# Coordinates and Transformations

## CSCI 420-02 Robotics

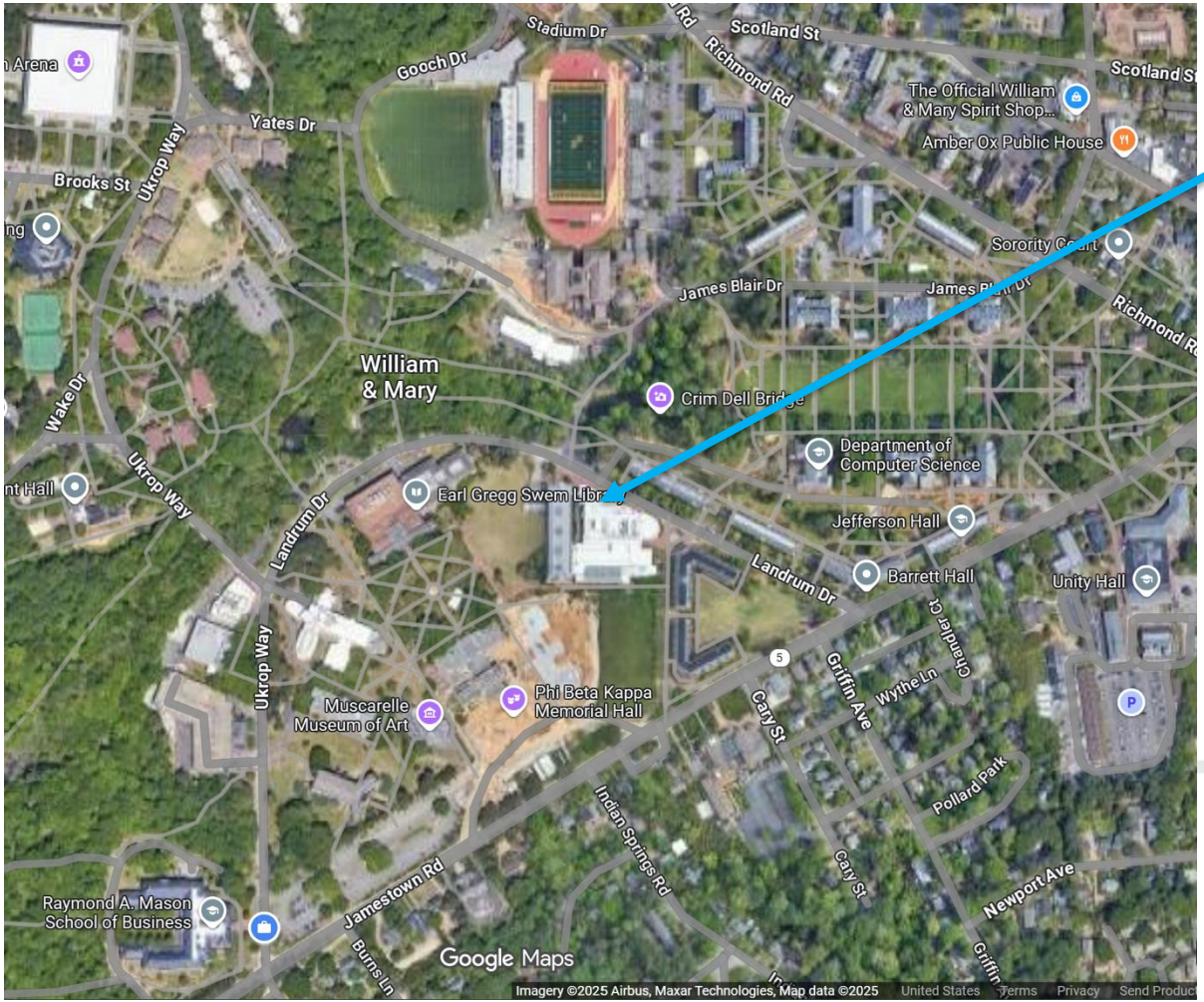


WILLIAM & MARY

CHARTERED 1693

# Where are you?


Stadium Dr
Scotland St
Gooch Dr
Richmond Rd
Yates Dr
The Official William & Mary Spirit Shop...
Amber Ox Public House
Brooks St
Ukrop Way
Sorority Court
James Blair Dr
James Blair Dr
Richmond Rd
William & Mary
Wake Dr
Crim Dell Bridge
Department of Computer Science
Ukrop Way
Earl Gregg Swem Library
Jefferson Hall
Landrum Dr
Landrum Dr
Barrett Hall
Unity Hall
Chandler Ct
Griffin Ave
Wythe Ln
Ukrop Way
Muscarelle Museum of Art
Phi Beta Kappa Memorial Hall
Cary St
Pollard Park
Indian Springs Rd
Cary St
Jamestown Rd
Newport Ave
Raymond A. Mason School of Business
Burns Ln
Google Maps
Imagery ©2025 Airbus, Maxar Technologies, Map data ©2025
United States
Terms
Privacy
Send Product



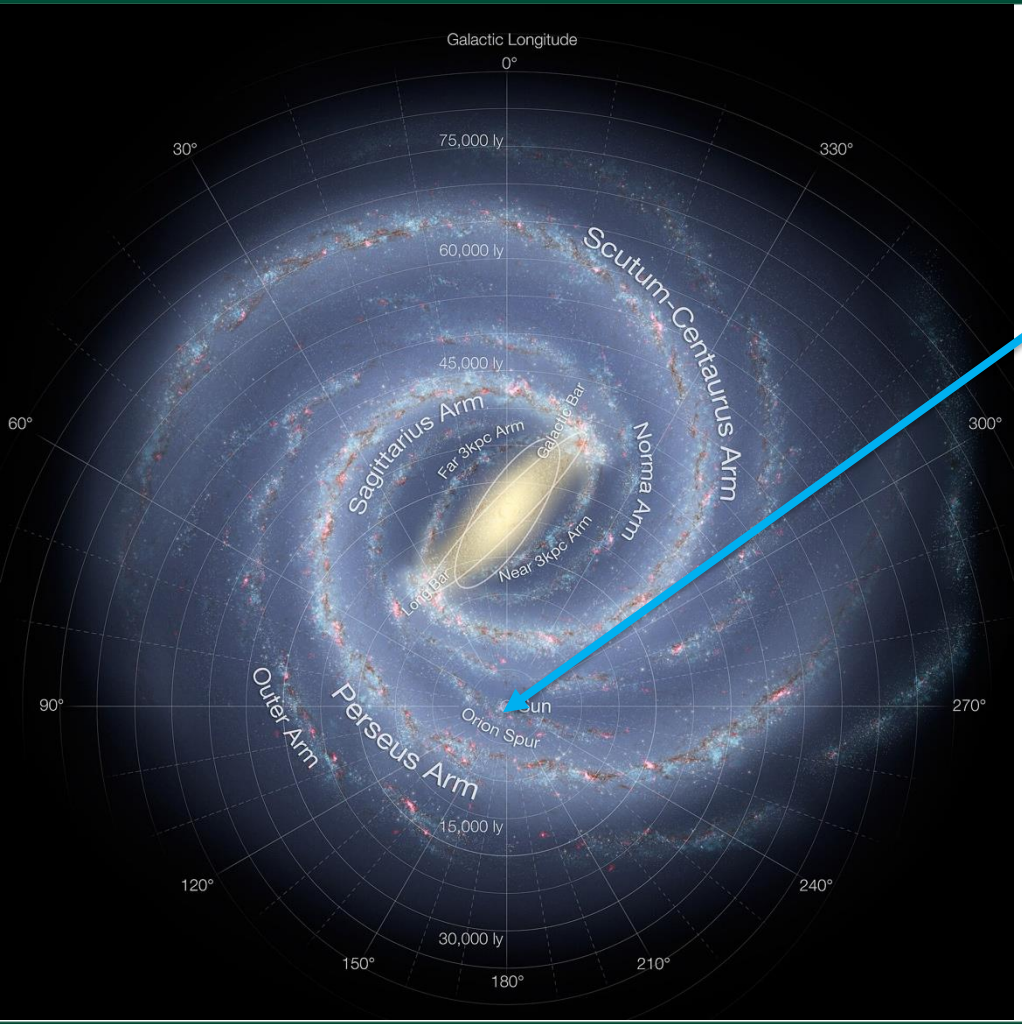

37°16’11.6"N
76°42’51.0"W

779P+X88
Williamsburg, Virginia

540 Ukrop Way
Williamsburg, Virginia

ISC Room 0248
Williamsburg, Virginia

What else can we use to refer to this place?




0°, 0.00001569ly


To School of Education, Dillard Complex Discovery 1, 2 & 3 Map Inset A & B
To Richmond Hall Map Inset E
WILLIAM & MARY
CHARTERED 1693
2025-2026
0 250 500 1,000 Ft.
N
Kaplan Arena
Zable Stadium
Crim Dell
Swem Library
Sunken Garden
Duke of Gloucester St. & Merchants Square
Colonial Williamsburg P6
Undergraduate Admission
To Law School, Grad Housing and Tennis Center Map Inset C
C Law School Grad Housing Tennis Center
Lake Matoaka
Raymond A. Mason School of Business
To Ludwell Apts. via Rolfe Rd Map Inset D
All Vehicles
Service Vehicles
Access Road
Road Parking
1
2
3
4
5
6
A
B
C
D
E
F
G

