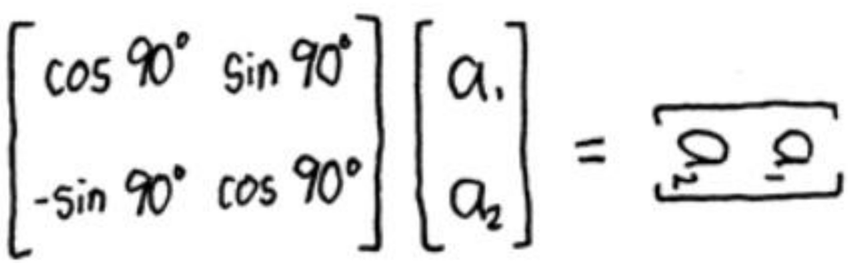
Representing and operating on pose transformations

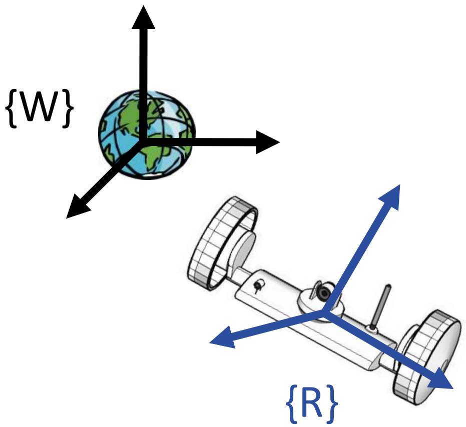


https://xkcd.com/184/

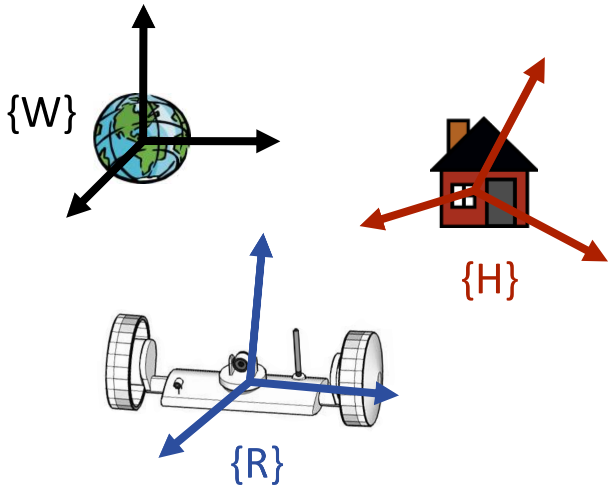
Slides in part adapted from
Gianni di Caro, CMU

Some important problems you will have soon

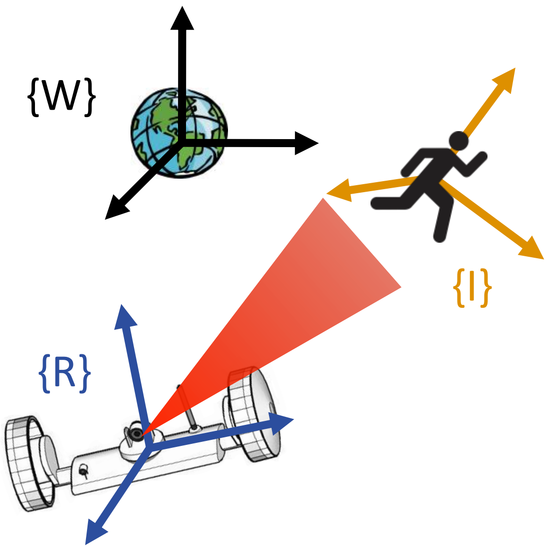
What is robot's pose with respect to the world reference frame {W}?



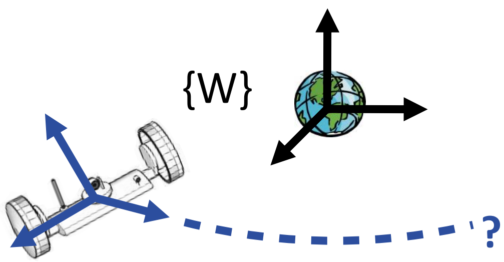
What is robot's pose with respect to the external frame {H}?



What is intruder's pose, observed using robot's lateral camera, in the world frame {W}?



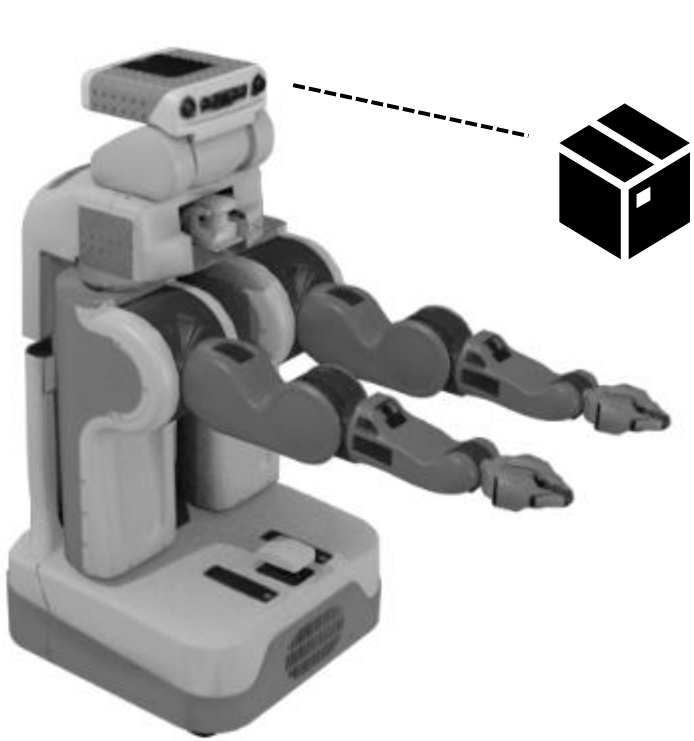
What is robot's pose in {W} after moving at a velocity **v** for 1 minute?

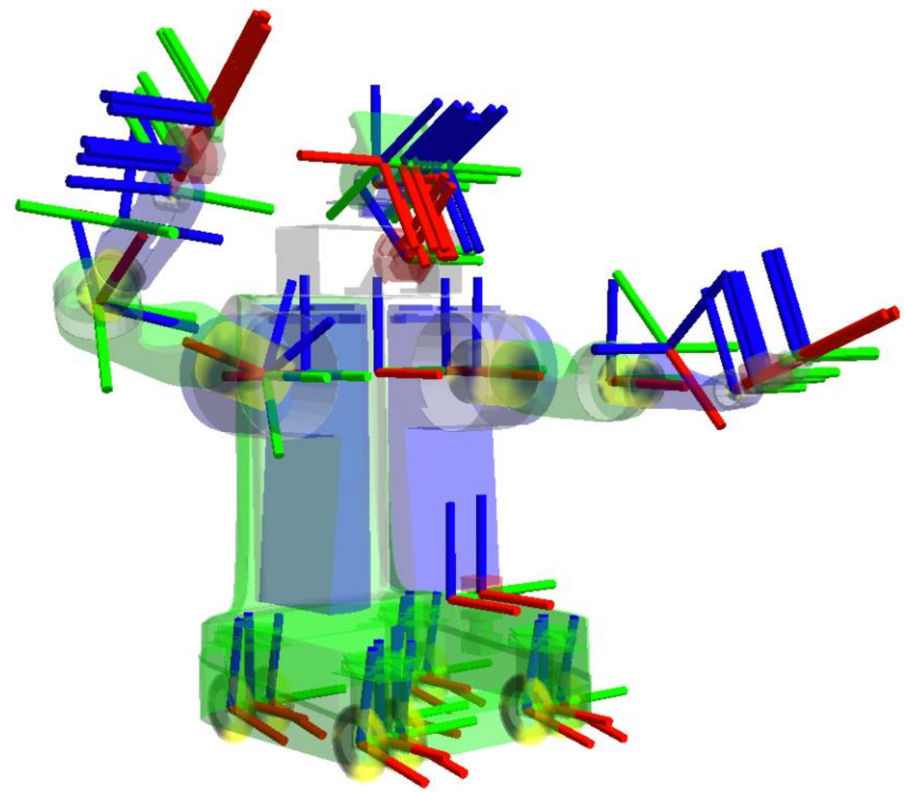


What is the velocity profile that allows to reach a pose ξ in {W}?

Some important problems you will have soon

What is the current pose with respect to my left hand of the object that my camera sees?



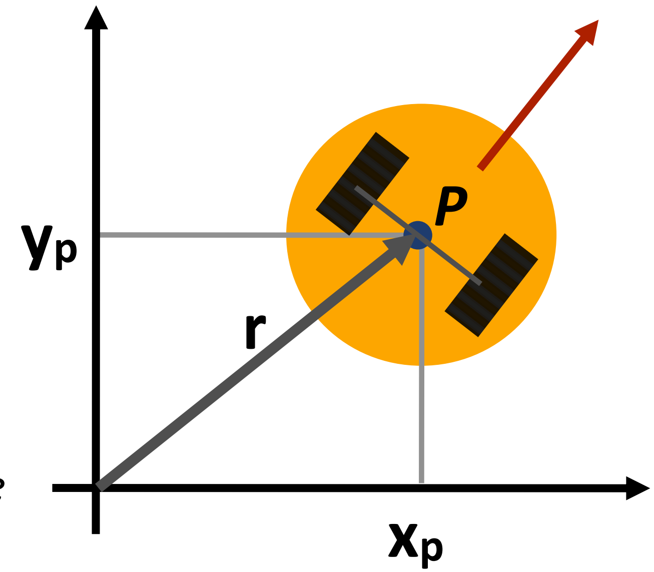


What is the pose of a robot?

