

Tag Engine

Antidoc v2.0.4, Ryan Porter

Table of Contents

1. Project description	1
2. Libraries	2
2.1. Tag Engine.lvlib	2
2.2. Tag Watchlist.lvlib	4
3. Classes	6
3.1. Classes overview	6
3.2. Tag Core.lvclass	6
3.3. Tag.lvclass	12
3.4. Tag Expr.lvclass	15
4. Legal Information	18
4.1. Document creation	18
4.2. Product used in the project	20

Chapter 1. Project description

The Tag Engine provides a simple API for sharing data across separate code modules.

At its core, a Tag is just a variant value that can be looked up globally by name or by data value reference. However, this API provides additional functionality that helps to reduce code redundancy and avoid common pitfalls.

The general rule for using the tag engine is that the module that creates a tag should be the only module that writes directly to the tag's value. Any other module must REQUEST to change the value of the tag by triggering the tag's SetRequest event. The tag's creator can listen for this event and decide how to handle the request.

Chapter 2. Libraries

This section describes the libraries contained in the project.

2.1. Tag Engine.lvlib

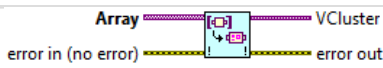


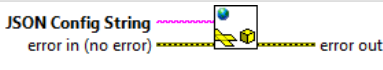
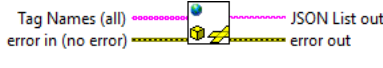



Responsibility: Tag Engine Library contains the tag engine core as well as the tag object definition.

Version: 1.0.0.0

Table 1. Nested libraries

Name	Type
Tag Core.lvclass	LVClass
Tag.lvclass	LVClass

Table 2. Functions (non private scope only)

Name	Connector pane	Description	S.	R.	I.
Array of VData to VCluster_openG		No description found (add content in vi description)			
Build Tags from JSON List		Build multiple Tags from a JSON array of Tag configurations.			
Flatten Tags to JSON List		Create a JSON array of Tag configurations from a list of Tags. If the input list of Tag Names is empty, all Tags will be used to generate the JSON array.			
Init SetRequest Events Cluster		Input a cluster of SetRequest events with labels corresponding to the desired Tag Names. Optionally the specified namespace can be prepended to the cluster labels to form the Tag Names. Output is the same cluster, but with event references populated with Tag's event references			

Name	Connector pane	Description	S.	R.	I.
Init Tag Ref Cluster		<p>Input a cluster of Tag references with labels corresponding to the desired Tag Name.</p> <p>Optionally the specified namespace can be prepended to the cluster labels to form the Tag Name.</p> <p>Output is the same cluster, but with Tag references populated with Tag's actual reference</p>			
Init ValChange Events Cluster		<p>Input a cluster of ValueChange events with labels corresponding to the desired Tag Name.</p> <p>Optionally the specified namespace can be prepended to the cluster labels to form the Tag Name.</p> <p>Output is the same cluster, but with event references populated with Tag's ValChange event references.</p>			
Variant to DBL		<p>Attempt to convert a variant value to Double precision floating point.</p> <p>This might work for the following types: - Boolean: T=1.0, F=0.0 - Unsigned Integer: U8..U64, Enum U8..U64 - Signed Integer: I8..I64 - Floating Point: SGL, DBL, EXT - String: Assuming Frac/Expr string using dot decimal place. - Timestamp: Converts to time in seconds.</p> <p>Unsupported types will return NaN for the value, without producing an error.</p>			
Variant to String		<p>Attempt to represent the input varriant value as a string.</p> <p>This VI just calls "Variant to JSON Text.vi"</p> <p>Returned strings are in JSON-compatible value format.</p>			

Scope: → Protected | → Community

Reentrancy:  → Preallocated reentrancy |  → Shared reentrancy

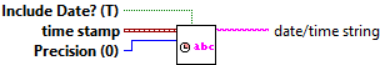
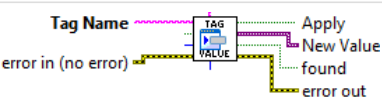





Inlining:  → Inlined

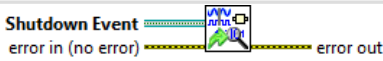

2.2. Tag Watchlist.lvlib


Responsibility: The tag watchlist library contains the Tag Watchlist UI and supporting subVIs.

