Thunder Version 5 Release Notes

The following document applies to the version 5 installation for XML Thunder, JSON Thunder and Thunder Suite.

Overview

Starting with Version 5, there are now three versions of the Thunder toolset available:

XML Thunder

- Functionality is the same as XML Thunder version 4. This toolset is used for generating COBOL and/or C code for reading and writing XML documents.
- Current XML Thunder V4 users will be upgraded to XML Thunder v5.

JSON Thunder (new**)**

- The equivalent of XML Thunder except that support is for JSON instead of XML.

Thunder Suite

- Previously XML Thunder for JSON. The toolset supports both XML and JSON. Functionality of XML Thunder and JSON Thunder is combined.
- Current "XML Thunder with JSON" customers will be upgraded to Thunder Suite.

This document covers:

- Thunder Version 5 Updates
- Thunder Version 5 New Features
- Installing Thunder Version 5

Thunder Version 5 Updates

There are over 30 enhancements and fixes in the Version 5 release. Major enhancements to be aware of include:

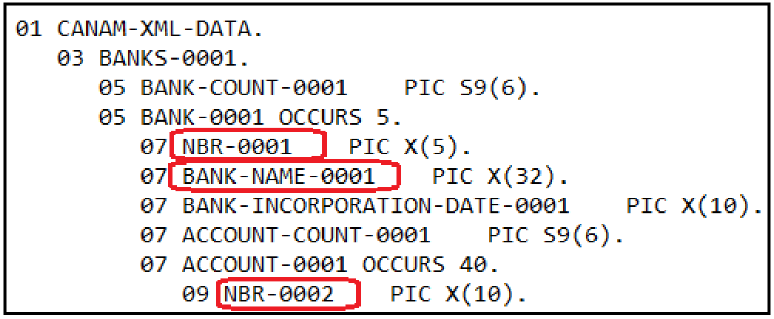
- A new “occurence2” IDS suffix setting (see below for details).
- A performance improvement for node-level Handlers in cases where entire XML document is loaded into XML-BUFFER before first call (see below for details).
- The ability to create a JSON Handler Design from a sample message.
- Suppport for JSON schema version 4.
- The ability to generate a JSON schema from a Handler Design (for cases when Handlers are created from copybooks).

New Occurrence2 Suffix Setting

A new suffix setting – “occurrence2” - has been added. It works identical to the “occurrence” setting with the exception that a suffix **will not** be applied to the first occurrence of field names.

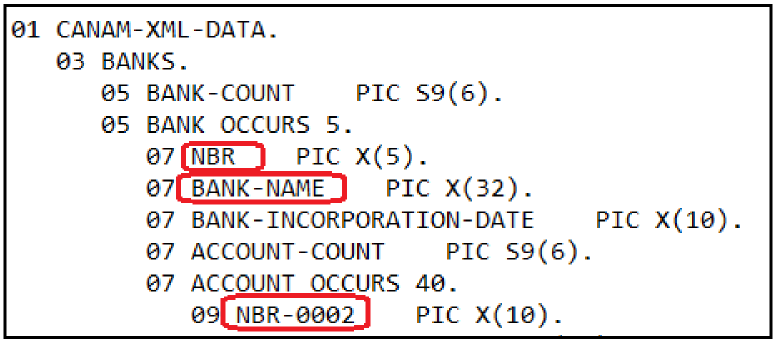
Field names generated with the suffix setting of “occurrence” aways get a “-000?” suffix where “?” indicates the occurrence instance of the field (e.g. first, second, third, etc.).

Example:



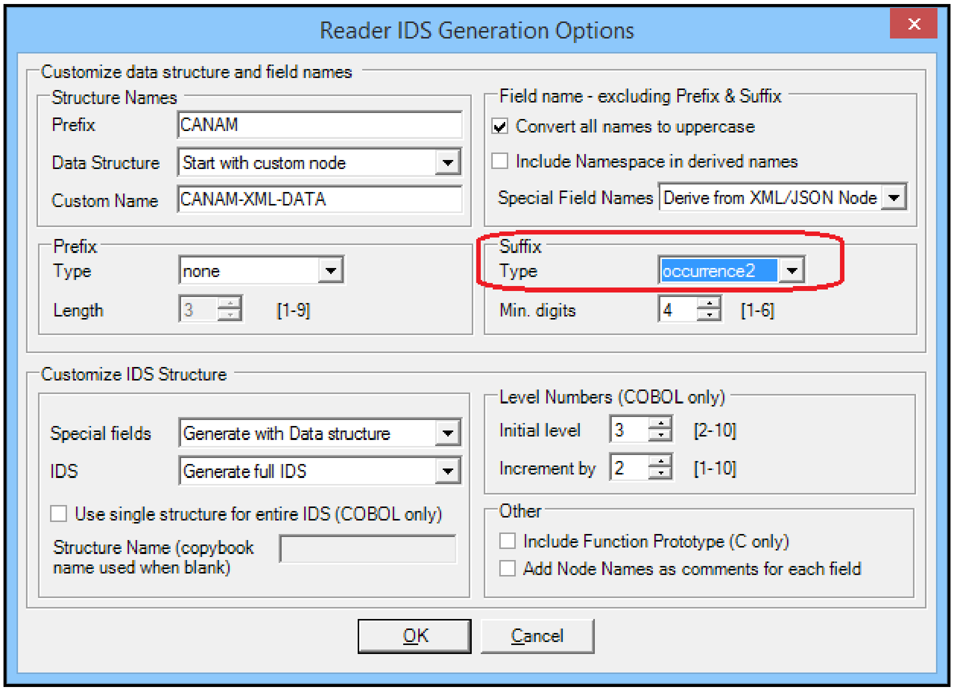
```
01 CANAM-XML-DATA.
   03 BANKS-0001.
      05 BANK-COUNT-0001    PIC S9(6).
      05 BANK-0001 OCCURS 5.
         07 NBR-0001     PIC X(5).
         07 BANK-NAME-0001     PIC X(32).
         07 BANK-INCORPORATION-DATE-0001    PIC X(10).
         07 ACCOUNT-COUNT-0001    PIC S9(6).
         07 ACCOUNT-0001 OCCURS 40.
            09 NBR-0002     PIC X(10).
```

Field names generated with the suffix setting of “occurrence2” will not have a suffix for the first occurrence. All subsequent occurrences of the Field name will get a suffix (to ensure unique names). Note that COBOL reserved words will also get a suffix to avoid compiler errors (e.g. Id will be generated as Id-0001).



```
01 CANAM-XML-DATA.
   03 BANKS.
      05 BANK-COUNT    PIC S9(6).
      05 BANK OCCURS 5.
         07 NBR     PIC X(5).
         07 BANK-NAME     PIC X(32).
         07 BANK-INCORPORATION-DATE    PIC X(10).
         07 ACCOUNT-COUNT    PIC S9(6).
         07 ACCOUNT OCCURS 40.
            09 NBR-0002     PIC X(10).
```

The suffix setting is established as a toolset default under Tools -> Default Properties -> Code Generation -> Reader IDS Settings / Writer IDS Settings. If can also be chnaged for existing Handlers by selecting “More Options” next to the copybook name on the Code Generation panel.



Reader IDS Generation Options

Customize data structure and field names

Structure Names

Prefix: CANAM

Data Structure: Start with custom node

Custom Name: CANAM-XML-DATA

Field name - excluding Prefix & Suffix

☑ Convert all names to uppercase

☐ Include Namespace in derived names

Special Field Names: Derive from XML/JSON Node

Prefix

Type: none

Length: 3 [1-9]

Suffix

Type: occurrence2

Min. digits: 4 [1-6]

Customize IDS Structure

Special fields: Generate with Data structure

IDS: Generate full IDS

☐ Use single structure for entire IDS (COBOL only)

Structure Name (copybook name used when blank):

Level Numbers (COBOL only)

Initial level: 3 [2-10]

Increment by: 2 [1-10]

Other

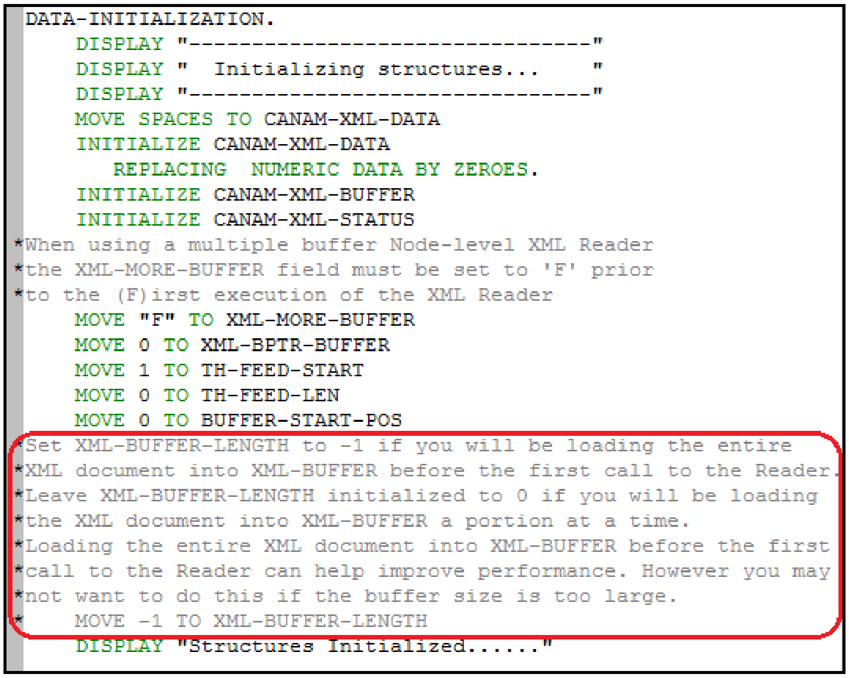
☐ Include Function Prototype (C only)

