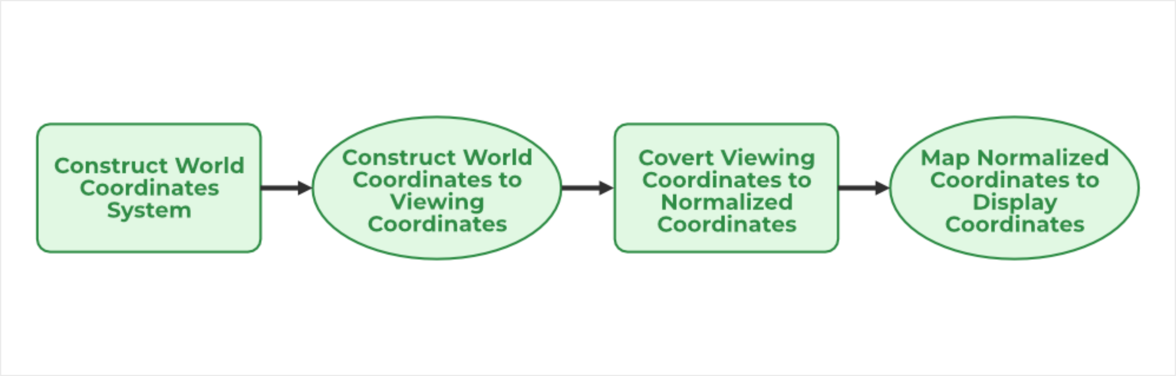
Unit -2 | 2d viewing pipeline

Two Dimension Viewing Function

Two-dimension viewing in Computer graphics is a technique to map the world coordinates system (actual coordinates of the object in the real world) to device coordinates (actual coordinates of the object on the output screen like a monitor).


Construct World Coordinates System
Construct World Coordinates to Viewing Coordinates
Covert Viewing Coordinates to Normalized Coordinates
Map Normalized Coordinates to Display Coordinates

Computer Graphics Window:

The method of selecting and enlarging a portion of a drawing is called windowing. The area chosen for this display is called a window. The window is selected by world-coordinate.

Sometimes we are interested in some portion of the object and not in full object. So we will decide on an imaginary box. This box will enclose desired or interested area of the object. Such an imaginary box is called a window.

Viewport:

An area on display device to which a window is mapped [where it is to displayed].

Basically, the window is an area in object space. It encloses the object. After the user selects this, space is mapped on the whole area of the viewport. Almost all 2D and 3D graphics packages provide means of defining viewport size on the screen. It is possible to determine many viewports on different areas of display and view the same object in a different angle in each viewport.

The size of the window is (0, 0) coordinate which is a bottom-left corner and toward right side until window encloses the desired area. Once the window is defined data outside the window is clipped before representing to screen coordinates. This process reduces the amount of data displaying signals.

The window size of the Tektronix 4.14 tube in Imperial College contains 4.96 points horizontally and 3072 points vertically.

Viewing transformation or window to viewport transformation or windowing transformation: The mapping of a part of a world-coordinate scene to device coordinates is referred to as a viewing transformation etc.

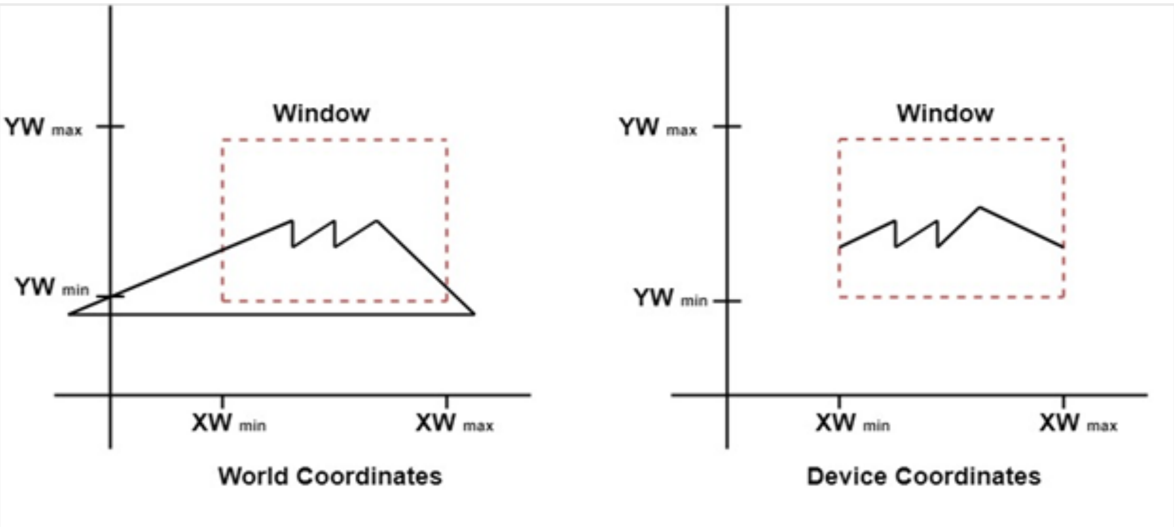


Fig: A viewing transformation using standard rectangles for the window and viewport.

