# BioCTS for ANSI/NIST-ITL v2

# User Guide: Version 1.2

**NIST/ITL CSD Biometric Conformance Test Software for ANSI/NIST-ITL Version 2 - Supporting: ANSI/NIST-ITL 1-2011 and ANSI/NIST-ITL 1-2011 Update: 2013 Traditional and NIEM XML Encoding**

**September 2016**

Christofer J. McGinnis
ID Technology Partners (NIST Associate) Software Developer

Dylan Yaga
NIST/ITL CSD Lead Software Designer

**National Institute of Standards and Technology (NIST)**

**Information Technology Laboratory (ITL)**

**Computer Security Division (CSD)**

## Contents

## Figures

# 1. Disclaimer

**NIST/ITL BioCTS**

**For ANSI/NIST-ITL**

**October 2010**

## 2. User Guide Version History

As updates and corrections are made to this user guide, new versions may be released. The following table summarizes the changes made during each revision.

| Version | Date | Summary |
|---|---|---|
| 1.0 | August 2015 | Initial release of the User Guide |
| 1.1 | October 2015 | -Correction to the Record Types supported. For the AN-2013 CTS: Removed support for Type 13, 14, 15, and 17 for XML. For AN-2011 and AN-2013, added support for Type 2.<br>-Updated Annex A: Schema modifications to reflect changes to the modified Schema.<br>-Updated references to GUI functionality due to slight changes |
| 1.2 | September 2016 | -Reviewed and modified descriptions of the tools functionality to reflect changes made to the latest version of BioCTS for AN-ITL v2 released in September 2016.<br>-Added clarification text<br>-Minor corrections to GUI images and other supporting text.<br>-Added page numbers. |

Table 1 – User Guide Version History Summary

# 3. Overview

This document describes the features available in the latest release of Biometric Conformance Test Software (BioCTS) for ANSI/NIST-ITL Version 2 (AN-ITL v2) developed by NIST/ITL Computer Security Division, and will be updated as minor revisions to the software are released. Note that in this document "Version 2" and "BioCTS for AN-ITL v2" describe any BioCTS software released using the major Version 2; all subsequent minor releases and updates are also included in this description (e.g. Beta 2.x.xxxx.xxxxx). The last major BioCTS version released prior to BioCTS for AN-ITL v2 was Beta 1.2.5353.15785 on September 10, 2014. For an overview of the changes made since the previous major version, see Updates from Previous Major Version.

BioCTS for AN-ITL v2 is a desktop application which tests electronic biometric data files, known as transactions, for conformance to *NIST Special Publication (SP) 500-290 Data Format for the Interchange of Fingerprint, Facial & Other Biometric Information* [1], and *NIST Special Publication 500-290 Rev1 (2013) Data Format for the Interchange of Fingerprint, Facial & Other Biometric Information* [2]. ANSI/NIST-ITL 1-2011 (AN-2011), ANSI/NIST-ITL 1-2011 Update: 2013 (AN-2013), and related information on the standard can be found at the ANSI/NIST-ITL Standard Homepage [3].

BioCTS for AN-ITL v2 contains many useful features for testing transactions individually or as groups, viewing and analyzing conformance errors and test results, and modifying transaction data to correct conformance errors. Some examples of the software's functionality include:

- Testing a significant number of transactions (tested at 1000+) in a single batch test
- Evaluating the batch and individual transaction results using visual and textual statistical analyses
- Loading transactions into the BioCTS Editor (Traditional Encoding only), which provides tools for viewing, expanding, and modifying the Records, Fields, Subfields, and Information Items which comprise a transaction

Test Results from each transaction are displayed on-screen and are saved to a user-specified folder with a time stamp. The user can access the Test Results files or the transaction source files by right clicking on any transaction loaded in the Batch File List and selecting the appropriate option from the context menu.

## 3.1. Conformance Testing Support

All record types defined in AN-2011 and AN-2013 are supported for conformance testing of Traditional Encoded Transactions, and selected record types are supported in NIEM XML Encoded Transactions.

BioCTS for AN-ITL v2 supports four Conformance Test Suites (CTS) shown below. For each CTS, the supported assertions and record types are also listed.

**CTSs for AN-2011**

The CTSs for AN-2011 implement all test assertions specified in *NIST SP 500-295 Revision 1 - Conformance Testing Methodology for ANSI/NIST-ITL 1-2011, Data Format for the Interchange of Fingerprint, Facial & Other Biometric Information* [4], plus additional assertions developed to address record types not supported in previous versions of BioCTS for AN-ITL.

- **CTS for AN-2011 Traditional Encoded Transactions**
  - Full Support (Transaction Level and All record types defined in AN-2011)
- **CTS for AN-2011 NIEM XML Encoded Transactions**
  - Transaction Level Assertions
  - Type 1, Transaction Information Record
  - Type 2, User-defined descriptive text record
  - Type 4, High-resolution grayscale fingerprint image
  - Type 10, Facial and SMT image
  - Type 13, Variable-resolution latent image
  - Type 14, Variable-resolution fingerprint image
  - Type 15, Palm print image
  - Type 17, Iris image

**CTSs for AN-2011 Update: 2013**

The CTSs for AN-2011 Update: 2013 implement the test assertions specified in *NIST Special Publication NIST SP 500-304 – Conformance Testing Methodology Framework for ANSI/NIST-ITL 1-2011 Update: 2013, Data Format for the Interchange of Fingerprint, Facial & Other Biometric Information* [5] as well as additional required test assertions specific to the 2013 Update that were not published at the time of writing.

- **CTS for AN-2011 Update: 2013 Traditional Encoded Transactions**
  - Full Support (Transaction Level and All record types defined in AN-2011 Update: 2013)
- **CTS for AN-2011 Update: 2013 NIEM XML Encoded Transactions**
  - Transaction Level Assertions
  - Type 1, Transaction information record
  - Type 2, User-defined descriptive text record
  - Type 4, High-resolution grayscale fingerprint image
  - Type 10, Facial and SMT image
  - Type 12, Forensic dental and oral data
  - Type 22, Non-photographic imagery

Note that Type-2 support in the above CTS's does not include support for specific profiles nor any user-defined fields. Future releases may add support for additional record types.

## 3.2. Requirements

- Supported Microsoft® Operating Systems
  - Windows® XP ™ Service Pack 3

    - Windows® Vista ™ Service Pack 2
    - Windows® 7 ™ Service Pack 1
    - Windows® 8.1 ™
    - Windows® 10 ™
- Microsoft® .NET 4.0 Framework
    - Microsoft® .NET 4.0 Web Installer (http://www.microsoft.com/en-us/download/details.aspx?id=17851)
    - Microsoft® .NET 4.0 Stand Alone Installer (http://www.microsoft.com/en-us/download/details.aspx?id=17718)

Links working as of 9/13/2016

## 4. Updates from Previous Major Version

For users already familiar with the previous major version of BioCTS for AN-ITL, this section provides an overview of changes in functionality and conformance support. All changes in this section are in reference to the previously released major version (Beta 1.2.5353.15785, September 10, 2014). The most notable change from the previous version of the software is the addition of full Traditional Encoding support for all record types defined in AN-2011 and AN-2013. Additionally, several new features have been added to the Editor to allow greater control over modifications to the Record, Field, Subfield, and Information Item structure and organization. The parser has also been improved to provide more meaningful feedback when transactions are malformed or when records are of incorrect length.

Any minor changes not related to adding conformance support or improving functionality and features are not listed in this document. Minor modifications to software are listed on the BioCTS website as a changelog that is updated over time.

### 4.1. Updates to Conformance Support

The information in this section lists only the conformance support that has been added since the previous major release.

**AN-2011 Traditional Encoded Transactions**

BioCTS for AN-ITL v2 fully supports all record types defined in AN-2011. Support for the following record types was added since the previous major version:

- Type 2, User-defined descriptive text*
- Type 7, User-defined image
- Type 8, Signature image
- Type 9, Minutiae data
- Type 16, User-defined variable-resolution testing image
- Type 18, DNA data
- Type 19, Variable-resolution plantar image
- Type 20, Source representation
- Type 21, Associated context
- Type 22, Non-photographic imagery
- Type 98, Information assurance
- Type 99, CBEFF biometric data record

**AN-2011 NIEM XML Encoded Transactions**

No changes were made to the supported Record Types since the previous version.

Annex A of this document provides details regarding the modifications that had to be made to the XML Schema provided with ANSI/NIST-ITL 1-2011.

**AN-2013 Traditional Encoded Transactions**

BioCTS for AN-ITL v2 fully supports all record types defined in AN-2013. Support for the following record types was added since the previous version:

- Type 2, User-defined descriptive text*
- Type 7, User-defined image
- Type 8, Signature image
- Type 9, Minutiae data
- Type 11, Forensic and investigatory voice data
- Type 12, Forensic dental and oral data
- Type 13, Variable-resolution latent friction ridge image
- Type 14, Variable-resolution fingerprint image
- Type 15, Variable-resolution palm print image
- Type 16, User-defined variable-resolution testing image
- Type 17, Iris Image
- Type 18, DNA data
- Type 19, Variable-resolution plantar image
- Type 20, Source representation
- Type 21, Associated context
- Type 22, Non-photographic imagery
- Type 98, Information assurance
- Type 99, CBEFF biometric data record

**AN-2013 NIEM XML Encoded Transactions**

Support for the following record types was added since the previous version:

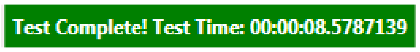
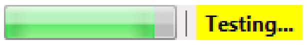
- Type 12, Forensic dental and oral data
- Type 22, Non-photographic imagery

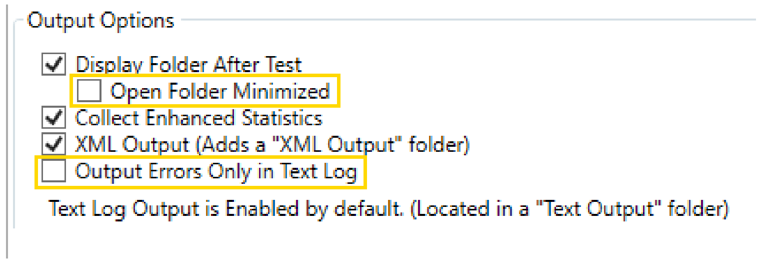
*Note that Type-2 support in the above CTS's does not include support for specific profiles nor any user-defined fields.

## 4.2. Updates to Features and Functionality

There are several new features included in the BioCTS User Interface since the last release:

- **Usability Updates** – Several improvements have been made to the tool's usability and workflow. Some example include:
    - Removal of pop-up prompts after testing and various actions. The user interface now displays a label at the bottom right of the window to show messages that do not interrupt the user's progress. For example:

        - Testing Complete: 
Test Complete! Test Time: 00:00:08.5787139
        - Tests In Progress: 
Testing...
    - Option to have output window minimized by default to avoid disrupting user as well as an option to output errors only in the text logs.
        - Shown highlighted in yellow:


Output Options

- [✓] Display Folder After Test
    - [ ] Open Folder Minimized
- [✓] Collect Enhanced Statistics
- [✓] XML Output (Adds a "XML Output" folder)
- [ ] Output Errors Only in Text Log

Text Log Output is Enabled by default. (Located in a "Text Output" folder)

    - Several usability improvements to the Editor (see below).
- **Advanced Editor Features** – Powerful new Editor features have been added to provide more robust editing and control over transaction data. For details on using these new features, refer to the Editor Guide. New features include:
    - Add Record
    - Remove Record
    - Sort Records (sort records numerically)
    - Arrange Records (reorder using up and down arrows)
    - Show Record Image
    - Sort By Field Number (sort fields numerically within a record)
    - Organize Fields (arrange using up and down arrows)
    - Add Binary Field - Revised (specify byte index for insertion)
    - Modify Subfields
    - View Data Pop-out Window
- **Parsing improvements** – The parsing methods in the previous version of the tool provided poor error messages in some cases by not providing enough detail, such as "Error: Data passed in but not parsed". Some of the parser improvements are listed below; for more detailed information on understanding errors when parsing, see Troubleshooting and Tips: Parsing.
    - The errors generated by the parser are more informative in the latest version. Parsing error messages now describe how much data was missing or how much extra data was present in relation to the Transaction Contents found in Field 1.003.
    - Previous versions of the tool would "short circuit" during parsing and not continue if any errors were detected during parsing. The latest version attempts to parse and report on as much data as possible, even if some errors are detected during parsing.
    - When modifying data in the Editor and re-running tests, the previous versions of the tool did not properly report errors at times because the data was not reparsed—it was instead created from the transaction saved in memory. The new version forces the data in memory to be reparsed just like a new transaction.

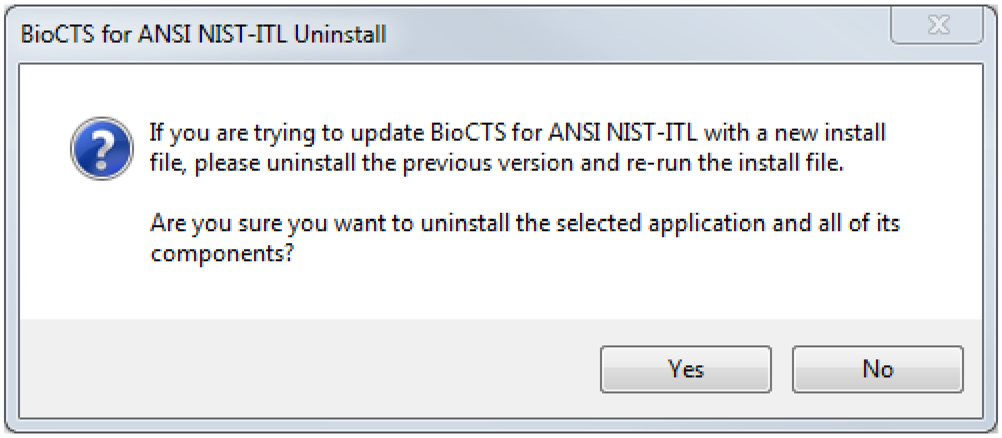
# 5. Guide

## 5.1. Download and Installation

Download the installer from the website http://www.nist.gov/itl/csd/biometrics/biocta_download.cfm .

After the download completes, run the install program and follow the on screen instructions presented in the dialog boxes.

If a previous version of BioCTS for AN-ITL v2 is installed on the machine when the installer is launched, it will prompt the user to uninstall the previous version and re-launch the installer:


BioCTS for ANSI NIST-ITL Uninstall

If you are trying to update BioCTS for ANSI NIST-ITL with a new install file, please uninstall the previous version and re-run the install file.

Are you sure you want to uninstall the selected application and all of its components?

**Figure 2 – Uninstall prompt**

To uninstall the previous version, select "Yes". After the previous version is uninstalled, the installer must be run again to install the new version.

## 5.2. Running the Conformance Test Architecture

To run the CTA software from the Start menu:
Select **All Programs** then select **NIST BioCTS**, then select **ANSI NIST ITL**. The application will be located in this directory for access after it is successfully installed.

After starting BioCTS for AN-ITL v2, specific Conformance Test Suites may be selected from the Options Tab (See Section 5.3.4 for more details). Four Conformance Test Suites are present, and are separated into two categories:

- Traditional Encoding Conformance Test Suites
    - ANSI/NIST-ITL 1-2011 Traditional Encoding Conformance Test Suite
    - ANSI/NIST-ITL 1-2011 Update: 2013 Traditional Encoding Conformance Test Suite
- NIEM XML Encoding Conformance Test Suites
    - ANSI/NIST-ITL 1-2011 NIEM XML Encoding Conformance Test Suite
    - ANSI/NIST-ITL 1-2011 Update: 2013 NIEM XML Encoding Conformance Test Suite

Within the Options Tab, a single Conformance Test Suite can be selected for each category.

## 5.3. Conformance Test Architecture Features

This section outlines features of the user interface that assist the end user in loading, testing, and analyzing transactions. Any images of the software are provided for illustration only and may not represent the final software release.

### 5.3.1. Traditional transaction Batch Testing

The “Traditional Transaction Batch Test” tab allows multiple traditionally encoded transactions (files) to be tested in groups, and displays the overall results for each transaction in the “Files Under Test” pane.

**Open Files:** loads files into the Batch File List and removes any previously loaded files

**Add Files:** loads files into the Batch File List by adding them after any previously loaded files[1]

**Test Files:** tests all files loaded in the Batch File List

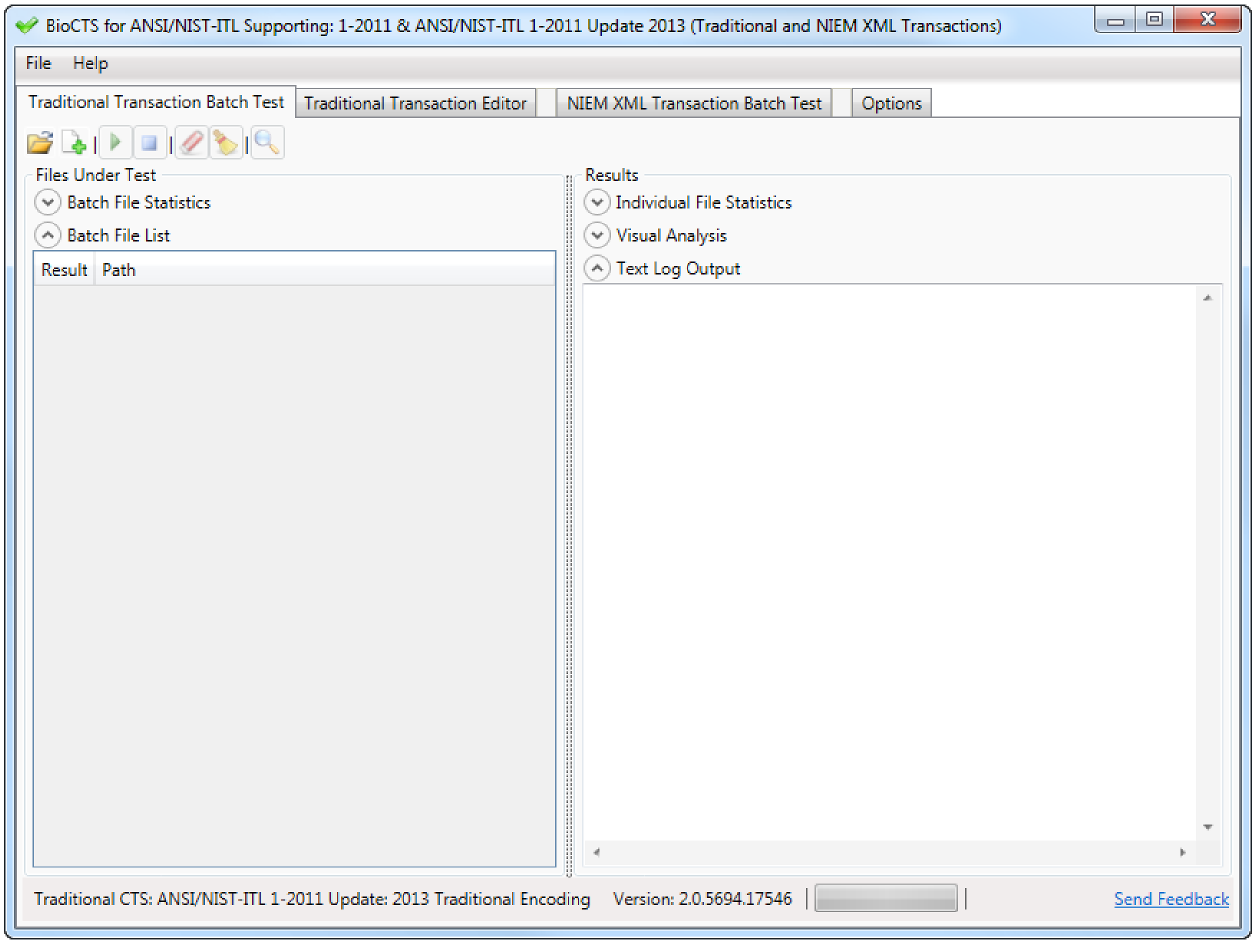
**Stop Testing:** immediately stops all testing of the files in the Batch File List

**Clear Results:** clears all testing results from the Batch File List (but does not remove the files)

**Clear Files:** clears all files from the Batch File List (and removes the test results)

**Find/Search:** Opens text-search window for searching test results (also Ctrl + f). The F3 key locates the next matching result. The Esc key closes the Find/Search Window.

