

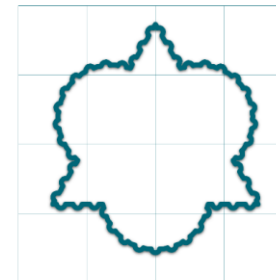
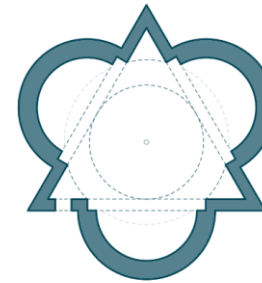
Digital Representation

BMEEPAGA205

➤ 2D drafting

➤ 3D modeling

lecturer: László Strommer PhD



Command Entry

Ribbon Bar / Tool Icon
Command icons
Show the pull-down menu (MenuBar)

Command Prompt
Every command appears in the command bar
Toggleable Pointer Input

PaperSpace

The drawing file can contain a model and several layout tabs.
Layouts can be defined, inserted, copied, removed, etc.
The following can be specified for layout:
→ printer type, page size, orientation,
→ plot setup (e.g. file name, date),
→ raster, etc.
→ print lineweight
The layout is the same as the print preview:
→ the content of the non-printed layers are visible,
→ and it is possible to draw in it.
Each layout has a unique 2D PaperSpace
contains standard elements (e.g. captions, lines)
and special ViewPort elements depicting the model.

View · Visual Styles

2D projection is performed by the program, the view card displays 3D vectors (axis)
can be stored in vector and/or plot format.
Pixel format
→ Drawing elements can be separated using layers,
→ modification usually results in loss of information
(freezing, random rotation, ... or sharpening).
Vector format
→ Drawing elements are geometric objects
(e.g. circle, polyline, spline).
→ Geometric transformations do not result in quality loss.
Mixed format
→ Pixel gradients, backgrounds, patterns, etc.,
but e.g. vector text, shapes.

Pixel / Vector

The computer displays a raster image—the drawing elements
can be stored in vector and/or plot format.
Pixel format
→ Drawing elements can be separated using layers,
→ modification usually results in loss of information
(freezing, random rotation, ... or sharpening).
Vector format
→ Drawing elements are geometric objects
(e.g. circle, polyline, spline).
→ Geometric transformations do not result in quality loss.
Mixed format
→ Pixel gradients, backgrounds, patterns, etc.,
but e.g. vector text, shapes.

Rendering

The level of realism of the visual styles can be adjusted
depending on the hardware (e.g. transparency, shadows).
Rendering can produce an even more realistic image,
but not in real time, but over several minutes.
→ the quality (and time) can be controlled,
→ the resolution of the output file can be set,
→ lights, materials, background can be used
(however, there can also appear in a realistic visual style).
Curved surfaces are approximated by a mesh with
controllable density (vertices / faces).
Surfaces whose normal points in the opposite
direction can be excluded from the calculation.

Drawing

Placing new elements into the drawing.
→ The drawing limit can be chosen freely (WxH).
→ The characteristic points of the drawing elements
can be specified either by the mouse or the keyboard,
or by combining the above.
→ Usually, several methods can be selected, e.g. for a circle
center / radius / diameter, 2 diameter endpoints, 3 points.
→ The saved elements are usually uniform,
independent from the method of creation.
→ Exceptions e.g. "associative" hatching_array
(in 3D the box and the extended rectangle are different).

Blocks

The use of blocks makes it easier to handle frequently used elements
(e.g. selecting each component of a complex shape when copying).
→ The drawing file will be smaller, as the block definition only needs to be saved once,
and it is enough to store the unique properties (e.g. location) of the inserted instance.
→ The blocks can be redefined, making it easier to make global changes.
* If the only goal is to keep the elements together,
it is enough to use the Group command.
Group operations can be switched on and off
(PLOTStyle, Ctr+L=SHA+CTRL).

3D Coordinate System

Cartesian x, y, z
→ x, y, z distances from the starting point
(from the current origin or from the last point)
Cylindrical d, c, a, z
→ the projection of the distance in the xy-plane,
→ the angle from the x-axis,
→ and the z coordinate.
Spherical d, c, a, b
→ distance from the starting point,
→ the angle from the x-axis,
→ the angle from the xy-plane.
Accuracy check: On and Off

Colors

Light visible to the human eye is the wavelength range of ~380 to 780 nm in the electromagnetic spectrum.

Information Exchange

→ Data exchange via the clipboard
→ Screenshots -
maximum data loss
→ Paste special command -
the format can be selected
→ Paste link -
the file can be edited with the original program
→ Data exchange using files
→ Intermediate file format (e.g. PDF, DWG, DXF) or
→ native format of a more common program (e.g. DWG).

Layers

Layers make the structure of the drawing more transparent and more organized.
→ The name of the layers can be freely changed (except 0), organized into a system (=templates).
→ All elements of the drawing are placed on figurative, three-dimensional layers.
→ Newly drawn elements are placed on the current layer.
→ Copies of existing elements are typically placed on the original layer.
Layer Status
→ Hidden/locked elements on a locked layer cannot be modified
→ On/Off: the content of a locked layer is not displayed
→ Thaw/Freeze: the content of a frozen layer is not generated, so it is invisible, unprintable,
(the current layer cannot be frozen)
→ Freeze in current / Thaw viewport: like freeze but only in given viewport
→ Print: the content of a frozen layer is not printed (even if it is visible on the screen)

Exporting / Properties

Drawing elements can be copied between drawings—but the pasted elements don't always appear the same.
Every element refers to its layer and linetype, some to text and/or dimension style, block definition.
→ General rule for these properties:
→ If an element layer, linetype, style has the same name exists in the target file, it is used unchanged,
→ if it doesn't exist, it is created automatically, as defined in the source file.
If a linetype/block has its dashed dimensions are pasted into a drawing (Ctrl+C / Ctrl+V).
→ If a block definition with the same name exists, that will be inserted instead,
→ If not, but a linetype/block definition exists, then the new linetype/block will use that,
→ If a dimension layer exists, all elements of the block with its layer color will inherit its color,
→ If a dashed linetype definition exists, the dimensions may appear differently from the original,
→ If a linetype style exists, the dimensions will be displayed according to that (e.g. with different size),
→ If not, but a linetype style exists, the dimension text is displayed according to that (e.g. using different font).

Model Types

2.5D model - thickness
3D surface model
3D solid model
3D primitives or 2D - Extrude / Revolve
→ Boolean operations or e.g. Slice
Region & Revolve
Law & Revolve
Law & Revolve
Law & Revolve

Image Formats

Bitmap
→ uncompressed format
→ large file size, low computing requirements
→ lossy and lossless, transparency, animation
→ less compression, but max. 256 colors
→ animation (poor efficiency and quality)
PNG
→ lossless compression + computing
→ palette or palette, transparency (4-8 bit)
JPEG
→ lossy compression + quality / loss of fidelity
→ photos, rendered images
WEBP
→ all-in-one solution, more efficient compression
→ lossy and lossless, transparency, animation
→ lossless compression, but max. 256 colors
→ layers, masks (adjustment layers)
RAW / DNG
→ native format of digital cameras
→ images with greater color depth (HDR)

Automatization Levels

→ A program module is a procedure written in a
programming language (AutoLisp, VBA, ...)
→ ActiveX enables you to manipulate AutoCAD
e.g. from an Excel VBA.
→ A macro (Action Macro, ACTX) is
a recorded command sequence.
→ Buttons and menus can be customized
using DDESXPL expressions.
→ A script is a list of program instructions.
→ Command aliases can be used (PCL) or
can be customized.

CAD

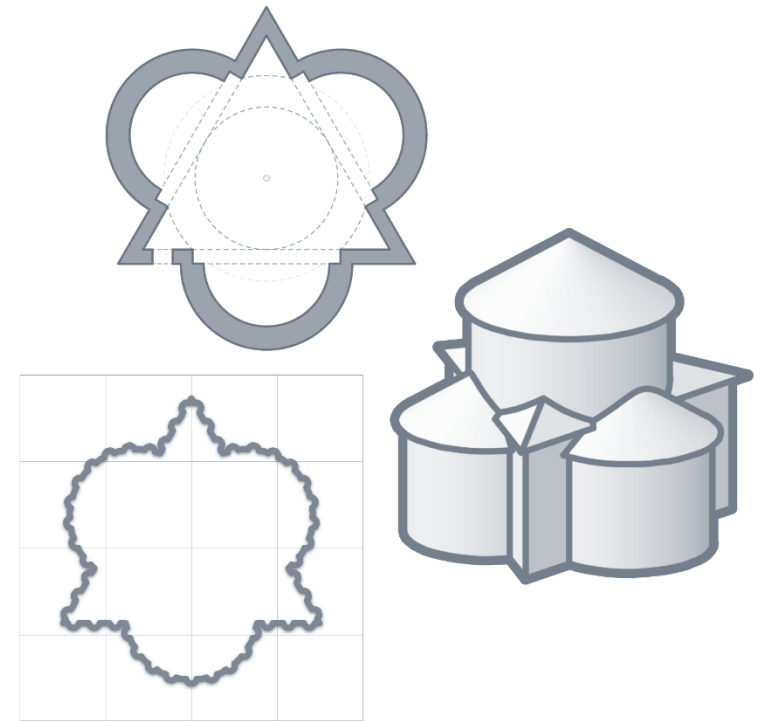
➤ Basics

➤ 2D drafting

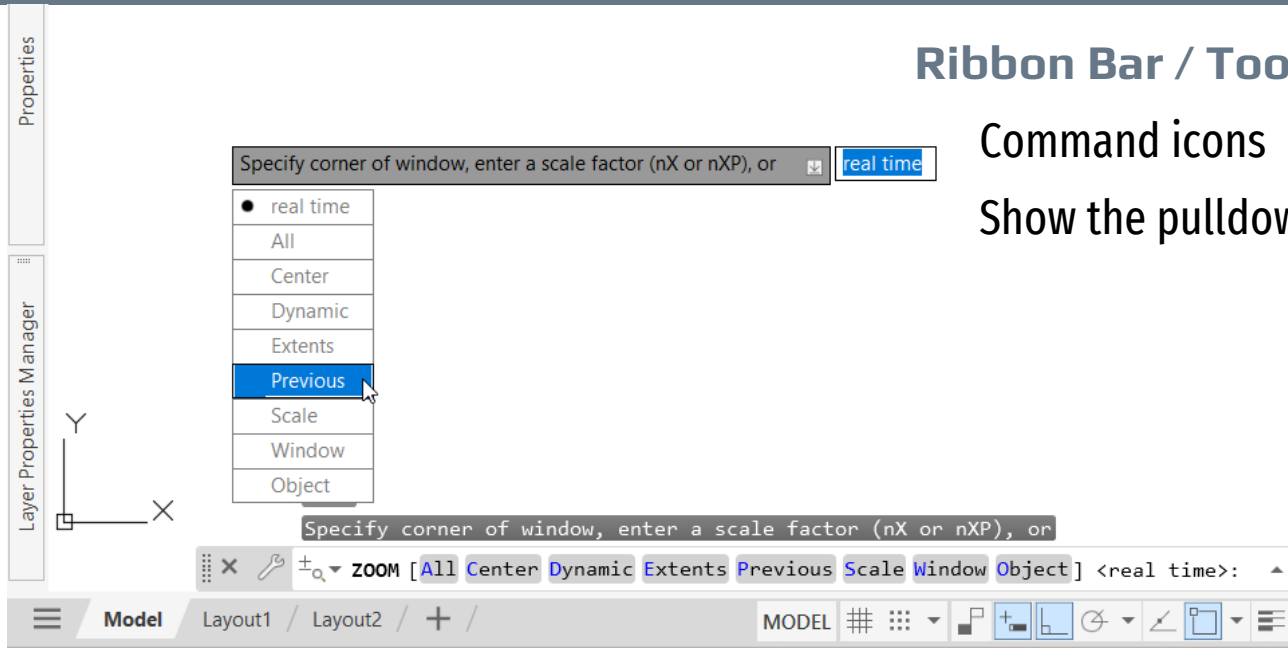
Digital Representation

BMEEPAGA205

László Strommer PhD.



Command Entry



Ribbon Bar / Tool Icon

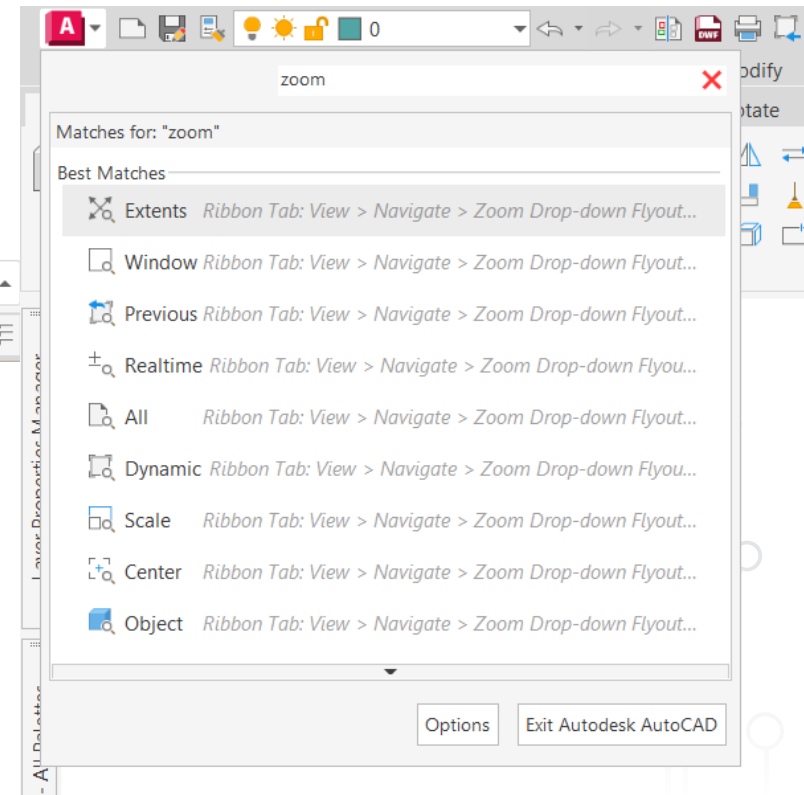
Command icons

Show the pulldown menu (MenuBar 1)

Command Prompt

Every command appears in the command bar

Toggleable Pointer Input



Coordinates · 2D

Cartesian coordinates: x, y

→ the x and y projections of the distance from the current origin or the last point

Polar coordinates : $d < a$

→ distance from the current origin or the last point, and the angle relative to the x-axis

Absolute coordinates : #

→ (*DynPiCoords* = 0)

→ coordinates measured from the current origin

Relative coordinates : @

→ coordinates measured from the last point

The most common way is to show the direction with the mouse and enter the distance from the keyboard ↵.

For the sake of cross-language compatibility, all language versions of AutoCAD...

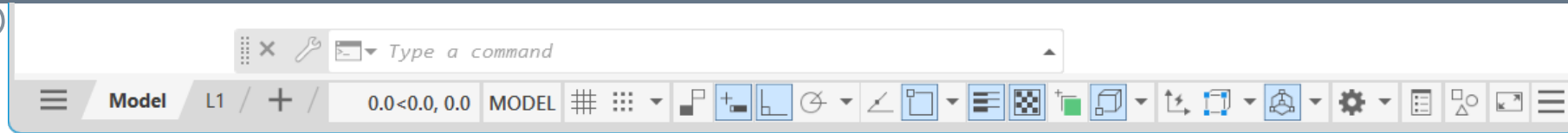
→ use point as decimal symbol, and comma as list separator,

→ variable names are always English, and

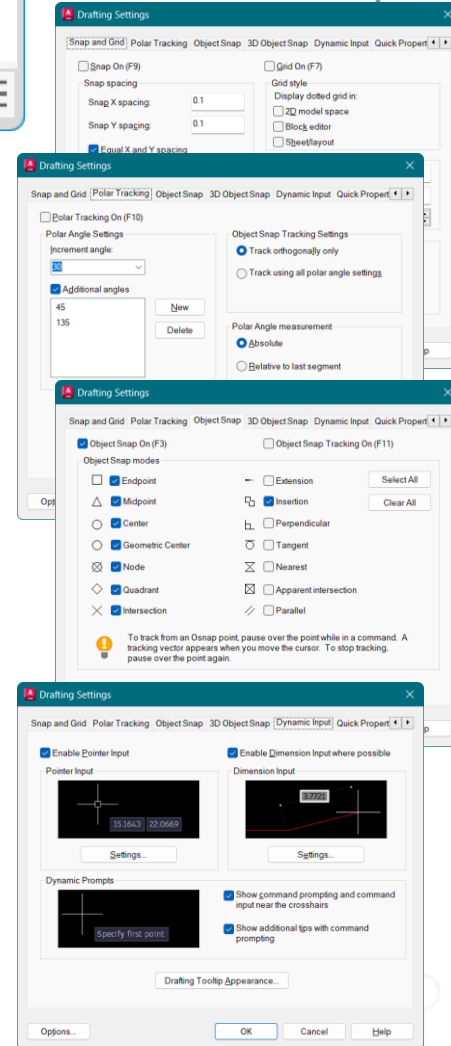
→ all English commands can be used after an underscore (e.g. _Line).



Drawing Aids · 2D



- Coord** → coordinates (abs. cart. / rel. polar / geographic) `Ctrl+I` · *Coords* 0-1-3
- Grid** → displaying points / lines at a given distance as visual help `F7` · *GridMode* 0-1
- Snap** → restricting cursor movement to grid points at a given distance `F9` · *SnapMode* 0-1
- Dyn** → dynamic (moving) entry field for commands or data `F12` · *DynMode* 0-3
- Ortho** → restricting vectors to perpendicular directions `F8` · *OrthoMode* 0-1
 - follows the angle of the grid → overrules the Snap mode in the given direction
- Polar** → restricting vectors to given angles `F10` · *PolarMode* 0-15
- OTrack** → showing snapping reference lines `F11` · *AutoSnap* 0-63
- OSnap** → showing characteristic points of drawing elements `F3` · *OSMode* 0-1151-16k
 - “running” object snap overrules the Snap and Ortho modes



Drawing

- Placing new elements into the drawing.
- The **Drawing Unit** can be chosen freely (M=1:1).
- The characteristic points of the drawing elements can be specified either by the mouse or the keyboard, or by combining the above.
- Usually, several methods can be selected, e.g. for a circle: center + radius / diameter, 2 diameter endpoints, 3 points.
- The saved elements are usually uniform, independent from the method of creation.
 - Exceptions: e. g. “associative” hatching, array (in 3D the box and the extended rectangle are different).



