

TxToolbox for ODX Manual

Version: 1.4
Issued: 12/05/2005
State: Release

Revision History

| Version | Date | Editor | Revisions / Comment |
|---------|------------|--------------------|---|
| 1.0 | 11/25/2005 | Alexander Heinrich | Initial version |
| 1.1 | 11/30/2005 | Alexander Heinrich | Batch mode & typo |
| 1.2 | 12/02/2005 | Alexander Heinrich | Cover redesigned & disclaimer added |
| 1.3 | 12/14/2005 | Alexander Heinrich | Chapter 8 – Java/C++ API |
| 1.4 | 12/21/2005 | Alexander Heinrich | Readapted for changed GUI, introduced TxDiff rules, Batch API: Correlation of -odx and -pdx parameters |

Disclaimer

All information provided in this document comes without any warranties, expressed or implied, of quality, correctness, performance, merchantability, fitness for a particular purpose or non-infringement.

1 Table of Contents

| | |
|---------------------------------|----------|
| TxToolbox for ODX Manual | i |
| Revision History..... | 1 |
| Disclaimer..... | 1 |
| 1 Table of Contents | 2 |
| 2 The Application | 4 |
| 2.1 System Requirements..... | 4 |
| 2.2 TxCheck..... | 5 |
| 2.2.1 TxCheck Rules | 5 |
| 2.3 TxDiff | 5 |
| 2.3.1 TxDiff Rules..... | 5 |
| 2.4 Projects..... | 5 |
| 3 The Menu Bar | 7 |
| 3.1 The File Menu | 7 |
| 3.1.1 Open ODX Files..... | 7 |
| 3.1.2 Open PDX File | 7 |
| 3.1.3 Open Project..... | 7 |
| 3.1.4 Save Project | 8 |
| 3.1.5 Close..... | 8 |
| 3.1.6 Exit | 8 |
| 3.2 The Edit Menu | 8 |
| 3.2.1 Set Active File | 8 |
| 3.3 The Checker Menu | 8 |
| 3.3.1 Check File..... | 9 |
| 3.3.2 Check All Files..... | 9 |
| 3.3.3 Report CheckResult | 9 |
| 3.3.4 Report RuleSet | 9 |
| 3.3.5 Reload Rules | 10 |
| 3.4 The Differ Menu..... | 10 |
| 3.4.1 Compare | 10 |
| 3.4.2 Report Result | 10 |
| 3.5 The Settings Menu..... | 11 |
| 3.5.1 Preferences..... | 11 |

| | | |
|----------|------------------------------------|-----------|
| 3.6 | The View Menu..... | 12 |
| 3.6.1 | Checker | 12 |
| 3.6.2 | Differ | 12 |
| 3.7 | The Help Menu..... | 12 |
| 3.7.1 | Help | 12 |
| 3.7.2 | Info | 12 |
| 4 | The Toolbar | 13 |
| 5 | The Checker View | 14 |
| 5.1 | File Panel | 14 |
| 5.2 | Console | 15 |
| 5.3 | Results Panel | 15 |
| 5.3.1 | Messages Tab | 15 |
| 5.3.2 | Rule Set Tab | 16 |
| 6 | The Differ View | 17 |
| 6.1 | Project Panels..... | 17 |
| 6.2 | Results Panel | 18 |
| 6.2.1 | Messages Tab | 18 |
| 6.2.2 | DIAG-SERVICES / DATA-OBJECTS | 18 |
| 6.3 | Details Panel | 18 |
| 6.3.1 | Console..... | 18 |
| 6.3.2 | Tree-Views | 18 |
| 6.3.3 | Two-Tree-View | 19 |
| 6.3.4 | Single-Tree-View | 20 |
| 7 | Batch Mode | 21 |
| 7.1 | Syntax..... | 21 |
| 7.2 | Examples | 22 |
| 8 | Licenses | 23 |
| 9 | References | 24 |

2 The Application

The TxToolbox application is a Java™ based toolset for developers of ASAM MCD-2D(ODX) / ISO22901 documents and provides the user with functionality for verifying the compliance of such documents to the ODX XML schema and a set of external (pluggable) rules (→ [TxCheck](#)) as well as for comparing documents with each other (→ [TxDiff](#)).

The TxCheck Tool comes with a set of rules that were defined by the ODX Joint Expert Group (JEG) and that are part of the ODX standard. These rules can be easily recognized by the "ASAM"-prefix in the filename - their numbering is analogous to that of the rule definitions in the standard.

The developer can extend or tailor the way the TxDiff Tool compares documents by providing his own rules for TxDiff (these are different rules than for TxCheck!).