☐ Add Node Names as comments for each field

Performance Improvement for Node-Level Handlers

In some cases, it is possible to load the entire XML document into the XML-BUFFER field prior to the first call to a node-level Reader. When doing this, you can now improve performance by initializing XML-BUFFER-LENGTH to -1. The XML-BUFFER-LENGTH was not previoulsy used by node-level Readers. Setting it to -1 will now tell the Reader that the XML-BUFFER field already holds the entire XML document. When this is the case, the logic to strip off the XML portion that was read and then shift all unread XML to the beginning of XML-BUFFER can be bypassed inside the Reader. There is no need to open up space on the end of XML-BUFFER since the entire XML document has already been loaded. Bypassing this logic can significanly improve performace.

Comments have been added to the node-level Reader Test Harness to indicate the new feature. By default the XML-BUFFER-LENGTH field **is not** initialized to -1.



```
 DATA-INITIALIZATION.
     DISPLAY "--------------------------------"
     DISPLAY "  Initializing structures...    "
     DISPLAY "--------------------------------"
     MOVE SPACES TO CANAM-XML-DATA
     INITIALIZE CANAM-XML-DATA
        REPLACING  NUMERIC DATA BY ZEROES.
     INITIALIZE CANAM-XML-BUFFER
     INITIALIZE CANAM-XML-STATUS
*When using a multiple buffer Node-level XML Reader
*the XML-MORE-BUFFER field must be set to 'F' prior
*to the (F)irst execution of the XML Reader
     MOVE "F" TO XML-MORE-BUFFER
     MOVE 0 TO XML-BPTR-BUFFER
     MOVE 1 TO TH-FEED-START
     MOVE 0 TO TH-FEED-LEN
     MOVE 0 TO BUFFER-START-POS
*Set XML-BUFFER-LENGTH to -1 if you will be loading the entire
*XML document into XML-BUFFER before the first call to the Reader.
*Leave XML-BUFFER-LENGTH initialized to 0 if you will be loading
*the XML document into XML-BUFFER a portion at a time.
*Loading the entire XML document into XML-BUFFER before the first
*call to the Reader can help improve performance. However you may
*not want to do this if the buffer size is too large.
*    MOVE -1 TO XML-BUFFER-LENGTH
     DISPLAY "Structures Initialized......"
```

The following is the complete list of changes introduced with version 5:

Fixes:

1. Sample JSON message should not enclose boolean values in quotes
2. Fix COBOL Reader index error when element has default value under repeating structure
3. Fix Invalid character error raised for arrays of numeric data in JSON
4. Fix COBOL compiler error when line wraps in middle of back to back quotes
5. Do not add 0 and 1 and enumeration for JSON data type of boolean
6. Remove Windows API calls when possible
7. Physical IDS tab not showing flag fields in correct position for Locked and NO-SYNC Handlers
8. Do not create duplicate enumeration values on union types
9. Fix issue loading WSDLs with namespaces declared at element level
10. Modify concurrent user security checking to test either Host Name or IP address. If either matches then allow access.
11. For oneOf we are adding separators after a choice property - when only one choice is output.
12. COBOL copybook parser fails when VALUE clause contains period.
13. Copybook with 01 level that also hold data does not load correctly
14. Do not include value of "0" when validating totalDigit count.
15. IDS property changes updating wrong node is certain cases.

Enhancements:

1. Updated comments at beginning of source code to include more Handler Settings.
2. Add ability to generate JSON Schema from COBOL copybook
3. Add support for 2 digit character references
4. Add support for JSON Schema draft 4
5. Add support for min sequence
6. Do not allow min Occurs to be greater than max Occurs for choice and sequence
7. Add support for oneOf, anyOf and AllOf for JSON
8. JSON Schema – add support for named root objects
9. Add support for Test Harness to generate inline sample JSON message for testing.
10. Add support for Reader to parse processing instruction attributes (e.g. version and encoding)
11. Provide JSON only product
12. Remember last folder location used when saving schemas and samples
13. Add ability for COBOL node-level Handler to receive entire XML document.
14. Add new "occrrence2" IDS suffix setting. Suffix only applied from second occurrence on.
15. Recognize "import XML namespace = OFF" setting for Handlers created from WSDL.
16. Add "truncate without error" setting for text fields
17. Allow Trim, "Trim Start" and "Trim End" to work on CDATA nodes

Thunder Version 5 Overview

In addition to the significant fixes and enhancements in this release, the following major changes are introduced with version 5.

