



OHDOT_DU_ItemTypeSync

Description

This app is a resizable C# Windows Presentation 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).

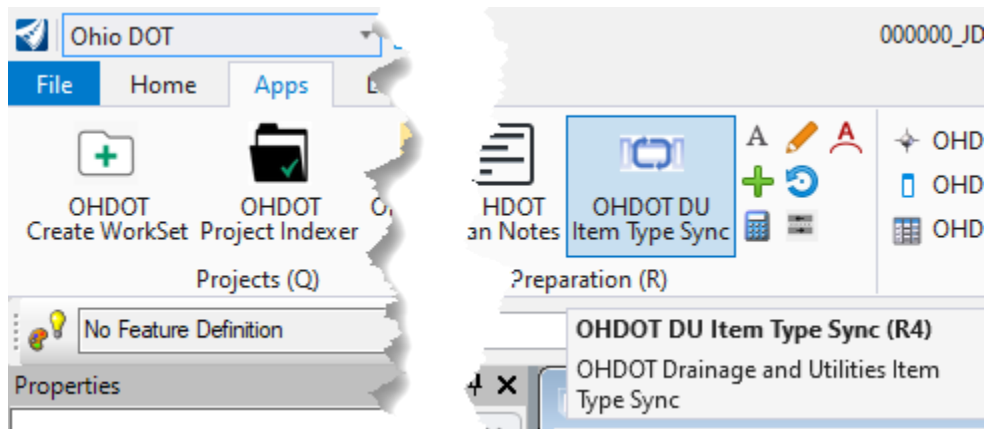
Loading the application

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 OHDOTItemTypeInfo; OHDOTItemTypeInfo OPEN

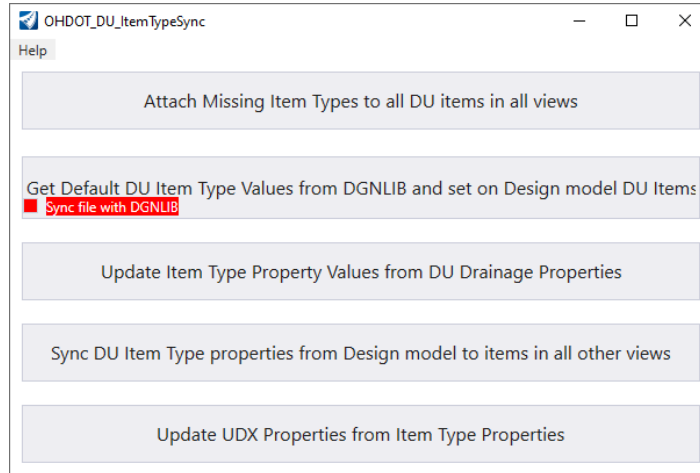
Note that the OHDOT workspace must be loaded.





Application Operation

Once the application is launched it will look as imaged below. This app should be run within the Drainage or Utility basemaps.



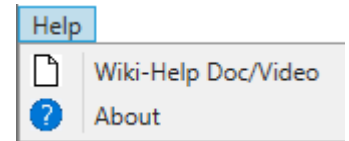
This application consists of a help menu and 5 buttons.

Help Menu:

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

Wiki-Help Doc/Video

Clicking this item will launch [this](#) 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
- Company
- Description



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 by 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 what's 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

OHDOT_StormSanConduit

- **FeatureName:** read only property pulling the Feature Name.
- **NodeToNodeLength:** populated via app from [3rd button](#) from drainage properties
- **isExisting:** defaulted to true for existing feature Definitions, populated from [2nd button](#)
- **Prefix:** defaulted to EX. for existing feature definitions, populated from [2nd button](#)
- **Size:** populated via app from [3rd button](#) 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 [2nd button](#)
- **Category:** Defaulted on Feature Definiens (i.e. STORM) used in XS annotation, populated from [2nd button](#)
- **NodeToNodeSlope:** populated via app from [3rd button](#) from drainage properties



- **StartNodeID:** populated via app from [3rd button](#) from drainage properties
- **StartNodeName:** populated via app from [3rd button](#) from drainage properties
- **StartElevation:** populated via app from [3rd button](#) from drainage properties
- **StopNodeID:** populated via app from [3rd button](#) from drainage properties
- **StopNodeName:** populated via app from [3rd button](#) from drainage properties
- **StopElevation:** populated via app from [3rd button](#) from drainage properties