Point



Line



Arc



Pline



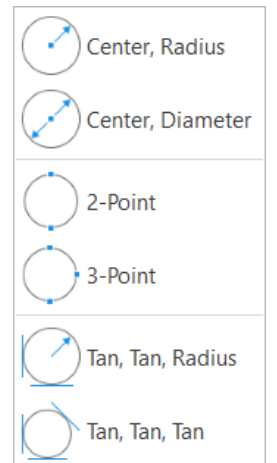
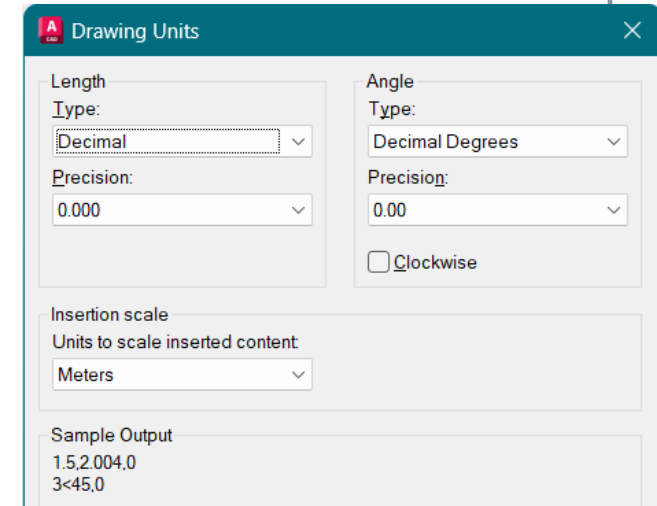
Circle



Ellipse



Spline

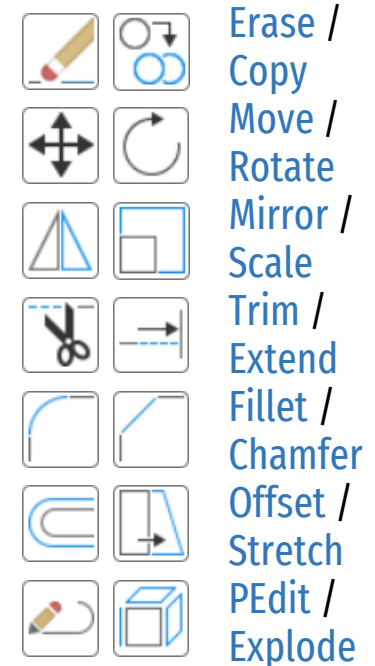


Modifying, Selection

To use modifying commands, the elements must be selected first – either before or after starting the command. These commands can also change the element type (e.g. Circle +Trim → Arc).

Selection methods

- **Auto**: clicking in an empty area specifies the first point of the diagonal of a selection box –
 - second point to the right → elements within the frame are selected (**Window**),
 - to the left → elements crossing the boundary are selected also (**Crossing**)
- **WP/CP**: Window/Crossing selection modes with a polygonal boundary
- **Previous**: re-selecting the previous item list
- **Last**: selecting the last created element
- **All**: selection of all available drawing elements
- **Undo**: undoing the last selection
- By pressing Shift, elements can be removed from the selection.
- The selection can be reversed with the **Remove / Add** options.



Element Types

The saved elements are usually uniform, independent from the method of creation.

The same item can be produced by multiple commands...

→ LWPOLYLINE element can be created for example...

→ by drawing a **PLine** a **Polygon** or a **Rectangle**,

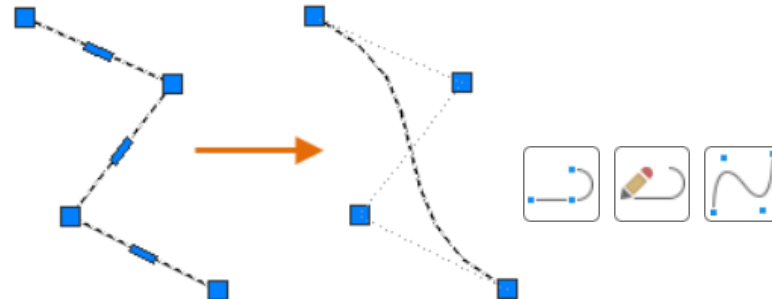
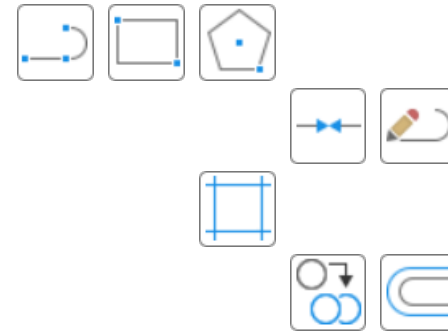
→ by connecting existing line and arc elements with **Join** or **Pedit**,

→ by drawing the outline of existing elements with **Boundary**,

→ by modifying existing polylines with **Stretch**, **Offset**.

→ SPLINE element can be created for example...

→ by drawing a polyline with **PLine**,
then modifying it by **Pedit / Spline**,
then converting it by the **Object** option
of the **Spline** command.



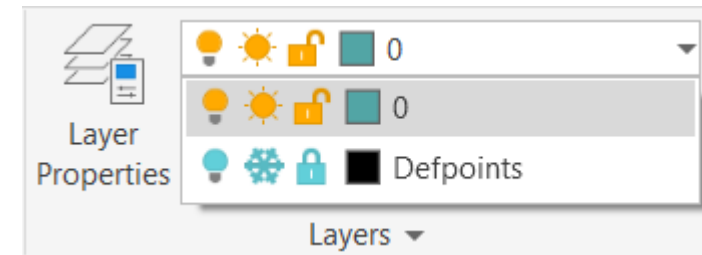
Layers

Layers make the structure of the drawing more transparent and more organized.

- The name of the layers can be freely changed (except 0), organized into a system (→ templates).
- All elements of the drawing are placed on figurative, three-dimensional layers.
 - Newly drawn elements are placed on the **Current** layer.
 - Copies of existing elements are typically placed on the original layer.

Layer Status

- **Unlock/Lock**: elements on a locked layer cannot be modified
- **On/Off**: the content of a turned off layer is not displayed
- **Thaw/Freeze**: the content of a frozen layer is not generated, so it is invisible, unprintable, (the current layer cannot be frozen)
- **Freeze in current / new viewport**: like **Freeze** but only in given viewports
- **Print**: the content of a frozen layer is not printed (even if it is visible on the screen)



Properties

The general properties are the same for all elements.

→ For color, linetype and thickness :

→ a specific value can be set (e.g. white, dashed, 0.3 mm),

→ the properties of the block containing it can be inherited (later),

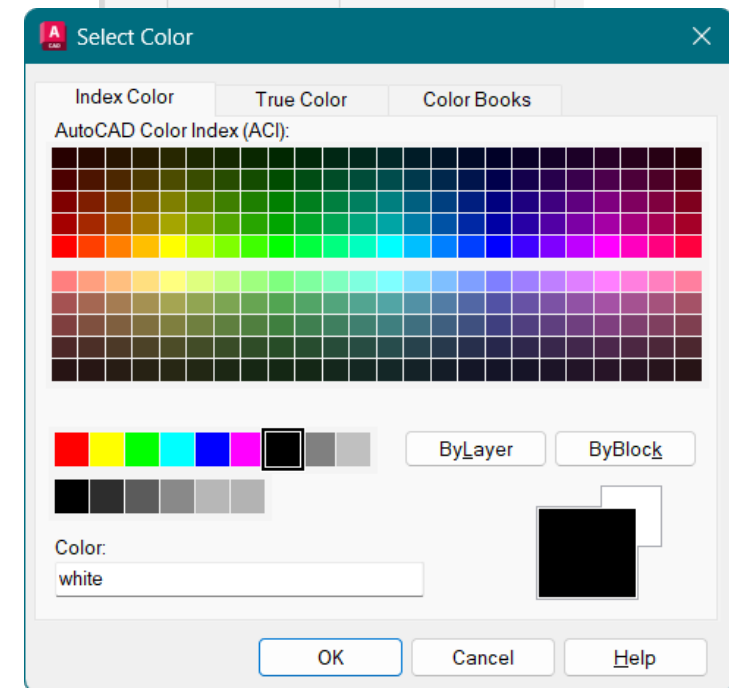
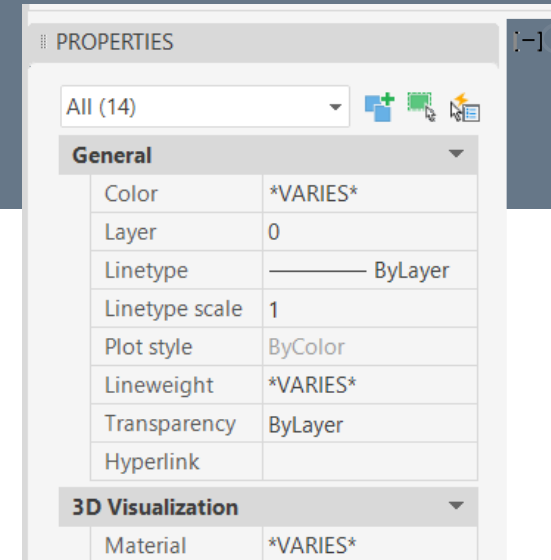
→ the properties of the element's layer can be inherited.

→ Layers are invisible – they set the default properties of their elements.

→ The line weight cannot be accurately judged on the screen, so the color mostly helps to identify the layer of the elements (especially in 2D).

Colors

→ seven basic colors by names for the (**white** → inverse),
the standard colors of the **AutoCAD Color Index** by numbers (1-255),
additional colors by RGB value or Pantone name.



CAD

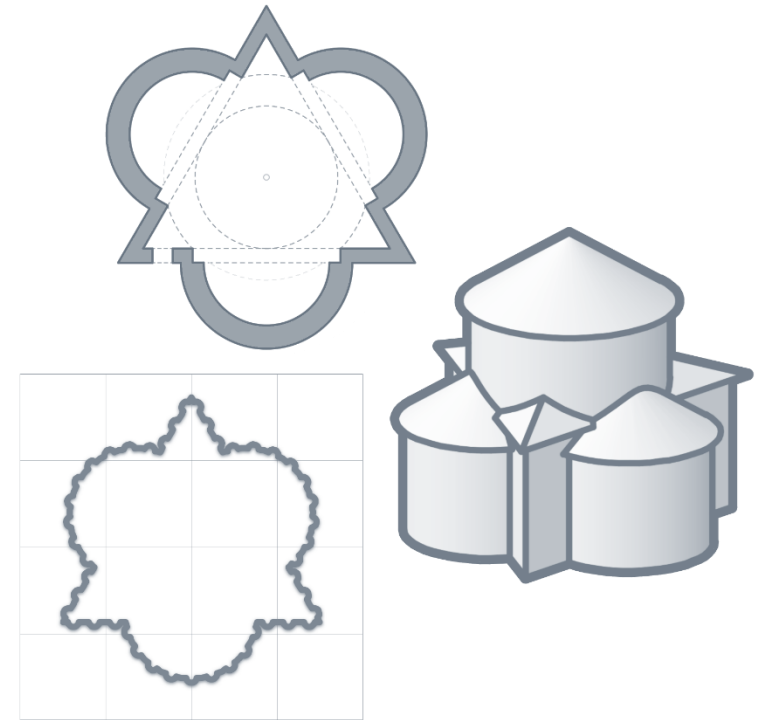
➤ **PaperSpace**

➤ **Blocks**

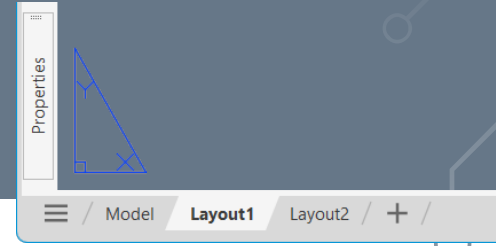
Digital Representation

BMEEPAGA205

László Strommer PhD.



PaperSpace



The drawing file can contain a model and several layout tabs.

Layouts can be deleted, inserted, copied, renamed, etc.

→ The following can be specified for layout:

→ printer type, page size, orientation,

→ plot stamp (e. g. file name, date),

→ printed lineweight.

→ The layout is not the same as the print preview:

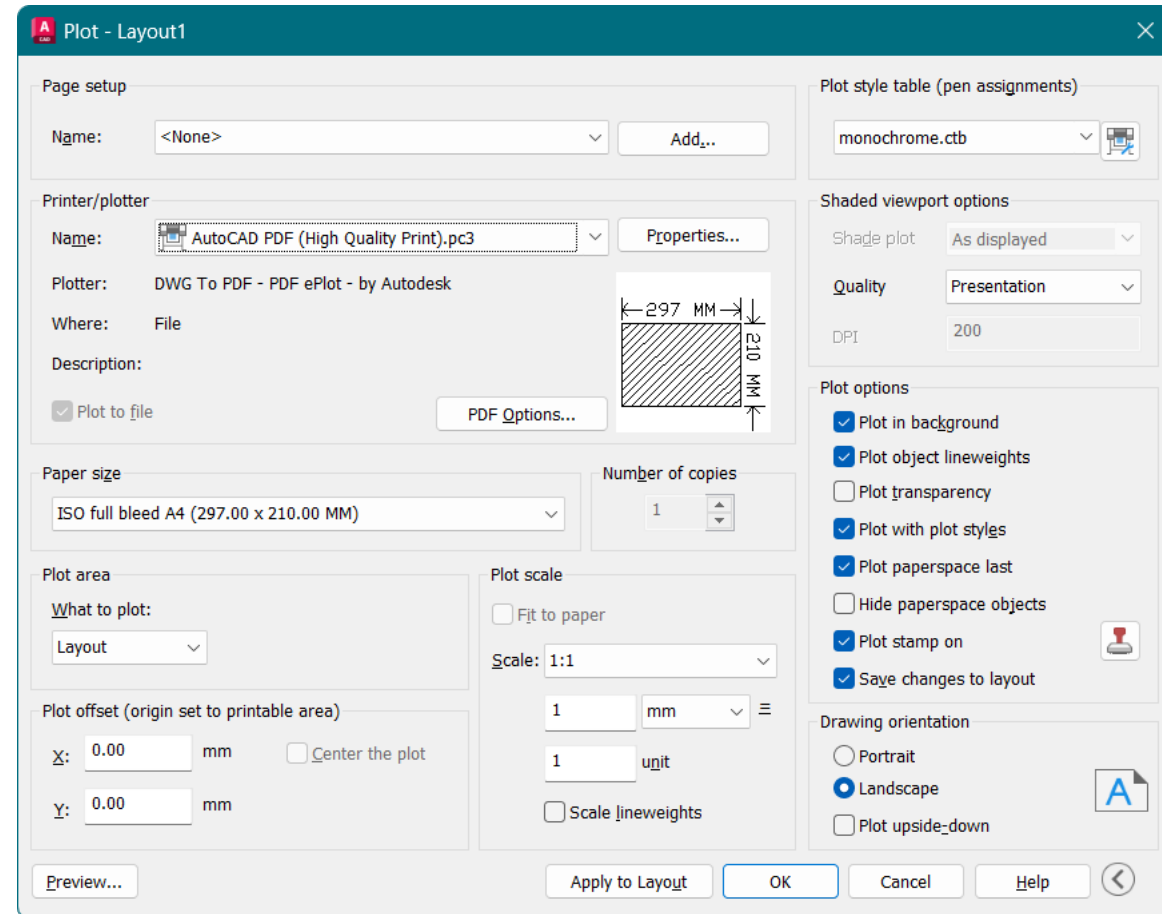
→ the content of the non-printed layers are visible,

→ and it is possible to draw in it.

→ Each layout has a unique 2D PaperSpace

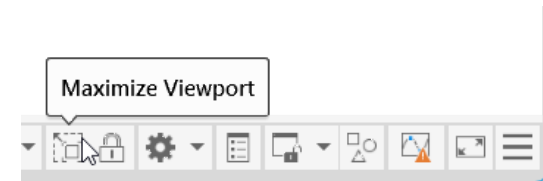
contain standard elements (e.g. captions, lines)

and special ViewPort elements depicting the modell.



Viewport

- A **Viewport** always shows the current state of the model.
- However, layers can be frozen per ViewPort, so their content is not necessarily the same.
- By double-clicking inside a ViewPort, we can move into **ModelSpace**, where we can modify the model itself – and the change will be visible in all viewports.
- After entering **ModelSpace**, we can maximize the ViewPort to full size, or we can switch between the **ViewPorts** (even during a command).
- The scale of the view can be adjusted:
 - if the model space unit is meter, the paper space unit is millimeter, the scale of a 1:50 drawing is 20xp ($50 \times 20 \text{ mm} = 1000 \text{ mm} = 1 \text{ m}$),
 - the list of standard scales can be modified (**ScaleListEdit**),
 - the scale and view of the viewport can be locked (**Display Locked**).

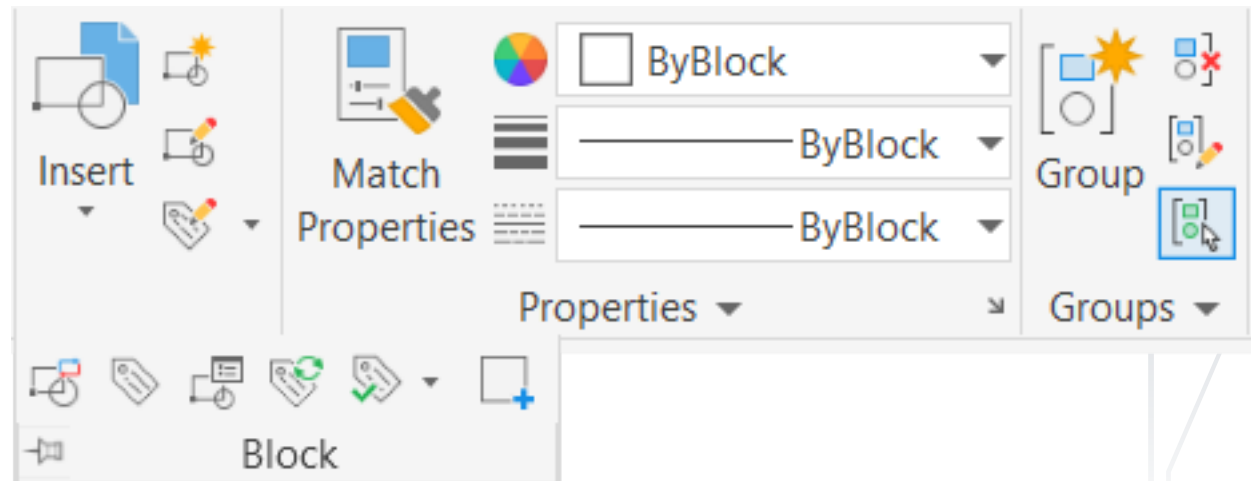


Blocks

The use of blocks makes it easier to handle frequently used elements (e.g. selecting each component of a complex shape when copying).*

- The drawing file will be smaller, as the block definition only needs to be saved once, and it is enough to store the unique properties (e.g. location) of the inserted instance.
- The blocks can be redefined, making it easier to make global changes.