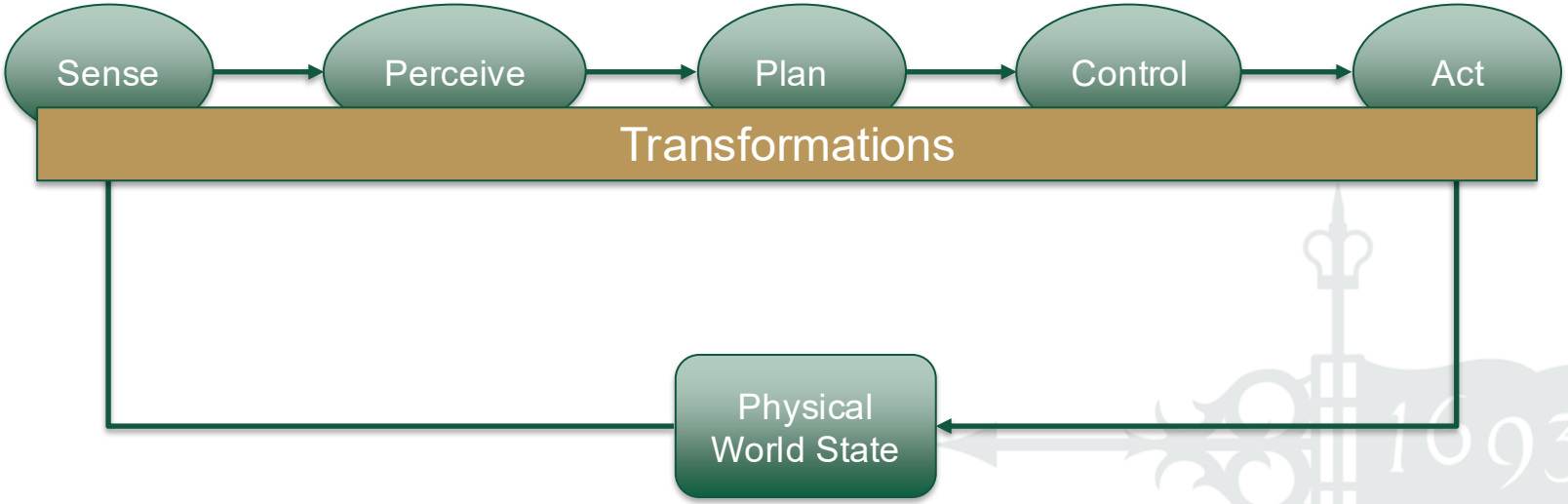
D4






https://nanomat.as.wm.edu/wiki/public/research/facilities/isc3

# Transformations



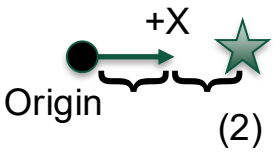
# Coordinate Systems

- Method to “label” locations
- Associate numbers with points
- Need:
  - Origin
  - Basis Vectors (positive, unit length)

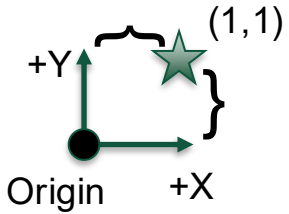
# Coordinate Systems

- Need:
  - Origin
  - Basis Vectors (positive, unit length)

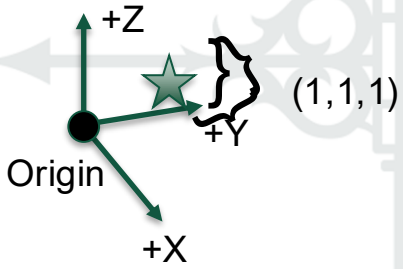
1D System




2D System

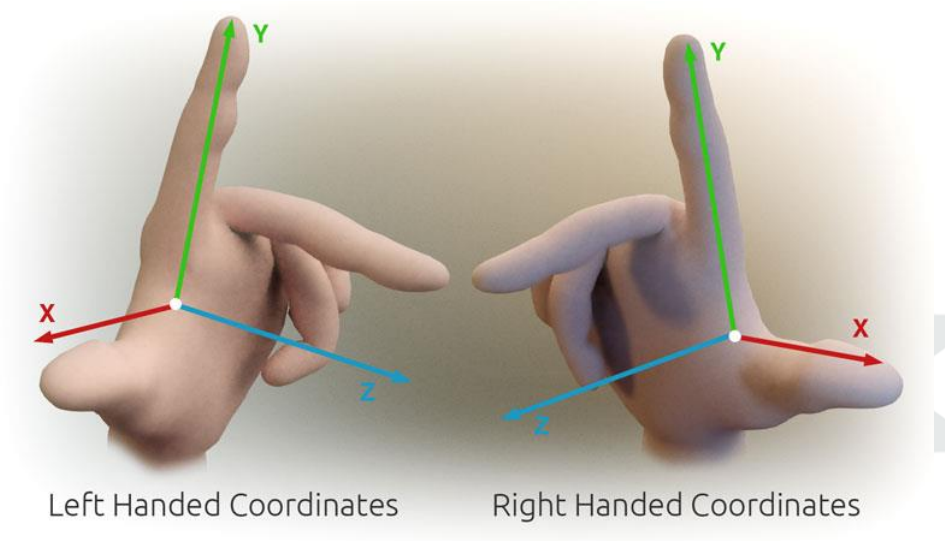



3D System



# Coordinate System Conventions

- Right-handed coordinate system
  - X – thumb
  - Y – pointer
  - Z – middle



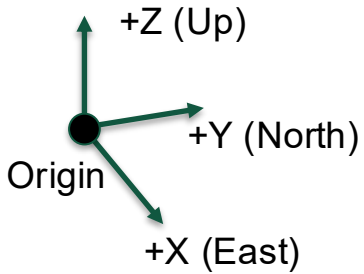

By Primalshell - Own work, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=27531327