2.1 System Requirements

It is recommended to run the TxToolbox application on a system with at least 512 MB RAM and a 1.6 GHz CPU. Nevertheless the application is likely to run on less powerful systems as well, but with decreased performance.

Since the application is based upon Java™ technology it should be able to run on any system that has a Java Runtime Environment (JRE) installed. Anyway the TxToolbox distribution comes in combination with its own JREs for MS Windows™ and UNIX platforms that will be installed alongside to possibly existing JREs.

2.2 TxCheck

The TxCheck Tool is the part of the application that is related to the verification of ODX documents.

This sub tool of the TxToolbox application can be selected from the View menu in the menu bar.

2.2.1 TxCheck Rules

The TxCheck Tools verification algorithm is based upon external rules. Rules are independent Java™ classes which the TxCheck Tool loads from a specific directory (→ [Preferences...](#)). When a check is performed on an ODX document the TxCheck Tool passes the document to each of the rules one after another which on their parts analyse the passed document and perform the actual compliance checking. If a rule detects a violation it will report this violation to the TxCheck Tool which presents it to the user in form of a message (→ [Messages Tab](#)). A rule will usually regard only a very small aspect of a document, e.g. whether the content of a certain element matches a predefined regular expression, whether links reference valid targets or even simple range checking for values. Nevertheless a rule can detect more than one (qualitative and quantitative) violation, meaning that it does not abort document analysis after a violation has been detected (=quantitative) and that it can distinguish between several characteristics of the same violation (=qualitative) which are identified by a sub-code.

2.3 TxDiff

The TxDiff Tool compares documents or parts of documents (e.g. Services or DOPs) to each other and lists the differences. It's capability to distinguish between real changes on an object basis and changes that only arise from an altered XML structure, but mean no change (e.g. changed IDs) can be utilized by the user. It also presents the compared data to the user in specialised views that reflect the compared datas structure and highlight changes for quick recognition.

2.3.1 TxDiff Rules

The TxDiff Tools comparison algorithm can be extended and tailored by external rules. Such rules are (just as TxCheck rules) Java™ based and will be dynamically loaded at the time the application starts. They are designed to match a certain group of elements and in case the TxDiff Tool encounters elements of that group during a comparison it will pass them to the appropriate rule instead of handling them itself. The TxDiff Tool recognizes rules for XML comparison (→ [XML comparison](#)) and object comparison (→ [Object comparison](#)).

2.4 Projects

A project is basically a set of ODX documents that are identified by their absolute path on the file system. The user can save a project (.cpr extension) and reload it later. This makes it easier to handle groups of files.

Note: The TxDiff Tool makes use of two projects! Thus saving a project (→ [Save Project](#)) while in TxDiff view will save only the currently active one!

3 The Menu Bar

The menu bar is located at the very top of the application window. The whole functionality of the TxToolbox is accessible via the menu bar.



Fig. 3.1 - The Menu Bar

3.1 The File Menu

3.1.1 Open ODX Files...

Opening ODX documents is easy: The user selects this menu item and chooses one or more documents from the file chooser dialog. The documents will then be loaded into the project. If the selected documents are numerous or the documents are large, the loading may take a few seconds. The user can add other documents to the project by repeating this procedure, but it is not possible to open a PDX after other documents have already been loaded. If the user tries to load a PDX after other documents have been loaded these will be unloaded before the PDX is opened.

For this menu item a speed button exists in the toolbar. The shortcut for this menu item is Ctrl-O.

3.1.2 Open PDX File

The TxToolbox supports the Packed ODX (PDX) format. Selecting this menu item will pop up a file chooser dialog from which the user chooses the PDX he wants to open. The file will then be unpacked and the contained ODX documents are loaded into the project. Only one PDX can be loaded at a time. Further more, if you open a PDX while having other documents loaded, all these documents will be unloaded before the PDX is opened. Nevertheless it is possible to add other ODX documents to a project when having a PDX opened by selecting the "Add ODX" menu item or by pressing its corresponding speed button in the toolbar.

For this menu item a speed button exists in the toolbar. The shortcut for this menu item is Ctrl-P.

3.1.3 Open Project

This will bring up the file chooser dialog allowing the user to choose a project file. The files referenced by this project will then be loaded into the TxToolbox. Since the files of a project are not stored in its .cpr file it may happen that the application cannot find the referenced documents (e.g. if they have been deleted or the project is being opened from another system where the referenced files do not exist). In this case an error message will be sent to the console for every file that's missing.

