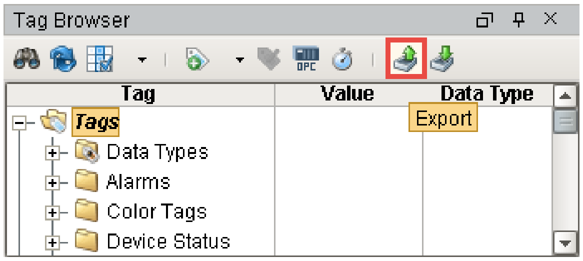
Importing and Exporting Tags

Ignition can export and import Tag configurations to and from **XML** (Extensible Markup Language) or **CSV** (Comma Separated Value) file formats. This allows you to make changes to the Tag structure outside the Designer. The importing and exporting feature also becomes really important if you have a Tag database that you exported from a legacy system that you want to import into an Ignition database.

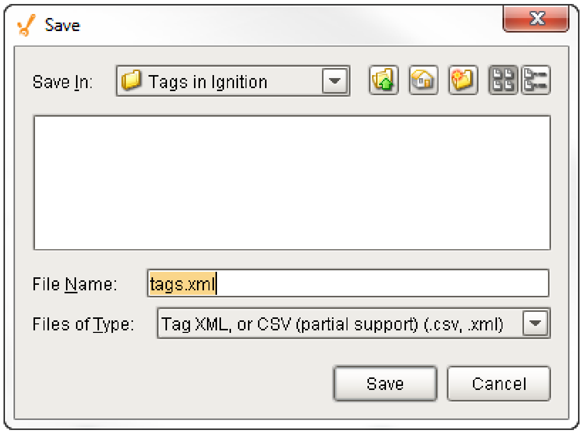
On this page ...

To Import and Export Tags

1. Go to the **Designer**, select the desired folder from the **Tag Browser** that contains the Tags that you want to export. If you want to select all your Tags, click on the Tags folder. If you want to click on one Tag, click on the individual Tag.
2. On the **Tag Browser** toolbar, click the **Export** icon to export, or **Import** icon to load a previously exported file.



3. Specify the folder you want to export/input your Tags either to or from. Then choose either the **.xml** or **.csv** file type to export or import your Tags. The **Save** (to export) or the **Open** (to import) window is displayed. **Note:**Importing Tags from XML will overwrite Tags if an *exact* match is found.



File Name: tags.xml
Files of Type: Tag XML, or CSV (partial support) (.csv, .xml)

INDUCTIVE UNIVERSITY

Importing and Exporting Tags

Watch the Video

This feature is new in Ignition version **7.9.13**
Click here to check out the other new features

Tag Export Localization

When exporting tags, we now include the designer's system locale and format numbers in the output according to that locale. When importing back in, the locale in the import file is honored and values interpreted accordingly. If a locale is not specified, the designer's system locale is used.

Dates are explicitly excluded from locale-based formatting.

XML Format

Extensible Markup Language (XML) is a markup language that defines a set of rules to encode documents. XML is easy for software to read and export, therefore, exporting Tags to an XML format results in simple and reliable transport of all Tag configuration settings.
The XML format is best for cases where you are interested in transferring Tag definitions and configurations from one Gateway to another. However, it is not optimized for directly editing its contents. Modern variations of Microsoft Excel can open and display the contents of an XML file. You can also open the contents of the XML file in Notepad or another text editor, edit the contents, and import the XML file back into Ignition.

CSV Format

The CSV format is often used to export Tags with the intent of making mass edits to the Tag definitions and configurations in a spreadsheet program such as Microsoft Excel. The Tags can be edited and imported back into the Gateway. Alarm properties are not included in CSV export format, but are included in XML export format.

> ⓘ There are many configuration settings for Tags than what is displayed in a CSV or XML export file. The Tag export feature only exports the configuration properties that have been edited in at least one of the Tags in the selected export folder. Therefore, to ensure the desired configuration setting is available in the export file, at least one Tag within the selected export folder must have that configuration property changed.

Property Values in XML or CSV Export Files

When viewing a CSV file which has been exported from Ignition, you will see many columns each showing a configuration property value. The property consists of its name and its value, in some cases there might be additional attributes available.

The following table shows the configuration property names and values you will see in an exported CSV file.