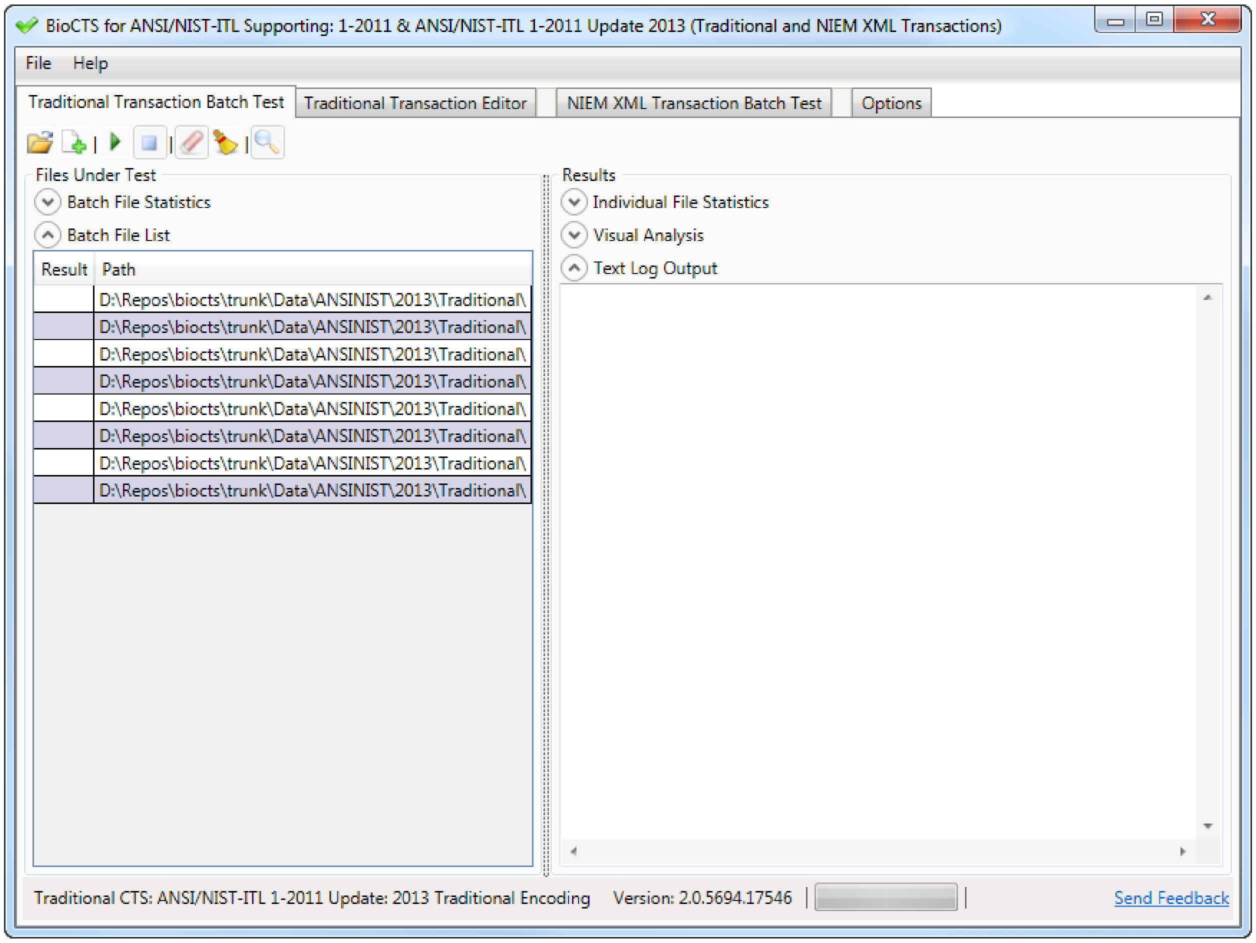
[1] Drag and drop is supported on the File Under Test pane of the Batch tabs, so that transaction files can be dropped into the batch list. Files dropped into the batch list are added to the existing list of files.


BioCTS for ANSI/NIST-ITL Supporting: 1-2011 & ANSI/NIST-ITL 1-2011 Update 2013 (Traditional and NIEM XML Transactions)
File
Help
Traditional Transaction Batch Test
Traditional Transaction Editor
NIEM XML Transaction Batch Test
Options
Files Under Test
Batch File Statistics
Batch File List
Result
Path
Results
Individual File Statistics
Visual Analysis
Text Log Output
Send Feedback

Traditional CTS: ANSI/NIST-ITL 1-2011 Update: 2013 Traditional Encoding Version: 2.0.5694.17546

**Figure 3 - Empty Batch Test Tab**

A Significant number of files (1000+) can be loaded at in a single batch. The more powerful the hardware used, the more files that can be tested in batch mode. BioCTS Team members perform the tests on conventional Desktop Computers (e.g., Intel® Xeon® Processors, 8 GB of Memory).


BioCTS for ANSI/NIST-ITL Supporting: 1-2011 & ANSI/NIST-ITL 1-2011 Update 2013 (Traditional and NIEM XML Transactions)
File
Help
Traditional Transaction Batch Test
Traditional Transaction Editor
NIEM XML Transaction Batch Test
Options
Files Under Test
Batch File Statistics
Batch File List
Results
Individual File Statistics
Visual Analysis
Text Log Output
Send Feedback

| Result | Path |
|---|---|
| | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\ |
| | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\ |
| | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\ |
| | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\ |
| | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\ |
| | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\ |
| | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\ |
| | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\ |

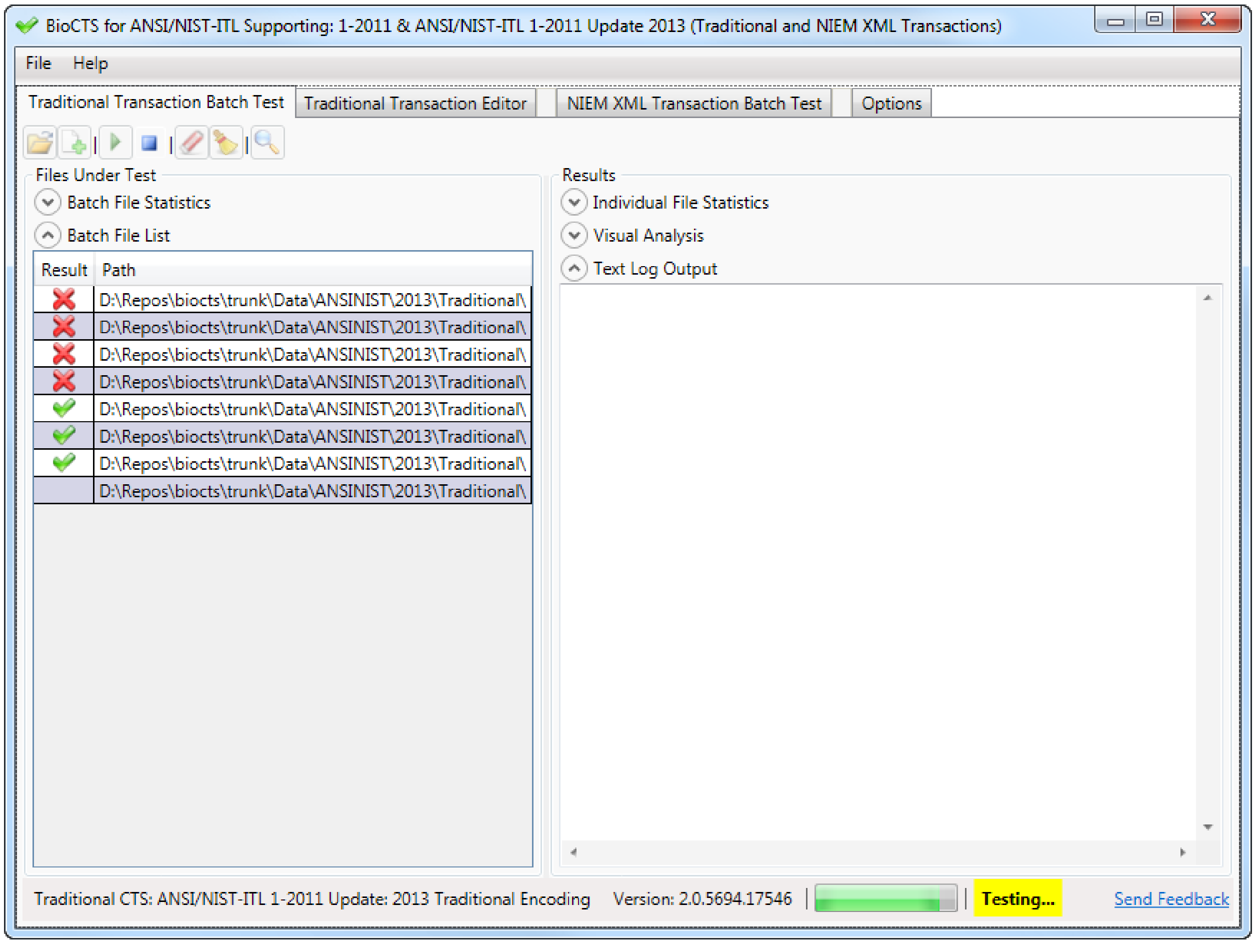
Traditional CTS: ANSI/NIST-ITL 1-2011 Update: 2013 Traditional Encoding    Version: 2.0.5694.17546

**Figure 4 - Files Loaded into the Batch Test Tab**

The “Batch Test” tab will display the transaction’s overall result with either:

- ✗ - Overall Result of Fail
- ✓ - Overall Result of Pass

The status banner at the bottom right side of the window displays messages regarding the tool’s current actions. While testing, the banner displays “Testing…” in yellow.


BioCTS for ANSI/NIST-ITL Supporting: 1-2011 & ANSI/NIST-ITL 1-2011 Update 2013 (Traditional and NIEM XML Transactions)
File
Help
Traditional Transaction Batch Test
Traditional Transaction Editor
NIEM XML Transaction Batch Test
Options
Files Under Test
Batch File Statistics
Batch File List
Results
Individual File Statistics
Visual Analysis
Text Log Output
Send Feedback

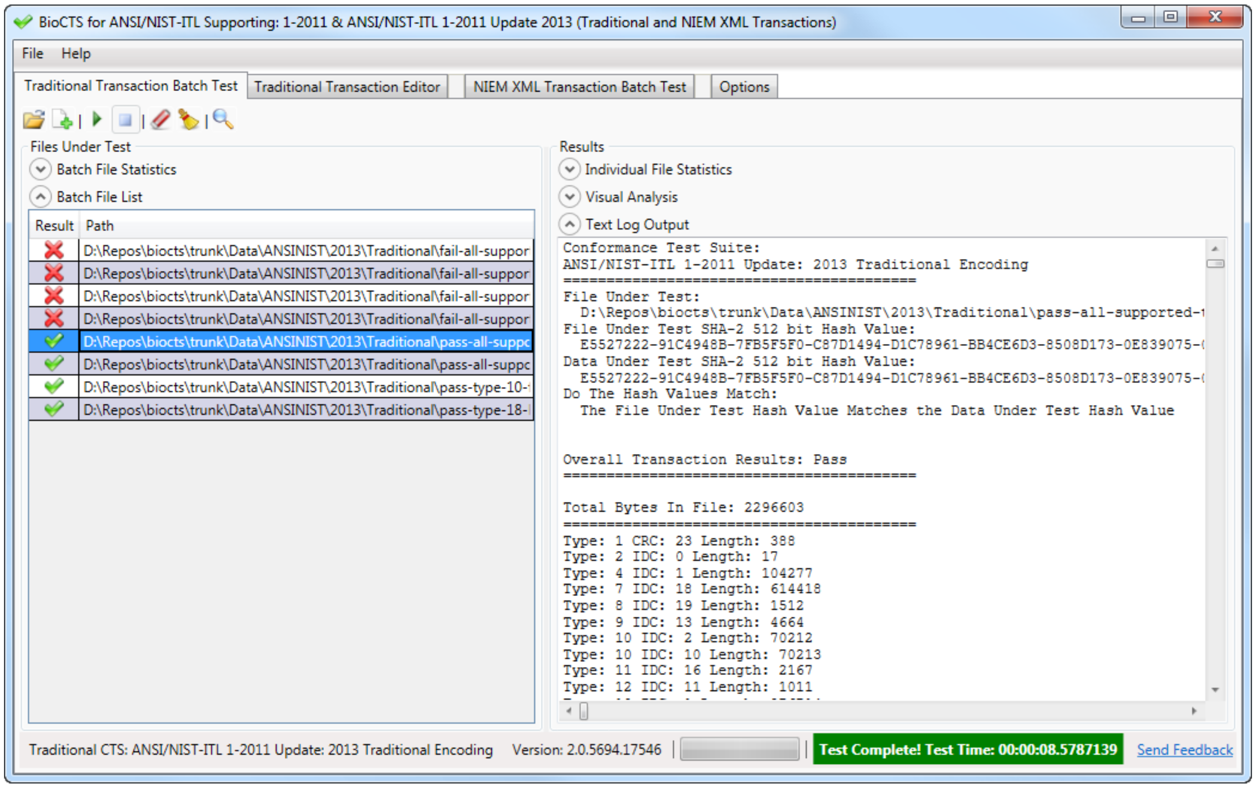
| Result | Path |
|---|---|
| ✗ | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\ |
| ✗ | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\ |
| ✗ | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\ |
| ✗ | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\ |
| ✓ | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\ |
| ✓ | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\ |
| ✓ | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\ |
|  | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\ |

Traditional CTS: ANSI/NIST-ITL 1-2011 Update: 2013 Traditional Encoding    Version: 2.0.5694.17546    Testing...

Figure 5 - Batch Test tab showing Overall Results

Once all files in the Batch File List have completed testing, the status banner at the bottom right side of the window displays "Test Complete! …" as well as the testing time in green.

Textual output results for each transaction can be viewed by clicking on the desired filename in the "File Under Test" pane. The complete textual results are displayed in the pane to the right.


BioCTS for ANSI/NIST-ITL Supporting: 1-2011 & ANSI/NIST-ITL 1-2011 Update 2013 (Traditional and NIEM XML Transactions)
File
Help
Traditional Transaction Batch Test
Traditional Transaction Editor
NIEM XML Transaction Batch Test
Options
Files Under Test
Batch File Statistics
Batch File List
Results
Individual File Statistics
Visual Analysis
Text Log Output
Send Feedback

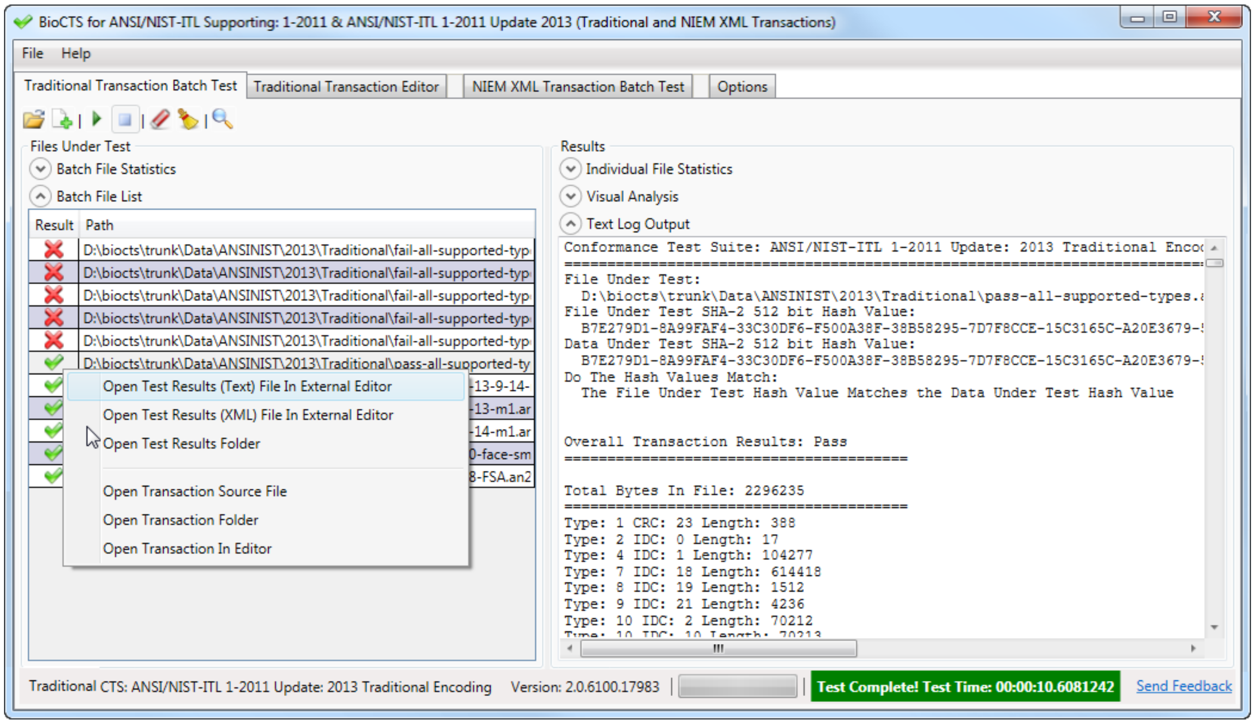
| Result | Path |
|---|---|
| ✗ | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\fail-all-suppor |
| ✗ | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\fail-all-suppor |
| ✗ | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\fail-all-suppor |
| ✗ | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\fail-all-suppor |
| ✓ | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\pass-all-suppo |
| ✓ | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\pass-all-suppo |
| ✓ | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\pass-type-10- |
| ✓ | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\pass-type-18- |

```
Conformance Test Suite:
ANSI/NIST-ITL 1-2011 Update: 2013 Traditional Encoding
=========================================
File Under Test:
  D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\pass-all-supported-
File Under Test SHA-2 512 bit Hash Value:
  E5527222-91C4948B-7FB5F5F0-C87D1494-D1C78961-BB4CE6D3-8508D173-0E839075-
Data Under Test SHA-2 512 bit Hash Value:
  E5527222-91C4948B-7FB5F5F0-C87D1494-D1C78961-BB4CE6D3-8508D173-0E839075-
Do The Hash Values Match:
  The File Under Test Hash Value Matches the Data Under Test Hash Value


Overall Transaction Results: Pass
=========================================

Total Bytes In File: 2296603
=========================================
Type: 1 CRC: 23 Length: 388
Type: 2 IDC: 0 Length: 17
Type: 4 IDC: 1 Length: 104277
Type: 7 IDC: 18 Length: 614418
Type: 8 IDC: 19 Length: 1512
Type: 9 IDC: 13 Length: 4664
Type: 10 IDC: 2 Length: 70212
Type: 10 IDC: 10 Length: 70213
Type: 11 IDC: 16 Length: 2167
Type: 12 IDC: 11 Length: 1011
```

Traditional CTS: ANSI/NIST-ITL 1-2011 Update: 2013 Traditional Encoding    Version: 2.0.5694.17546    Test Complete! Test Time: 00:00:08.5787139

Figure 6 - Batch Test Tab with a Transaction Selected and Results Displayed in Right Pane

A context menu is available when transaction files are loaded in the Batch File List found in the “File Under Test” pane. By right-clicking on any of the file names, a context menu is displayed with options to open the results files or the source data files. An option to open the transaction in the Traditional Transaction Editor is also available (Traditional encoding only).


