Rotation around an arbitrary axis

Euler's theorem: Any rotation or sequence of rotations around a point is equivalent to a single rotation around an axis that passes through the point.

What does the matrix look like?

Rotation around an arbitrary axis through the origin

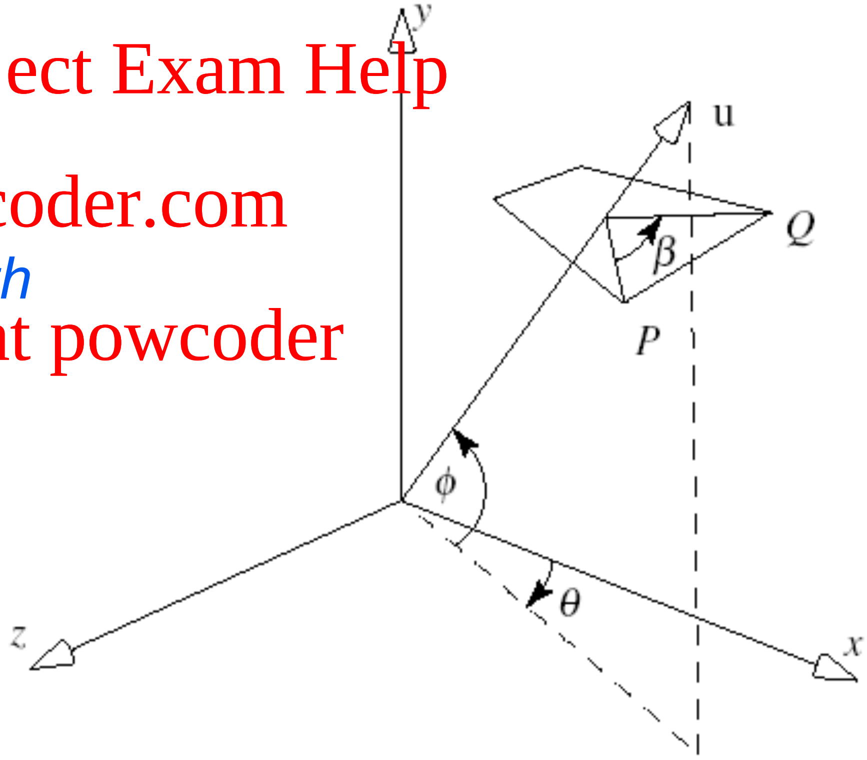
Axis: **u**

Point: P

Angle: β

Approach (one of many):

1. *Two rotations to align* ***u*** *with x-axis (arbitrary choice)*
2. *Do x-roll by β*
3. *Undo the alignment*

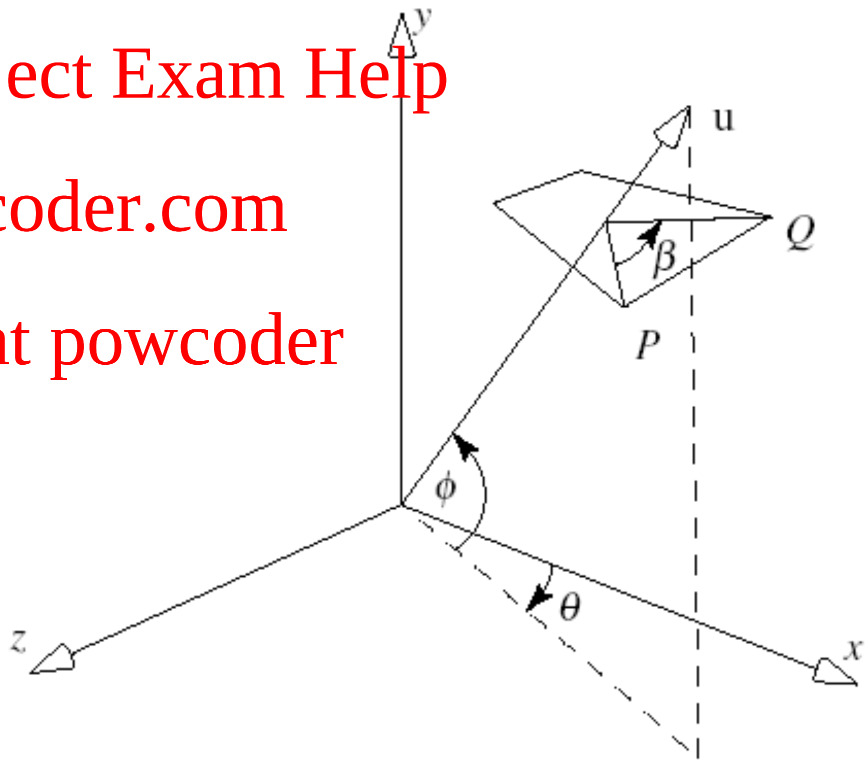


Derivation

1. $R_z(-\phi)R_y(\theta)$
2. $R_x(\beta)$
3. $R_y(-\theta)R_z(\phi)$

Altogether:

$$\boxed{R_y(-\theta)R_z(\phi)\ R_x(\beta)\ R_z(-\phi)R_y(\theta)}$$

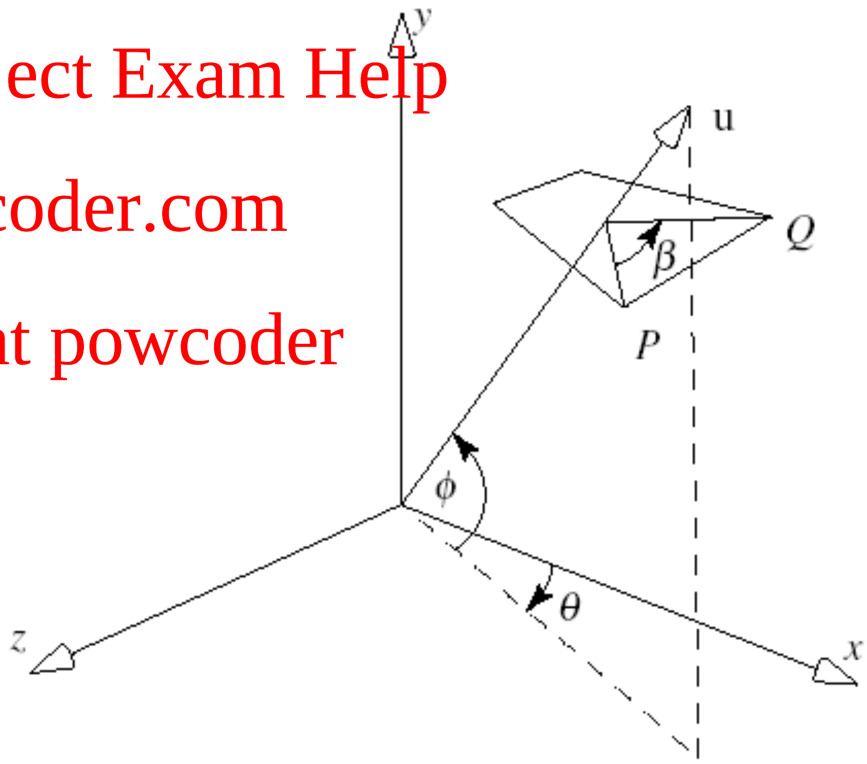


Derivation

1. $R_z(-\phi)R_y(\theta)$
2. $R_x(\beta)$
3. $R_y(-\theta)R_z(\phi)$

Altogether:

$$\boxed{R_y(-\theta)R_z(\phi)\, R_x(\beta)\, R_z(-\phi)R_y(\theta)}$$



Parameters:

$$cos(\theta) = u_x/\sqrt{u_x^2 + u_z^2}$$
$$sin(\theta) = u_z/\sqrt{u_x^2 + u_z^2}$$
$$sin(\phi) = u_y/|\mathbf{u}|$$
$$cos(\phi) = \sqrt{u_x^2 + u_z^2}/|\mathbf{u}|$$

Derivation

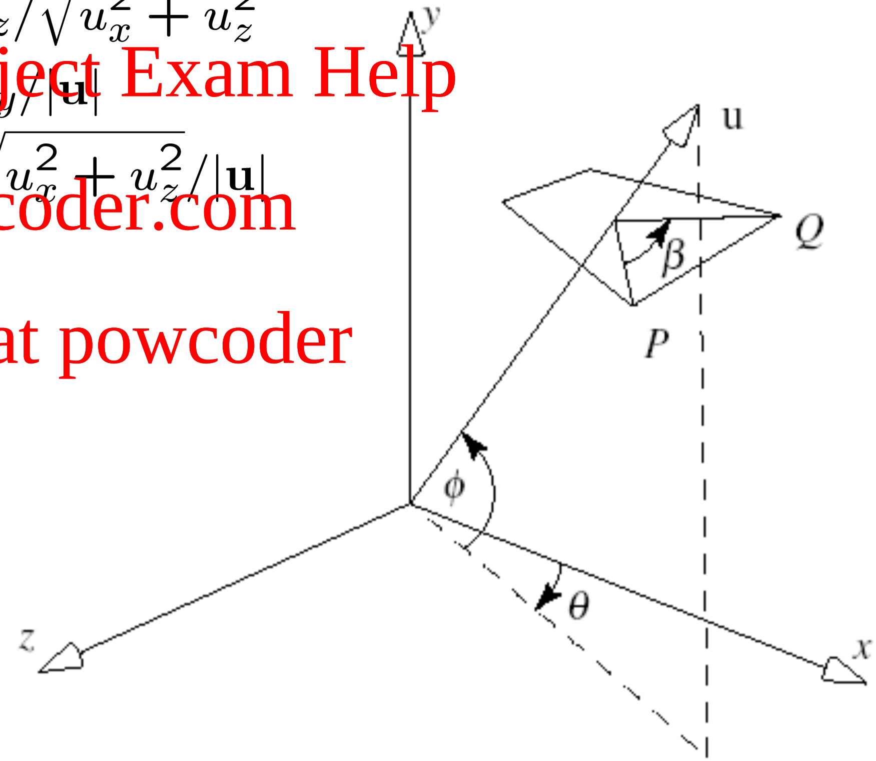
1. $R_z(-\phi)R_y(\theta)$
2. $R_x(\beta)$
3. $R_y(-\theta)R_z(\phi)$

$$cos(\theta) = u_x/\sqrt{u_x^2 + u_z^2}$$

$$sin(\theta) = u_z/\sqrt{u_x^2 + u_z^2}$$

$$sin(\phi) = u_y/|\mathbf{u}|$$

$$cos(\phi) = \sqrt{u_x^2 + u_z^2}/|\mathbf{u}|$$



Altogether:

$$\boxed{R_y(-\theta)R_z(\phi)\ R_x(\beta)\ R_z(-\phi)R_y(\theta)}$$

Exercise:

Derive the matrix for rotation around an axis that does not pass through the origin

Properties of affine transformations

1. ***Preservation of affine combinations of points.***
2. ***Preservation of lines and planes.***
3. ***Preservation of parallelism of lines and planes.***
4. ***Relative ratios on a line are preserved.***
5. ***Affine transformations are composed of elementary ones.***

Affine Combinations of Points

$$W = a_1 P_1 + a_2 P_2$$

$$T(W) = T(a_1 P_1 + a_2 P_2) = a_1 T(P_1) + a_2 T(P_2)$$

Proof: from linearity of matrix multiplication

$$MW = M(a_1 P_1 + a_2 P_2) = a_1 M P_1 + a_2 M P_2$$

Preservations of Lines and Planes

Line:

$$L(t) = (1-t)P_1 + tP_2$$

$$T(L(t)) = (1-t)T(P_1) + tT(P_2)$$

Plane

$$Pl(s,t) = (1-s-t)P_1 + tP_2 + sP_3$$

$$T(Pl(s,t)) = (1-s-t)T(P_1) + tT(P_2) + sT(P_3)$$

Proof: Direct consequence of previous property

Preservation of Parallelism for Lines and Planes

$$L(t) = P + t\mathbf{u}$$

$$ML = M(P + t\mathbf{u}) = MP + M(t\mathbf{u}) \rightarrow$$

$$ML = MP + t(M\mathbf{u})$$

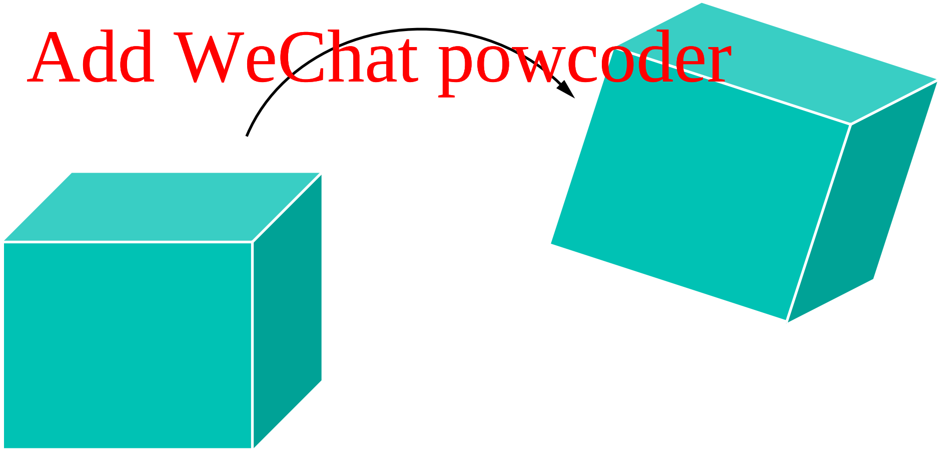
$M\mathbf{u}$ independent of P.