Tag Properties

Property Name	Type	Values (if applicable)	Description
Value			The value of the Tag, dependent on the data type.
Data Type	Int	0 - Int1 1 - Int2 2 - Int4 3 - Int8 4 - Float4 5 - Float8 6 - Boolean 7 - String 8 - DateTime 9 - DataSet	
Enabled	Boolean	true/false	
Tagtype	Int	0 - OPC Tag 1 - DB Tag (see ExpressionType) 2 - Client Tag 6 - Folder 13 - Derived Tag	Determines the type of the tag. A value of one is a "DB Tag", which is either a Memory Tag, Query Tag, or Expression Tag, depending on the value of the ExpressionType field.
ExpressionType	Int	0 - None 1 - Expression 2 - SQL Query	Used in conjunction when the TagType is set to 1 (DB Tag). Otherwise this field is ignored.
AccessRights	Int	0 - Read Only 1 - Read/Write 2 - Custom	If custom, will be defined by a **Permissions** Tag.
OPCServer	String		
OPCItemPath	String		

OPCWriteBackServer	String		Write back target for expression Tags.
OPCWriteBackItemPath	String		
ScaleMode	Int	0 - Off 1 - Linear 2 - Square Root 3 - Exponential Filter	
ScaleFactor	Float		For exponential filter
RawLow	Float		Defines scale range
RawHigh	Float		
ScaledLow	Float		
ScaledHigh	Float		
ClampMode	Int	0 - None 1 - Low 2 - High 3 - Both	
Deadband	Float		
DeadbandMode	Int	0 - Absolute 1 - Percentage	
FormatString	String		
EngUnit	String		
EngLow	Float		
EngHigh	Float		
EngLimitMode	Int	0 - None 1 - Low 2 - High 3 - Both	
Tooltip	String		
Documentation	String		
DriverName	String		Used for external Tags
ScanClass	String		The export will only include the name of the Scanclass, not the configuration of the Scanclass itself. A Scanclass with the same name needs to already exist on the Gateway that the Tags are being imported to, prior to importing them.
HistoryEnabled	Boolean	true/false	
PrimaryHistoryProvider	String		The history provider to use if storing history
HistoricalDeadband	Float		
HistoricalDeadbandMode	Int	0 - Absolute 1 - Percentage	
HistoricalScanclass	String		
InterpolationMode	Int	0 - Discrete 2 - Analog (deadband) 3 - Analog (compressed)	How values are interpolated. 2 exists for backwards compatibility (and is equivalent to 1), but only 0 or 3 should be used in the future.
HistoryTimestampSource	Int	0 - System 1 - Value	

HistoryMaxAge Mode	Int	0 - Unlimited 1 - Limited	
HistoryMaxAge	Int		Max cycles between storage.
UDTParentType	String		The path to the parent UDT type. Used by sub-types and instances.

Alarm Property Values in XML Export Files

When viewing an XML file which has been exported from Ignition, you will see the above fields as well as extra fields for any configured alarms.

The following table shows the alarm property names and possible values you will see in an exported XML file.

Alarm Properties

Property Name	Type	Values (if applicable)	Description
Enabled	Boolean	true/false	
Priority	String	Diagnostic, Low, Medium, High, Critical	May also be numeric, 0-4.
DisplayPath	String		
ActivePipeline	String		
ClearPipeline			
Deadband	Float		
DeadbandEvalMode	Integer	0 - Absolute 1 - Percentage	
TimeOnDelaySeconds	Float		
TimeOffDelaySeconds	Float		
TimestampSource	Int	0 - System 1 - Value	
AckMode	Int	0 - Unused 1 - Auto 2 - Manual	
Notes	String		
AckNotesReqd	Boolean	true/false	
ShelvingAllowed	Boolean	true/false	
Mode	String	Equality Inequality AboveValue BelowValue BetweenValues OutsideValues OutOfEngRange BadQuality AnyChange Bit OnCondition	
SetpointA			The setpoint, or the **low** setpoint for dual value modes.
SetpointB			The **high** setpoint for dual setpoint modes.
InclusiveA	Boolean		
InclusiveB	Boolean		

BitOnZero	Boolean		Used by the **Bit** condition, to indicate that 0 is **active**.
BitPosition			Bit to use in **Bit** condition.
ActiveCondition	Boolean		The property that drives the **OnCondition** mode.
AnyChange	Boolean		Whether or not to apply the **any change** behavior in applicable modes.

Related Topics ...

- Array and Dataset Tags