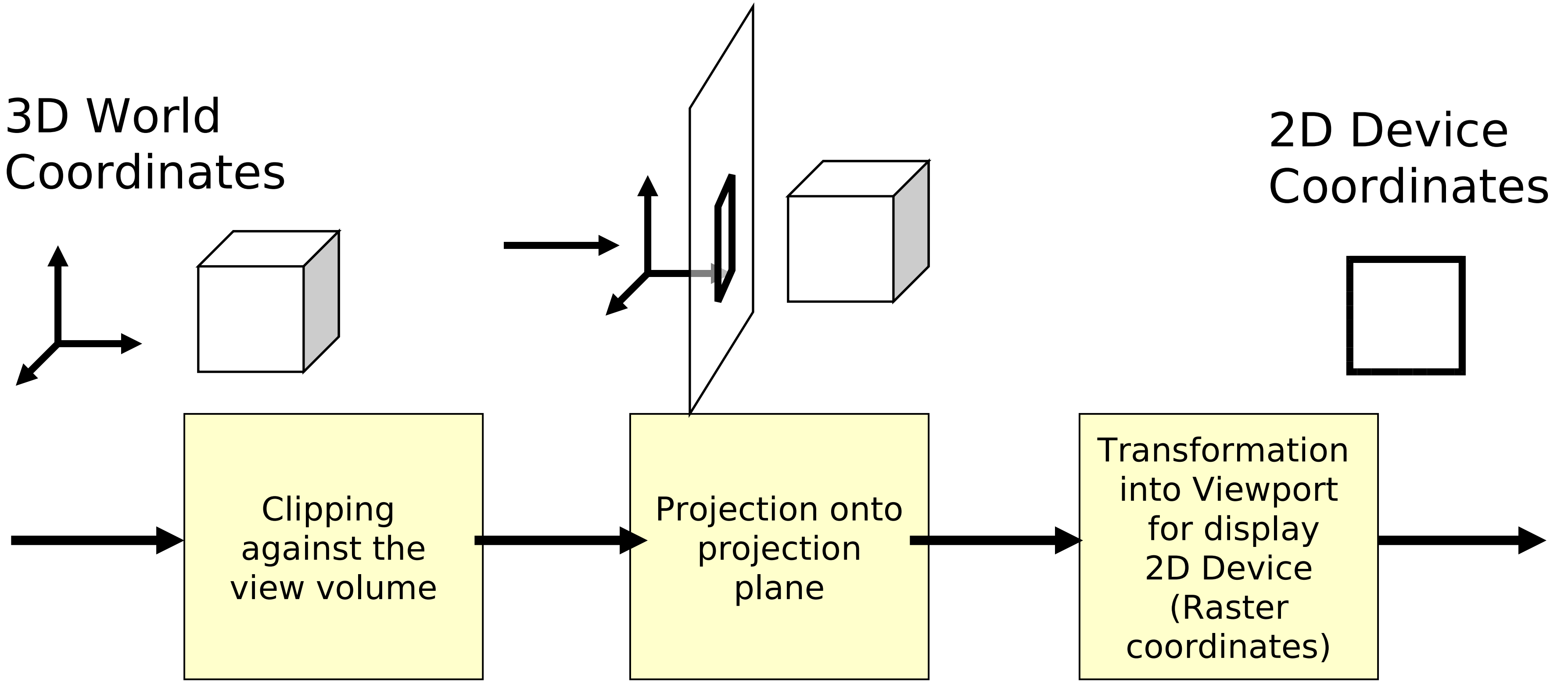
Computer Graphics: 2-Viewing

Prof. Dr. Charles A. Wüthrich,
Fakultät Medien, Medieninformatik
Bauhaus-Universität Weimar
caw AT medien.uni-weimar.de

Viewing

- Here:
  - Viewing in 3D
    - Planar Projections
    - Camera and Projection
    - View transformation