BioCTS for ANSI/NIST-ITL Supporting: 1-2011 & ANSI/NIST-ITL 1-2011 Update 2013 (Traditional and NIEM XML Transactions)
File
Help
Traditional Transaction Batch Test
Traditional Transaction Editor
NIEM XML Transaction Batch Test
Options
Files Under Test
Batch File Statistics
Batch File List
Result
Path
Open Test Results (Text) File In External Editor
Open Test Results (XML) File In External Editor
Open Test Results Folder
Open Transaction Source File
Open Transaction Folder
Open Transaction In Editor
Results
Individual File Statistics
Visual Analysis
Text Log Output
Send Feedback

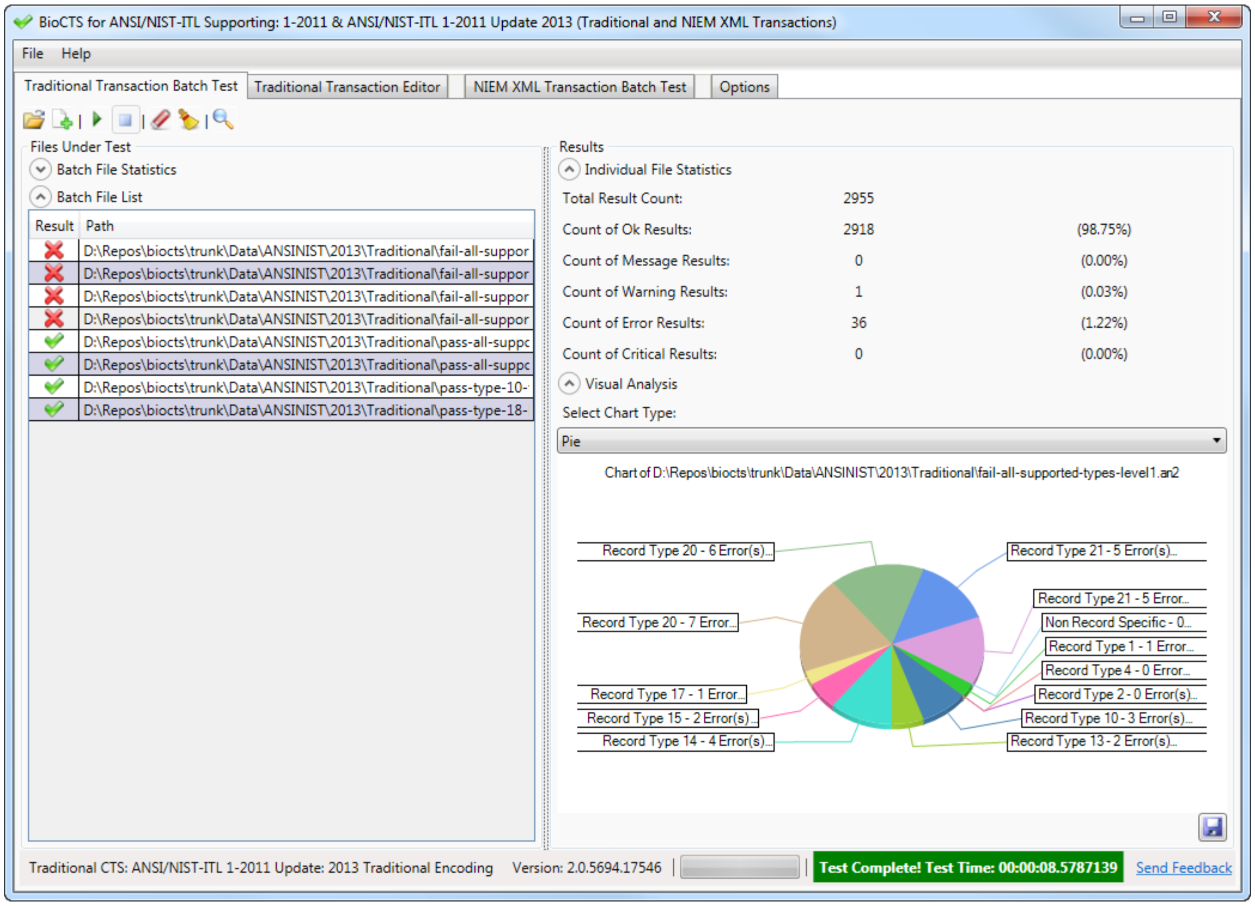
| Result | Path |
|---|---|
| ✗ | D:\biocts\trunk\Data\ANSINIST\2013\Traditional\fail-all-supported-typ |
| ✗ | D:\biocts\trunk\Data\ANSINIST\2013\Traditional\fail-all-supported-typ |
| ✗ | D:\biocts\trunk\Data\ANSINIST\2013\Traditional\fail-all-supported-typ |
| ✗ | D:\biocts\trunk\Data\ANSINIST\2013\Traditional\fail-all-supported-typ |
| ✗ | D:\biocts\trunk\Data\ANSINIST\2013\Traditional\fail-all-supported-typ |
| ✓ | D:\biocts\trunk\Data\ANSINIST\2013\Traditional\pass-all-supported-ty |
| ✓ | -13-9-14- |
| ✓ | -13-m1.ar |
| ✓ | -14-m1.ar |
| ✓ | 0-face-sm |
| ✓ | 8-FSA.an2 |

```
Conformance Test Suite: ANSI/NIST-ITL 1-2011 Update: 2013 Traditional Encoc
============================================================================
File Under Test:
  D:\biocts\trunk\Data\ANSINIST\2013\Traditional\pass-all-supported-types.a
File Under Test SHA-2 512 bit Hash Value:
  B7E279D1-8A99FAF4-33C30DF6-F500A38F-38B58295-7D7F8CCE-15C3165C-A20E3679-5
Data Under Test SHA-2 512 bit Hash Value:
  B7E279D1-8A99FAF4-33C30DF6-F500A38F-38B58295-7D7F8CCE-15C3165C-A20E3679-5
Do The Hash Values Match:
  The File Under Test Hash Value Matches the Data Under Test Hash Value


Overall Transaction Results: Pass
========================================

Total Bytes In File: 2296235
========================================
Type: 1 CRC: 23 Length: 388
Type: 2 IDC: 0 Length: 17
Type: 4 IDC: 1 Length: 104277
Type: 7 IDC: 18 Length: 614418
Type: 8 IDC: 19 Length: 1512
Type: 9 IDC: 21 Length: 4236
Type: 10 IDC: 2 Length: 70212
Type: 10 IDC: 10 Length: 70213
```

Traditional CTS: ANSI/NIST-ITL 1-2011 Update: 2013 Traditional Encoding    Version: 2.0.6100.17983    Test Complete! Test Time: 00:00:10.6081242

Figure 7 - Batch Test Tab with the Context Menu Open After Right Click

In addition to the detailed Textual Output Log, each file has a statistical breakdown of the results. There is also a high level graphical view of the errors per Record within a Transaction.  These graphs can be viewed within BioCTS itself, or saved as an image.


BioCTS for ANSI/NIST-ITL Supporting: 1-2011 & ANSI/NIST-ITL 1-2011 Update 2013 (Traditional and NIEM XML Transactions)

File
Help
Traditional Transaction Batch Test
Traditional Transaction Editor
NIEM XML Transaction Batch Test
Options
Files Under Test
Batch File Statistics
Batch File List

| Result | Path |
|---|---|
| ✗ | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\fail-all-suppor |
| ✗ | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\fail-all-suppor |
| ✗ | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\fail-all-suppor |
| ✗ | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\fail-all-suppor |
| ✓ | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\pass-all-suppc |
| ✓ | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\pass-all-suppc |
| ✓ | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\pass-type-10- |
| ✓ | D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\pass-type-18- |

Results

Individual File Statistics

| | | |
|---|---|---|
| Total Result Count: | 2955 | |
| Count of Ok Results: | 2918 | (98.75%) |
| Count of Message Results: | 0 | (0.00%) |
| Count of Warning Results: | 1 | (0.03%) |
| Count of Error Results: | 36 | (1.22%) |
| Count of Critical Results: | 0 | (0.00%) |

Visual Analysis

Select Chart Type: Pie

Chart of D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\fail-all-supported-types-level1.an2

Record Type 20 - 6 Error(s)...
Record Type 21 - 5 Error(s)...
Record Type 21 - 5 Error...
Record Type 20 - 7 Error...
Non Record Specific - 0...
Record Type 1 - 1 Error...
Record Type 4 - 0 Error...
Record Type 17 - 1 Error...
Record Type 2 - 0 Error(s)...
Record Type 15 - 2 Error(s)...
Record Type 10 - 3 Error(s)...
Record Type 14 - 4 Error(s)...
Record Type 13 - 2 Error(s)...

Traditional CTS: ANSI/NIST-ITL 1-2011 Update: 2013 Traditional Encoding    Version: 2.0.5694.17546    Test Complete! Test Time: 00:00:08.5787139    Send Feedback

Figure 8 - Batch Test Tab with a Transaction Selected and Statistics Displayed in Right Pane

### 5.3.2. Traditional Transaction Editor

The Traditional Transaction Editor is capable of interactively displaying all entities contained in a traditionally encoded transaction, allowing a user to edit, add and remove records and their contents.

The Editor[2] is designed to display as little or as much data as desired by the user by making use of expander sections, which can be expanded to display more information by pressing the button within sections. Below is a brief description of the tools[3] available in the Editor:

**Open File:** Loads a single file into the Editor and automatically tests it[4]

**Save File:** Saves the file to disk (location chosen by user) with any modifications from Editor

**Test Files** Tests or re-tests the file currently loaded into the Editor

**Reload from Disk:** Reloads the current file from disk, discarding any changes from the Editor

**Auto Fix:** Attempts to fix common file issues with Record Length and Transaction Contents

**Sort:** Sorts fields or records numerically (context dependent)

**Organize/Arrange:** Manually repositions fields or records using arrows (context dependent)

**Add:** Add field or record (context dependent)

**Remove/Delete:** Removes a field or record (context dependent)

**Modify Subfields:** Change number of subfields and information items (with or without data)

**View/Modify Data:** Displays data in a pop-out window and allows editing

**Show Record Image:** Attempts to display the image found in the record (context dependent)

The Editor displays three types of Results indicators for Records, Fields, Subfields, and Information Items:

✓ - **Passing Test Indicator** ✗ - **Failing Test Indicator** ◢ - **No Tests Results Indicator**

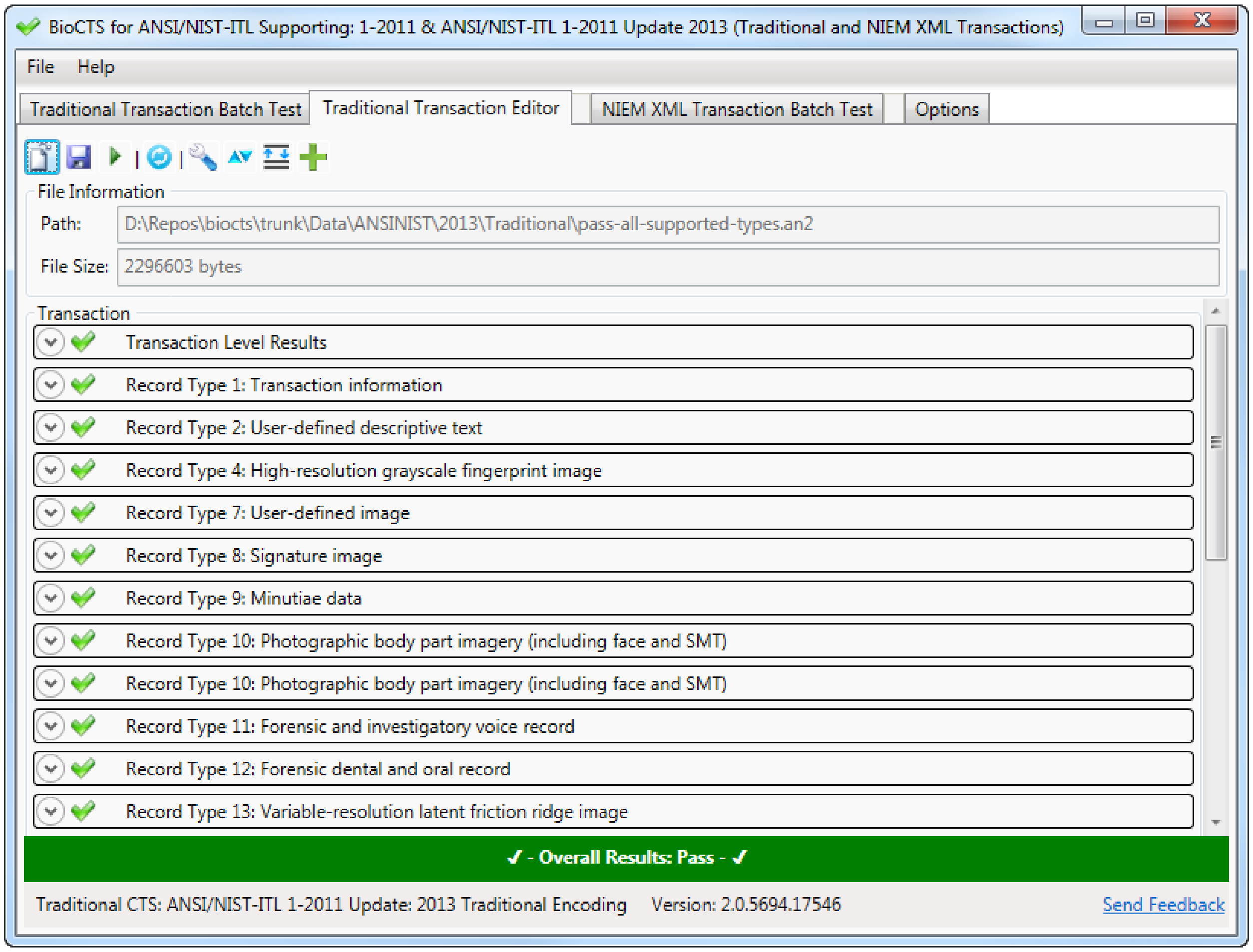
[2] Many Editor tools and actions, including adding or removing Records, Fields, and Subfields, cause the transaction to be re-tested automatically. In some cases, the Auto Fix feature will also be run to ensure the Transaction is formatted properly after the change takes place. For example, the removal of a Field will necessitate an adjustment to the Record Header Length to prevent parsing errors.

[3] All options for modifying or organizing fields are disabled for the first field in each record type (Field xx.001). The first field in each record is required for parsing because it specifies the record length.

[4] Drag and drop is supported on the Transaction area of the Traditional Transaction Editor tab so that files may be loaded into the editor. If multiple files are dragged, only the first one is loaded.

## *Transaction-Level Editing*

The Editor displays the data and associated results in an expandable, hierarchical format, and allows editing of existing data using text fields.


BioCTS for ANSI/NIST-ITL Supporting: 1-2011 & ANSI/NIST-ITL 1-2011 Update 2013 (Traditional and NIEM XML Transactions)

File Help

Traditional Transaction Batch Test | Traditional Transaction Editor | NIEM XML Transaction Batch Test | Options

File Information

Path: D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\pass-all-supported-types.an2

File Size: 2296603 bytes

Transaction

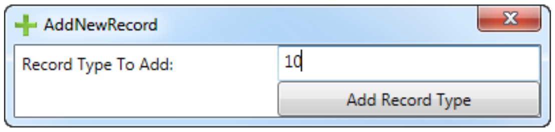
- Transaction Level Results
- Record Type 1: Transaction information
- Record Type 2: User-defined descriptive text
- Record Type 4: High-resolution grayscale fingerprint image
- Record Type 7: User-defined image
- Record Type 8: Signature image
- Record Type 9: Minutiae data
- Record Type 10: Photographic body part imagery (including face and SMT)
- Record Type 10: Photographic body part imagery (including face and SMT)
- Record Type 11: Forensic and investigatory voice record
- Record Type 12: Forensic dental and oral record
- Record Type 13: Variable-resolution latent friction ridge image

✓ - Overall Results: Pass - ✓

Traditional CTS: ANSI/NIST-ITL 1-2011 Update: 2013 Traditional Encoding Version: 2.0.5694.17546 Send Feedback

Figure 9 - The Editor with a Passing Transaction Loaded

At this level, new Records may be added by clicking the green plus icon (the user will be prompted for a Record Type number).

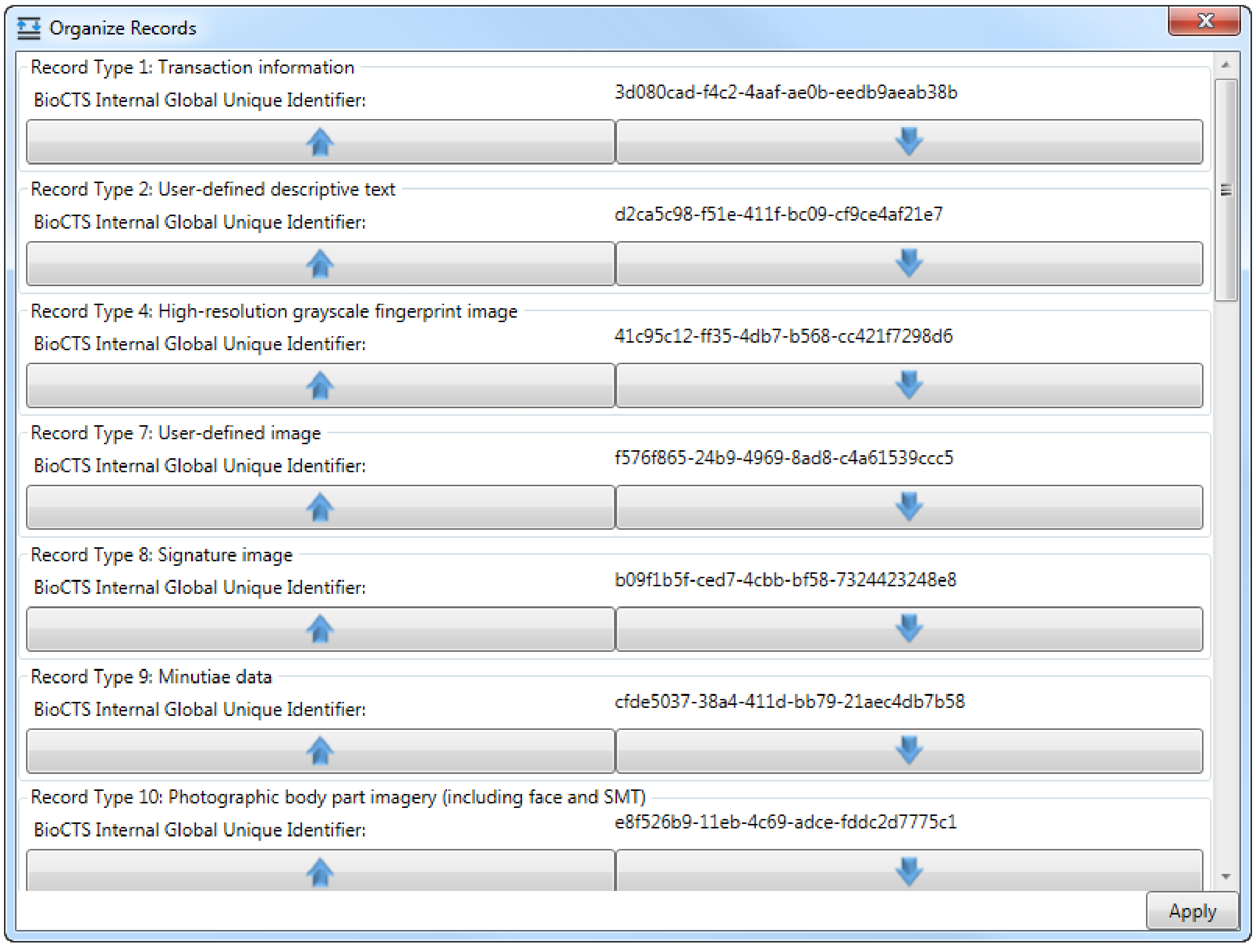

AddNewRecord

Record Type To Add: 10

Add Record Type

Figure 10 – The Editor Add Record Dialog

The Records may also be reorganized within the Transaction by using the numerical sort or the manual rearrange buttons.



Record Type 1: Transaction information
BioCTS Internal Global Unique Identifier: 3d080cad-f4c2-4aaf-ae0b-eedb9aeab38b

Record Type 2: User-defined descriptive text
BioCTS Internal Global Unique Identifier: d2ca5c98-f51e-411f-bc09-cf9ce4af21e7

Record Type 4: High-resolution grayscale fingerprint image
BioCTS Internal Global Unique Identifier: 41c95c12-ff35-4db7-b568-cc421f7298d6

Record Type 7: User-defined image
BioCTS Internal Global Unique Identifier: f576f865-24b9-4969-8ad8-c4a61539ccc5

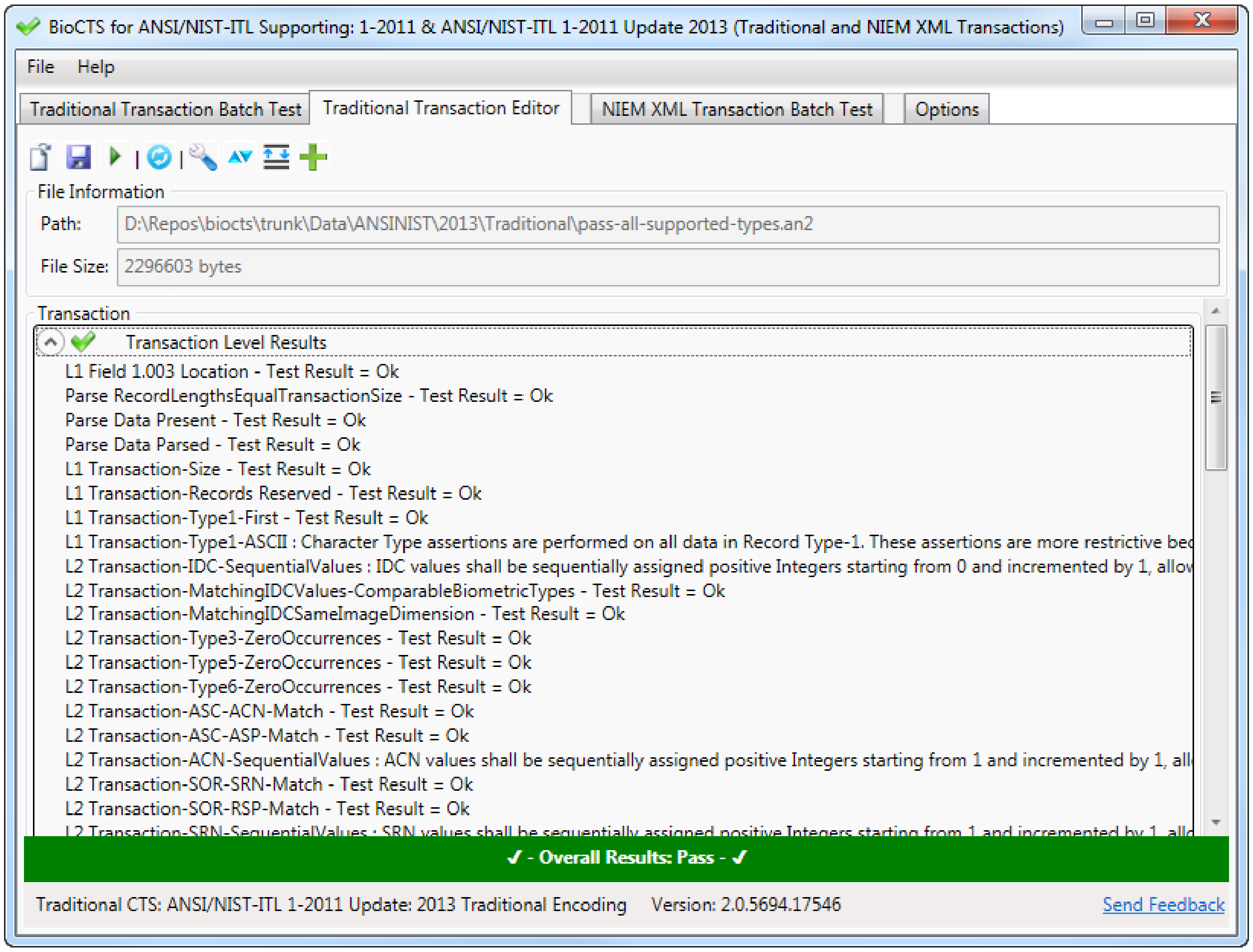
Record Type 8: Signature image
BioCTS Internal Global Unique Identifier: b09f1b5f-ced7-4cbb-bf58-7324423248e8

Record Type 9: Minutiae data
BioCTS Internal Global Unique Identifier: cfde5037-38a4-411d-bb79-21aec4db7b58

Record Type 10: Photographic body part imagery (including face and SMT)
BioCTS Internal Global Unique Identifier: e8f526b9-11eb-4c69-adce-fddc2d7775c1

**Figure 11 – The Editor Organize Records Dialog**

The Editor displays results in expander sections where appropriate. In this example, the Transaction-level results are shown. This method of viewing results applies to all constructs in the Transaction, including Records, Fields, Subfields, and Information Items.


