAttribute() command to just change the formatting behavior for a specific element. If you map is a target based map you can specify the keyword “This” as the path to indicate the command applies to the current element (the element that is the parent node of the command).

It is important to use as few SetElementAttribute() commands as you can to conserve computer memory. If most elements need to have their default formatting behavior overridden, then specify the SetElementAttributes() command once using “TargetRoot” as the path. If only a few elements need the default behavior changed then put specific SetElementAttribute() commands in just for those few elements.

Examples:

The following command alters the setting for the SuppressEmptyElements element attribute for the LOOPX02 compound element within the target document. The command indicates that the LOOPX02 compound element will not be written to the output document if all simple elements within it contain no significant value.

```
SetElementAttribute ( \DATHDR_LOOP2\LOOPX02\, "SuppressEmptyElements", "Y")
```

- **XML** – For the following formatting examples, assume XML element “M803” contains an attribute element named “desc” and an element containing the value associated with the “M803” element. It is assumed that the SuppressEmptyElements element attribute is set to “N” for the “M803” element.

Example		Attribute element	Value element	M803	Output Result
1	Element value	“ ”	“ ”	-	<M803 desc=“ ”> </M803>
	SuppressEmptyElement	N	N	N	
2	Element value	“ ”	“ ”	-	<M803 desc=“ ”/>
	SuppressEmptyElement	N	Y	N	
3	Element value	“ ”	“ ”	-	<M803> </M803>
	SuppressEmptyElement	Y	N	N	
4	Element value	“ ”	“ ”	-	<M803/>
	SuppressEmptyElement	Y	Y	N	
6	Element value	“ABC”	“ ”	-	<M803 desc=“ABC”/>
	SuppressEmptyElement	Y	Y	N	
7	Element value	“ ”	“DEF”	-	<M803>DEF</M803>
	SuppressEmptyElement	Y	Y	N	
7	Element value	“ABC”	“ ”	-	<M803 desc=“ABC”/>
	SuppressEmptyElement	Y	Y	Y	
8	Element value	“ ”	“ ”	-	”M803” XML element is not written.
	SuppressEmptyElement	Y	Y	Y	

- **Data Format** – If a Record contains four fields, RECORDID, TYPE, DESCRIPTION, and QUANTITY with the values “RECID”, “ABC”, “” and 0 respectively, then the comma separated values record will appear as follows when the SuppressEmptyElements element attribute is set to “N” for the third Field:

“RECID”,“ABC”,”,0

The record will appear as follows when the SuppressZeroValues element attribute is set to “Y” for the third Field:

“RECID”,“ABC”,,0

If the suppressed empty value is the last Field in the Data Format Record then the terminating comma for the last Field will not be written.