* If the only goal is to keep the elements together, it is enough to use the **Group** command. Group selection can be switched on and off (*PickStyle*, `Ctrl+Shift+A`).



Block Definition

A block...

- can be defined with the [Block](#) command
- must have a unique name in the drawing file ([Purge](#)),
- can contain drawing elements and other blocks (except a block containing itself),
- should have an insertion point that can be easily specified when inserted
- should have elements with 0-degree rotation angle to facilitate insertion.

A defined block can be inserted into the drawing with the [Insert](#) command (or by copying an existing instance).

By specifying an existing name, the block can be redefined:

- every instance of the block takes over the new definition,
- the base point and rotation angle should not be changed ([RefEdit](#)).

› **See also:** [Layers](#) · [Properties](#)

Block Properties

Color (linetype, linewidth...) and layer of block elements:

Given Color – the element keeps its original color

→ all such elements have the same color.

ByBlock Color – the element inherits the color of its block

→ elements of different blocks can have different color.

ByLayer Color – the element inherits the color of its layer.

General Layer – the element remains on its original layer

→ the block definition refers to the layer

(hence the layer cannot be purged from the drawing),

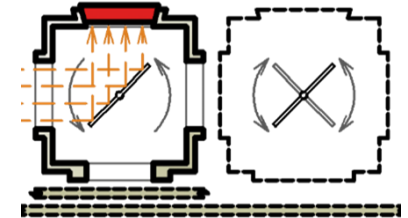
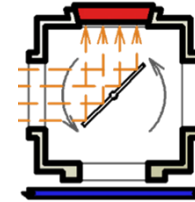
→ the element disappears when *its original layer* is turned off.

0 Layer – the element is placed on the layer of its block

→ each block instance can be on a different layer

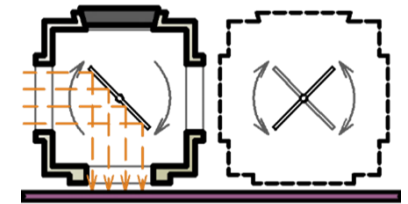
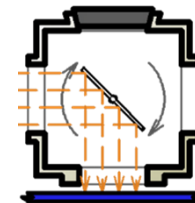
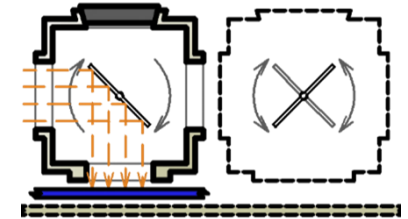
→ the element disappears when the *layer of its block* is turned off.

Given Color >



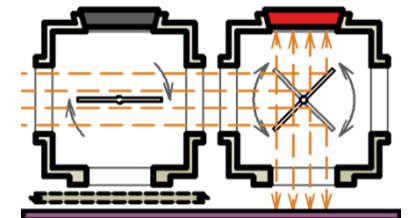
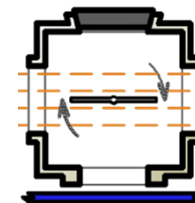
Bylayer Color >

General Layer >



0 Layer >

Block Color >



Block Behaviour

An element's color...

if the original had

and it was on layer 0:

it was on another layer:

<i>ANY Color</i>	<i>ByBlock color</i>	<i>ByLayer color</i>	
<i>A</i>	<i>B</i>	<i>insertion layer</i>	color,
<i>A (no!)</i>	<i>B (NO!)</i>	<i>original layer</i>	color.

→ Elements with different properties can be combined into a block with complex behavior (e.g. in a chair block the textile can have **ByBlock**, the legs **ByLayer**, the castor grey color).

→ The properties of the elements included in the block can only be changed by redefining the block.

Blocks / Layers

Block elements on layer 0 – ByBlock properties:

- the block behaves like a normal drawing element (it can be moved from one layer to another, and it can have varying colors or linetype).

Block elements on a separate layers:

- the drawing can be made more structured but requires more care.

Block elements on a multiple layers:

- one instance can have several colors (lineweights, linetypes),
- the block can be displayed in different representations (e.g. the 2D symbol in one viewport, the 3D model in the other)

Block elements on layer 0 (n×8):

```
1_DOOR
1_GLAZ
1_EQPM
1_FURN
1_WALL
1_FILL
1_FLOR
1_DIMS
2_DOOR
...
```

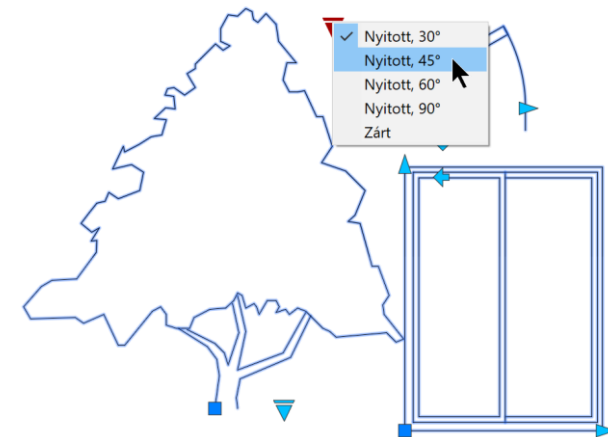
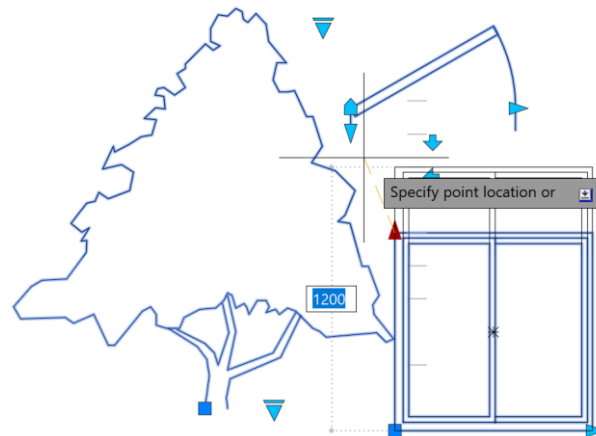
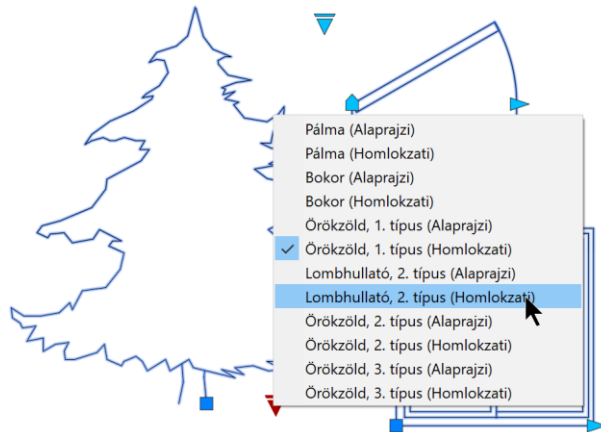
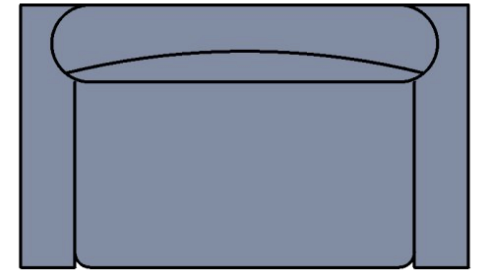
Elements on separate layers (4+n×4):

```
-DOOR
-GLAZ
-EQPM
-FURN
1_WALL
1_FILL
1_FLOR
1_DIMS
2_WALL
...
```

Dynamic Blocks / 5D Blocks

Multiple contents in a single block definition:

- can be used to display different versions (e.g. different types of trees, cars...),
- can be used to manage size variations (e.g. different window sizes),
- can be used to display different states (e.g. door opening angle),
- can contain the 2D symbol and the 3D model of an object.



Blocks With Attributes

Blocks can contain numerical and/or textual data in addition to or instead of drawing information (e.g. the name, covering and area of the room).

Attribute definitions can be included in the block (creation: [AttDef](#)):

- when inserted, they are either use default values, or request data,
- the information stored this way can be collected and saved into a file.

Kitchen
ceramic
6 m²

Text Style

- New styles (font, style, size) can be created ([Style](#))
 - existing styles can be used ([DText](#)).

Dimension Style

- The style of dimensions can be defined and modified ([DimStyle](#)).

Exporting / Properties

Drawing elements can be copied between drawings – but the pasted elements don't always appear the same.

→ Every element refers e.g. to its layer and linetype, some to text and/or dimension style, block definition.

→ General rule for these properties:

→ if an element (layer, linetype, style) with the same name exists in the target file, it is used unchanged,

→ if it doesn't exist, it is created automatically, as defined in the source file.

If a `DiningTable` block and its dashed dimensions are pasted into a drawing (`Ctrl+C / Ctrl+V`)...

→ if a block definition with the same name exists, that will be inserted instead,

→ if not, but a `Chair` block definition exists, then the new `DiningTable` block will use that,

→ if a `Furniture` layer exists, all elements of the block with `ByLayer` color will inherit its color,

→ if a `Dashed` linetype definition exists, the dimensions may appear differently from the original,

→ if a `Dim` dimension style exists, the dimensions will be displayed according to that (e.g. with different size),

→ if not, but a `Dim` text style exists, the dimension text is displayed according to that (e.g. using different font).

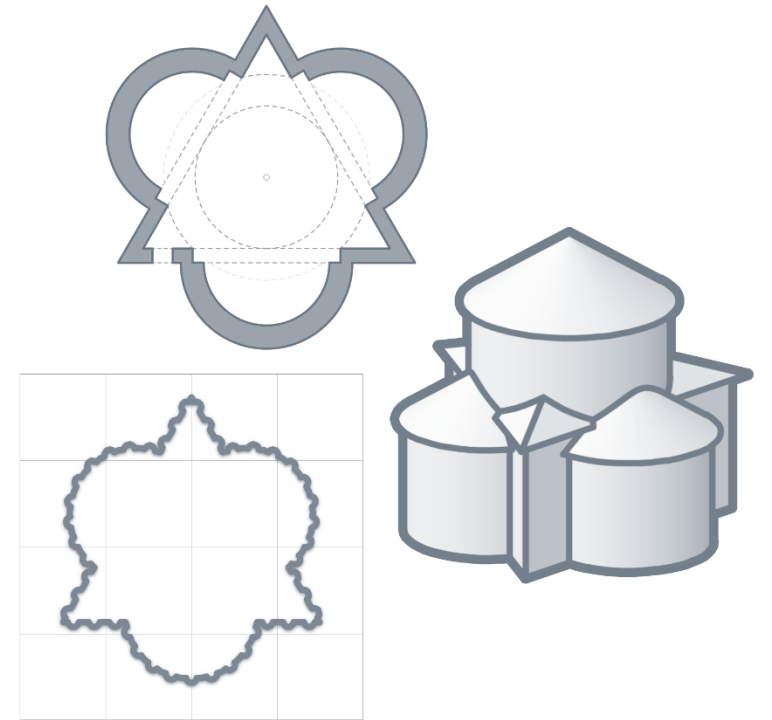
CAD

> 3D modelling

Digital Representation

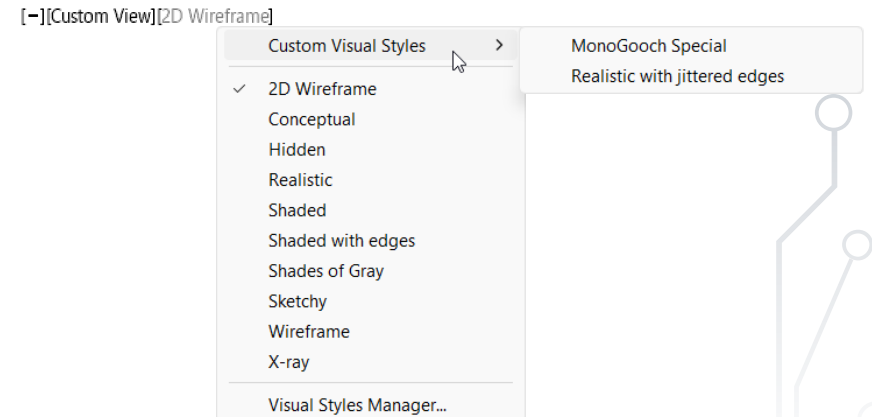
BMEEPAGA205

László Strommer PhD.



View - Visual Styles

- 2D Wireframe 2D projection is performed by the program, the video card displays 2D vectors (alias) faster, but often less reliable display (depends on video card)
- Wireframe overwrite hidden edges with background color
- Hidden warm-cold color transition instead of light-dark
- Conceptual display materials, textures (transparency).
- Realistic transparent surfaces (all edges visible)
- X-Ray schematic display (e.g. line extension, edge vibration)
- Sketchy



View - Projection Types

Any viewing angle can be used – either in parallel or perspective projection mode.

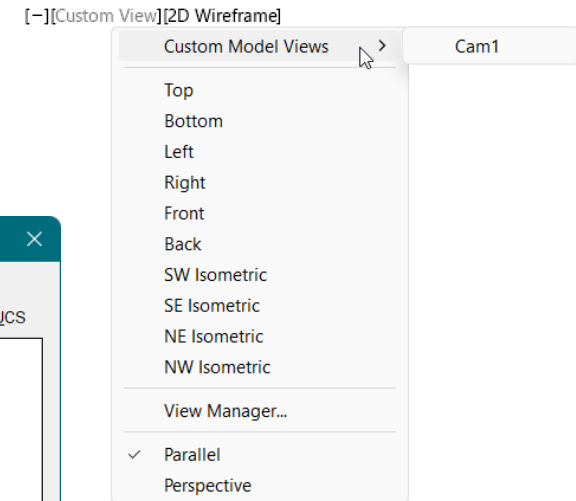
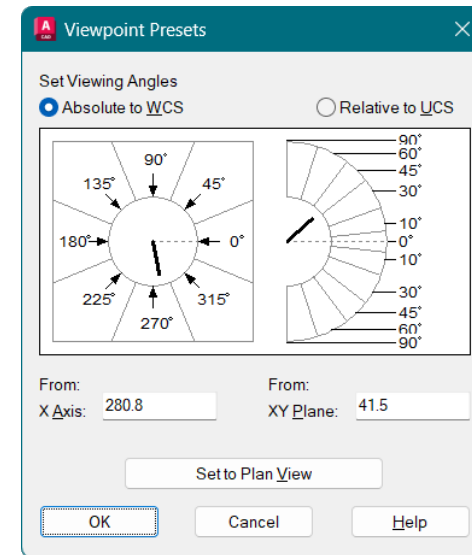
→ Using a general viewing angle helps to avoid overlapping lines.

→ Free movement and rotation using the mouse: [3dOrbit](#).

→ 2D angle in the base xy-plane + 3D angle from the xy-plane: [ddVPoint](#). >

→ Exact 3D position: [Camera](#):
viewpoint +target point +properties
(e.g. viewing angle / zoom)

→ Alternative that also works on Mac: [Dview](#):
Points +Distance +Zoom.



View - Viewports

Viewports can also be used in 2D (e.g. floor plan variations, multiple nodes, detailed drawing...).

Their use in 3D is almost inevitable (e.g. for views of the same model)...

- alignment of views is important ([Zoom Center](#)),
- the dimensions shouldn't be seen in any other viewports than where they belongs.

Model space dimensions – the size of the dimension depends on the printing scale ([DimScale](#)) ...

- separate layer for each view (frozen in the other viewport),
 - it is worth using the [frozen in new viewports](#) layer setting,
 - in existing viewports, the display must be disabled one by one.

Paper space dimensions – it is enough to use one size ([DimScale](#)), but the scale of the view becomes important

- scale the paper space distances ([DimLFac](#)) e.g. meter in 1:200 → 1 m = 5 mm → [DimLFac](#) = 20
- connect dimensions to drawing elements e.g. showing meters in cm → [DimLFac](#) = 100
- enable associative dimensioning → [DimAssoc](#) = 2 ([Options](#))

3D Coordinate System

Cartesian x, y, z

→ x, y, z distances from the starting point
(from the current origin or from the last point)

Cylindrical $d < a, z$

→ the projection of the distance in the xy plane,

→ the angle from the x -axis,

→ and the z coordinate.

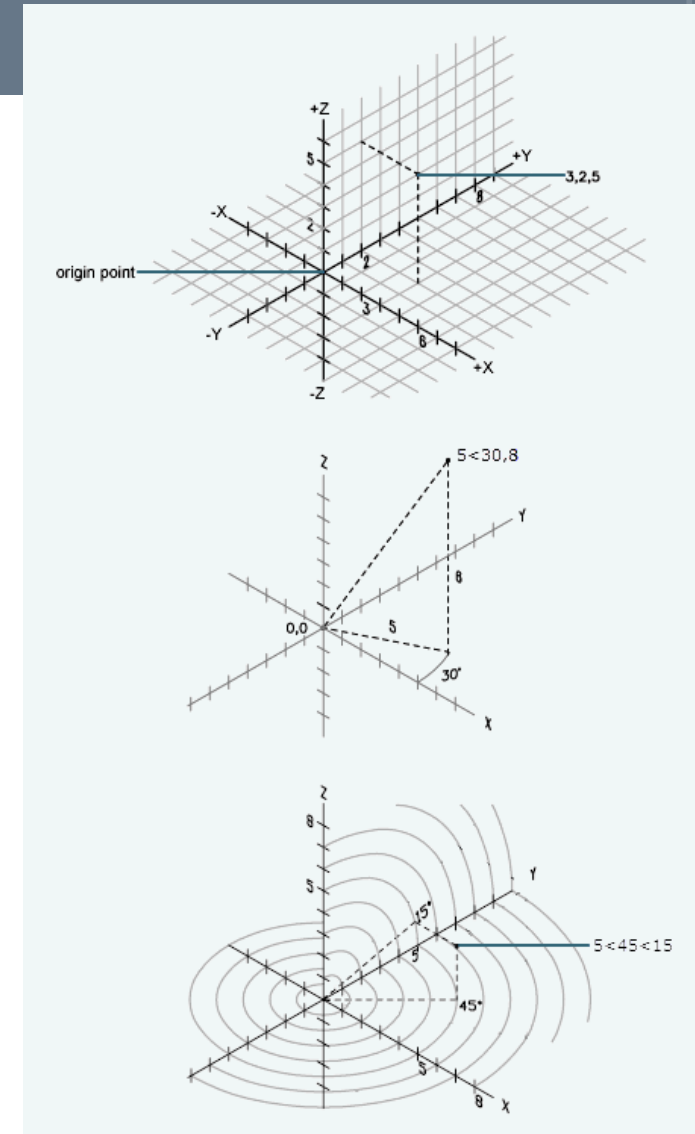
Spherical $d < a < b$

→ distance from the starting point,

→ the angle from the x -axis,

→ the angle from the xy -plane.