3.1.4 Save Project

This menu item lets the user choose a name and destination for a project file (.cpr extension) from the file chooser dialog which will store the absolute paths to all files that are currently loaded (but not the files!). Project files intend to make it easier for the user to repeatedly handle multiple documents that are possibly located in different directories of the file system.

Note that a project file alone is insufficient - the referenced PDX/ODX documents are always required!

3.1.5 Close

This will unload all currently opened documents. Because documents may be dependent from each other it is not possible to close only a selection of files. Close also clears the console (→ [Console](#)) as well as the results of a previous check or diff.

For this menu item a speed button exists in the toolbar. The shortcut for this menu item is Ctrl-Q.

3.1.6 Exit

This will unload all documents, discard all unreported results and close the TxToolbox application.

The shortcut for this menu item is ESC.

3.2 The Edit Menu

3.2.1 Set Active File

If the current view is the Checker view, this lets the user set the active file of the project. The active file is the only one which will be checked if the user invokes the rule checking mechanism (→ [Check Rules](#)).

It may be easier to select the active file by a simple click on it in the File panel (→ [File Panel](#)).

Note that selecting the active file does not clear the check results. In order not to misinterpret the results the user is advised to take a look at the console - it contains detailed information about the last checked file.

The shortcut for this menu item is F9.

3.3 The Checker Menu

This menu is only enabled if the application is showing the Checker view which can be selected from the View menu. Otherwise it is disabled (grey).

3.3.1 Check File

This will perform a check on the currently active (selected) document. If no document is active, an appropriate message will be sent to the console. Only those rules will be checked that have been loaded and enabled in the Rule Set tab. If the checked document is big or the number of rules is large (i.e. if the checking process takes some time), a progress bar will pop up showing information about how many rules have been processed yet. After all rules have been checked a message will be sent to the console, telling whether any errors or warnings have been found and whether (runtime) errors occurred during the checking process. After a successful check the detailed results (if any) will be displayed in the Messages tab of the checker view. If the Checker view was showing the Rule Set tab it will automatically switch to the Messages tab.

If the current project is empty this menu item is disabled.

For this menu item a speed button exists in the toolbar. The shortcut for this menu item is F7.

3.3.2 Check All Files

This will perform a check on all the currently loaded documents.

The shortcut for this menu item is Shift-F7.

3.3.3 Report CheckResult

After a check the user may save a report about all the detected rule violations (even if there are none). Such a report contains information about the date and time of the check, the checked file and a detailed list of rule violations with the error code, severity (error or warning) and runtime relevance for each as well its textual description and the XPath identifying the flawed elements location in the ODX document. Results can be reported either in XML or HTML format.

If the current project is empty or no rule check has been performed yet this menu item is disabled.

The shortcut for this menu item is Ctrl-Alt-V.

3.3.4 Report RuleSet

The verification of ODX documents is based upon rules (→ [Rules](#)). Each rule has a unique code, a severity, runtime relevance and a (short and long) description explaining the analysis it makes on the data. Further more rules can enabled/disabled separately. Reporting a rule set means to save the details about all currently loaded rules (including the disabled ones) in either an XML or HTML file.

The shortcut for this menu item is Ctrl-Alt-R.

3.3.5 Reload Rules

The TxToolbox application loads all rules that it finds in the specified rule directory (→ [Preferences...](#)) at start-up. Changes that are being made to these rules while the application is running have no effect until the application has either been restarted or the user explicitly reloads the rules by selecting this menu item.

For developers of rules it can be very helpful to reload a rule set without having to restart the TxToolbox application.

3.4 The Differ Menu

This menu is only enabled if the TxToolbox application is showing the Differ view. Select the Differ view from the View menu.

3.4.1 Compare

Invokes the comparison algorithm of the TxDiff Tool. The active file from Project1 will be compared to the active file of Project2 and the differences (if any) will be listed in the Result panel (→ [Results Panel](#)). What the TxDiff Tool regards as a difference and how results are presented to the user depends on the used comparison algorithm. The TxDiff Tool distinguishes between two algorithms:

3.4.1.1 XML Diff

When the user compares documents (Files tab in the Project panel) rather than DiagLayers this algorithm will be used. It compares XML elements one-by-one and collects all differences except those that derive from textual differences but have the same interpretation on XML level. Thus if (for example) the attributes of an element including their values are identical in both documents, but their ordering is different this will not be considered a difference. Another example would be an element with no content which can be written either as `<ELEMENT></ELEMENT>` or as `<ELEMENT/>` - this is not a difference to the algorithm either.