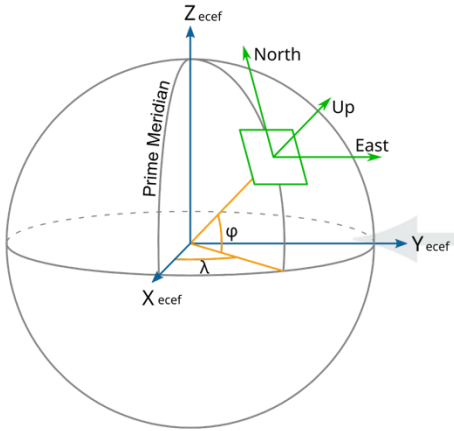
# Multiple Conventions

- 3D Reference Frames
- Multiple Conventions

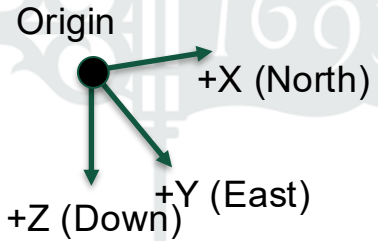



ENU – East North Up

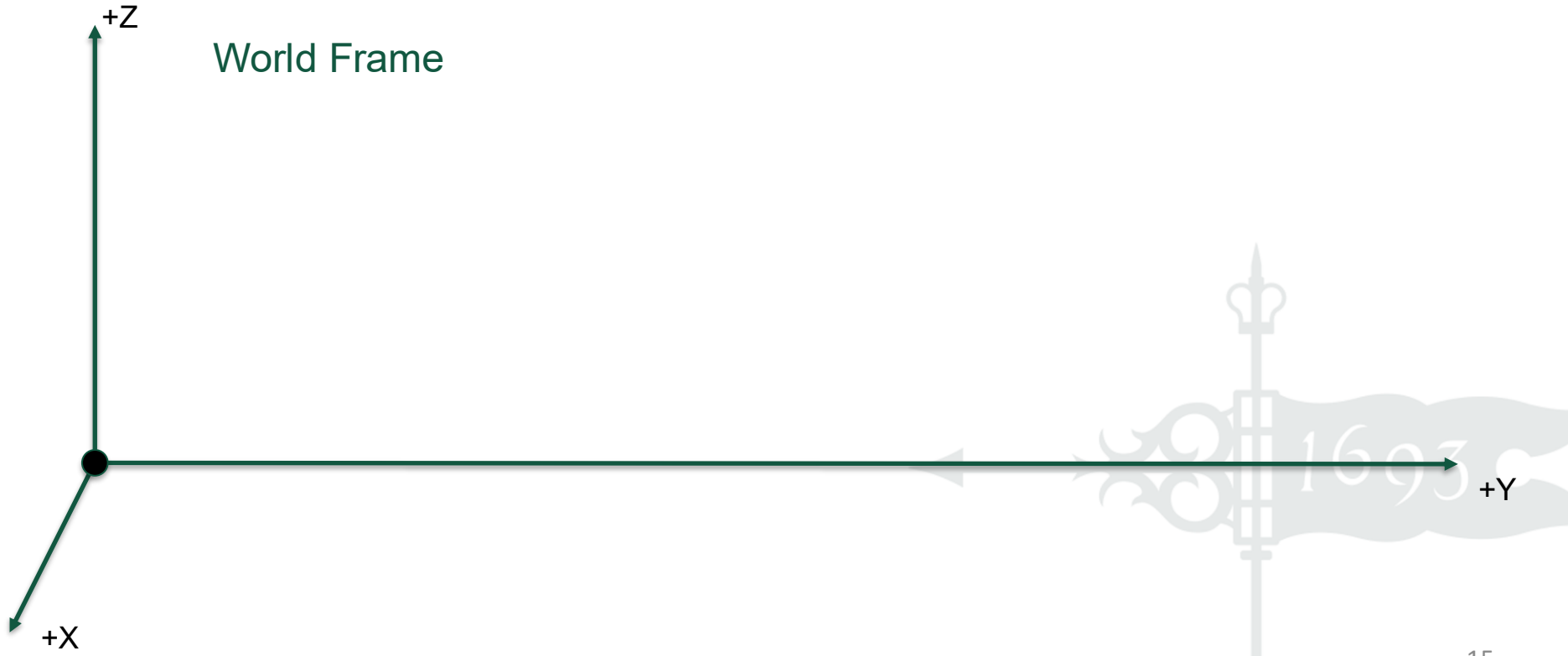


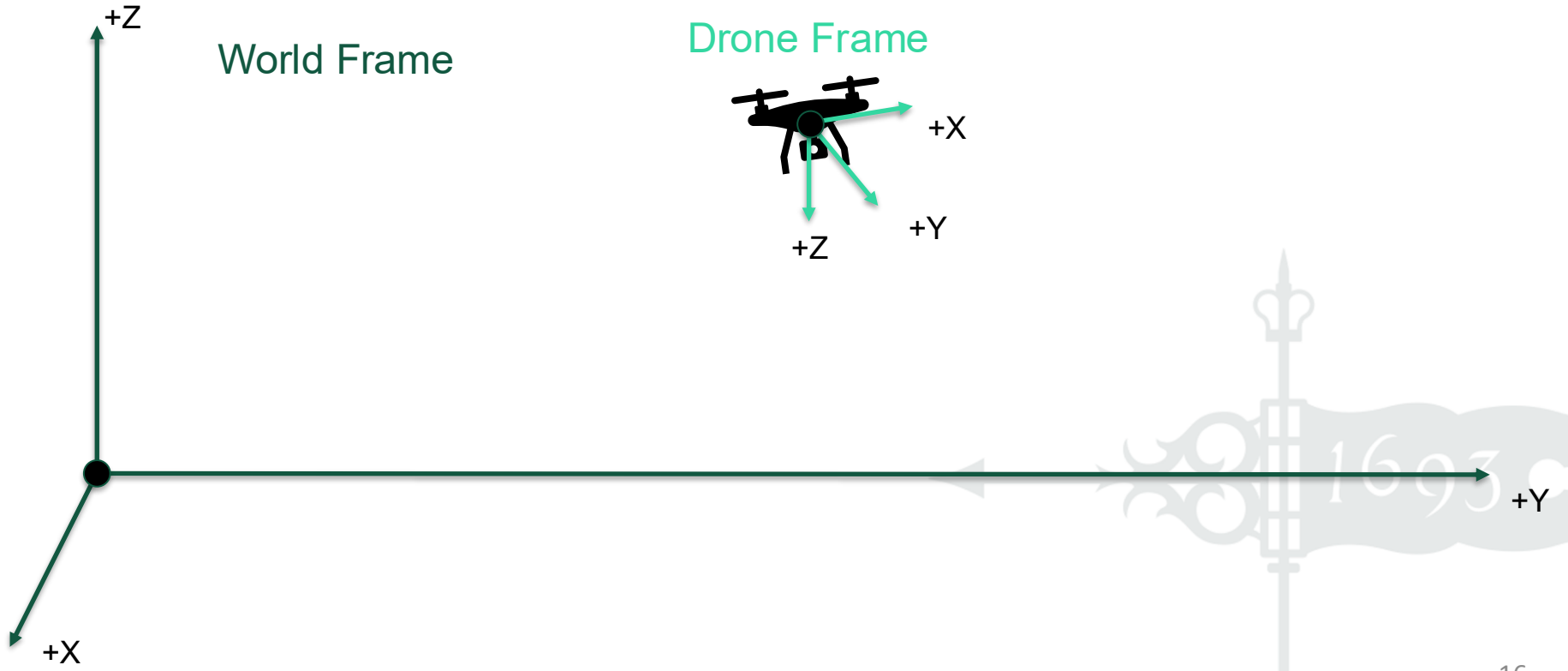

NED – North East Down



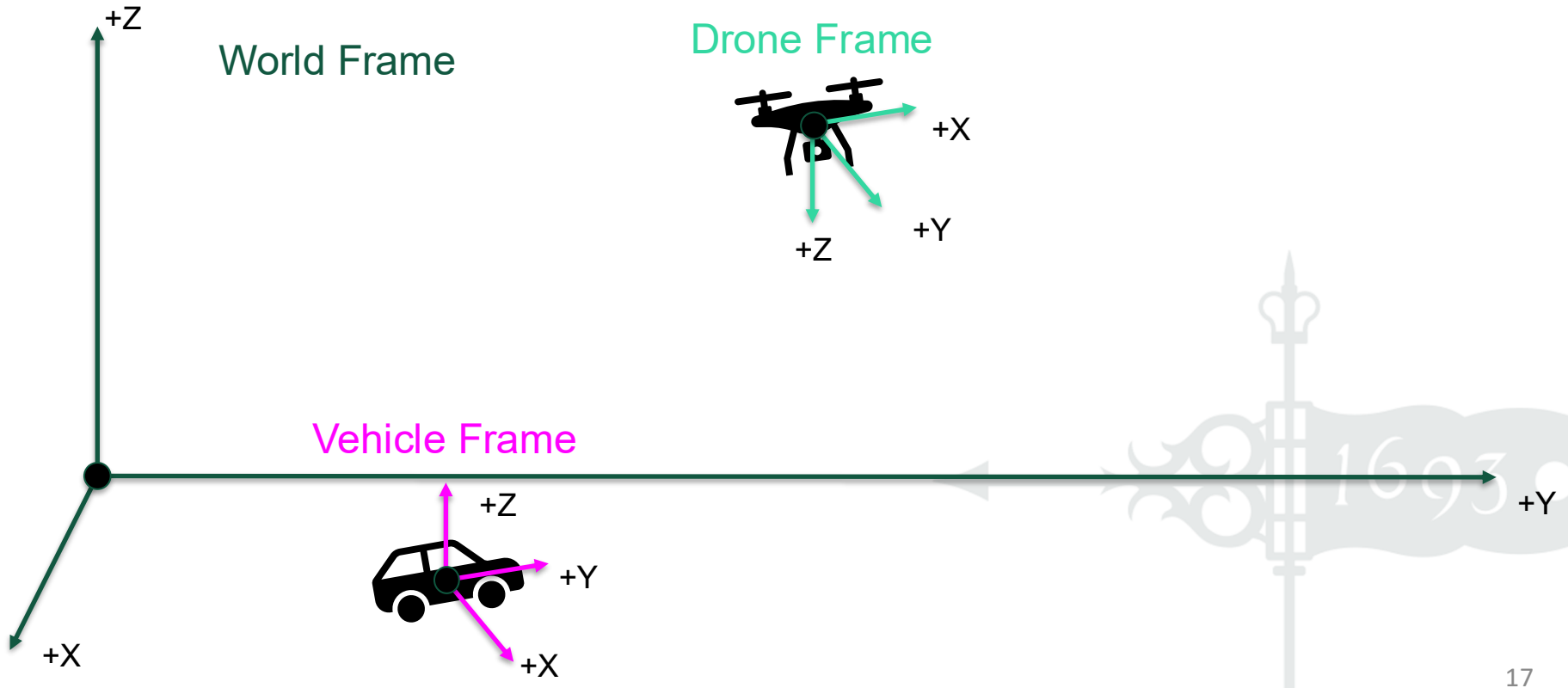
# Multiple Coordinate Frames



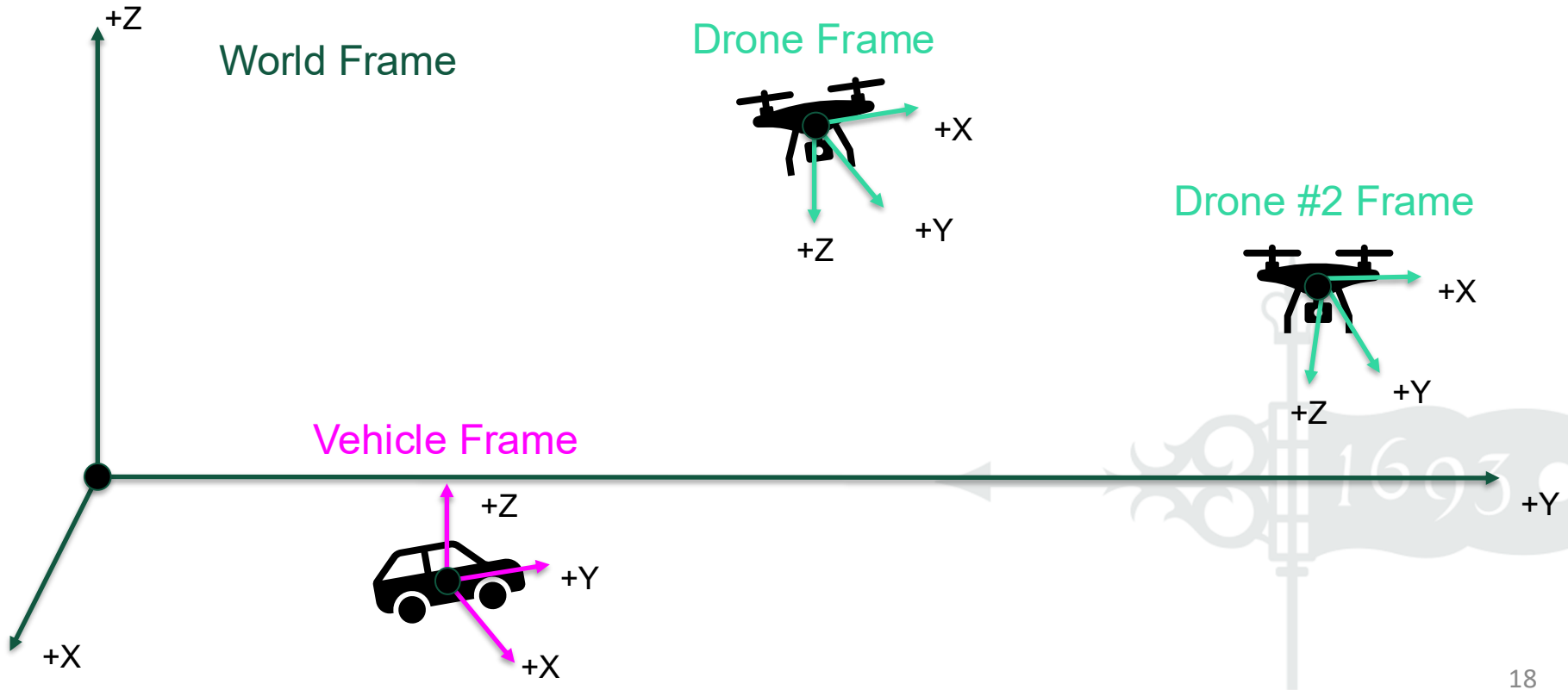
# Multiple Coordinate Frames



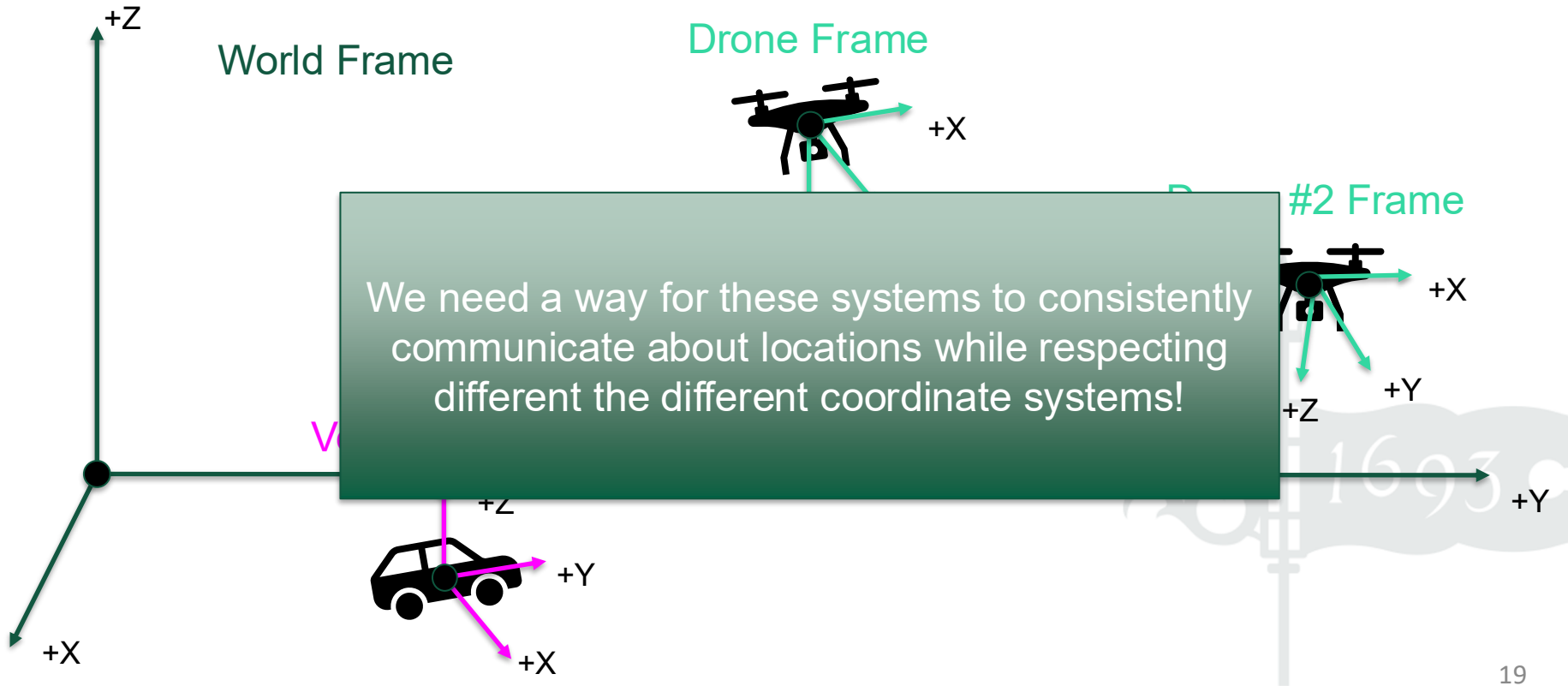