Accuracy check: **ID** and **Dist**



3D Points

3D point entry can often be omitted ...

- by selecting a point of an existing element using **OSnap**,
- by using filters (e.g. `.x`, `.xy`), combining the coordinates of several elements (and/or coordinate entry),
- by aligning the UCS so that 2D (rectangular or polar) coordinates could be used.

If there are different coordinate systems in the viewports,
the coordinate system of the active window will be valid during its execution.

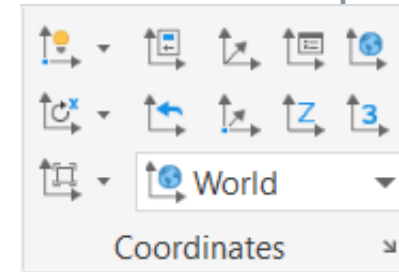
UCS

Every drawing has a **World Coordinate System, WCS**.

Any number of **User Coordinate Systems UCS**, can be defined.

A new UCS can be defined in many ways:

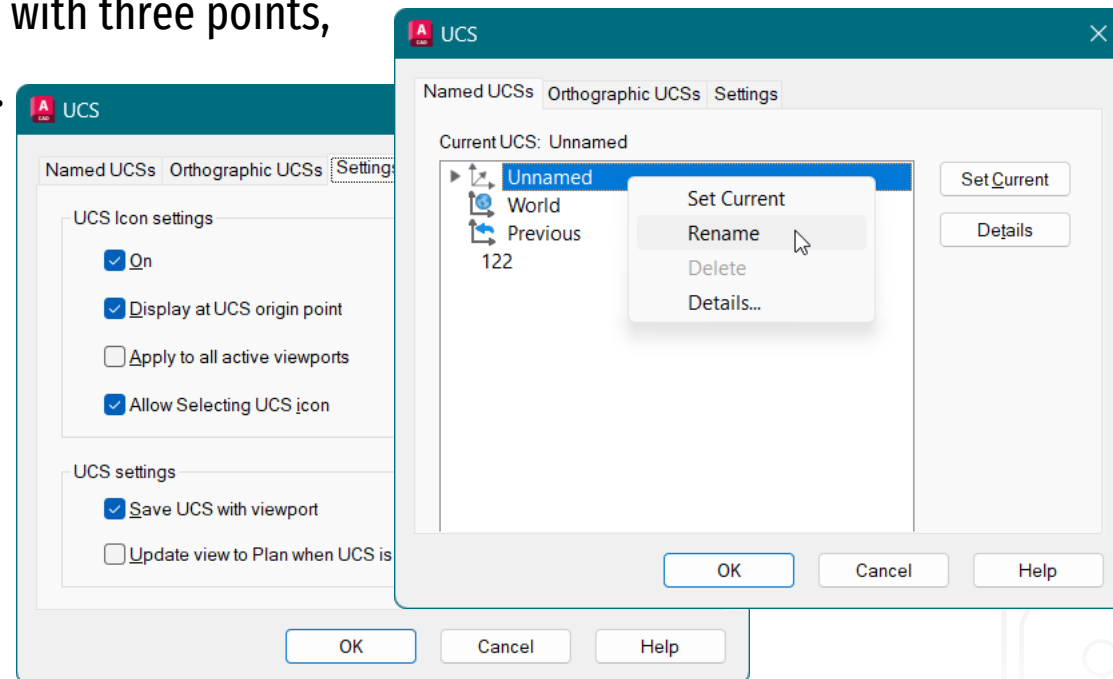
- can be generated by rotating the current UCS around the x, y, or z axis,
- a new origin, z-axis can be specified, or it can be defined with three points,
- can be aligned with the viewing direction or to an object.



The active UCS can be saved (**UCSMAN**). ▶

The value of the **UCSVP** variable:

- 1 → every viewport can have its own UCS,
- 0 → the UCS of the current viewport applies to all.



Coordinate System · Elements

The objects behave differently in 3D.

2D Objects

- e.g. [PLine](#), [Arc](#), [Circle](#), [Ellipse](#)
- their plane is always parallel to the xy plane of the current UCS
- their height is determined by their first point, so the z-coordinates of the other points can be omitted.

2D-3D Objects

- e.g. [Line](#), [3dFace](#)
- the height of their points are arbitrary.

Coordinate System · Editing

The modifying commands behave differently in 3D.

2D Commands

→ e.g. [ROtate](#), [MIRROR](#), [TRIM](#), [EXTEND](#)



→ their working plane is always parallel to the xy plane of the current UCS, i.e. the rotation axis is parallel to the z-axis, so it is enough to specify one of its points.

3D Commands

→ e.g. [MIRROR3D](#), [ROTATE3D](#)



→ they doesn't have a fixed working plane, which can be good (no need to change the UCS) but it can be unnecessary, because they require extra data (time / errors).

Dimension-independent Commands

→ e.g. [COPY](#), [MOVE](#), [STRETCH](#)



→ by entering two coordinates they work in 2D, by entering the third coordinate they become 3D.

Modell Types

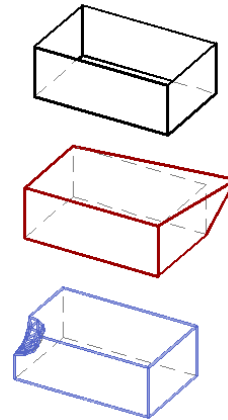
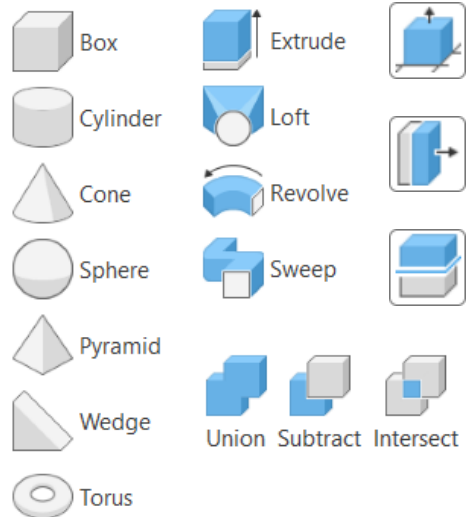
2,5D model · thickness

3D surface model

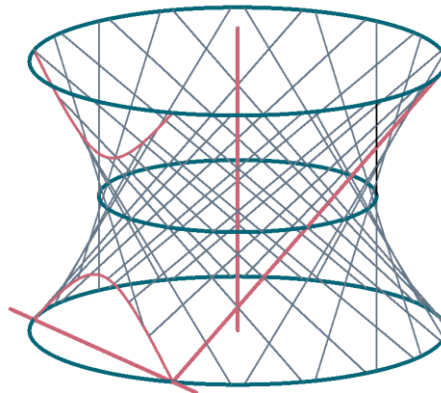
3D solid model

3D primitives or 2D + Extrude / Revolve

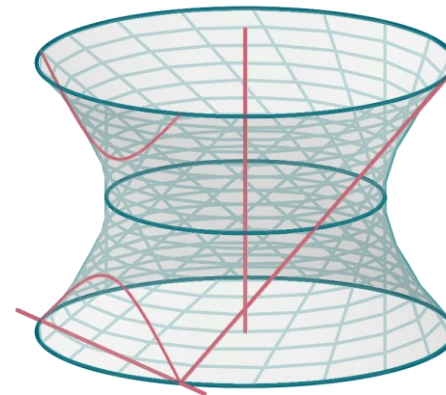
+Boole operations or e.g. Slice



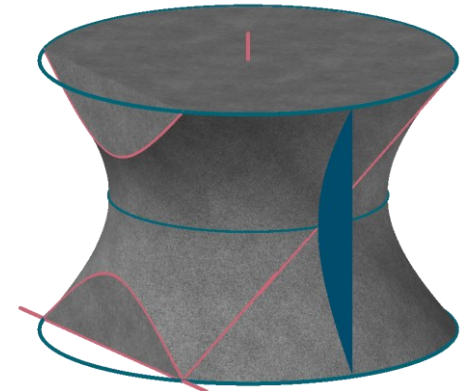
Line
& Array



Line
& RevSurf

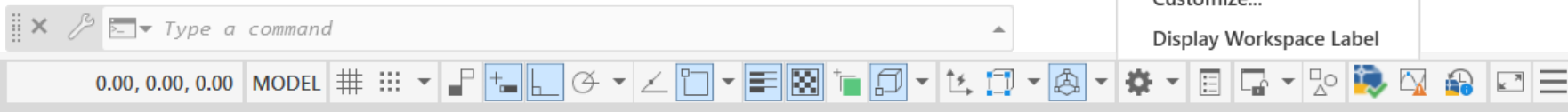


Region
& Revolve

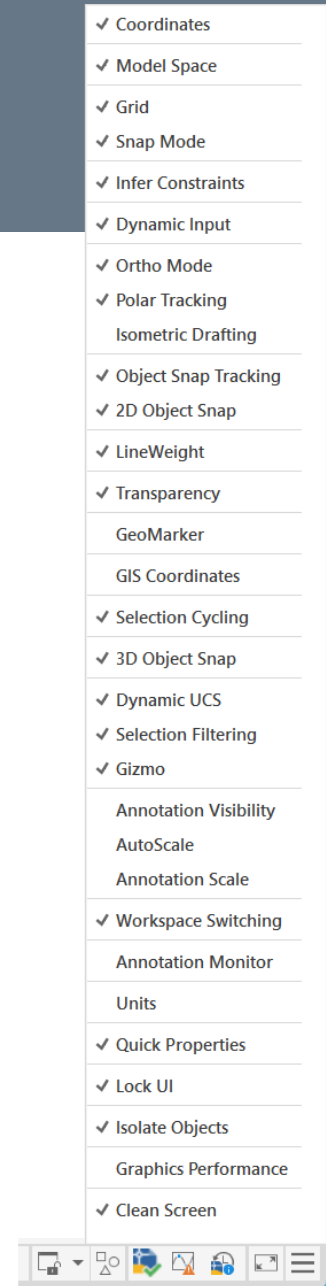


3D Workspace

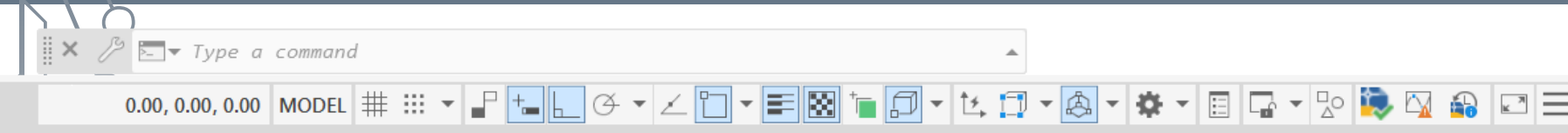
By changing the Workspace ►
the ribbon menus and palettes can be modified.
→ Personalized settings can be saved.



By default, the 3D drawing aids are not visible in the status line,
it is advisable to turn them on in order to see (and toggle) their status. ►



3D Drawing Aids



It is important to turn on some 3D drawing aids – and turn off others.

Dynamic UCS → automatic adjustment of the *xy* plane of the coordinate system to a 3D reference plane, e.g. to a given face of a solid F6 • *UCSDetect* 0-1

3D Object Snap → Finding characteristic points of 3D objects (3D drawing elements and scanned point clouds). F4 • *3DOSMode* 10

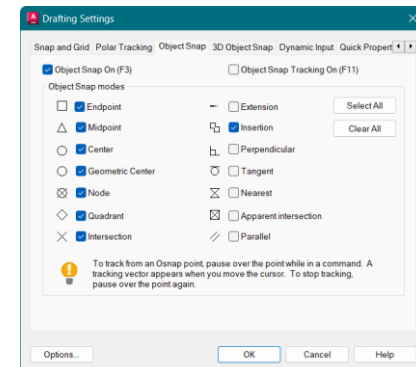
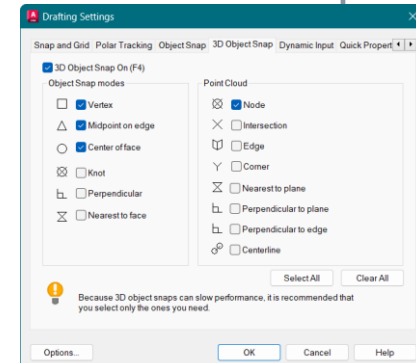
Ortho → restriction of vectors to axial directions F8 • *OrthoMode*

Show / hide lineweights (real width only in the layout) • *LwDisplay* ON

Display the transparency of elements (on / off) • *TransparencyDisplay* ON

Select overlapping elements from a pop-up list `Ctrl+W` • *SelectionCycling* OFF

Selected elements can be temporarily hidden or isolated: *HideObjects* / *IsolateObjects* / *UnisolateObjects*



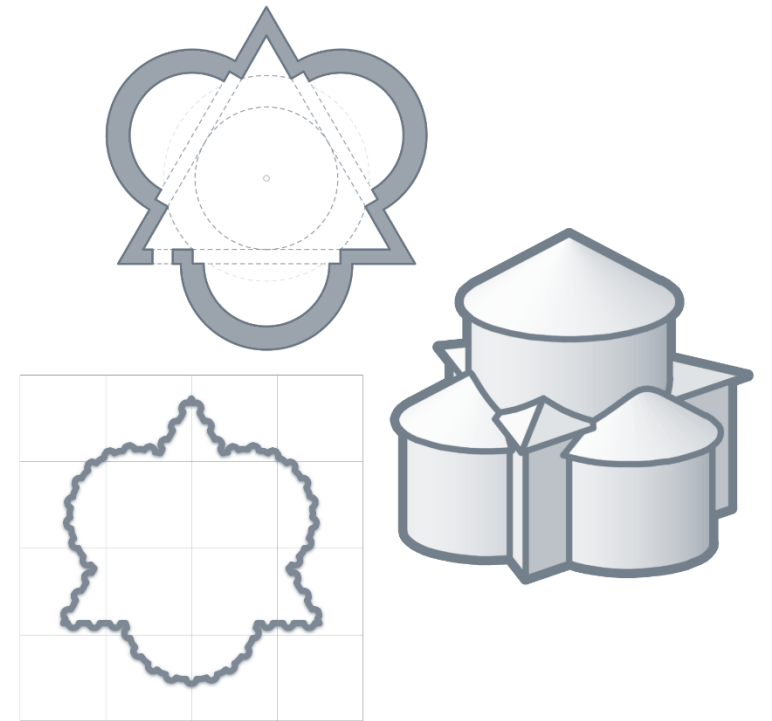
CAD

➤ Basics of Pixel Graphics

Digital Representation

BMEEPAGA205

László Strommer PhD.



Pixel / Vector

The computer displays a raster image – the drawing elements can be stored in vector and/or pixel format.

Pixel format

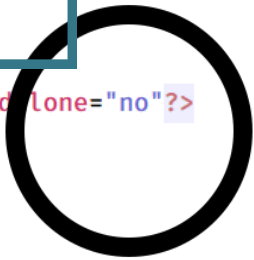
- Drawing elements can be separated using layers.
- Modification usually results in loss of information (resizing, random rotation... – or sharpening).

Vector format

- Drawing elements are geometric objects (e.g. circle, polygon, spline).
- Geometric transformations do not result in quality loss.

Mixed format

- Pixel gradients, backgrounds, patterns, etc., but e.g. vector texts, shapes.



```
1 <?xml version="1.0" encoding="UTF-8" standalone="no"?>
2 <svg
3   width="256.0px" height="256.0px"
4   viewBox="0 0 256.0 256.0"
5   version="1.1"
6   id="SVGRoot"
7   xmlns="http://www.w3.org/2000/svg"
8 <g
9   id="layer1">
10  <circle
11    style="fill:none; stroke:#000000; stroke-width:16;
12    stroke-linecap:round; stroke-linejoin:round"
13    id="circle"
14    cx="128"
15    cy="128"
16    r="96" />
16 </g>
17 </svg>
```

Pixel

The images stored and displayed by the computer consist of discrete picture elements – pixels.

Pixel

- Brightness and color information for a given coordinate of the image.
- It is often approximated by a rectangle, but it has no defined shape.
- *Pixel color* → typically by mixing primary colors. ►



Resolution

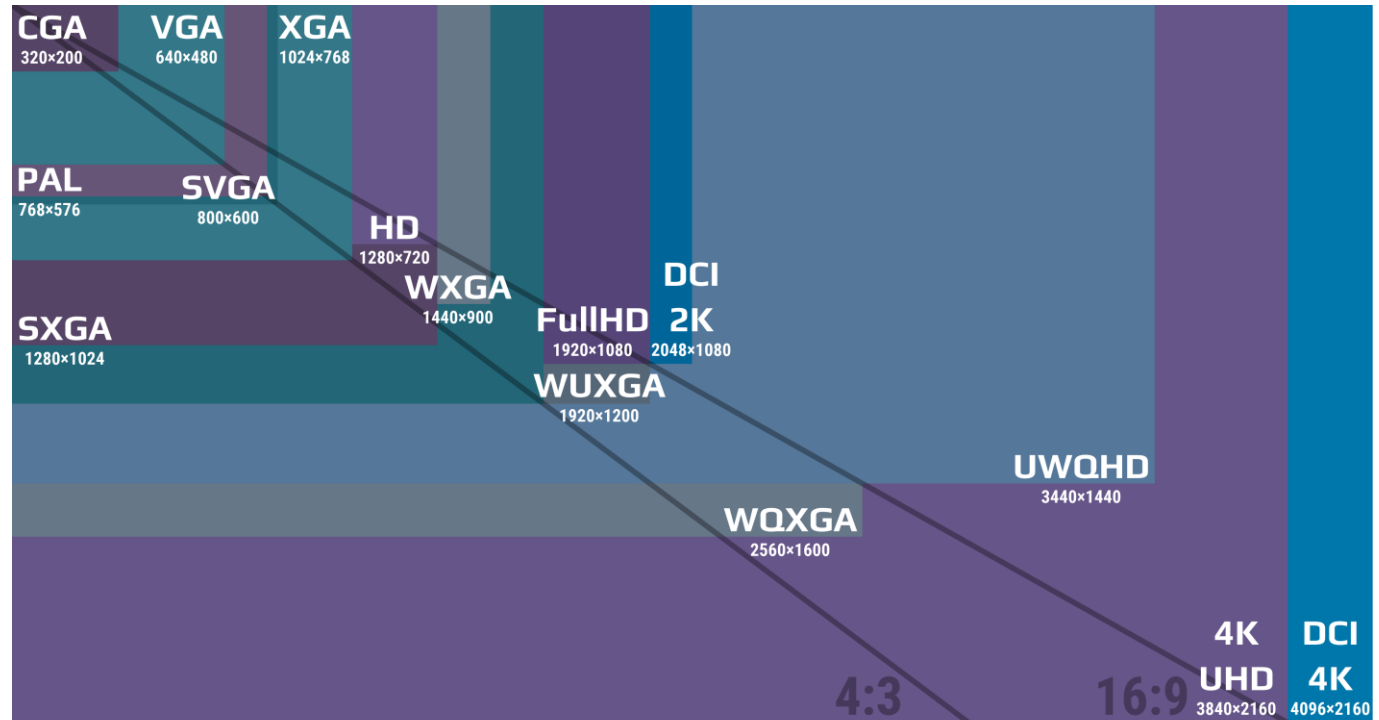
Resolution can characterize the pixel number or pixel density.