Pipeline

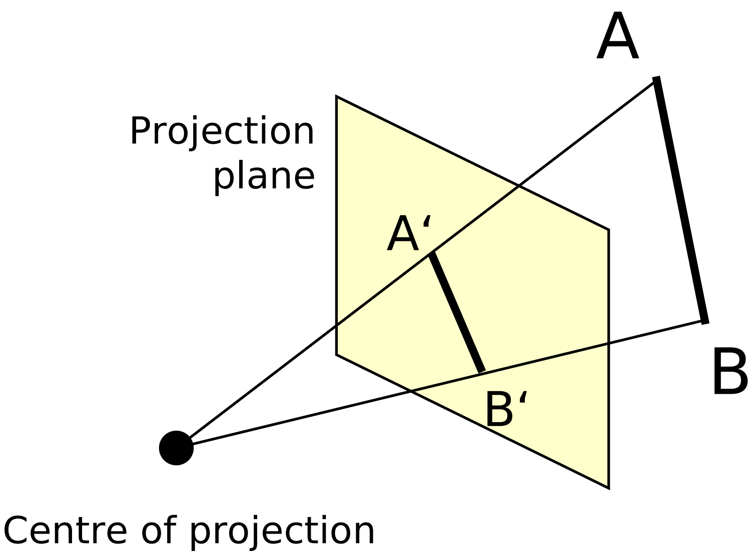

3D World Coordinates
2D Device Coordinates
Clipping against the view volume
Projection onto projection plane
Transformation into Viewport for display 2D Device (Raster coordinates)

Projections

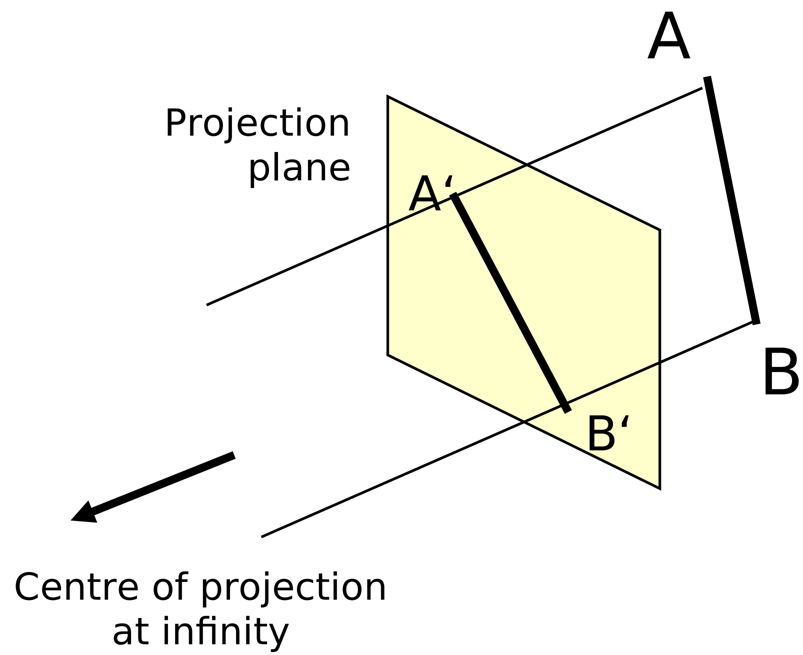
- Maps Points of a coordinate system in the n-dimensional space into a space of smaller dimension.
  In computer graphics :3D -> 2D
- Idea:
  - Compute intersections of projection rays p with a projection plane π
  - The rays pass through point to be projected and the centre of projection
- NOTE: you can‘t invert this!
  ~ loss of information

