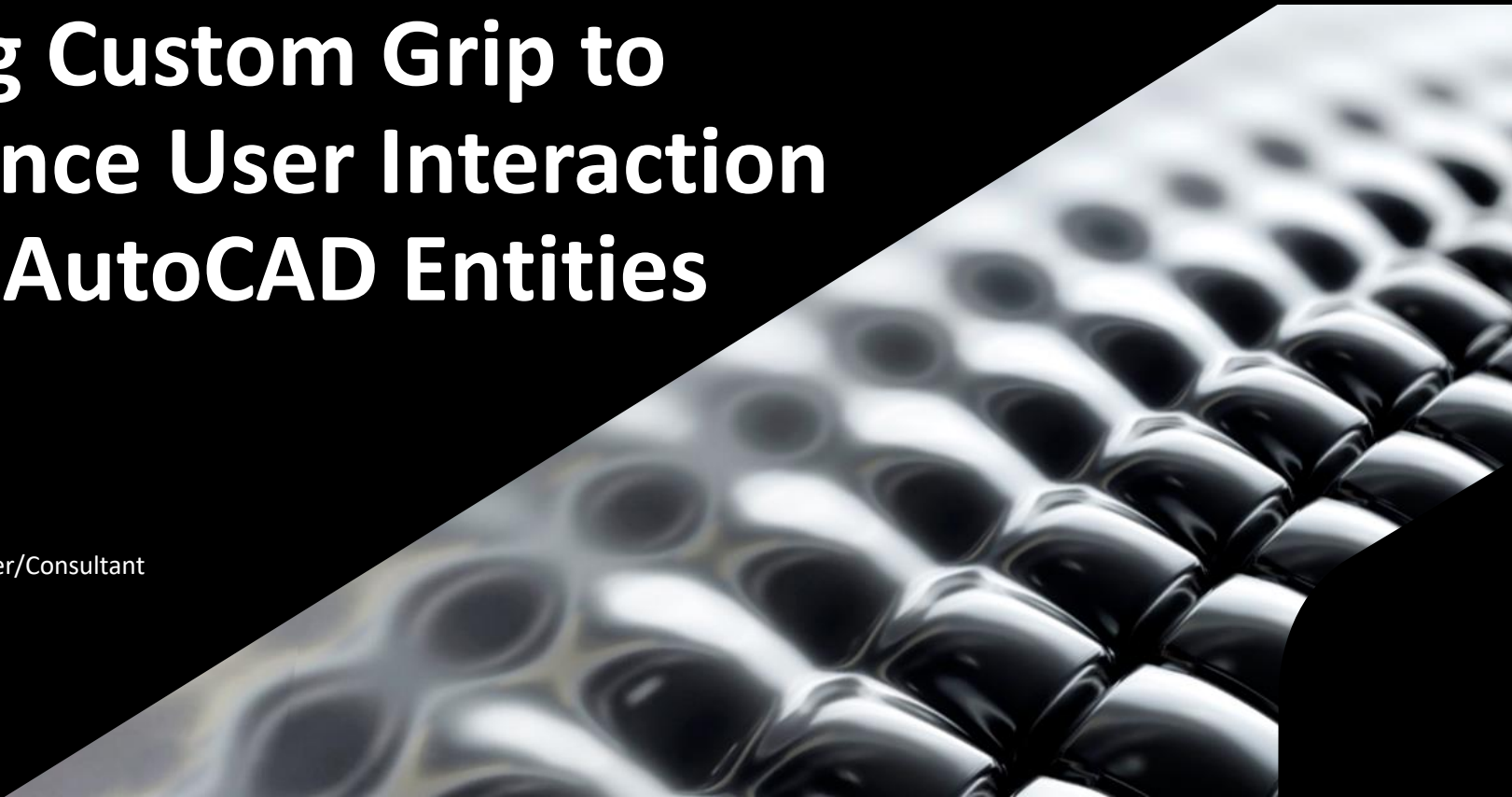




Using Custom Grip to Enhance User Interaction with AutoCAD Entities

Norman Yuan
Software Developer/Consultant



About the Speaker

Norman Yuan

Software Developer, Independent IT Consultant

Email: norman.yuan@live.com

Blog: [Drive AutoCAD With Code](#)