3.4.1.2 Object Diff

This algorithm will be used if the user selected the DiagLayers tab in the Project panel. It compares only DIAG-SERVICES and DOPs (simple and complex) from the selected layers, but unlike the XML Diff it will compare these on a more runtime oriented approach. That means it will not only look at the XML structure of the (wlog) service but also at its related request, responses and DOPs. A changed ID or ID-REF (in odxlink) does not necessarily mean a difference to this algorithm as long as the corresponding element is not changed.

3.4.2 Report Result

The (message-) results of a comparison can be saved to an XML, HTML or XLS (MS Excel) file. A comparison report will contain the names of the compared files, the current date and a list of differences. Each difference has a description and one or two XPath paths identifying the location of the differing elements.

3.5 The Settings Menu

3.5.1 Preferences

The preferences dialog is divided into three tabs, each containing related settings:

3.5.1.1 General

The only general setting which the user can change is the language of the graphical user interface (GUI). Changes to this will not become effective instantly, but when the application is started the next time.

3.5.1.2 Differ

The user may set paths to external rules for file and object comparison here and specify whether external rules shall be activated or not. Changes to this setting will become effective immediately.

3.5.1.3 Checker

On this tab the user may set the language for reports and error messages as well as the directory where the rules are located.

The shortcut for the Preferences dialog is Ctrl-Alt-S.

3.6 The View Menu

This lets the user switch between the TxCheck and the TxDiff Tools, which do not share a common client area.

3.6.1 Checker

Switches to the Checker view.

3.6.2 Differ

Switches to the Differ view.

3.7 The Help Menu

3.7.1 Help

A browsable online help.

3.7.2 Info


Pops up an info dialog providing information about the product version, installed components and copyright.

4 The Toolbar

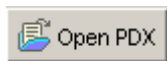
Underneath the menu bar there is the tool bar. It provides the user with buttons for the most frequently used functions of the TxToolbox.



Fig. 4.1 – The Toolbar

The toolbar can be moved to different positions by simply dragging it to one of the four sides of the application window. Dragging the toolbar an empty rectangle will be shown at the mouse pointers position - its frame will become red if docking is possible at this position or remain black otherwise. If the user drops the toolbar while its frame is black, the toolbar will become floating. Clicking on the -button of a floating toolbar will automatically dock it at its last docking position.

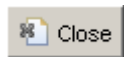
The toolbar contains the following buttons:



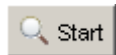
This is a speed button for the Open PDX File... menu item from the File menu.



This is a speed button for the Open ODX Files... menu item from the File menu.



This is a speed button for the Close menu item from the File menu.



This is a speed button for the Check Rules menu item from the Checker menu or for the Compare menu item from the Differ menu depending on the currently active view: If the application is showing the Differ view this button will invoke a comparison, if the application is showing the Checker view this will invoke a rule check.

5 The Checker View

The Checker View is a set of panels that represent data or invoke functionality related to the rule based verification of ODX documents.

You can switch to the Checker View by choosing "Checker" from the "View" menu in the menu bar. The client area will instantly change.

