

MARCOS DE REFERENCIA EN CG

Computación Gráfica

World Frame

- Marco del mundo (world frame): \vec{w}^t
- de mano derecha,
- no lo vamos a alterar.
- describimos otros marcos respecto al marco del mundo.
- si expresamos un vector de coordenadas respecto a este marco, les llamamos coordenadas del mundo (world coordinates).

Object Frame

- Asociamos uno a cada objeto de la escena.
- vector de coordenadas de objeto.
- Para transformar un objeto transformamos su marco de referencia de objeto.
- La relación entre el marco del objeto y el marco del mundo se puede escribir con una transformación afín (matriz 4x4) O .

$$\vec{o}^t = \vec{w}^t O$$

- Para mover un objeto en la escena cambiamos entonces la matriz O .

Camera Frame

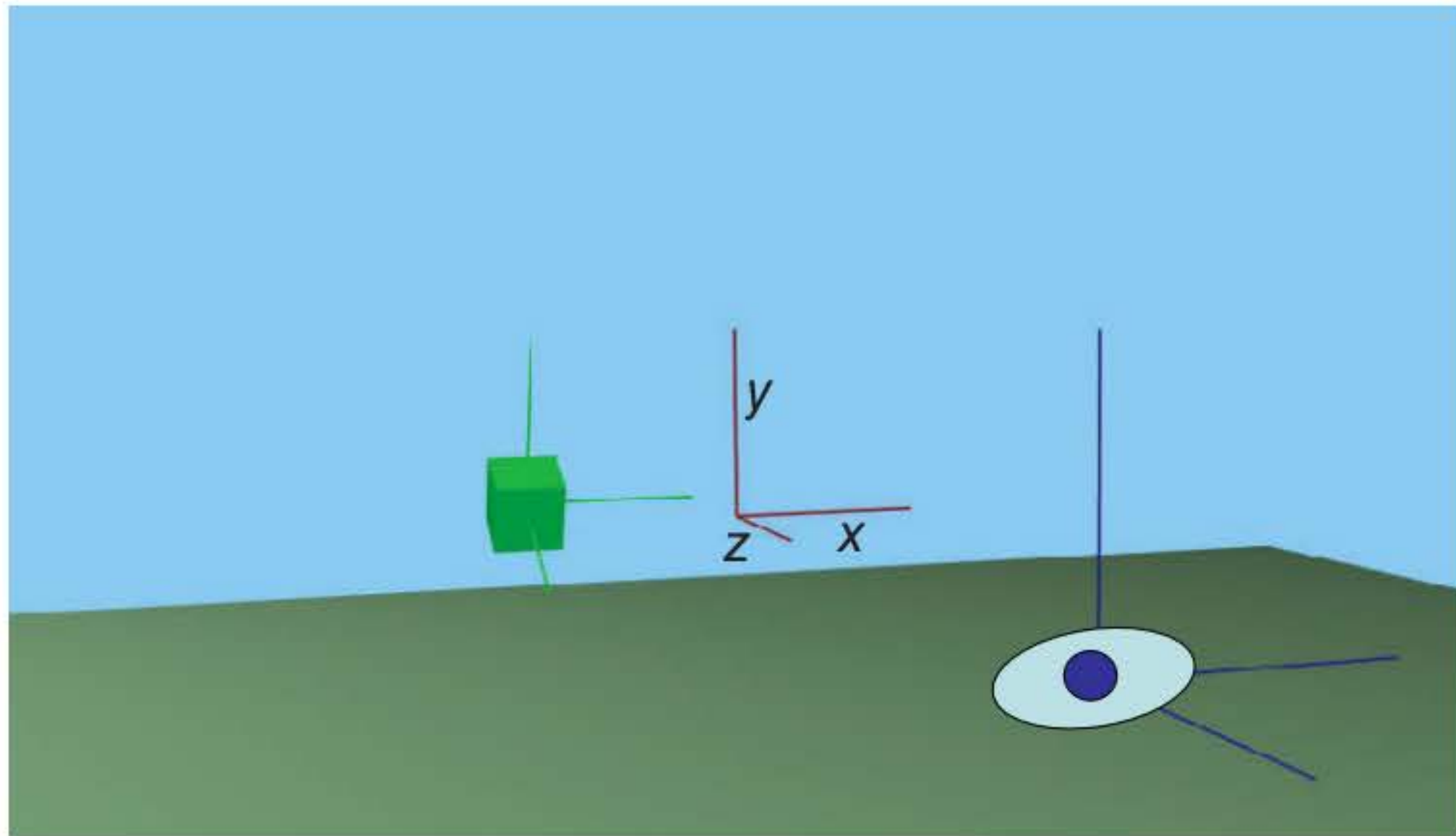
- En el mundo real, queremos crear una imagen 2D de un ambiente 3D.
- Para esto ponemos una cámara en la escena.
- La posición de cada objeto en la imagen se basa en su relación 3D con la cámara (sus coordenadas respecto una base apropiada).
- Definimos un marco de referencia \mathbf{e} llamado “eye frame”.
- Se interpreta un ojo viendo hacia su eje z negativo para hacer una foto.
- El marco de la cámara se relaciona con una matriz de transformación 4x4 E .