Version: 1.0.0.0

Table 3. Functions (non private scope only)

Name	Connector pane	Description	S.	R.	I.
Timestamp to String		No description found (add content in vi description)			
Tag_Value_Entry_PopUp		<p>Tag value entry pop-up. By specifying a Tag Name, the tag's value is looked up and the appropriately typed popup is launched.</p> <p>Select either SetRequest or WriteValue for Write Type. Write value should be for debugging and testing purposes only. Set Request send the value by Set Request event instead of writing directly to the Tag's value.</p> <p>Supported Types are: - Boolean - Signed Integer - Unsigned Integer - Floating Point - String</p> <p>Any other type will launch the string value entry popup with the JSON representation of the value. BE CAREFUL.</p>			
Value Entry_BOOL		Boolean value pop-up window. Use spacebar to toggle the value.			
Value Entry_DBL		DBL value entry Pop-up window.			
Value Entry_I32		I32 value entry pop-up window.			
Value Entry_STR		String value entry pop-up window.			
Value Entry_U32		U32 value entry pop-up window.			

Name	Connector pane	Description	S.	R.	I.
Launch Tag Watchlist UI		<p>Tag watchlist launcher. Allows non-blocking launch of Tag Watchlist window (fire and forget), while allowing only one instance to be running.</p> <p>The input Shutdown Event should be used to shutdown the Tag Watchlist window in the event of main application shutdown.</p>			
Tag Watchlist UI		<p>Tag Watchlist window can view and modify any existing Tag.</p> <p>Use the Watchlist launcher to run this VI in your application.</p>			

Scope:  → Protected |  → Community

Reentrancy:  → Preallocated reentrancy |  → Shared reentrancy

Inlining:  → Inlined

Chapter 3. Classes

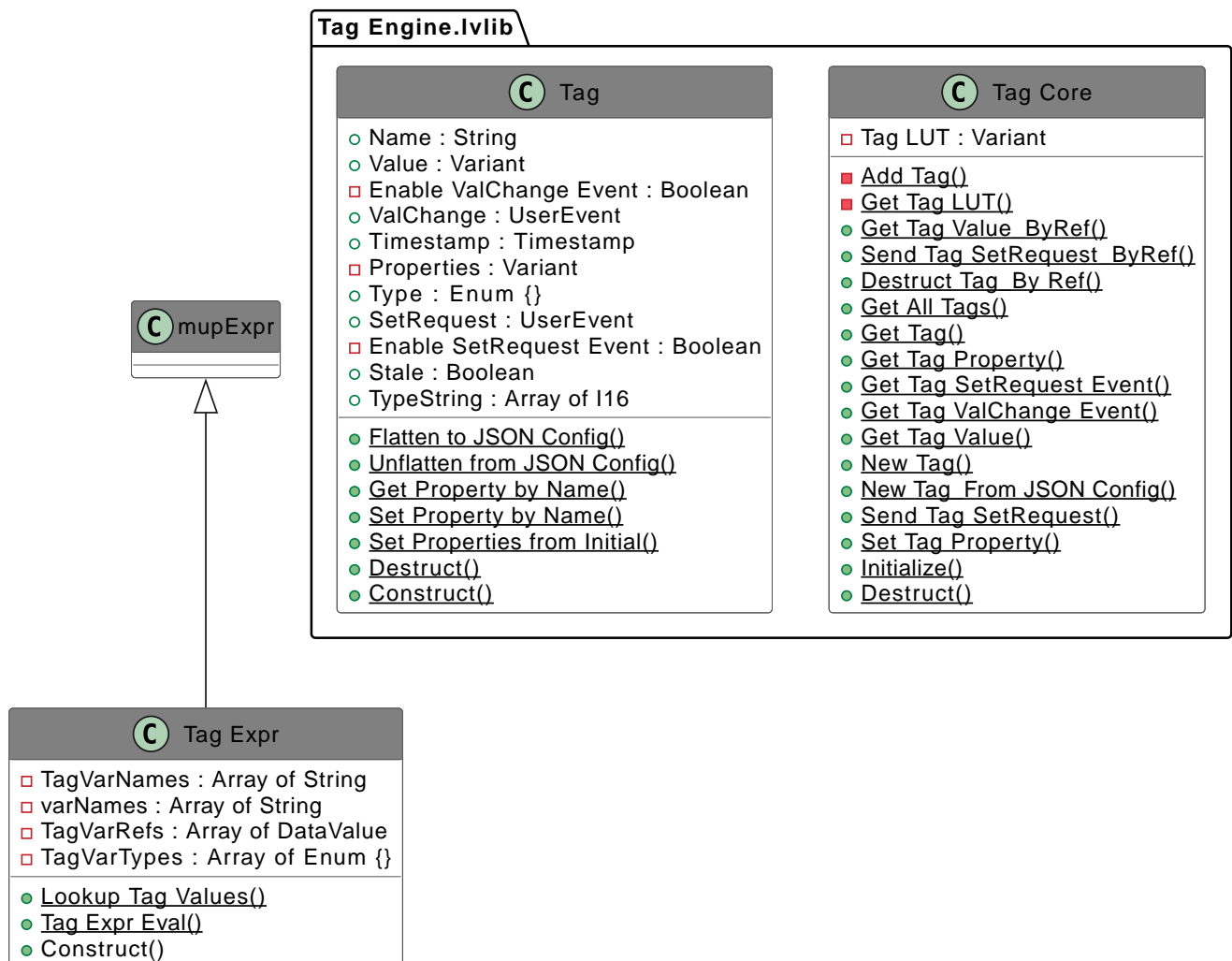
This section describes the classes contained in the project.