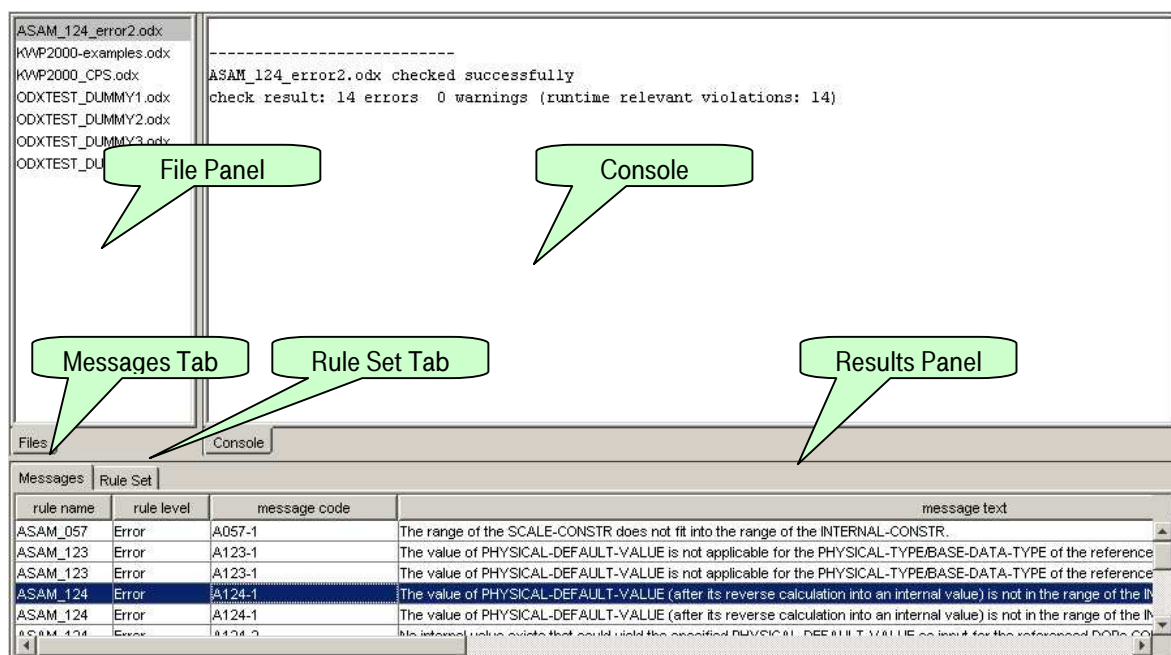


Fig. 5.1 – The Checker views client area

5.1 File Panel

On the upper left side of the client area is the File Panel. It shows all currently loaded ODX documents of the opened project. The active file is the highlighted. The File panel is equivalent to Project1 in the Differ view. Thus if the user invokes a rule check via the Check Rules speed button from the Differ view, the checked document will be the active document from Project1.

5.2 Console

The console is the largest part of the TxCheck Tools client area and is located in the upper right. When working with the TxToolbox application it will regularly send messages to the Console informing the user about errors, warnings or results his actions. The user can clear all messages from the Console by right-clicking on it and selecting "Clear Messages".

5.3 Results Panel

The TxCheck Tools Results panel consists of two tabs:

5.3.1 Messages Tab

After a successful rule check the Messages tab will contain a list of messages that describe the detected rule violations. Each message is described by the name of the rule it was generated by (rule name), its severity (rule level), an error sub-code (message code), its runtime relevance and the message text. Since the message text is usually too long to be displayed in one row the user can double-click on a message to bring up a dialog which presents the information in a more legible form. This dialog additionally contains an XPath to the invalid element.

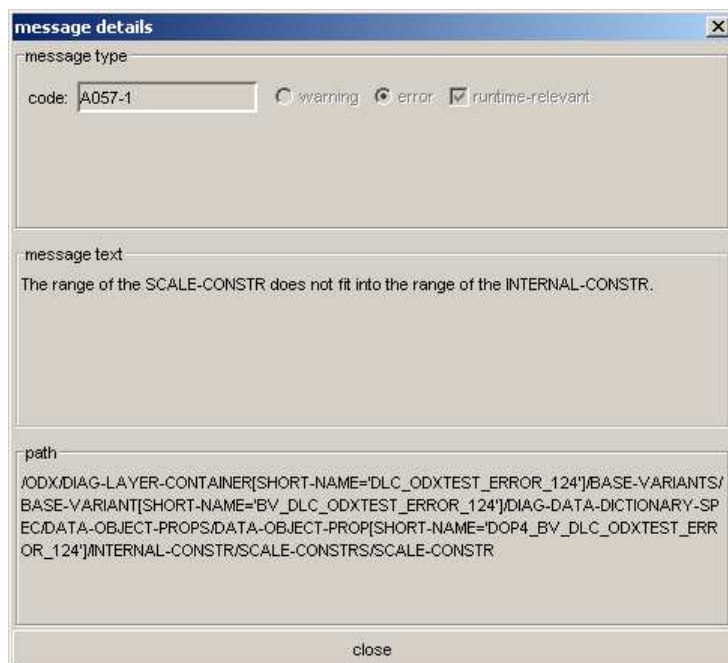
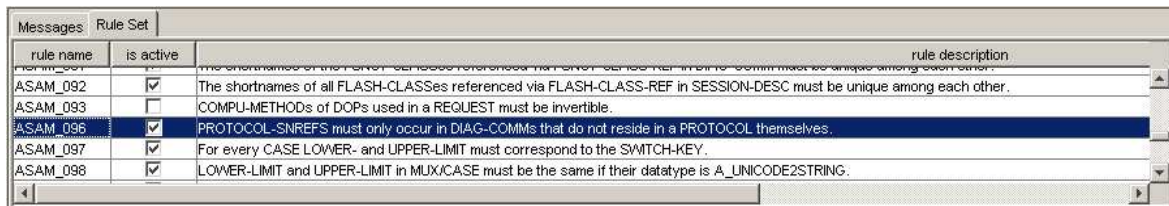


Fig. 5.2 – The message details dialog

5.3.2 Rule Set Tab

The Rule Set tab contains a list of all currently loaded rules. For each rule a name and a short description is provided. Rules can be enabled/disabled by checking/unchecking their "is active"-checkbox. Disabled rules will not be checked during a rule check, but they will be reported in a rule set report. More information about a rule can be gained by double-clicking on it. This will bring up a rule details dialog containing additional information about its severity and runtime-relevance and a more detailed description.