$$\tilde{p} = \vec{o}^t \mathbf{c} = \vec{w}^t O \mathbf{c} = \vec{e}^t E^{-1} O \mathbf{c}$$

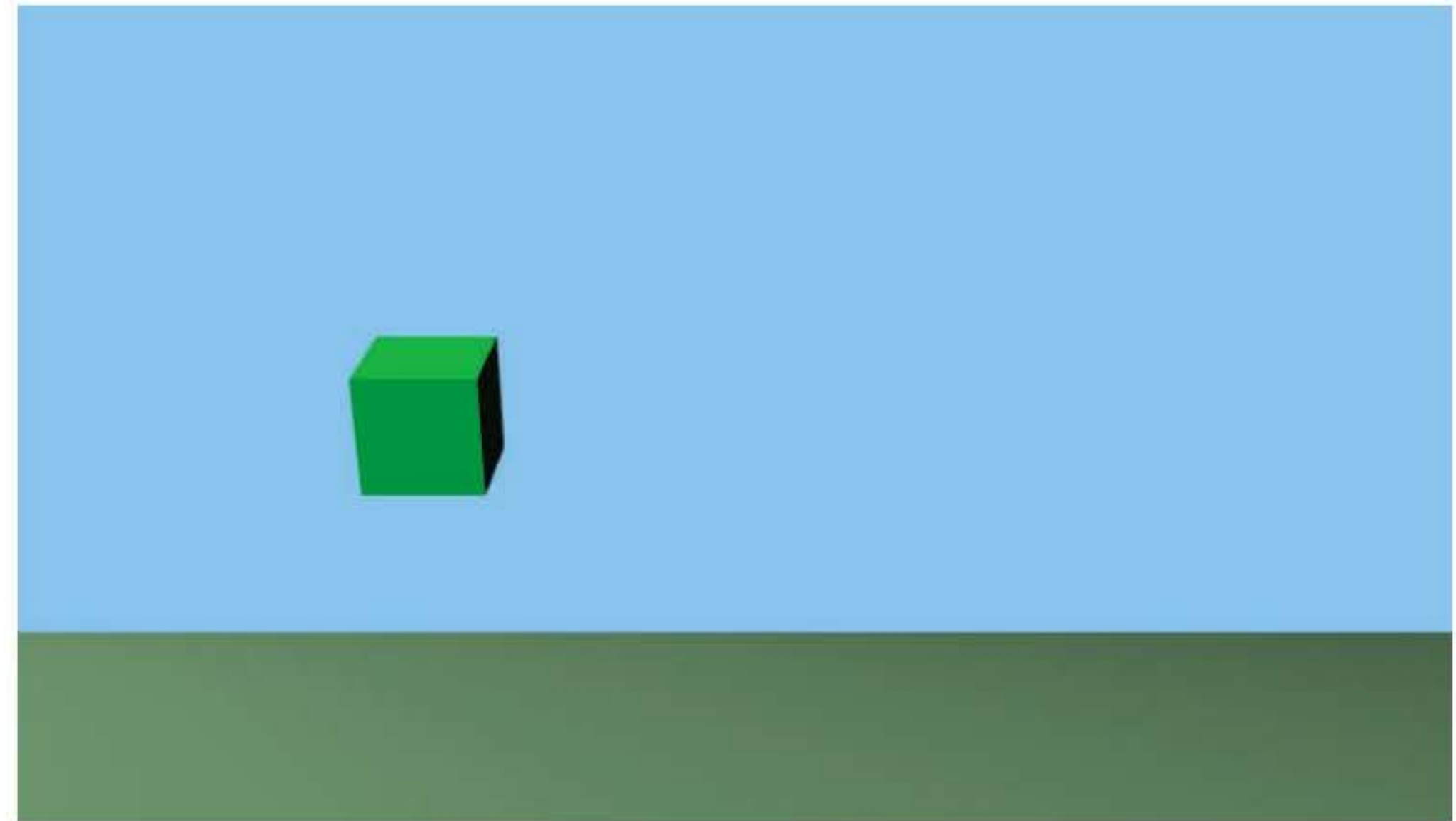
$$\vec{e}^t = \vec{w}^t E$$

$$\begin{bmatrix} x_e \\ y_e \\ z_e \\ 1 \end{bmatrix} = E^{-1} O \begin{bmatrix} x_o \\ y_o \\ z_o \\ 1 \end{bmatrix}$$

