OHDOT_DU_ItemTypeSync

Description

This app is a resizable C# Windows Presentaiton Form OpenRoads Designer Addin application.

It is used to help populate data used for drainage and utility annotation.

Software Requirements

This application was developed using .NET Framework 4.7.2 in 64 bit Windows 10.

This application requires OpenRoads Designer CONNECT Edition to run (MicroStation CONNECT edition cannot be used with this addin).

Loading the application

The application is loaded via the OHDOT DU Item Type Sync button from the Ohio DOT Workflow within the Plan Preparation Group.

It can also be launched with the following Key-in commands

```
mdl load OHDOTItemTypeSync; OHDOTItemTypeSync OPEN
```

Note that the OHDOT workspace must be loaded.



Application Overview

This app should be run within the Drainage or Utility basemaps. Depending on the version of ORD the application is load differently.

Running ORD 2023 or newer:

Once the application is launched it will look as imaged below if the version of ORD is 2023 or newer.

OHDOT_DU_ItemTypeSync					– 🗆 X
Help Table View Hierarchical View Conduit	Refresh All Update All Attach All Missi	g SSOT DU IT Detach All View	DU IT Instances	View Item Type Lib Info	Output found alignment names to Log
Actions Type Link Type Name	FD Name	DU IT Name Has	s U	Property \	/alue Overwrite Value Type Ca
Update 🗐 🐞 Link Conduit D-1	Type A-Elliptical Concrete CL-P	OHDOT_StormSanConduit True	2		
Ů Update 및 ¥ Link Conduit D-2	Type B-Round Concrete_Clay_Plastic-P	OHDOT_StormSanConduit True	2		
Update 🗐 💃 Link Conduit utxCulv	Box Culvert-X	OHDOT_StormSanConduit True	2		
Update 🗐 🏌 Link Conduit D-3	Type A-PipeArch Corr. 1"-P	OHDOT_StormSanConduit True	2		
Update 🗉 🐐 Link Conduit D-4	Type B-Round Concrete_Clay_Plastic CL-F	OHDOT_StormSanConduit True	2		
Update 🗐 💃 Link Conduit D-5	Type B-Round Concrete_Clay_Plastic CL-F	OHDOT_StormSanConduit True	2		
Update 🗐 🎇 Link Gutter Gutter	Type1,2A,2B,6_Curb	False	e		
Ů Update 및 ¥ Node D-1	CB-3 Curb-P	OHDOT_StormSanNode True	2		
Update 🗉 🐐 Node D-2	MH-3-048" Base-P	OHDOT_StormSanNode True	2		
Update 🗐 🦹 Node D-3	CB-5A-42"-P	OHDOT_StormSanNode True	2		
Update 🗐 🐮 Node D-4	DummyNode_0Capture	OHDOT_StormSanNode True	•		Apply
			r		

Running Prior to ORD 2023:

Once the application is launched it will look as imaged below if the version of ORD is prior to 2023.

OHDOT_DU_ItemTypeSync Help	-		×					
Attach Missing Item Types to all DU items in all vi	ews							
Get Default DU Item Type Values from DGNLIB and set on Desig	n mode	el DU I	tems					
Update Item Type Property Values from DU Drainage Properties								
Sync DU Item Type properties from Design model to items in all other views								
Update UDX Properties from Item Type Properti	es							

Help Menu:

The Help Menu consists of two items Wiki-Help Doc/Video and About.

Wiki-Help Doc/Video

Clicking this item will launch <u>this</u> wiki page. The wiki page contains a link to this document as well as a YouTube video explaining how to use this application.





About

Clicking this item will open another dialog, pictured to the right. This dialog displays the following information.

- Application Name
- Application version
- Copyright
- o Company
- Description



Application Operation ORD 2023 or newer

This dialog has 5 main areas, View Display, Bulk actions buttons, DU Items View, Properties List View, and the Results section.

View Display

Table View	Hierarchical View
All ~	Gutter V

There are two main display modes for this application, Table View and Hierarchical View.

Table View



Table view, when selected will display all the found DU features an a list format in the DU Items View area. There are there filter options for the view; All, Conduit, Gutter.

This will display all the found DU features in the <u>DU Items View</u> area.

Conduit:

All:

This, when selected will display only DU conduit features and the nodes that are connected to them. (does not show gutter features and nodes only connected to gutters)

Gutter:

This, when selected will display only Drainage gutter features and the nodes that are connected to them. (Does not show conduit features and nodes only connected to conduit)

Hierarchical View



When this is selected the DU Items View will change to show a graphical based hierarchical view of the DU features. The display can be filtered to either Conduit or Gutter. Which are described in the previous section. An example of a Hierarchical view is shown below.



Bulk Action Buttons

Re	fresh All	Update All	Attach All Missing SSOT DU IT	Detach All View DU IT Instances	View Item Type Lib Info	Output found alignment names to Log

Refresh All

This button will reload all the date for all found DU Features.

Update All

This button will apply all item type property value updates for all found DU Features.

Attach All Missing SSOT DU IT

This button will attach the DU Item type to any features that are missing a SSOT DU IT. Note that this can be done on individual DU features from the <u>DU Items View</u> area, specially with in the <u>Has</u> <u>SSOT DU IT</u> column.

Detach All View DU IT Instances

This button will detach any view specific DU Item type found for any DU Feature. Note that this can be done on individual DU features from the <u>DU Items View</u> area, specially with in the <u># View DU IT</u> <u>Instances</u> column.

View Item Type Lib Info

This button will open the Item Type Lib Info View Dialog.

Output found alignment names to Log

This button will output the names of all found alignments with the active file or its references.