Image resolution / dimension

- The number of pixels, as a number (x×y, e.g. 800×600), with a name (e.g. SVGA), or approximately (e.g. 16MP camera).

Pixel density

- The distance between the pixels.
- If it is the same in the x and y directions, the number of pixels determines the aspect ratio.
- Typical resolution (dot/inch):
monitor 72–96 dpi / printer 300-1200 dpi.



Color Depth

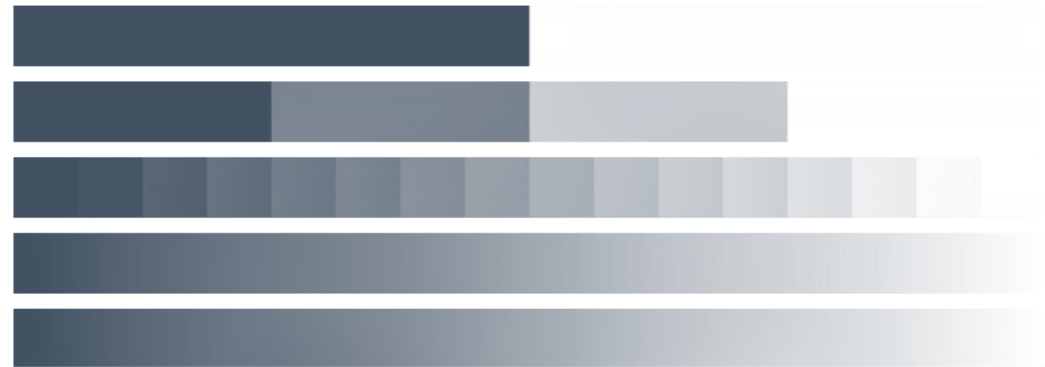
Color depth (bits/pixel) is the sampling accuracy of colors.

Typical color depth values:

- 1 bit → $2^1 = 2$ colors (monochrome)
- 2 bits → $2^2 = 4$ colors
- 4 bits → $2^4 = 16$ colors or shades (two pixels / byte)
- 8 bits → $2^8 = 256$ colors or shades (e.g. greyscale)
- 24 bits → $(2^8)^3 = 16.7$ million colors (256 shades / RGB)
- 32 bits → 16.7 million colors
& 256 steps of transparency (alpha channel)

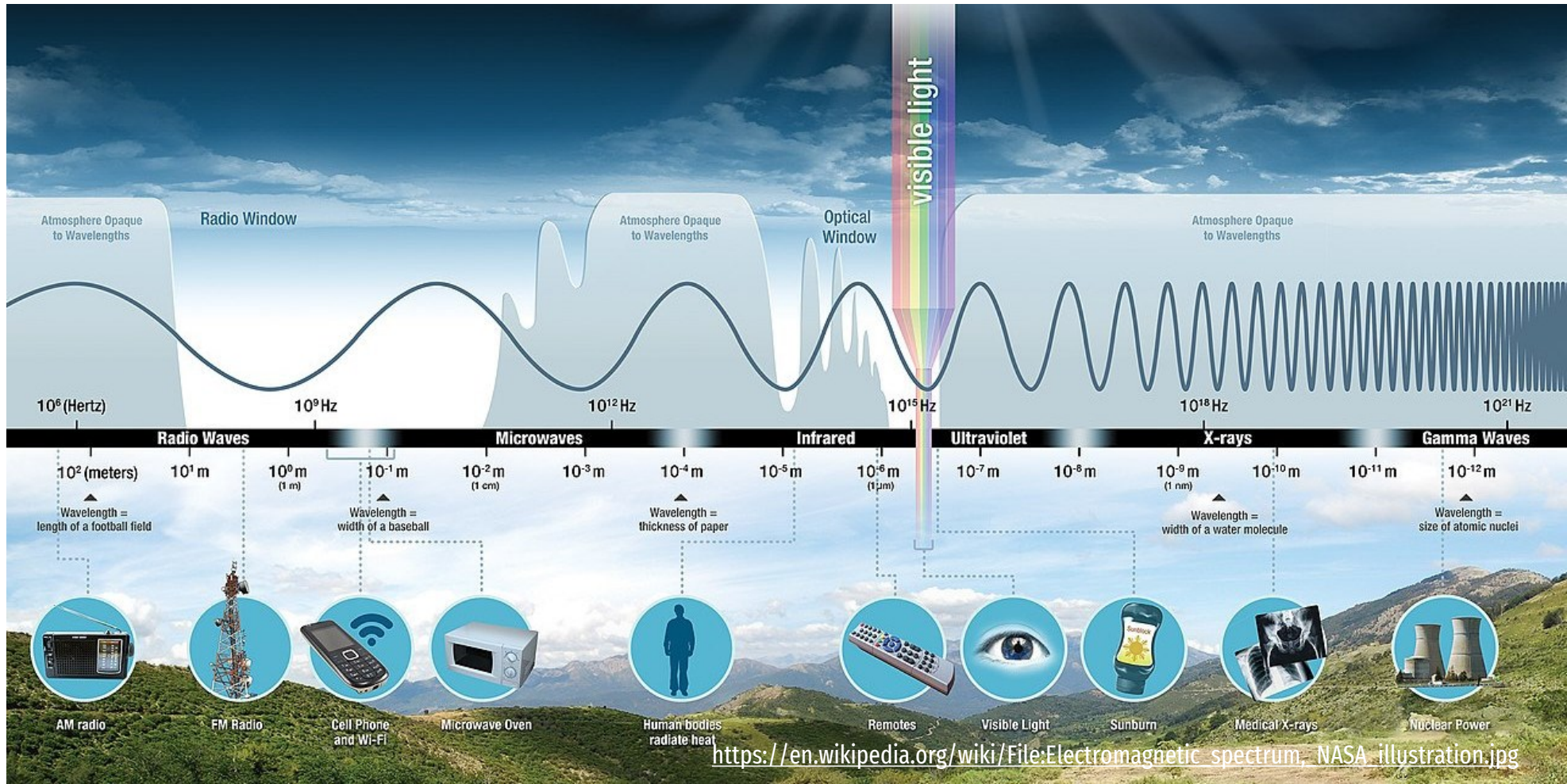
HDR · High-Definition Range

- 30 bits → $(2^{10})^3 = 1$ billion colors (1024 shades / RGB)
- 36 bits → $(2^{12})^3 = 68.7$ billion colors (4096 shades / RGB)...



Colors

Light visible to the human eye is the wavelength range of ~380 to 780 nm in the electromagnetic spectrum.



Spectrum

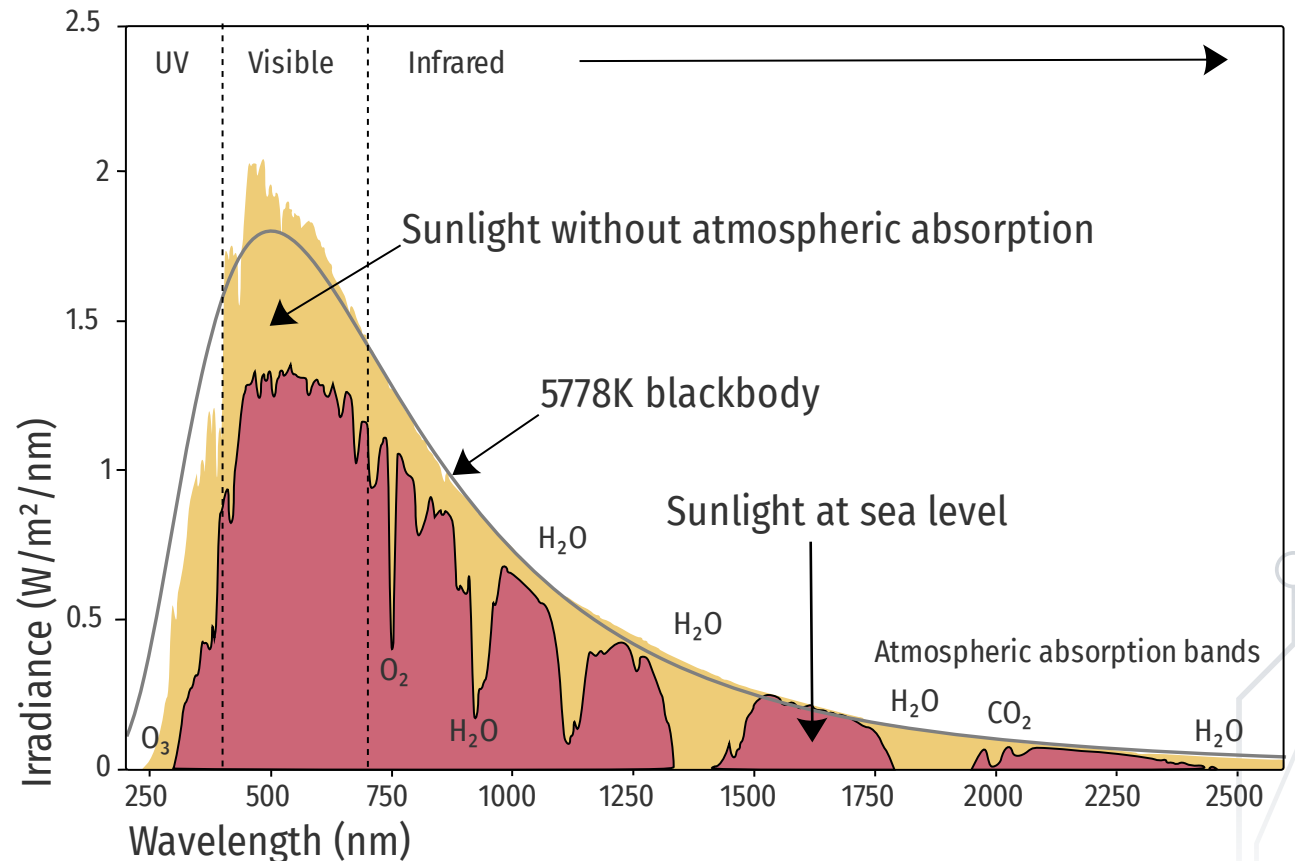
The spectral power distribution function of the Sun's radiation energy shows the amount of energy for uniform wavelength intervals (with atmospheric absorption and without it).

The Φ light falling on a material is either transmitted (τ), absorbed (α), or reflected (ρ).

$$\tau(\lambda) + \alpha(\lambda) + \rho(\lambda) = 1$$

The source of the color stimulus can be a light source (light-color), or a light reflected from a surface (surface-color).

Spectrum of Solar Radiation (Earth)



Color Sense

The visual receptors of the human eye:

- light-sensitive rods → “night vision”
- three types of color-sensitive cones
 - for long (L), medium (M), short (S) wavelengths
 - can distinguish 100-120 levels of brightness.

Brightness and Color Constancy

- regardless of the changing light conditions, the sheet of paper is perceived as equally bright and the taxi is perceived yellow.

Metamerism

- spectrally different colors can result in the same color perception, if the stimuli for the three receptors are identical.

Characteristics of the human color perception.

Lightness

- the relative intensity of a colored light.

Saturation

- how much white and gray color the light contains.

Hue / characteristic wavelength

- the wavelength of the monochromatic light that appears to have the same color. Helmholtz

Light characterized by a given spectral distribution

- *appears red (green, blue) if it mostly contains that given range of the spectrum,*
- *appears white, gray, or black if it stimulates the three receptors to the same extent.*

CIE Tristimulus Values

The CIE (International Commission on Lighting) color spaces were the first quantitative links between distributions of wavelengths in the visible spectrum, and the perceived colors.

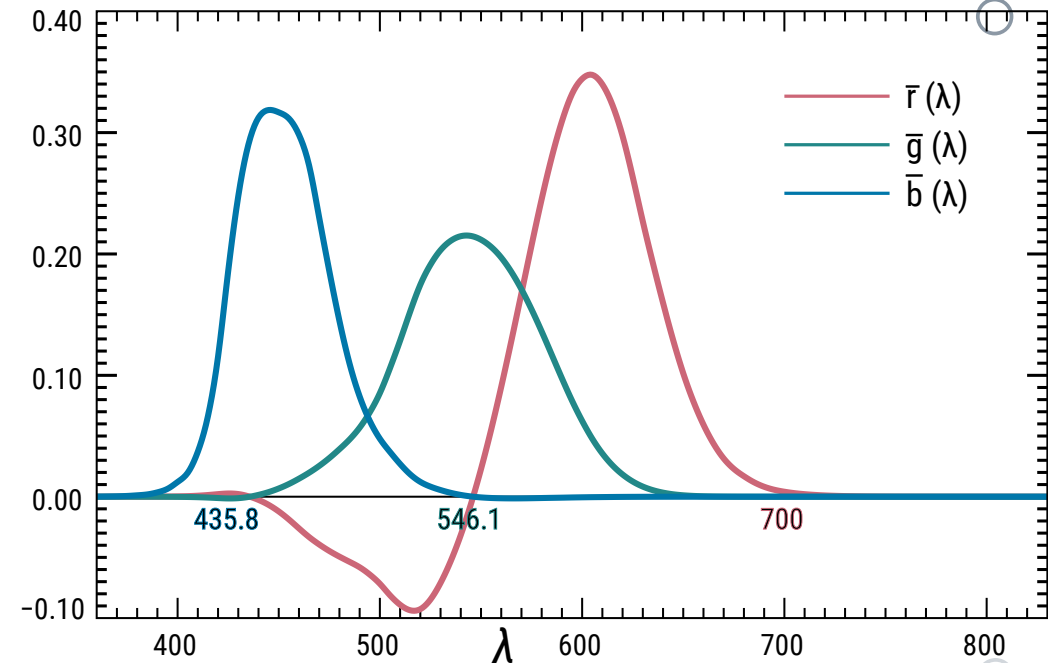
→ Selected red, green, blue basic colors:

$\lambda_{(R)} = 700 \text{ nm}$, $\lambda_{(G)} = 546.1 \text{ nm}$, $\lambda_{(B)} = 435.8 \text{ nm}$.

→ Experimentally established primary color units, whose sum appears to be a white light of equal energy:

$R = 0.17697 \text{ lm}$ | $G = 0.81240 \text{ lm}$ | $B = 0.01063 \text{ lm}$.

→ They tried to determine the ratio of the primary colors that results in the same color sensation as monochromatic light of a given wavelength – but one of the primary colors always turned up negative (had to be added to the target).



→ Other spectral colors cannot be reproduced by mixing the three primary spectral colors.

CIE xy Chromaticity Diagram

By placing the chosen primary colors on the corners of a triangle

- colors that can be mixed from them are located inside,
- colors with negative values are located outside.

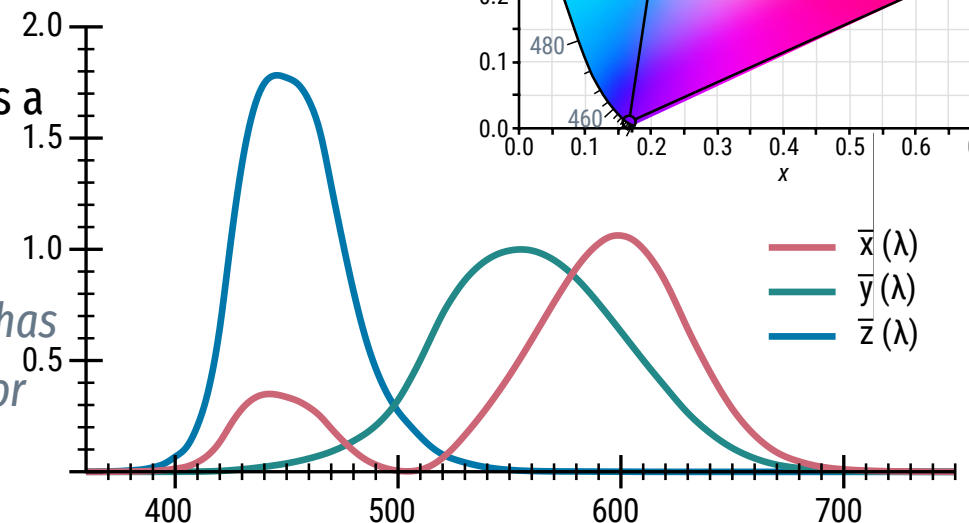
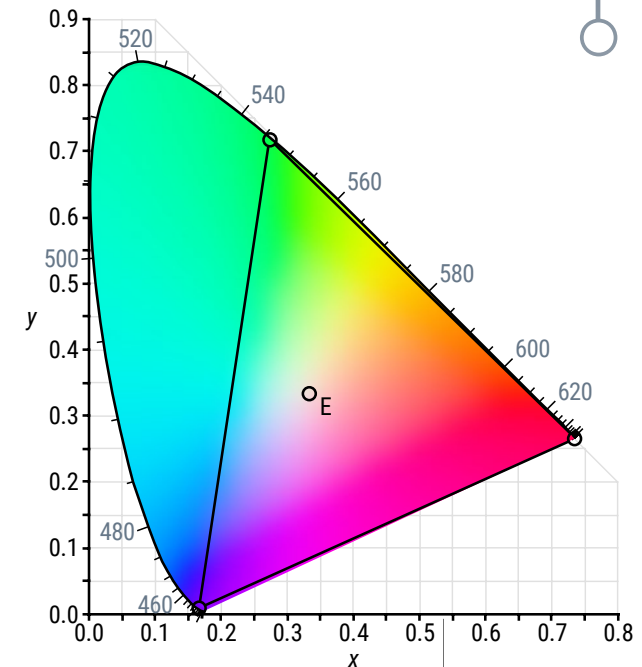
It was more expedient to use virtual primary colors X, Y, Z,

- resulting in three positive values for each real color,
- for this, the XYZ triangle must enclose the spectral locus (the pure hues of a single wavelength) and the purple line.