BioCTS for ANSI/NIST-ITL Supporting: 1-2011 & ANSI/NIST-ITL 1-2011 Update 2013 (Traditional and NIEM XML Transactions)

File Help

Traditional Transaction Batch Test | Traditional Transaction Editor | NIEM XML Transaction Batch Test | Options

File Information

Path: D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\pass-all-supported-types.an2

File Size: 2296603 bytes

Transaction

Transaction Level Results

```
L1 Field 1.003 Location - Test Result = Ok
Parse RecordLengthsEqualTransactionSize - Test Result = Ok
Parse Data Present - Test Result = Ok
Parse Data Parsed - Test Result = Ok
L1 Transaction-Size - Test Result = Ok
L1 Transaction-Records Reserved - Test Result = Ok
L1 Transaction-Type1-First - Test Result = Ok
L1 Transaction-Type1-ASCII : Character Type assertions are performed on all data in Record Type-1. These assertions are more restrictive bec
L2 Transaction-IDC-SequentialValues : IDC values shall be sequentially assigned positive Integers starting from 0 and incremented by 1, allov
L2 Transaction-MatchingIDCValues-ComparableBiometricTypes - Test Result = Ok
L2 Transaction-MatchingIDCSameImageDimension - Test Result = Ok
L2 Transaction-Type3-ZeroOccurrences - Test Result = Ok
L2 Transaction-Type5-ZeroOccurrences - Test Result = Ok
L2 Transaction-Type6-ZeroOccurrences - Test Result = Ok
L2 Transaction-ASC-ACN-Match - Test Result = Ok
L2 Transaction-ASC-ASP-Match - Test Result = Ok
L2 Transaction-ACN-SequentialValues : ACN values shall be sequentially assigned positive Integers starting from 1 and incremented by 1, all
L2 Transaction-SOR-SRN-Match - Test Result = Ok
L2 Transaction-SOR-RSP-Match - Test Result = Ok
L2 Transaction-SRN-SequentialValues : SRN values shall be sequentially assigned positive Integers starting from 1 and incremented by 1, allo
```

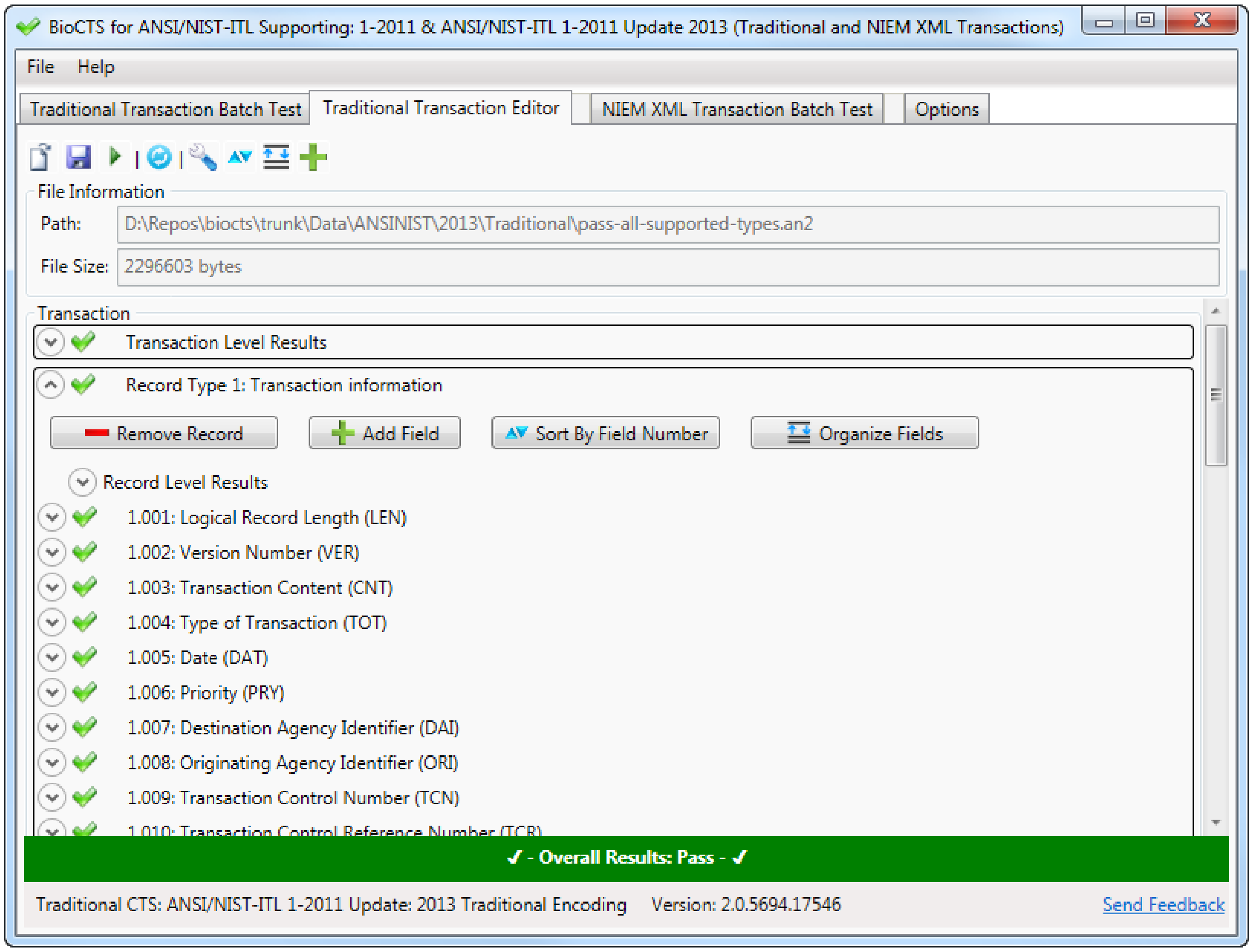
✓ - Overall Results: Pass - ✓

Traditional CTS: ANSI/NIST-ITL 1-2011 Update: 2013 Traditional Encoding Version: 2.0.5694.17546 Send Feedback

Figure 12 - The Editor Tab with a Passing Transaction Loaded - Displaying Transaction Level Results

### *Record-Level Editing*

When a record is expanded, the Editor displays a list of the fields contained in the record. The selected or expanded Record may be deleted by clicking the “Remove Record” button. In any expanded Record, the Add, Sort, and Organize Field options are available. If the Record contains an image, the option to “Show Record Image” will be available (not shown in Record-Type 1 below).


BioCTS for ANSI/NIST-ITL Supporting: 1-2011 & ANSI/NIST-ITL 1-2011 Update 2013 (Traditional and NIEM XML Transactions)
File
Help
Traditional Transaction Batch Test
Traditional Transaction Editor
NIEM XML Transaction Batch Test
Options
Remove Record
Add Field
Sort By Field Number
Organize Fields
Send Feedback

File Information

Path: D:\Repos\bioacts\trunk\Data\ANSINIST\2013\Traditional\pass-all-supported-types.an2

File Size: 2296603 bytes

Transaction

Transaction Level Results

Record Type 1: Transaction information

Record Level Results

1.001: Logical Record Length (LEN)

1.002: Version Number (VER)

1.003: Transaction Content (CNT)

1.004: Type of Transaction (TOT)

1.005: Date (DAT)

1.006: Priority (PRY)

1.007: Destination Agency Identifier (DAI)

1.008: Originating Agency Identifier (ORI)

1.009: Transaction Control Number (TCN)

1.010: Transaction Control Reference Number (TCR)

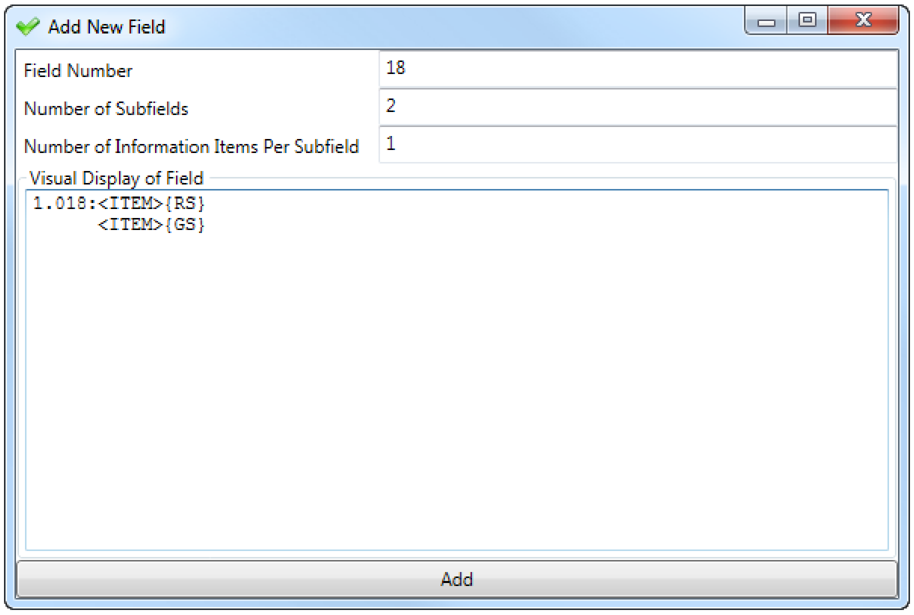
✓ - Overall Results: Pass - ✓

Traditional CTS: ANSI/NIST-ITL 1-2011 Update: 2013 Traditional Encoding  Version: 2.0.5694.17546

**Figure 13 - Editor Tab with Record Expanded**

The Add New Field prompt is context-dependent and will provide different options based upon whether the field is in an ASCII/Unicode-encoded Record Type or a Binary Record Type. The window provides visual feedback of what the Field will look like.

For ASCII/Unicode Records, the Visual Display will show the breakdown of the Subfield and Information Item organization.



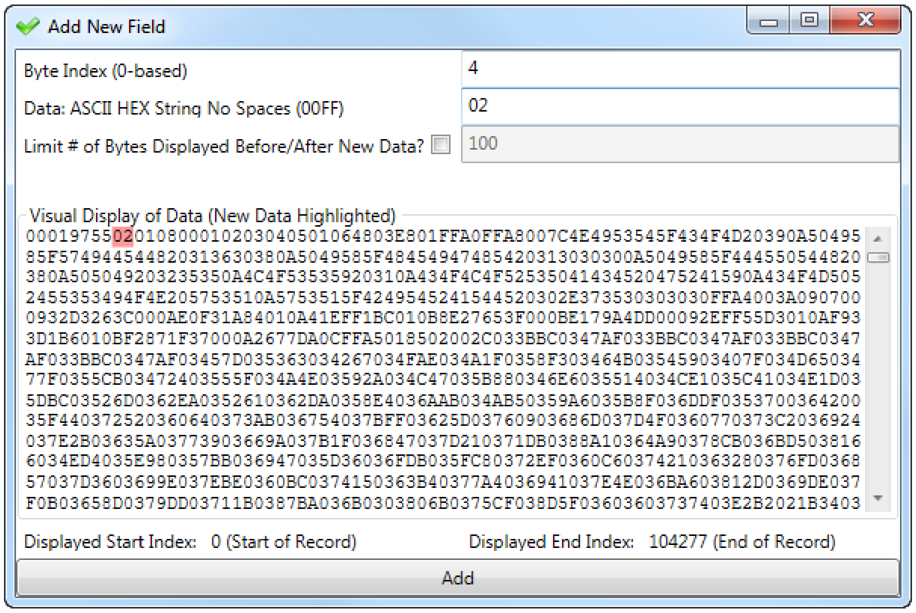
| | |
|---|---|
| Field Number | 18 |
| Number of Subfields | 2 |
| Number of Information Items Per Subfield | 1 |

Visual Display of Field

```
1.018:<ITEM>{RS}
      <ITEM>{GS}
```

Figure 14 - The Add New Field Dialog for ASCII/Unicode Records

For Binary Records, the Visual Display will show the inserted binary field data in the context of the binary Record data.



| | |
|---|---|
| Byte Index (0-based) | 4 |
| Data: ASCII HEX String No Spaces (00FF) | 02 |
| Limit # of Bytes Displayed Before/After New Data? ☐ | 100 |

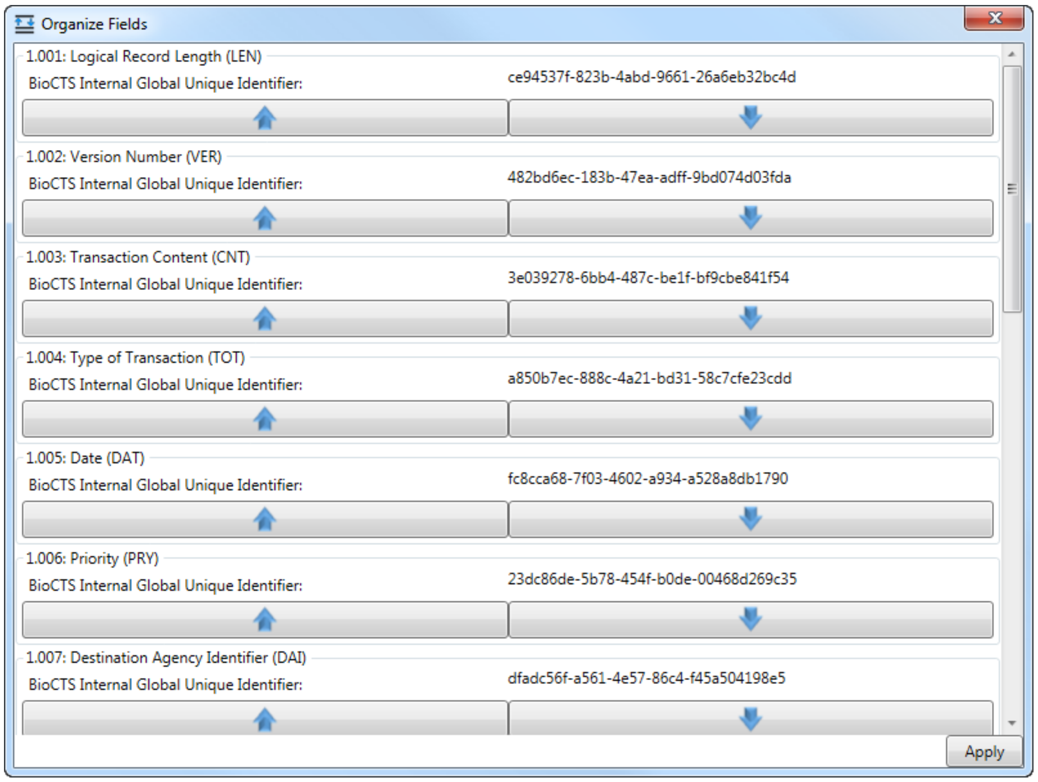
Visual Display of Data (New Data Highlighted)

```
0001975502010800010203040501064803E801FFA0FFA8007C4E4953545F434F4D20390A50495
85F574944544482031363038 0A5049585F48454947485420313030300A5049585F444550544820
```

Displayed Start Index: 0 (Start of Record) Displayed End Index: 104277 (End of Record)

Figure 15 - The Add New Field Dialog for Binary Records

The Fields may be reorganized within the Transaction by using the numerical sort or the manual rearrange buttons.



Organize Fields

1.001: Logical Record Length (LEN)
BioCTS Internal Global Unique Identifier: ce94537f-823b-4abd-9661-26a6eb32bc4d

1.002: Version Number (VER)
BioCTS Internal Global Unique Identifier: 482bd6ec-183b-47ea-adff-9bd074d03fda

1.003: Transaction Content (CNT)
BioCTS Internal Global Unique Identifier: 3e039278-6bb4-487c-be1f-bf9cbe841f54

1.004: Type of Transaction (TOT)
BioCTS Internal Global Unique Identifier: a850b7ec-888c-4a21-bd31-58c7cfe23cdd

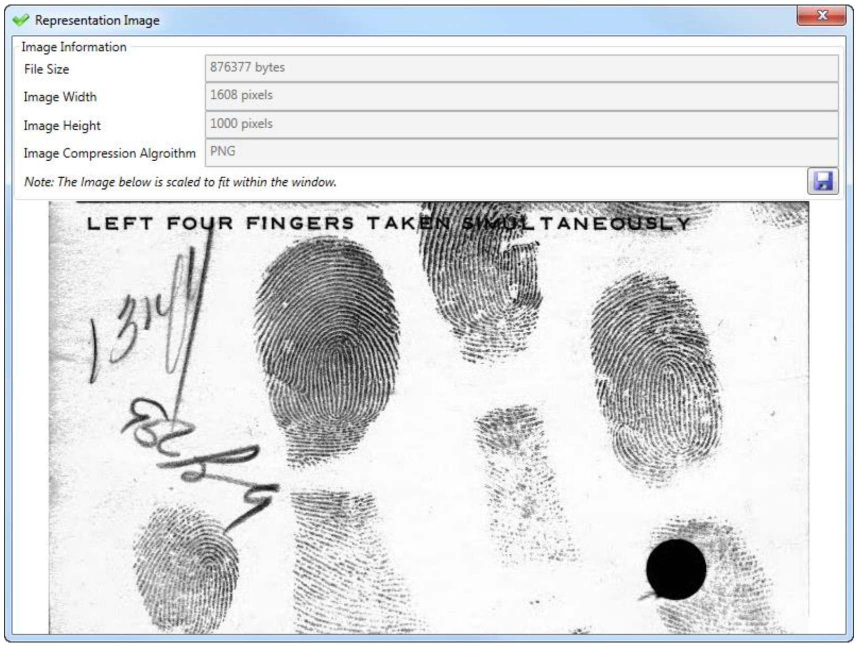
1.005: Date (DAT)
BioCTS Internal Global Unique Identifier: fc8cca68-7f03-4602-a934-a528a8db1790

1.006: Priority (PRY)
BioCTS Internal Global Unique Identifier: 23dc86de-5b78-454f-b0de-00468d269c35

1.007: Destination Agency Identifier (DAI)
BioCTS Internal Global Unique Identifier: dfadc56f-a561-4e57-86c4-f45a504198e5

Figure 16 – The Editor Organize Fields Dialog

If the selected Record contains an image, the "Show Record Image" button will be available. Clicking the button will show the image and its metadata. If the image type is not supported or cannot be read by BioCTS, a "No Preview Available" message will be displayed.