DU Items View

This view is dependent on if <u>Table View</u> or <u>Hierarchical View</u> is selected. The Properties List View will be populated with the current selection.



Table ListView Columns:

Actions	Туре	Link Type	Name	FD Name	DU IT Name	Has Updates	Has UDX Updates	Has SSOT DU IT	Has FD Defaults	# View DU IT Instances	FD Issue	SSOT IT Issue
🖸 Update 🗉 🐐	Link	Conduit D-1 Type A-Elliptical Concrete CL-P OF		OHDOT_StormSanConduit	True	True	True	True	0			
ଧ Update 🗉 🐐	Link	Conduit	D-2	Type B-Round Concrete_Clay_Plastic-P	OHDOT_StormSanConduit	True	True	True	True	0		
🖸 Update 🗉 🐐	Link	Conduit	utxCulv	Box Culvert-X	OHDOT_StormSanConduit	True	True	True	True	0		
ି Update 🗐 🐐	Link	Conduit	D-3	Type A-PipeArch Corr. 1"-P	OHDOT_StormSanConduit	True	True	True	True	0		
한 Update 🗉 🐐	Link	Conduit	D-4	Type B-Round Concrete_Clay_Plastic CL-P	OHDOT_StormSanConduit	True	True	True	True	0		
Update 🗉 🐐	Link	Conduit	D-5	Type B-Round Concrete_Clay_Plastic CL-P	OHDOT_StormSanConduit	True	True	True	True	0		
ଧ Update 🗉 🐐	Link	Gutter	Gutter	Type1,2A,2B,6_Curb		False	False	False	False	0	Conduit\Sto	No SSOT DU Item Types
🖸 Update 🗉 🐐	Node		D-1	CB-3 Curb-P	OHDOT_StormSanNode	True	False	True	True	0		
ଧ Update 🗉 🐐	Node		D-2	MH-3-048" Base-P	OHDOT_StormSanNode	True	True	True	True	0		
🖸 Update 🗉 🐐	Node		D-3	CB-5A-42"-P	OHDOT_StormSanNode	True	True	True	True	0		
신 Update 🗉 🐐	Node		D-4	DummyNode_0Capture	OHDOT_StormSanNode	True	True	True	False	0	Node\Storm	Cant get default values!
🖸 Update 🗉 🐐	Node		Unnamed	CB-5A-42"-P	OHDOT_StormSanNode	True	True	True	True	0		
인 Update 🗉 🐐	Node		D-5	CB-5A-66"-P	OHDOT_StormSanNode	True	True	True	True	0		
진 Update 🗉 🐐	Node		D-6	CB-5A-66"-P	OHDOT_StormSanNode	True	True	True	True	0		
じ Update 🗐 🐐	Node		D-7	CB-5A-66"-P	OHDOT_StormSanNode	True	True	True	True	0		

Actions

Refresh

This button will reload all the date for this specific feature.

Update

This button will apply all item type property value updates for this specific feature.

Fit View

This button will fix the active view window to this feature (currently only works in design or Design-3D models, not profile/XS views).

Add to Selection

This button will add this feature to the current selection set.

Туре

Lists the DU Feature type (Link or Node).

Link Type

Lists the Link feature sub type (Conduit or Gutter).

Name

Lists the Feature Name.

FD Name

Lists the Feature Definition Name.

DU IT Name

Lists the Drainage & Utilities Item Type Name.

Has Updates

(True/False), indicates is that DU feature's Item Type property values are out of date and need updated.

Has UDX Updates

(True/False), indicates is the DU Feature's UDX property values are out of date and need updated.

Has SSOT DU IT

(True/False), indicates if the DU Feature has a Single Source of Truth (SSOT) Item Type attached to it.



Has SSOT DU IT False Attach If the SSOT DU IT is missing, an Attach button will be shown. If selected it will attach the DU IT to the feature.

Has FD Defaults

(True/False), indicates if the Feature Definition defined on the Feature contains defaults values for the DU IT.

View Du IT Instances

Lists the number of found view specific (not SSOT) DU Item Type instances

# View DU IT Instanc	es
⁵ Detach	

If there are view specific DU Item types instances a Detach button will be shown. If selected it will detach all the view specific DU item type instances from that specific DU Feature.

FD Issue

Lists any issues with defined Feature Definition like if no default values could be found.

SSOT IT Issue

Lists any issues with features Item Type, like if the item type is not attached.

Hierarchical View:



Right Click Menu:

2	Update 🗐 🐐
	Name: D-3
	Type: Link
	Link Type: Conduit
	FD Name: Type A-PipeArch Corr. 1"-P
	DU IT Name: OHDOTStormSanConduit
	Has Updates: True
	Has UDX Name: True
	Has SSOT DU IT: True
	Has FD Defaults: True
	# View DU IT Instances: 0
	FD Issue:
	SSOT IT Issue:

This lists the same info as the <u>Table ListView</u> <u>Columns</u>.

Note that just as in table view, if the SSOT DU IT is missing the attach button will display as well if there are found View specific instances of DU IT the Detach button will display.

Has SSOT DU IT: False Attach

View DU IT Instances: 5 Detach

Properties List View

	Property	Value	Overwrite	Value Type	Calc Type	Calc/Default Value	UDX Value	Has Update	Has UDX Update	*
U	Offset			number	Calc	52.393		True	True	
U	AbsOffset			number	Calc	52.393		True	True	
Q	Grate_Rim_Elev			number	Calc	990		True	False	
U	Conduit1Invert			number	Calc	980		True	False	
Q	Conduit1Angle			number	Calc	178.897312740218		True	False	
Q	Conduit2Angle			number	Calc	0		False	False	
Q	Conduit3Angle			number	Calc	0		False	False	
Q	Conduit4Angle			number	Calc	0		False	False	
U	Conduit5Angle			number	Calc	0		False	False	
U	OutgoingCond			number	Calc	341.599399355824		True	False	
Q	Conduit2Invert			number	Calc	Failed		False	False	
U	Conduit3Invert			number	Calc	Failed		False	False	
Q	Conduit4Invert	0		number	Calc	Failed		False	False	•
					Apply					

When a feature is selected the DU Item Type properties are listed in this view.