Similarly for planes.

Rigid body transformation

Combination of a translation and a rotation

- Preserve lines, angles and distances
- 6 Degrees of freedom in 3D



General form of 3D affine transformations

Rotation, Scaling, Shear

Translation

$$\left(\begin{array}{ccc|c} \boxed{\begin{matrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{matrix}} & \boxed{\begin{matrix} m_{14} \\ m_{24} \\ m_{34} \end{matrix}} \\ 0 \quad 0 \quad 0 & 1 \end{array}\right)$$

Transforming Points and Vectors

- Points

$$\begin{pmatrix} Q_x \\ Q_y \\ Q_z \\ 1 \end{pmatrix} = \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} P_x \\ P_y \\ P_z \\ 1 \end{pmatrix}$$

- Vectors

$$\begin{pmatrix} w_x \\ w_y \\ w_z \\ 0 \end{pmatrix} = \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} v_x \\ v_y \\ v_z \\ 0 \end{pmatrix}$$

Advanced concepts

Generalized shears

Decomposition of 2D AT:

2D : M = T Sh S R

3D: M = T S R Sh_1 Sh_2

Rotations in 3D

Gimbal lock

Quaternions

Exponential maps

Transformations of Coordinate systems

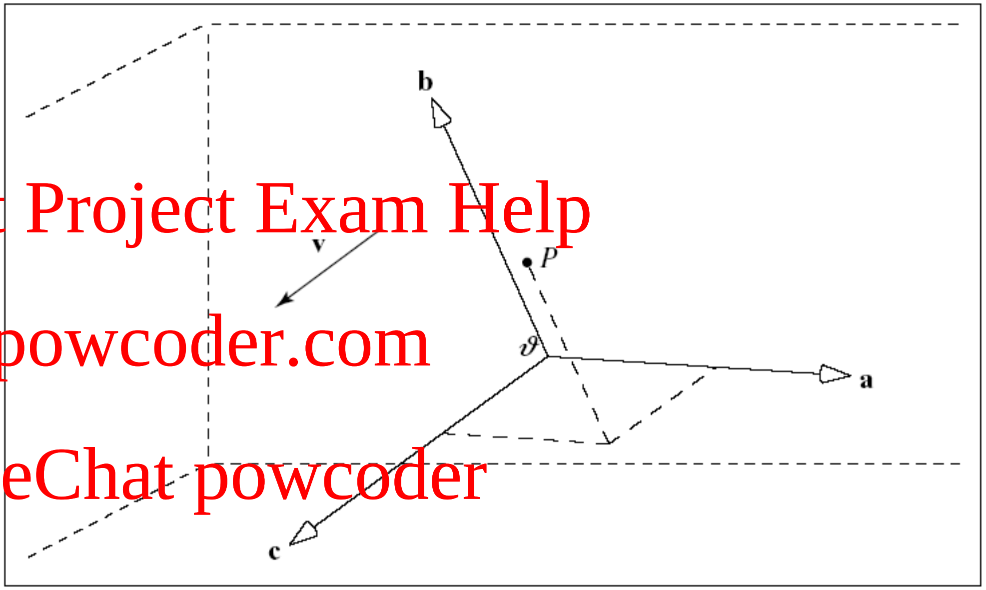
Coordinate systems consist of vectors and an origin (point), therefore we can transform them just like any other group of points and vectors

Alternative way to think of transformations:

- Transformations as a change of basis

Reminder: Coordinate systems

Coordinate system: ($\mathbf{a}$,$\mathbf{b}$,$\mathbf{c}$,θ)



$$\mathbf{v} = (v_1, v_2, v_3) \rightarrow \mathbf{v} = v_1\mathbf{a} + v_2\mathbf{b} + v_3\mathbf{c}$$

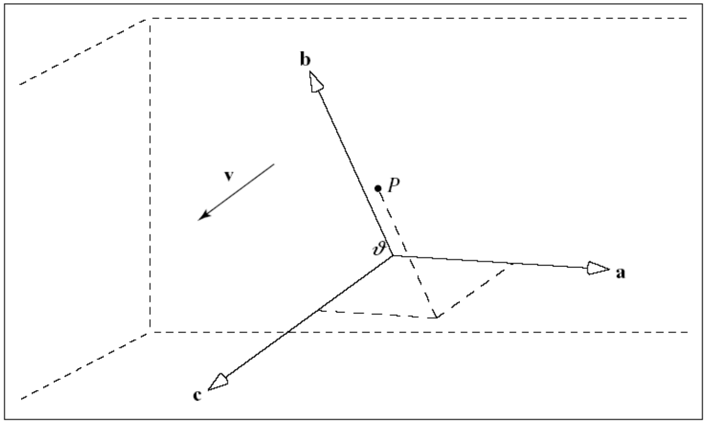
$$P = (p_1, p_2, p_3) \rightarrow P - \theta = p_1\mathbf{a} + p_2\mathbf{b} + p_3\mathbf{c}$$

$$P = \theta + p_1\mathbf{a} + p_2\mathbf{b} + p_3\mathbf{c}$$

Reminder: The homogeneous representation of points and vectors

$$\mathbf{v} = v_1\mathbf{a} + v_2\mathbf{b} + v_3\mathbf{c} \rightarrow \mathbf{v} = (\mathbf{a}, \mathbf{b}, \mathbf{c}, \theta)\begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ 0 \end{pmatrix}$$

$$P = \theta + p_1\mathbf{a} + p_2\mathbf{b} + p_3\mathbf{c} \rightarrow P = (\mathbf{a}, \mathbf{b}, \mathbf{c}, \theta)\begin{pmatrix} p_1 \\ p_2 \\ p_3 \\ 1 \end{pmatrix}$$

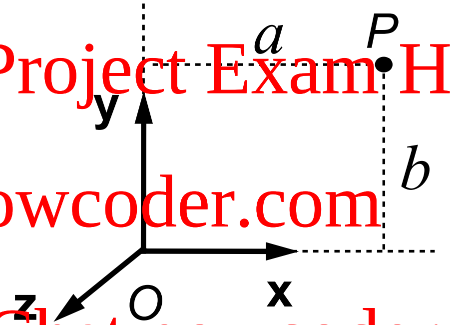


Transformation as a change of CS

Assume a coordinate system A and a point P

$P_A = (a, b, c, 1)_A \rightarrow$

$P_A = a\mathbf{x} + b\mathbf{y} + c\mathbf{z} + O$



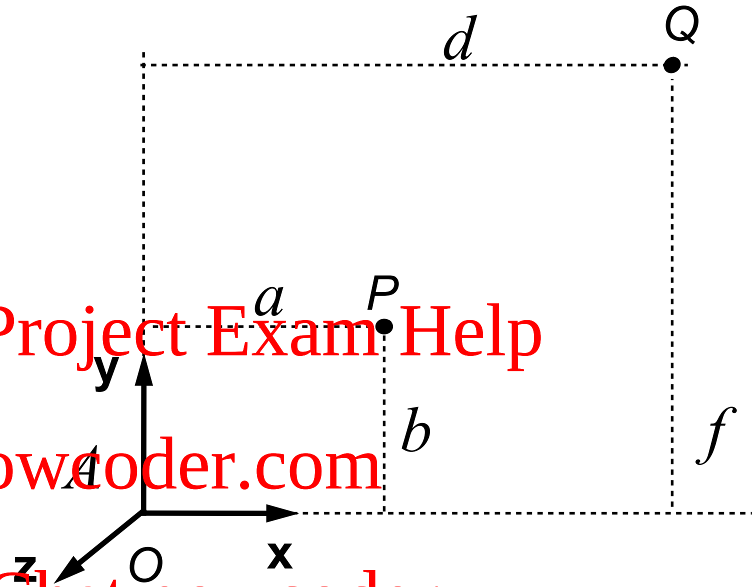
Transformation as a change of CS

$P_A = (a, b, c, 1)_A \rightarrow$

$P_A = a\mathbf{x} + b\mathbf{y} + c\mathbf{z} + O$

Transform with $\mathbf{M}$

$Q_A = \mathbf{M}P_A \rightarrow Q_A = (d, e, f, 1)_A$



Transformation as a change of CS

$P_A = (a, b, c, 1)_A \rightarrow$

$P_A = a\mathbf{x} + b\mathbf{y} + c\mathbf{z} + O$

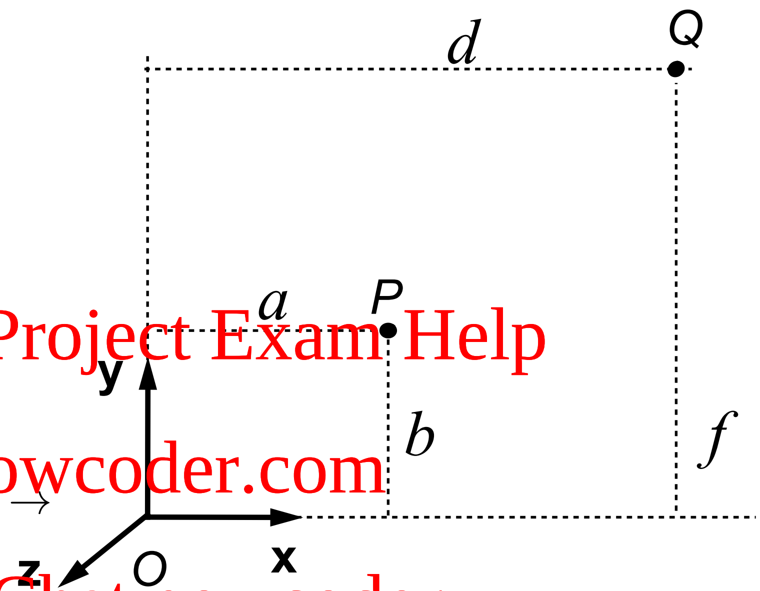
Transform with $\mathbf{M}$

$Q_A = \mathbf{M}P_A \rightarrow Q_A = (d, e, f, 1)_A$

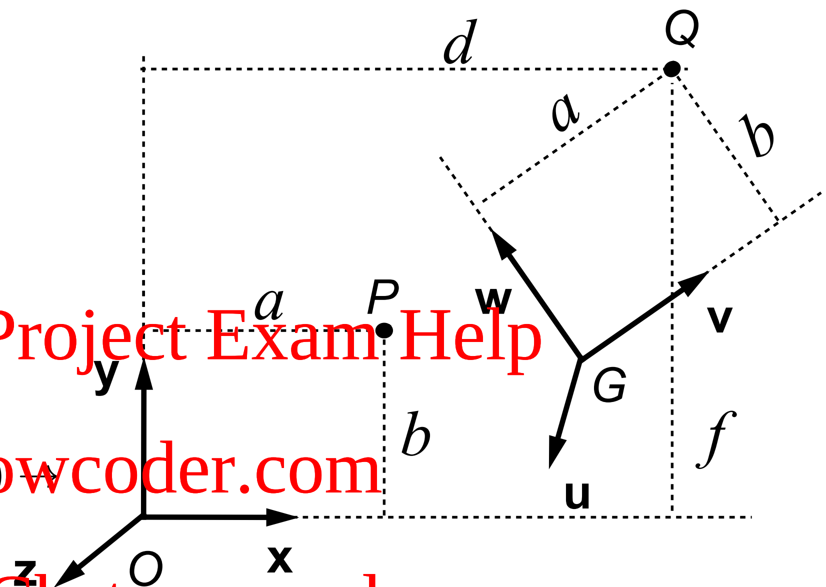
Using the full form of P_A

$Q_A = \mathbf{M}P_A = \mathbf{M}\left(a\mathbf{x}_A + b\mathbf{y}_A + c\mathbf{z}_A + O_A\right) \rightarrow$