Marcos de Referencia



(a) The frames



(b) The eye's view

Figure 5.1

The world frame is in red, the object frame is in green, and the eye frame is in blue. The eye is looking down its negative z axis toward the object.

Moviendo objetos

- Movemos un objeto actualizando su marco de referencia (a través de la matriz O).
- Si queremos aplicar una transformación M al marco de referencia del objeto respecto a un marco auxiliar $\vec{a}^t = \vec{w}^t A$.

$$\vec{o}^t$$

$$= \vec{w}^t O$$

$$= \vec{a}^t A^{-1} O$$

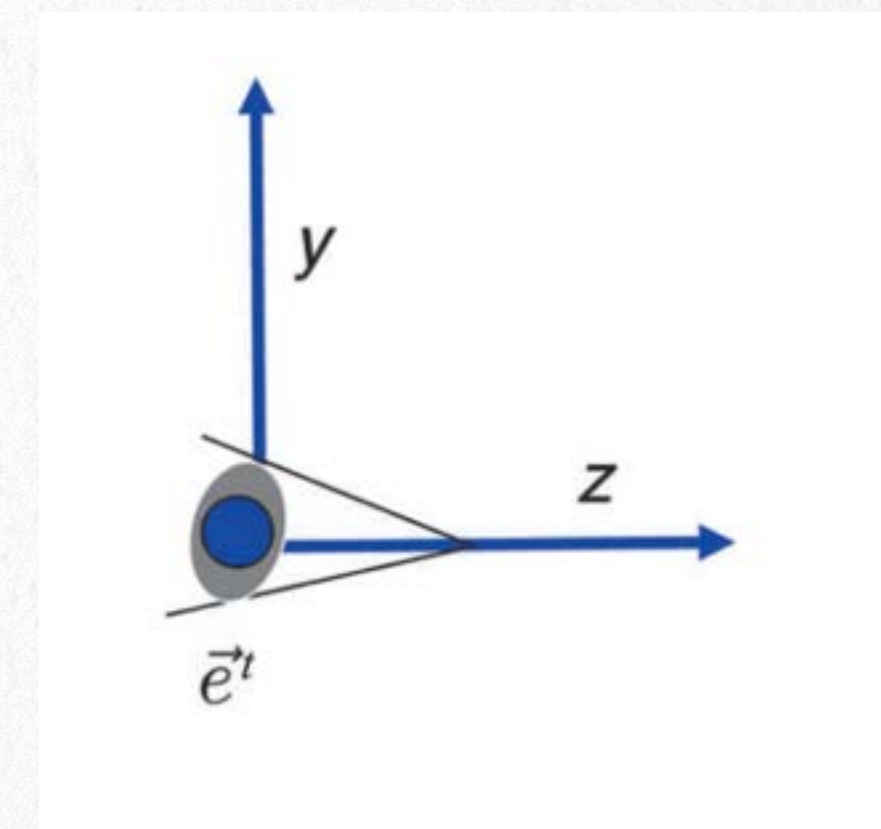
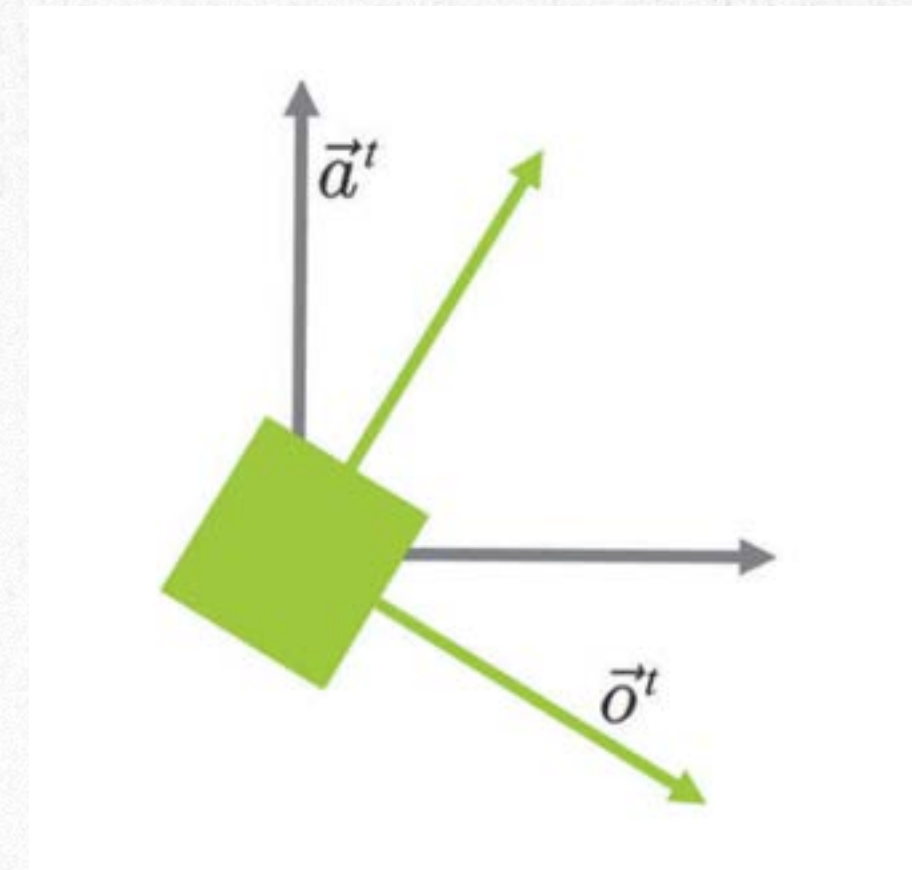
$$\Rightarrow \vec{a}^t M A^{-1} O$$