- Define a fixed world reference coordinate frame {W}

- Center a local coordinate frame {R} in the robot's reference point *P*, oriented according to robot's natural orientation

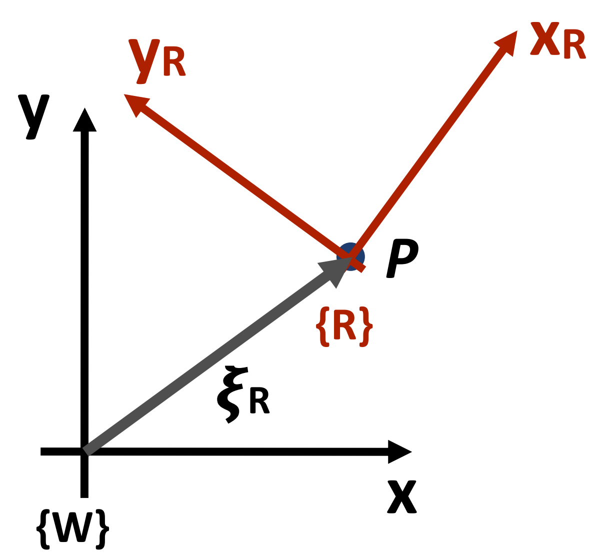
- A point in space (= chosen reference point *P*) is described by a **coordinate vector *r*** representing the displacement of the point with respect to the *reference coordinate frame* {W} (e.g., using cartesian coordinates)



The <u>pose/configuration</u> of the object/robot in {W} is described by the position and orientation of the *local coordinate frame* {R} wrt {W}

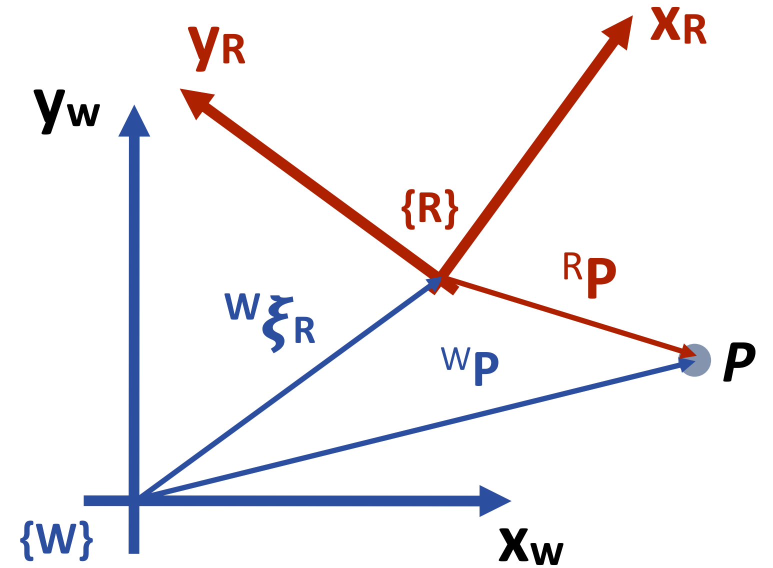
pronounced /ksaɪ/

ξ_R is the **relative pose** of the frame/robot with respect to the reference coordinate frame



Use and properties of relative poses

- **The relative pose ${}^{W}\xi_{R}$ describes the frame {R} with respect to the frame {W}**
- The leading superscript denotes the reference coordinate frame, the trailing subscript denotes the frame being described
- If the leading superscript is missing the reference frame is a world coordinate frame.
- ξ is the object being described



${}^{W}\xi_{R}$ could be seen as describing some *motion*: first applying a displacement and then a rotation to {W}

The point $\boldsymbol{P}$ can be described with respect to either coordinate frame:

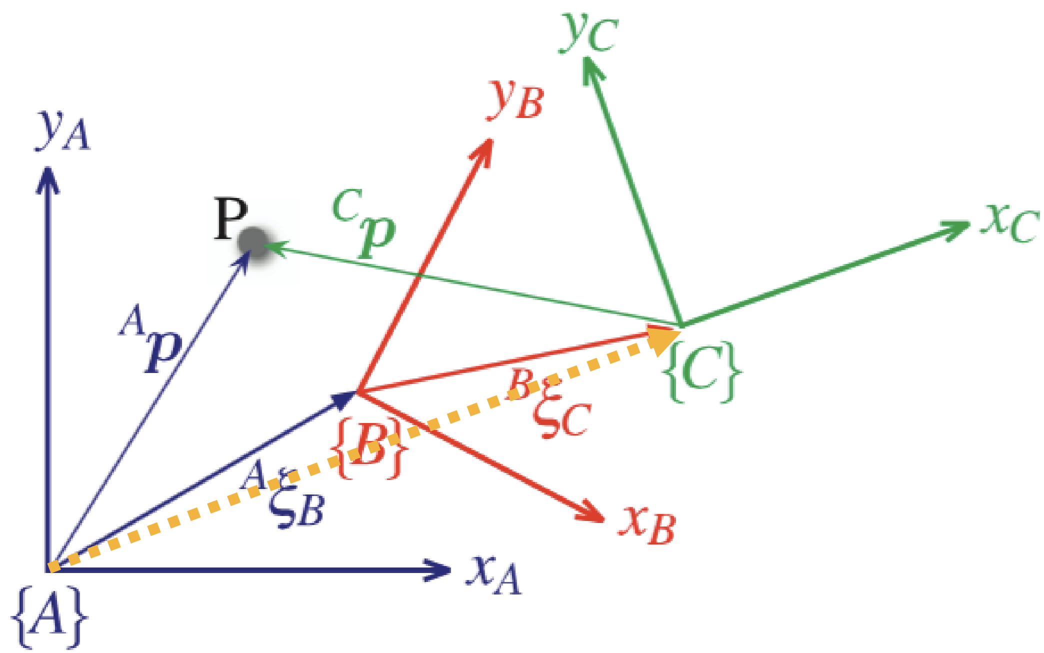
$${}^{W}\boldsymbol{P} = {}^{W}\xi_{R} \cdot {}^{R}\boldsymbol{P}$$

the right-hand side expresses the *motion*

$\{W\} \longrightarrow \{R\} \longrightarrow \boldsymbol{P}$

The operator · transforms the vector, resulting in a new vector that describes the same point but with respect to a different coordinate frame.

Composition of relative poses



One frame can be described in terms of another by a relative pose
The procedure can be applied sequentially by a relative pose composition operator $\oplus$

Pose({C} relative to {A}) = Pose({B} relative to {A}) $\oplus$ Pose({C} relative to {B})

$$ {}^A\xi_C = {}^A\xi_B \oplus {}^B\xi_C \qquad {}^A\boldsymbol{p} = ({}^A\xi_B \oplus {}^B\xi_C) \cdot {}^C\boldsymbol{p} = {}^A\xi_C \cdot {}^C\boldsymbol{p} $$

A relative pose can transform a point expressed as a vector relative to one frame to a vector relative to another

$$ {}^A\boldsymbol{p} = {}^A\xi_B \cdot {}^B\boldsymbol{p} $$

Poses form an additive group

The set of poses equipped with the combination operator $\oplus$ form an ***additive group.***
In the case of a 2d pose this is the **SE(2) Special Euclidean group**
In the case of a 3d pose this is the **SE(3) Special Euclidean group**

Closure:	$\xi_1 \oplus \xi_2 = \xi_3$	Composition of two rel. poses results in a new rel. pose ${}^{A}\xi_{B} \oplus {}^{B}\xi_{C} = {}^{A}\xi_{C}$ ${}^{B}\xi_{A} \oplus {}^{A}\xi_{C} = {}^{B}\xi_{C}$
Associativity:	$({}^{A}\xi_{B} \oplus {}^{B}\xi_{C}) \oplus {}^{C}\xi_{D} = {}^{A}\xi_{B} \oplus ({}^{B}\xi_{C} \oplus {}^{C}\xi_{D})$	
Identity element:	$\xi \oplus 0 = 0 \oplus \xi = \xi$	0 is the *null relative pose*
Inverse:	$\ominus\, {}^{A}\xi_{B} = {}^{B}\xi_{A}$	$(\ominus \xi) \oplus \xi = \xi \oplus (\ominus \xi) = 0$ $\xi \oplus (\ominus\, 0) = \xi$