$Q = a(\mathbf{Mx}) + b(\mathbf{My}) + c(\mathbf{Mz}) + \mathbf{M}O$



Transformation as a change of CS



$P_A = (a, b, c, 1)_A \rightarrow$

$P_A = a\mathbf{x} + b\mathbf{y} + c\mathbf{z} + O$

Transform with $\mathbf{M}$

$Q_A = \mathbf{M}P_A \rightarrow Q_A = (d, e, f, 1)_A$

Using the full form of P_A

$Q_A = \mathbf{M}P_A = \mathbf{M}\left(a\mathbf{x}_A + b\mathbf{y}_A + c\mathbf{z}_A + O_A\right) \rightarrow$

$Q = a(\mathbf{M}\mathbf{x}) + b(\mathbf{M}\mathbf{y}) + c(\mathbf{M}\mathbf{z}) + \mathbf{M}O$

We define coordinate system B :

$\mathbf{u} = \mathbf{M}\mathbf{x}, \quad \mathbf{v} = \mathbf{M}\mathbf{y}, \quad \mathbf{w} = \mathbf{M}\mathbf{z}, \quad G = \mathbf{M}O$

Which means:

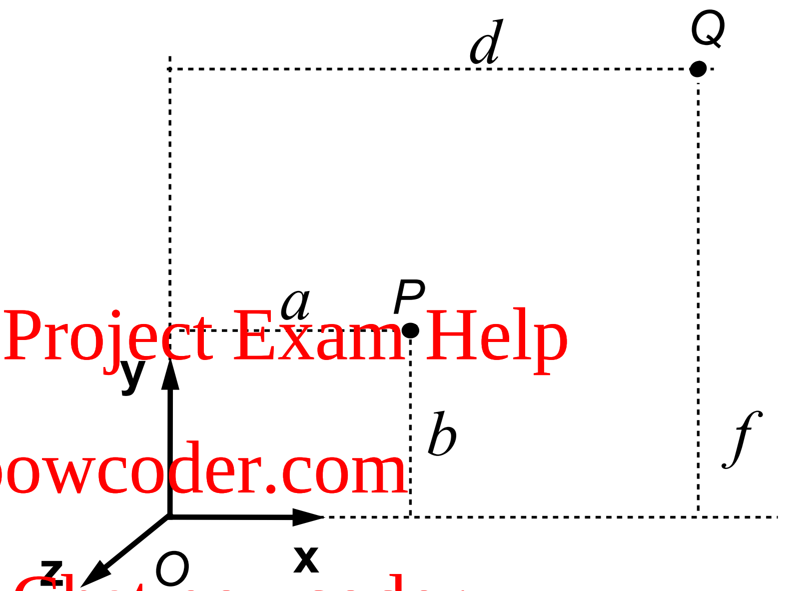
$Q = a\mathbf{u} + b\mathbf{v} + c\mathbf{w} + G$

Notice that by definition:

$Q_B = (a, b, c, 1)_B$

Transformation as a change of CS

So interpretation one:



$$P_A = (a, b, c, 1)_A \rightarrow$$

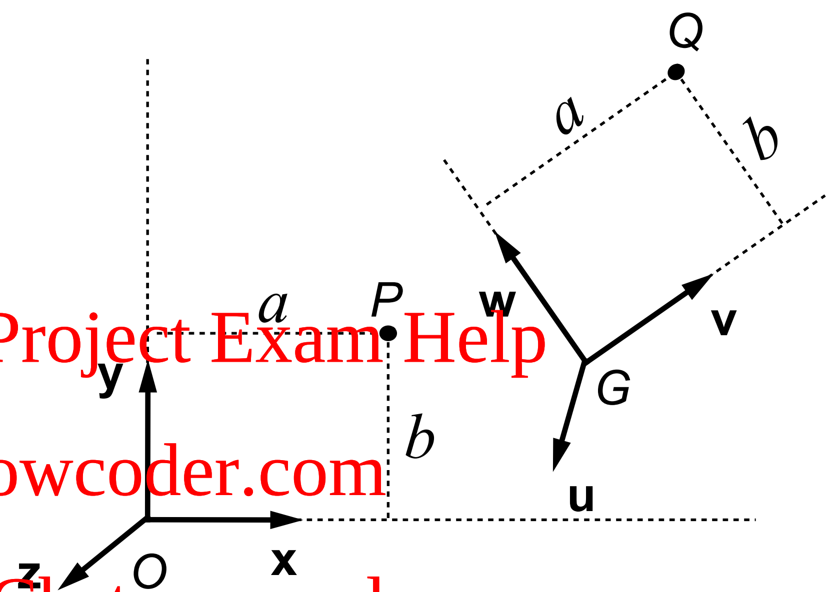
$$P_A = a\mathbf{x} + b\mathbf{y} + c\mathbf{z} + O$$

$$\text{Transform with } \mathbf{M}$$

$$Q_A = \mathbf{M}P_A \rightarrow Q_A = (d, e, f, 1)_A$$

Transformation as a change of CS

So interpretation two:



$$P_A = (a, b, c, 1)_A \rightarrow$$
$$P = a\mathbf{x} + b\mathbf{y} + c\mathbf{z} + O$$

Transform CS A with $\mathbf{M}$ into CS B

$$\mathbf{u} = \mathbf{M}\mathbf{x}, \ \ \mathbf{v} = \mathbf{M}\mathbf{y}, \ \ \mathbf{w} = \mathbf{M}\mathbf{z}, \ \ G = \mathbf{M}O$$

The point maintains its coordinates but with respect to the new CS B

$$Q_B = (a, b, c, 1)_B$$

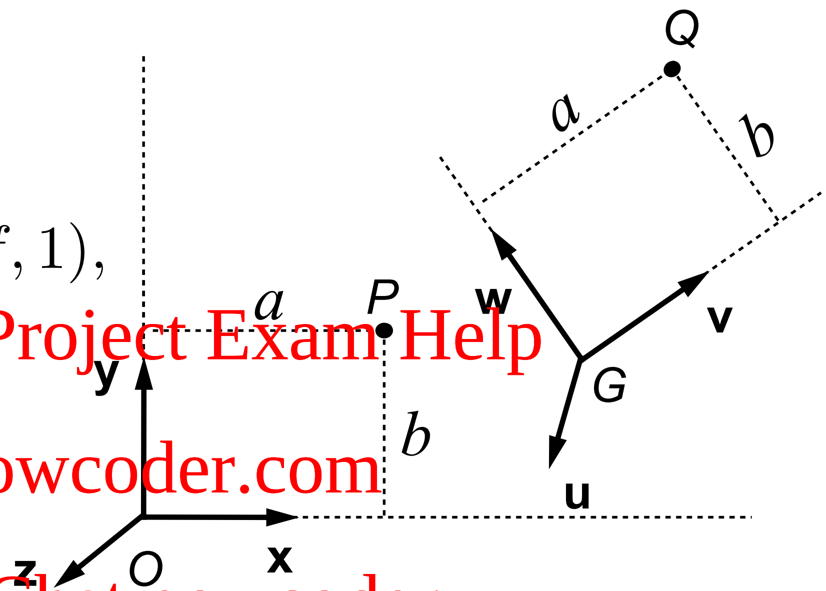
In other words the point is fixed with respect to the moving CS

Transformation as a change of CS

So we have:

$P_A = (a, b, c, 1), \quad Q_A = (d, e, f, 1),$

$Q_B = (a, b, c, 1), \quad Q_A = \mathbf{M} P_A$



Which means

$Q_A = \mathbf{M} P_A \rightarrow Q_A = \mathbf{M} Q_B$

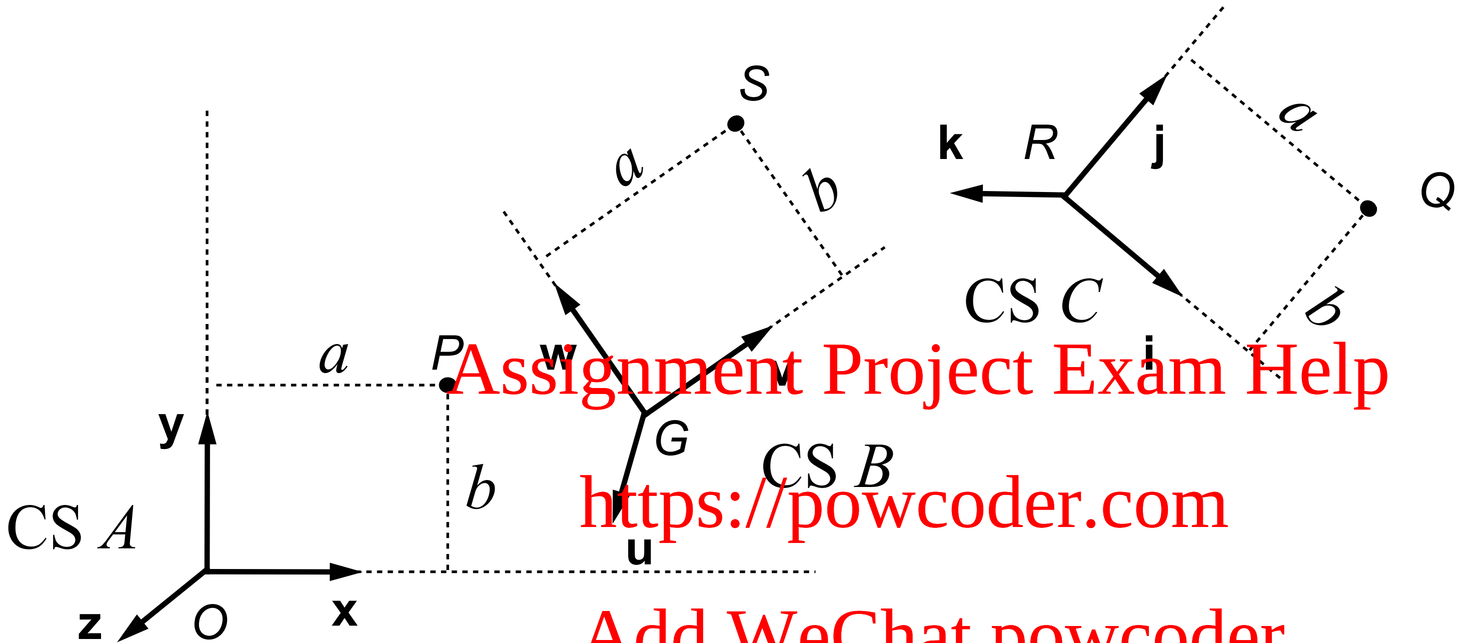
Let's show explicitly the coordinates of each side of the matrix

$Q_A = {}_A\mathbf{M}_B Q_B$

Remember, the same matrix transforms CS A into CS B, e.g.

$\mathbf{u}_A = {}_A\mathbf{M}_B \mathbf{x}_A$

Transformation as a change of CS



Fixed point: $Q = (a, b, c, 1)$

We can repeat the process for system B and C ignoring A

$CS_C = T(CS_B)$: $\mathbf{i}_B = {}_B\mathbf{M}_C \mathbf{u}_B$, $\mathbf{j}_B = {}_B\mathbf{M}_C \mathbf{u}_j$ etc

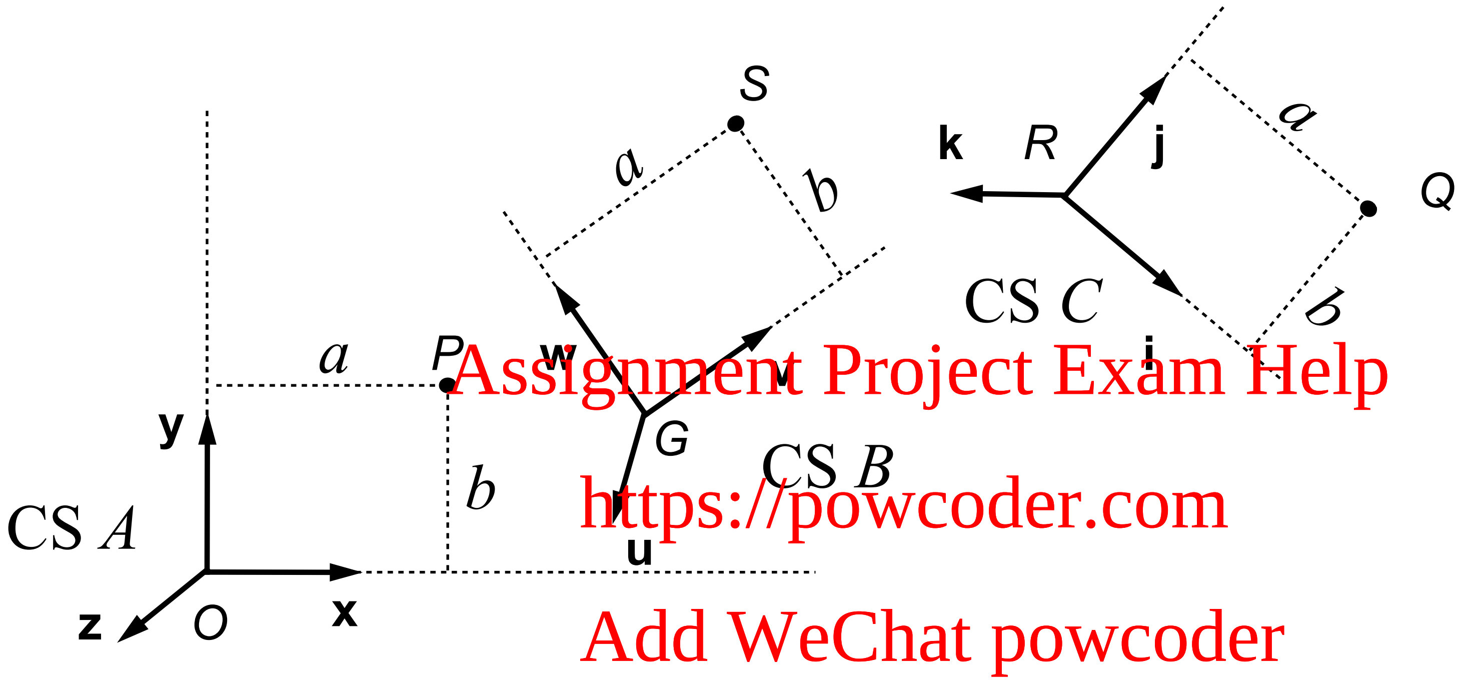
and

$Q_B = {}_B\mathbf{M}_C Q_C$

Then chain them all together:

$Q_A = {}_A\mathbf{M}_B {}_B\mathbf{M}_C Q_C$

Chain of CS

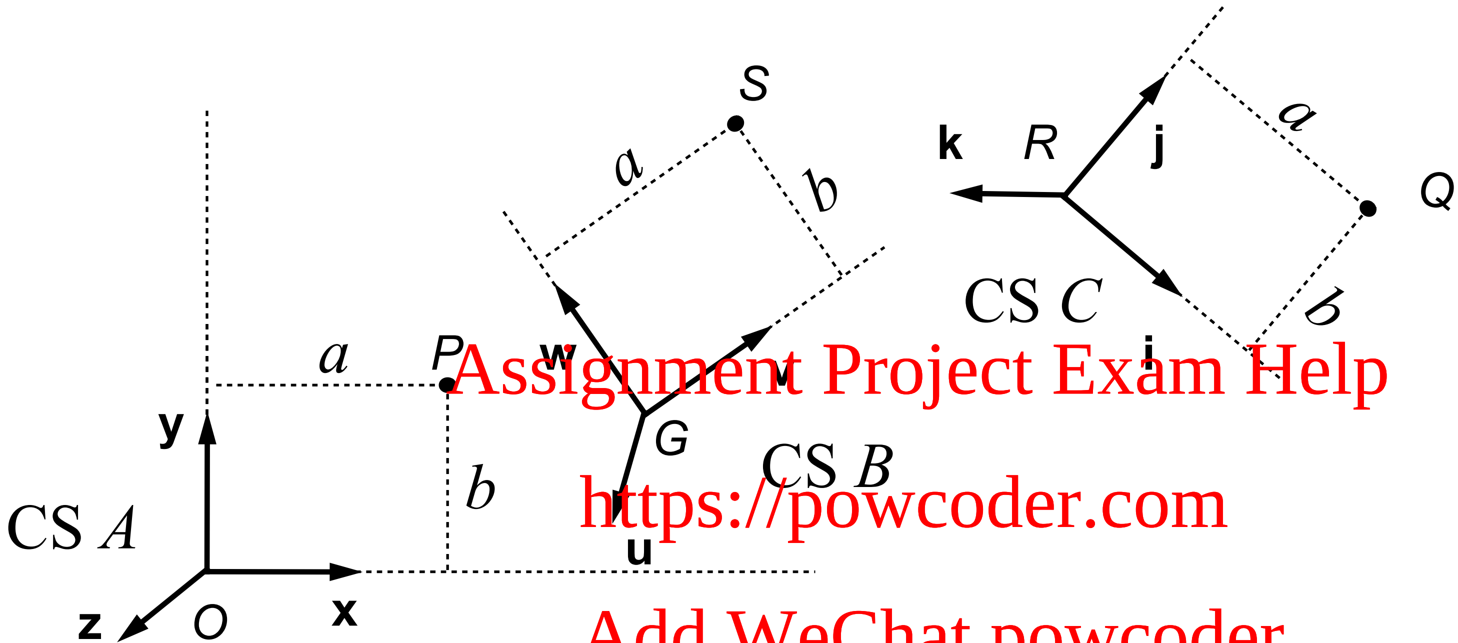


Chain or hierarchy of CS (frames): $A \to B \to C$

Represented by the matrix relationships:

$$Q_B = {}_B\mathbf{M}_C Q_C, \;\; Q_A = {}_A\mathbf{M}_B Q_B, \;\; Q_A = {}_A\mathbf{M}_B {}_B\mathbf{M}_C Q_C$$

Chain can be reformulated



Chain or hierarchy of CS (frames): $A \rightarrow B \rightarrow C$

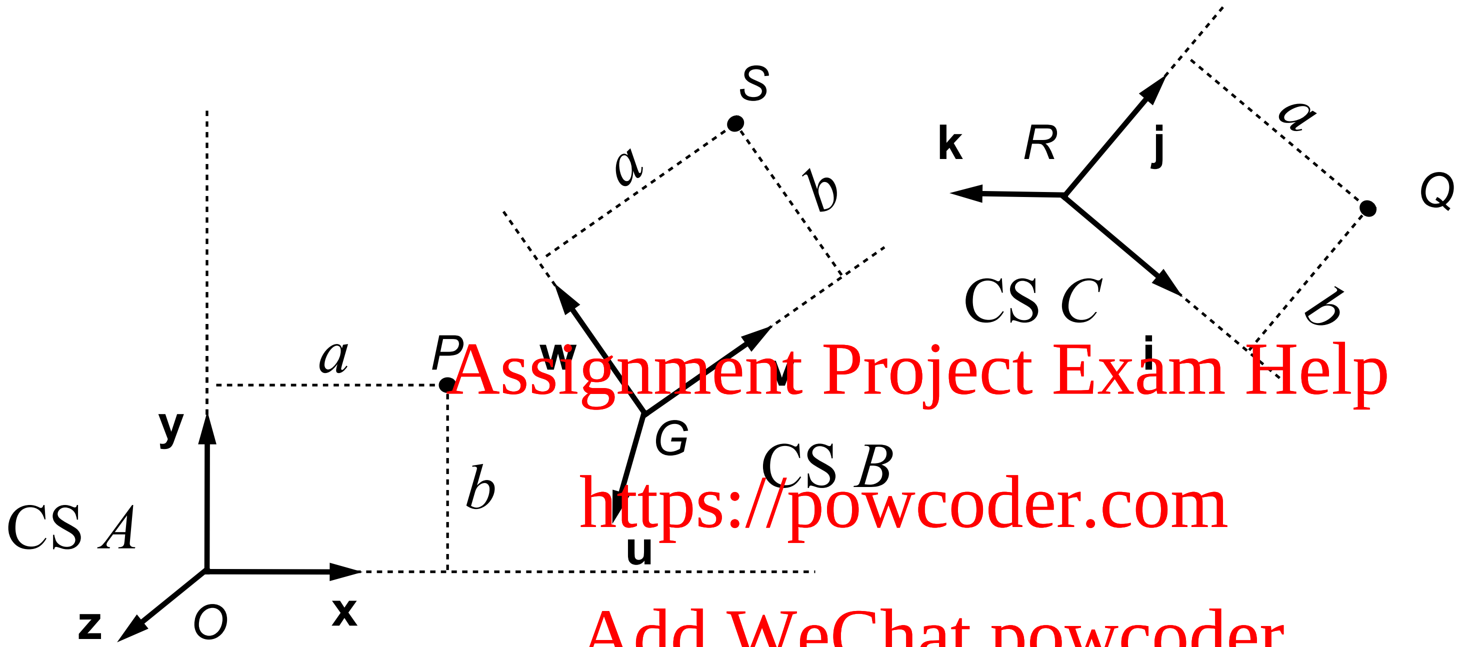
Represented by the matrix relationships:

$$Q_B = {}_B\mathbf{M}_C Q_C, \;\; Q_A = {}_A\mathbf{M}_B Q_B, \;\; Q_A = {}_A\mathbf{M}_B {}_B\mathbf{M}_C Q_C$$

Reformulate chain $B \rightarrow A \rightarrow C$

Represented by the matrix relationships:... ?

Chain can be reformulated



Chain or hierarchy of CS (frames): $A \to B \to C$

$Q_B = {}_B\mathbf{M}_C Q_C, \;\; Q_A = {}_A\mathbf{M}_B Q_B, \;\; Q_A = {}_A\mathbf{M}_B {}_B\mathbf{M}_C Q_C$

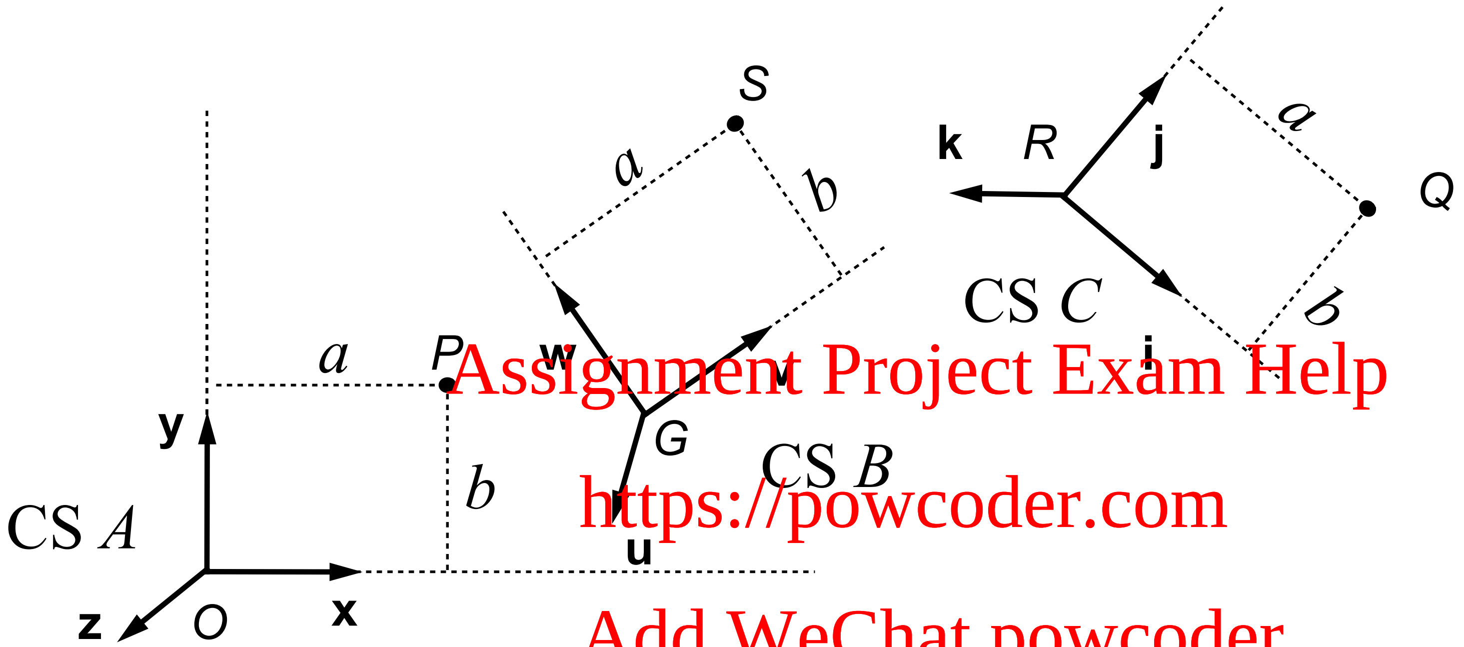
Reformulate chain $B \to A \to C$

$Q_B = {}_B\mathbf{M}_A Q_A, \;\; Q_A = {}_A\mathbf{M}_C Q_C, \;\; Q_B = {}_B\mathbf{M}_A {}_A\mathbf{M}_C Q_C$

remember: Non-trivial affine transformations can be inverted

${}_B\mathbf{M}_A = ({}_A\mathbf{M}_B)^{-1}, {}_A\mathbf{M}_C = {}_A\mathbf{M}_B {}_B\mathbf{M}_C$

Exercise



Chain: $A \rightarrow B \rightarrow C$

$Q_B = {}_B\mathbf{M}_C Q_C, \;\; Q_A = {}_A\mathbf{M}_B Q_B, \;\; Q_A = {}_A\mathbf{M}_B\, {}_B\mathbf{M}_C Q_C$