Representation Image

Image Information

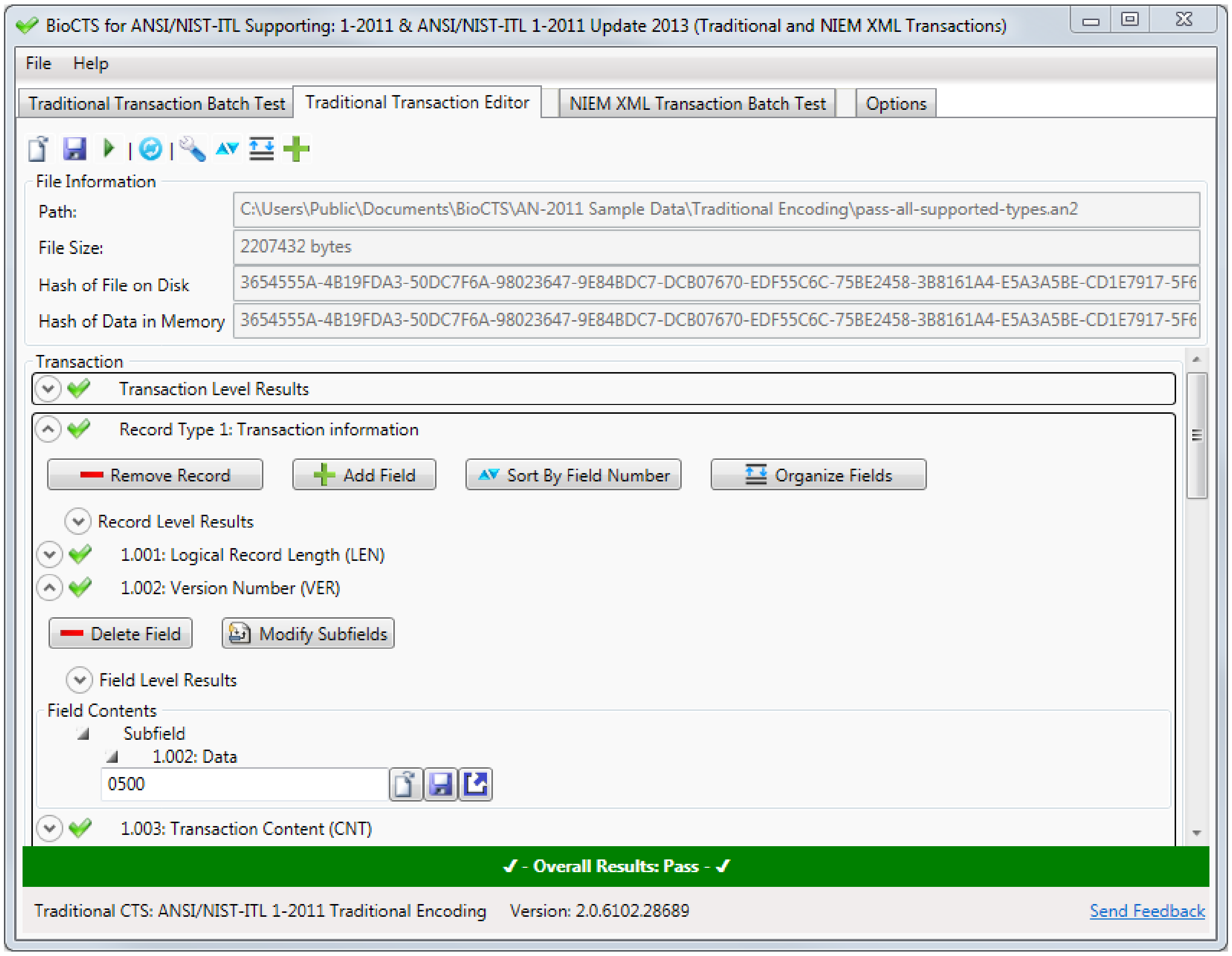
| | |
|---|---|
| File Size | 876377 bytes |
| Image Width | 1608 pixels |
| Image Height | 1000 pixels |
| Image Compression Algroithm | PNG |

*Note: The Image below is scaled to fit within the window.*

## *Field-Level Editing*

When expanded, a field displays expanders for the field-level results and the data that is held within the field. At this level, a Field can be deleted from a record. The values in the data text fields can be edited. Three buttons are provided to assist when a large amount of data is being manipulated (such as in an image field):

- **Load Data:** Loads data into the contents of a field/subfield/information item
- **Save Data:** Saves the contents of a field/subfield/information item to a file
- **Pop-Out Data:** Displays data in a large pop-out window for easy viewing and editing


BioCTS for ANSI/NIST-ITL Supporting: 1-2011 & ANSI/NIST-ITL 1-2011 Update 2013 (Traditional and NIEM XML Transactions)

File Help

Traditional Transaction Batch Test | Traditional Transaction Editor | NIEM XML Transaction Batch Test | Options

File Information

| | |
|---|---|
| Path: | C:\Users\Public\Documents\BioCTS\AN-2011 Sample Data\Traditional Encoding\pass-all-supported-types.an2 |
| File Size: | 2207432 bytes |
| Hash of File on Disk | 3654555A-4B19FDA3-50DC7F6A-98023647-9E84BDC7-DCB07670-EDF55C6C-75BE2458-3B8161A4-E5A3A5BE-CD1E7917-5F6 |
| Hash of Data in Memory | 3654555A-4B19FDA3-50DC7F6A-98023647-9E84BDC7-DCB07670-EDF55C6C-75BE2458-3B8161A4-E5A3A5BE-CD1E7917-5F6 |

Transaction

Transaction Level Results

Record Type 1: Transaction information

Remove Record | Add Field | Sort By Field Number | Organize Fields

Record Level Results

1.001: Logical Record Length (LEN)

1.002: Version Number (VER)

Delete Field | Modify Subfields

Field Level Results

Field Contents

Subfield

1.002: Data

0500

1.003: Transaction Content (CNT)

✓ - Overall Results: Pass - ✓

Traditional CTS: ANSI/NIST-ITL 1-2011 Traditional Encoding Version: 2.0.6102.28689 Send Feedback

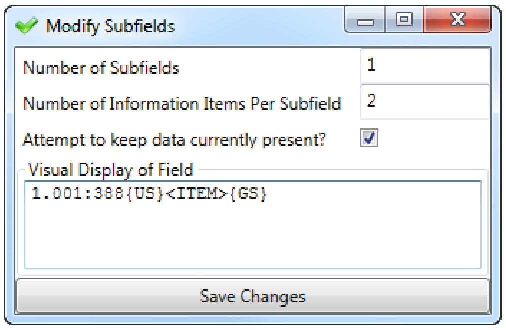
Figure 17 - The Editor Tab with Expanded Field

## Subfield Structures

All Fields contain at least one Subfield which contains at least one Information Item (see "How separators are used" in Annex B of NIST SP 500-290 Rev. 1). For Fields described as "Fields containing a single value" in NIST SP 500-290 Rev. 1 in Record Layout Tables, BioCTS displays one Subfield with one Information Item.

## Modifying Subfields

The "Modify Subfields" allows the user to modify the number of Subfields and Information Items, and decide whether to keep the current data. In the example below, a second Information Item is being added, and the previous data is retained in the first Information Item.



| | |
|---|---|
| Number of Subfields | 1 |
| Number of Information Items Per Subfield | 2 |
| Attempt to keep data currently present? | ✓ |

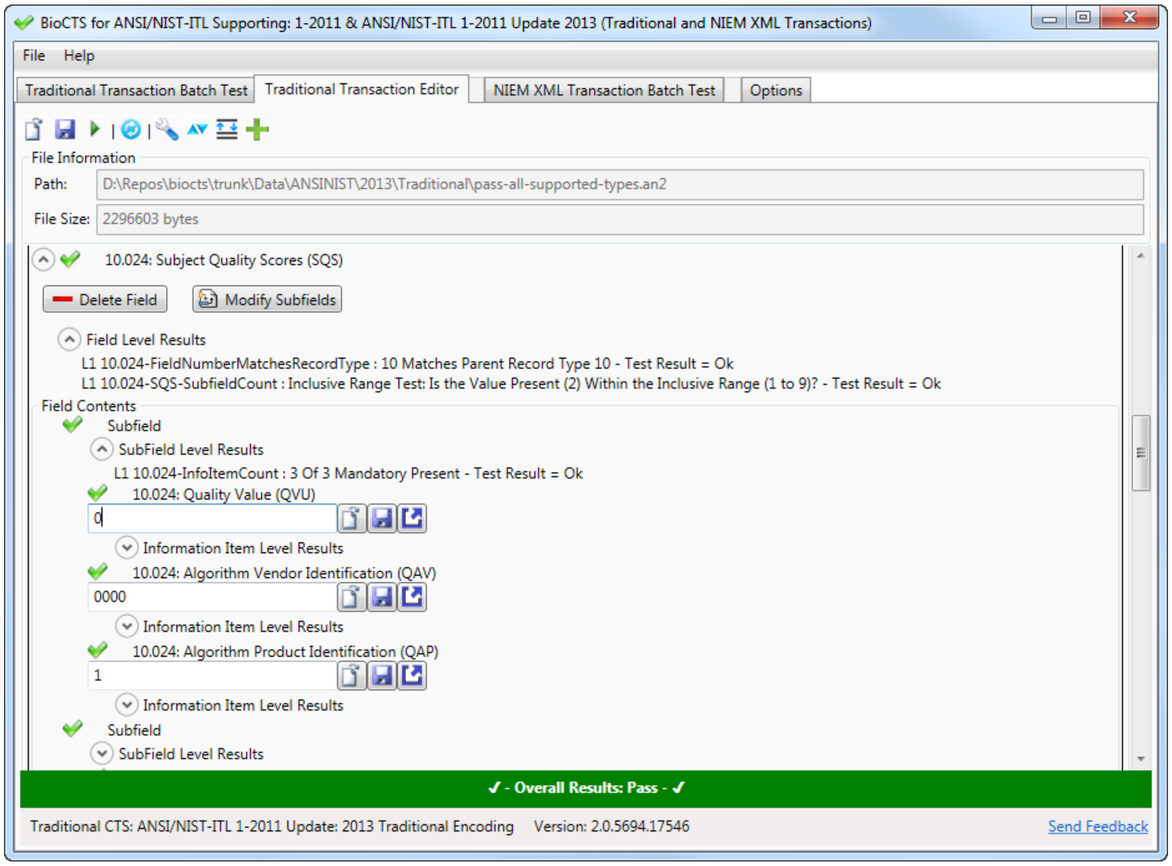
Visual Display of Field

```
1.001:388{US}<ITEM>{GS}
```

Figure 18 - The Modify Subfields Dialog

## *Subfield-Level Editing*

When a Field is expanded that contains multiple Subfields or Information Items, the containing Subfields and Information Items are displayed with their own set of data and Results. The Subfields and Information Item data can be edited in the same manner as Field data, by using the provided text field or the provided buttons for loading, saving, and modifying data.


BioCTS for ANSI/NIST-ITL Supporting: 1-2011 & ANSI/NIST-ITL 1-2011 Update 2013 (Traditional and NIEM XML Transactions)
File
Help
Traditional Transaction Batch Test
Traditional Transaction Editor
NIEM XML Transaction Batch Test
Options
File Information
Delete Field
Modify Subfields
Send Feedback

Path: D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\pass-all-supported-types.an2

File Size: 2296603 bytes

10.024: Subject Quality Scores (SQS)

Field Level Results

L1 10.024-FieldNumberMatchesRecordType : 10 Matches Parent Record Type 10 - Test Result = Ok

L1 10.024-SQS-SubfieldCount : Inclusive Range Test: Is the Value Present (2) Within the Inclusive Range (1 to 9)? - Test Result = Ok

Field Contents

Subfield

SubField Level Results

L1 10.024-InfoItemCount : 3 Of 3 Mandatory Present - Test Result = Ok

10.024: Quality Value (QVU)

0

Information Item Level Results

10.024: Algorithm Vendor Identification (QAV)

0000

Information Item Level Results

10.024: Algorithm Product Identification (QAP)

1

Information Item Level Results

Subfield

SubField Level Results

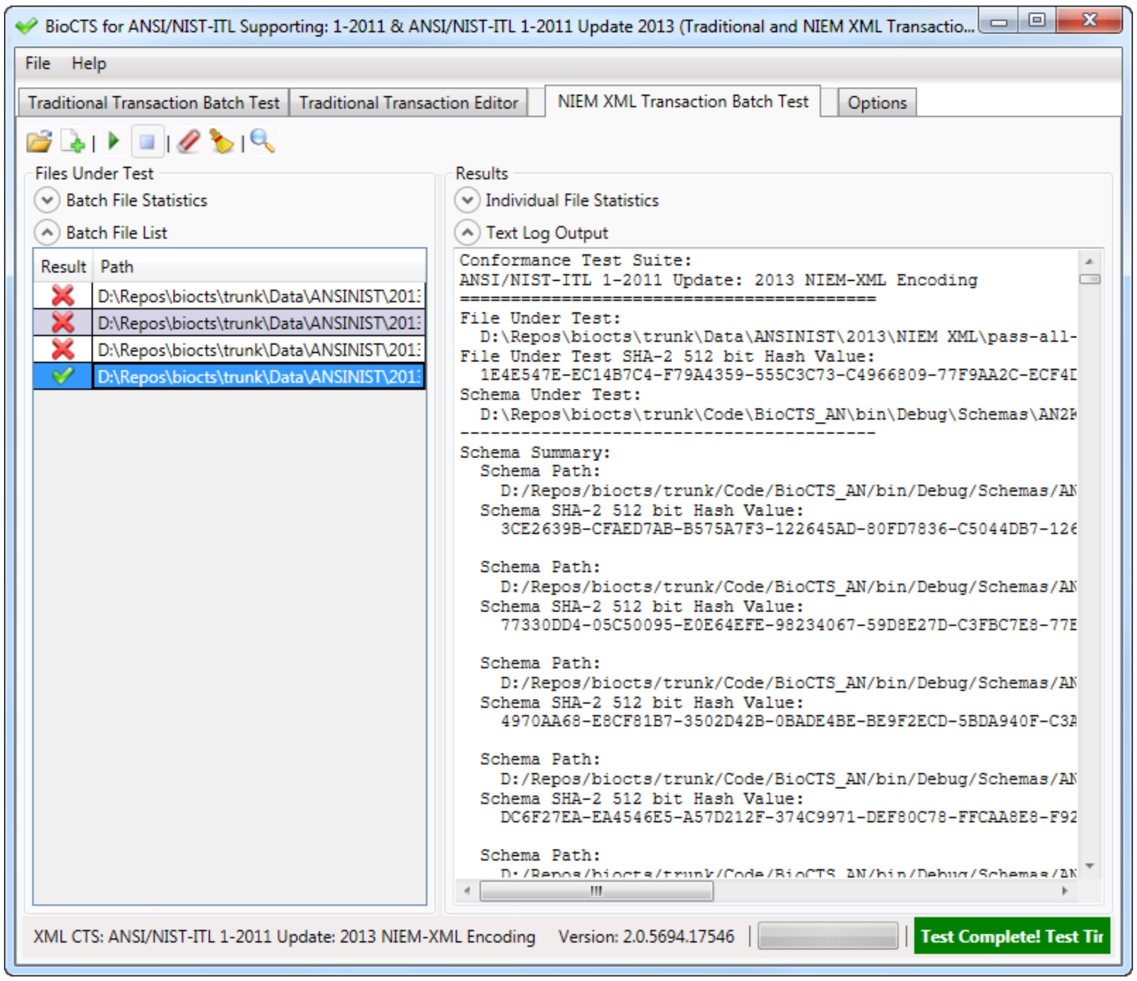
✓ - Overall Results: Pass - ✓

Traditional CTS: ANSI/NIST-ITL 1-2011 Update: 2013 Traditional Encoding Version: 2.0.5694.17546

Figure 19 – The Editor Tab with Expanded Subfields and Information Items

### 5.3.3. NIEM XML Transaction Batch Testing

The NIEM XML Transaction Batch Testing tab contains the same features found within the Traditional Transaction Batch Testing Tab, with the exception of the Graphical Display of Statistics and Editor. The context menu contains an option to open the XML Transaction in an external editor for editing. In XML Transaction Batch, the results display a summary of the SHA-2 Hash values of the Schema files used during testing of the files.


BioCTS for ANSI/NIST-ITL Supporting: 1-2011 & ANSI/NIST-ITL 1-2011 Update 2013 (Traditional and NIEM XML Transactio...
File
Help
Traditional Transaction Batch Test
Traditional Transaction Editor
NIEM XML Transaction Batch Test
Options
Files Under Test
Batch File Statistics
Batch File List
Result
Path
Results
Individual File Statistics
Text Log Output

D:\Repos\biocts\trunk\Data\ANSINIST\201
D:\Repos\biocts\trunk\Data\ANSINIST\201
D:\Repos\biocts\trunk\Data\ANSINIST\201
D:\Repos\biocts\trunk\Data\ANSINIST\201

```
Conformance Test Suite:
ANSI/NIST-ITL 1-2011 Update: 2013 NIEM-XML Encoding
=========================================
File Under Test:
  D:\Repos\biocts\trunk\Data\ANSINIST\2013\NIEM XML\pass-all-
File Under Test SHA-2 512 bit Hash Value:
  1E4E547E-EC14B7C4-F79A4359-555C3C73-C4966809-77F9AA2C-ECF4D
Schema Under Test:
  D:\Repos\biocts\trunk\Code\BioCTS_AN\bin\Debug\Schemas\AN2K
-----------------------------------------
Schema Summary:
  Schema Path:
    D:/Repos/biocts/trunk/Code/BioCTS_AN/bin/Debug/Schemas/AN
  Schema SHA-2 512 bit Hash Value:
    3CE2639B-CFAED7AB-B575A7F3-122645AD-80FD7836-C5044DB7-126

  Schema Path:
    D:/Repos/biocts/trunk/Code/BioCTS_AN/bin/Debug/Schemas/AN
  Schema SHA-2 512 bit Hash Value:
    77330DD4-05C50095-E0E64EFE-98234067-59D8E27D-C3FBC7E8-77E

  Schema Path:
    D:/Repos/biocts/trunk/Code/BioCTS_AN/bin/Debug/Schemas/AN
  Schema SHA-2 512 bit Hash Value:
    4970AA68-E8CF81B7-3502D42B-0BADE4BE-BE9F2ECD-5BDA940F-C3A

  Schema Path:
    D:/Repos/biocts/trunk/Code/BioCTS_AN/bin/Debug/Schemas/AN
  Schema SHA-2 512 bit Hash Value:
    DC6F27EA-EA4546E5-A57D212F-374C9971-DEF80C78-FFCAA8E8-F92

  Schema Path:
    D:/Repos/biocts/trunk/Code/BioCTS_AN/bin/Debug/Schemas/AN
```

XML CTS: ANSI/NIST-ITL 1-2011 Update: 2013 NIEM-XML Encoding | Version: 2.0.5694.17546 | Test Complete! Test Tir

Figure 20 - The NIEM XML Transaction Batch Testing Tab with Output

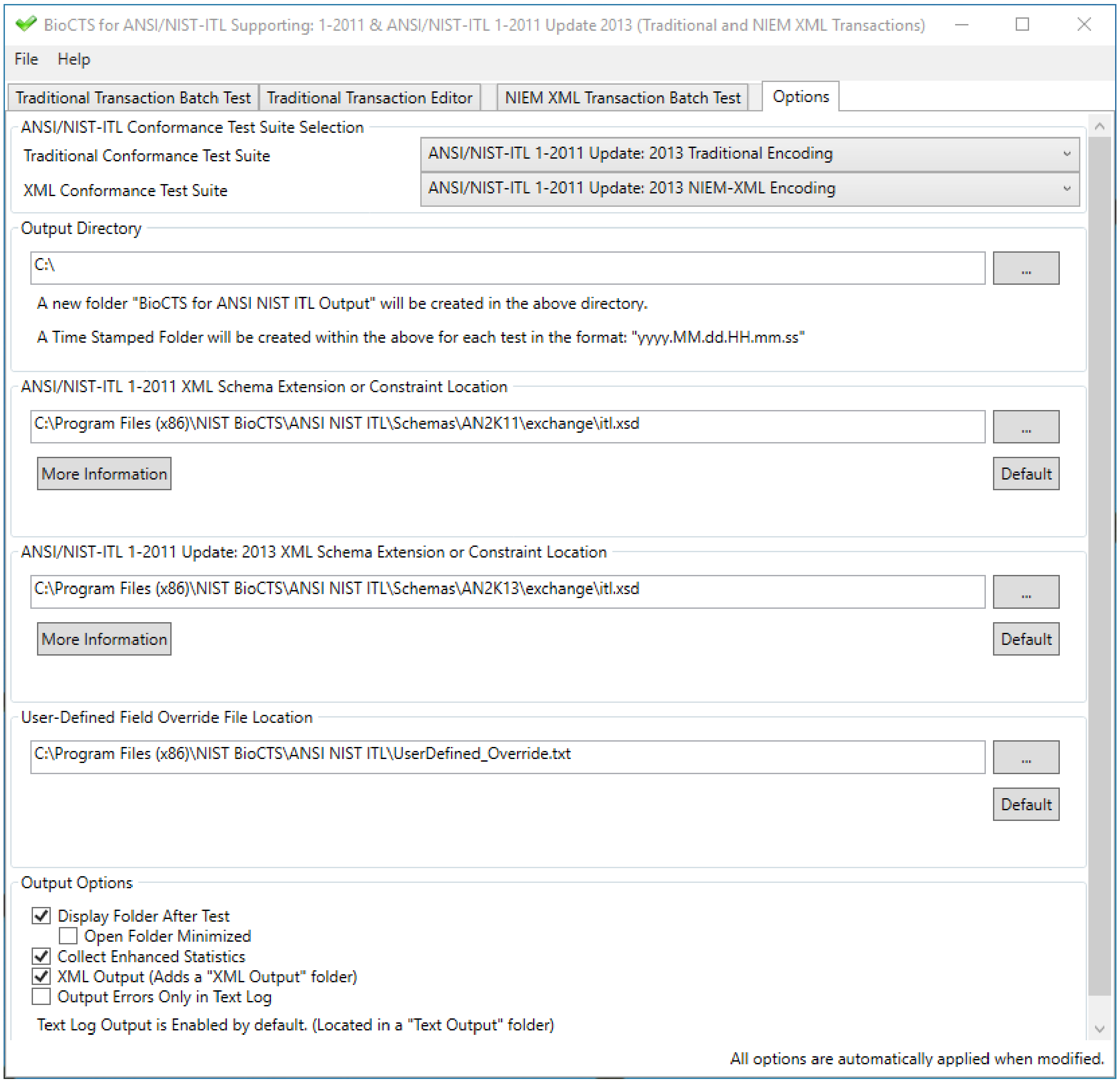
### 5.3.4. Options

The “Options” tab provides the testing and output customization choices described below[5]. Note that the Options tab is disabled during testing to prevent changes that may compromise the integrity of the Test Results.