3.1. Classes overview

This project contains 3 classes and 0 interface.

Table 4. Classes list

Classes	Interfaces
Tag Core.lvclass	
Tag.lvclass	
Tag Expr.lvclass	



3.2. Tag Core.lvclass

Responsibility: Tag engine core holds the lookup table of all tags. It also defines the public API for interacting with tags.

Version: 1.0.0.0

Tag Engine.lvlib

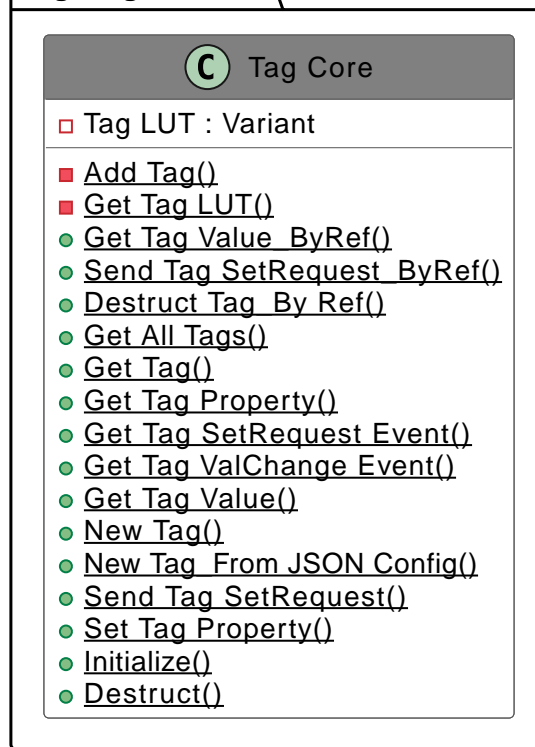


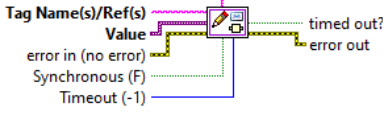


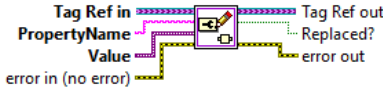


Table 5. Functions (non private scope only)

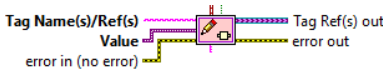




Name	Connector pane	Description	S.	R.	I.
Get Tag Value_ByRef		No description found (add content in vi description)			
Send Tag SetRequest_ByRef		No description found (add content in vi description)			
Write Tag Value_ByRef		No description found (add content in vi description)			
Destruct Tag_By Ref		Destruct a single CVT item by reference. No error is returned if the tag is not found.			
Get All Tags		Return a list of all Tag Names and references. Optionally specify a filter. Only matched Tag names will be returned. Standard "Match Pattern" syntax applies. Match is not case sensitive.			