The Audience of This Class

Who You Are

Someone who does AutoCAD customization and programming

What Is Your Knowledge/Skill Level

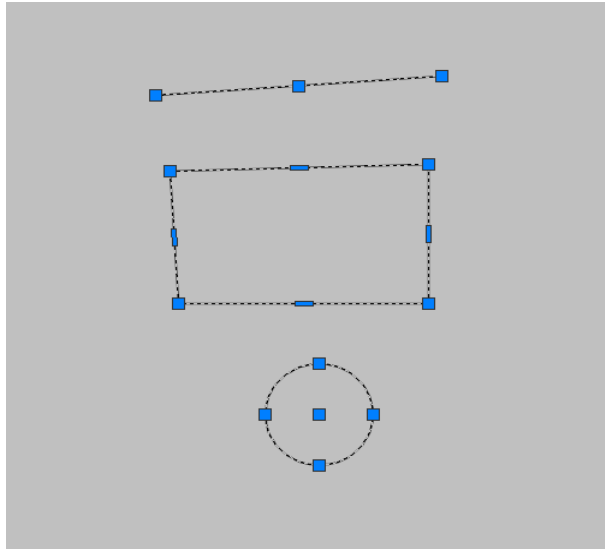
- Basic AutoCAD .NET API programming, beginner or intermediate level
- Basic general MS .NET programming, again, beginner or intermediate level
- Knowing little about AutoCAD .NET API, but familiar to basic programming knowledge gained from AutoCAD LISP/VBA work

AutoCAD Entity Grip

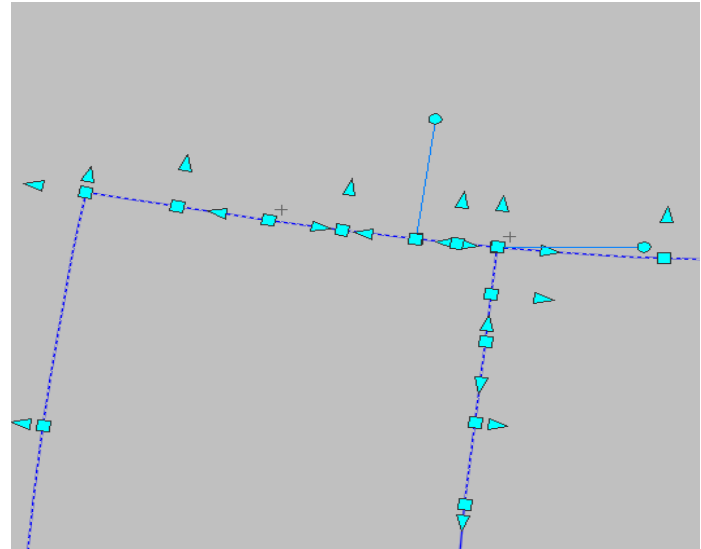
Entity Grip

When entity is selected in AutoCAD Editor, a set of grips are presented to show significant geometrical information of the entity and to provide a shortcut of certain typical operation to the selected entity. Besides grips built-in with plain AutoCAD, other AutoCAD verticals, such as C3D/Arch/MEP use vertical specific entity grips extensively.