Property List Columns

Refresh

Button to refresh this specific property.

Property The name of the property

Value

The Value of the property. If Value Type is Default or Calc then this will not be enabled for editing unless Overwrite is checked up (only available if property is set up to allow overwriting the

default/calc value). If the Value Type is readonly then this will not be enabled for editing. If the value type is none then this will be enabled.

The entered value must be in the format accepted by the value type. Meaning if the value type is a number and the entered value contains text, that is not allowed, and will be indicated in the dialog as shown below.



Overwrite

 \geq

This column contains a checkbox. If check then that is saying the properties value is overwritten and a manually entered value is being used instead of the default of calc value. Overwrite checkbox will only be visible for properties with a calc type of default or calc as well as the property has been set up to allow overwriting (not all properties are allowed to be overwritten, for example a node name property wouldn't make to allow overwriting).

Value Type

This indicated the type of value the Item Type property is set to. Currently only the following types are accepted by this app.

- Text
- Number
- Integer
- True/False
- Unsupported (all other types will be indicated as with, and will not work with this app)

Calc Type

Below lists the different calculation types a property can have.

- Default, Property has a default value specified on its feature definition
- Calc, Property has defined calculation(s) for how to populate its value
- Readonly, the Item Type property is defined as readonly, IT property could be an expression or simply is set to readonly.
- None, property is none of the above. Must be filled out manually.

Calc/Default Value

This column will show what the calculated or default value is for that property for the selected DU Feature. If this value is different the whats listed in the Value column (and overwrite is not checked on) then when the apply button is selected or the update button, the property value will be updated to match this calc/default value. this is how the Has Updates True/False field is populated.

Note that if a calculation fails the value will be listed as Failed.

UDX Value

If the property has a UDX value defined than text box will be visible showing the currently value of that UDX property. If the UDX property value does not match the Calc/Default value (or if overwritten then the Value column is used) then when Apply button or Update button is selected the UDX value is updated to match the I.T. value. this is how Has UDX updates True/False field is populated.

Has Update

This True/False field indicates if the current value of the property is different from the calc/default value. if it is different then has updates will be true. If any of the properties have updates then the <u>Has Updates</u> column in the <u>DU Item View</u> area will be set to true.

Has UDX Update

This True/False field indicates if the current value of the UDX property is different from the I.T. property value. The i.t. property value used is the calc/default value if there is one or if overwritten then the overwritten value.

Results Section

Copy Results to Clipboard Clear Log Refreshed data successfully, see log for details Found Geometry Model in active File	
Found 2 alignments in active file!	
Found a total of 2 alignments (2 in file, 0 attached)	
Found the following DU Item Types within the OHDOTDU Item Type Library OHDOT_StormSanConduit, with calculated properties data, with default properties data, with overwrite data, with UDX properties data OHDOT_StormSanNode, with calculated properties data, with default properties data, with overwrite data, with UDX properties data OHDOT_UtilityConduit, with calculated properties data, with default properties data, with overwrite data, NO UDX properties data	
Found and populated 7 Link Features in active file	
Found and populated 8 Node Features in active file	
All DU Item Type instances only exist on DU Features (this is good)	-

This section is update through the use of the app. It will provide the result of any action taken. If something failed then it will be highlighted in red.

This section can be expanded to view more detailed. But an overall summary result of the last action will be displayed in the top section, so that it is visible even if the section is collapsed.

Copy results to Clipboard

This button is copy all log entries to the clipboard.

Clear Log

This button will clear out all log entries.

Item Type Lib Info View Dialog

引 ItemTypeLibInfoView								- 🗆 X
Name	Has Defaults Setup	Has Calcs Setup	Has OverWrites	Has UDX Setup	Property	Calc Type	Overwritab	Calc Property Info/ FD Default Values
OHDOT_StormSanConduit	True	True	True	True				Valid:True Property Pull ConduitElementManager_Presentation.DerivedLinkResults_UnifiedLength Unit
OHDOT_StormSanNode	True	True	True	True	NodeToNodel	Calc	True	
OHDOT_UtilityConduit	True	True	True	False				
					Size	Calc	True	Validi.True Conduit Size Lookup Prop: ConduitElementManager_Presentation.ConduitShape = 2 Forma Validi.True Conduit Size Lookup Prop: ConduitElementManager_Presentation.ConduitShape = 3 Forma
		ß			Туре	Default	True	[Type A-Elliptical Concrete CL-P, TYPE A] [Type B-Round Concrete_Clay_Plastic-P, TYPE B] [Box Culvert-X,]
					NodeToNodeS	Calc	True	Valid:True Property Pull ConduitElementManager_Presentation.DerivedLinkResults_Slope Unit: percent
					StartElevation	Calc	True	Valid:True Property Pull StormWaterSegment_Presentation.Physical_UpstreamInvert Unit: feet Roundin Valid:True Property Pull WasteWaterSegment_Presentation.Physical_UpstreamInvert Unit: feet Roundin

This dialog has two main area, DU Item Types List View, and Du Item Type Property info List View.



DU Item Types List View

This section will list the found Item Types that are for Drainage and Utilities with the following columns. The DU Item Types Property Info List View is populated based on the selected item type from this view.