| rule name | is active | rule description |
|-----------|-------------------------------------|---|
| ASAM_092 | <input checked="" type="checkbox"/> | The shortnames of all FLASH-CLASSES referenced via FLASH-CLASS-REF in SESSION-DESC must be unique among each other. |
| ASAM_093 | <input type="checkbox"/> | COMPU-METHODs of DOPs used in a REQUEST must be invertible. |
| ASAM_096 | <input checked="" type="checkbox"/> | PROTOCOL-SNREFS must only occur in DIAG-COMMs that do not reside in a PROTOCOL themselves. |
| ASAM_097 | <input checked="" type="checkbox"/> | For every CASE LOWER- and UPPER-LIMIT must correspond to the SWITCH-KEY. |
| ASAM_098 | <input checked="" type="checkbox"/> | LOWER-LIMIT and UPPER-LIMIT in MUX/CASE must be the same if their datatype is A_UNICODE2STRING. |

Fig. 5.3 – The Rule Set Tab

6 The Differ View

The Differ View is a set of panels and buttons that represent data or invoke functionality related to the comparison of data objects.

Data objects can be either complete ODX documents (Files) or parts of these (e.g. Services or DOPs).

You can switch to the Differ view by choosing "Differ" from the "View" menu in the menu bar. The client area will instantly change.

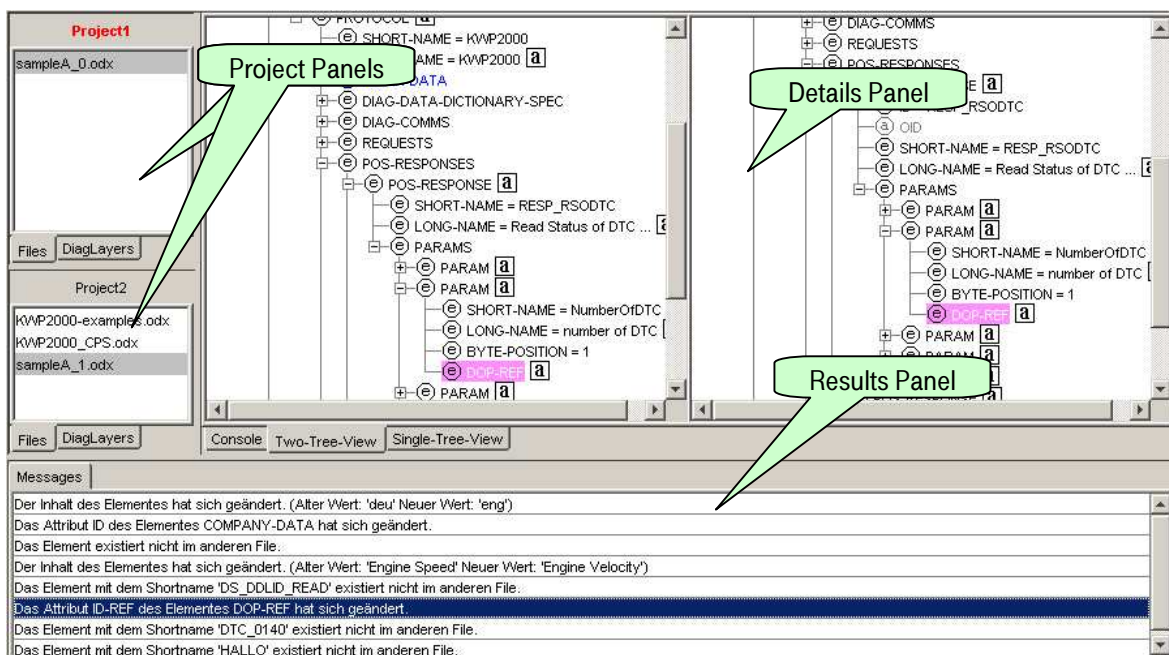


Fig. 6.1 – The Differ views client area

6.1 Project Panels

On the upper left side of the client area are two project panels, Project1 and Project2. A project panel shows all "data objects" of its project. Each project panel has two tabs, Files and DiagLayers, allowing the user to either regard (i.e. compare) each ODX document as a single and closed instance or else to view the embedded DiagLayers of all ODX documents in the project. (→ [Compare](#)). It is not possible to view the files of one project and simultaneously the DiagLayers of another. Thus if the user switches to a different tab in one project the second project will automatically switch to this tab too.