$$= \vec{w}^t A M A^{-1} O$$

- En código implementamos: $O \leftarrow A M A^{-1} O$

Moviendo objetos

- ¿Qué marco de referencia \vec{a}^t elegir?
 - Aplicar la transformación del objeto respecto al objeto (¿orientación de ejes?)
 - Transformar al objeto respecto el marco de la cámara (¿origen del marco?)
 - Marco de referencia auxiliar con origen en el marco del objeto y ejes con la orientación del marco de la cámara.
-
- Empezar en WCS.
 - Traducir WCS al centro del OCS.
 - Rotar el marco resultante en la dirección del ECS.



Moviendo los objetos

$$O = (O)_T(O)_R$$

$$E = (E)_T(E)_R$$

$$\vec{a}^t = \vec{w}^t(O)_T(E)_R$$

$$A = (O)_T(E)_R$$

Moviendo objetos

- Rotar un objeto alrededor de su propio centro utilizando un eje de rotación \vec{k} con coordenadas \mathbf{k} respecto a \vec{e}^t .
- Encontrar el valor de \mathbf{k}' (coordenadas del vector \vec{k} respecto al marco de referencia \vec{o}^t).
- Hacer la conversión de parametrización eje-ángulo a matriz de rotación M' (ambas respecto a \vec{o}^t).
- Actualizar la matriz O : $O \leftarrow OM'$

Moviendo la cámara

- Modificar el marco de referencia \vec{e}^t (actualizar la matriz E).
- Utilizando el mismo marco que para los objetos, la cámara gira alrededor del centro del objeto.
- Utilizando un marco ego-céntrico (respecto a la cámara misma). $E \leftarrow EM$
- ej. juegos primera persona.

Lookat

- Describir el marco de referencia de la cámara $\vec{e}^t = \vec{w}^t E$ con una posición del ojo \tilde{p} , un punto hacia donde el ojo mira \tilde{q} y el vector “hacia arriba” \vec{u} describiendo la dirección vertical de la cámara.
- En el marco de referencia del mundo:

$$\mathbf{z} = \text{normalize}(\mathbf{q} - \mathbf{p})$$

$$\mathbf{y} = \text{normalize}(\mathbf{u})$$

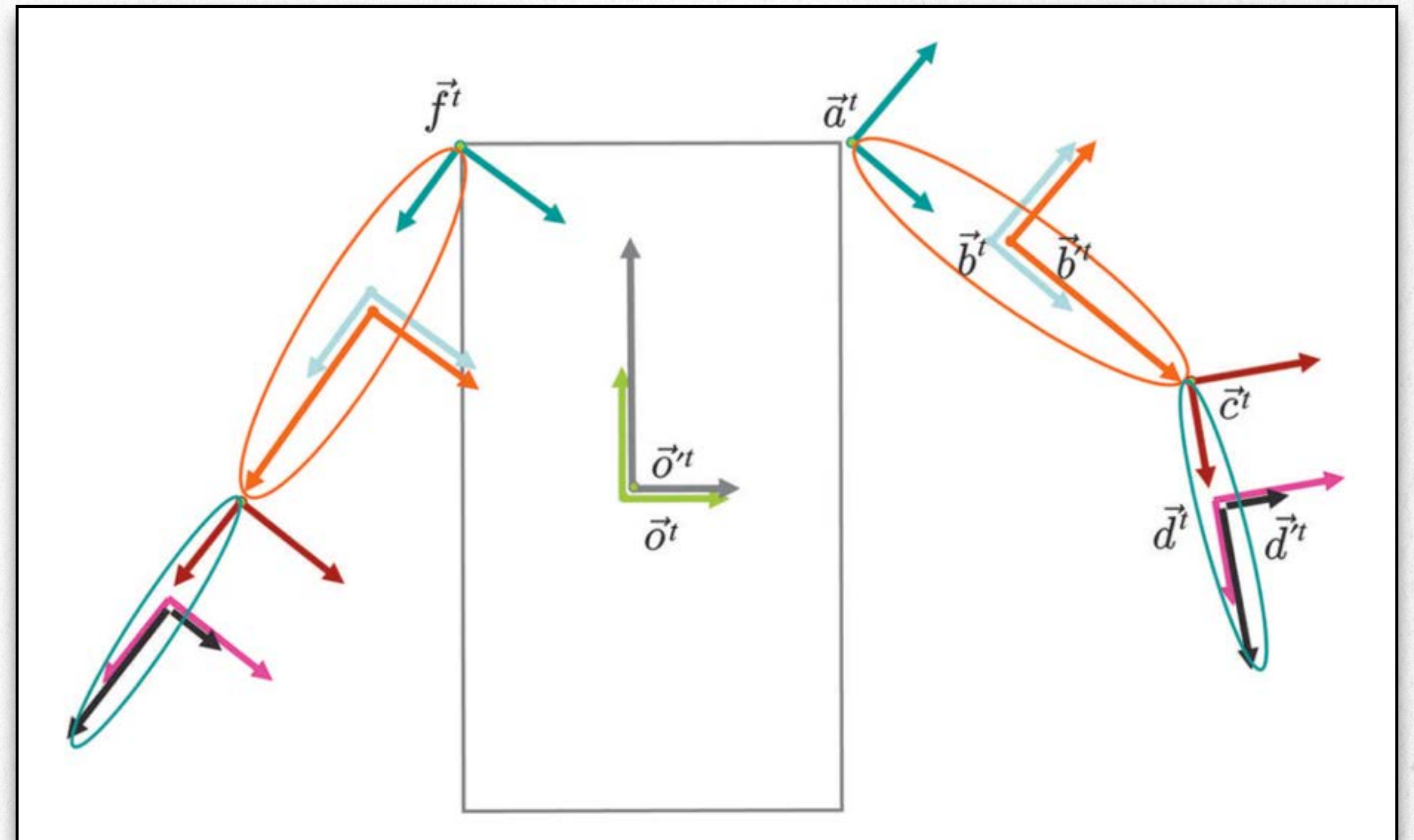
$$\mathbf{x} = \mathbf{y} \times \mathbf{z}$$

$$\text{normalize}(\mathbf{c}) = \frac{\mathbf{c}}{\sqrt{c_1^2 + c_2^2 + c_3^2}}$$

$$E = \begin{bmatrix} x_1 & y_1 & z_1 & p_1 \\ x_2 & y_2 & z_2 & p_2 \\ x_3 & y_3 & z_3 & p_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Escenas u objetos jerárquicos

- Objetos o escenas se ensamblan con un marco de referencia fijo al cual se atan subobjetos con sus propios marcos de referencia.
- Queremos modelar el movimiento del objeto o escena completa así como sus subpartes.
- Ej.



Estructura de pilas para jerarquías

- Sirve para modelar jerarquías.
- **push(M)** crea una nueva matriz en el nivel más alto de la pila que es una copia de la matriz que estaba previamente en ese lugar.
- multiplica por la derecha esta matriz con la matriz argumento **M**.
- **pop(M)** elimina la matriz lo más arriba de la pila.
- Cuando “bajamos” en la jerarquía hacia un subobjeto se hace una operación **push**.
- Cuando regresamos de descendiente a padre se hace una operación **pop**.