Name

The name of the Item Type

Has Defaults Setup

(True/False), indicates if FD defaults are set up for this Item Type

Has Calc Setup

(True/False), indicates if custom calculations are set up for properties of this item type

Has OverWrites

(True/False), indicates if default/calc property values can be manually overwritten.

Has UDX Setup

(True/False), indicates if Item Type properties are set up to sync to UDX properties.

DU Item Types Property Info List View;

Property	Calc Type	Overwritab	Calc Property Info/ FD Default Values	UDX Property Info	-
NodeToNodeL	Calc	True	Valid:True Property Pull ConduitElementManager_Presentation.DerivedLinkResults_UnifiedI		
Size	Calc	True	Valid:True Conduit Size Lookup Prop: ConduitElementManager_Presentation.ConduitSh Valid:True Conduit Size Lookup Prop: ConduitElementManager_Presentation.ConduitSh		
Туре	Default	True	[Type A-Elliptical Concrete CL-P, TYPE A] [Type B-Round Concrete_Clay_Plastic-P, TYPE B] [Box Culvert-X,]	Valid:True ConduitElementManager_Presentation.Type_Ol	
NodeToNodeS	Calc	True	Valid:True Property Pull ConduitElementManager_Presentation.DerivedLinkResults_Slope U		

This section will list all the defined properties for the selected item type.

Item Type Property Columns

Property

Name of Property

Calc Type

Below lists the different calculation types a property can have.

- Default, Property should have a default value specified on feature definitions.
- Calc, Property has defined calculation(s) for how to populate its value.
- Readonly, the Item Type property is defined as readonly, I.T. property could be an expression or simply is set to readonly.
- None, property is none of the above. Must be filled out manually.

Overwritable

True/False, indicates if this property is set up to allow overwriting of default/calc value. Only properties that have calc type of Calc or Default can be overwritten.

Calc Property Info/FD Default Values

This section will list all the information pertaining to either calculation information of default values.

When the property is of Default Calc Type this section will list all the FD currently set on placed DU features and their default value defined for that FD. For example, the below screenshot shows the Type property for the feature definition named Type A-Elliptical Concrete CL-P it has a default value of TYPE A defined for this Type Property. Further you can see the Feature Definition named Type B-Round Concrete_Clay_Plastic-P has a default value of TYPE B defined.

			[Type A-Elliptical Concrete CL-P, TYPE A]	•
Туре	Default	True	[Type B-Round Concrete_Clay_Plastic-P, TYPE B]	
			[Box Culvert-X,]	-

When the property is of Calc Calc type this section will list all the information on what calculations are defined for this property.

A property can have multiple calculations defined (each line is a new calculation). The calculations are processed in the order they are shown. If a calculation fails then it will move to the next (if defined) calculation.

Each calculations info is displayed in the following format

Valid:True/False | Custom Calc Type | defining parameters

See custom Calculations section for more information on the different calculations and there parameters.

If a calculation is not valid then there is a problem with how that calculation was defined.

UDX Property Info

If a property has an equivalent UDX property defined. Info on that UDX property will be shown here. It will have a Valid:True/False to indicate if the information defining the UDX property is correct or not. And then will list the Property Class and UDX property name. this is how the application will find this UDX property that goes with this Item Type Property.

UDX Property Info
Valid:True ConduitElementManager_Presentation.Existing_Action_OHDOT

Custom Calculations:

Below lists the available calculation options and there parameters.

PropertyPull

Desc: defines what property to pull its value and set to this Item Type Property value.

Parameters:

• Property Lookup: defines the property on the feature to get its value for. Must include class definition. Also can specific what unit and rounding the property value should have as well as a suffix value like " for inches or ' for feet. See <u>units</u> for available units this app can return values in. Unit, rounding, and unit suffix are optional and don't have to be set. Example to get a conduit length in feet rounded to 2 decimals with no unit suffix would be

 $Conduit Element Manager_Presentation. Derived Link Results_Unified Length. feet. 2$

Syntax: PropertyPull~ClassDef.PropertyName.Unit.rounding#decimals.unitSuffix

Notes:

FromLinkIT

Desc: can only be defined on Node Features, it will pull a property from a connected link feature.

Parameters:

- Node Property containing Link Label: this is the property on the node that provides the name to the link feature you want to pull a property from. Must list the Class Definition and the property name. (example CatchBasinElementManager Presentation.Conduit1Label)
- Link property: the I.T. property on the link feature you want to pull the property value from. Must list the Item Type name and the property name (example. OHDOT_StormSanConduit.StopElevation)

Syntax:

 $From Link IT \- Class Def. Property Name On Node Containing Link Label \- Node IT Name. IT Property On Link \\- Node IT N$

Notes: No Unit Conversions.

FromNodelT

Desc: can only be defined on Link Features, it will pull a property from a connected Node feature.

Parameters:

- Link Property containing Node Label: this is the property on the Link that provides the name to the Node feature you want to pull a property from. Must list the Class Definition and the property name. (example
 - CLASS_SU_Link_Presentation.PROP_SU_StartNode)
- Node property: the I.T. property on the Node feature you want to pull the property value from. Must list the Item Type name and the property name (example. OHDOT_StormSanNode.BaselineName)

Syntax:

 $From NodeIT \sim Class Def. Property NameOnLink Containing NodeLabel \sim Link IT Name. IT Property On NodeLabel \sim Link IT$

Notes: No Unit Conversions.

NodeStation

Desc: Calculates the station of a node feature. Can only be used on node features. The Baseline Reference property must be defined on the node feature for this to work as it uses that baseline reference to calculate the station.

Parameters: No parameters.