The heading of the active project (i.e. the one which has the focus) is shown in red, the inactive one is shown in black. The user can make a project the active one by simply clicking on it.

Each project has an active data object on which comparisons will be performed - selecting a data object (by clicking on it) in the project panel will make it active.

The user can incrementally add documents to the active project panel by clicking Add ODX or remove all documents by clicking Close.

6.2 Results Panel

On the bottom of the client area is the Results Panel. It contains either the Messages tab (if files were compared) or the tabs DIAG-SERVICES and DATA-OBJECTs (if DiagLayers were compared).

6.2.1 Messages Tab

After a comparison of files this will contain a list of messages, explaining the detected differences of the compared data objects. If no changes were detected the Messages tab remains empty. If the Two- or Single-Tree-View is active in the Details panel a double-click on one of the messages will directly jump to the affected node in the tree.

6.2.2 DIAG-SERVICES / DATA-OBJECTS

If DiagLayers have been compared rather than files the Result Panel will contain these two tabs. The DIAG-SERVICES tab contains the information about differences detected during the comparison of services and the DATA-OBJECTS tab the information about differences detected during the comparison of DOPs. Each tab contains a list of messages, one for every detected difference. Unlike the Messages tab messages of these tabs also have a second field containing the name of the compared service / DOP where the difference was detected.

6.3 Details Panel

The largest (and probably most important) part of the Differ views client area is the Details Panel. The Details Panel consists of multiple tabs:





6.3.1 Console

The Differ views console is the same as the TxCheck Tools console (→ [Console](#)). Thus console messages generated during a check will also appear in the Differs console and vice versa.

6.3.2 Tree-Views

Besides the Console tab one or more (depending on the comparison method) Tree-View tabs are available. Tree-Views are only generated when a comparison has been performed, otherwise they are empty. They generally visualize the compared data objects as tree structures. Nodes of such trees usually represent an element or

an attribute from the ODX documents XML code, but there are exceptions to this rule. The name of a node is (usually) the name of the element or attribute it represents (e.g. the node for an <AUDIENCE> will be named "AUDIENCE" and the node for an attribute OID will also be named "OID"). If an element or attribute has a value (instead of sub-elements) this is appended to the name (e.g. <TROUBLE-CODE>128</TROUBLE-CODE> will yield a tree node called "TROUBLE-CODE = 666". Sometimes values are too long to be displayed on screen - in these cases only the beginning of the value will be shown followed by "...". Hovering the mouse pointer over the node will then pop up a tool tip showing the complete value.

Initially the tree structure of a Tree-View is collapsed. Clicking on the -icon of a node will expand its sub tree, clicking the -icon will collapse it again. If a node has no  or  icon it has no sub-elements.

To make it easier for the user to distinguish nodes in a list of homogeneous elements (e.g. PARAMs) without expanding these it is possible to hover the mouse pointer over such a node and if it has a SHORT-NAME it will be shown as tool tip.

The type of a node (element or attribute) can be determined by its icon:

 for elements

 for attributes

Differences between data objects are illustrated by colours:

If an element has an identical counter-part (i.e. there's no difference) in the other data object it will be displayed black on white background.

If an element has no counter-part in the other data object it will be coloured blue in the tree representing the data object that contains the element (and won't be displayed in the second tree).

Elements that differ from their existing counter-parts are pink.


If an element is marked changed (pink) or single (blue) all of its sub-elements will share this colour.

The following concrete occurrences of Tree-Views can be found in the Differ view:

6.3.3 Two-Tree-View

A Two-Tree-View is a pair of tree structures where the left tree refers to the compared data object from Project1 and the right tree refers to the compared data object of Project2. Two-Tree-Views exist for XML Diffs (Files tab selected in project panel) as well as for Object Diffs (DiagLayers selected in project panel), but their interests are different:

6.3.3.1 Two-Tree-View for XML comparison

Both trees reflect exactly the XML structure of the compared files. Initially the attributes of all elements are hidden, but they can be shown (and hidden again) by clicking on the -button to the right of each node. If a node has no such button it means that it has no attributes. Attributes (if shown) are listed in the order they were defined in the schema. Attributes which are defined in the schema but not specified in the document will be displayed grey. If for such an attribute a default value has been defined in the schema this will also be shown,

otherwise the attribute will be shown with no value. Attributes that don't occur in the concrete ODX document will never be marked changed or inserted.

If a node represents an odxlink and the odxlink refers to an object in the same document it can be followed by clicking on it while having Ctrl pressed. If the odxlink is valid and refers to an object inside the document then the node that corresponds to the referenced element will be selected in the view.