Name	Connector pane	Description	S.	R.	I.
Get Tag		<p>Lookup the reference of a Tag.</p> <p>Optionally specify a namespace, which is prepended to the tag name (colon separator).</p> <p>If the tag is not found, the found output is false and Error 5602 is returned.</p>			
Get Tag Property		<p>Read a Tag's property.</p> <p>If the property does not exist, error 5602 is returned.</p> <p>Specify relaxed type along with a DBL or String default value to attempt to coerce the returned type to string or DBL.</p> <p>Specify a variant default value to return the property's variant value (untouched)</p>			
Get Tag SetRequest Event		<p>Lookup a Tag's set request event.</p> <p>If the tag is not found, error 5602 is returned.</p>			
Get Tag ValChange Event		<p>Lookup a Tag's value change event.</p> <p>If the CVT item is not found, error 5602 is returned.</p>			

Name	Connector pane	Description	S.	R.	I.
Get Tag Value		<p>Get a Tag's (or a list of tags) current value(s) and timestamp(s).</p> <p>The Tag Name(s)/Ref(s) input can be either the Tag Name, an array of Tag Names, a Tag Reference, or an array of Tag References.</p> <p>Optionally specify a namespace, which is prepended to the tag name(s), using colon as separator.</p> <p>If a tag is not found, the default value is returned along with Error 5602.</p> <p>The Default input defines the expected data type of the value, as well as the default value returned in case of error. If no default value is specified, the value is returned as a variant.</p> <p>Specify relaxed type along with a DBL or String default value to attempt to coerce the returned type to string or DBL.</p>			

Name	Connector pane	Description	S.	R.	I.
New Tag		<p>Create a new Tag</p> <p>The Tag Name, including the optional namespace, can be used for item lookup operations after creation. Namespace separator is a colon. Max length 256 characters. Valid characters are: a-z A-Z 0-9 : _ Although names are stored in original case, lookups are not case sensitive.</p> <p>Optional namespace to be prepended to the Tag name (separated by a colon). Do not specify a namespace if the full tag name had been specified by the Tag Name input.</p> <p>Set init as stale to true to indicate that no real measurement has occurred since the tag was created. Default value is false.</p> <p>Units of measure for numeric-type tags.</p> <p>Optional description of the tag can be specified.</p> <p>Initial value for the Tag also defines the Tag's datatype.</p> <p>Error 5601 is returned if the Tag Name already exists.</p>			
New Tag_From JSON Config		<p>Build a new Tag from a JSON config string.</p> <p>The JSON config string must at least specify the Name, Type, Typestring, and initial value property.</p>			

Name	Connector pane	Description	S.	R.	I.
Send Tag SetRequest	 <p>Tag Name(s)/Ref(s) Value error in (no error) Synchronous (F) Timeout (-1)</p> <p>timed out? error out</p>	<p>Trigger a set value request event on the specified Tag.</p> <p>Specify either the Tag name or Tag reference, or a list of Tag names or references.</p> <p>Optionally specify a namespace to prepend to the Tag name(s).</p> <p>Specify the requested set value (or list of values for multiple Tags)</p> <p>Set synchronous to true to wait (up to the timeout) for a response from the Tag's event handler that the request has been completed. With a timeout of -1, the set request will wait indefinitely. If the request times out, timeout is set to true but NO ERROR IS RETURNED.</p> <p>Set synchronous to false, will trigger the event and move on without waiting for confirmation of completion.</p>			 
Set Tag Property	 <p>Tag Ref in PropertyName Value error in (no error)</p> <p>Tag Ref out Replaced? error out</p>	<p>Set a Tag's property.</p> <p>If the property doesn't exist, it will be created. Otherwise the value will be replaced.</p>			 

Name	Connector pane	Description	S.	R.	I.
Write Tag Value		<p>Write the value of the specified Tag(s).</p> <p>Specify either the Tag name, Tag reference, or a list of Tag names or references.</p> <p>Optionally specify a namespace to prepend to the Tag name(s).</p> <p>Specify the requested set value (or list of values for multiple Tags)</p> <p>Specify a timestamp to associate with the value, otherwise a new one is created (now).</p> <p>Set IgnoreIfEqual to true will ignore the new value if it is equal to the current value (a ValChange event will not be triggered and timestamp will not be updated).</p>			
Initialize		<p>Initialize the Tag store.</p> <p>Call this VI at the startup of your main process (that does not leave memory until your application is stopped). The Tag store will be owned by this process.</p>			
Destruct		<p>Destroy every Tag and then the Tag store</p>			

