

HL SEGMENT CONTENT

Request

The following text was extracted from the request for interpretation:

“For a long time, ASC X12 has supported Hierarchical level transactions using the BHT and HL segments. Recently, my company has had difficulties with error reporting for received transactions that utilize hierarchical levels, and have errors in the HL segment. While attempting to find a solution, we have received multiple opinions from knowledgeable individuals. Some see the HL content as syntax, and needing to be rejected using a Functional Acknowledgement. Others see it as semantics.”

Background

Within hierarchical transactions, the BHT and HL segments define the appropriate structure of the transaction, as well as the general ASC X12 syntax for the transaction. Generally, the only aspects of syntax checking for an element of a segment would be minimum/maximum checks, character set verification, internal code validation and possibly date/time structure validation, however, there are aspects of the elements in an HL segment that are structural in nature. For other segments, these elements would always be evaluated as semantic aspects of the transaction, and not be reportable in a Functional Acknowledgement (997) transaction. In the HL segment, however, the elements take on a structural importance.

Example 1 – Two HL segments within a transaction use the same value in HL01. This is logically equivalent to exceeding the max use on a loop or segment. If that same number is identified as the parent ID (HL02) in a subsequent HL segment, the logical structure is completely lost.

Example 2 – An HL02 Hierarchical Parent ID Number does not exist within another HL segment as the HL01 Hierarchical ID Number. This would make the processing of the data as impossible as if one or more loops were missing in a nested loop structure.

Example 3 - An HL03 Hierarchical Level Code value that is a valid ASC X12 code, but is not part of the structure defined in the BHT01 structure for a specific inbound transaction is used. This would make the transaction as unreadable by an application as if an undefined segment/loop were to be received in a non-hierarchical level transaction.

Example 4 – An HL04 Hierarchical Child Code indicates there is a child (code 1), but there is no related HL segment identifying itself as the child of that prior HL segment. This would be equivalent to receiving a loop with a missing mandatory nested loop.

So, while a normal segment’s content would always be considered to be of a semantic nature, the HL segment content is critical to the structure (the syntax) of the transaction. A bad structure can’t be processed by the receiver.

Questions:

- 1 - How should the HL segment contents be evaluated – semantically or syntactically?
- 2 - What transactions and coding should be used to report errors in the content of the HL segment?

Referenced Standards

A “Request for Interpretation” applies to a specific version of the X12 Standards. The author failed to provide a specific version of the standard in the request. We have chosen to base this response on Version 5 Release 1 of the X12 Standard. As the areas of the X12 Standards applicable to this interpretation have been relatively stable over time, it is likely that the same interpretation would be provided for earlier versions of the X12 Standards.

X12.6 Application Control Structure (Version 005 Release 001)
Defines X12 Syntax

X12.22 Segment Directory (Version 005 Release 001)
Defines referenced HL and BHT segments

X12.3 Data Element Dictionary (Version 005 Release 001)
Defines data elements referenced in HL and BHT segments

X12.1 X12 Transaction Sets (Version 005 Release 001)
Defines TS 997 Functional Transaction Set, and other transaction sets of relevance to this interpretation.

Formal Interpretation

X12.6 defines the syntax of X12 Transaction Sets. X12.1 Transaction Set 997 Functional Acknowledgment reports on compliance with the syntax of a Functional Group of Transaction Sets. From an X12.6 perspective, and hence also from the perspective of X12 Transaction Set 997 Functional Acknowledgment, the two segments referenced in the formal request for interpretation lie outside the scope of X12.6. Hence, TS997 may only report on syntactic errors in constructing these two segments, not on semantic errors regarding their content.

Informal Discussion

Simple compliance with X12.6 syntax does not provide a guarantee that a Transaction Set is semantically complete and useful. In the cited examples, the transaction set instances are not valid X12 transaction set instances, as they violate semantic constraints imposed within the standard. Such semantic constraints are not restricted to the two referenced segments, but indeed occur elsewhere within the standard.

1 - How should the HL segment contents be evaluated – semantically or syntactically?

Both syntactic and semantic evaluation is essential to the processing of X12 transaction set instances.

2 - What transactions and coding should be used to report errors in the content of the HL segment?

Transaction Set 997 Functional Acknowledgment may be used to report syntactic errors in the content of the HL segment. (e.g., a mandatory field is missing data)

Transaction Set 824 Application Advice may be used to report semantic errors in the content of the HL segment.

Additional constraints of an industry or application nature may also render a transaction set instance unusable, even though the instance may be wholly compliant with the X12 Standard.

That the logical effect of the cited examples mirrors structural syntax violations does not qualify such violations as reportable using the TS 997 Functional Acknowledgment Transaction Set. A new Transaction Set TS 999 Implementation Acknowledgment has been recently introduced in the X12 Standards. It is intended to provide an alternative to TS 824 in reporting the types of semantic errors you provide in your examples. No implementation instruction has been provided to aid in using TS 999 to report the specific HL error scenarios provided in your examples. The X12 Communication and Controls Subcommittee (X12C) is currently developing implementation advice in an appropriate X12 publication.