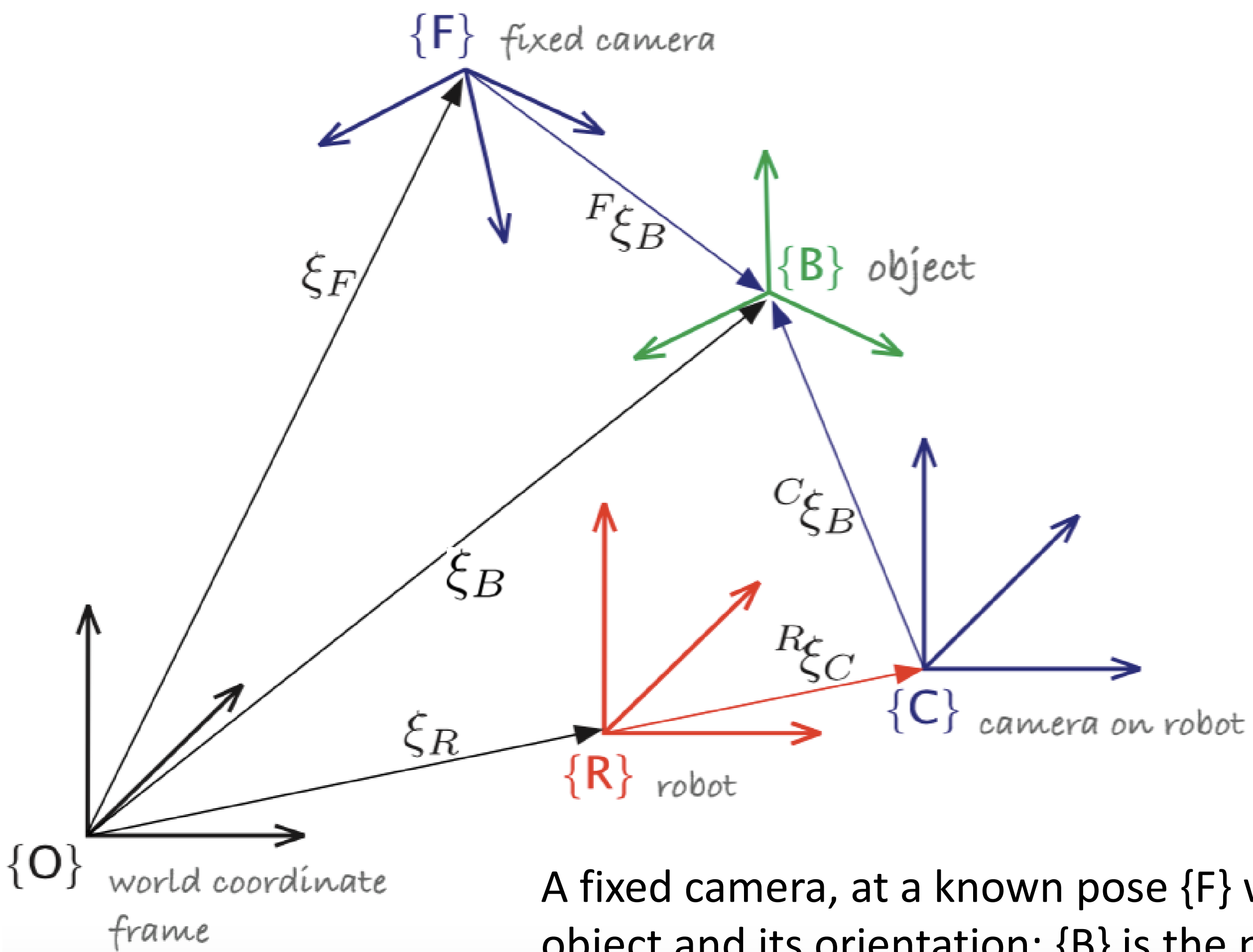
Composition is NOT commutative:
(because of the angle part of poses)

$$\xi_1 \oplus \xi_2 \neq \xi_2 \oplus \xi_1$$

Summary of operators

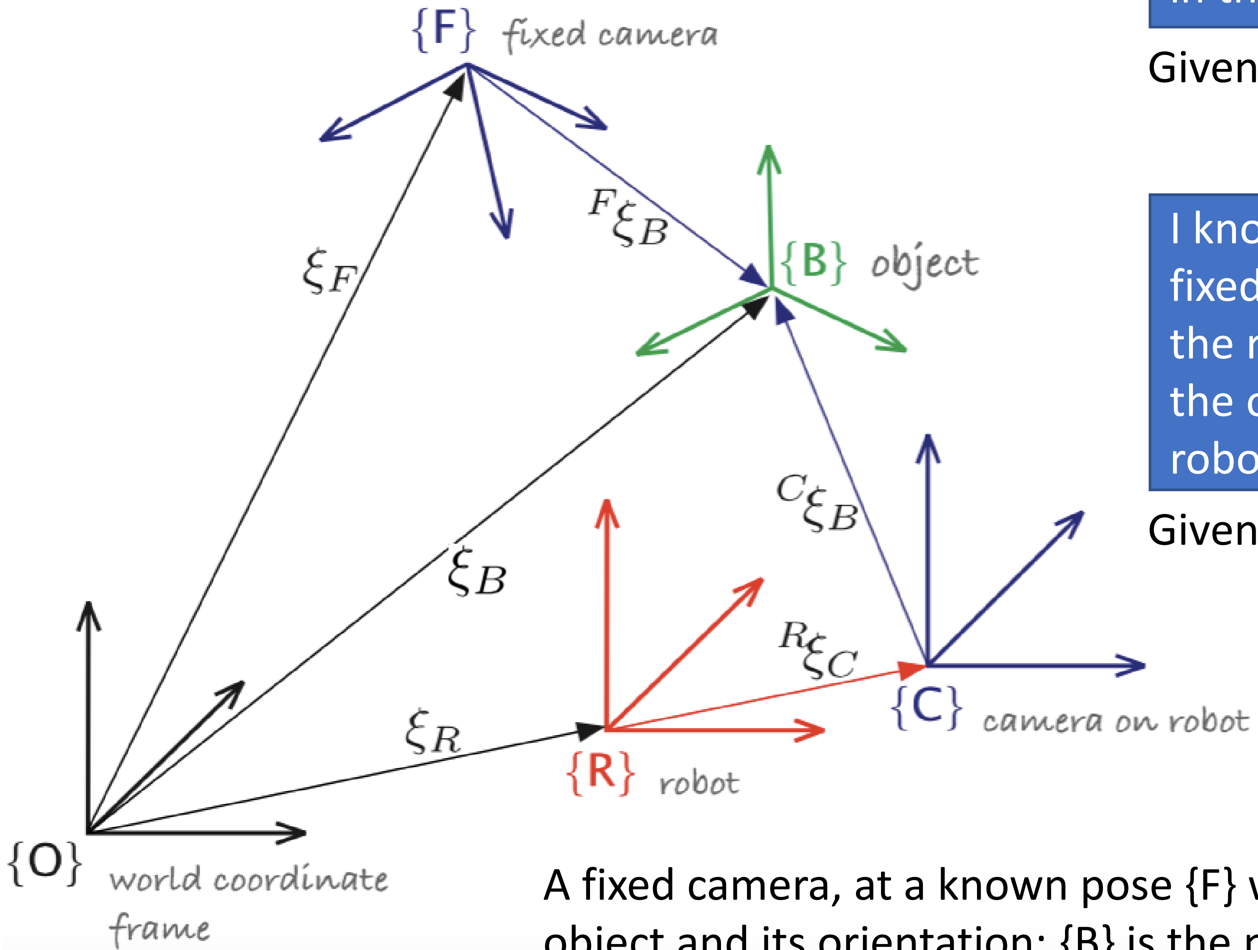
${}^{A}\xi_{B} \oplus {}^{B}\xi_{C} = {}^{A}\xi_{C}$	Pose composition (binary)
$\ominus\, {}^{A}\xi_{B} = {}^{B}\xi_{A}$	Pose inversion (unary)
${}^{A}\xi_{B} \cdot {}^{B}P = {}^{A}P$ To the right we always must have a point	Change of reference frame for a point

A more complex 3d example



A fixed camera, at a known pose {F} wrt the world, observes an object and its orientation; {B} is the pose of an object.
A robot {R}, at a known pose {R} wrt the world, is equipped with a camera (mounted at a known pose {C} wrt the robot). The robot camera observes the same object and its orientation.

A more complex 3d example



I see the object in the robot's camera. I know where the robot is in the world. Where is the object in the world?