Scope:  → Protected |  → Community

Reentrancy:  → Preallocated reentrancy |  → Shared reentrancy

Inlining:  → Inlined

3.3. Tag.lvclass

Responsibility: Variant value with some standard properties

Version: 1.0.0.0

Tag Engine.lvlib

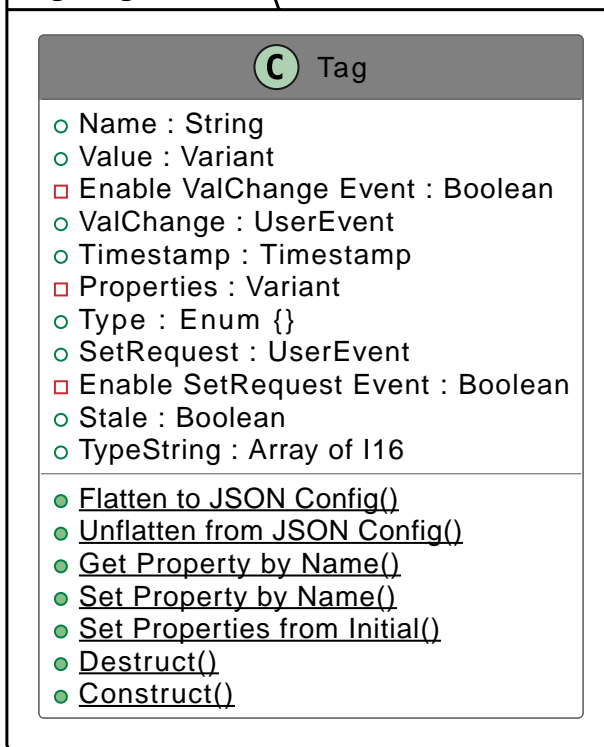
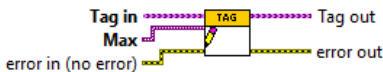


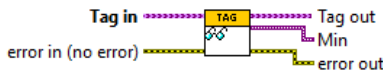


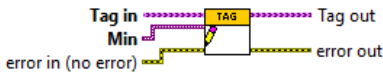


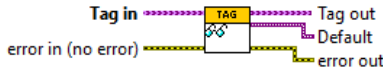


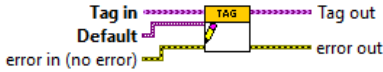





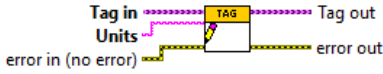


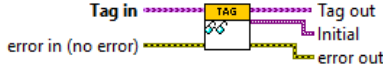





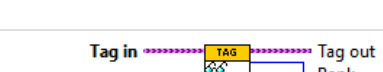





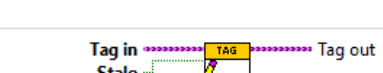


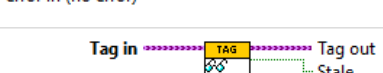









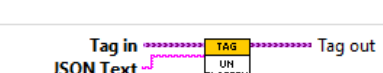





Table 6. Functions (non private scope only)

Name	Connector pane	Description	S.	R.	I.
Read Name		No description found (add content in vi description)			
Read Timestamp		No description found (add content in vi description)			
Read Value		No description found (add content in vi description)			
Write Value		No description found (add content in vi description)			
Read Type		No description found (add content in vi description)			
Read ValChange		No description found (add content in vi description)			
Read Config String		No description found (add content in vi description)			
Read Description		No description found (add content in vi description)			
Write Description		No description found (add content in vi description)			
Read Max		No description found (add content in vi description)			

Name	Connector pane	Description	S.	R.	I.
Write Max		No description found (add content in vi description)			
Read Min		No description found (add content in vi description)			
Write Min		No description found (add content in vi description)			
Read Default		No description found (add content in vi description)			
Write Default		No description found (add content in vi description)			
Read Units		No description found (add content in vi description)			
Write Units		No description found (add content in vi description)			
Read Initial		No description found (add content in vi description)			
Read SetRequest Event		No description found (add content in vi description)			
Read Rank		No description found (add content in vi description)			
Read ElementType		No description found (add content in vi description)			
Write Stale		No description found (add content in vi description)			
Read Stale		No description found (add content in vi description)			
Read Property List		No description found (add content in vi description)			
Read TypeString		No description found (add content in vi description)			
Flatten to JSON Config		No description found (add content in vi description)			
Unflatten from JSON Config		No description found (add content in vi description)			
Get Property by Name		No description found (add content in vi description)			