- **ANSI/NIST-ITL Conformance Test Suite Selection:** The Options tab is where the Conformance Test Suites are selected. There are drop-down menus that allow for selection of a Traditional Conformance Test Suite, and a NIEM XML Conformance Test Suite.
- **Output Directory:** The Options tab also includes a directory selection of where BioCTS saves log files. Within this directory, time-stamped folders are generated for each batch test performed.
    - The time-stamped folder has the following format:
        - yyyy – 4 digit year (e.g. 2012)
        - MM – 2 digit month (e.g. 10)
        - dd – 2 digit day (e.g. 31)
        - HH – 2 digit hour in 24-hour scale (e.g. 13)
        - mm – 2 digit minutes (e.g. 59)
        - ss – 2 digit seconds (e.g. 22)
    - In the example provided below:
        - Text Output will be generated in the directory:
            - `C:\Users\dyaga\Desktop\BioCTS for ANSI NIST ITL Output\2012.10.31.13.59.22\Text Output`
        - XML Output will be generated in the directory:
            - `C:\Users\dyaga\Desktop\BioCTS for ANSI NIST ITL Output\2012.10.31.13.59.22\XML Output`
- **ANSI/NIST-ITL 1-2011 XML Schema Extension or Constraint Location / ANSI/NIST-ITL 1-2011 Update: 2013 XML Schema Extension or Constraint Location:** The Options tab is also where the Schema file location is specified. BioCTS provides a set of default Schema Files which are based on the official ANSI/NIST-ITL Schemas released with the standard. However, the 2011 Schema files have been modified for use in the BioCTS AN-2011 CTS; see Annex A of this document, which provides details regarding the XML Schema modifications and rationale for the changes. A Constraint or Extension Schema may be used in place of the Schema files provided with the software. The features provided (and test outcome for each usage are detailed below):
    - **Extension Schemas**: Support for Extension Schemas is provided to allow substitution elements to be specified for user-defined information in the standard such as the user-defined abstract elements found in Record Type-2.

[5] The Apply button has been removed based upon user feedback—changes made in the Options Tab are automatically applied.

    - **Constraint Schemas**: Support for Constraint Schemas is provided to allow constraints and restrictions to be defined for use in particular applications such as standard profiles like the FBI EBTS.
    - **User Defined Fields Tested Against Default Schemas:** When XML files with User-Defined fields are tested with the Default Schema Files, the XML files may fail the Schema validation test (A User-Defined Schema should be used).
    - **Tests against User-Defined Schemas**: When a user-defined (constraint or extension) Schema is used, the validity of the test results reported by BioCTS depends upon the validity of the User-Defined Schema. User-Defined Schemas are valid only if they conform to the requirements in Section C.3 of NIST SP 500-290 and Section C.3 of NIST SP 500-290 Rev1 (2013). BioCTS does not verify that User-Defined Schemas are compliant with NIST SP 500-290—only use Schema files from trusted sources.
        - If the user-defined Schema is not syntactically correct, BioCTS will warn the user.
        - If the XML files tested contain User-Defined Fields that the user-defined Schema does not contain, the XML files will fail the Schema validation test.
- **User-Defined Field Override File Location:** A New feature to this release is the specification of a User-Defined Field Override File. There are many "User Defined" Fields specified in the ANSI/NIST-ITL standards (e.g., Type 02 Fields), and by default they are shown as "User Defined Field" within the Editor and Test Output Logs. This feature allows a user to override the names of these fields so that they are more descriptive than the default "User Defined Field".
- **Output Options:** Options exist to display the Log folder after a test, the collection of enhanced statistics of the Tests performed within the software, and the option to generate XML logs or not. An option is available to force the Log Folder to be opened minimized to avoid interrupting the user with a pop-up window. An option is also available to output only errors in the text logs.


BioCTS for ANSI/NIST-ITL Supporting: 1-2011 & ANSI/NIST-ITL 1-2011 Update 2013 (Traditional and NIEM XML Transactions)

File Help

Traditional Transaction Batch Test | Traditional Transaction Editor | NIEM XML Transaction Batch Test | Options

ANSI/NIST-ITL Conformance Test Suite Selection

Traditional Conformance Test Suite: ANSI/NIST-ITL 1-2011 Update: 2013 Traditional Encoding

XML Conformance Test Suite: ANSI/NIST-ITL 1-2011 Update: 2013 NIEM-XML Encoding

Output Directory

C:\

A new folder "BioCTS for ANSI NIST ITL Output" will be created in the above directory.

A Time Stamped Folder will be created within the above for each test in the format: "yyyy.MM.dd.HH.mm.ss"

ANSI/NIST-ITL 1-2011 XML Schema Extension or Constraint Location

C:\Program Files (x86)\NIST BioCTS\ANSI NIST ITL\Schemas\AN2K11\exchange\itl.xsd

More Information | Default

ANSI/NIST-ITL 1-2011 Update: 2013 XML Schema Extension or Constraint Location

C:\Program Files (x86)\NIST BioCTS\ANSI NIST ITL\Schemas\AN2K13\exchange\itl.xsd

More Information | Default

User-Defined Field Override File Location

C:\Program Files (x86)\NIST BioCTS\ANSI NIST ITL\UserDefined_Override.txt

Default

Output Options

- [✓] Display Folder After Test
  - [ ] Open Folder Minimized
- [✓] Collect Enhanced Statistics
- [✓] XML Output (Adds a "XML Output" folder)
- [ ] Output Errors Only in Text Log

Text Log Output is Enabled by default. (Located in a "Text Output" folder)

All options are automatically applied when modified.

**Figure 16 - The Options Tab**

## 5.4. Elements of a Result

The result of any test is comprised of 4 elements:

- Test Name – A string to define what test took place
- Test Level – Testing can take place at multiple levels
    - Parse – A parsing error occurred with this element
    - L1 – A level 1 test
    - L2 – A level 2 test
    - L3 – A level 3 test
- Test Result
    - Ok – The test passed
    - Warning – The test passed, but warranted a warning statement (i.e. an "Unspecified" value, etc.)
    - Error – The test failed
    - Critical Error – The test failed and may be preventing other tests from being performed
- Test Message – A description of what went on during the test, and possible error message

The Text Output Log's format has been altered since the initial release of BioCTS for ANSI/NIST-ITL. The information presented in a Test Result has been significantly expanded; the test message is generated with as much information as possible, including various operators and operands used within the test. The following is an example of the current Test Results, where each element of the Result is clearly labelled within the Text Output Log:

```
-----------------------------------------------------------------------------------------
 6.|     Test Name| 1.001-LEN-CharCount
-----------------------------------------------------------------------------------------
   |         Level| L1
    -------------------------------------------------------------------------------------
   |        Result| Ok
    -------------------------------------------------------------------------------------
   | Test Message| Greater Than or Equal Test: Is the character count present (3) greater than or
   |              | equal to the minimum non-negative Integer value (1)?
-----------------------------------------------------------------------------------------
```

Figure 21 - Text Output Result Example

The same exact Result is encoded in the XML Log as follows:

```
<Result>
  <Level>L1</Level>
  <Message>
    Greater Than or Equal Test: Is the character count present (3) greater than or equal
    to the minimum non-negative Integer value (1)?
  </Message>
  <Results>Ok</Results>
  <Test>1.001-LEN-CharCount</Test>
</Result>
```

Figure 22 - XML Output Result Example

## 5.5. Troubleshooting and Errors

This section contains information to assist in understanding the normal operation of BioCTS for AN-ITL v2 and common errors encountered when using the software. This is not a comprehensive listing of all errors that may be reported.

### *Parsing (Traditional Encoding)*

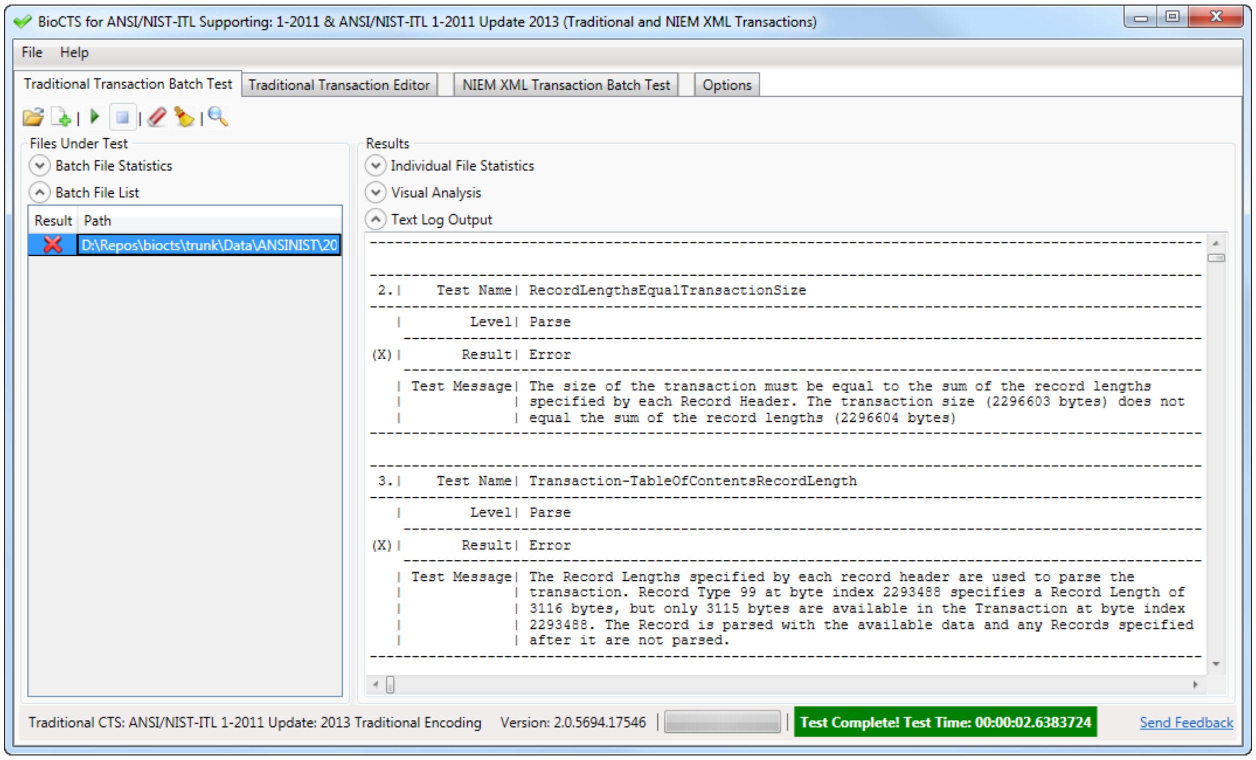
The parsing engine will continue to parse the transaction even when it encounters some critical errors such as data length discrepancies. Errors messages are reported to assist in discovering why the transaction is malformed.

**Operation:** The parsing engine relies heavily on the contents of Field 1.003-CNT (Transaction Content) and the Record Header (Field xx.001) for each Record. The general parsing procedure follows:

- Each pair of REC (record category code) and IDC (information designation character) values found in 1.003-CNT is read and saved to a list.
- The value of Field 1.001 (the record length) is used to progress to the next Record location.
- The parser uses each REC value (beginning with the first) to determine if the Record is binary or ASCII/Unicode.
    - If ASCII, the field at the location is parsed and the Field value is saved as the record length.
    - If binary, the first 4 bytes are read in as the Field value, which is saved as the record length.
- The IDC value (Field xx.002) is also saved, and the record length value from the previous step is used to progress to the next Record location. The process is repeated until all REC values have been exhausted or the end of the file is encountered. If the end of the file is encountered before processing all REC values, or if extra data is discovered after processing all REC values, an error is reported.
- The list of REC values, record lengths, and IDC values is used to create Records and parse their containing Fields. These Records and IDC values are compared against the list obtained from 1.003-CNT. Any discrepancies are reported as errors.

**Data Too Short / Missing Data:**

If the data in the Transaction is too short when compared to the values found in 1.003-CNT and Field xx.001 of each Record, the tool will report the error in the Transaction-related results under the test name “Transaction-TableOfContentsRecordLength”.


BioCTS for ANSI/NIST-ITL Supporting: 1-2011 & ANSI/NIST-ITL 1-2011 Update 2013 (Traditional and NIEM XML Transactions)
File Help
Traditional Transaction Batch Test
Traditional Transaction Editor
NIEM XML Transaction Batch Test
Options
Files Under Test
Batch File Statistics
Batch File List
Result Path
D:\Repos\biocts\trunk\Data\ANSINIST\20
Results
Individual File Statistics
Visual Analysis
Text Log Output

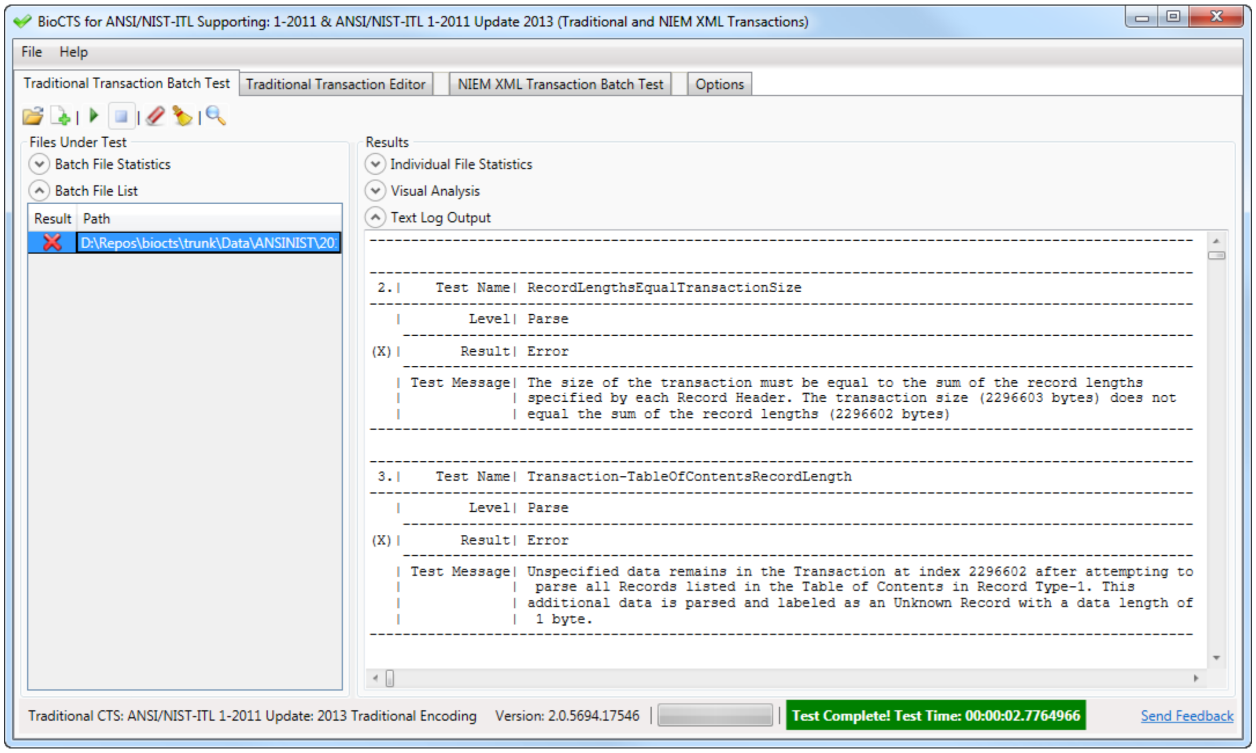
```
------------------------------------------------------------------------------------

------------------------------------------------------------------------------------
 2.|    Test Name| RecordLengthsEqualTransactionSize
------------------------------------------------------------------------------------
   |        Level| Parse
------------------------------------------------------------------------------------
(X)|       Result| Error
   ---------------------------------------------------------------------------------
   | Test Message| The size of the transaction must be equal to the sum of the record lengths
   |             | specified by each Record Header. The transaction size (2296603 bytes) does not
   |             | equal the sum of the record lengths (2296604 bytes)
------------------------------------------------------------------------------------

------------------------------------------------------------------------------------
 3.|    Test Name| Transaction-TableOfContentsRecordLength
------------------------------------------------------------------------------------
   |        Level| Parse
------------------------------------------------------------------------------------
(X)|       Result| Error
   ---------------------------------------------------------------------------------
   | Test Message| The Record Lengths specified by each record header are used to parse the
   |             | transaction. Record Type 99 at byte index 2293488 specifies a Record Length of
   |             | 3116 bytes, but only 3115 bytes are available in the Transaction at byte index
   |             | 2293488. The Record is parsed with the available data and any Records specified
   |             | after it are not parsed.
------------------------------------------------------------------------------------
```

Traditional CTS: ANSI/NIST-ITL 1-2011 Update: 2013 Traditional Encoding    Version: 2.0.5694.17546    Test Complete! Test Time: 00:00:02.6383724    Send Feedback

Figure 23 – Error: Transaction-TableOfContentsRecordLength (Transaction too short / missing data)

**Data Too Long / Extra Data:**

If the data in the Transaction is too long when compared to the values found in 1.003-CNT and Field xx.001 of each Record, the tool will report the error in the Transaction-related results under the test name “Transaction-TableOfContentsRecordLength”.