Projections

Perspective Projection

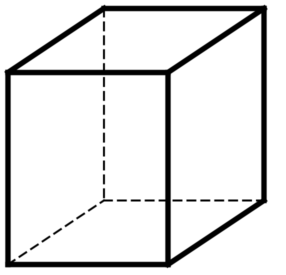


Parallel Projection

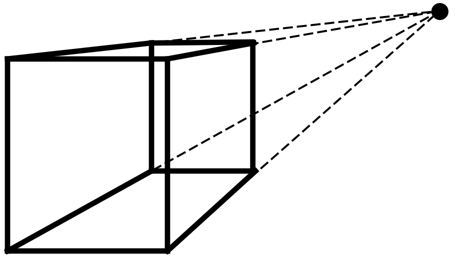


Projections

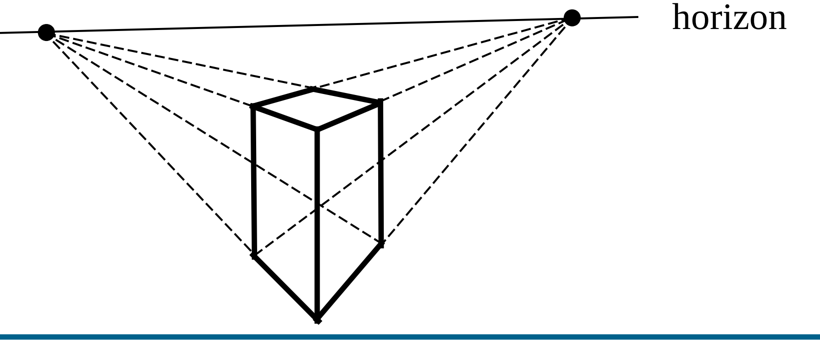
Parallel
(orthographic)
Projection



Perspective
Projection
(1 vanishing pt)



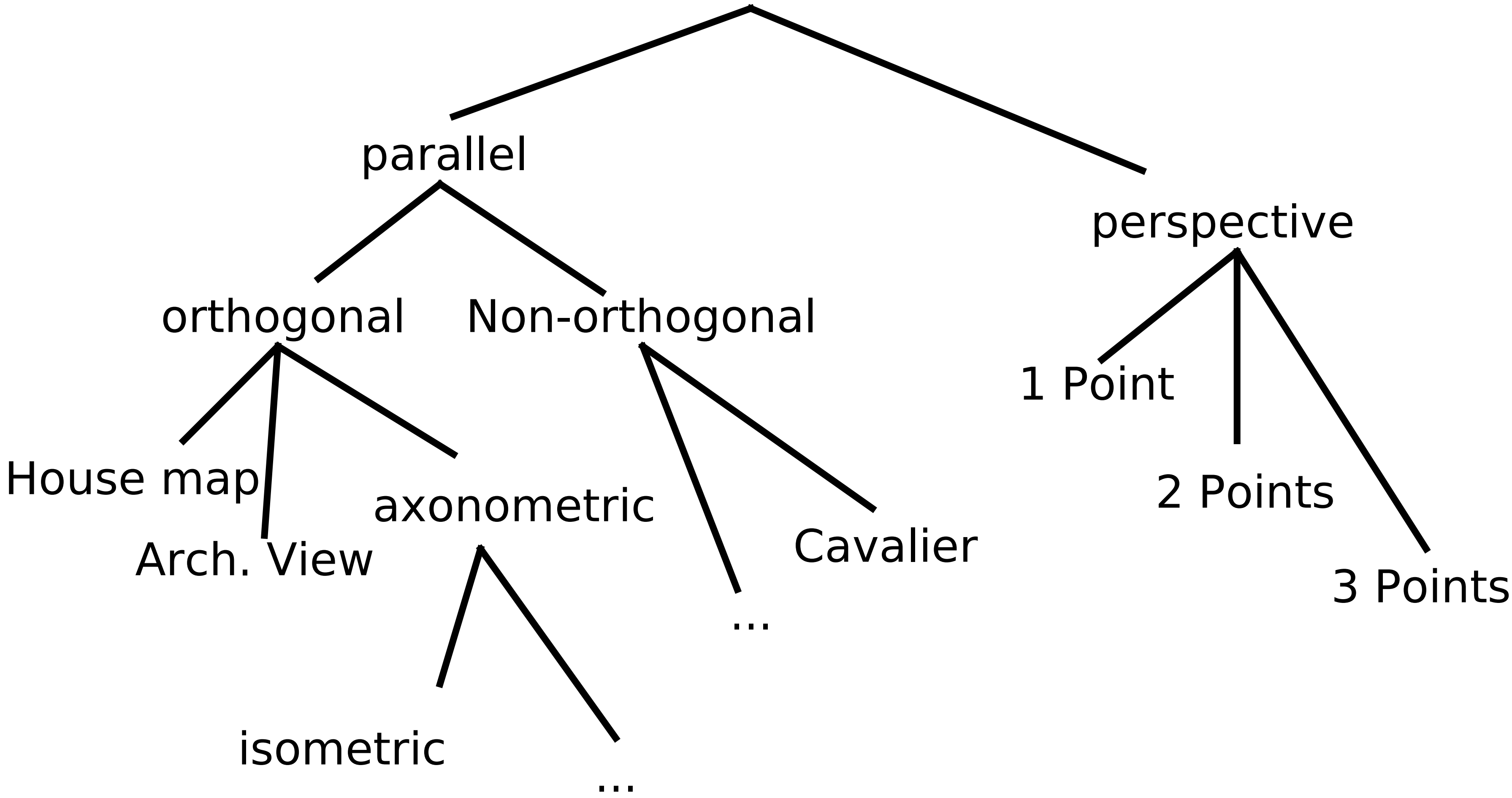
Perspective
Projection
(2 vanishing pts)