Syntax: NodeStation

Notes: Using the Node_Presenation.ActiveStationOffsetReference property (Baseline Reference) to calculation station. Station format uses active settings.

NodeStationOffset

Desc: Calculates the offset of a node feature from an alignment. Can only be used on node features. The Baseline Reference property must be defined on the node feature for this to work as it uses that baseline reference to calculate the offset.

Parameters: No parameters.

Syntax: NodeStationOffset

Notes: Using the Node_Presenation.ActiveStationOffsetReference property (Baseline Reference) to calculation station. Offset format uses active settings.

NodeStationAbsOffset

Desc: Calculates the absolute offset of a node feature from an alignment. Can only be used on node features. The Baseline Reference property must be defined on the node feature for this to work as it uses that baseline reference to calculate the offset.

Parameters: No parameters.

Syntax: NodeStationAbsOffset

Notes: Using the Node_Presenation.ActiveStationOffsetReference property (Baseline Reference) to calculation station. Offset format uses active settings.

NodeStationOffsetSide

Desc: Calculates the offset side (Left/Right) of a node feature from an alignment. Can only be used on node features. The Baseline Reference property must be defined on the node feature for this to work as it uses that baseline reference to calculate the offset.

Parameters:

- Left Format: Display value for if the offset is on the left side of the alignment (Example LT).
- Right Format: Display value for if the offset is on the right side of the alignment (Example RT).

Syntax: NodeStationOffsetSide~LeftFormat~RightFormat

Notes: Using the Node_Presenation.ActiveStationOffsetReference property (Baseline Reference) to calculation station.

AngleToCompassDirection

Desc: Calculates the Compass Direction (N, NW, W, SW, S, SE, E, NE) a link is connected to a node (from node out perspective).

Parameters:

• Angle Property: property on Node feature containing the angle for the specific link connected to the node feature. Must include Class definition with property name. (example CatchBasinElementManager_Presentation.Conduit1Angle)

Syntax: AngleToCompassDirection~ClassDef.PropertyNameForLinkAngle.

Notes:

ConnectedPipeITCombine

Desc: Can only be used on node features. Combines item type properties into one property value.

Parameters:

- Conduit Look up names: conduit names list. This lists the conduit name prefix or suffix for all the possible connected conduits. This will be used with the next two parameters as replacement text to find Item Type properties specific to that conduit. Example format is (Conduit1^Conduit2^Conduit3^Conduit4^Conduit5^OutgoingConduit). Each conduit lookup name is separated by a ^.
- Conduit Check Name: this parameter defines the Item Type property to check to see if the specific conduit is in use on the node. Example, if a node only has 3 connected pipes then the properties relevant to conduit4 would not be in use and should be included. This property is typically the conduit label property. An example format is [x]Label. The [x] is replaced with each of defined conduit look up names. This will be an Item Type Property. If this check property has a value of NA, (N/A),o, or nothing then that conduit is not included.
- Combining Format: this defines the format for a connected pipe that passes the Conduit Check Name parameter described above. The format is a combination of three different things
 - 1. (IN/OUT): if the pipe flowing into the node then this is evaluated to IN, if the pipe is flowing out of the node then this is evaluated to OUT.
 - 2. Text: defines text in the format like spaces or dashes between other pieces of the format. Must be encompassed with double quotes, for example to add a space between format pieces it would be this " ".
 - 3. IT property lookup: The name of the item type property to get a value from this can make use of the conduit Lookup names to find properties on this nodes item type. For example [x]Invert would be evaluated for each lookup name like Conduit1Invert, Conduit2Invert. These are item type properties on the node.

Each piece of the format is separated by a ^ and is then evaluated for each conduit lookup name to generate a text concatenation.

Example Syntax:

ConnectedPipeITCombine~Conduit1^Conduit2^Conduit3^Conduit4^Conduit5^OutgoingConduit~[X]Label~(IN/O UT)^" "^[X]Label

Notes: Link Item Type Property names can't have spaces!

ConduitSize

Desc: defines how to display a conduit size based on its shape.

Parameters:

- Shape Check Property: this is the property name use to check the shape of the conduit.
- Shape Check Value: this is the value the shape check property should match in order to complete the calculation.
- Size Display format: defines the format to display. Format pieces can be either plain text or a property lookup.
 - 1. Plain text: defines plain text with in the format. Typically used to add spaces or dashes between other pieces of the format. Piece must be encompassed with double quotes. Example to add a dash to the format it would be "-".
 - 2. Property Lookup: defines the property on the feature to get its value for. Must include class definition. Also can specific what unit and rounding the property value should have as well as a suffix value like " for inches or ' for feet. See <u>units</u> for available units this app can return values in. Example to get a conduit rise in inches rounded to a whole number with " suffix it would be ConduitElementManager_ Presentation.ConduitRise.inches.o."

Each piece of the format is separated by a ^.

Example Syntax: for Box shape conduit (ConduitShape =3) display the rise and span in inches rounded to a whole number. Display if rise was 24" and run was 36" would be 24" x 36" ConduitSize~ConduitElementManager_Presentation.ConduitShape~3~ConduitElementManager_Presentation.Con duitRise.inches.0."^" x "^ConduitElementManager_Presentation.ConduitSpan.inches.0."

Syntax:

 $Conduit Size \sim Class Def. Property Name For Shape Check \sim Shape Check Value \sim Size Display Format.$

Notes:

Unknown

Means the specified calculation cannot be evaluated by this app.