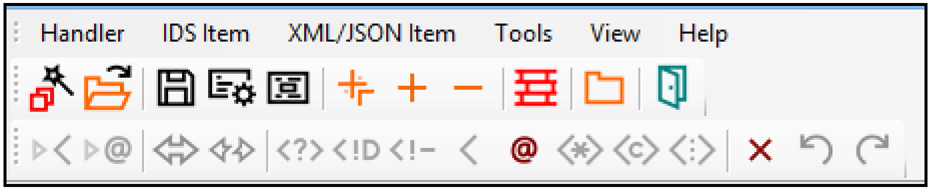
Move to .Net Framework 4

The toolset now runs under the Microsoft .Net 4 framework (previously the 3.5 framework was used). The .Net 4 framework is standard on most workstations.

New Icons and Image

The toolset images and icons have been updated. All functionality associated with the icons remains the same.

Example:



JSON-only Support Now Available

The Thunder toolset is now available in three versions:

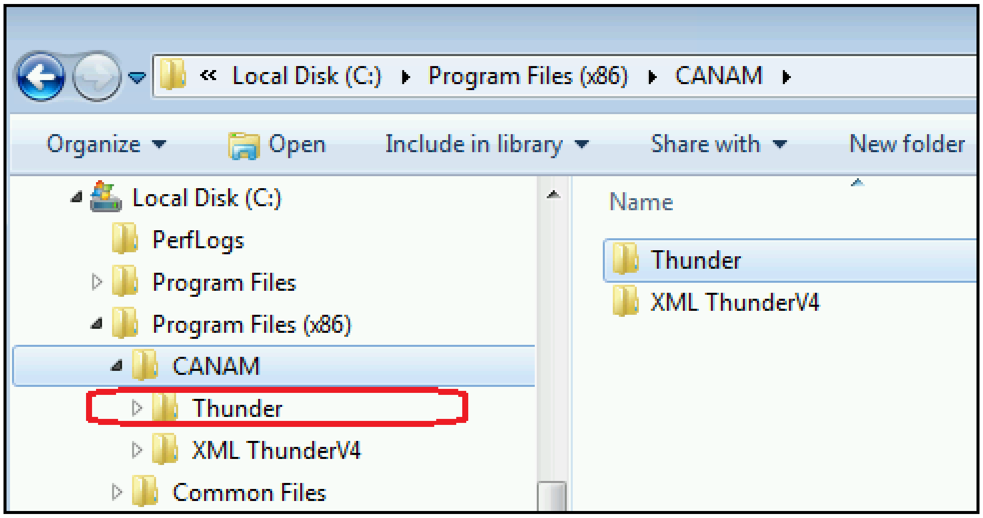
- XML Thunder (same functionality as XML Thunder V4)
- JSON Thunder (JSON only support **new**)
- Thunder Suite (same functionality as XML Thunder V4 with JSON)

Default Installation Folder

The default installation folder for version 5 has been changed to C:\Program Files (x86)\CANAM\Thunder (or Thunder NS). Previously the default installation folder was C:\Program Files (x86)\CANAM\XML ThunderV4 (or XML Thunder NSV4).

Installing Thunder Version 5

Thunder version 5 can co-exist on the same machine as XML Thunder version 4. You can have both versions of the toolset active; however, you most likely will eventually want to uninstall Version 4. When you install version 5, it will be placed in a new installation folder by default (e.g. c:\program files (x86)\Canam\Thunder).



IMPORTANT.

Handlers saved under XML Thunder version 5 cannot be opened in version 4. If you need to open a version 5 Handler in version 4, please contact support for assistance in doing this.

Uninstalling XML Thunder Version 4

There are two considerations when uninstalling XML Thunder V4.

1. **License Location**
   In most cases the toolset license files (TX1E.lic and TX2E.lic) will be located on a different machine (i.e. a server that allows the license files to be accessed by multiple workstations). However, during trials the license files may be located on the same workstation as the toolset. In this case, be sure to copy the TX1E.lic and TX2E.lic files to a new location on the workstation before uninstalling version 4 as the uninstall will delete the license files.

2. **Files Under Version 4 Installation Folder**
   Uninstalling version 4 will delete the installation folder from the workstation (e.g. C:\Program Files (x86)\CANAM\XML ThunderV4). If you have any files under this folder that you would like to keep (e.g. Handler Definitions), then you should move them to a new location before uninstalling.

If you have any questions or would like more information, please contact us at support@canamsoftware.com.