# Multiple Coordinate Frames



# Multiple Coordinate Frames



# Multiple Coordinate Frames

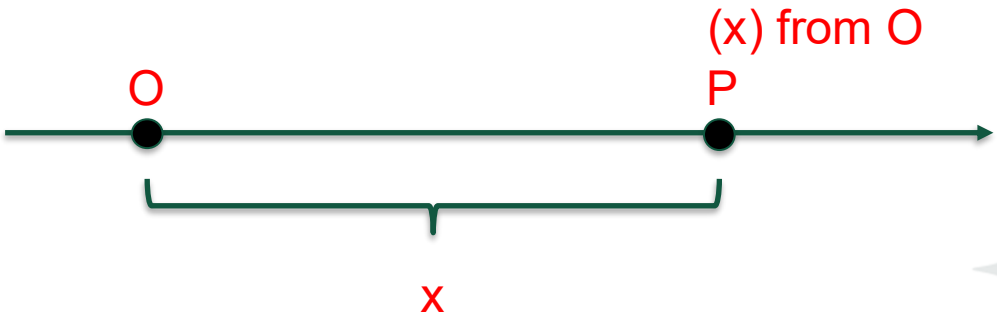



We need a way for these systems to consistently communicate about locations while respecting different the different coordinate systems!

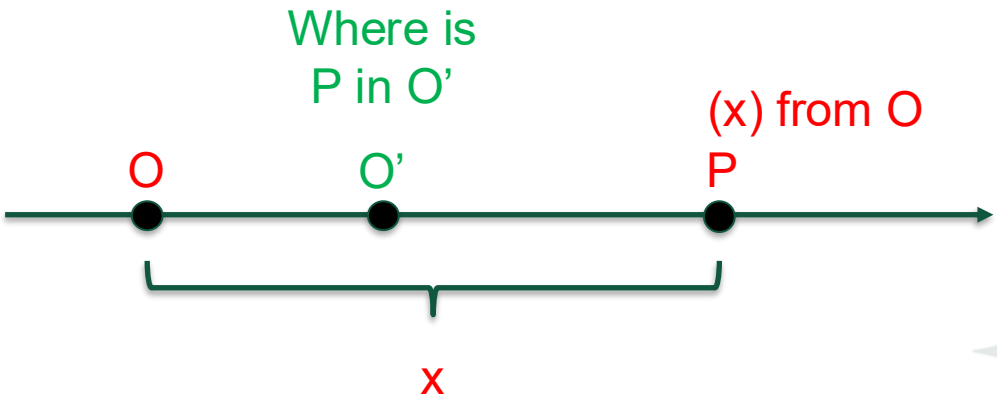
# Transforms

- Function to “transform” from one coordinate system to another
  - Input: point/vector in frame A, target frame B
  - Output: point/vector in frame B
- Pseudocode:
  - Translate
  - Rotate