Viewing transformation in several steps:

First, we construct the scene in world coordinate using the output primitives and attributes.

To obtain a particular orientation, we can set up a 2-D viewing coordinate system in the window coordinate plane and define a window in viewing coordinates system.

Once the viewing frame is established, are then transform description in world coordinates to viewing coordinates.

Then, we define viewport in normalized coordinates (range from 0 to 1) and map the viewing coordinates description of the scene to normalized coordinates.

At the final step, all parts of the picture that (i.e., outside the viewport are dipped, and the contents are transferred to device coordinates).

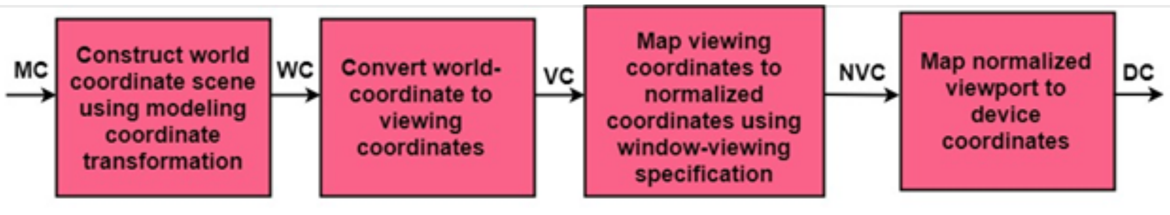

MC
Construct world coordinate scene using modeling coordinate transformation
WC
Convert world-coordinate to viewing coordinates
VC
Map viewing coordinates to normalized coordinates using window-viewing specification
NVC
Map normalized viewport to device coordinates
DC

Fig: The two-dimensional viewing-transformation.

By changing the position of the viewport: We can view objects at different locations on the display area of an output device as shown in fig:

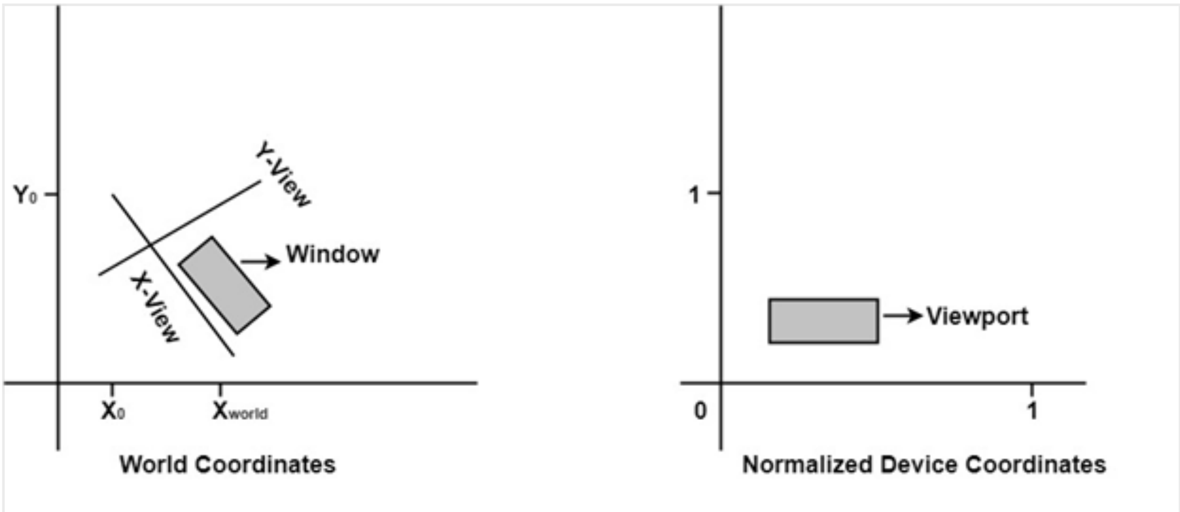


Fig:Setting up a rotated world window and corresponding normalized coordinate viewport.