Projections

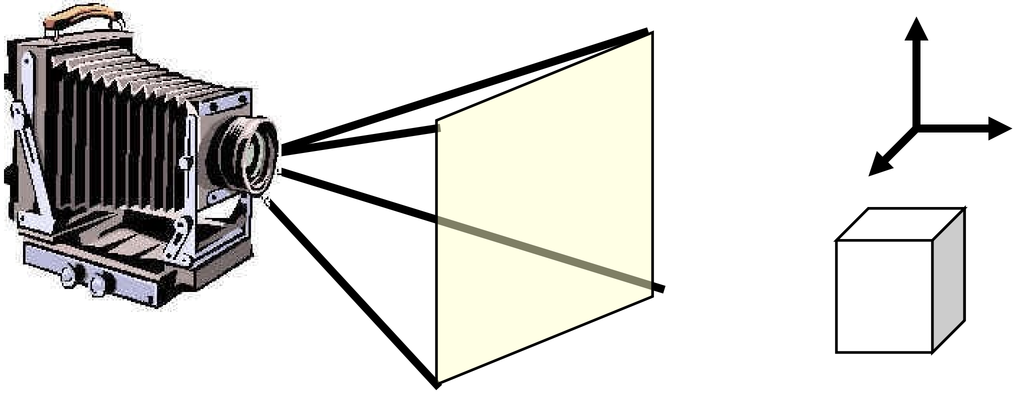
- Perspective projection models human view system (or photography)
- Realistic but:
  - Scales not preserved
  - Angles not preserved
- parallel projection less realistic but
  - preserve scales and angles
  - Preserve parallel lines

Planar projections



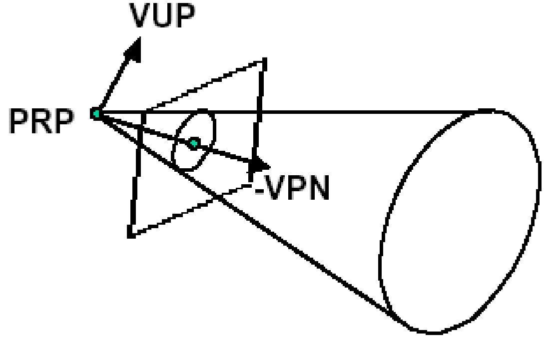
Camera metaphor

- Goal: use camera to transform world coordinates into screen coordinates
- Requirement: description of the camera