Each x, y color coordinate of the chromaticity diagram defines a color hue (z is not needed because $x + y + z = 1$),

- but the brightness factor Y must be specified.

Obviously, there are other color schemes, and the CIE system has also been refined several times (for example, to make the color space more uniform).



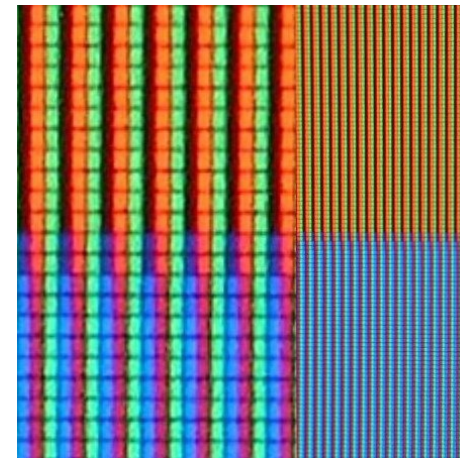
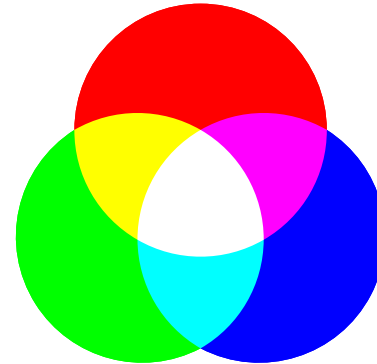
RGB Model

Additive color mixing takes place in the human eye:

- by projecting colored lights onto a reflective surface,
- by projecting different colors into the eye in rapid succession (with a frequency greater than 50 Hz)
- with densely placed small dots of the colors to be mixed (so that the eye cannot separate them).

RGB / Displays

- Additive mixing of red, green and blue primary colors.
- The value of the basic colors can vary between 0-1.



CMY / CMYK Model

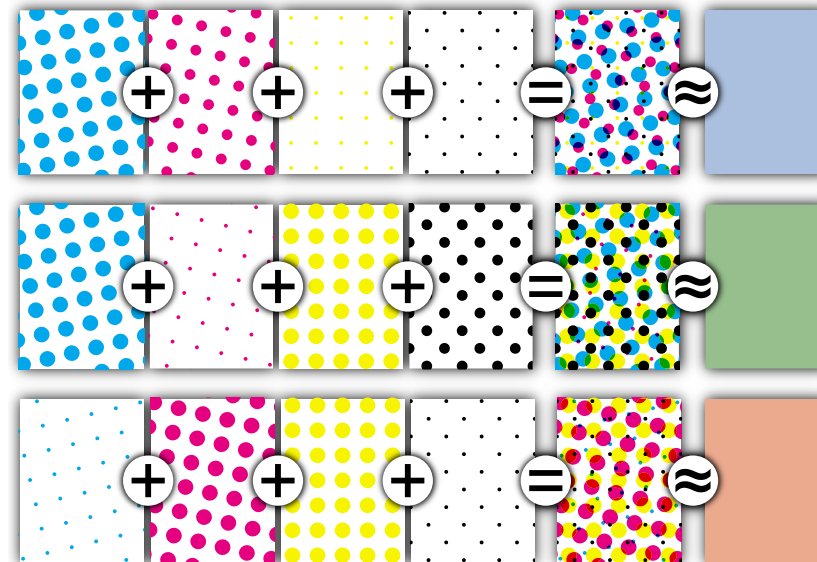
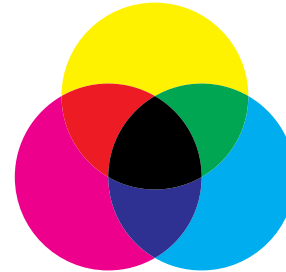
In the case of subtractive color mixing, the color is created jointly by the light source, the colored surfaces that reflect the light, and the color filters placed in the path of the light.

$$\varphi(\lambda) = \Phi(\lambda) \cdot \rho(\lambda) \cdot \tau(\lambda)$$

The color of the light is decided by the dominant spectral range in the distribution of the relative power of the light.

CMY-CMYK / Printing

- Subtractive mixing of cyan, magenta, yellow.
- The reflected light decides the color:
 - the colors of the points are given,
 - more saturated color → bigger point size,
 - to avoid overlapping → rotated grids.
- Black ink yields much better blacks.



YCbCr Model

A Y'CbCr color space can be determined by coordinate transformation of the primary colors and white point of an RGB color space.

- Y' the luma component, which can be stored and transmitted in high resolution,
- Cb and Cr are the blue-difference and red-difference color components, which can have reduced bandwidth for the sake of efficiency.

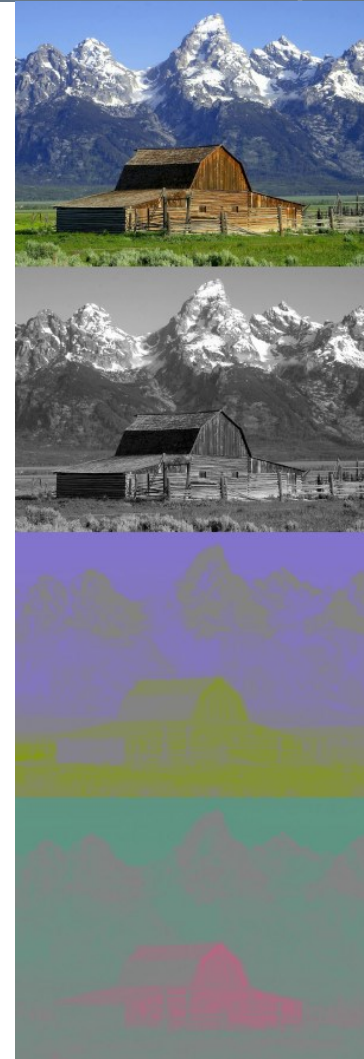
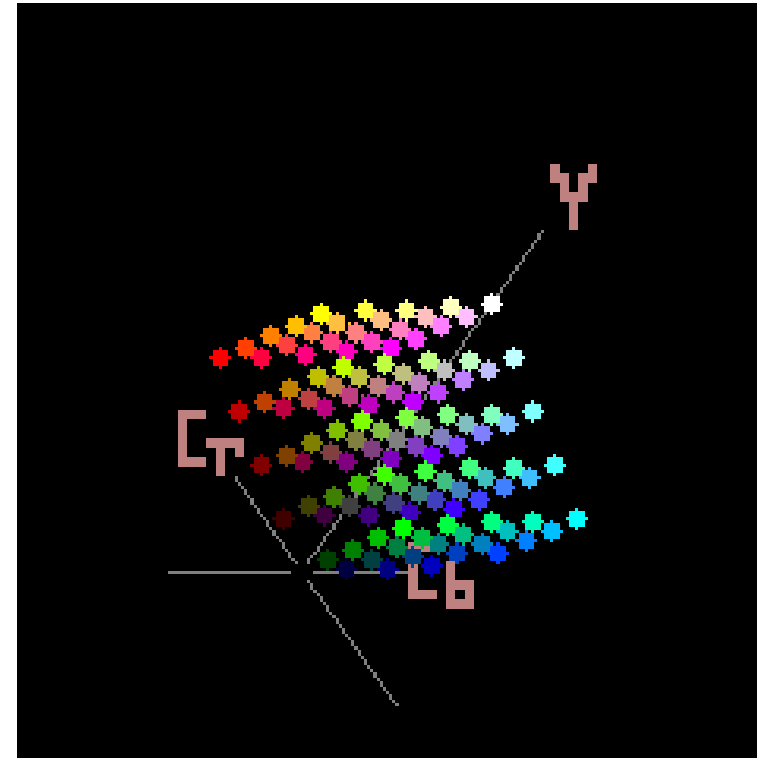
Chroma subsampling

The human visual system is less sensitive to color differences than to luminance,

- when encoding images, a lower resolution can be used for color than for luma information.

<https://en.wikipedia.org/wiki/YCbCr>

https://en.wikipedia.org/wiki/Chroma_subsampling



https://en.wikipedia.org/wiki/Gamma_correction

https://en.wikipedia.org/wiki/File:Barns_grand_tetons_YCbCr_separation.jpg

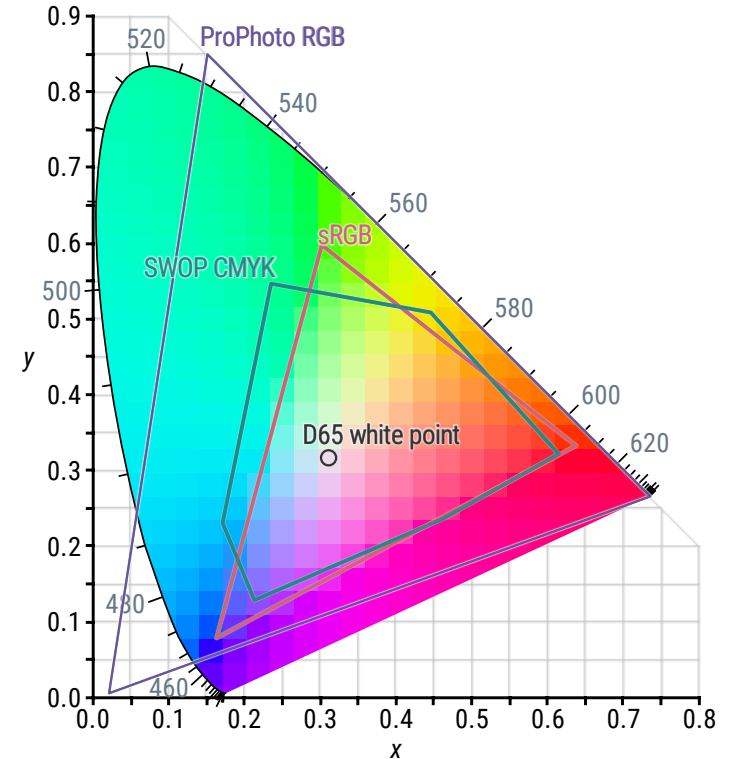
Gamut

The CIE xy chromacity diagram represents all the chromaticities visible to the average person. The colored region is called the *gamut* of human vision.

- Gamut is the set of colors that can be accurately represented by a device or a color space.
- The gamut of the printer (CMYK) and the monitor (RGB) do not match – there are colors that can be reproduced by one, but not by the other (vica versa).
- Devices obviously have to use real primary colors, so their gamut is always incomplete.
- A color system, can contain fictitious colors that humans cannot see (in order to increase the gamut).

If the M cone alone could be stimulated, the brain would see a color that is greener than what is physically possible.

<https://en.wikipedia.org/wiki/Gamut>



https://en.wikipedia.org/wiki/Impossible_color

https://en.wikipedia.org/wiki/File:CIE1931xy_gamut_comparison.svg

Image Formats

BMP

- uncompressed format
- large file size, low computing requirements

GIF

- lossless compression, but max. 256 colors
- animation (poor efficiency and quality)

PNG

- lossless compression → computing
- truecolor or palette, +transparency (1-8 bit)

JPG

- lossy compression → quality / loss of fidelity
- photos, rendered images

WEBP

- all-in-one solution, more efficient compression
- lossy and lossless, transparency, animation

ICO

- images with multiple resolutions and color

TIFF

- flexible professional format
- several color modes, 1-64bit, more images

PSD / CPT / ACDC ...

- layers, masks (adjustment layers)

RAW / DGN

- native format of digital cameras
- images with greater color depth (HDR)

Specialities

Multiple images in one file

- thumbnail (e.g. JPG)
- alpha channel (e.g. PNG)
- combinations of images, masks (e.g. PSD)
- multi-page (e.g. TIFF)
- multi-image (e.g. ICL)
- animations (e.g. GIF)

Palette

- defining palette colors
(e.g. 0=FFFFFF 1=000000 2=990033 3=0066CC)
- image color by the palette (2 bits instead of 3×8)
- compression

Metadata / [EXIF](#)

- Machine > Make, Model, Software
- Photo > Aperture, Shutter, ISO, Focal Length
- Image > Color Profile, White Balance
- Color > Color Space (sRGB), Mode (YCbCr)
- GPS > Latitude, Longitude, Timestamp
- Thumbnail > Latitude, Longitude, Timestamp

Metadata / [IPTC](#)

- Location > Country, City, Location
- Info > Keywords, Credit, Contact

Metadata / [XMP sidecar](#)

- if the original is not suitable for storing metadata
(Extensible Metadata Platform)

CAD

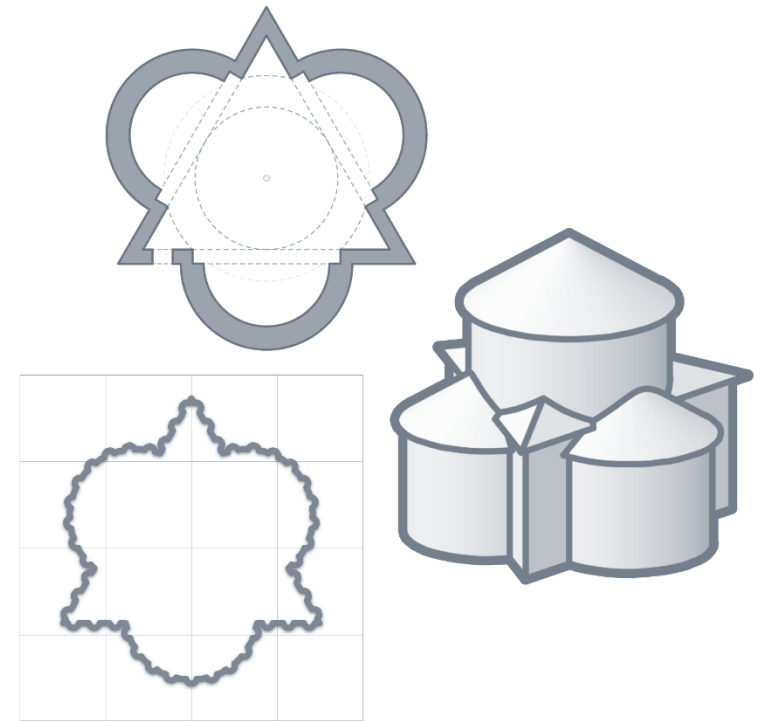
➤ Rendering

➤ Data Exchange

Digital Representation

BMEEPAGA205

László Strommer PhD.



Rendering

The level of realism of the visual styles can be adjusted depending on the hardware (e.g. transparency, shadows).

Rendering can produce an even more realistic image, but not in real time, but over several minutes:

- the quality (and time) can be controlled,
- the resolution of the output file can be set,
- lights, materials, background can be used (however, these can also appear in a realistic visual style).

Curved surfaces are approximated by a mesh with controllable density ($ViewRes \times FacetRes$). Surfaces whose normal points in the opposite direction can be excluded from the calculation.



Views

The aspect ratio of the paper space viewport decides the dimensions

→ using standard ratios is recommended (4:3, 16:9, etc.)..

The view can be specified and adjusted in several ways:

→ **3dOrbit**:

projection: **Parallel** / **Perspective**

→ **ddVPpoint**:

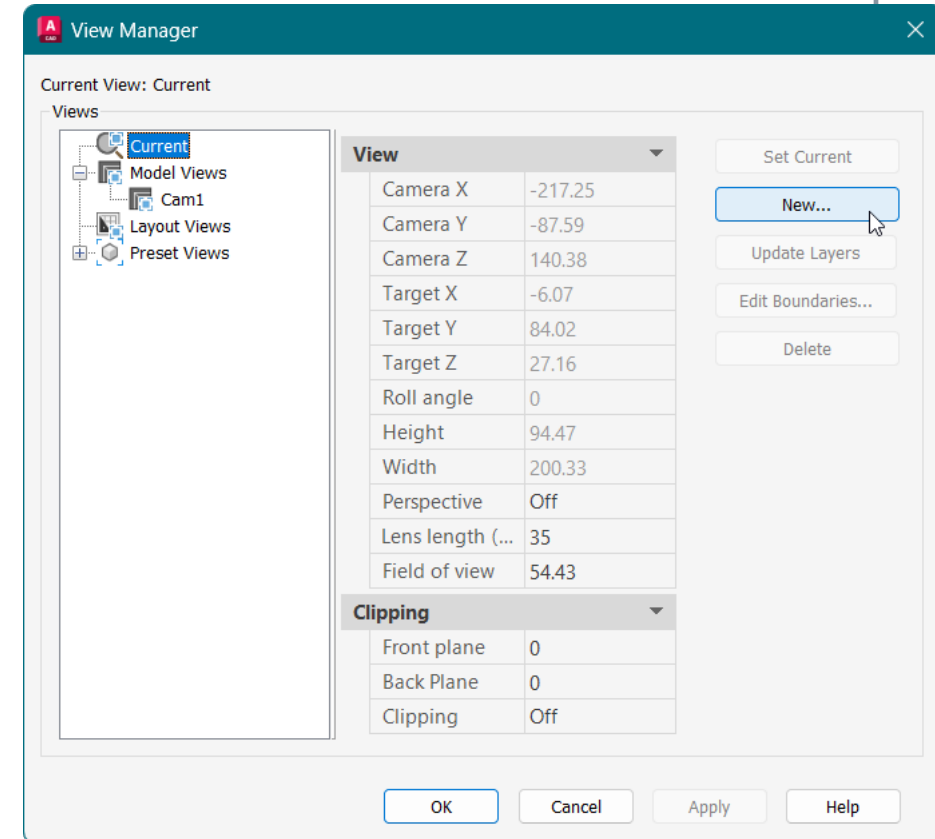
directions in the base plane + from the base plane

→ **Camera**:

camera +target +properties (e.g. zoom)

→ **DView** (an alternative that also works on Macs):

Points (target +camera) +**Distance** +**Zoom**



It is recommended to save the view with the **View** command (*this is automatic for the camera*).▲

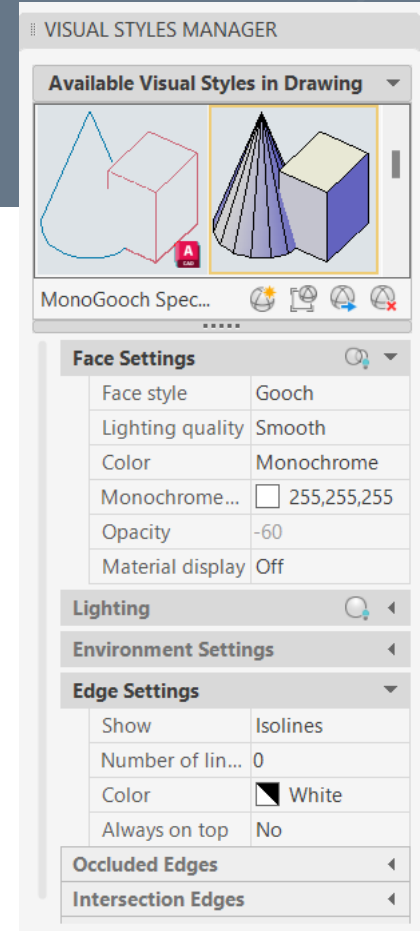
Visual Styles

Visual Styles

- 2D Wireframe the projection on the plane is performed by the program (antialiasing)
- Wireframe the projection is done by the video card
- Hidden overwriting hidden edges with background color
- Conceptual warm-cold transition instead of light-dark
- Realistic displaying materials, textures (transparency)
- X-Ray transparent surfaces (all edges are visible)
- Sketchy e.g. line extension, edge vibration

Visual styles can be customized by changing their edge and surface settings. ►

Styles can be transferred between drawings using the [Tool Palette](#), first by exporting them to the active sheet of the palette, then adding them to the other drawing from there.