Units:

Available length units

- centimeters
- decimeters
- feet
- inches
- kilometers
- meters
- mfeet
- miles
- millifeet

- millimeters
- usSurveyFoot
- yards

Available slope units



- percentSlope
- oneOverSlope
- centimeterPerMeter
- footPer1000Feet
- footPerFoot
- footHorizontalPerFootVertica
 l
- footVerticalPerFootHorizonta
- footPerMile

- horizontalPerVertical
- inchPerFoot
- meterPerKilometer
- meterPerMeter
- meterHorizontalPerMeterVertical
- meterVerticalPerMeterHorizontal
- millimeterPerMeter
- millimeterHorizontalPerMeterVertical
- millimeterVerticalPerMeterHorizontal
- verticalPerHorizontal

Configuration Variables:

OHDOT_DU_ItemTypeLibName: Used the configuration variable to define the name of the Item Type Library that contains the drainage and utility item types.

Item Type Library Setup:

In order for this application to work all the item types this app will update need to be within the same item type library. This library needs to be defined via configuration variable as described in the <u>Configuration Variables</u> section.

Each property within an DU Item Type can be set up to handle the following 4 abilities:

1. be able to pull default values from feature definitions

This is set up by creating an additional item type with the same name as your DU Item Type but appended with |DefaultPropsSetup. For example if you have an Item Type named OHDOT_StormSanConduit then to set up which properties within this item type should pull Default values from feature definitions you would create an item type named OHDOT_StormSanConduit|DefaultPropsSetup. With in the DefaultPropsSetup item type, each property you want to pull defaults for needs to be a True/False property in this setup item type. The property name should be the same but appended with |DefaultSetup. Set the value to true. This is how you define a DU Item Type property to pull a default value from a feature definition. Make sure to attach the DU Item type to the desired Feature definitions and then set the default value for each of the properties specific to that feature definition.

2. set calculation(s) for determining a value

This is set up by creating an additional item type with the same name as your DU Item Type but appended with |CalcPropsSetup. For example if you have an Item Type named OHDOT_StormSanConduit then to set up specific calculations for the desired properties within this item type you would create an item type named OHDOT_StormSanConduit|CalcPropsSetup. With in the CalcPropsSetup item type, each property you want to specify a calculation for needs to be a text property in this setup item type. The property name should be the same but appended with |CalcSetup. Set the value to the syntax for the desired calculation as described in <u>Custom</u> <u>Calculations</u> Section. To define multiple calculations just add a | between calculations syntaxes. This is how you define a DU Item Type property to use a calculation to get its value.

3. allow default/calc values to be manually overwritten

This is set up by creating an additional item type with the same name as your DU Item Type but appended with |Overwrite. For example if you have an Item Type named OHDOT_StormSanConduit then to setup which properties can be overwritten (only properties that have defaults or calculations defined can be overwritten) you would create an item type named OHDOT_StormSanConduit|Overwrite. Within the overwrite item type, each property you want to allow to be overwritten needs to be a True/False property in this overwrite item type. The Property name should be the same but appended with |Overwrite. Set the value to false as a default. Make sure to attach this Overwrite to the desired Feature definitions. Then as features are placed this overwrite Item type will automatically be there. When a user wants to overwrite a property value for a specific feature this overwrite item type instance on that feature will get updated to set the given property to true to indicate that the property is being over written. This is how you define a DU Item Type property to be able to be overwritten. If you have a property you don't want to allow to be overwritten just don't add it to the Overwrite item type.

This is set up by creating an additional item type with the same name as your DU Item Type but appended with |UDXSetup. For example if you have an Item Type named OHDOT_StormSanConduit then to set up which properties have equivalent UDX properties within this item type you would create an item type named OUDOT_StormSanConduit|UDXSetup_With in the UDXSetup item type coch property you

OHDOT_StormSanConduit|UDXSetup. With in the UDXSetup item type, each property you want to specify an equivalent UDX property for needs to be a text property in this setup item type. The property name should be the same but appended with |UDXSetup. Set the value to the name of the UDX property with the class definition the UDX property is contained in (ClassDef.PropName). This is how you define a DU Item Type property to sync its property to a udx property.

Application Operation Prior to ORD 2023

Attach Missing Item Types to all DU items in all views Button:

When initially placing Nodes and conduits a item type is attached to each node/conduit during placement and default values are also set during placement.

*Note ORD 10.10 versions there is a bug where when conduits are placed the item type is not attached during placement.

The problem currently in ORD 10.10 is that these item types are only seen from the design model. this means if you open a profile view or 3d view (or profile run, or profile run projection) you not only won't be able to see these item type properties but you also won't be able to annotate with these item type properties. Note in ORD 10.11 you can now see item type properties across all views but are still unable to annotate with these item type properties.

So for the reasons listed above we need each view to have an additional instance of the item types (so that we can annotate from the item type properties).

This button will go through and attach missing item types to nodes and conduits in every view in the active file. not that this simply attaches the Item type instance and does not set any default values like when the nodes/conduits were initially placed.

It is important to note that in each view the item type is a separate instance so values entered in one view do not automatically get applied to the other views.

Get Default DU Item Type Values from DGNLIB and set on Design model DU Items Button:

After item type instances are attached to nodes and conduits in the ONLY the default 2d design view this button will set the default values for all item types within in all views.

The default values are set on each feature definition. When a node/conduit is initially placed the feature definition is copied into the file. this local copy of the feature definition (within this file) contains the default values specific to that feature definition. This local copy of the feature definition is what the app is using to get the default values which are then set on all the individual nodes/conduits within the default 2d design view. Note that these local feature definition copies can be used be the user to override the default values, the user would just need to change the default values on the local feature definition copies before clicking this button.

Note that if default values are not set after hitting this button, then the dgn file could be older and the local copies of the feature definitions were from before default values were set. Use the Sync file with DGNLIB checkbox (described below).