By varying the size of viewports:

We can change the size and proportions of displayed objects. We can achieve zooming effects by successively mapping different-sized windows on a fixed-size viewport.

As the windows are made smaller, we zoom in on some part of a scene to view details that are not shown with larger windows.

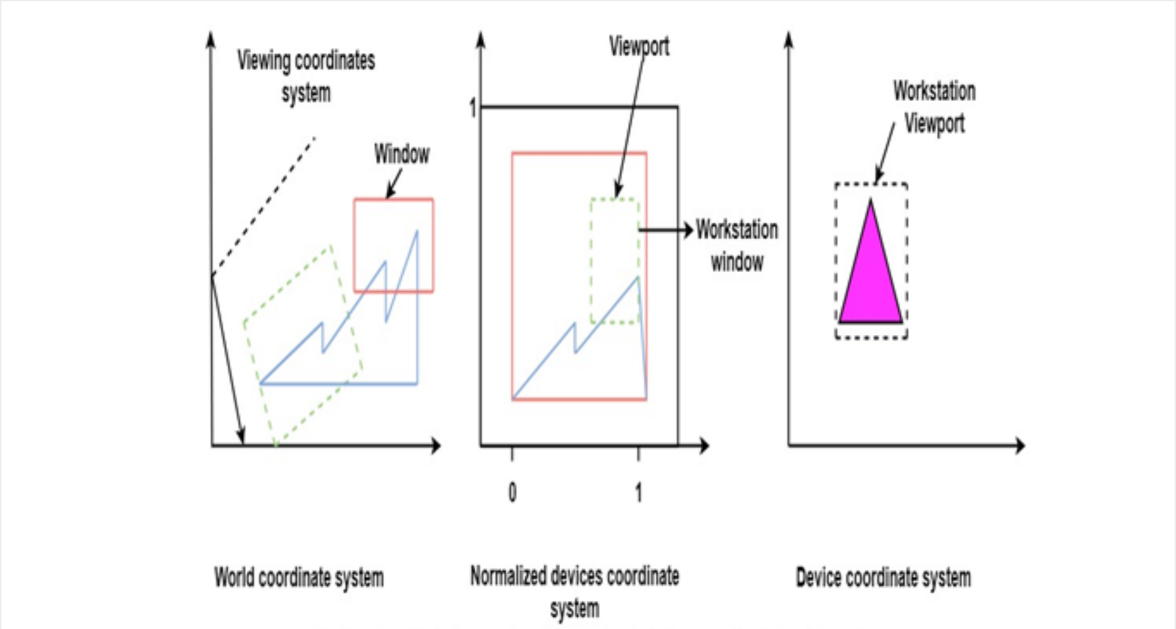


Fig: Zooming effects by mapping different-sized windows on a fixed-size viewport.

Computer Graphics Window to Viewport Co-ordinate Transformation

Once object description has been transmitted to the viewing reference frame, we choose the window extends in viewing coordinates and selects the viewport limits in normalized coordinates. Object descriptions are then transferred to normalized device

coordinates:
We do this thing using a transformation that maintains the same relative placement of an object in normalized space as they had in viewing coordinates.
If a coordinate position is at the center of the viewing window:
It will display at the center of the viewport.
Fig shows the window to viewport mapping. A point at position (xw, yw) in window mapped into position (xv, yv) in the associated viewport.

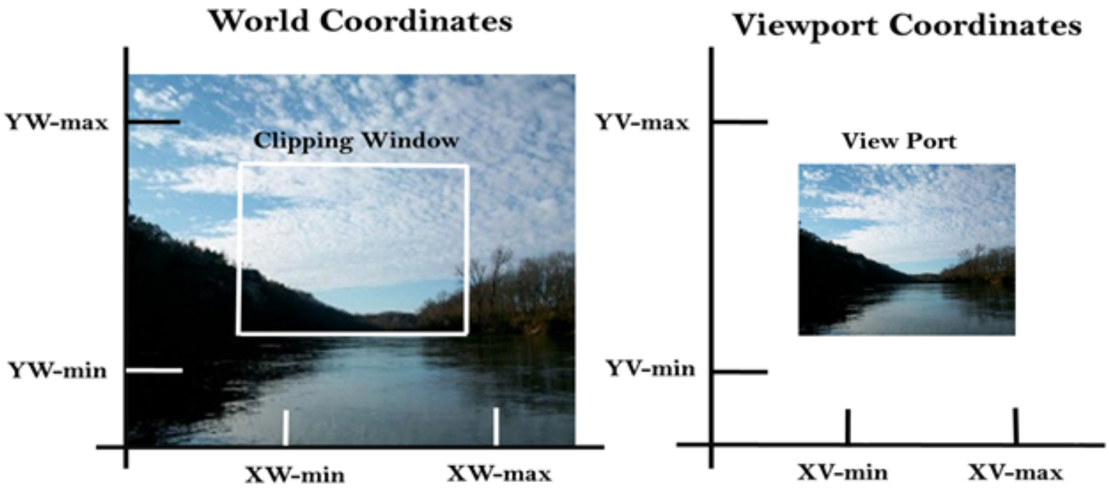


Fig: Window to viewport mapping

In order to maintain the same relative placement of the point in the viewport as in the window, we require:

$$\frac{xv-xv_{min}}{xv_{max}-xv_{min}} = \frac{xw-xw_{min}}{xw_{max}-xw_{min}} \text{ ....equation 1}$$

$$\frac{yv-yv_{min}}{yv_{max}-yv_{min}} = \frac{yw-yw_{min}}{yw_{max}-yw_{min}}$$

Solving these impressions for the viewport position (xv, yv), we have
$xv=xv_{min}+(xw-xw_{min})sx$
$yv=yv_{min}+(yw-yw_{min})sy$ ....equation 2
Where scaling factors are

$$sx = \frac{xv_{max}-xv_{min}}{xw_{max}-xw_{min}}$$

$$sy = \frac{yv_{max}-yv_{min}}{yw_{max}-yw_{min}}$$

Equation (1) and Equation (2) can also be derived with a set of transformation that converts the window or world coordinate area into the viewport or screen coordinate area. This conversation is performed with the following sequence of transformations:

1. Perform a scaling transformation using a fixed point position (xw_{min},yw_{min}) that scales the window area to the size of the viewport.
2. Translate the scaled window area to the position of the viewport. Relative proportions of objects are maintained if the scaling factors are the same (sx=sy).

From normalized coordinates, object descriptions are mapped to the various display devices.

Any number of output devices can we open in a particular app, and three windows to viewport transformation can be performed for each open output device.

This mapping called workstation transformation (It is accomplished by selecting a window area in normalized space and a viewport area in the coordinates of the display device).

As in fig, workstation transformation to partition a view so that different parts of normalized space can be displayed on various output devices).