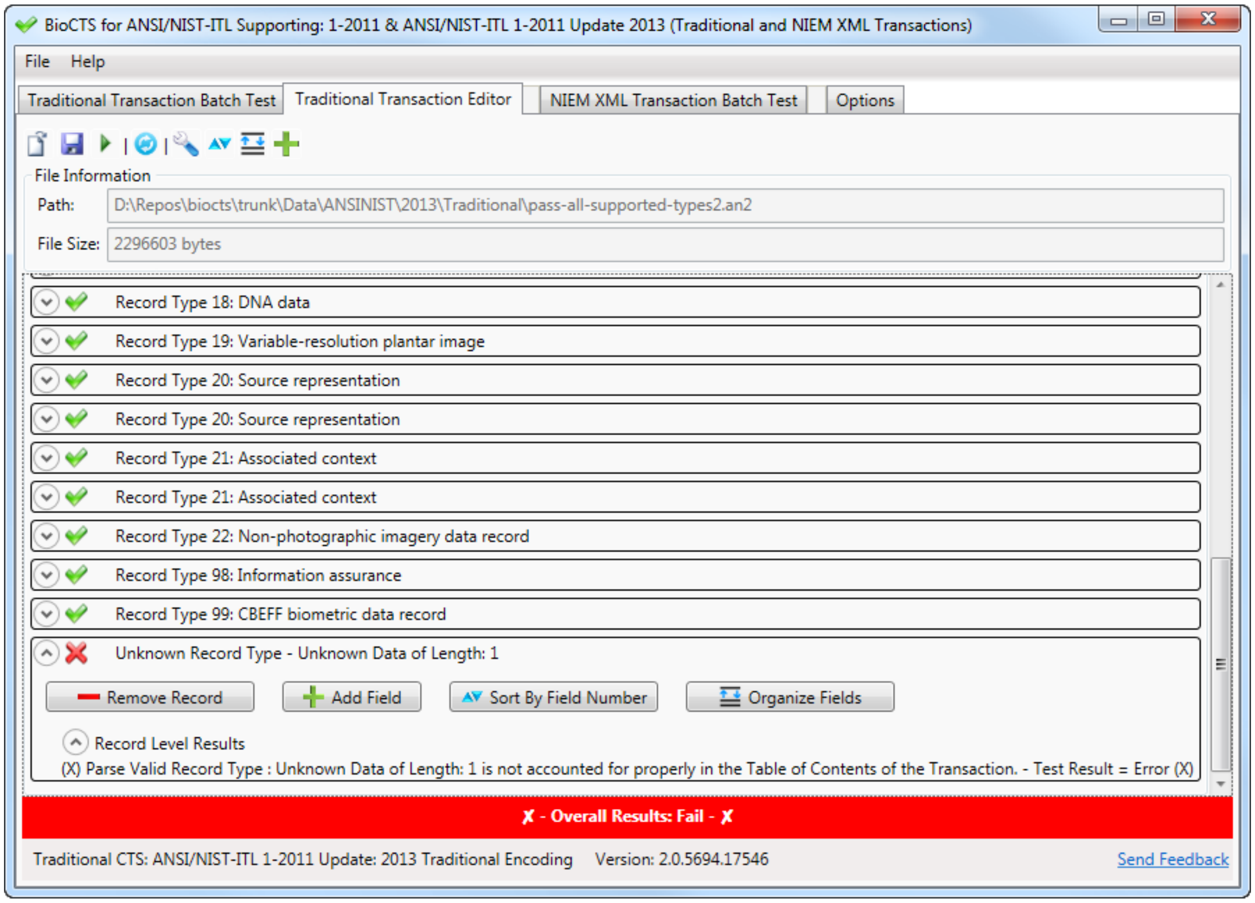

BioCTS for ANSI/NIST-ITL Supporting: 1-2011 & ANSI/NIST-ITL 1-2011 Update 2013 (Traditional and NIEM XML Transactions)
File
Help
Traditional Transaction Batch Test
Traditional Transaction Editor
NIEM XML Transaction Batch Test
Options
Files Under Test
Batch File Statistics
Batch File List
Result
Path
D:\Repos\biocts\trunk\Data\ANSINIST\20
Results
Individual File Statistics
Visual Analysis
Text Log Output

```
-------------------------------------------------------------------------------------------------

-------------------------------------------------------------------------------------------------
 2.|    Test Name| RecordLengthsEqualTransactionSize
-------------------------------------------------------------------------------------------------
   |        Level| Parse
-------------------------------------------------------------------------------------------------
(X)|       Result| Error
-------------------------------------------------------------------------------------------------
   | Test Message| The size of the transaction must be equal to the sum of the record lengths
   |             | specified by each Record Header. The transaction size (2296603 bytes) does not
   |             | equal the sum of the record lengths (2296602 bytes)
-------------------------------------------------------------------------------------------------

-------------------------------------------------------------------------------------------------
 3.|    Test Name| Transaction-TableOfContentsRecordLength
-------------------------------------------------------------------------------------------------
   |        Level| Parse
-------------------------------------------------------------------------------------------------
(X)|       Result| Error
-------------------------------------------------------------------------------------------------
   | Test Message| Unspecified data remains in the Transaction at index 2296602 after attempting to
   |             |  parse all Records listed in the Table of Contents in Record Type-1. This
   |             | additional data is parsed and labeled as an Unknown Record with a data length of
   |             |  1 byte.
-------------------------------------------------------------------------------------------------
```

Traditional CTS: ANSI/NIST-ITL 1-2011 Update: 2013 Traditional Encoding    Version: 2.0.5694.17546    Test Complete! Test Time: 00:00:02.7764966    Send Feedback

Figure 24 – Error: Transaction-TableOfContentsRecordLength (Transaction too long / additional data)

The software will save any additional data to an “Unknown Record Type”, shown below in the Editor:


BioCTS for ANSI/NIST-ITL Supporting: 1-2011 & ANSI/NIST-ITL 1-2011 Update 2013 (Traditional and NIEM XML Transactions)

File
Help
Traditional Transaction Batch Test
Traditional Transaction Editor
NIEM XML Transaction Batch Test
Options
File Information

Path: D:\Repos\biocts\trunk\Data\ANSINIST\2013\Traditional\pass-all-supported-types2.an2

File Size: 2296603 bytes

✓ Record Type 18: DNA data
✓ Record Type 19: Variable-resolution plantar image
✓ Record Type 20: Source representation
✓ Record Type 20: Source representation
✓ Record Type 21: Associated context
✓ Record Type 21: Associated context
✓ Record Type 22: Non-photographic imagery data record
✓ Record Type 98: Information assurance
✓ Record Type 99: CBEFF biometric data record
✗ Unknown Record Type - Unknown Data of Length: 1

Record Level Results

(X) Parse Valid Record Type : Unknown Data of Length: 1 is not accounted for properly in the Table of Contents of the Transaction. - Test Result = Error (X)

X - Overall Results: Fail - X

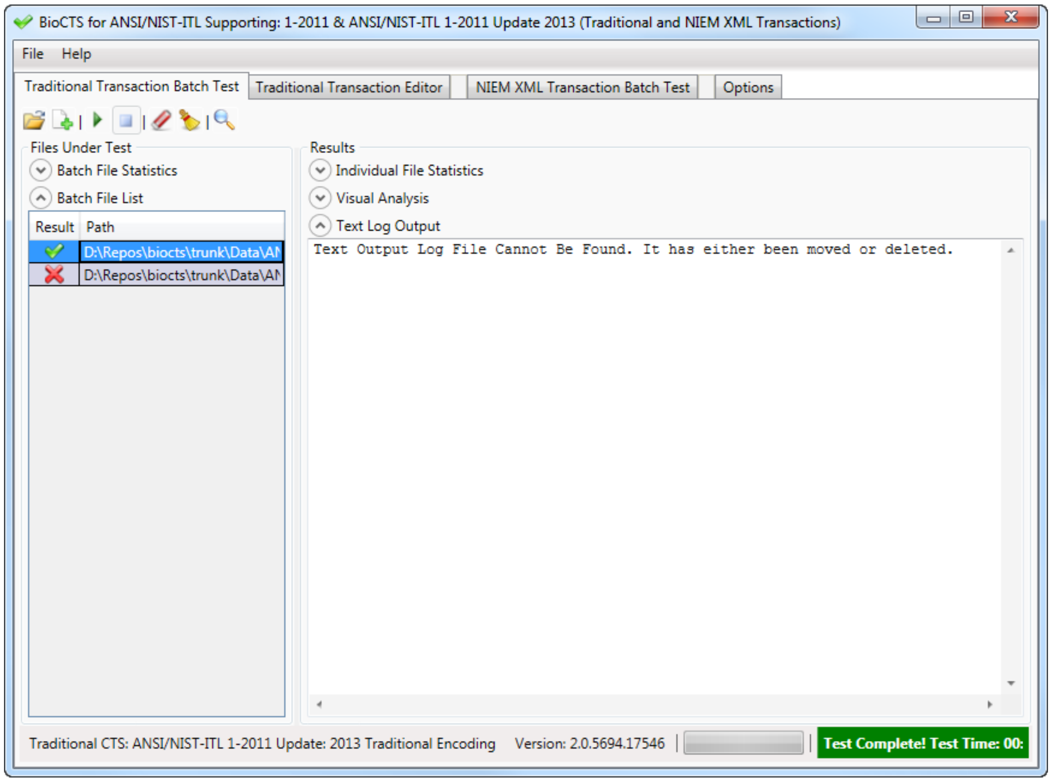
Traditional CTS: ANSI/NIST-ITL 1-2011 Update: 2013 Traditional Encoding Version: 2.0.5694.17546 Send Feedback

Figure 25 – Error: Additional data saved as Unknown Record Type

## *Batch & Testing*

When testing under batch mode, there are a few common errors that are encountered due to the way the tools processes and reports logged test results.

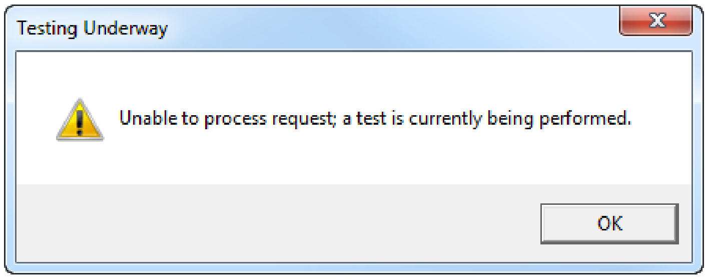
**Log File Missing:** The Batch Tester relies on saved text data log files to report results to the screen in the user interface. If those log files are deleted or moved, or if permission settings do not allow access to the folder where they are saved, the software may report the following error:


BioCTS for ANSI/NIST-ITL Supporting: 1-2011 & ANSI/NIST-ITL 1-2011 Update 2013 (Traditional and NIEM XML Transactions)
File
Help
Traditional Transaction Batch Test
Traditional Transaction Editor
NIEM XML Transaction Batch Test
Options
Files Under Test
Batch File Statistics
Batch File List
Result
Path
D:\Repos\biocts\trunk\Data\AN
D:\Repos\biocts\trunk\Data\AN
Results
Individual File Statistics
Visual Analysis
Text Log Output
Traditional CTS: ANSI/NIST-ITL 1-2011 Update: 2013 Traditional Encoding
Version: 2.0.5694.17546
Test Complete! Test Time: 00:

```
Text Output Log File Cannot Be Found. It has either been moved or deleted.
```

Figure 26 – Error: Text Output Log File Cannot Be Found / Missing

**Testing Underway:** The Batch Tester is designed to be responsive, allowing the user to interact with the tool while tests are running. However, some features cannot run concurrently with Batch testing. For example, if the user attempts to run the Editor while the Batch Testing is in progress, the following error message may be displayed:



Unable to process request; a test is currently being performed.

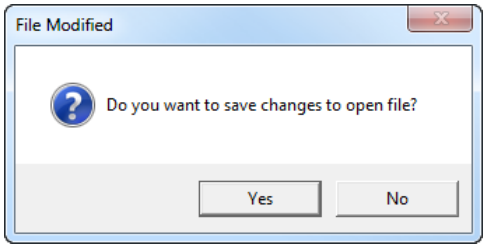
Figure 27 – Error: Testing Underway - A test is currently being performed

## *Editor*

The Editor provides robust features for viewing and editing data at all levels of the transaction, which may introduce some confusion in the way it processes handles modified data. It is important to note that many options cause the Auto Fix to run automatically to avoid parsing and data length errors.

**Operation:** The Editor works by reading in data from the file and creating a copy of that data as structured Records, Fields, Subfields, and Information Items in the computer memory. When the data is modified, it is changed only in memory, not on disk. Any changes must be saved to disk using the Save button, or they will be lost after closing the tool. When the test/run button is pressed, the data in memory is written out to binary/ASCII form in memory and reparsed to simulate reading the newly modified data from a file. This method increases test time by a small margin, but ensures that all errors are caught, even if they are testing during parsing. When the Auto Fix button is pressed, common errors are checked, including Record Header Lengths and Transaction Content (Field 1.003). If these common parsing errors are found, the contents of the Records are used to correct the header/metadata. For example, the length of the modified record is inserted into Field xx.001 / Record Header Length, and its IDC value is inserted into 1.003-IDC.

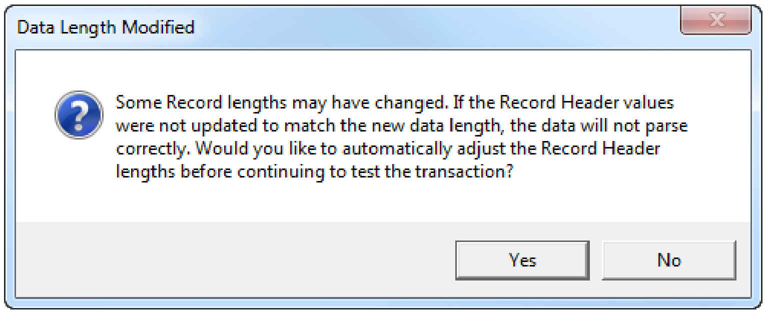
**Save Modified File:** If modifications were made to the transaction in the Editor and the changes were not saved to disk, the software will provide the following prompt if the user attempts to leave the Editor tab:


File Modified

Do you want to save changes to open file?

Figure 28 – Editor Message: File Modified / Save Changes to Open File?

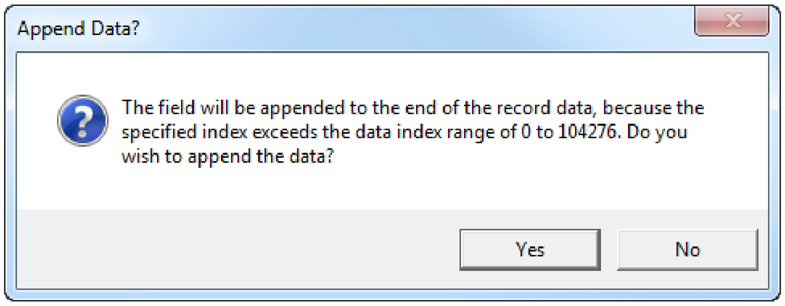
**Data Length Modified:** If modifications made in the Editor may have caused the data length to change and the user attempts to re-test the data, the software will prompt the user to verify that the Record Header lengths have been updated (with the option to automatically adjust it).


Data Length Modified

Some Record lengths may have changed. If the Record Header values were not updated to match the new data length, the data will not parse correctly. Would you like to automatically adjust the Record Header lengths before continuing to test the transaction?

Figure 29 – Editor Message: Data Length Modified – Adjust Record Header Length

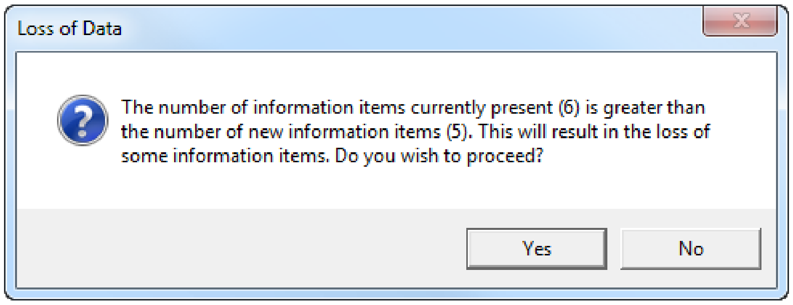
**Append Binary Data:** When adding a binary field, if the data insertion index exceeds the end of the field, the user will be prompted to append the data to the end of the field. Specifying an index beyond the end of the field is also the recommended method for appending new field data.


Append Data?

The field will be appended to the end of the record data, because the specified index exceeds the data index range of 0 to 104276. Do you wish to append the data?

Figure 30 – Editor Message: Append Data when adding Binary Field

**Modify Subfields – Loss of Data:** When modifying Subfields and Information Items, if the new total number of Information Items is less than the original total, the following warning will be displayed to avoid unintentional data loss:


Loss of Data

The number of information items currently present (6) is greater than the number of new information items (5). This will result in the loss of some information items. Do you wish to proceed?

Figure 31 – Editor Message: Modify Subfields Loss of Data

## *Options*

The Options tab is disabled during testing in the Editor or Batch to prevent undesired changes that may negatively affect the Test Results. Also, the Apply Button has been removed due to user feedback; all changes made in the Options Tab are automatically applied.

## 6. References

1. NIST Special Publication 500-290 Data Format for the Interchange of Fingerprint, Facial & Other Biometric Information
http://biometrics.nist.gov/cs_links/standard/AN_ANSI_1-2011_standard.pdf
2. NIST Special Publication 500-290 Rev1 (2013) Data Format for the Interchange of Fingerprint, Facial & Other Biometric Information
http://biometrics.nist.gov/cs_links/standard/ansi_2012/Update-Final_Approved_Version.pdf
3. ANSI/NIST-ITL Standard Homepage
https://www.nist.gov/programs-projects/ansinist-itl-standard
4. NIST SP 500-295 Revision 1 – Conformance Testing Methodology for ANSI/NIST-ITL 1-2011, Data Format for the Interchange of Fingerprint, Facial & Other Biometric Information
http://csrc.nist.gov/groups/ST/BiomResCenter/CTA_BETA/DRAFT_NIST_SP_500_295_Revision1_October2013.pdf
5. NIST SP 500-304 – Conformance Testing Methodology Framework for ANSI/NIST-ITL 1-2011 Update: 2013, Data Format for the Interchange of Fingerprint, Facial & Other Biometric Information
http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.500-304.pdf
6. Federal Information Processing Standards Publication 180-4, Secure Hash Standard (SHS)
http://csrc.nist.gov/publications/fips/fips180-4/fips-180-4.pdf
7. SHA-2 Wikipedia Summary
https://en.wikipedia.org/wiki/SHA-2

Links working as of 9/13/2016

## Annex A: Schema Modifications

In cases where the officially released ANSI/NIST-ITL Schema files interfere with conformance of the base standard requirements or are more restrictive than the base standard, the BioCTS team made modifications necessary to ensure proper testing of the transactions. Some examples include but are not limited to invalid value restrictions and enumerated values, missing or invalid element definitions, and improper ordering of elements. All modifications to the Schema files are tagged in the Schema files with a comment <!—COMMENT--> that includes the tag "BioCTS". Searching the modified Schema for "BioCTS" will locate all of the modifications.

### Current Modified Schema Files - All releases AFTER Beta 2.0.5724.16724:

For AN-2013, no modifications are made to the XML Schema files posted on the ANSI/NIST-ITL Standard Website (http://biometrics.nist.gov/cs_links/standard/ansi_2015/schema-MRT/ANSI-NIST-ITL-2013_FINAL.zip). Modifications are no longer necessary because many of the BioCTS team's recommendations were adopted in the latest update to the XML Schema.

For AN-2011, the modifications in this XML Schema are marked with <!--[BioCTS]--->. Searching for [BioCTS] will located the changes. The modifications are located in biometrics.xsd, which is created by using biometrics.errata.xsd (http://biometrics.nist.gov/cs_links/standard/ansi_2012/biometrics.errata.xsd) and adding the following modifications:

- **Modified**: Modified the element `biom:FaceImageAcquisitionProfile` (a child element of the complex type `FaceImageType)` to include `minOccurs="0"`
  *Reason*: Field 10.013 has a min occurrence of 0, max occurrence of 1. By not having the `minOccurs="0"` the Schema was requiring this element to always be present.
  **BEFORE:**
  ```
  <xsd:element ref="biom:FaceImageAcquisitionProfile"/>
  ```
  **AFTER:**
  ```
  <xsd:element ref="biom:FaceImageAcquisitionProfile" minOccurs="0"/><!--BioCTS added minOccurs="0", see field 10.013 descriptions in Table 58 or Annex G-->
  ```
- **Added**: The value 18 for "Unknown friction ridge" to the Simple Type `FingerPositionCodeSimpleType`.
  **Reason**: Value 18 is specified as a valid value in Table 8 of the AN-2011 standard, but was not present in the Schema.
  **ADDED:**
  ```
  <xsd:enumeration value="18"><!--BioCTS Added enumeration 18, see Table 8-->
      <xsd:annotation>
              <xsd:documentation>Unknown friction ridge</xsd:documentation>
      </xsd:annotation>
  </xsd:enumeration>
  ```

### Previous Versions - Beta 2.0.5724.16724 and earlier:

The following sections provide detailed justifications for the XML Schema modifications.

**AN-2011 Schema Modifications (applicable to 2013 Update):**

The following list of modifications was made to the AN-2011 version of the Schema and also retained for the AN-2013 update.

**I. Additions and Modifications to:**
**`subset\niem\domains\biometrics\1.0\biometrics.xsd`**

- **Modified**: Modified the element **`biom:FaceImageAcquisitionProfile`** (a child element of the complex type **`FaceImageType)`** to include **`minOccurs="0"`**
  *Reason*: Field 10.013 has a min occurrence of 0, max occurrence of 1. By not having the **`minOccurs="0"`** the Schema was requiring this element to always be present.
  **BEFORE:**
  ```
  <xsd:element ref="biom:FaceImageAcquisitionProfile"/>
  ```
  **AFTER:**
  ```
  <xsd:element ref="biom:FaceImageAcquisitionProfile" minOccurs="0"/><!--BioCTS added minOccurs="0", see field 10.013 descriptions in
  Table 58 or Annex G-->
  ```
- **Added**: The value 18 for "Unknown friction ridge" to the Simple Type **`FingerPositionCodeSimpleType`**.
  **Reason**: Value 18 is specified as a valid value in Table 8 of the AN-2011 standard, but was not present in the Schema.
  **ADDED:**
  ```
  <xsd:enumeration value="18"><!--BioCTS Added enumeration 18, see Table 8-->
      <xsd:annotation>
              <xsd:documentation>Unknown friction ridge</xsd:documentation>
      </xsd:annotation>
  </xsd:enumeration>
  ```
- **Modified**: The Element **`FingerprintImageStitchedIndicator`** to only accept the character "Y".
  **Reason**: The Element was defined in the Schema as a Boolean type (**`true`** or **`false`**). This prevented the base requirement of only allowing a single alphabetic character of "Y" (Field 14.027 in Table 71 of the AN-2011 standard).
  **BEFORE:**
  ```
  <xsd:element name="FingerprintImageStitchedIndicator" type="niem-xsd:boolean">
      <xsd:annotation>
              <xsd:documentation>True if the image was formed by stitching together separately captured images; false
  otherwise.</xsd:documentation>
      </xsd:annotation>
  </xsd:element>
  ```
  **AFTER:**
  ```
  <xsd:element name="FingerprintImageStitchedIndicator"> <!--BioCTS changed to allow Y only, see 14.027 description in Table 92-->
      <xsd:simpleType>
              <xsd:annotation>
                      <xsd:documentation>Y if the image was formed by stitching together separately captured images; not present
  otherwise.</xsd:documentation>
              </xsd:annotation>
              <xsd:restriction base="xsd:string">
  ```

```
                    <xsd:enumeration value="Y">
                            <xsd:annotation>
                                    <xsd:documentation>The image was formed by stitching together separately captured
images.</xsd:documentation>
                            </xsd:annotation>
                    </xsd:enumeration>
            </xsd:restriction>
    </xsd:simpleType>
</xsd:element>
```

**Additional AN-2011: Update 2013 Schema Modifications:**

The following list of modifications was made to the AN-2011: Update 2013 version of the Schema to address instances where the Schema conflicted with the base standard in a way that interfered with conformance testing.