Sync file with DGNLIB checkbox

This checkbox is used to synchronize the local copy of the feature definitions with the source feature definitions defined within the dgnlib. This allows any changes made to the dgnlib to be brought into the active file. For example if a default item type property value was changed in the dgnlib then when this button is selected with this checkbox checked on the application will first update the local copies of the feature definitions to match whats in the dgnlib and then perform the actions described in the paragraph above.

Note that any local overrides to the default values in the local copied of the feature definitions will be reverted to the original values defined in the source dgnlib if this button is selected with this checkbox checked.

Update Item Type Property Values from DU Drainage Properties Button:

This button will grab all the calculated properties directly from the features drainage properties and sets the corresponding item type property value. This button will need to be ran after any changes are made to the drainage/utility network since the calculated values are static and do not automatically update. This is only done on the default 2d design view. Examples of calculated properties would be the length, slope, station, offset, connected node data, size.

Station Offset Values

In order for the application to populate the station offset values the Node MUST have the baseline reference set in the element properties.

Drainage and Utilities	*
Vertical Offset	0.000'
Ground Elevation	995.098'
Invert Elevation	991.784'
Elevation Reference	L-ACSC_T_OS : ProfileByTempla
Use Slope of Surface	True
Use Road Cross Slope	False
Road Cross Slope Offset	3.281'
Baseline Reference	CLP_S185
Baseline Station	401+20.00

Sync DU Item Type properties from Design model to items in all other views Button:

The previous two buttons set the default values and calculate values for all nodes and conduits but only in the default 2d design view. If further changes are needed to be made to individual nodes/conduit item type properties, then that should be done in the default 2d design view before running this button. This button will take all the item type property values from the default 2d design view and sets those values in all other views.

Update UDX Properties from Item Type Properties Button:

This button will update the following UDX properties from the Nodes/Conduit item type properties. Note that if UDX are missing then the file needs synced with DGNLIB in order to bring in the UDX properties.

- Type (Nodes and Conduit)
- **Start Station** (Conduit)
- Start Offset (Conduit)
- Start BL Name (Conduit)
- Stop Station (Conduit)
- Stop Offset (Conduit)
- Stop BL Name (Conduit)

Item Types

 \ge

Item types specific to running versions prior to ORD 2023. (for ORD 2023 and newer item types are not hard coded see Item Type Library Setup)

OHDOT_StormSanConduit

- **FeatureName**: read only property pulling the Feature Name.
- **NodeToNodeLength**: populated via app from <u>3rd button</u> from drainage properties
- **isExisting**: defaulted to true for existing feature Definitions, populated from <u>2nd button</u>
- **Prefix**: defaulted to EX. for existing feature definitions, populated from <u>2nd button</u>
- **Size**: populated via app from <u>3rd button</u> from drainage properties
- **Prefix_Size**: read only property combines Prefix and Size Properties
- **Type**: Defaulted on Feature Definitions (i.e., Type B, CONC, PVC...), populated from <u>2nd</u> <u>button</u>
- **Category**: Defaulted on Feature Definiens (i.e. STORM) used in XS annotation, populated from <u>2nd button</u>
- **NodeToNodeSlope**: populated via app from <u>3rd button</u> from drainage properties
- **StartNodeID**: populated via app from <u>3rd button</u> from drainage properties
- StartNodeName: populated via app from <u>3rd button</u> from drainage properties
- StartElevation: populated via app from <u>3rd button</u> from drainage properties
- **StopNodeID**: populated via app from <u>3rd button</u> from drainage properties
- **StopNodeName**: populated via app from <u>3rd button</u> from drainage properties
- **StopElevation**: populated via app from <u>3rd button</u> from drainage properties

OHDOT_StormSanNode

- FeatureName: read only property pulling the Feature Name.
- **BaselineName**: populated via app from <u>3rd button</u> from drainage properties, must have Baseline Reference property set.

 \ge

- Station: populated via app from <u>3rd button</u> from drainage properties, must have Baseline Reference property set.
- **Offset**: populated via app from <u>3rd button</u> from drainage properties, must have Baseline Reference property set.
- **AbsOffset**: populated via app from <u>3rd button</u> from drainage properties, must have Baseline Reference property set.
- **OffsetSide**: populated via app from <u>3rd button</u> from drainage properties, must have Baseline Reference property set.
- **isExisting**: defaulted to true for existing feature Definitions, populated from <u>2nd button</u>
- **Prefix**: defaulted to EX. for existing feature definitions, populated from <u>2nd button</u>
- Type: Defaulted on Feature Definitions (i.e., CB2-2A), populated from <u>2nd button</u>
- **Prefix_Type**: read only property combines Prefix and Type Properties
- TypeLabel: Defaulted on Feature Definitions (i.e., GRATE ELEV, RIM ELEV), populated from <u>2nd button</u>
- **Existing_Action_Label**: Defaulted on Feature Definitions (initially blank possible values TO REMAIN, TO BE REMOVED...), populated from <u>2nd button</u>
- **Grate_Rim_Elevation**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit1Label**: populated via app from <u>3rd button</u> from drainage properties
- **ConduittisExisting**: populated via app from <u>3rd button</u> from conduit item type properties
- **Conduit1Prefix**: populated via app from <u>3rd button</u> from conduit item type properties
- **Conduit1Size**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit1Prefix_Size**: : read only property that combined the conduit prefix and its size properties
- **Conduit1Type**: populated via app from <u>3rd button</u> from conduit item type properties
- **Conduit1Invert**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit1Angle**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit1Direction**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit2Label**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit2isExisting**: populated via app from <u>3rd button</u> from conduit item type properties
- **Conduit2Prefix**: populated via app from <u>3rd button</u> from conduit item type properties
- **Conduit2Size**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit2Prefix_Size**: read only property that combined the conduit prefix and its size properties
- **Conduit2Type**: populated via app from <u>3rd button</u> from conduit item type properties
- **Conduit2Invert**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit2Angle**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit2Direction**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit3Label**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit3isExisting**: populated via app from <u>3rd button</u> from conduit item type properties
- **Conduit3Prefix**: populated via app from <u>3rd button</u> from conduit item type properties
- **Conduit3Size**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit3Prefix_Size**: read only property that combined the conduit prefix and its size properties