Traditional Lights

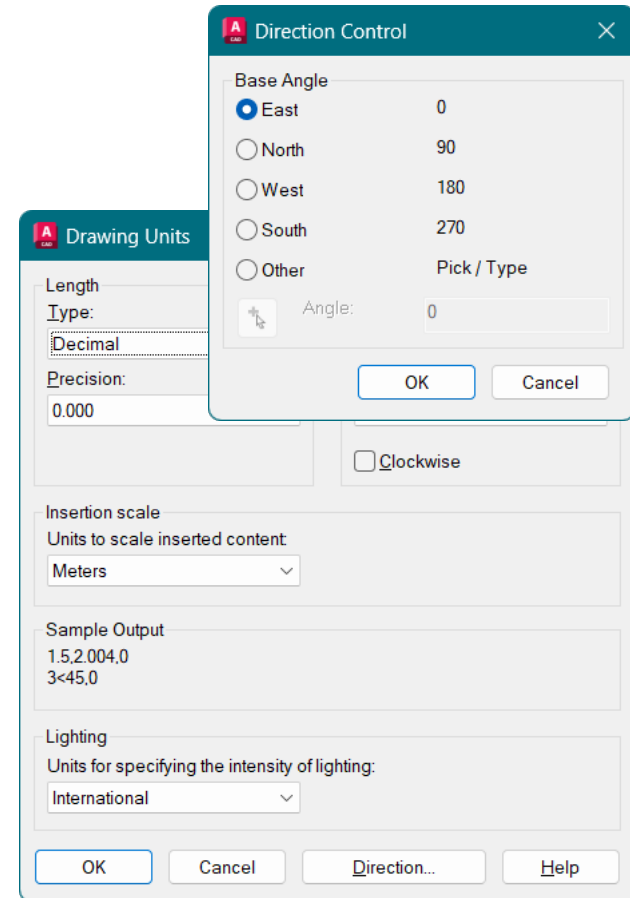
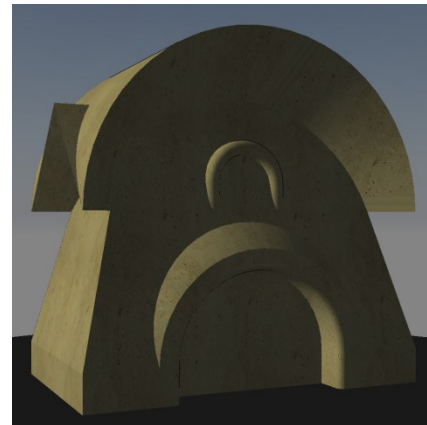
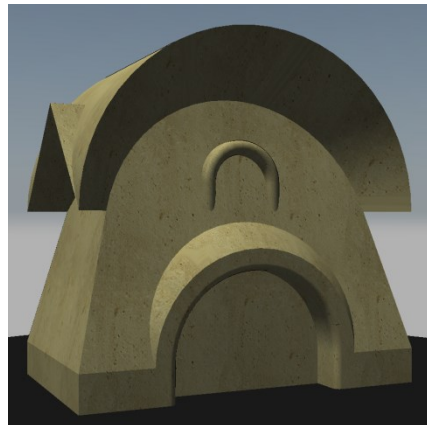
Several types of lights can be used...

- **Ambient** instead of calculating light scattered in the atmosphere and from surfaces, too much → blurred image | not enough → over-contrast lunar landscape
- **Distant** the power of a light source characterized only by its direction (and intensity) its power does not decrease with the distance between objects (e.g. sun)
- **Point** light source radiating equally in all directions (e.g. lamp) attenuation with distance can be applied
- **Spot** light source characterized by its position and direction (e.g. spotlight) the light cones of full and dimmed lighting can be specified

Photometric Lights

In newer versions, photometric lights can be used – it is important to specify the drawing unit (_Units)...

- physically more realistic, predefined lights from the **Tool Palette**,
 - the sun and the sky can be used
 - as a changing background (according to the date and time),
 - to illuminate the model with ambient light,
 - to display realistic sun angles (_Units / north direction, geolocation).
- the default lighting should be turned off.



Materials

The display can be made more realistic by using materials.

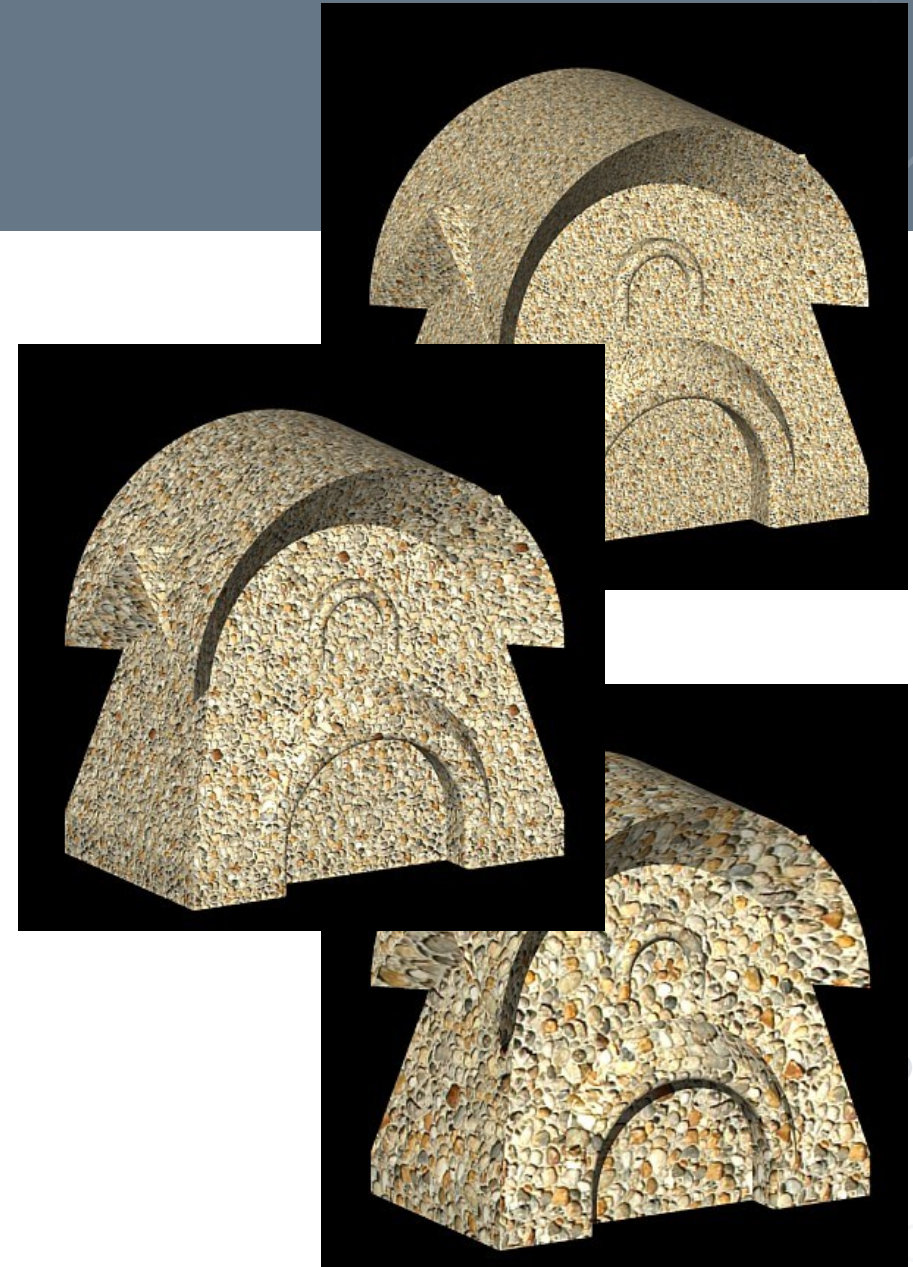
The materials can be loaded from the library ([Material Browser](#)):

- the definitions are included in the file (the textures are not!),
- the material characteristics can be assigned
 - to the layers ([_MaterialAttach](#)),
 - to the elements (by dropping the material on them or through their [Properties](#) panel),
 - or to the individual surfaces of the elements (`Ctrl+drop`).

The scale of the pattern can be adjusted

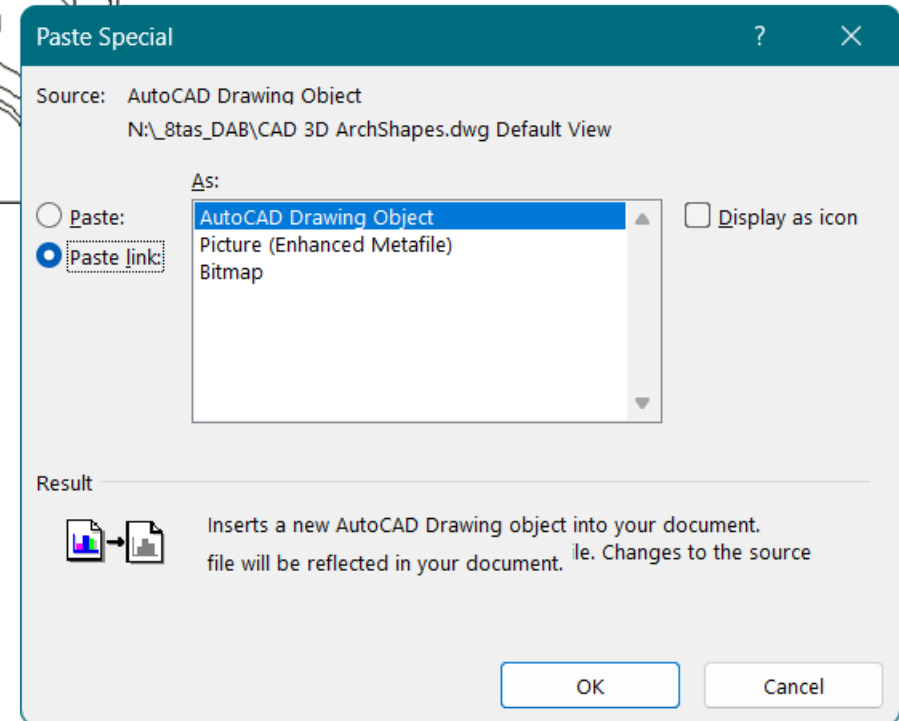
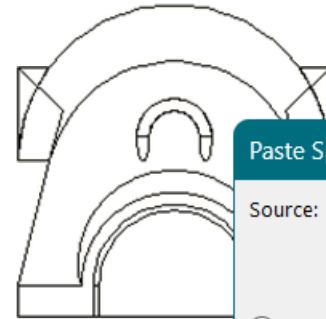
- bump map and/or transparency map can be used.

The scale of the texture also scales the model. ►



Information Exchange

- Data exchange via the clipboard
 - Screenshot – maximum data loss
 - Paste special command – the format can be selected
 - Paste link – the file can be edited with the original program
- Data exchange using files
 - intermediate file format (e.g. PDF, DXF, DWF) or
 - native format of a more common program (e.g. DWG).



Pixelgrafic Formats

- Biggest data loss / widest availability
- Image capture (printscreen) or save

Most common formats currently:

PNG

- lossless compression,
- adjustable computational demand (time),
- for line drawings (`_PNGOUT`).

JPG

- lossy compression,
- adjustable quality / loss,
- for photos, rendered images (`_JPGOUT`).

Data Exchange Formats

WMF (Windows metafile)

- clipboard equivalent,
- can contain vector and raster graphic information,
- the receiving application can choose the appropriate format,
- created by export (`_WMFOUT`),
- background can be disabled (`WMFBKGND`).

DXF (Drawing Interchange Format)

- CAD data exchange format (`_DXFOUT` `_DXFIN`),
- almost all CAD programs can read/write it,
- its use usually results in serious data loss.

Web Formats

Mixed (pixel + vector) 2D formats

→ Often created by “printing”.

PDF (Portable Document Format)

→ for displaying documents faithfully

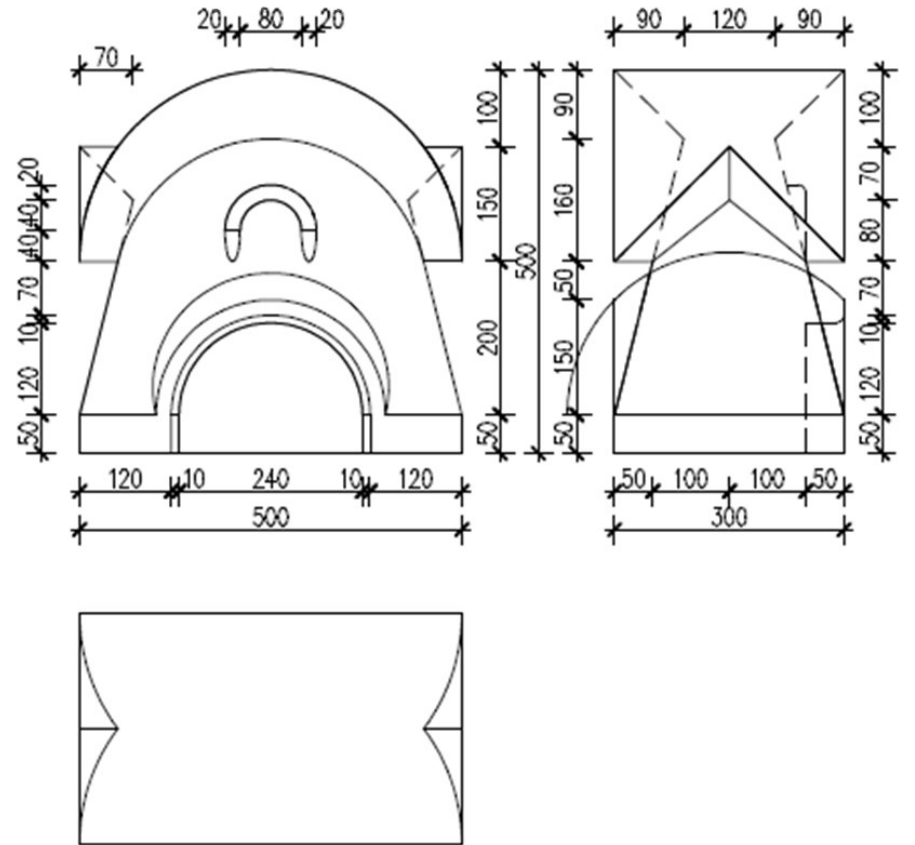
→ fonts + text information

→ pixel graphics (e.g. JPG)

DWF (Drawing Web Format)

→ primarily for plans

→ vector graphic format + pixel graphic images



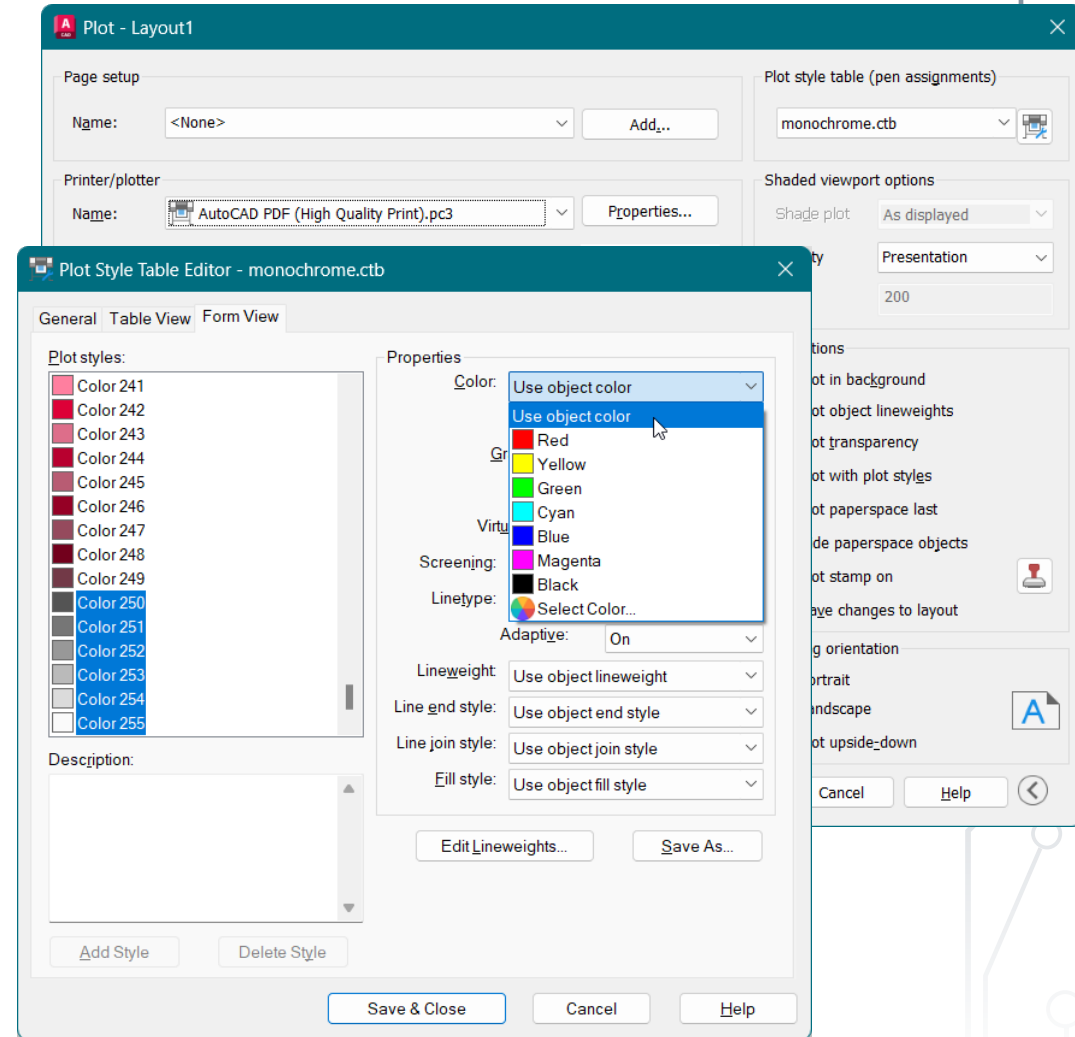
Printing

Print settings...