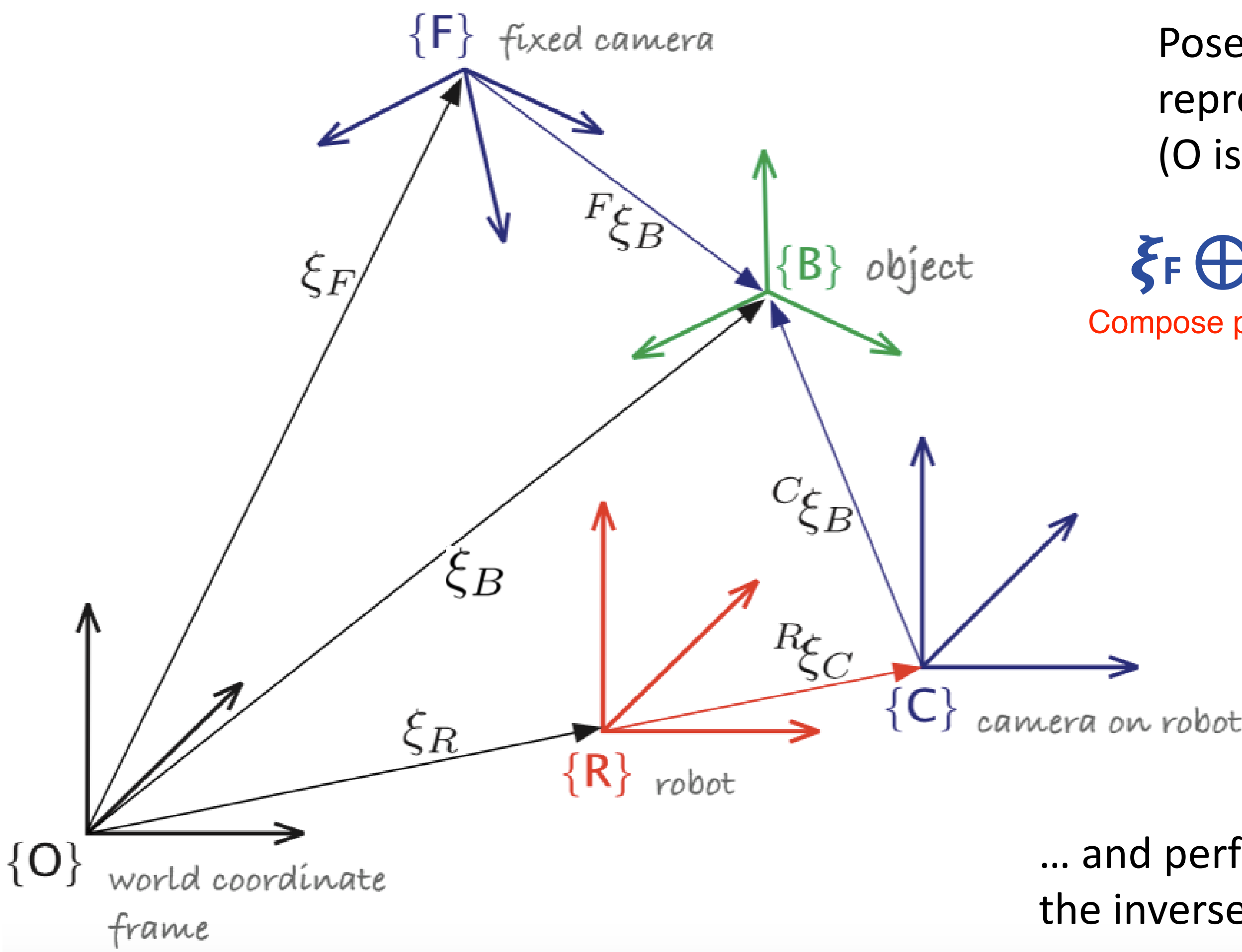
Given $^{C}\xi_{B}$, $^{R}\xi_{C}$, ξ_{R} find ξ_{B}

I know the object pose in the fixed camera frame. I know where the robot is in the world. Where is the object with respect to the robot?

Given $^{F}\xi_{B}$, ξ_{F} , ξ_{R} find $^{R}\xi_{B}$

A fixed camera, at a known pose {F} wrt the world, observes an object and its orientation; {B} is the pose of an object.
A robot {R}, at a known pose {R} wrt the world, is equipped with a camera (mounted at a known pose {C} wrt the robot). The robot camera observes the same object and its orientation.

A more complex 3d example



Pose composition allows us to represent useful spatial relations … (O is dropped for short)

$$\xi_F \oplus {}^F\xi_B = \xi_R \oplus {}^R\xi_C \oplus {}^C\xi_B$$

Compose position of the object in two different ways

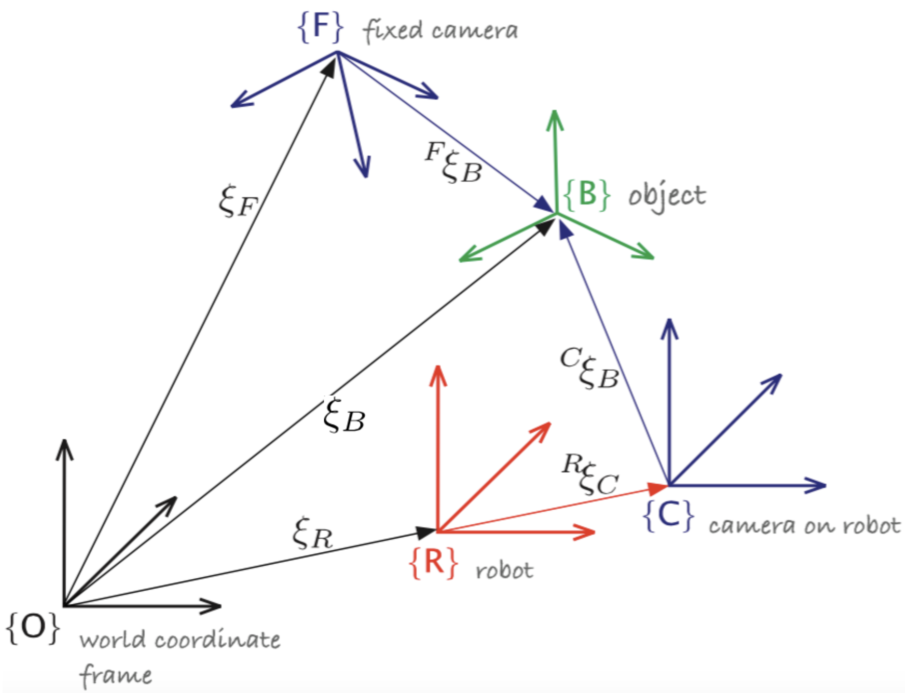
$$\xi_F \oplus {}^F\xi_R = \xi_R$$

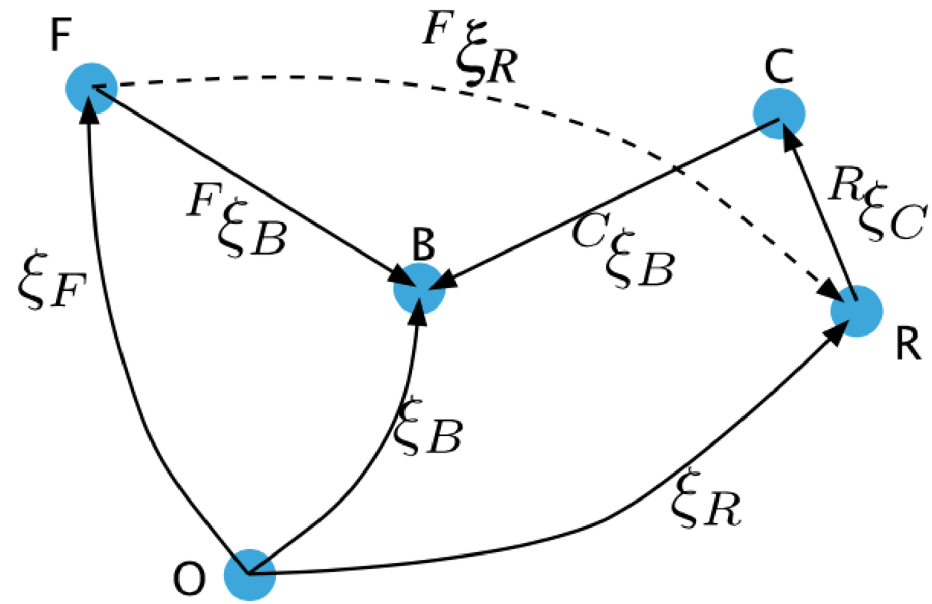
… and perform computations (adding the inverse to both sides to 2nd eq.)

$$\ominus\xi_F \oplus \xi_F \oplus {}^F\xi_R = \ominus\xi_F \oplus \xi_R$$

Pose of the robot relative to the fixed camera $\quad {}^F\xi_R = \ominus\xi_F \oplus \xi_R \qquad {}^F\xi_R = {}^F\xi_O \oplus \xi_R$

A more complex 3d example





In the graph, each node represents a pose, and each arc is a relative pose

A spatial relation is a *loop in the graph*
Both sides of an equation start and end at the same node.

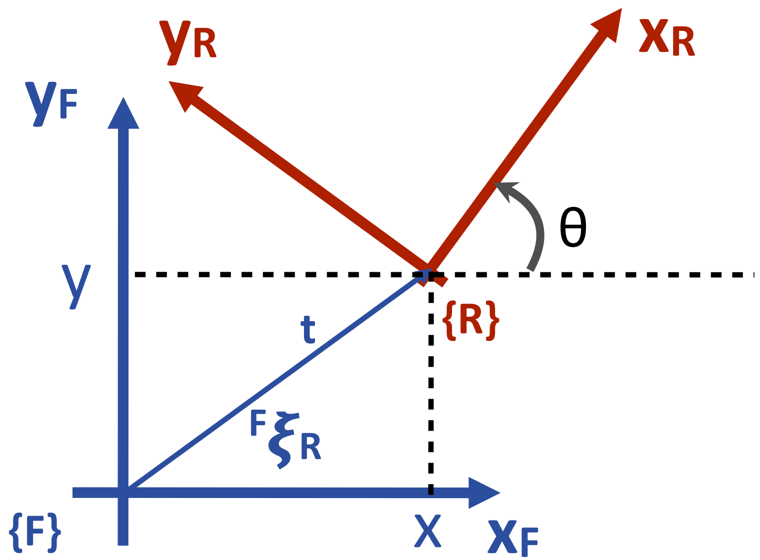
$$\xi_F \oplus {}^F\xi_B = \xi_R \oplus {}^R\xi_C \oplus {}^C\xi_B$$

$$\xi_F \oplus {}^F\xi_R = \xi_R$$

Representation of poses in 2d

What are in practice ξ and ⊕?
Any mathematical objects/operators that support the described group properties (algebra) and are suited to the problem at hand.