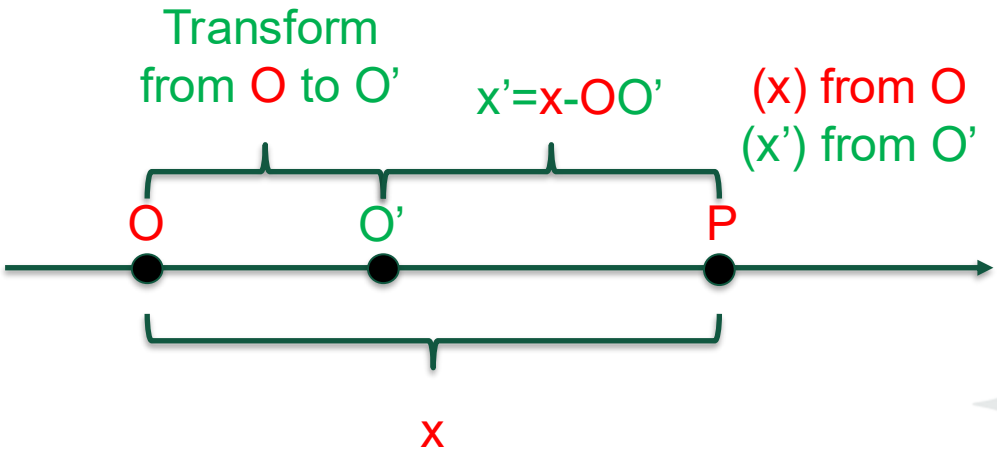
# 1D Transform



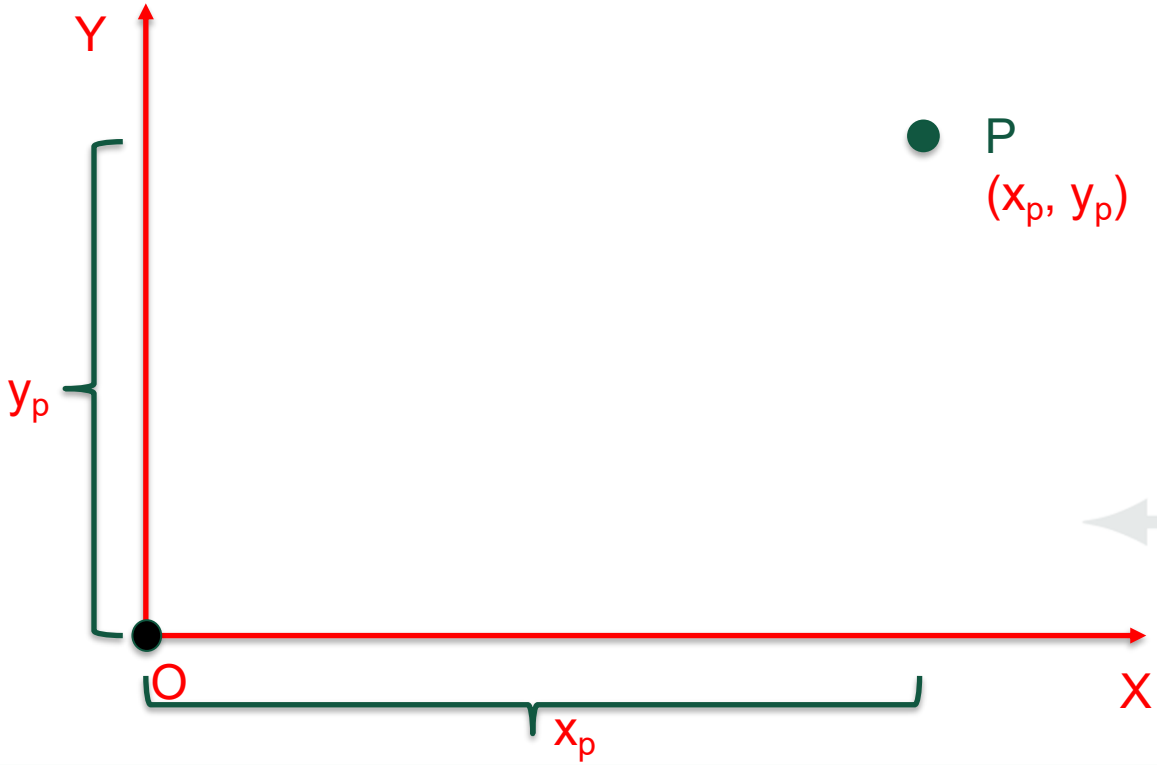
# 1D Transform



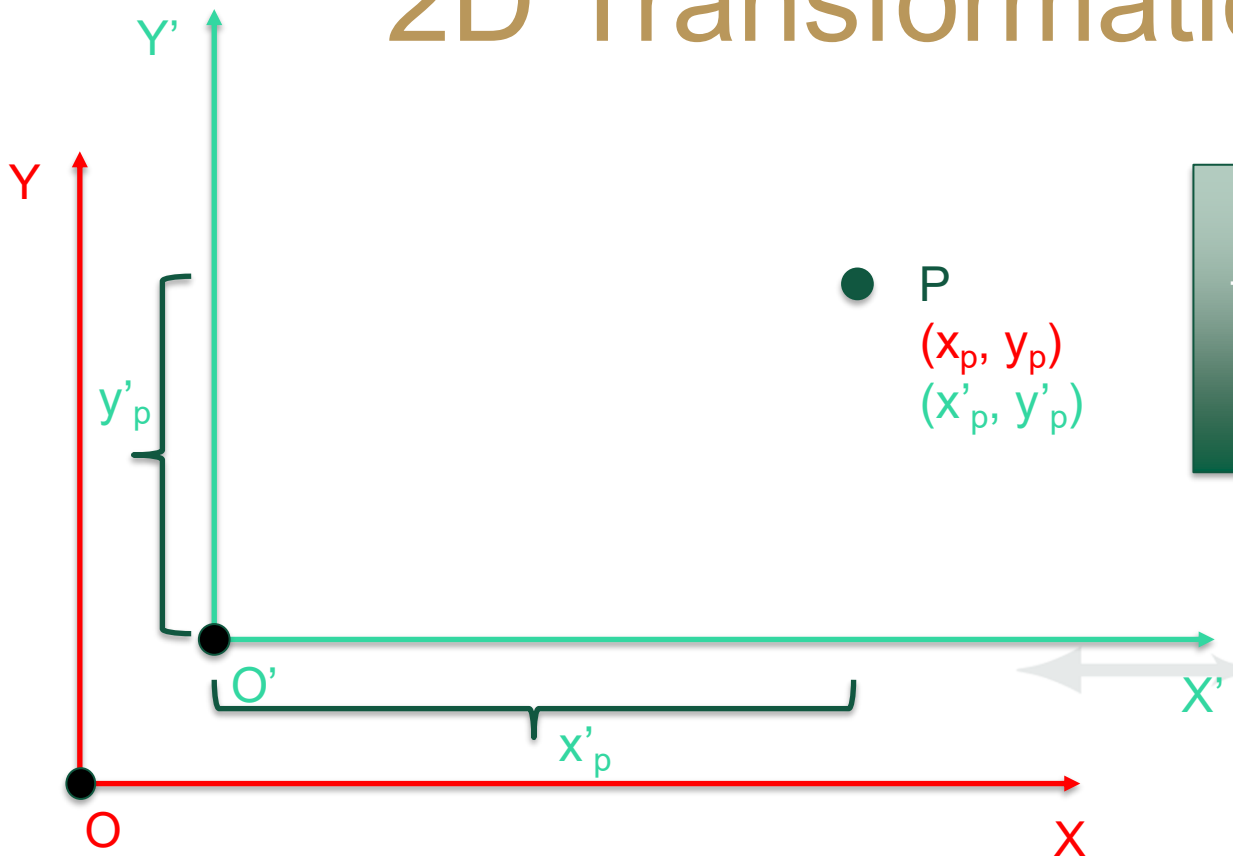
# 1D Transform



# 2D Transformation

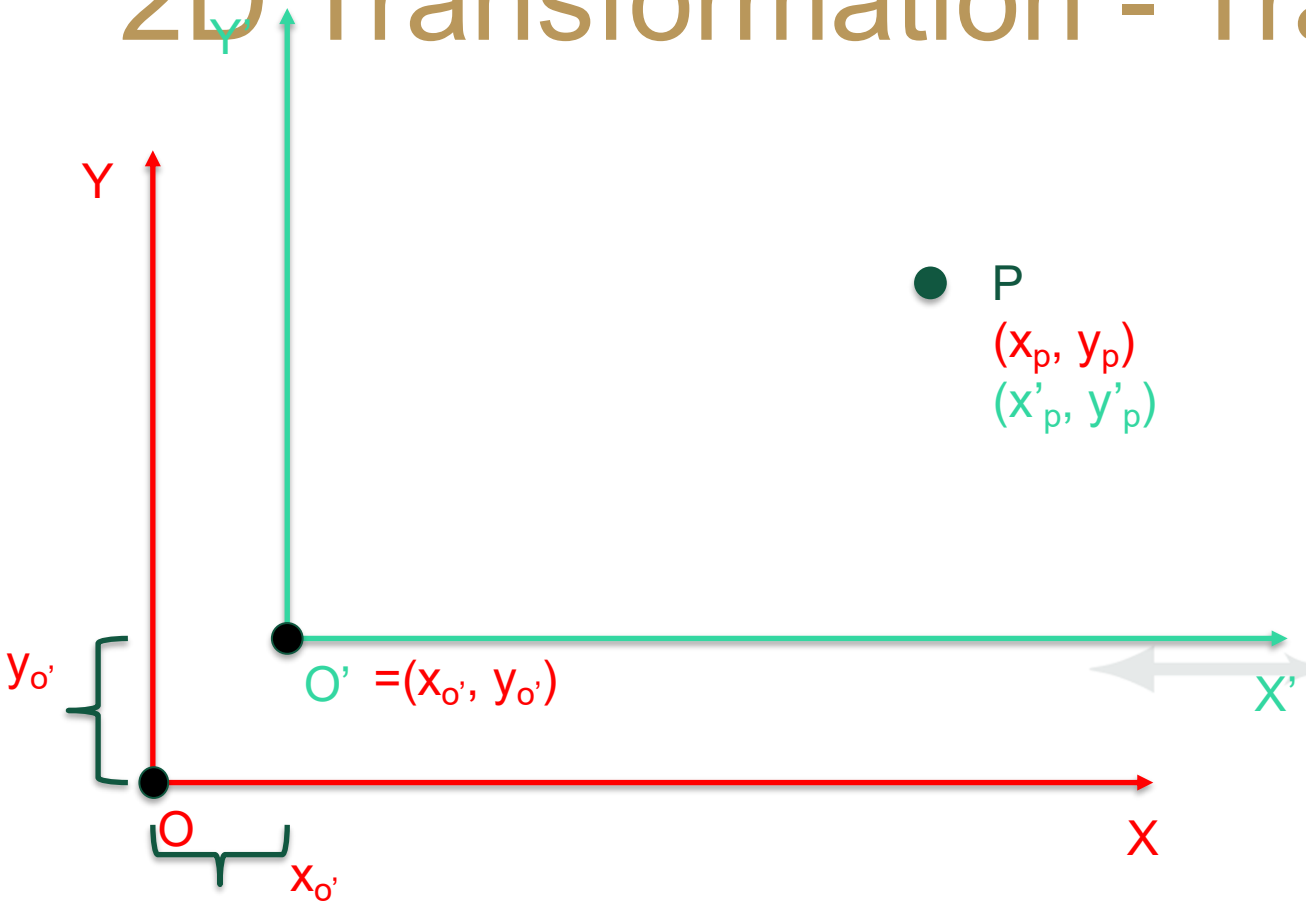


# 2D Transformation

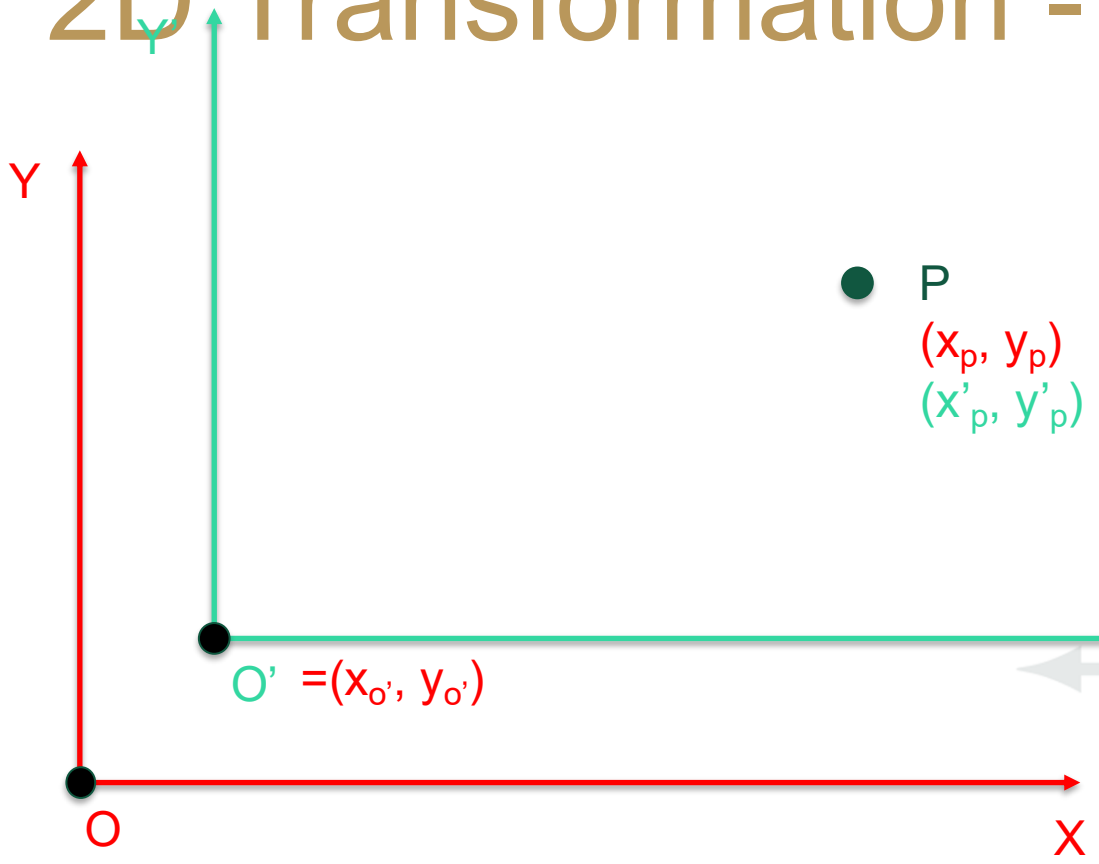



How do we relate these two coordinates so they can communicate?