Chain $B \rightarrow A \rightarrow C$

$Q_B = {}_B\mathbf{M}_A Q_A, \;\; Q_A = {}_A\mathbf{M}_C Q_C, \;\; Q_B = {}_B\mathbf{M}_A\, {}_A\mathbf{M}_C Q_C$

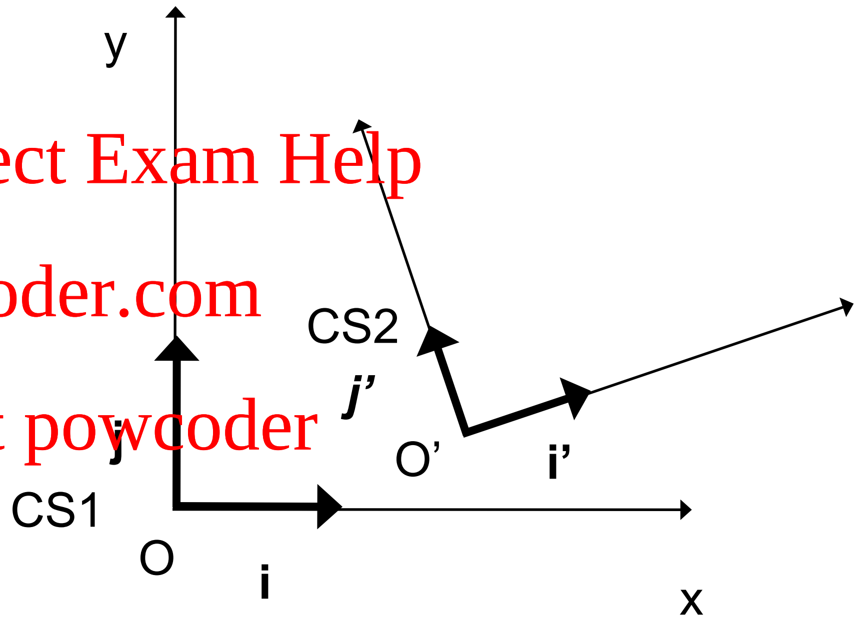
${}_B\mathbf{M}_A = ({}_A\mathbf{M}_B)^{-1}, {}_A\mathbf{M}_C = {}_A\mathbf{M}_B\, {}_B\mathbf{M}_C$

What is ${}_C\mathbf{M}_A$?

More details and derivations

Transforming CS1 into CS2

- Using an affine matrix M



Transforming CS1 into CS2

- Using an affine matrix M

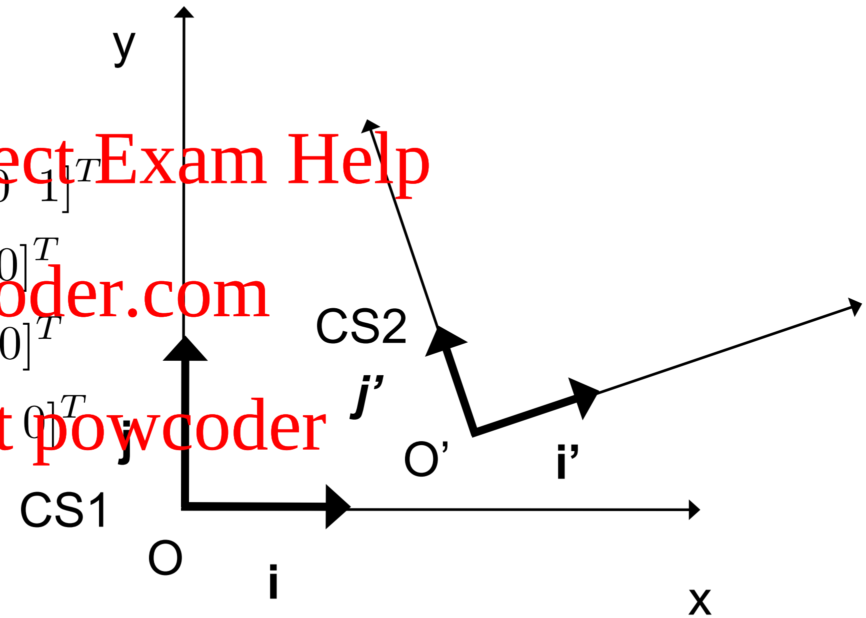
Transform each element of the frame:

$$O'_{CS1} = T(O_{CS1}) = T((0,0,0,1)) = \mathbf{M}[0\ \ 0\ \ 0\ \ 1]^T$$

$$\mathbf{i}'_{CS1} = T(\mathbf{i}_{CS1}) = T((1,0,0,0)) = \mathbf{M}[1\ \ 0\ \ 0\ \ 0]^T$$

$$\mathbf{j}'_{CS1} = T(\mathbf{j}_{CS1}) = T((0,1,0,0)) = \mathbf{M}[0\ \ 1\ \ 0\ \ 0]^T$$

$$\mathbf{k}'_{CS1} = T(\mathbf{k}_{CS1}) = T((0,0,1,0)) = \mathbf{M}[0\ \ 0\ \ 1\ \ 0]^T$$



The following hold by definition:

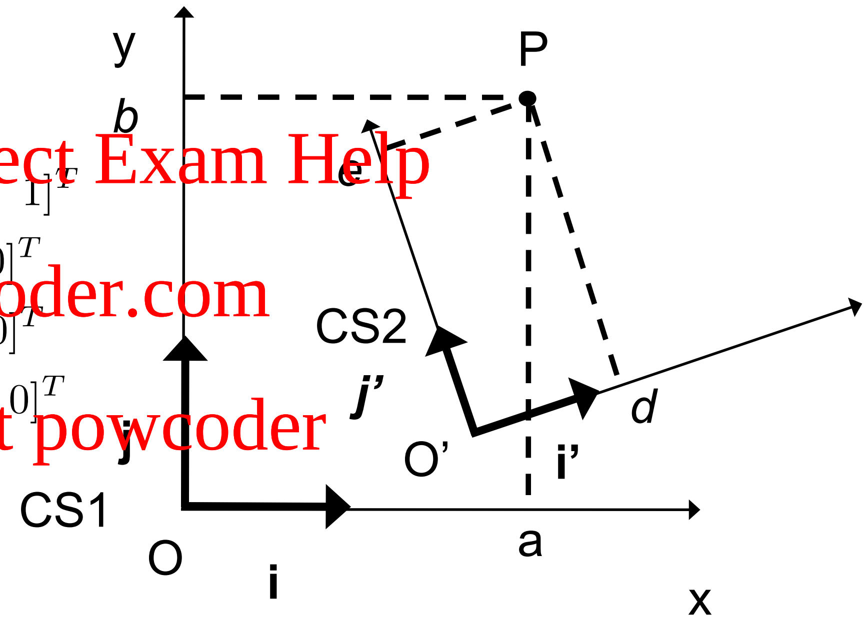
$$O'_{CS2} = (0,0,0,1)_{CS2}, \quad \mathbf{i}'_{CS2} = (1,0,0,0)_{CS1} \cdots$$

Also remember that:

$$\mathbf{i}_{CS1} = (0,0,0,0)_{CS1} \ \text{ means } \ \mathbf{i}_{CS1} = 1\mathbf{i}_{CS1} + 0\mathbf{j}_{CS1} + 0\mathbf{jK}_{CS1} + 0O_{CS1}$$

Transforming CS1 into CS2

- Using an affine matrix M



Transform each element of the frame:

$$O'_{CS1} = T(O_{CS1}) = T((0,0,0,1)) = \mathbf{M}[0\ \ 0\ \ 0\ \ 1]^T$$

$$\mathbf{i}'_{CS1} = T(\mathbf{i}_{CS1}) = T((1,0,0,0)) = \mathbf{M}[1\ \ 0\ \ 0\ \ 0]^T$$

$$\mathbf{j}'_{CS1} = T(\mathbf{j}_{CS1}) = T((0,1,0,0)) = \mathbf{M}[0\ \ 1\ \ 0\ \ 0]^T$$

$$\mathbf{k}'_{CS1} = T(\mathbf{k}_{CS1}) = T((0,0,1,0)) = \mathbf{M}[0\ \ 0\ \ 1\ \ 0]^T$$

The following hold by definition:

$O'_{CS2} = (0,0,0,1), \mathbf{i}'_{CS2} = (1,0,0,0) \cdots$

The following hold by definition for a point P in the space:

$$P_{CS1} = (a,b,c,1)_{CS1} = a\mathbf{i}_{CS1} + b\mathbf{j}_{CS1} + c\mathbf{k}_{CS1} + O_{CS1}$$

$$P_{CS2} = (d,e,f,1)_{CS2} = d\mathbf{i}'_{CS2} + e\mathbf{j}'_{CS2} + f\mathbf{k}'_{CS2} + O'_{CS2}$$

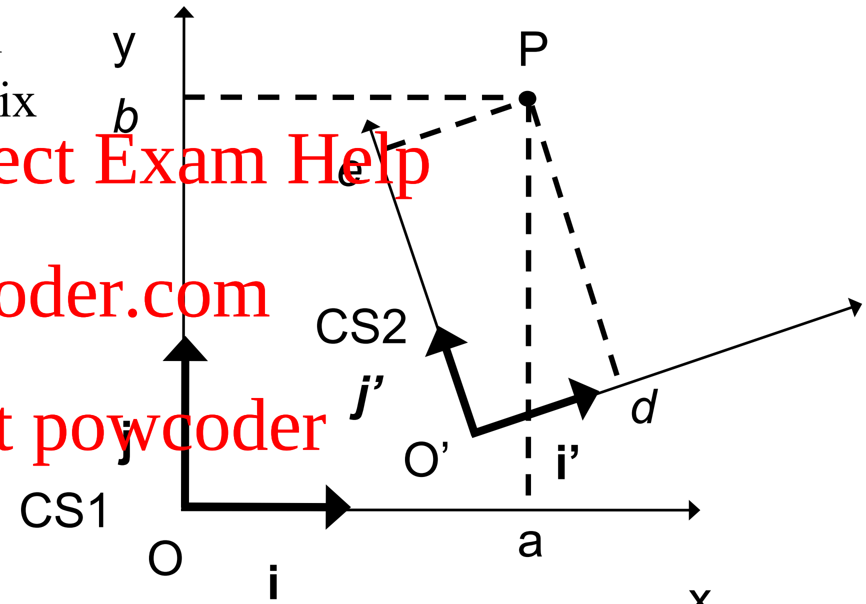
$$P_{CS1} = (a,b,c,1)_{CS1} = d\mathbf{i}'_{CS1} + e\mathbf{j}'_{CS1} + f\mathbf{k}'_{CS1} + O'_{CS1}$$

$$P_{CS2} = (d,e,f,1)_{CS2} = a\mathbf{i}_{CS2} + b\mathbf{j}_{CS2} + c\mathbf{k}_{CS2} + O_{CS2}$$

Both systems are equivalent.
There is nothing special about CS1

Transforming CS1 into CS2

- What is the relationship between P in CS2 and P in CS1 if CS2 = T(CS1) and T is an affine transformation represented by matrix M?