- In 2D, for a wheeled robot, a concrete representation of a pose is in a Cartesian coordinate system through the (x,y) coordinates and the θ angle for the orientation



To describe a coordinate frame {R} wrt a reference frame {F} it can be noticed that:

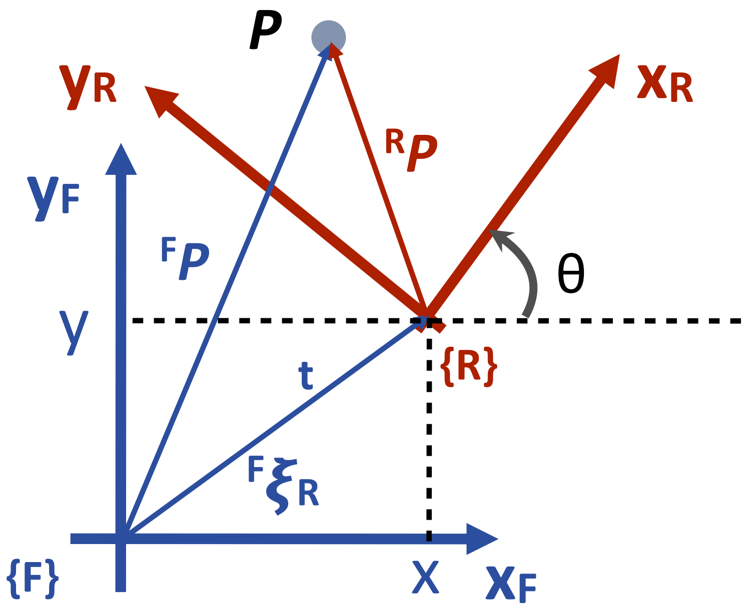
1. Origin of {R} has been displaced by the vector t = (x,y) ➔ **Translation**
2. {R} has been rotated counter-clockwise by an angle θ ➔ **Rotation**

$^F\xi_R \sim (x, y, \theta)$ $\sim$ stands for *equivalency*

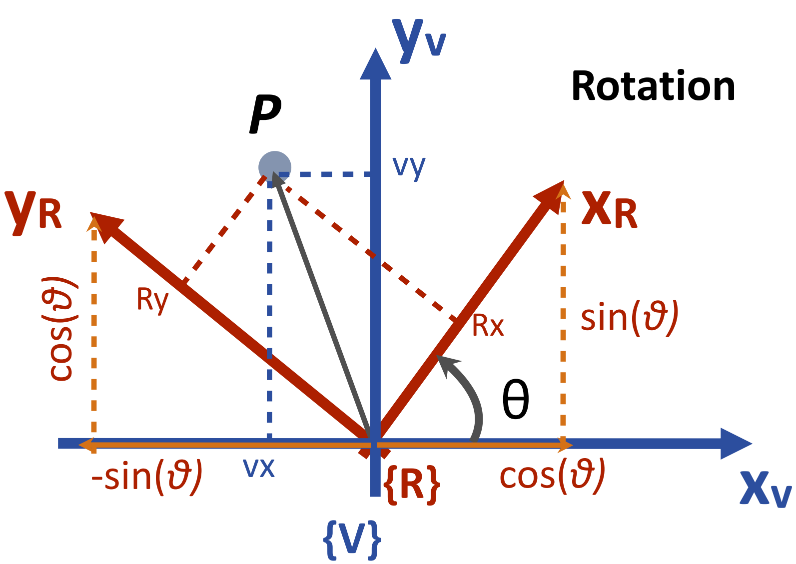
We can represent R with 3 values

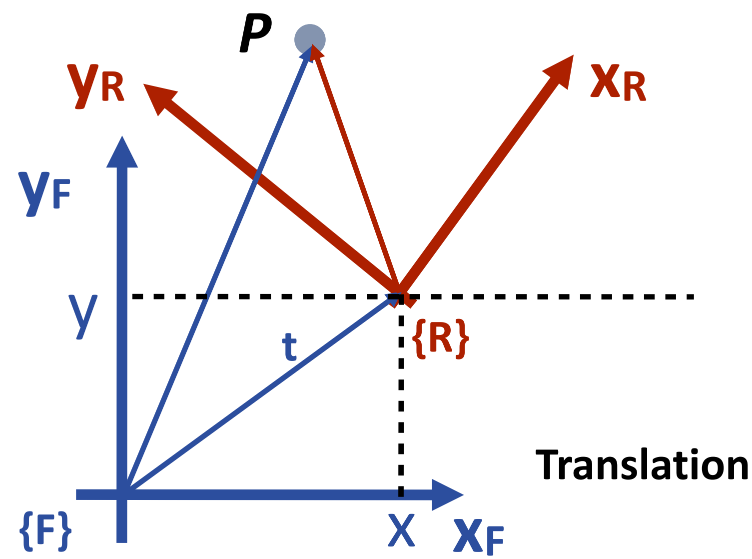
Unfortunately this representation is not really convenient for compounding since $(x_1, y_1, \theta_1) \oplus (x_2, y_2, \theta_2)$ would require some complex trigonometric functions

Representation of poses in 2d

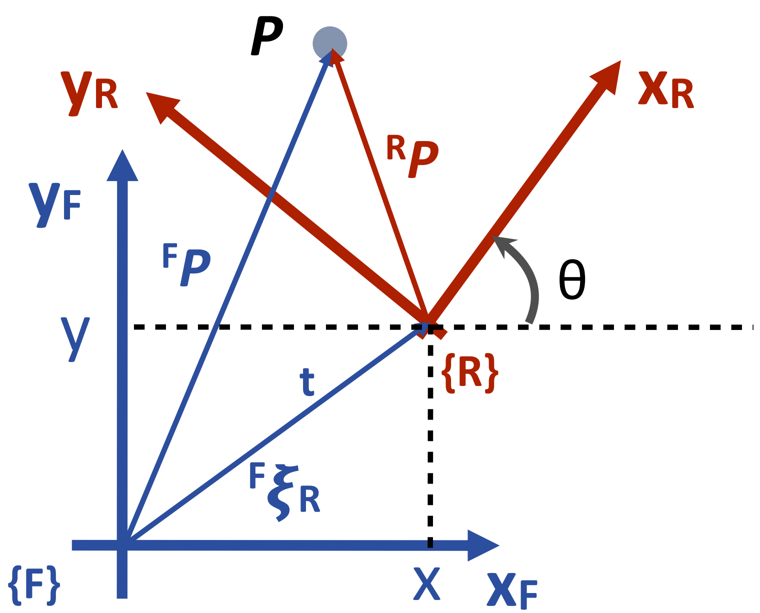


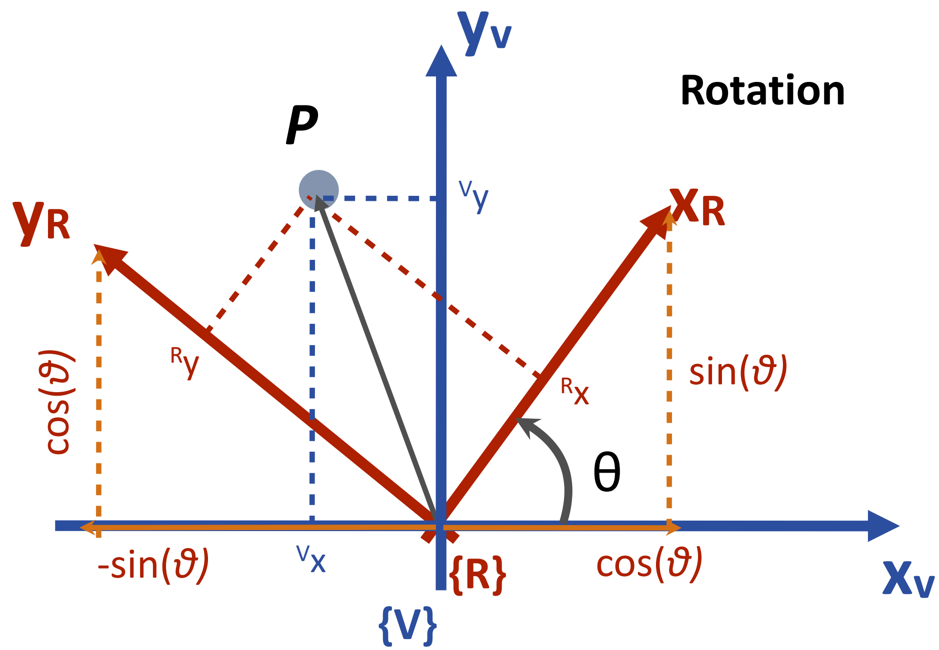
- Let's consider an arbitrary point ***P*** with respect to each of the coordinate frames {F} and {R} and see how the relationship between $^F\boldsymbol{P}$ and $^R\boldsymbol{P}$ can be determined as translation ⊕ rotation
- Let {V} be a new coordinate frame, centered in {R} but rotated as {F}