# 2D Transformation - Translation



# 2D Transformation - Translation

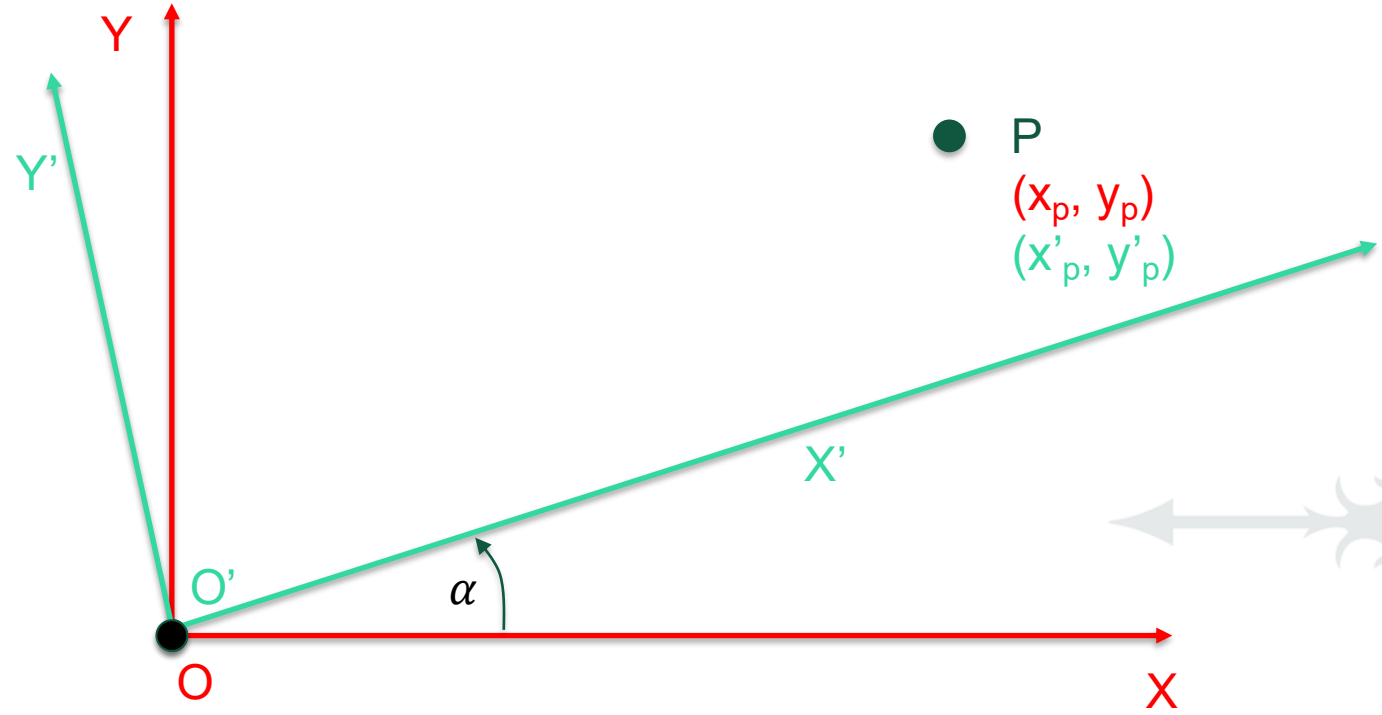



$$x'_p = x_p - x_{o'}$$
$$y'_p = y_p - y_{o'}$$

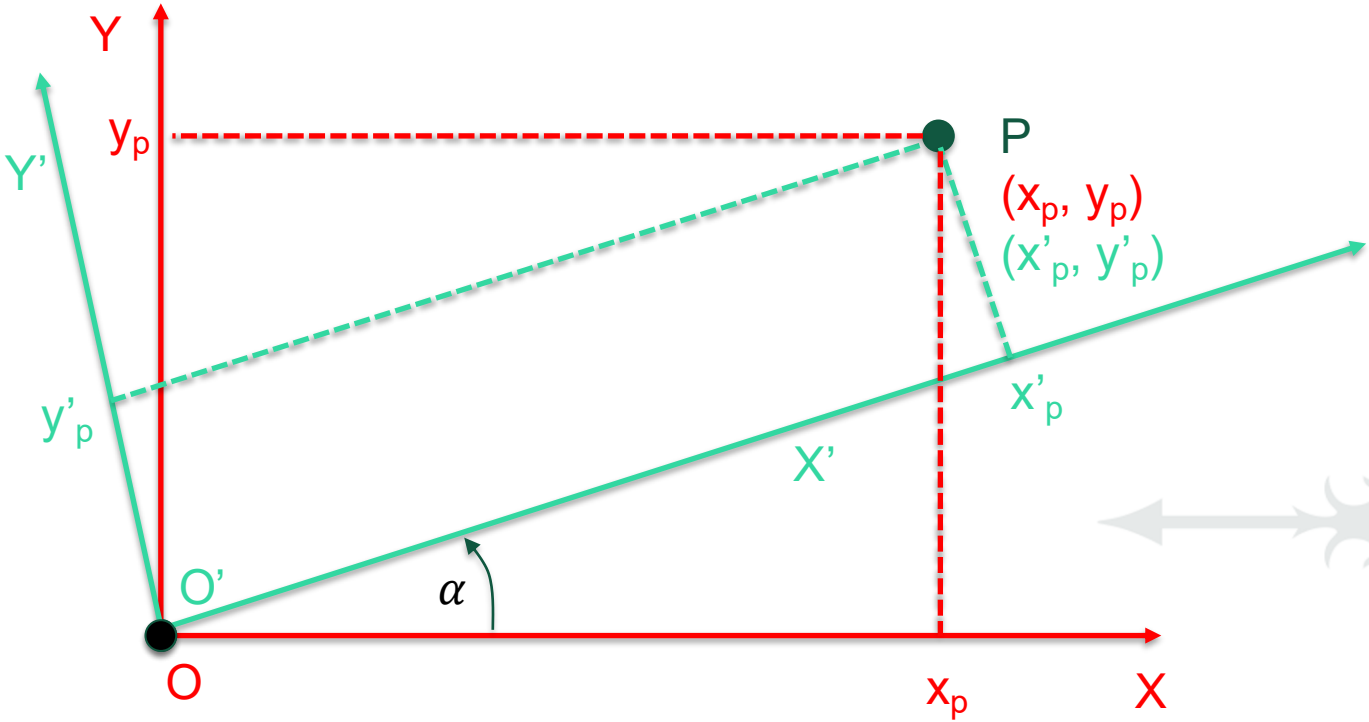
This is a system of equations!

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} x_o \\ y_o \end{bmatrix}$$