OHDOT_StormSanNode

- **FeatureName:** read only property pulling the Feature Name.
- **BaselineName:** populated via app from [3rd button](#) from drainage properties, must have Baseline Reference property set.
- **Station:** populated via app from [3rd button](#) from drainage properties, must have Baseline Reference property set.
- **Offset:** populated via app from [3rd button](#) from drainage properties, must have Baseline Reference property set.
- **AbsOffset:** populated via app from [3rd button](#) from drainage properties, must have Baseline Reference property set.
- **OffsetSide:** populated via app from [3rd button](#) from drainage properties, must have Baseline Reference property set.
- **isExisting:** defaulted to true for existing feature Definitions, populated from [2nd button](#)
- **Prefix:** defaulted to EX. for existing feature definitions, populated from [2nd button](#)
- **Type:** Defaulted on Feature Definitions (i.e.. CB2-2A) , populated from [2nd button](#)
- **Prefix_Type:** read only property combines Prefix and Type Properties
- **TypeLabel:** Defaulted on Feature Definitions (i.e.. GRATE ELEV, RIM ELEV) , populated from [2nd button](#)
- **Existing_Action_Label:** Defaulted on Feature Definitions (initially blank possible values TO REMAIN, TO BE REMOVED...) , populated from [2nd button](#)
- **Grate_Rim_Elevation:** populated via app from [3rd button](#) from drainage properties
- **Conduit1Label:** populated via app from [3rd button](#) from drainage properties
- **Conduit1isExisting:** populated via app from [3rd button](#) from conduit item type properties
- **Conduit1Prefix:** populated via app from [3rd button](#) from conduit item type properties
- **Conduit1Size:** populated via app from [3rd button](#) from drainage properties
- **Conduit1Prefix_Size:** : read only property that combined the conduit prefix and its size properties
- **Conduit1Type:** populated via app from [3rd button](#) from conduit item type properties
- **Conduit1Invert:** populated via app from [3rd button](#) from drainage properties
- **Conduit1Angle:** populated via app from [3rd button](#) from drainage properties
- **Conduit1Direction:** populated via app from [3rd button](#) from drainage properties
- **Conduit2Label:** populated via app from [3rd button](#) from drainage properties
- **Conduit2isExisting:** populated via app from [3rd button](#) from conduit item type properties
- **Conduit2Prefix:** populated via app from [3rd button](#) from conduit item type properties
- **Conduit2Size:** populated via app from [3rd button](#) from drainage properties
- **Conduit2Prefix_Size:** read only property that combined the conduit prefix and its size properties



- **Conduit2Type:** populated via app from [3rd button](#) from conduit item type properties
- **Conduit2Invert:** populated via app from [3rd button](#) from drainage properties
- **Conduit2Angle:** populated via app from [3rd button](#) from drainage properties
- **Conduit2Direction:** populated via app from [3rd button](#) from drainage properties
- **Conduit3Label:** populated via app from [3rd button](#) from drainage properties
- **Conduit3isExisting:** populated via app from [3rd button](#) from conduit item type properties
- **Conduit3Prefix:** populated via app from [3rd button](#) from conduit item type properties
- **Conduit3Size:** populated via app from [3rd button](#) from drainage properties
- **Conduit3Prefix_Size:** read only property that combined the conduit prefix and its size properties
- **Conduit3Type:** populated via app from [3rd button](#) from conduit item type properties
- **Conduit3Invert:** populated via app from [3rd button](#) from drainage properties
- **Conduit3Angle:** populated via app from [3rd button](#) from drainage properties
- **Conduit3Direction:** populated via app from [3rd button](#) from drainage properties
- **Conduit4Label:** populated via app from [3rd button](#) from drainage properties
- **Conduit4isExisting:** populated via app from [3rd button](#) from conduit item type properties
- **Conduit4Prefix:** populated via app from [3rd button](#) from conduit item type properties
- **Conduit4Size:** populated via app from [3rd button](#) from drainage properties
- **Conduit4Prefix_Size:** read only property that combined the conduit prefix and its size properties
- **Conduit4Type:** populated via app from [3rd button](#) from conduit item type properties
- **Conduit4Invert:** populated via app from [3rd button](#) from drainage properties
- **Conduit4Angle:** populated via app from [3rd button](#) from drainage properties
- **Conduit4Direction:** populated via app from [3rd button](#) from drainage properties
- **Conduit5Label:** populated via app from [3rd button](#) from drainage properties
- **Conduit5isExisting:** populated via app from [3rd button](#) from conduit item type properties
- **Conduit5Prefix:** populated via app from [3rd button](#) from conduit item type properties
- **Conduit5Size:** populated via app from [3rd button](#) from drainage properties
- **Conduit5Prefix_Size:** read only property that combined the conduit prefix and its size properties
- **Conduit5Type:** populated via app from [3rd button](#) from conduit item type properties
- **Conduit5Invert:** populated via app from [3rd button](#) from drainage properties
- **Conduit5Angle:** populated via app from [3rd button](#) from drainage properties
- **Conduit5Direction:** populated via app from [3rd button](#) from drainage properties
- **OutgoingConduitLabel:** populated via app from [3rd button](#) from drainage properties
- **OutgoingConduitisExisting:** populated via app from [3rd button](#) from conduit item type properties
- **OutgoingConduitPrefix:** populated via app from [3rd button](#) from conduit item type properties
- **OutgoingConduitSize:** populated via app from [3rd button](#) from drainage properties
- **OutgoingConduitPrefix_Size:** read only property that combined the conduit prefix and its size properties



- **OutgoingConduitType:** populated via app from [3rd button](#) from conduit item type properties
- **OutgoingConduitInvert:** populated via app from [3rd button](#) from drainage properties
- **OutgoingConduitAngle:** populated via app from [3rd button](#) from drainage properties
- **OutgoingConduitDirection:** populated via app from [3rd button](#) 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 [2nd button](#)
- **Prefix:** defaulted to EX. for existing feature definitions, populated from [2nd button](#)
- **Type:** Defaulted on Feature Definitions (i.e. ELECTRIC, GAS...) , populated from [2nd button](#)
- **Prefix_Type:** read only property combines Prefix and Type Properties
- **Size:** populated via app from [3rd button](#) from drainage properties
- **Prefix_Size:** read only property combines Prefix and Size Properties
- **Suffix:** Defaulted on Feature Definitions (initially blank), populated from [2nd button](#)

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.

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