- **Conduit3Type**: populated via app from <u>3rd button</u> from conduit item type properties
- **Conduit3Invert**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit3Angle**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit3Direction**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit4Label**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit4isExisting**: populated via app from <u>3rd button</u> from conduit item type properties
- **Conduit4Prefix**: populated via app from <u>3rd button</u> from conduit item type properties
- **Conduit4Size**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit4Prefix_Size**: read only property that combined the conduit prefix and its size properties
- **Conduit4Type**: populated via app from <u>3rd button</u> from conduit item type properties
- **Conduit4Invert**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit4Angle**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit4Direction**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit5Label**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit5isExisting**: populated via app from <u>3rd button</u> from conduit item type properties
- **Conduit5Prefix**: populated via app from <u>3rd button</u> from conduit item type properties
- **Conduit5Size**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit5Prefix_Size**: read only property that combined the conduit prefix and its size properties
- **Conduit5Type**: populated via app from <u>3rd button</u> from conduit item type properties **Conduit5Invert**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit5Angle**: populated via app from <u>3rd button</u> from drainage properties
- **Conduit5Direction**: populated via app from <u>3rd button</u> from drainage properties
- **OutgoingConduitLabel**: populated via app from <u>3rd button</u> from drainage properties
- **OutgoingConduitisExisting**: populated via app from <u>3rd button</u> from conduit item type properties
- **OutgoingConduitPrefix**: populated via app from <u>3rd button</u> from conduit item type properties
- **OutgoingConduitSize**: populated via app from <u>3rd button</u> from drainage properties
- **OutgoingConduitPrefix_Size**: read only property that combined the conduit prefix and its size properties
- **OutgoingConduitType**: populated via app from <u>3rd button</u> from conduit item type properties
- **OutgoingConduitInvert**: populated via app from <u>3rd button</u> from drainage properties
- **OutgoingConduitAngle**: populated via app from <u>3rd button</u> from drainage properties
- **OutgoingConduitDirection**: populated via app from <u>3rd button</u> from drainage properties
- **ConnectedPipes**: read only property combines the labels, if any for Conduit 1-5 and outgoing conduits connected to this node includes (In) or (Out). Ex (In) P-02, (Out) P-03

OHDOT_UtilityConduit

- **isExisting**: defaulted to true for existing feature Definitions, populated from <u>2nd button</u>
- **Prefix**: defaulted to EX. for existing feature definitions, populated from <u>2nd button</u>



- **Type**: Defaulted on Feature Definitions (i.e. ELECTRIC, GAS...) , populated from <u>2nd</u> <u>button</u>
- **Prefix_Type**: read only property combines Prefix and Type Properties
- **Size**: populated via app from <u>3rd button</u> from drainage properties
- **Prefix_Size**: read only property combines Prefix and Size Properties
- **Suffix**: Defaulted on Feature Definitions (initially blank), populated from <u>2nd button</u>

Updates

Version 1.0.0.0: Initial Release Version 1.0.0.1-6:

- Fixed baseline offset value being in meters rather than feet
- Fixed problem calculating Station/offset if alignment is in nested reference.
- Fixed node to node slope value. It was not converting to percentage (value needed multiplied by 100)
- Added new button to populate new conduit UDX properties from item type properties.
- Added help menu item with about dialog.
- Added checkbox to specify if the slope property should be multiplied by 100 (check this on if the DU slope property units is set to ft/ft)
- Updated populating Utility Size Item Type property correctly now that the field is formatted as text rather than a number.
- Switched the order of two buttons on the dialog
- Switched the order of the first two buttons on the dialog
- Removed multiple slope by 100 checkbox
- App will now handle unit conversations correctly for all length, elevation, and slope properties.
- Fixed Utility properties not getting filled out.
- Updated to work in ORD 10.11 (accounted for SDK changes)
- Fixed UDX properties not getting populated.

Version 1.0.0.7:

- Updated to include populating invert elevation for nodes (new item type property)
- Updated button to update UDX properties to include the Existing Action UDX property
- Fixed UDX start and stop alignment name properties not populating
- Fixed UDX Baseline Name, Station, offset properties for nodes not populating
- Updated to set the FeatureName, ConnectedPipes, and ConnectedPipes_withInvert Item Type properties with the Update Item Type Property Values from DU Drainage Properties button is selected.

Version 1.0.0.8:

• Code optimizations to improve processing times.

Version 1.0.0.9

• Updated to work ORD10.12. previously the station/offset values were being calculated incorrectly within ORD 10.12 due to a change in the api call. Will still work with older versions of ORD as well.

Version 1.0.0.10



- Updated code to work with headwall type nodes.
- Updated to handle 2023 version number change.

Version 1.0.1.0

• Complete remake to work with SSOT Item types. Remake only works if running within ORD 2023 or newer. Older versions of ORD will continue to work the old way.

Contacts

For any questions, suggestions, or problems with this document please contact the ODOT Office of CADD and Mapping Services by use of the following form on the ODOT website:

https://odot.formstack.com/forms/cadd_servicerequest