Basic AutoCAD Entity Grips



AutoCAD Vertical Custom Entity Grips

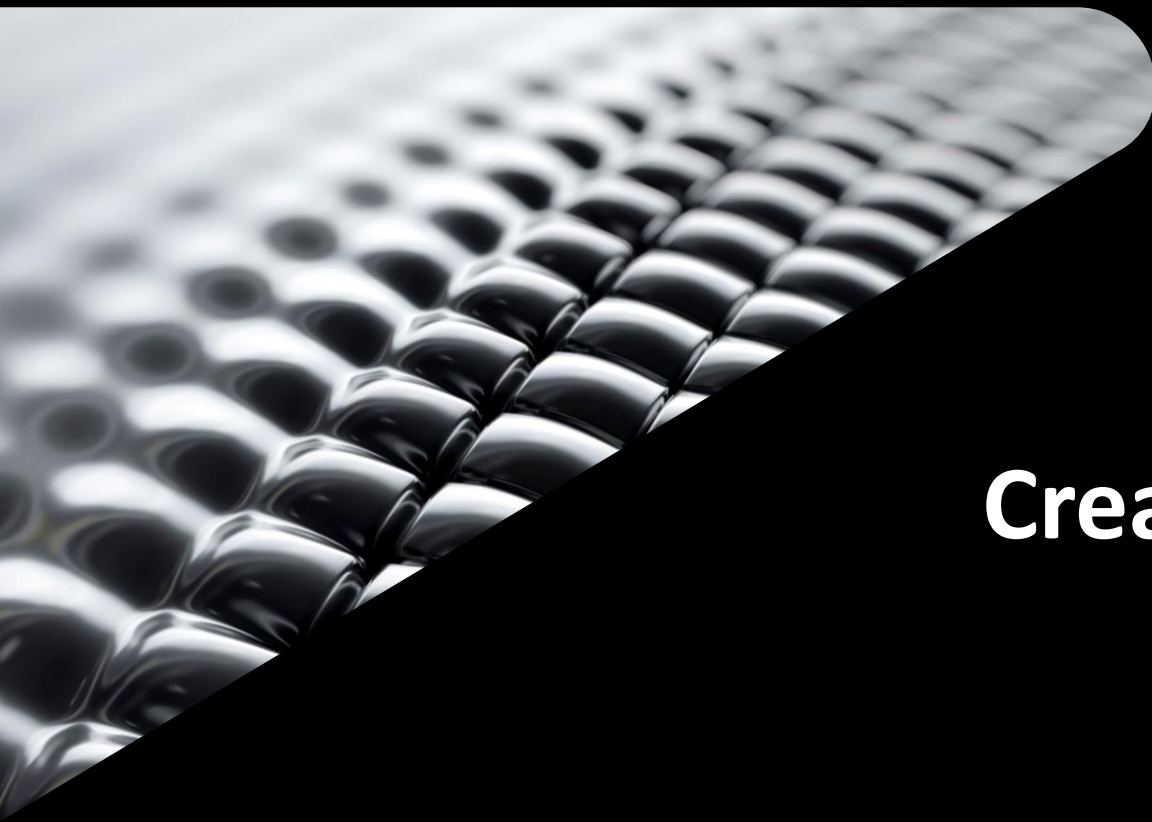


Custom Entity Grips

AutoCAD .NET API allow us to customize entity's grips: adding extra custom grips and, if desirable, suppress the built-in grips.

Creating custom grips involves 2 tasks:

- Create custom grips that visually appear when an entity is selected. It is also optional to suppress built-in grips, if needed
- Associate actions to the custom grips to allow users to interact with them to trigger certain operation against the entity



Create Custom Grip

Example 1: A Do-Nothing Grip

The sample code project (in C#) includes a simple custom grip overrule, which adds a custom grip at an closed polyline's GeoExtents center.

This grip overrule only applies to a polyline that is closed. It has option to suppress built-in grips.

<https://autode.sk/3OAiweM>

Example 2: An Incremental Drag Grip

This custom grip is used to drag an end of a Line entity to change its length by given increment. The custom grip overrule applies this grip to entity of Line type.

<https://autode.sk/3PD5EWu>

Example 3: An Attribute Move Grip

This custom grip is used to drag an attribute for moving. This example shows how to apply the grip overrule to a nested entity (AttributeReference entity in a BlockReference). That is, while the custom grip belongs to the BlockReference, it looks like a grip of the attribute.

<https://autode.sk/3zgUVdw>



Custom Grip with Context Menu

Custom Grip with Context Menu

Very often, an action against selected entity triggered from a grip may be different with different options. While it is possible to add more custom grips so that each grip does one thing, this approach would certainly lead to too many grips crowded together ugly. Here is where a grip with context menu comes handy.

Let Custom Grip Show Context Menu

There are different ways to bring context menu to a custom grip:

- Override custom GripData class's OnRightClick() method, which returns a collection of IMenulitem interface object as menu item. While it is possible to use existing context menu in .NET Framework, the actual implementation of the IMenulitem interface might be quite challenging.
- Derive a custom “class extension object” from MultiModeGripPE class and add it to the overruled entity via RXClass.AddX() method. With the custom MultiModeGripPE class in place, the custom Grip will automatically show a context menu when the mouse cursor hover the grip.

Example: An Attribute Multi-Action Grip

This custom grip has 3 action options: increasing and reducing attribute's text height, and toggling attribute's Invisible property.

<https://autode.sk/3J9eGYS>



Using Custom Grip with Data

Custom Grip with Data

Many CAD operations against selected entities deal with data that is not from the entity itself. For example, the attribute values of a block reference are often updated with data from external data source (database, or data files).

Using custom grip to deal with data properly can lead a better user experience (remember, grip is mainly meant for user to interact with the selected entity).