- Scale (`_ScaleListEdit`)
- Plot stamp
- real printer/plotter or PDF/DWF converter
- page size and output resolution
- Print color and lineweight for ACI colors
- Visual Style (e.g. hidden / rendered)

`_SolProf`

- Creating an axonometric projection of a 3D model
- Visible and hidden lines on separate layers (per viewport)
- the 2D projections can be transferred to paperspace, antialiasing can be enabled.



CAD formats

CAD data exchange formats (2D-3D)

- For data exchange between CAD and render programs
- can be created with exporting

DWG

- Quasi-industry standard
- Almost all CAD programs can read/write it

CAD viewer and editor

Windows viewer

Online viewer



AUTODESK VIEWER

View 3D designs in a browser

Autodesk Viewer supports most 2D and 3D files, including DWG, STEP, DWF, RVT, and Solidworks, and works with over 80 file types on any device. Get the feedback you need with Autodesk Viewer's annotation and drawing tools for easy online collaboration.

Over 80 file types

- ✓ DWG
 - ✓ STEP
 - ✓ DWF
- [and more](#)

Works with

- ✓ AutoCAD
 - ✓ Revit
 - ✓ Fusion 360
- [and more](#)

Platform

- ✓ Browser-based
- ✓ Windows, Mac
- ✓ Chrome OS

Key features

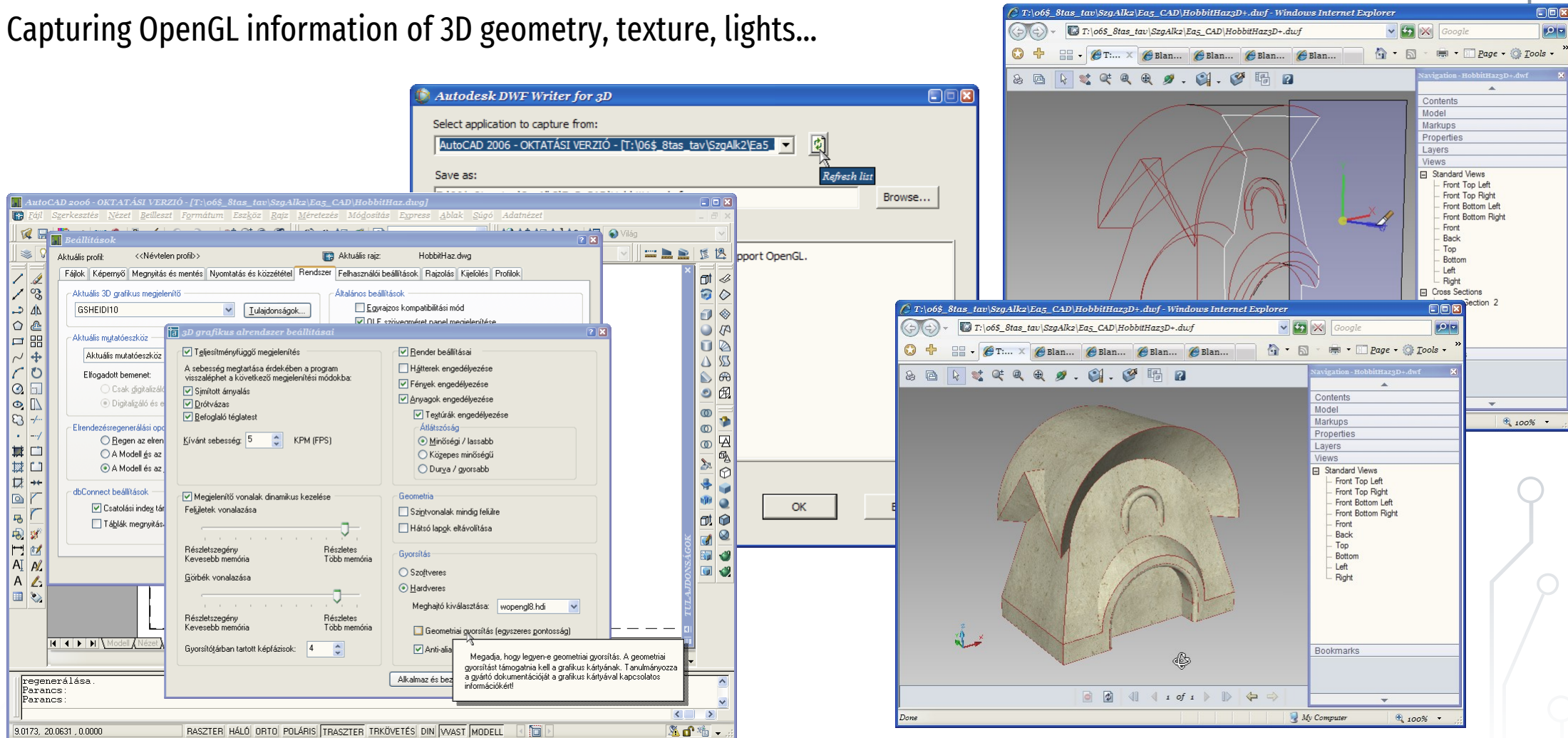
- ✓ Measure
- ✓ Mark up
- ✓ Share

[View now](#)



OpenGL Data Exchange

Capturing OpenGL information of 3D geometry, texture, lights...



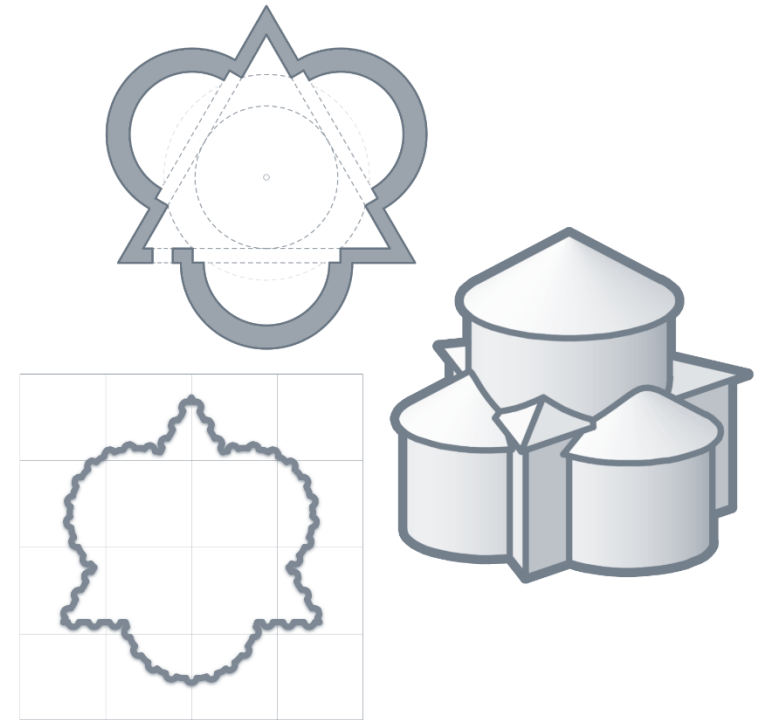
CAD

> Script, Macro

Digital Representation

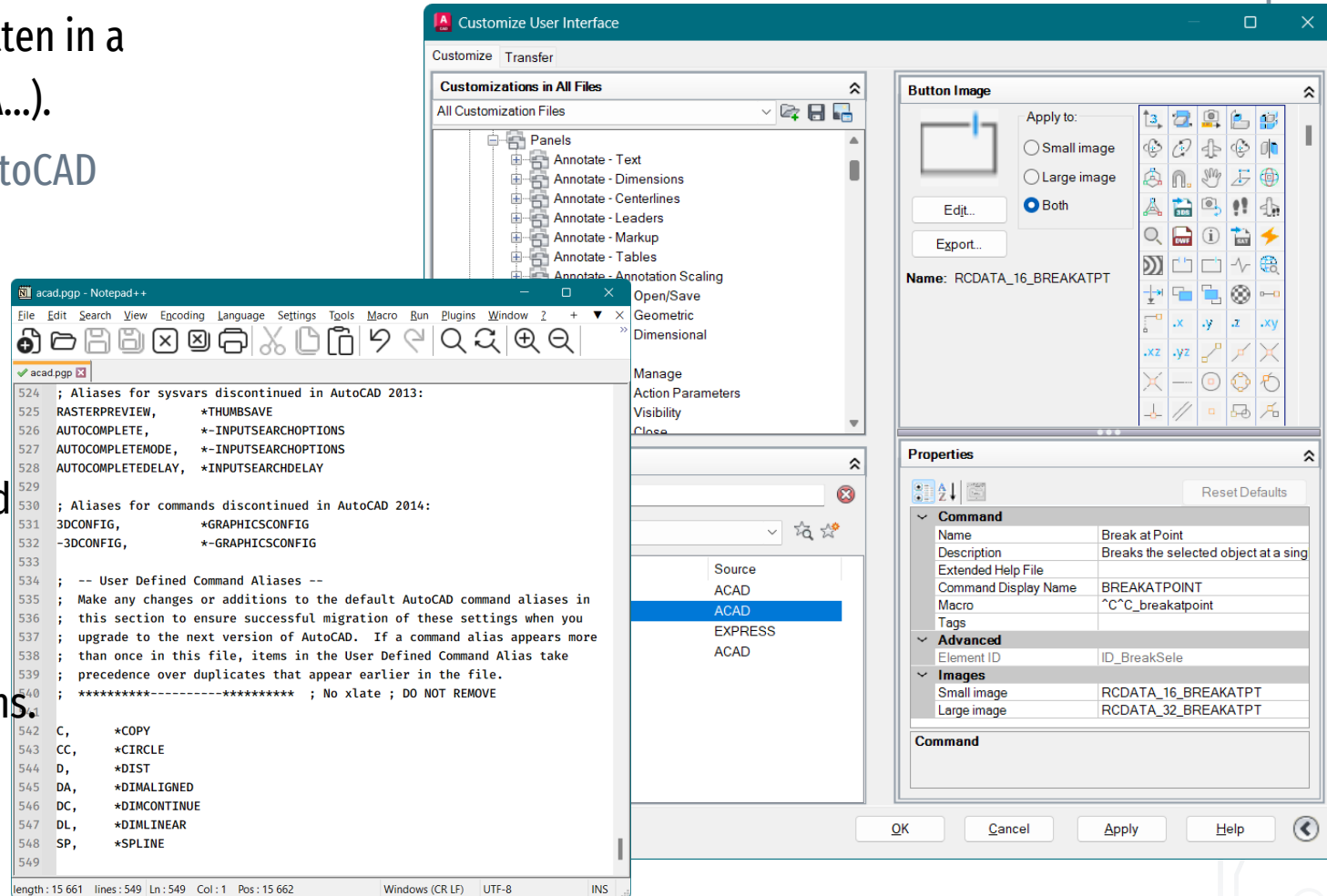
BMEEPAGA205

László Strommer PhD.



Automatization Levels

- A program module is a procedure written in a programming language (AutoLisp, VBA...).
- ActiveX enables you to manipulate AutoCAD e.g. from an Excel VBA.
- A macro (Action Macro, ACTM) is a recorded command sequence.
- Buttons and menus can be customized using DIESEL* expressions.
- A script is a list of program instructions.
- Command aliases can be used (PGP), or can be customized.



* Dumb Interpretively Evaluated String Expression Language

AutoCAD Script

The purpose of the script is usually to automate repetitive processes, command sequences...

- can be started with the Script command or copied directly to the command line,
- the SCR file's default folder is the drawing's folder.

Can be created in a text editor (e.g. Notepad++) – mustn't contain formatting codes (no docx).

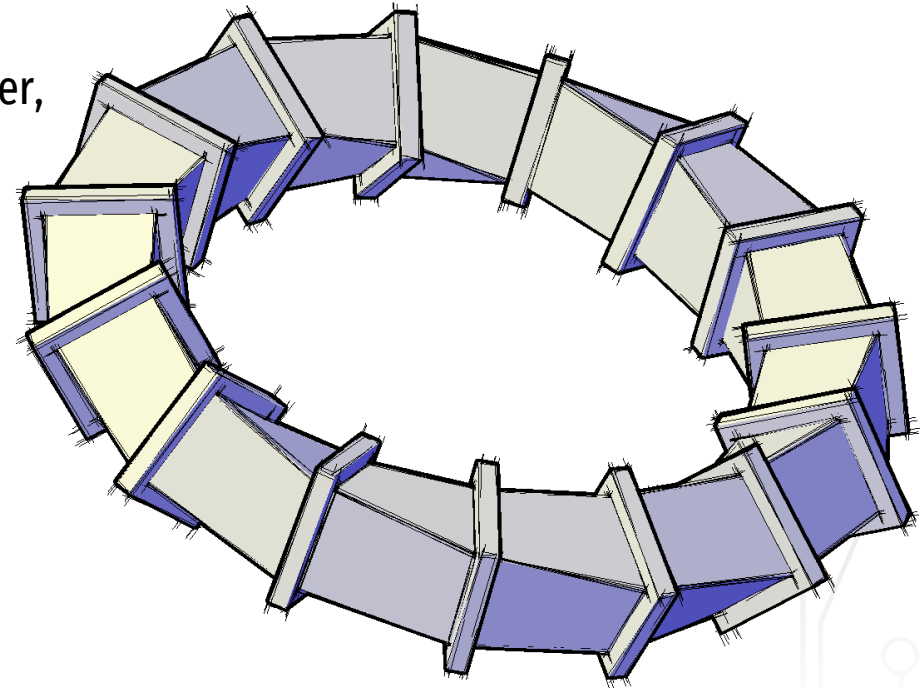
The Enter character (CR/LF) can be replaced by a space (Space),

- commands can be written in one line, the script is clearer and shorter,
- the separation of parameters is more difficult to notice.

Can contain any number of instructions, but an error can derail it.

- the **Undo** undoes the entire script,
- debugging is easier if it's broken down into shorter files.

Use and understanding are aided by the use of eloquent names, comments, and explanations.



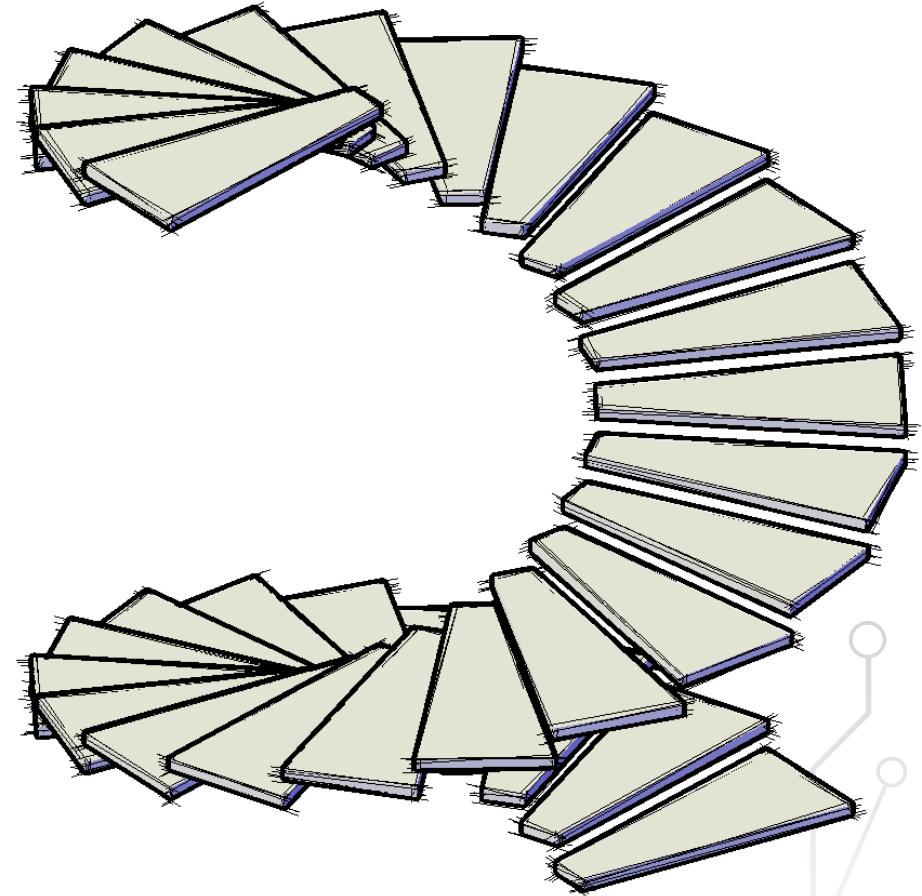
Parametric Script

Spiral staircase steps

- Copying, rotating and lifting
- Parameters specified in a spreadsheet
 - Inner radius → $r_1 = 60$ cm
 - Outer radius → $r_2 = 200$ cm
 - Step angle → $\varphi = 18^\circ$
 - Step width → $th = 5$ cm
 - Step height → $ht = 15$ cm

```
="_PLine " &r1 &"<" &\varphi &" " &r1 &","0 "  
    &r2 &","0 " &r2 &"<" &\varphi  
    &" _C_Extrude _L -" &th &" 0"
```

```
="_Copy_L _Rotate_L 0,0 " &\varphi &" _Move_L 0,0 0,0," &ht
```





BME Faculty of Architecture
Department of Morphology and Geometric Modeling