Notation and what we know:

$$P_{CS1} = (a, b, c, 1)^T$$

$$P_{CS2} = (d, e, f, 1)^T$$

$$O' = T(O), \quad \mathbf{i}' = T(\mathbf{i}), \quad \mathbf{j}' = T(\mathbf{j}), \quad \mathbf{k}' = T(\mathbf{k})$$

Reminder:

$$O'_{CS2} = (0, 0, 0, 1), \quad O'_{CS1} = T((0, 0, 0, 1)) = M\mathbf{0}$$

$$\mathbf{i}'_{CS2} = (1, 0, 0, 0), \quad \mathbf{i}'_{CS1} = T((1, 0, 0, 0))$$

Derivation

In any frame $P = d\mathbf{i}' + e\mathbf{j}' + f\mathbf{k}' + O'$ so in CS1 $P_{CS1} = d\mathbf{i}'_{CS1} + e\mathbf{j}'_{CS1} + f\mathbf{k}'_{CS1} + O'_{CS1}$

We know that $(\mathbf{i}, \mathbf{j}', \mathbf{k}', O') = T((\mathbf{i}, \mathbf{j}, \mathbf{k}, O))$

All vectors below are in frame CS1 so for example $\mathbf{i} = (1, 0, 0, 0)$

$$P_{CS1} = dT(\mathbf{i}) + eT(\mathbf{j}) + fT(\mathbf{k}) + T(O) = d(\mathbf{Mi}) + e(\mathbf{Mj}) + f(\mathbf{Mk}) + \mathbf{M}O$$

$$= d(\mathbf{M}\begin{bmatrix}1\\0\\0\\0\end{bmatrix}) + e(\mathbf{M}\begin{bmatrix}0\\1\\0\\0\end{bmatrix}) + f(\mathbf{M}\begin{bmatrix}0\\0\\1\\0\end{bmatrix}) + \mathbf{M}\begin{bmatrix}0\\0\\0\\1\end{bmatrix}$$

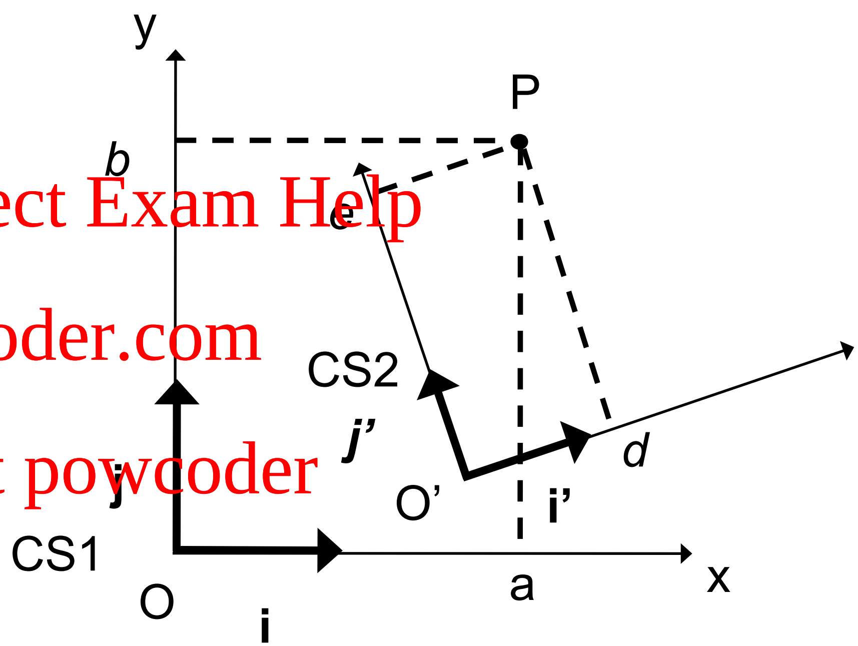
$$= \mathbf{M}\begin{bmatrix}d\\0\\0\\0\end{bmatrix} + \mathbf{M}\begin{bmatrix}0\\e\\0\\0\end{bmatrix} + \mathbf{M}\begin{bmatrix}0\\0\\f\\0\end{bmatrix} + \mathbf{M}\begin{bmatrix}0\\0\\0\\1\end{bmatrix}$$

$$= \mathbf{M}(\begin{bmatrix}d\\0\\0\\0\end{bmatrix} + \begin{bmatrix}0\\e\\0\\0\end{bmatrix} + \begin{bmatrix}0\\0\\f\\0\end{bmatrix} + \begin{bmatrix}0\\0\\0\\1\end{bmatrix}) = \mathbf{M}\begin{bmatrix}d\\e\\f\\1\end{bmatrix}$$

P in CS1 vs P in CS2

$$P_{CS1} = \mathrm{M} P_{CS2}$$

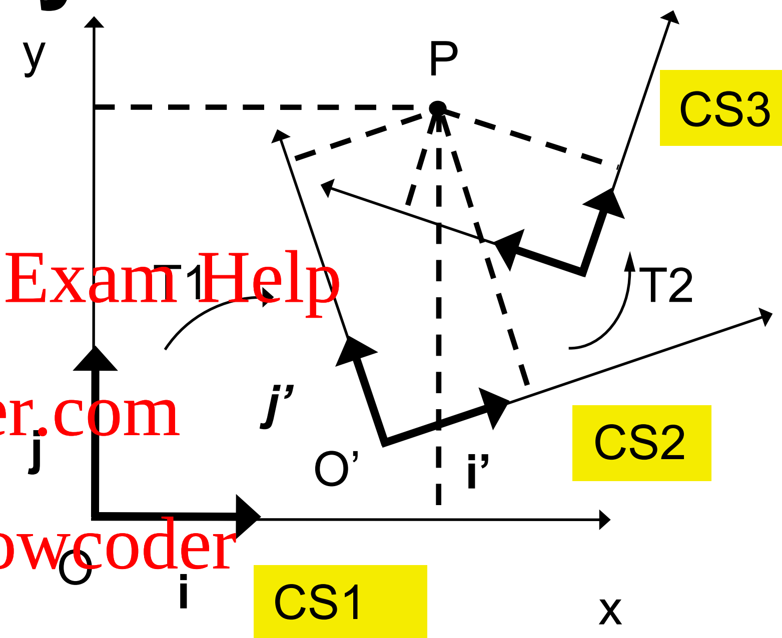
$$\begin{pmatrix} a \\ b \\ c \\ 1 \end{pmatrix} = \mathrm{M} \begin{pmatrix} d \\ e \\ f \\ 1 \end{pmatrix}$$



Successive transformations of the Coordinate System

CS1 → CS2 → CS3

T1 T2



Working backwards:

$$P_{CS2} = M_2 P_{CS3} \rightarrow \begin{pmatrix} d \\ e \\ f \\ 1 \end{pmatrix} = M_2 \begin{pmatrix} g \\ h \\ m \\ 1 \end{pmatrix}$$

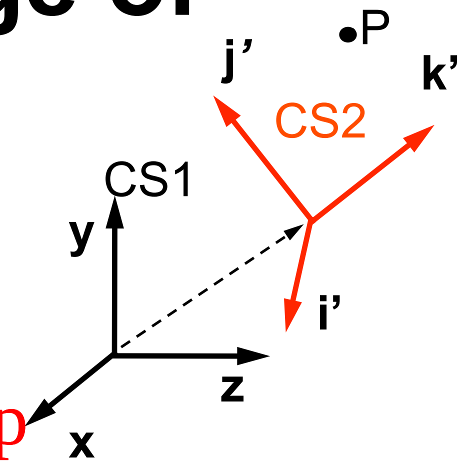
$$P_{CS1} = M_1 P_{CS2} \rightarrow \begin{pmatrix} a \\ b \\ c \\ 1 \end{pmatrix} = M_1 \begin{pmatrix} d \\ e \\ f \\ 1 \end{pmatrix} = M_1 M_2 \begin{pmatrix} g \\ h \\ m \\ 1 \end{pmatrix}$$

Transformations as a change of basis

Another way of approaching the issue of relating two coordinate systems

Similar to the previous one but from a different point of view

Transformations as a change of basis



We know the basis of CS2 with respect to CS1 i.e.:

$$\mathbf{i}'_{CS1} = (i'_x, i'_y, i'_z)$$

$$\mathbf{j}'_{CS1} = (j'_x, j'_y, j'_z)$$

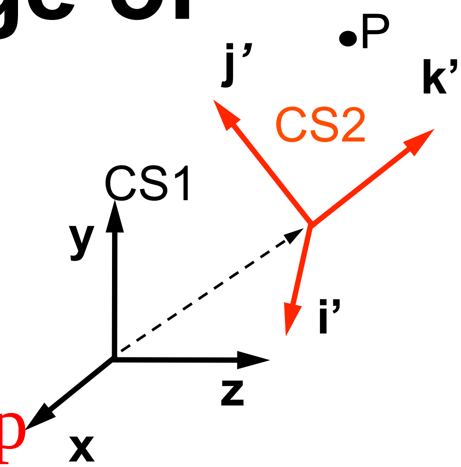
$$\mathbf{k}'_{CS1} = (k'_x, k'_y, k'_z)$$

$$\mathbf{O}'_{CS1} = (O'_x, O'_y, O'_z)$$

Can we find the matrix *M* that transforms points from CS2 to CS1?

$$P_{CS1} = MP_{CS2}$$

Transformations as a change of basis



We know the basis vectors and we know that

$$P_{CS1} = MP_{CS2}$$

What is M with respect to the basis vectors?

$$P_{CS2} = a\mathbf{i}'_{CS2} + b\mathbf{j}'_{CS2} + c\mathbf{k}'_{CS2} + O'_{CS2} = a\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + b\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} + c\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$P_{CS1} = a\mathbf{i}'_{CS1} + b\mathbf{j}'_{CS1} + c\mathbf{k}'_{CS1} + O'_{CS1} = a\begin{bmatrix} i'_x \\ i'_y \\ i'_z \end{bmatrix} + b\begin{bmatrix} j'_x \\ j'_y \\ j'_z \end{bmatrix} + c\begin{bmatrix} k'_x \\ k'_y \\ k'_z \end{bmatrix} + \begin{bmatrix} O'_x \\ O'_y \\ O'_z \end{bmatrix}$$

$$P_{CS1} = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} i'_x & j'_x & k'_x & O'_x \\ i'_y & j'_y & k'_y & O'_y \\ i'_z & j'_z & k'_z & O'_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ 1 \end{bmatrix} = MP_{CS2}$$

Transformations as a change of basis

- Note that this is actually the matrix that transforms CS1 into CS2 with respect to CS1

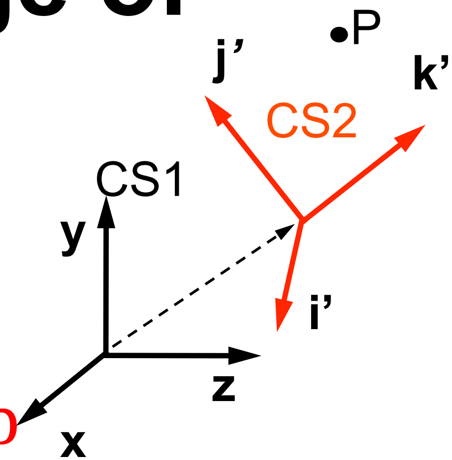
- Sanity check:

$$M\mathbf{x}_{CS1} = \begin{bmatrix} i'_x & j'_x & k'_x & O'_x \\ i'_y & j'_y & k'_y & O'_y \\ i'_z & j'_z & k'_z & O'_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} i'_x \\ i'_y \\ i'_z \\ 0 \end{bmatrix} = \mathbf{i}'_{CS1}$$

Similarly

$$M\mathbf{y}_{CS1} = \mathbf{j}'_{CS1}, \quad M\mathbf{z}_{CS1} = \mathbf{k}'_{CS1}, \quad MO_{CS1} = O'_{CS1}$$

Transformations as a change of basis



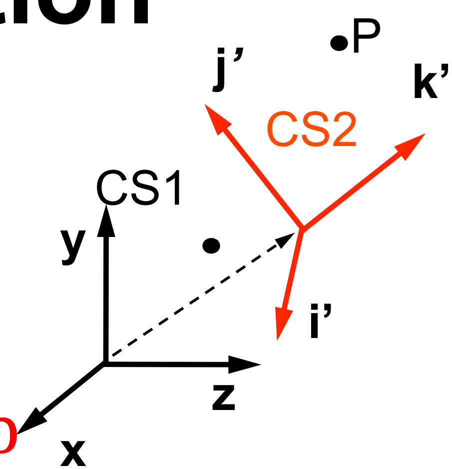
$$P_{CS1} = MP_{CS2}$$

$$P_{CS1} = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} i'_x & j'_x & k'_x & O'_x \\ i'_y & j'_y & k'_y & O'_y \\ i'_z & j'_z & k'_z & O'_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ 1 \end{bmatrix} = MP_{CS2}$$

That is:

We can view transformations as a change of coordinate system

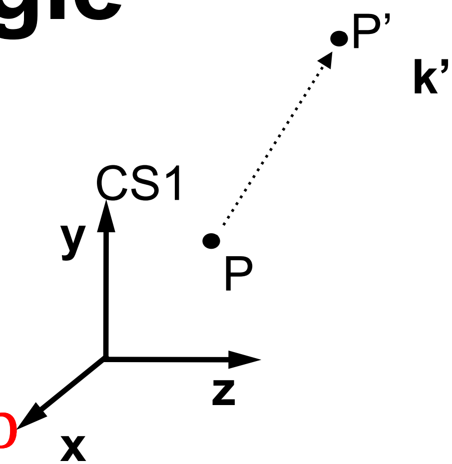
So really this matrix operation has two interpretations



Mathematically equivalent but conceptually different

$$P_{CS1} = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} i'_x & j'_x & k'_x & O'_x \\ i'_y & j'_y & k'_y & O'_y \\ i'_z & j'_z & k'_z & O'_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ 1 \end{bmatrix} = MP_{CS2}$$

A. Transformation in a single coordinate system

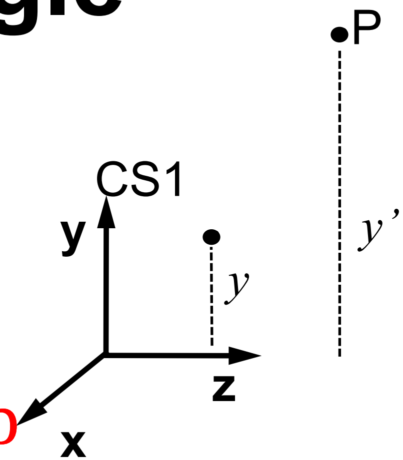


Ignore CS2:

- Point (x,y,z,1) in CS1 is transformed to point P'=(x',y',z',1) in CS1 by a transformation represented by *M*

$$P'_{CS1} = \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} i'_x & j'_x & k'_x & O'_x \\ i'_y & j'_y & k'_y & O'_y \\ i'_z & j'_z & k'_z & O'_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = M P_{CS1}$$

A. Transformation in a single coordinate system

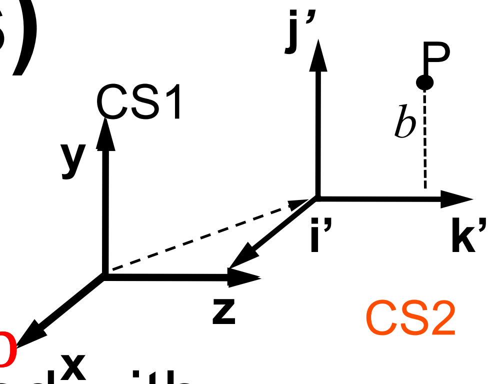


Ignore CS2:

- Point (x,y,z,1) in CS1 is transformed to point P=(x',y',z',1) in CS1 by a transformation represented by *M*
- The transformation happens wrt to CS1

$$P'_{CS1} = \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} i'_x & j'_x & k'_x & O'_x \\ i'_y & j'_y & k'_y & O'_y \\ i'_z & j'_z & k'_z & O'_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = MP_{CS1}$$

B. Transformation of Coordinate System (change of basis)

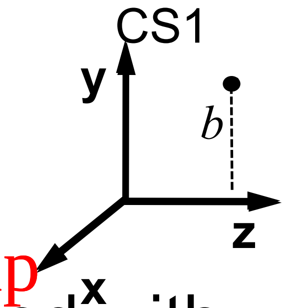


Interpretation two:

- CS1 is transformed to CS2 through a transformation and the point remains fixed with respect to CS2

$$P_{CS1} = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} i'_x & j'_x & k'_x & O'_x \\ i'_y & j'_y & k'_y & O'_y \\ i'_z & j'_z & k'_z & O'_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ 1 \end{bmatrix} = MP_{CS2}$$

B. Transformation of Coordinate System (change of basis)

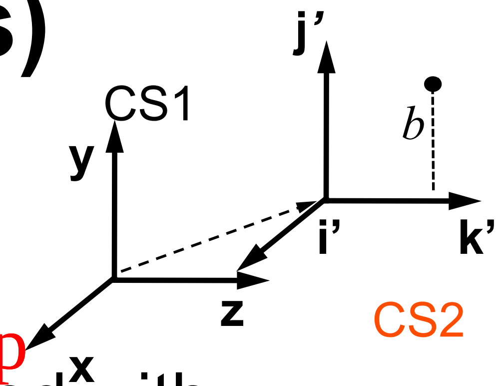


Interpretation two:

- CS1 is transformed to CS2 through a transformation and the point remains fixed with respect to CS2

$$P_{CS1} = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} i'_x & j'_x & k'_x & O'_x \\ i'_y & j'_y & k'_y & O'_y \\ i'_z & j'_z & k'_z & O'_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ 1 \end{bmatrix} = MP_{CS2}$$

B. Transformation of Coordinate System (change of basis)

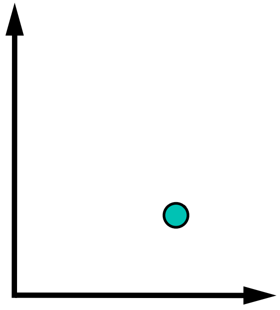


Interpretation two:

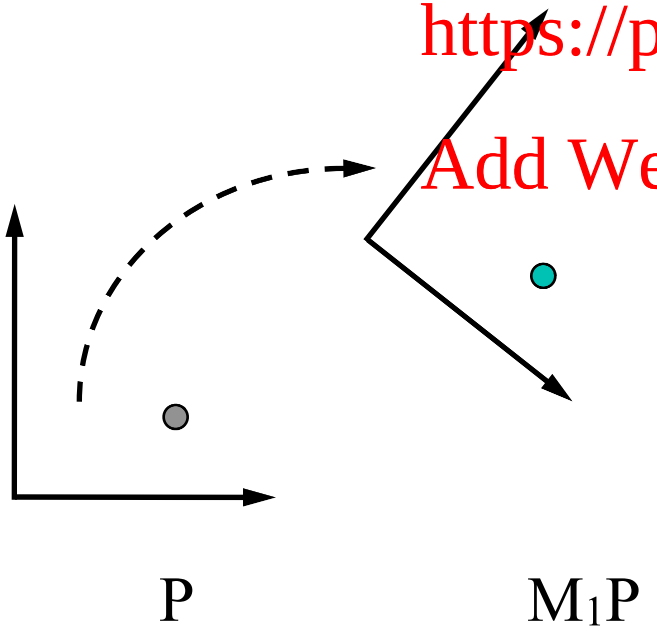
- CS1 is transformed to CS2 through a transformation and the point remains fixed with respect to CS2

$$P_{CS1} = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} i'_x & j'_x & k'_x & O'_x \\ i'_y & j'_y & k'_y & O'_y \\ i'_z & j'_z & k'_z & O'_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ 1 \end{bmatrix} = MP_{CS2}$$

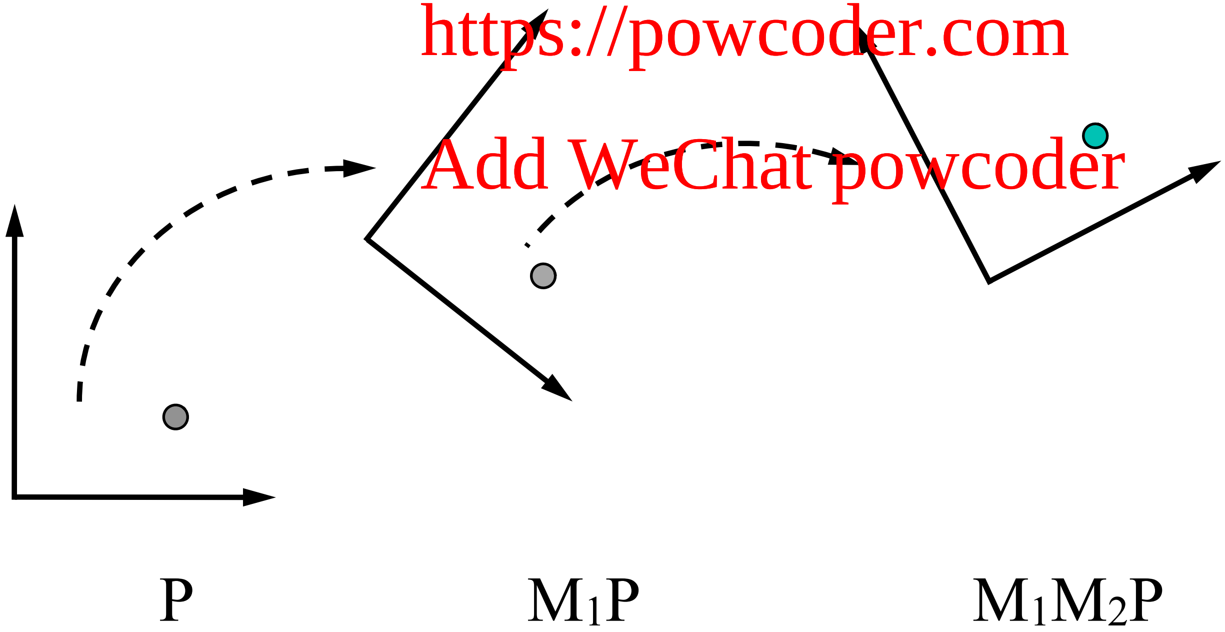
Transforming a point through transforming coordinate systems



Transforming a point through transforming coordinate systems



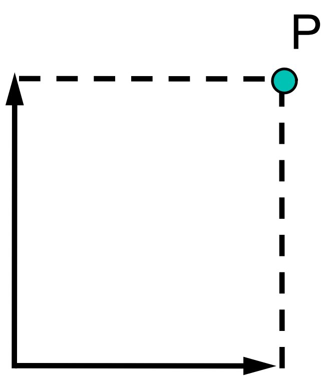
Transforming a point through transforming coordinate systems



Example

In 2D homogeneous coordinates

$P = [1,1,1]^T$

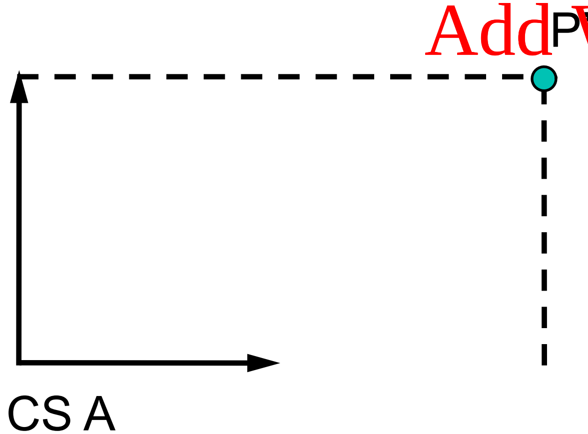


CS A

Example

Transformation T(1,0): M

$P' = M[1,1,1]^T = MP = [2,1,1]^T$



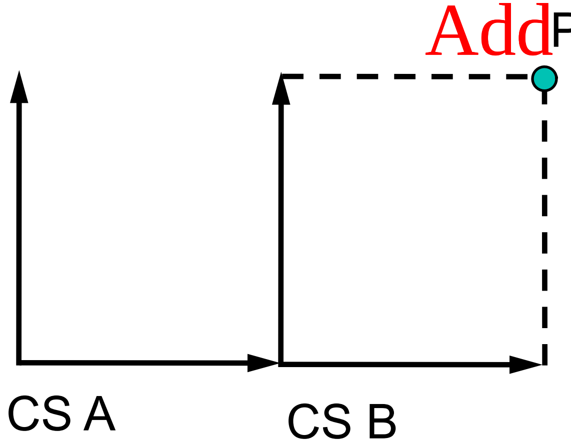
Example

Equivalently

Transformation T(1,0): M on CSA

$P_A = {}_A M_B P$

$P = [1,1,1]^T$



Conceptual difference: the local coordinates of P stay the same, the local system changes and becomes CSB.
In other words we transformed system A and P along with it.

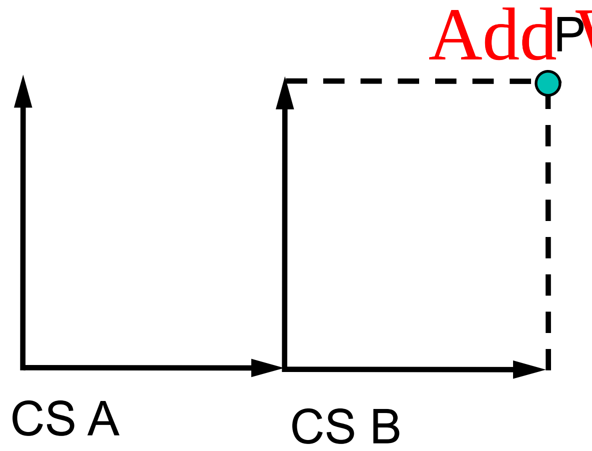
However the fixed coordinates of P are now in CSB

Example

Transformation T(1,0): M on CSA

$P_A = {}_A M_B P$

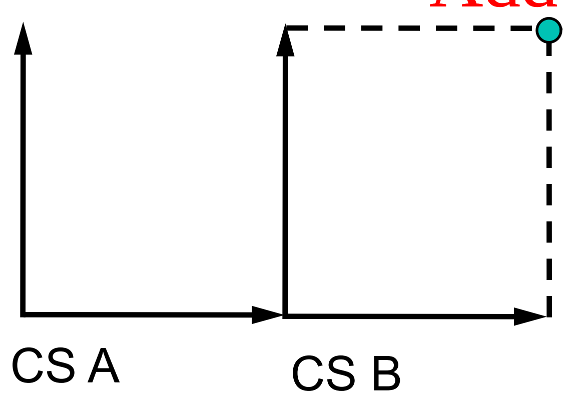
Next transformation?



Example

Two choices!