6.3.3.2 Two-Tree-View for Object comparison

Both trees show the structure of the service or DOP which the user selected in the Result panel. The displayed structure will embed all of its related structures (e.g. requests, responses, DOPs). In this view attributes are being shown unconditionally. Unset attributes are being ignored.

6.3.4 Single-Tree-View

This view is only available for XML Diffs. It assumes that the compared documents have major similarities and thus displays them as one document merging their differences together. Elements and attributes that appear white in a Single-Tree-View occur in both documents and are identical. Only differing elements are coloured (→ [Tree-View](#)). To recognize a changed elements origin their icons have indices:

 for the element from Project1

 for the element from Project2

The same applies for single elements (elements that appear in only one document):

 for the element from Project1

 for the element from Project2

7 Batch Mode

To access the functionality of the TxCheck Tool it is not necessary to start the Graphical User Interface (GUI). It is possible to check documents directly from a command prompt.

Note: Checking from a command prompt usually yields a */log4j* warning which can be neglected.

7.1 Syntax

```
java [-Xms<size> -Xmx<size>] -cp ./Checker.jar com.tsystems.checker.Checker
[-pdx <pdx>] -odx <odx>[,<odx>]* -result <out> [-de|-en] [-xml|-html]
```

<pdx> is the (absolute or relative) path to a PDX file.

<odx> is the (absolute or relative) path to an ODX file or a filename referring a file inside a PDX.

<out> is the (absolute or relative) path of the report file.

The above syntax implies that the user must specify one or more input files which can be either one PDX using the **-pdx** parameter or a number of ODX files using the **-odx** parameter and separating files with comma. If the user specifies the **-pdx** parameter he must also specify the **-odx** parameter to tell the TxCheck Tool which document within the PDX he wants to check. In this case the user must pass only one (unqualified) filename for **-odx**. If the user does not specify the **-pdx** parameter but more than one file for the **-odx** parameter, then the passed filenames must be (either absolutely or relatively) qualified and the Checker will assume the first file as the active one. The user must further specify an output file using the **-result** parameter where the check report will be saved. If the specified file already exists it will be overwritten. The report format can be chosen via the optional parameters **-xml** (XML) or **-html** (HTML), default is XML. The language of the report can be chosen via the optional parameters **-de** (german) or **-en** (english), default is english.

If the checked files are very large it may happen that the TxCheck Tool abnormally terminates with an out-of-memory error. In this case the user can assign more memory to the virtual machine using **-Xms** (initial heap size) and **-Xmx** (maximum heap size). What are good values for these parameters depends on the system the TxCheck Tool is being run on. Good values for the system described in [section 2.1](#) are for instance **-Xms100m -Xmx500m**. Generally the user is advised to set these values neither to high nor to low. Assigning more memory than physically available will make the virtual machine swap data to the hard disk and performance will decrease tremendously while assigning to little memory won't solve the out-of-memory problem.

7.2 Examples

Check the ODX document 'KWP2000_examples.odx' and store the report in 'result.xml' using the default settings for output format and language (XML & english):

```
java -cp .;Checker.jar com.tsystems.checker.Checker -odx KWP2000_examples.odx  
-result result.xml
```

Check the ODX documents 'KWP2000_examples.odx' and 'KWP2000_CPS.odx' and save the report in the file 'result.html' with output format and language explicitly set to HTML and german.

```
java -cp .;Checker.jar com.tsystems.checker.Checker  
-odx KWP2000_examples.odx,KWP2000_CPS.odx -result result.html -html -de
```

Check the ODX documents 'KWP2000_examples.odx' contained in the PDX 'KWP2000.pdx':

```
java -cp .;Checker.jar com.tsystems.checker.Checker -pdx KWP2000.pdx -odx  
KWP2000_examples.odx -result result.xml
```

Check the large file 'big.odx'

```
java -Xms100m -Xmx500m -cp .;Checker.jar com.tsystems.checker.Checker -odx big.odx  
-result result.xml
```

8 Licenses

The TxToolbox Application includes software developed by

The Apache Software Foundation (<http://www.apache.org>) and

The DOM4J Project (<http://www.dom4j.org>)

For details about license agreements visit these sites.

9 References

1. ASAM ODX Schema Version 2.0.1:

<http://www.asam.net/xml/odx/2.0.1/odx.xsd>

<http://www.asam.net/xml/odx/2.0.1/odx-xhtml.xsd>

<http://www.asam.net/xml/odx/2.0.1/odx-cc.xsd>

2. ODX Specification ASAM MCD-2D (ODX) 2.0.1 http://www.asam.net/O3_standards_06.php