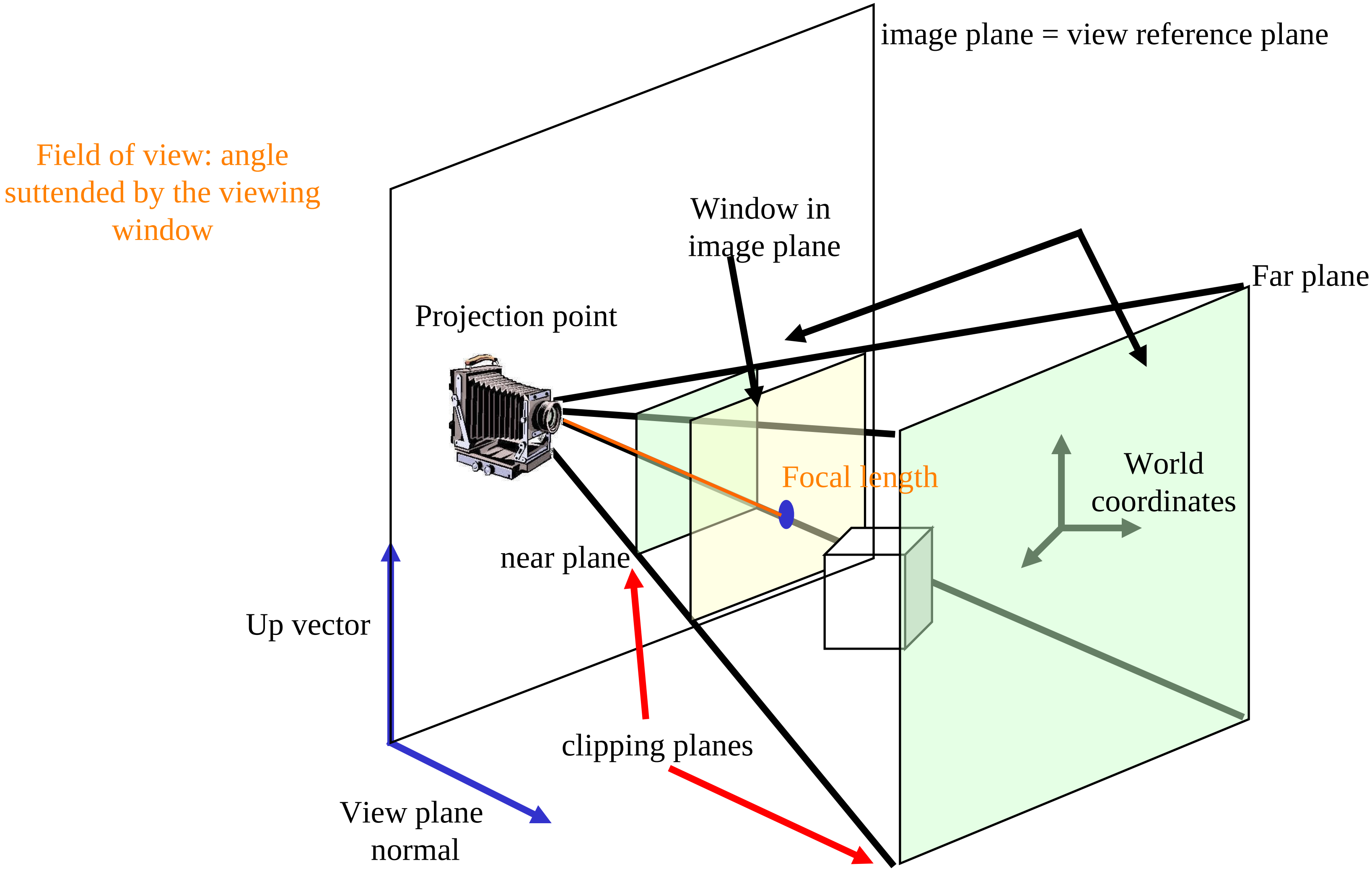


Description of the camera

- Position and orientation in World Coordinates (WCS)
  - Projection point (projection reference point, PRP)
  - Normal to the projection plane (view plane normal, VPN)
  - Up-vector (view up vector, VUP)



Camera description



Camera description

- Clipping
  - Window on projection plane (e.g., 35mm film)
  - Determines also the view direction (von PRP t the mid point CW of the Window)
- Field of View
  - Distance of the view plane from the origin (focal length). Alternatively,
  - Opening angle (field of view) (FOV)
- Mapping to raster coordinates
  - Resolution
  - Aspect ratio
- Front and back clipping-planes
  - Limits view to „interesting part“ of the scene.
  - Avoids singularities in computations (by looking back)
  - Limits objects that are too far away (background)