$P_A = {}_A M_B P$

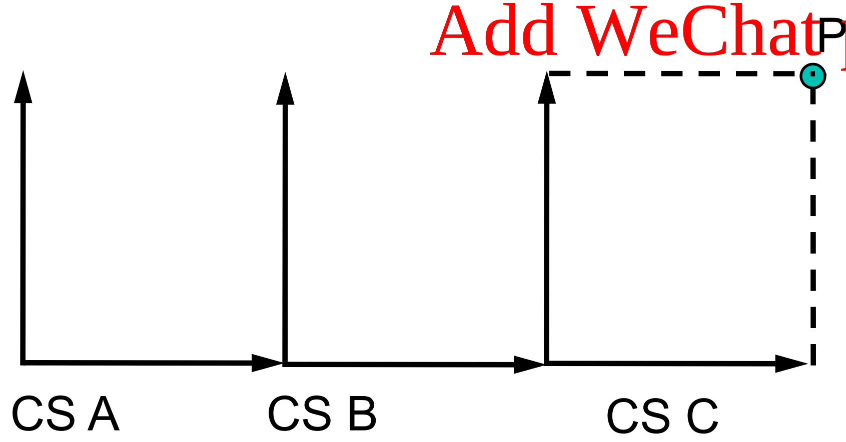


Example

After the last matrix T(1,0): ${}_BM_C$

This transformation now happens in CSB

$P_A = {}_AM_B\,{}_BM_C P$

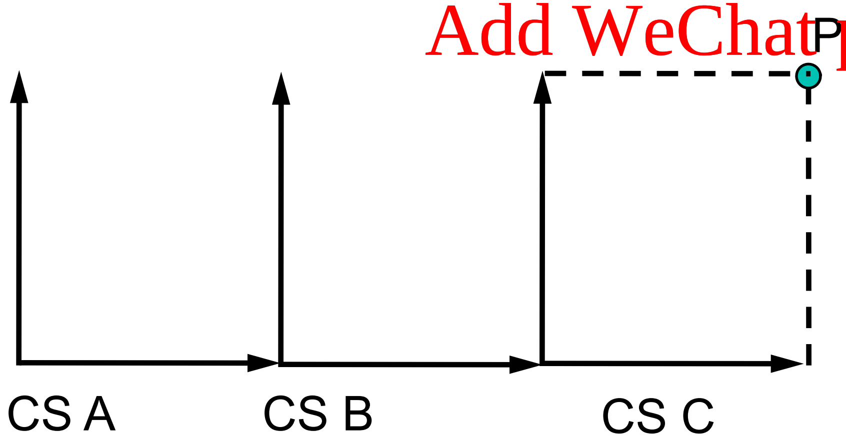


Hierarchy of systems

Example

We now have 3 systems we can work in

$P_A = {}_AM_B {}_BM_C P$



Hierarchy of systems

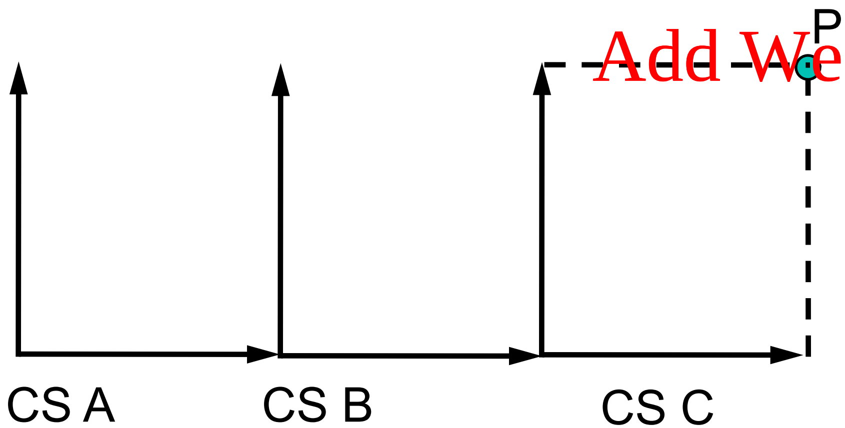
Example

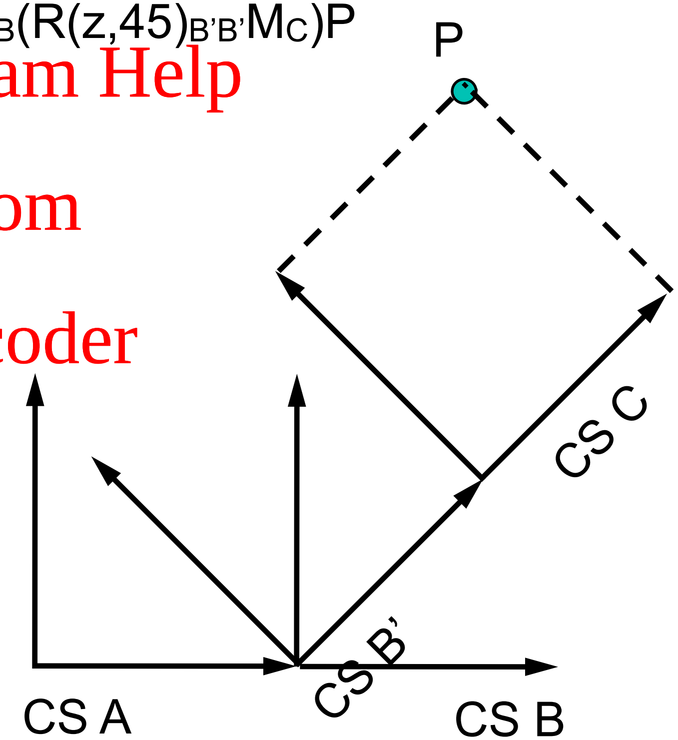
Let's rotate in CSB by R(z,45):

$P_A = {}_AM_B\,{}_BM_C P$

After the matrix R(z,45) is inserted

$P_A = {}_AM_B\,{}_B(R(z,45)_{B'}\,{}_{B'}M_C)P$



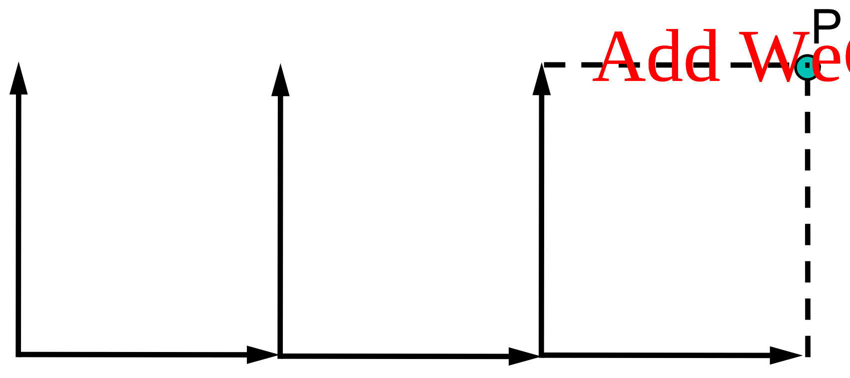


Hierarchy of systems

Example

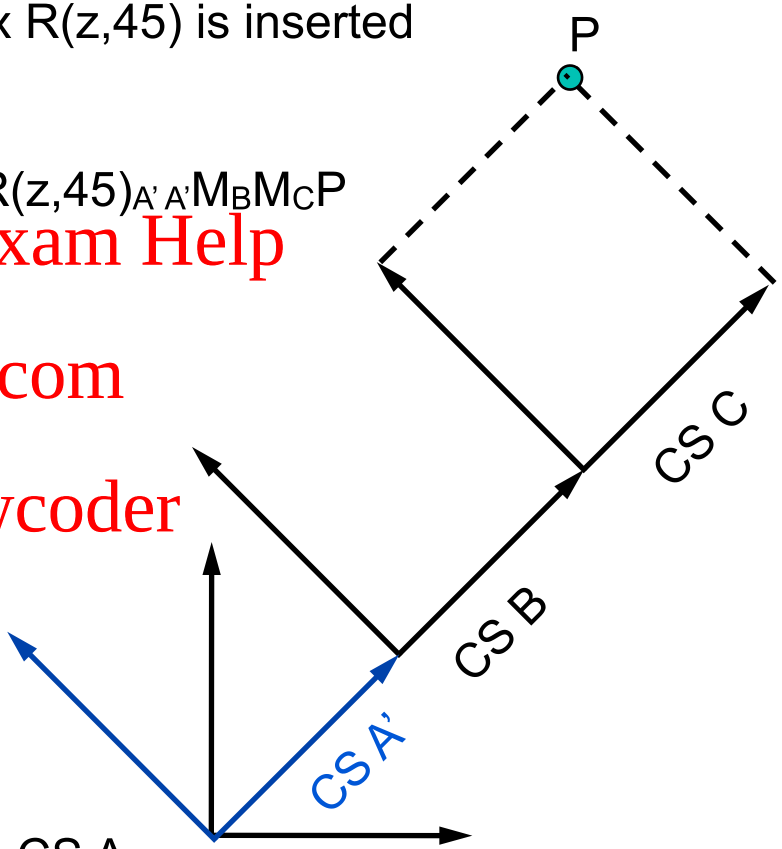
Same matrix here R(z,45):

$P_A = {}_AM_B {}_BM_C P$



After the matrix R(z,45) is inserted

$P_A = {}_AR(z,45)_{A'} {}_{A'}M_B M_C P$



Hierarchy of systems

Main point

Interpreting a transformation matrix

- $P_A = {}_AM_B\ P_B$

  transforms a point within system A, from its current location to a new one

- $P_A = {}_AM_B\ P_B$

  transforms system A into B. Right of matrix M we talk in B coordinates. Left of matrix M we talk in A coordinates

Rule of thumb

Transforming a point P:

Transformations: T1,T2,T3

Matrix: M = M3 x M2 x M1

Point transformed by: MP

Successive transformations happen with respect to the same CS

Transforming a CS

Transformations: T1, T2, T3

Matrix: M = M1 x M2 x M3

A point has original coordinates MP

Each transformations happens with respect to the new CS

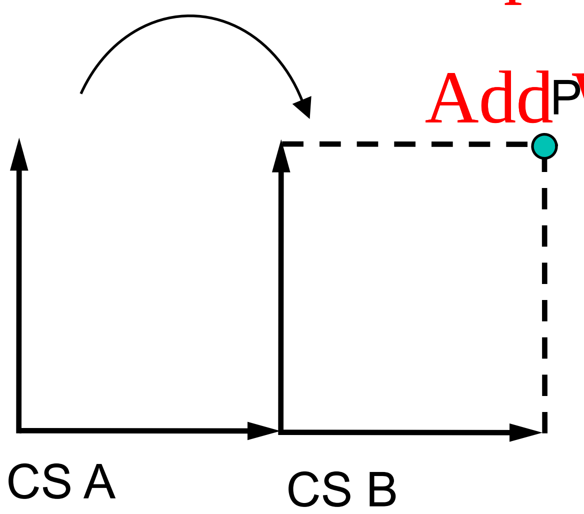
The **last** coordinate system (right most) represents the **first** transformation applied to the point

Rule of thumb

To find the transformation matrix that transforms P from CSB coordinates to CSA coordinates, we find the sequence of transformations that align CSA to CSB accumulating matrices from left to right.

Explanation of this rule

Transformation M: ${}_AM_B$



If we think transforming systems, M takes CS A from the left and produces B on the right.

$${}_A\overrightarrow{M}_B$$

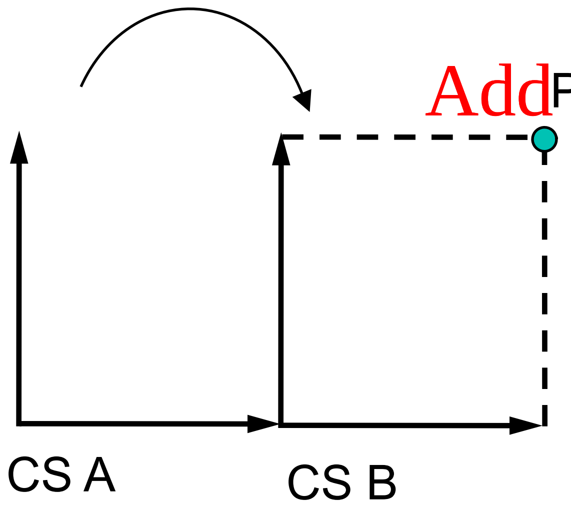
After this transformation we talk in B coordinates (right side).

If we think about points then we move the other way. M takes B on the right and produces the A coordinates on the left:

$${}_A\overleftarrow{M}_B$$

Explanation of this rule

Transformation M: $_{A}M_{B}$



Take this simple example where to produce B we translate A by 1 on x axis.

$P_B = (1,1)$ $P_A = (2,1)$

If we move A by +1 to transform it into B then the coordinates of P with respect to the new system are shortened by 1 (B is closer to P than A by 1). So if we want to transform the coordinates of P from B to A we need to add 1 in x. Exactly what we need to do to transform system A to B.