Frame rotation





The new frame {V} has axes parallel to {F} but origin is the same as {R}

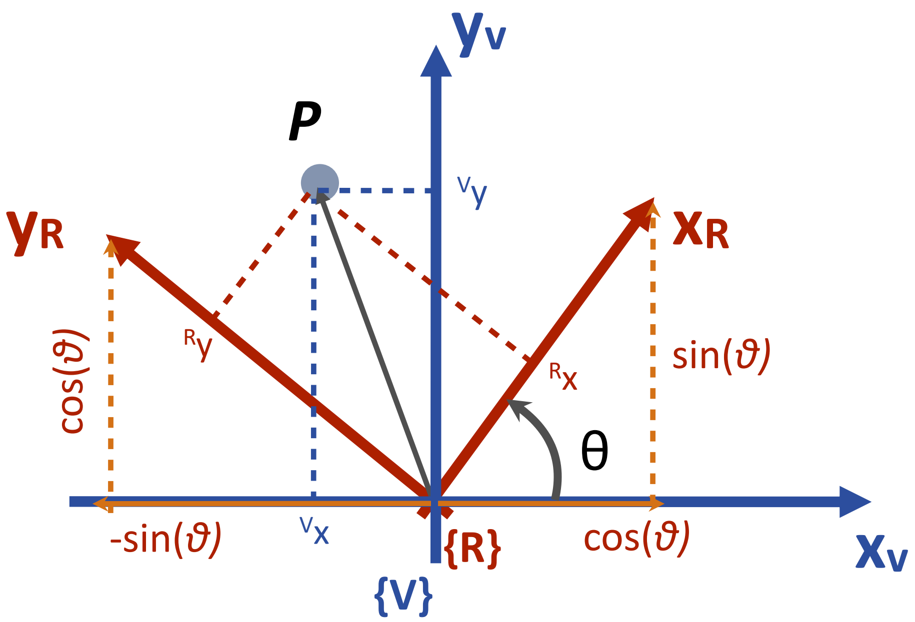
Point **P** in {V}: $${}^{\mathrm{V}}\mathbf{P} = {}^{\mathrm{V}}\mathrm{x}\,\hat{\mathbf{x}}_{\mathrm{V}} + {}^{\mathrm{V}}\mathrm{y}\hat{\mathbf{y}}_{\mathrm{V}} = \begin{bmatrix}\hat{\mathbf{x}}_{\mathrm{V}} & \hat{\mathbf{y}}_{\mathrm{V}}\end{bmatrix}\begin{bmatrix}{}^{\mathrm{V}}\mathrm{x} \\ {}^{\mathrm{V}}\mathrm{y}\end{bmatrix}$$

Coordinate frame {R} is fully described by its orthogonal axes, whose *unit vectors* can be expressed in terms of {V}'s *unit vectors*:

$$\hat{\mathbf{x}}_{\mathrm{R}} = \cos(\theta)\hat{\mathbf{x}}_{\mathrm{V}} + \sin(\theta)\hat{\mathbf{y}}_{\mathrm{V}}$$
$$\hat{\mathbf{y}}_{\mathrm{R}} = -\sin(\theta)\hat{\mathbf{x}}_{\mathrm{V}} + \cos(\theta)\hat{\mathbf{y}}_{\mathrm{V}}$$

$$\begin{bmatrix}\hat{\mathbf{x}}_{\mathrm{R}} & \hat{\mathbf{y}}_{\mathrm{R}}\end{bmatrix} = \begin{bmatrix}\hat{\mathbf{x}}_{\mathrm{V}} & \hat{\mathbf{y}}_{\mathrm{V}}\end{bmatrix}\begin{bmatrix}\cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta)\end{bmatrix}$$

Frame rotation



$$[\hat{\mathbf{x}}_R \quad \hat{\mathbf{y}}_R] = [\hat{\mathbf{x}}_v \quad \hat{\mathbf{y}}_v] \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$$

$${}^{v}\mathbf{P} = {}^{v}x\,\hat{\mathbf{x}}_v + {}^{v}y\hat{\mathbf{y}}_v = [\hat{\mathbf{x}}_v \quad \hat{\mathbf{y}}_v] \begin{bmatrix} {}^{v}x \\ {}^{v}y \end{bmatrix}$$

$${}^{R}\mathbf{P} = {}^{R}x\,\hat{\mathbf{x}}_R + {}^{R}y\hat{\mathbf{y}}_R = [\hat{\mathbf{x}}_R \quad \hat{\mathbf{y}}_R] \begin{bmatrix} {}^{R}x \\ {}^{R}y \end{bmatrix}$$

Substituting …

$${}^{R}\mathbf{P} = [\hat{\mathbf{x}}_v \quad \hat{\mathbf{y}}_v] \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} {}^{R}x \\ {}^{R}y \end{bmatrix}$$

This is the rotation matrix

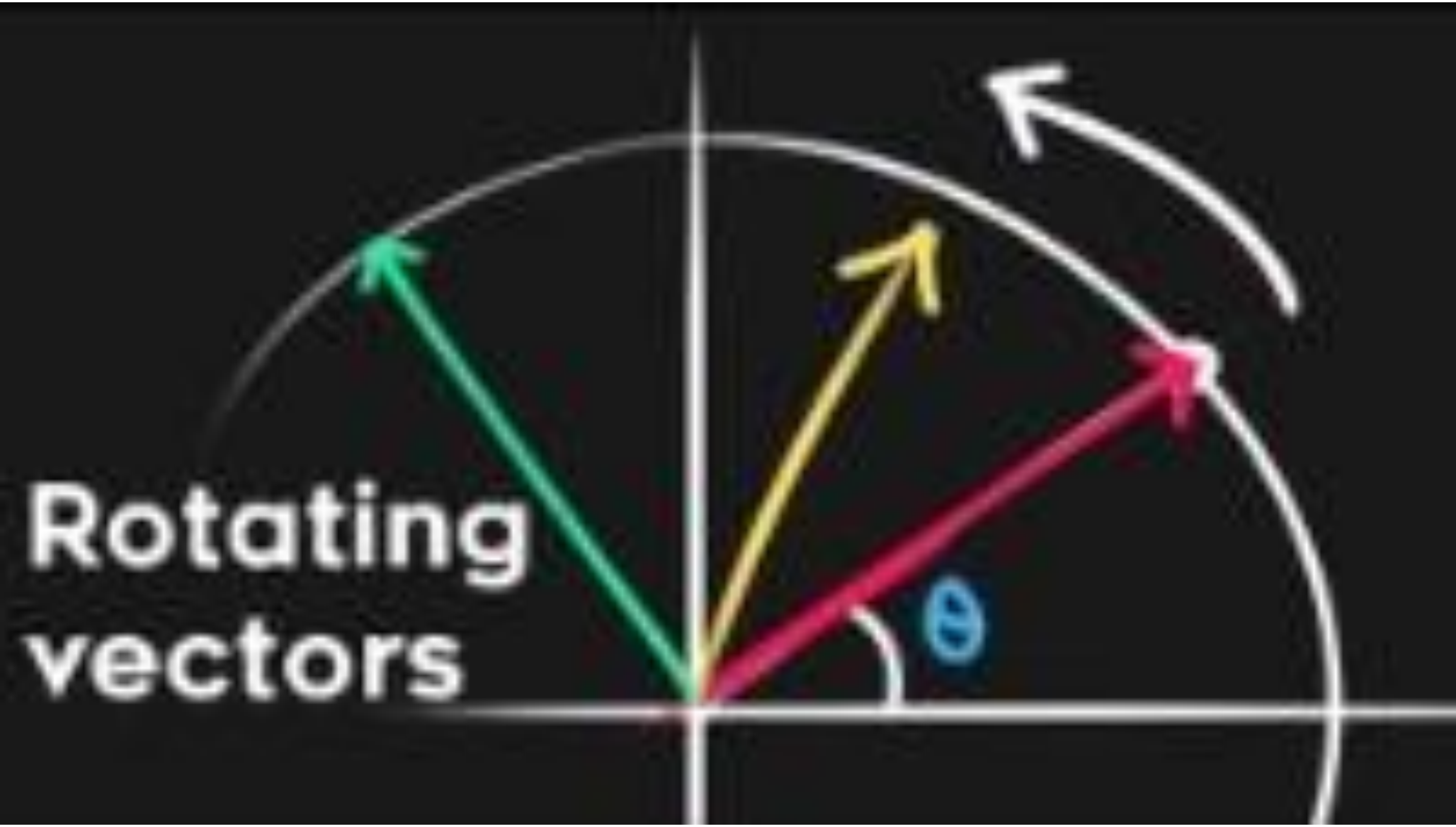
How points are transformed (from {R} to {V}) when the frame is rotated:

$$\begin{bmatrix} {}^{v}x \\ {}^{v}y \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} {}^{R}x \\ {}^{R}y \end{bmatrix}$$

$$\begin{bmatrix} {}^{v}x \\ {}^{v}y \end{bmatrix} = {}^{v}\mathbf{R}_R(\theta) \begin{bmatrix} {}^{R}x \\ {}^{R}y \end{bmatrix} \qquad \begin{bmatrix} {}^{R}x \\ {}^{R}y \end{bmatrix} = {}^{v}\mathbf{R}_R^{-1}(\theta) \begin{bmatrix} {}^{v}x \\ {}^{v}y \end{bmatrix}$$

Rotation matrices

2D Vector Rotation



https://youtu.be/7j5yW5QDC2U

Rotation matrix

$$\mathbf{R}(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$$ Has nice properties …

- Instead of using one scalar, the angle θ, we are using a 2x2 matrix to represent orientation

- The rotation matrix is **orthonormal**: each of its columns is a unit vector and the columns are orthogonal, that is, represent an orthonormal *basis* (the columns are the unit vectors that define {R} with respect to {V}) → 4 parameters, 3 functional relations/constraints → one independent value (the angle!)