Example – Block Data Grip

Many CAD operations against selected entities deals with data that is not from the entity itself. For example, the attribute values of a block reference are often updated with data from external data source (database, or data files).

Using custom grip to deal with data properly can lead a better user experience (remember, grip is mainly meant for user to interact with the selected entity).

The Scenario

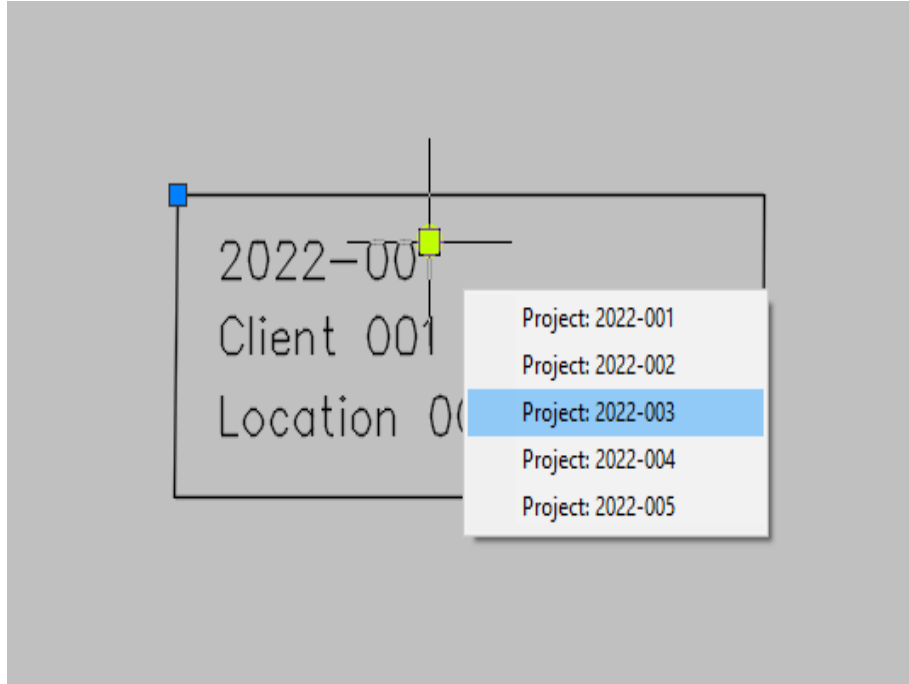
A drawing has multiple block references of a block, which has a few attributes which are to be updated with a set of data from a list of data records from external data source. For various reasons the attribute update operation requires user to do it manually.

The usual way to update the attributes with external data is to identify a target block reference (user selects it, for example), and run a command to get the attribute updated. During the command, user would be presented a list of the data record and has to select one record.

The Data

```
[
  {
    "project": "2022-001",
    "location": "Location 001",
    "client": "client 001"
  },
  {
    "project": "2022-002",
    "location": "Location 002",
    "client": "client 002"
  },
  {
    "project": "2022-003",
    "location": "Location 003",
    "client": "client 003"
  },
  {
    "project": "2022-004",
    "location": "Location 004",
    "client": "client 004"
  },
  ...
]
```

The Block



The Grip's Operation

<https://autode.sk/3cKHVoy>

A Bonus Example: An Attribute Edit Grip

In this example, data that is stored in the block is used to update block's attributes when the custom grip is clicked.

<https://autode.sk/3oxMSE0>

Class Summary

Summary

- Entity grip is meant for user to interact with selected entity
- Custom grip provide a user-friendly shortcut for user to perform certain operations on selected entity, which can, and usually, be executed from a command manually or automatically.
- Only associate operations on entities that absolutely needs user interaction with selected entities, because grips are only available when user selects an entity without a command being active.

Closing

Resources

<https://www.keanw.com/2009/08/knowning-when-an-autocad-object-is-grip-edited-using-overrules-in-net.html>

<https://forums.autodesk.com/t5/net/override-dynamic-block-grip-editing/m-p/3125022/highlight/true#M24726>

<https://forums.autodesk.com/t5/net/gripdata-onhover-how-to-add-a-context-menu/m-p/5667410#M44865>

<https://adndevblog.typepad.com/autocad/>

<https://through-the-interface.typepad.com/>

Contact

Email: norman.yuan@live.com

Blog: [Drive AutoCAD with Code](#)