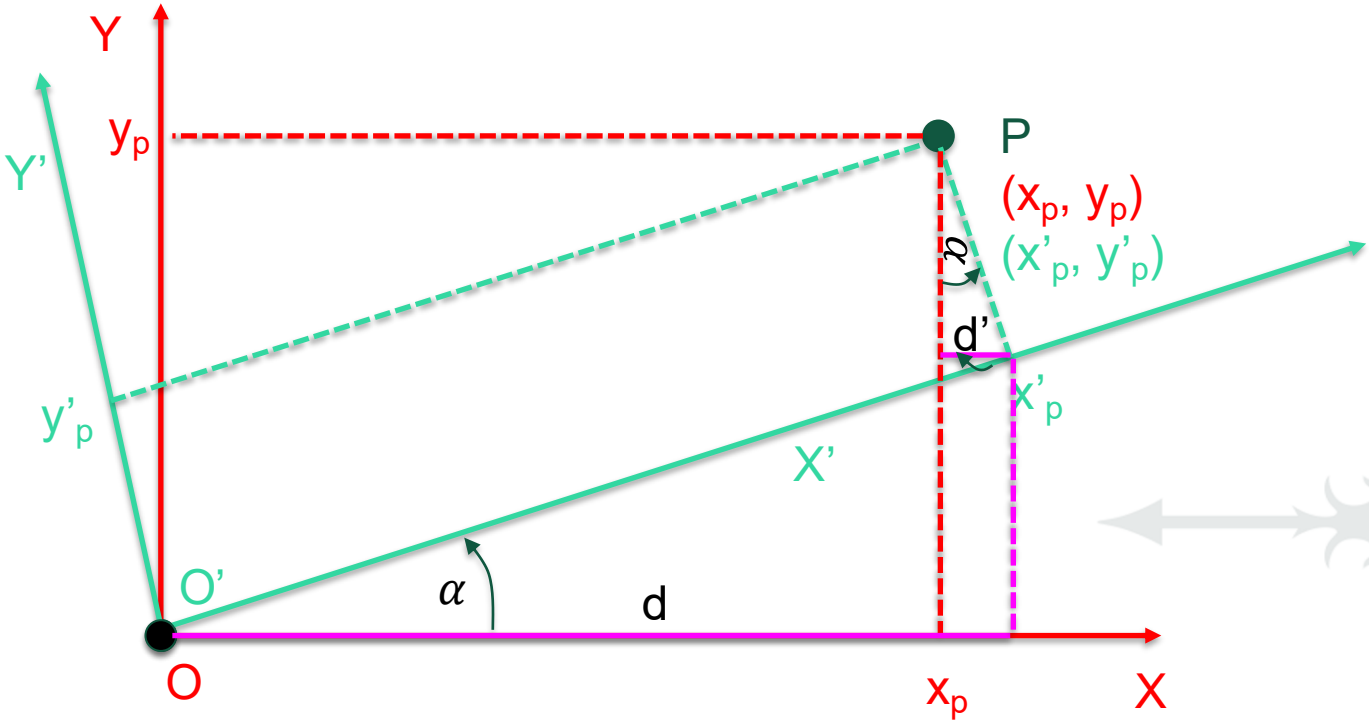
# 2D Transformation - Rotation



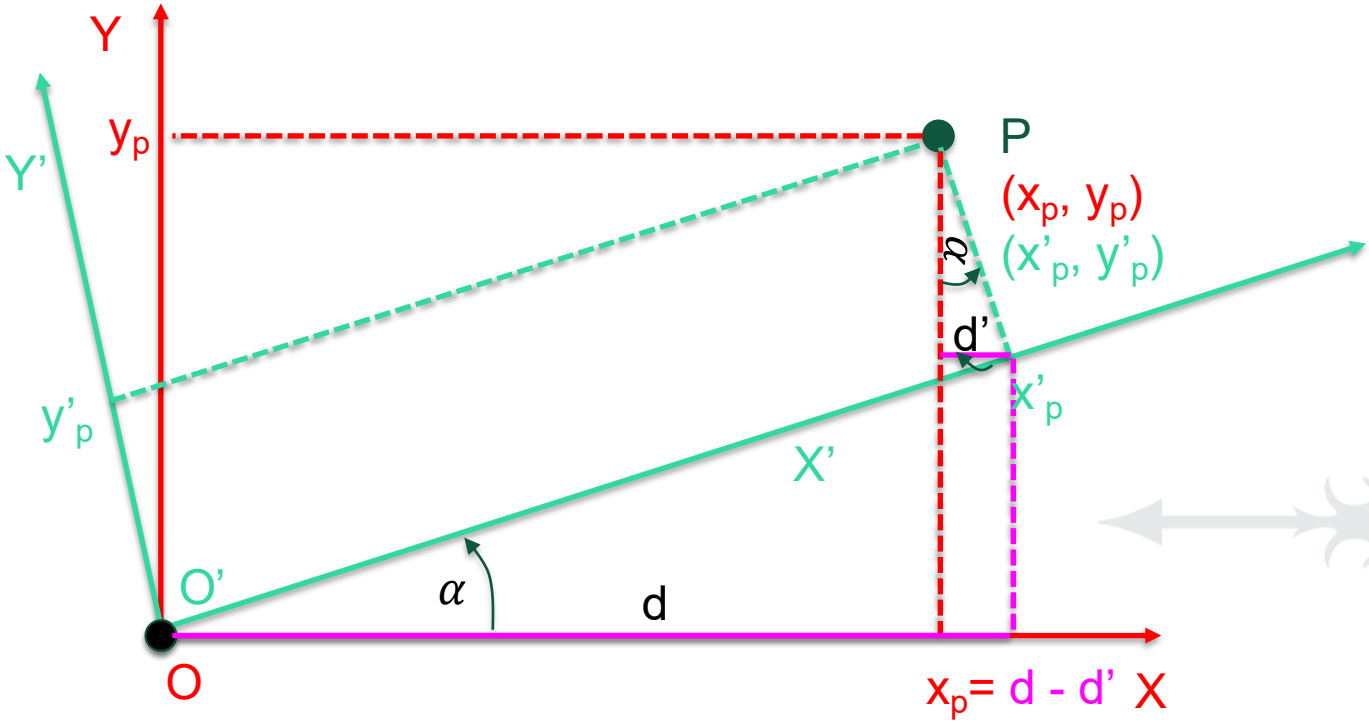
# 2D Transformation - Rotation



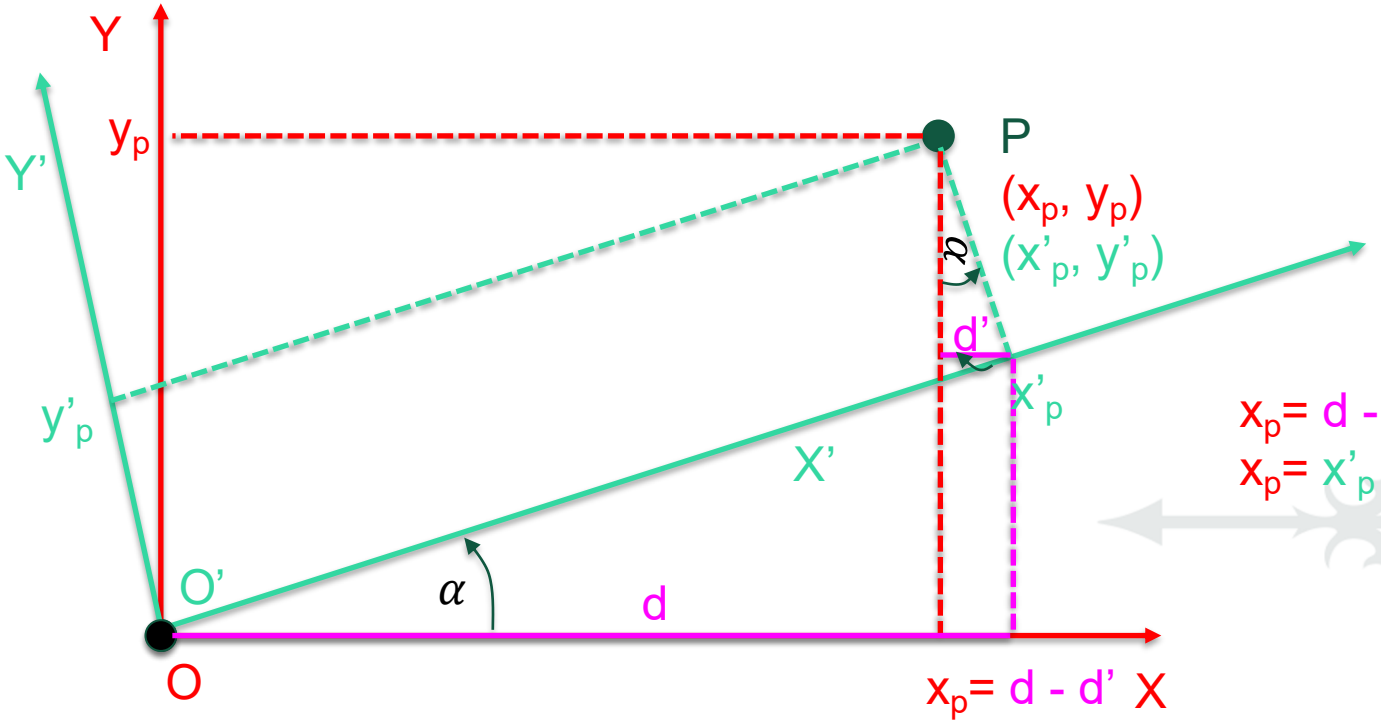
# 2D Transformation - Rotation



# 2D Transformation - Rotation



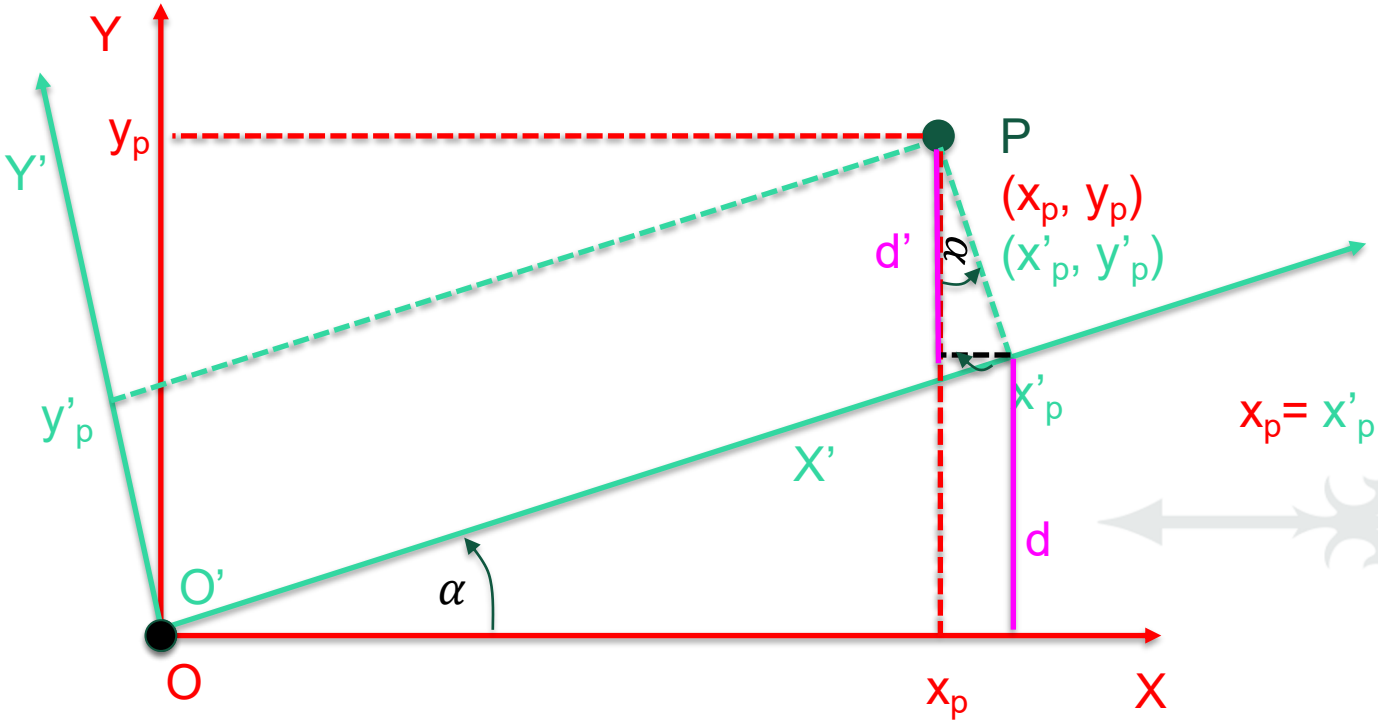
# 2D Transformation - Rotation




$x_p$= d - d'

$x_p$= $x'_p$ cos($\alpha$) - $y'_p$sin($\alpha$)