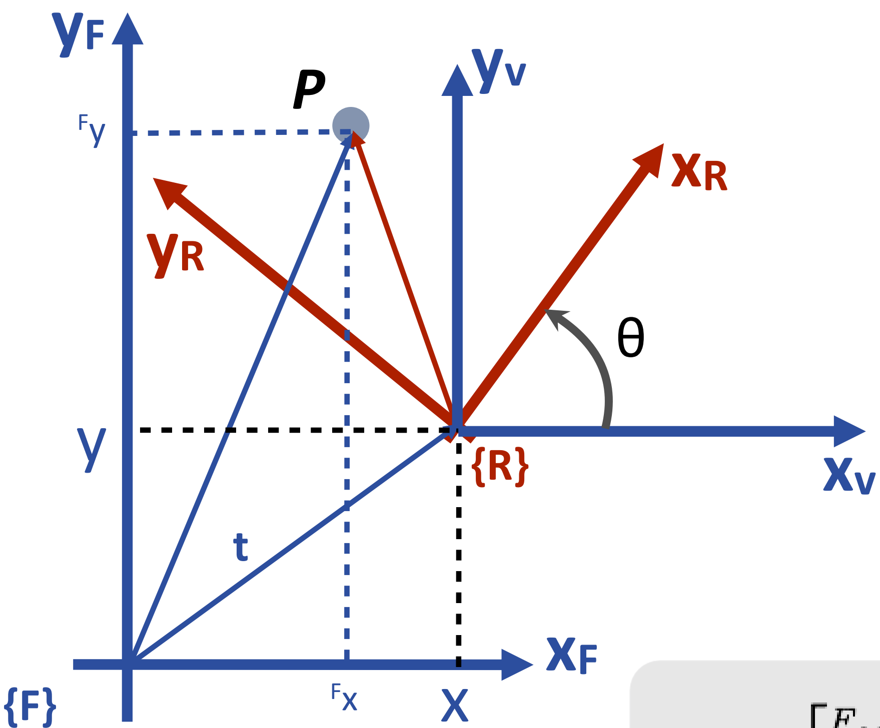
- The determinant is +1: **R** belongs to the special orthogonal group of dimension 2, ***SO(2),*** acting as an isometry ➔ the **length of a vector is unchanged after a rotation**

- **The inverse is the same as the transpose: $\mathbf{R}^{-1} = \mathbf{R}^T$**

- Inverting the matrix is the same as swapping superscript and subscript, which leads to the identity $\mathbf{R}(-\theta) = \mathbf{R}^T(\theta) = \mathbf{R}^{-1}(\theta)$

$$\begin{bmatrix} {}^{R}x \\ {}^{R}y \end{bmatrix} = ({}^{V}\boldsymbol{R}_{R})^{-1} \begin{bmatrix} {}^{V}x \\ {}^{V}y \end{bmatrix} = ({}^{V}\boldsymbol{R}_{R})^{T} \begin{bmatrix} {}^{V}x \\ {}^{V}y \end{bmatrix} = {}^{R}\boldsymbol{R}_{V} \begin{bmatrix} {}^{V}x \\ {}^{V}y \end{bmatrix}$$

Add the translation



{F} and {V} have parallel coordinate axes:

$$^F\mathbf{P} = \begin{bmatrix} ^F x \\ ^F y \end{bmatrix} = \begin{bmatrix} ^V x \\ ^V y \end{bmatrix} + \begin{bmatrix} x \\ y \end{bmatrix}$$

$^V\mathbf{P}$ was obtained from the rotation:

$$\begin{bmatrix} ^V x \\ ^V y \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} ^R x \\ ^R y \end{bmatrix}$$

$$^F\mathbf{P} = \begin{bmatrix} ^F x \\ ^F y \end{bmatrix} = \underbrace{\begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} ^R x \\ ^R y \end{bmatrix}}_{Rotation} + \underbrace{\begin{bmatrix} x \\ y \end{bmatrix}}_{Translation}$$

In a more compact form:

$$\begin{bmatrix} ^F x \\ ^F y \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & x \\ \sin(\theta) & \cos(\theta) & y \end{bmatrix} \begin{bmatrix} ^R x \\ ^R y \\ 1 \end{bmatrix}$$

Homogeneous transformation Matrix

$$\begin{bmatrix} {}^F\mathsf{x} \\ {}^F\mathsf{y} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & \mathsf{x} \\ \sin(\theta) & \cos(\theta) & \mathsf{y} \end{bmatrix} \begin{bmatrix} {}^R\mathsf{x} \\ {}^R\mathsf{y} \\ 1 \end{bmatrix}$$

Homogenous
2D vector!
(allows us to represent a translation by a matrix product)

$$\begin{bmatrix} {}^F\mathsf{x} \\ {}^F\mathsf{y} \\ 1 \end{bmatrix} = \begin{bmatrix} {}^F\mathbf{R}_\mathsf{R} & \mathbf{t} \\ \mathbf{0}_{1\times 2} & 1 \end{bmatrix} \begin{bmatrix} {}^R\mathsf{x} \\ {}^R\mathsf{y} \\ 1 \end{bmatrix}$$

Coordinate vectors for ***P*** are now
in 2D homogenous form:
homogenous coordinate transformation

$${}^F\tilde{\mathbf{P}} = \begin{bmatrix} {}^F\mathbf{R}_\mathsf{R} & \mathbf{t} \\ \mathbf{0}_{1\times 2} & 1 \end{bmatrix} {}^R\tilde{\mathbf{P}}$$

$${}^F\tilde{\mathbf{P}} = {}^F\mathbf{T}_R\,{}^R\tilde{\mathbf{P}}$$

$${}^F\mathbf{T}_R = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & \mathsf{x} \\ \sin(\theta) & \cos(\theta) & \mathsf{y} \\ 0 & 0 & 1 \end{bmatrix}$$

The roto-translation matrix **T** is a *homogenous transformation* and belongs to SE(2)

The pose of a robot / rigid body is fully described by a *homogeneous transformation matrix*

Homogeneous transformation Matrix

$^F\xi_R \sim {}^F\mathbf{T}_R$

$\oplus$ becomes a matrix product: $^A\xi_R \oplus {}^R\xi_B \equiv {}^AT_R\,{}^RT_B$

$\cdot$ becomes a matrix-vector product: $^AP = {}^A\xi_B \cdot {}^BP \equiv {}^AT_B \cdot {}^BP$

The identity element is the identity matrix

The inverse rel pose is the matrix inverse

Relative pose transformation of a point:

$$^AP = {}^A\xi_B \cdot {}^BP$$

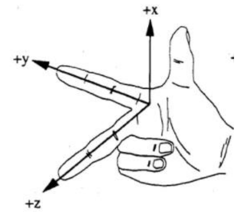
2D

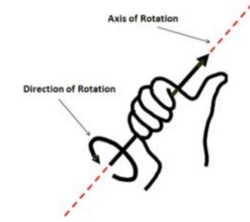
$$^F\tilde{\mathbf{P}} = \begin{bmatrix} ^F\mathbf{R}_\mathrm{R}^{2\times2} & \mathbf{t}_{2\times1} \\ \mathbf{0}_{1\times2} & 1 \end{bmatrix} {}^R\tilde{\mathbf{P}}$$

3D

$$^F\tilde{\mathbf{P}} = \begin{bmatrix} ^F\mathbf{R}_\mathrm{R}^{3\times3} & \mathbf{t}_{3\times1} \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix} {}^R\tilde{\mathbf{P}}$$

Rotation matrices in 3D

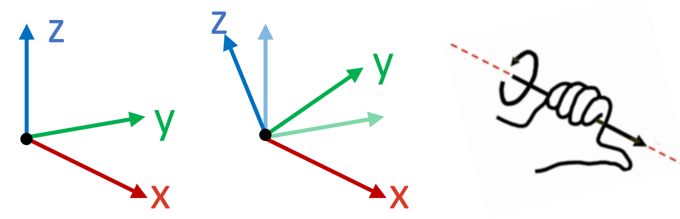




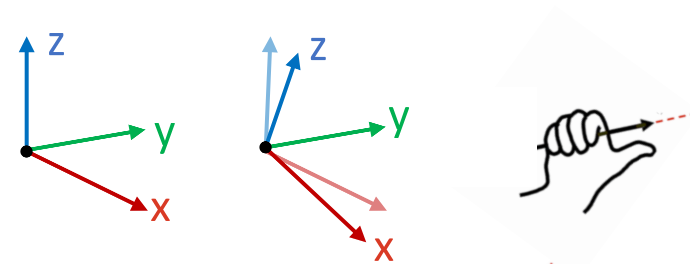
Rotations about the main axes:

In 3D we can rotate around 3 different axis

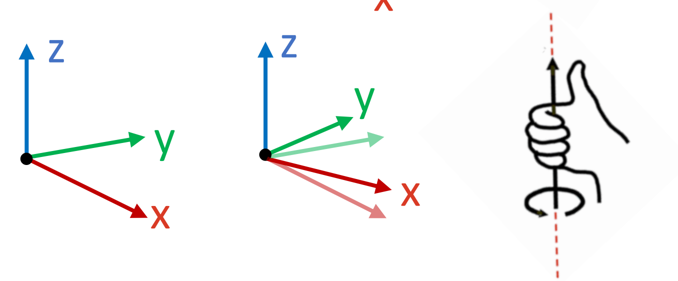
$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix}$$



$$R_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}$$



$$R_z(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



Rotation matrices in 3D

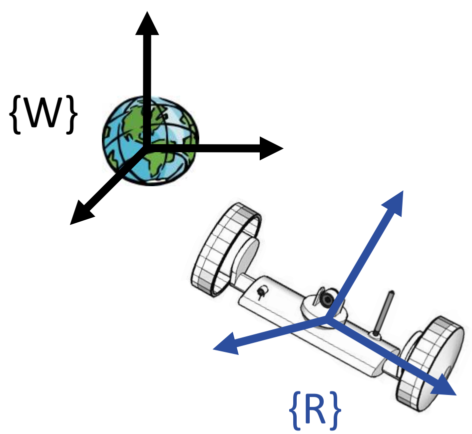
Careful! 3D rotations are not commutative!

$$R_x(\frac{\pi}{4})\cdot R_y(\frac{\pi}{4}) = \begin{bmatrix} 0.707 & 0 & -0.707 \\ -0.5 & 0.707 & -0.5 \\ 0.5 & 0.707 & 0.5 \end{bmatrix},\ R_x(\frac{\pi}{4})\cdot R_y(\frac{\pi}{4})\cdot \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} -1.414 \\ 0.586 \\ 3.414 \end{bmatrix}$$

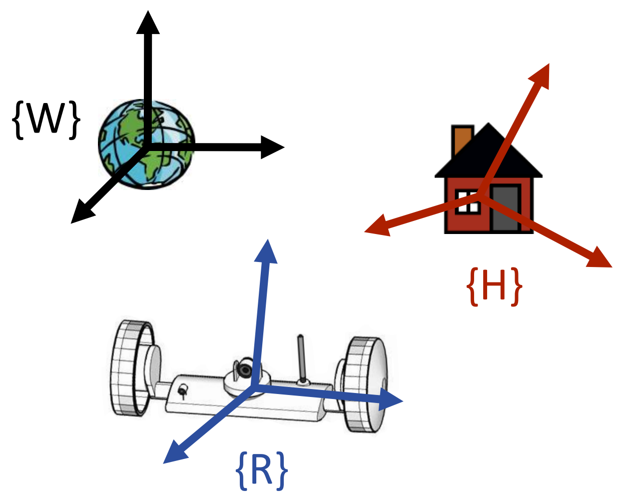
$$R_y(\frac{\pi}{4})\cdot R_x(\frac{\pi}{4}) = \begin{bmatrix} 0.707 & -0.5 & -0.5 \\ 0 & 0.707 & -0.707 \\ 0.707 & 0.5 & 0.5 \end{bmatrix},\ R_y(\frac{\pi}{4})\cdot R_x(\frac{\pi}{4})\cdot \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} -1.793 \\ 0.707 \\ 3.207 \end{bmatrix}$$

Now we can deal with these problems

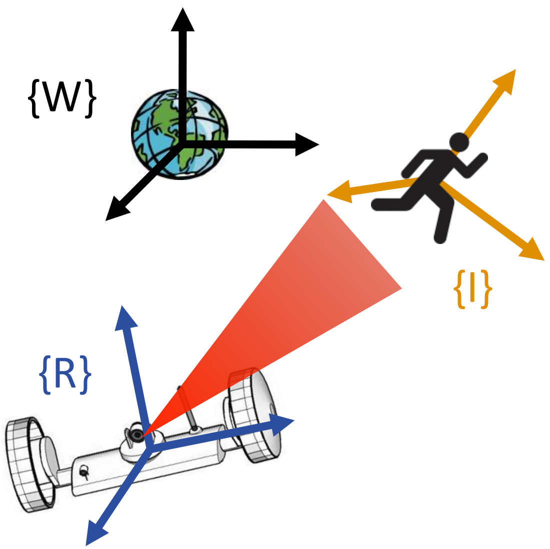
What is robot's pose with respect to the world reference frame {W}?



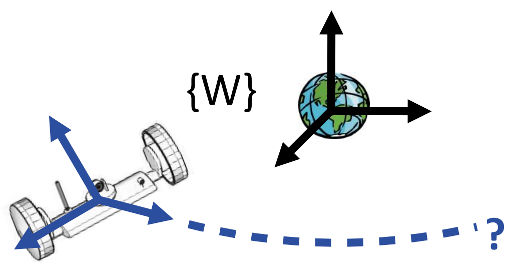
What is robot's pose with respect to the external frame {H}?



What is intruder's pose, observed using robot's lateral camera, in the world frame {W}?

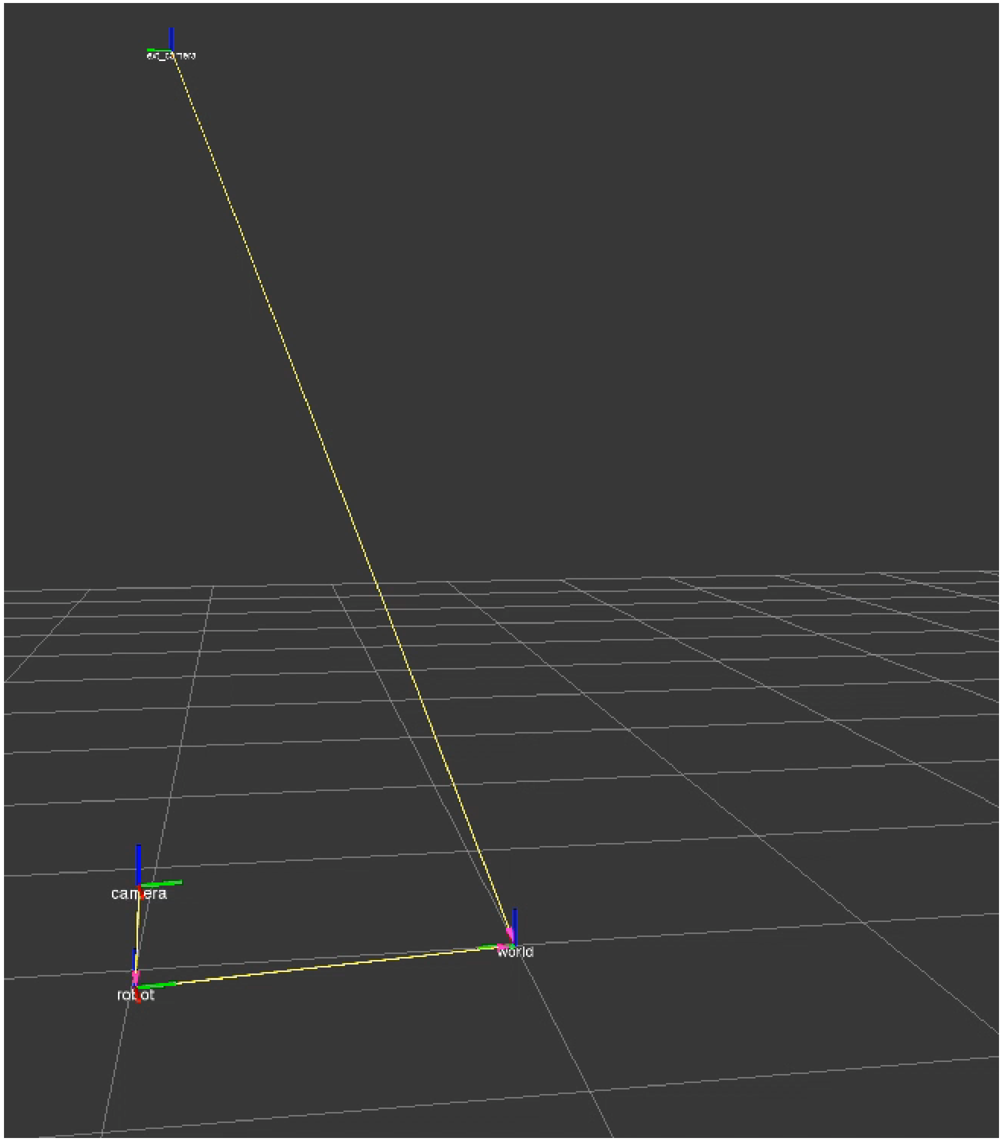


What is robot's pose in {W} after moving at a velocity **v** for 1 minute?



What is the velocity profile that allows to reach a pose ξ in {W}?

Abstractions



Libraries often offer convenient abstractions to represent pose transforms as objects (that can be combined, etc)