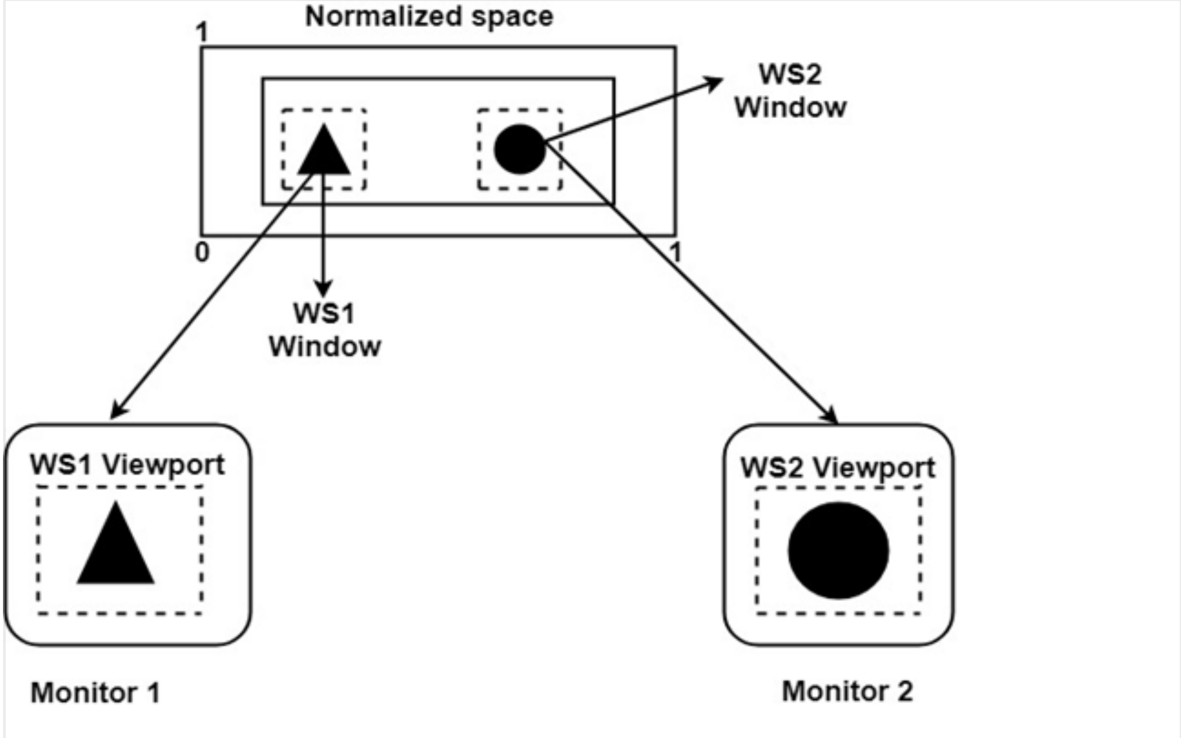
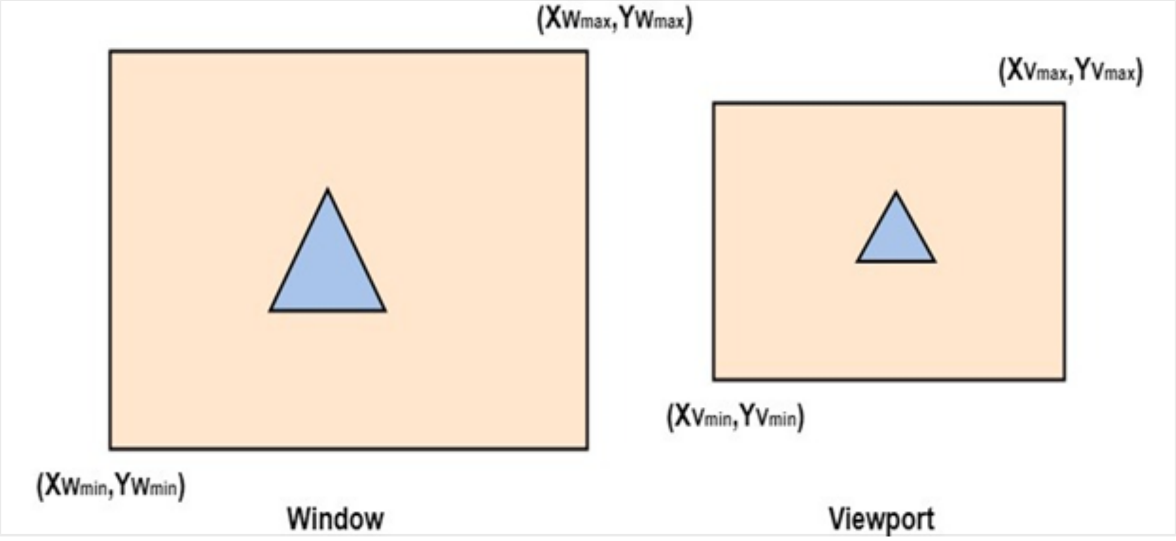


Fig:Mapping selected parts of a scene in normalized coordinates to different video monitors with workstation transformation.

Matrix Representation of the above three steps of Transformation:



Step1: Translate window to origin 1

T_x=-Xw_{min} T_y=-Yw_{min}

Step2: Scaling of the window to match its size to the viewport

$S_x=(Xv_{max}-Xv_{min})/(Xw_{max}-Xw_{min})$

$S_y=(Yv_{max}-Yv_{min})/(Yw_{max}-Yw_{min})$

Step3: Again translate viewport to its correct position on screen.

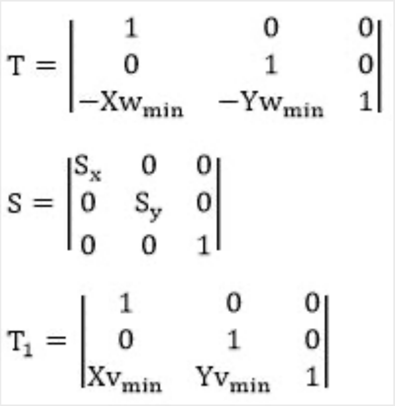
$T_x=Xv_{min}$

$T_y=Yv_{min}$

Above three steps can be represented in matrix form:

$$VT=T * S * T_1$$

T = Translate window to the origin
S=Scaling of the window to viewport size
T_1=Translating viewport on screen.



$$T = \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -Xw_{min} & -Yw_{min} & 1 \end{vmatrix}$$

$$S = \begin{vmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

$$T_1 = \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ Xv_{min} & Yv_{min} & 1 \end{vmatrix}$$

Viewing Transformation= $T * S * T_1$

Advantage of Viewing Transformation:

We can display picture at device or display system according to our need and choice.

Note:

- World coordinate system is selected suits according to the application program.
- Screen coordinate system is chosen according to the need of design.
- Viewing transformation is selected as a bridge between the world and screen coordinate.