Name	Connector pane	Description	S.	R.	I.
Set Property by Name		No description found (add content in vi description)			
Set Properties from Initial		No description found (add content in vi description)			
Write SetRequest		No description found (add content in vi description)			
Write Value with Timestamp		No description found (add content in vi description)			
Destruct		No description found (add content in vi description)			
Construct		No description found (add content in vi description)			

Scope: → Protected | → Community

Reentrancy: → Preallocated reentrancy | → Shared reentrancy

Inlining: → Inlined

3.4. Tag Expr.lvclass

Responsibility: The Tag Expr class extends the mupExpr class (from muParser library), adding support for the Tag Engine.

Expressions can include any existing Tag Name. Tag references are cached for faster value lookup during expression evaluation.

Note that Tag values are coerced to DBL before evaluating the expression.

Also note that any mupExpr eval function can be used, however Tag Expr Eval is included in this class for convenience.

Version: 1.0.0.0

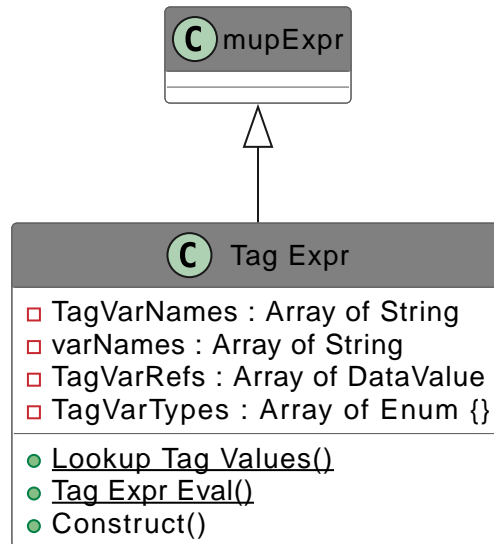







Table 7. Functions (non private scope only)

Name	Connector pane	Description	S.	R.	I.
Construct		<p>Construct a new muExpr instance. Input an expression or a set of expressions. Returns the expression handle and the list of variables found in the expression.</p> <p>Optionally input a list of constant name-value pairs.</p> <p>Set "Use system decimal point" to true if you want to adapt to the system's decimal point. If the system's decimal point is a comma, the argument separator will be changed to a semi-colon. The default decimal point is a period. The default argument separator is a comma.</p> <p>Set Bulk Size sets the range for bulk mode evaluations. Default is 1 (single result).</p>			
Lookup Tag Values		<p>Lookup the current values of all Tags within the expression.</p> <p>Values are coerced to DBL type.</p> <p>Supported types are for Coercion are: - Boolean - Signed Integer - Unsigned Integer - Floating Point - String (with period decimal point) - Timestamp</p> <p>All other types will return NaN with error 5605 reported.</p>			

Name	Connector pane	Description	S.	R.	I.
Tag Expr Eval		<p>This VI looks up current Tag values then evaluates the expression.</p> <p>The Boolean result is true if the numeric result is not equal to zero or NaN.</p>			

Scope:  → Protected |  → Community

Reentrancy:  → Preallocated reentrancy |  → Shared reentrancy

Inlining:  → Inlined

Chapter 4. Legal Information

4.1. Document creation

This document has been generated using the following tools.

4.1.1. Antidoc

Project website: [Antidoc](#)

Maintainer website: [Wovalab](#)

BSD 3-Clause License

Copyright © 2019, Wovalab, All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

4.1.2. AsciiDoc for LabVIEW™

Project website: [AsciiDoc toolkit](#)

Maintainer website: [Wovalab](#)

BSD 3-Clause License

Copyright © 2019, Wovalab, All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

4.1.3. **classy Diagram Viewer**

Project website: [classy Diagram Viewer](#)

BSD 3-Clause License

Copyright © 2021, Tatiana Boyé All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES

(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

4.2. Product used in the project

Antidoc hasn't been able to detect third party products in the project. This is the author's responsibility to list any of the missing product used.