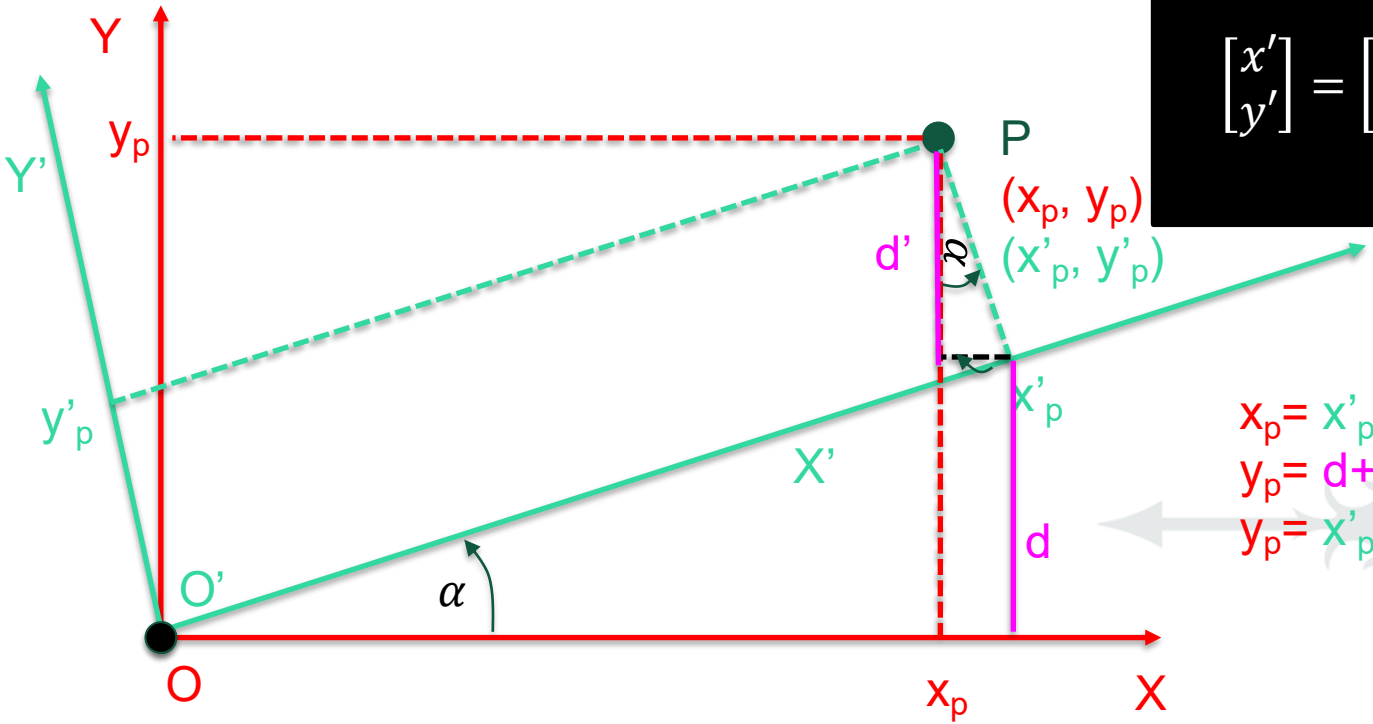
# 2D Transformation - Rotation




$$x_p = x'_p \cos(\alpha) - y'_p \sin(\alpha)$$

# 2D Transformation - Rotation

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

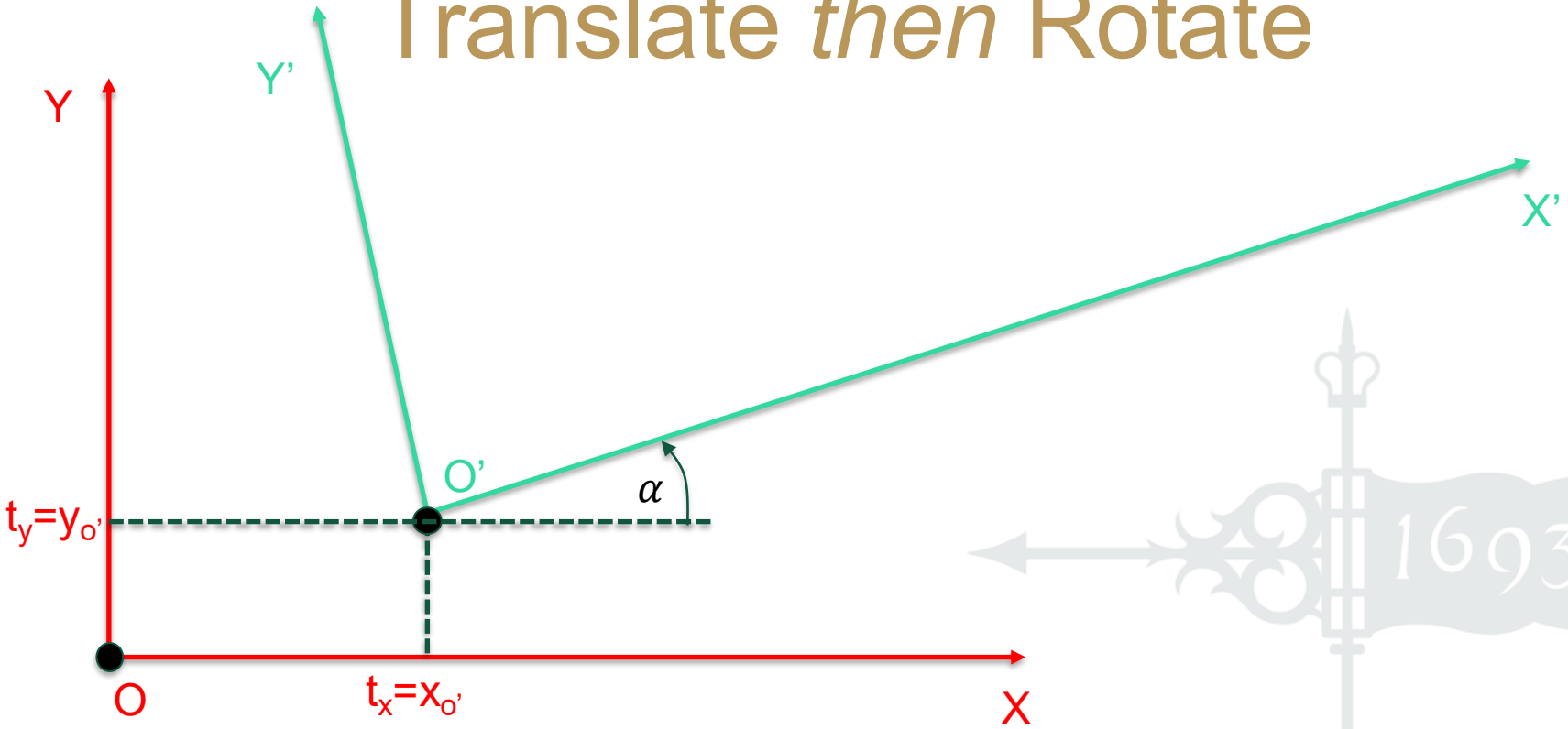



$x_p = x'_p \cos(\alpha) - y'_p \sin(\alpha)$

$y_p = d + d'$

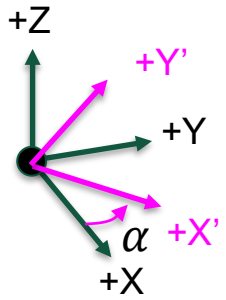
$y_p = x'_p \sin(\alpha) + y'_p \cos(\alpha)$

# 2D Transformation: Translate *then* Rotate



# 3D Transformations

- Rotation around Z




$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

# 3D Transformations

- Rotation around X, Y, Z all different
  - Composition of rotations
  - Multiplication of matrices is **not commutative**
  - Must agree on order

*Rotation about X*

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

*Rotation about Y*

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} \cos(\alpha) & 0 & \sin(\alpha) \\ 0 & 1 & 0 \\ -\sin(\alpha) & 0 & \cos(\alpha) \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

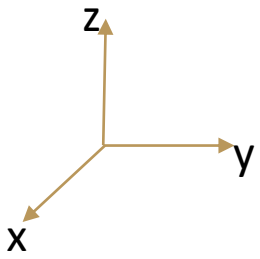
*Rotation about Z*

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

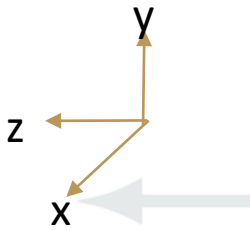
# 3D Transformations

- Rotation around X, Y, Z all different
  - Composition of rotations
  - Multiplication of matrices is **not commutative**
  - Must agree on order

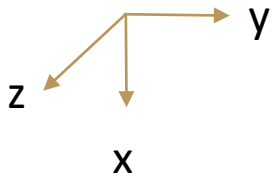
90 degrees
counter-clockwise



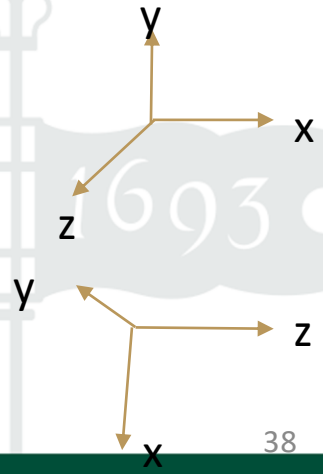

Rx




Ry




Ry

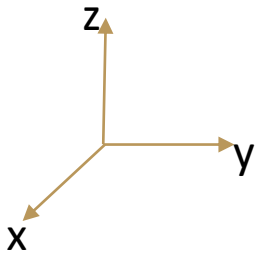



Rx

# 3D Transformations

- Rotation around X, Y, Z all different
  - Composition of rotations
  - Multiplication of matrices is **not commutative**
  - Must agree on order

90 degrees counter-clockwise



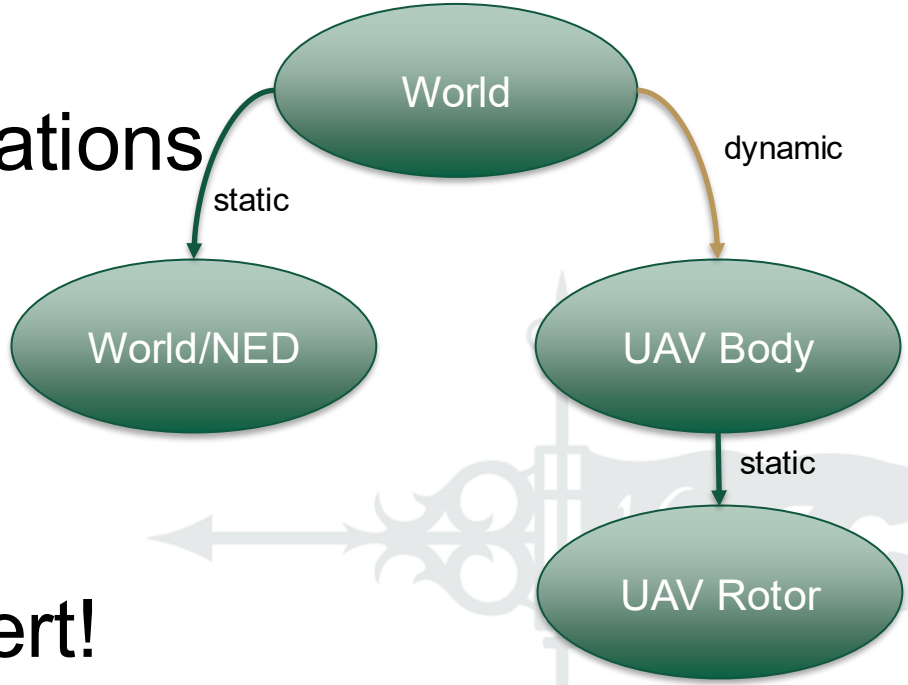

Rx

Ry

ROS has transformation libraries that automatically handle the math from one coordinate frame to another!

# Frames in ROS

- tf2 api
- Define frames and relations
  - Tree!
- Transforms between
  - Static
  - Dynamic
- Helper library to convert!



# Transformations

- Physical data without a coordinate frame is **useless!**

- The frame is part of the data type!