Projection with Matrices

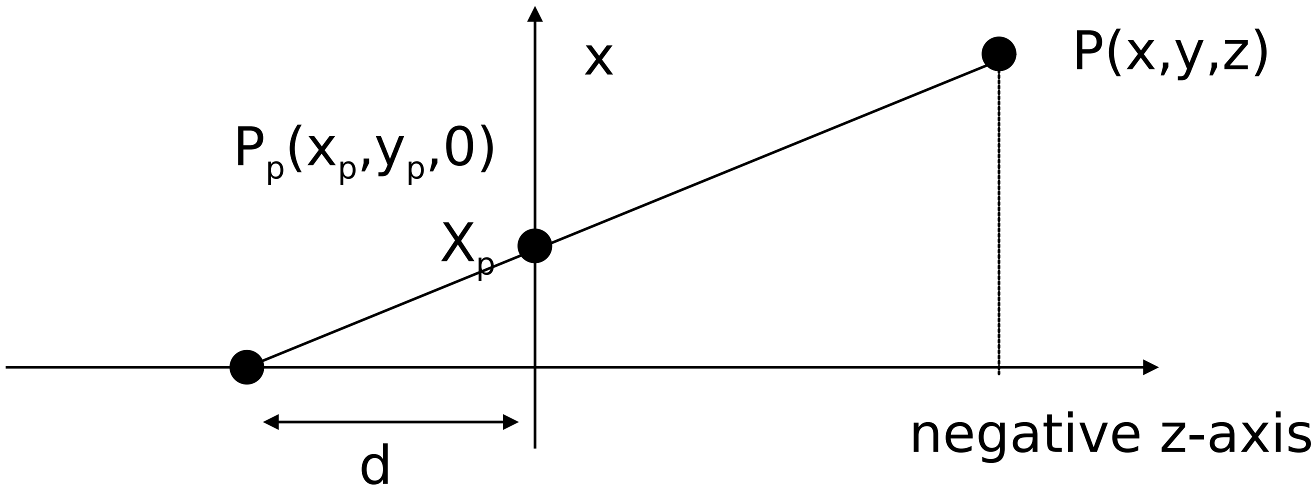
- Projective transformations can be represented through Matrices
- Easy example:
  - Parallel projection onto x-y plane

$$\xi_\pi = \xi \qquad \psi_\pi = \psi \qquad \zeta_\pi = 0$$

$$M_{ort} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$P_{ort} = M_{ort} P$$

Perspective projection



$$\frac{x_p}{d} = \frac{x}{d - z}$$

$$x_p = \frac{d \cdot x}{d - z} = \frac{x}{1 - \frac{z}{d}}$$

Perspective projection

- The transformation $P(x,y,z) \rightarrow P_p(x_p,y_p,0)$ is performed by multiplying with the matrix M_{per}:

$$P_p = M_{per} P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{d} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ 0 \\ 1 - \frac{z}{d} \end{bmatrix}$$

View Transformation

- Problem:
  - This works well if coordinate systems are already unified and aligned with world coordinates, but not for the general case.
  - Thus we transform the world to where we need it.
- Goal:
  - VRP is at origin
  - View direction is -Z, Y ist Up vector

Normalization

- Moving VRP to the origin: T(-VRP)
- Rotate coordinate system, so that Up vector points UP and the view direction is –Z
  - orthonormed basis of the Camera Coordinate system with

$$R_z = \frac{VPN}{\left\|VPN\right\|} \qquad R_x = \frac{VUP \cdot R_z}{\left\|VUP \cdot R_z\right\|} \qquad R_y = R_z \cdot R_x$$

Normalization

- This results in the rotation matrix:

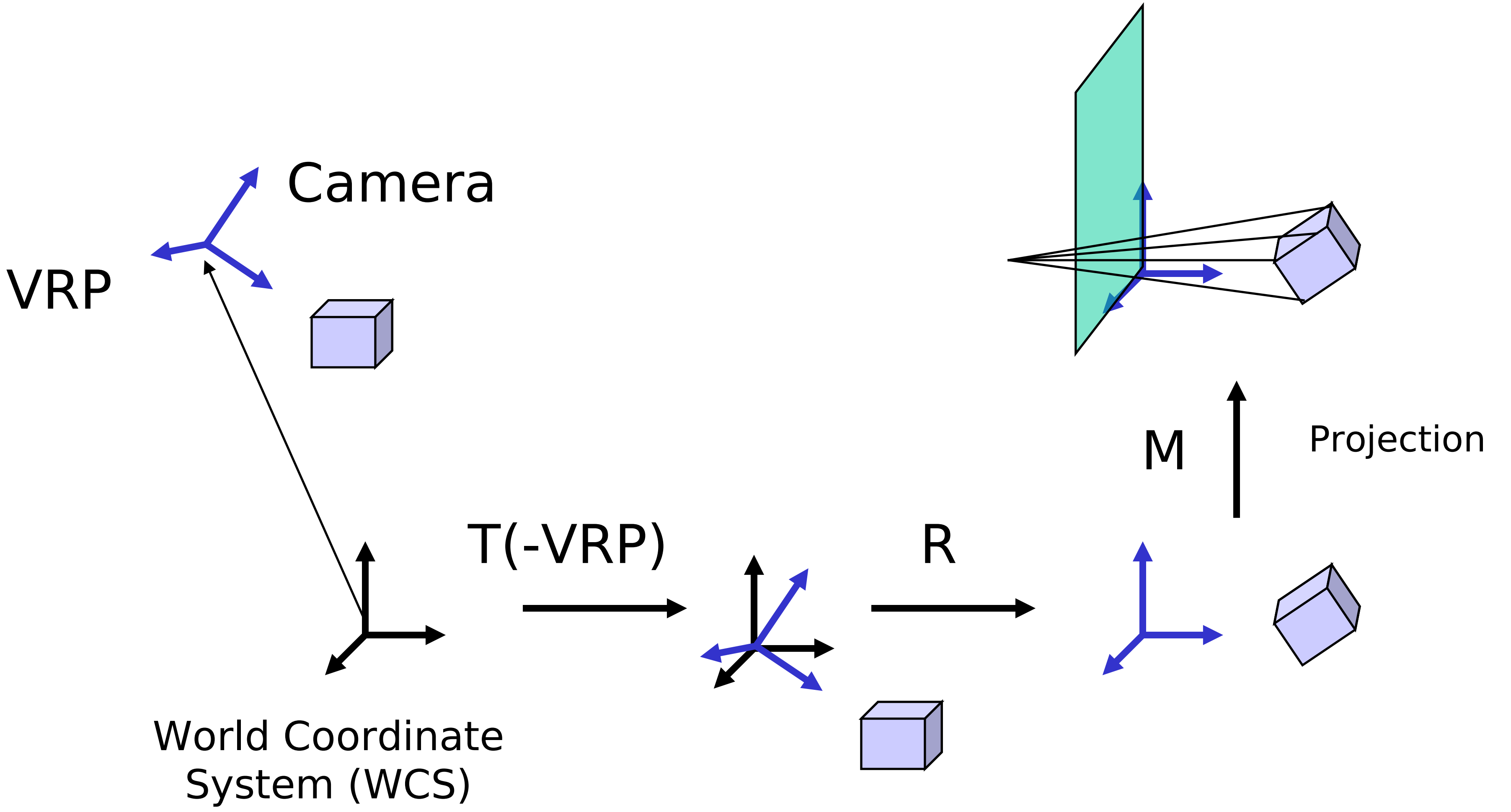
$$R = \begin{bmatrix} r_{1x} & r_{2x} & r_{3x} & 0 \\ r_{1y} & r_{2y} & r_{3y} & 0 \\ r_{1z} & r_{2z} & r_{3z} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R_x{}^T = \begin{bmatrix} r_{1x} & r_{1y} & r_{1z} & 1 \end{bmatrix}$$

$$R_y{}^T = \begin{bmatrix} r_{2x} & r_{2y} & r_{2z} & 1 \end{bmatrix}$$

$$R_z{}^T = \begin{bmatrix} r_{3x} & r_{3y} & r_{3z} & 1 \end{bmatrix}$$

Recapping



Recapping

- Transformation of the WCS into 2D screen coordinates through matrix multiplication
- Parameter of the virtual camera determine the composing transformation steps
- Of course, if I describe otherwise the camera and viewing system -> different matrices

Note: Some camera parameters are missing, e.g. CW and the aspect ratio of the window. Such parameter can be integrated through simple transformations in the viewing transformations.

End

+++ Ende - The end - Finis - Fin - Fine +++ Ende - The end - Finis - Fin - Fine +++