- **Added**: Before line 2397, added support for Field 10.039 (`biom:PhysicalFeatureReferenceIdentification`)
  **Reason**: The base standard text states that 10.039 should be allowed for any image type (it is not allowed for FaceImageType in the original Schema).
  **Added XML Content:**
  ```
  <xsd:element ref="biom:PhysicalFeatureReferenceIdentification" minOccurs="0"/><!--BioCTS added (10.039 can be used for any IMT
  type) -->
  ```
- **Added**: After line 2397, added support for Fields 10.034, 10.046, 10.047, 10.048, and 10.051. (`biom:ImageCaptureDateEstimateRange, biom:SubjectExistentialDetails, nc:OrganizationUnitName, biom:PatternedInjuryDetail, biom:RulerScalePresenceInformation`)
  **Reason**: The base standard text allows these fields for any image type (they are not allowed for FaceImageType in the original Schema).
  **Added XML Content:**
  ```
  <!--BioCTS added support for 2013 fields (base standard does not exclude these fields from Face types)-->
  <xsd:element ref="biom:ImageCaptureDateEstimateRange" minOccurs="0"/>
  <xsd:element ref="biom:SubjectExistentialDetails" minOccurs="0"/>
  <xsd:element ref="nc:OrganizationUnitName" minOccurs="0"/>
  <xsd:element ref="biom:PatternedInjuryDetail" minOccurs="0"/>
  <xsd:element ref="biom:RulerScalePresenceInformation" minOccurs="0"/>
  ```
- **Modified**: Starting at line 12469, modified enumeration values.
  **Reason**: The value <> (less than greater than) is indicated by the base standard, but the Schema only allows LTGT. This modification allows the <> value instead.
  **BEFORE:**
  ```
  <xsd:simpleType name="LipCharacterizationCodeSimpleType">
          <xsd:annotation>
                  <xsd:documentation>A data type for lip print characterization code</xsd:documentation>
          </xsd:annotation>
          <xsd:restriction base="xsd:token">
                  <xsd:enumeration value="I">
                          <xsd:annotation>
                                  <xsd:documentation>A clear cut groove running vertically across the lip</xsd:documentation>
  ```

```
            </xsd:annotation>
        </xsd:enumeration>
        <xsd:enumeration value="IP">
            <xsd:annotation>
                <xsd:documentation>Partial-length groove of Type I</xsd:documentation>
            </xsd:annotation>
        </xsd:enumeration>
        <xsd:enumeration value="II">
            <xsd:annotation>
                <xsd:documentation>Branched groove</xsd:documentation>
            </xsd:annotation>
        </xsd:enumeration>
        <xsd:enumeration value="III">
            <xsd:annotation>
                <xsd:documentation>An intersected groove</xsd:documentation>
            </xsd:annotation>
        </xsd:enumeration>
        <xsd:enumeration value="IV">
            <xsd:annotation>
                <xsd:documentation>A reticular groove</xsd:documentation>
            </xsd:annotation>
        </xsd:enumeration>
        <xsd:enumeration value="LTGT">
            <xsd:annotation>
                <xsd:documentation>Indicator for center of the lip</xsd:documentation>
            </xsd:annotation>
        </xsd:enumeration>
        <xsd:enumeration value="O">
            <xsd:annotation>
                <xsd:documentation>Other pattern(s)</xsd:documentation>
            </xsd:annotation>
        </xsd:enumeration>
    </xsd:restriction>
</xsd:simpleType>
```

**AFTER:**

```
<xsd:simpleType name="LipCharacterizationCodeSimpleType"><!--BioCTS modified to allow value '<>'-->
    <xsd:annotation>
        <xsd:documentation>A data type for lip print characterization code</xsd:documentation>
    </xsd:annotation>
    <xsd:union>
        <xsd:simpleType>
            <xsd:restriction base="xsd:token">
                <xsd:enumeration value="I">
                    <xsd:annotation>
                        <xsd:documentation>A clear cut groove running vertically across the
lip</xsd:documentation>
                    </xsd:annotation>
                </xsd:enumeration>
                <xsd:enumeration value="IP">
                    <xsd:annotation>
                        <xsd:documentation>Partial-length groove of Type I</xsd:documentation>
                    </xsd:annotation>
                </xsd:enumeration>
                <xsd:enumeration value="II">
                    <xsd:annotation>
                        <xsd:documentation>Branched groove</xsd:documentation>
                    </xsd:annotation>
```

```
                                        </xsd:enumeration>
                                        <xsd:enumeration value="III">
                                                <xsd:annotation>
                                                        <xsd:documentation>An intersected groove</xsd:documentation>
                                                                </xsd:annotation>
                                        </xsd:enumeration>
                                        <xsd:enumeration value="IV">
                                                <xsd:annotation>
                                                        <xsd:documentation>A reticular groove</xsd:documentation>
                                                </xsd:annotation>
                                        </xsd:enumeration>
                                        <xsd:enumeration value="O">
                                                <xsd:annotation>
                                                        <xsd:documentation>Other pattern(s)</xsd:documentation>
                                                </xsd:annotation>
                                        </xsd:enumeration>
                                </xsd:restriction>
                        </xsd:simpleType>
                        <xsd:simpleType>
                                <xsd:restriction base="xsd:token">
                                        <xsd:pattern value="&lt;&gt;"/>
                                </xsd:restriction>
                        </xsd:simpleType>
                </xsd:union>
        </xsd:simpleType>
```

- **Modified**: Starting at line 12647, modified incorrect enumerated values.
  **Reason**: The enumerated values were incomplete/invalid.
  **BEFORE:**

```
<xsd:enumeration value="10">
        <xsd:annotation>
                <xsd:documentation>Cleft lip (cheiloschisis) – bilateral complete</xsd:documentation>
        </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="11">
        <xsd:annotation>
                <xsd:documentation>Piercing – upper lip</xsd:documentation>
        </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="12">
        <xsd:annotation>
                <xsd:documentation>Piercing – lower lip</xsd:documentation>
        </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="13">
        <xsd:annotation>
                <xsd:documentation>Tattoo – upper lip</xsd:documentation>
        </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="14">
        <xsd:annotation>
                <xsd:documentation>Tattoo – lower lip</xsd:documentation>
        </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="15">
        <xsd:annotation>
                <xsd:documentation>Other</xsd:documentation>
        </xsd:annotation>
</xsd:enumeration>
```

**AFTER:**

```
<xsd:enumeration value="10"><!--BioCTS modified values 10-17, 99-->
        <xsd:annotation>
                <xsd:documentation>Cleft lip (cheiloschisis) – unilateral complete - left</xsd:documentation>
        </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="11">
        <xsd:annotation>
                <xsd:documentation>Cleft lip (cheiloschisis) – unilateral complete - right</xsd:documentation>
        </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="12">
        <xsd:annotation>
                <xsd:documentation>Cleft lip (cheiloschisis) – bilateral incomplete</xsd:documentation>
        </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="13">
        <xsd:annotation>
                <xsd:documentation>Cleft lip (cheiloschisis) – bilateral complete</xsd:documentation>
        </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="14">
        <xsd:annotation>
                <xsd:documentation>Piercing – upper lip</xsd:documentation>
        </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="15">
        <xsd:annotation>
                <xsd:documentation>Piercing – lower lip</xsd:documentation>
        </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="16">
        <xsd:annotation>
                <xsd:documentation>Tattoo – upper lip</xsd:documentation>
        </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="17">
        <xsd:annotation>
                <xsd:documentation>Tattoo – lower lip</xsd:documentation>
        </xsd:annotation>
        </xsd:enumeration>
<xsd:enumeration value="99">
        <xsd:annotation>
                <xsd:documentation>Other (describe in LPPT)</xsd:documentation>
        </xsd:annotation>
</xsd:enumeration>
```

- **Modified**: Starting at line 12736, modified incorrect enumerated value.

  **Reason**: The enumerated value in `LipPrintSurfaceCodeSimpleType` was invalid.

  **BEFORE:**

  ```
  <xsd:enumeration value="4">
  ```

  **AFTER:**

  ```
  <xsd:enumeration value="9"><!--BioCTS modified (was incorrect value of 4)-->
  ```

- **Modified**: Starting at line 12783, modified incorrect enumerated value.
  **Reason**: The enumerated value in `LipPrintMediumCodeSimpleType` was invalid.
  **BEFORE:**
  ```
  <xsd:enumeration value="4">
  ```
  **AFTER:**
  ```
  <xsd:enumeration value="9"><!--BioCTS modified (was incorrect value of 4)-->
  ```
- **Modified**: Starting at line 15784, modified the last element in the sequence.
  **Reason**: The element name `biom:CaptureOrganization` should not be used to represent field 12.047. Instead, `nc:OrganizationUnitName` should be used because it matches the description in the base standard text, and is used in other record types to represent field xx.047.
  **BEFORE:**
  ```
  <xsd:complexType name="DentalCaptureType">
  	<xsd:annotation>
  		<xsd:documentation>A data type for a set of information regarding the dental image</xsd:documentation>
  	</xsd:annotation>
  	<xsd:complexContent>
  		<xsd:extension base="s:ComplexObjectType">
  			<xsd:sequence>
  				<xsd:element ref="biom:CaptureLocation" minOccurs="0"/>
  				<xsd:element ref="biom:CaptureUTCDateTime" minOccurs="0"/>
  				<xsd:element ref="biom:CaptureOrganization" minOccurs="0"/>
  			</xsd:sequence>
  		</xsd:extension>
  	</xsd:complexContent>
  </xsd:complexType>
  ```
  **AFTER:**
  ```
  <xsd:complexType name="DentalCaptureType">
  	<xsd:annotation>
  		<xsd:documentation>A data type for a set of information regarding the dental image</xsd:documentation>
  	</xsd:annotation>
  	<xsd:complexContent>
  		<xsd:extension base="s:ComplexObjectType">
  			<xsd:sequence>
  				<xsd:element ref="biom:CaptureLocation" minOccurs="0"/>
  				<xsd:element ref="biom:CaptureUTCDateTime" minOccurs="0"/>
  				<xsd:element ref="nc:OrganizationUnitName" minOccurs="0"/><!--BioCTS modified (was
  previously biom:CaptureOrganization-->
  			</xsd:sequence>
  		</xsd:extension>
  	</xsd:complexContent>
  </xsd:complexType>
  ```
- **Modified**: Starting at line 177, modified the last element in the sequence.
  **Reason**: The element name `biom:CaptureOrganization` should not be used to represent field 22.047. Instead, `nc:OrganizationUnitName` should be used because it matches the description in the base standard text, and is used in other record types to represent field xx.047.
  **BEFORE:**
  ```
  <xsd:complexType name="BiometricCaptureType">
  	<xsd:annotation>
  		<xsd:documentation>A data type for a set of information regarding the captured biometric
  information</xsd:documentation>
  	</xsd:annotation>
  ```

```
        <xsd:complexContent>
                <xsd:extension base="s:ComplexObjectType">
                        <xsd:sequence>
                                <xsd:element ref="biom:CaptureLocation" minOccurs="0"/>
                                <xsd:element ref="biom:CaptureUTCDateTime" minOccurs="0"/>
                                <xsd:element ref="biom:CaptureDate" minOccurs="0"/>
                                <xsd:element ref="biom:CaptureDeviceIdentification" minOccurs="0"/>
                                <xsd:element ref="biom:CaptureDeviceMakeText" minOccurs="0"/>
                                <xsd:element ref="biom:CaptureDeviceModelText" minOccurs="0"/>
                                <xsd:element ref="biom:CaptureDeviceSerialNumberText" minOccurs="0"/>
                                <xsd:element ref="biom:CaptureOrganization" minOccurs="0"/>
                        </xsd:sequence>
                </xsd:extension>
        </xsd:complexContent>
</xsd:complexType>
```

**AFTER:**

```
xsd:complexType name="BiometricCaptureType">
        <xsd:annotation>
                <xsd:documentation>A data type for a set of information regarding the captured biometric
information</xsd:documentation>
        </xsd:annotation>
        <xsd:complexContent>
                <xsd:extension base="s:ComplexObjectType">
                        <xsd:sequence>
                                <xsd:element ref="biom:CaptureLocation" minOccurs="0"/>
                                <xsd:element ref="biom:CaptureUTCDateTime" minOccurs="0"/>
                                <xsd:element ref="biom:CaptureDate" minOccurs="0"/>
                                <xsd:element ref="biom:CaptureDeviceIdentification" minOccurs="0"/>
                                <xsd:element ref="biom:CaptureDeviceMakeText" minOccurs="0"/>
                                <xsd:element ref="biom:CaptureDeviceModelText" minOccurs="0"/>
                                <xsd:element ref="biom:CaptureDeviceSerialNumberText" minOccurs="0"/>
                                <xsd:element ref="nc:OrganizationUnitName" minOccurs="0"/><!--BioCTS modified (was
previously biom:CaptureOrganization-->
                        </xsd:sequence>
                </xsd:extension>
        </xsd:complexContent>
</xsd:complexType>
```

- **Modified**: Starting at line 16036, modified the type of the element to match the standard description.
  **Reason**: The element `biom:SubjectEstimatedDeathDate` should be of type `niem-xsd:dateTime` to accommodate the time values (as indicated by the base standard description for Field 12.006-DEDD).
  **BEFORE:**

```
<xsd:element name="SubjectEstimatedDeathDate" type="nc:DateType">
        <xsd:annotation>
                <xsd:documentation>A date of the subject death</xsd:documentation>
        </xsd:annotation>
</xsd:element>
```

**AFTER:**

```
<xsd:element name="SubjectEstimatedDeathDate" type=" niem-xsd:dateTime"><!—BioCTS changed type (used to be nc:DateType)-->
        <xsd:annotation>
                <xsd:documentation>A date of the subject death</xsd:documentation>
        </xsd:annotation>
</xsd:element>
```

- **Modified**: Starting at line 16031, modified the type of the element to match the standard description.
  **Reason**: The element `biom:SubjectDateRangeText` should be of type `xsd:duration` to accommodate the values indicated by the base standard description for Field 12.006-DRDE.
  **BEFORE:**
```
<xsd:element name="SubjectDateRangeText" type="nc:TextType">
    <xsd:annotation>
            <xsd:documentation>A kind of subject range of death date estimate</xsd:documentation>
    </xsd:annotation>
</xsd:element>
```
  **AFTER:**
```
<xsd:element name="SubjectDateRangeText" type="xsd:duration"><!--BioCTS modified type (was nc:TextType)-->
    <xsd:annotation>
            <xsd:documentation>A kind of subject range of death date estimate</xsd:documentation>
    </xsd:annotation>
</xsd:element>
```
- **Added**: After line 16972, added support for Field 22.999 (`nc:BinaryBase64Object`)
  **Reason**: The base standard allows for Field 22.999 to contain an image, but the Schema does not.
  **Added XML Content:**
```
<!--BioCTS added 22.999-->
<xsd:element ref=" nc:BinaryBase64Object" minOccurs="0"/>
```
- **Added**: After line 16983, added support for Field 22.994 (`biom:SourceExternalFileReferenceText`)
  **Reason**: The base standard allows for Field 22.999 to contain an image, but the Schema does not.
  **Added XML Content:**
```
<xsd:element ref="biom:SourceExternalFileReferenceText" minOccurs="0"/><!--BioCTS added
22.994-->
<!--22.046-->
```

## II. Additions and Modifications to: `exchange\itl.xsd`

- **Modified**: Starting at line 87, modified ordering of elements.
  **Reason**: T2C comes before user defined fields, not after (according to Annex G). Changed ordering of elements defined in `CrossReferencePackageDataRecordType`
  **BEFORE:**
```
<xsd:complexType name="CrossReferencePackageDataRecordType">
    <xsd:annotation>
            <xsd:documentation>A data type for a biometric record including Type 2 cross reference.</xsd:documentation>
    </xsd:annotation>
    <xsd:complexContent>
				<xsd:extension base="itl:PackageDataRecordType">
					<xsd:sequence>
						<xsd:element ref="biom:Type2CrossReferenceIdentification" minOccurs="0"/>
					</xsd:sequence>
				</xsd:extension>
		</xsd:complexContent>
</xsd:complexType>
```

**AFTER:**

```
<xsd:complexType name="CrossReferencePackageDataRecordType"><!--BioCTS modified--T2C comes before User-defined-->
		<xsd:annotation>
				<xsd:documentation>A data type for a biometric record including Type 2 cross reference.</xsd:documentation>
		</xsd:annotation>
		<xsd:complexContent>
				<xsd:extension base="s:ComplexObjectType">
					<xsd:sequence>
						<xsd:element ref="biom:RecordCategoryCode"/>
						<xsd:element ref="biom:ImageReferenceIdentification"/>
						<xsd:element ref="biom:Type2CrossReferenceIdentification" minOccurs="0"/>
						<xsd:element ref="itl:UserDefinedFields" minOccurs="0" maxOccurs="unbounded"/>
					</xsd:sequence>
				</xsd:extension>
		</xsd:complexContent>
</xsd:complexType>
```

- **Modified**: Starting at line 99, modified ordering of elements.
  **Reason**: Cross references come before user defined fields, not after (according to Annex G).
  Changed ordering of elements defined in
  `MultipleCrossReferencePackageDataRecordType`

**BEFORE:**

```
<xsd:complexType name="MultipleCrossReferencePackageDataRecordType">
		<xsd:annotation>
				<xsd:documentation>A data type for a biometric record including Type 2, 10 and 22 cross
references</xsd:documentation>
		</xsd:annotation>
		<xsd:complexContent>
				<xsd:extension base="itl:PackageDataRecordType">
					<xsd:sequence>
						<xsd:element ref="biom:Type2CrossReferenceIdentification" minOccurs="0"/>
						<xsd:element ref="biom:Type10CrossReferenceIdentification" minOccurs="0" maxOccurs="99"/>
						<xsd:element ref="biom:Type22CrossReferenceIdentification" minOccurs="0" maxOccurs="99"/>
					</xsd:sequence>
				</xsd:extension>
		</xsd:complexContent>
</xsd:complexType>
```

**AFTER:**

```
<xsd:complexType name="MultipleCrossReferencePackageDataRecordType"><!--BioCTS modified--references before User-defined-->
		<xsd:annotation>
				<xsd:documentation>A data type for a biometric record including Type 2, 10 and 22 cross
references</xsd:documentation>
		</xsd:annotation>
		<xsd:complexContent>
				<xsd:extension base="s:ComplexObjectType">
					<xsd:sequence>
						<xsd:element ref="biom:RecordCategoryCode"/>
						<xsd:element ref="biom:ImageReferenceIdentification"/>
						<xsd:element ref="biom:Type2CrossReferenceIdentification" minOccurs="0"/>
```

```
                                        <xsd:element ref="biom:Type10CrossReferenceIdentification" minOccurs="0" maxOccurs="99"/>
                                        <xsd:element ref="biom:Type22CrossReferenceIdentification" minOccurs="0" maxOccurs="99"/>
                                        <xsd:element ref="itl:UserDefinedFields" minOccurs="0" maxOccurs="unbounded"/>
                                </xsd:sequence>
                        </xsd:extension>
                </xsd:complexContent>
        </xsd:complexType>
```

- **Modified**: Starting at line 506, removed the reference to `biom:SourceExternalFileReferenceText`.
  **Reason**: `biom:SourceExternalFileReferenceText`user is already listed in `biom:NonPhotographicImageryDetail`, this listing was a duplicate.
  **REMOVED:**
  ```
  <xsd:element ref="biom:SourceExternalFileReferenceText" minOccurs="0"/>
  ```
- **Modified**: Starting at line 507, removed the reference to `biom:SourceImage`.
  **Reason**: `biom:SourceImage` user is a segmented image type that is not used in Type-22 records. Also, `nc:BinaryBase64Object`  is already specified for the image field 22.999.
  **REMOVED:**
  ```
  <xsd:choice>
          <xsd:element ref="biom:SourceImage"/>
  </xsd:choice>
  ```