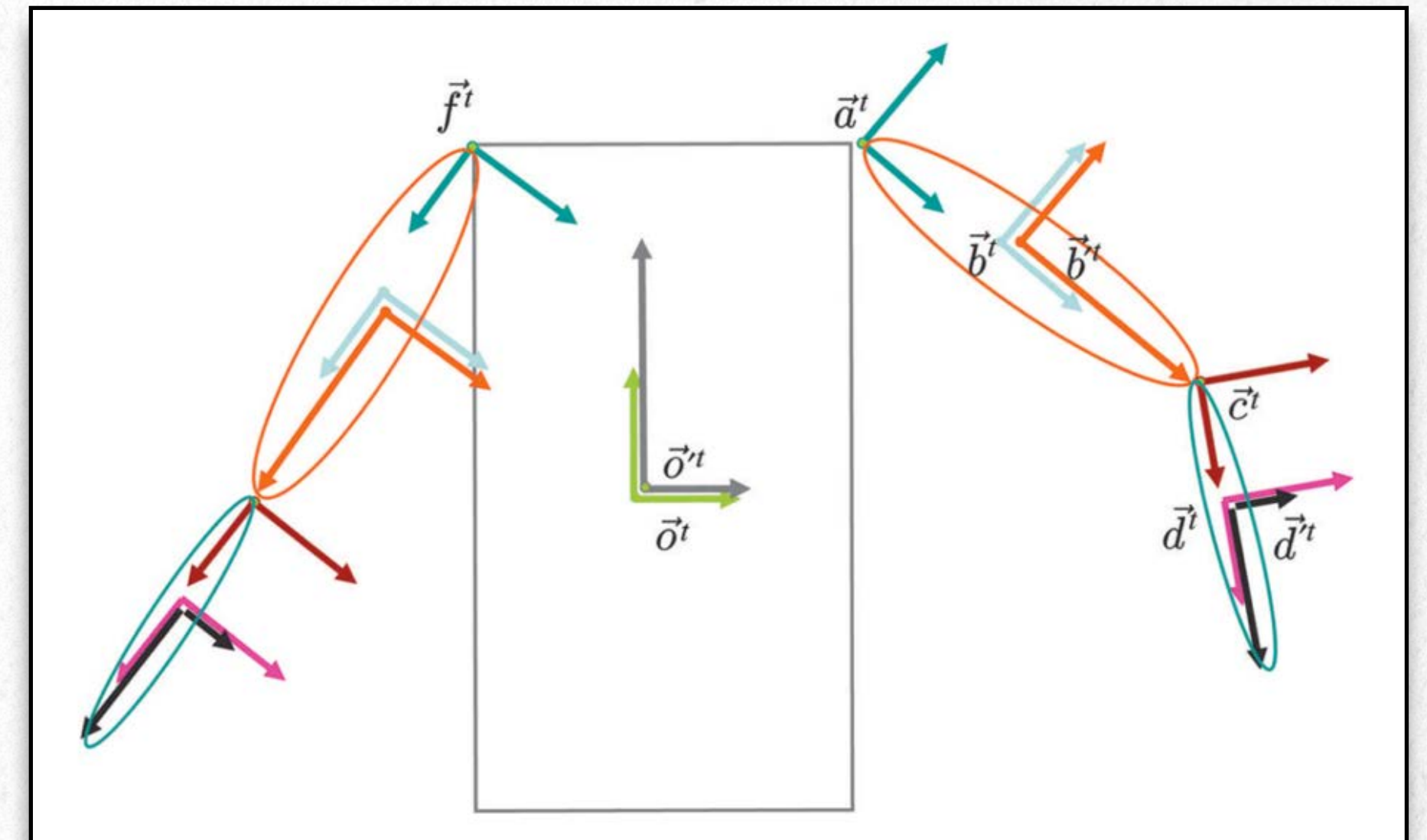
Escenas u objetos jerárquicos

```
...
matrixStack.initialize(inv(E));
matrixStack.push(0);
  matrixStack.push(0');
    draw(matrixStack.top(), cube); //body
  matrixStack.pop(); //0'

matrixStack.push(A);
  matrixStack.push(B);
    matrixStack.push(B');
      draw(matrixStack.top(), sphere); //upper arm
    matrixStack.pop(); // B'

matrixStack.push(C);
  matrixStack.push(C');
    draw(matrixStack.top(), sphere); //lower arm
  matrixStack.pop(); // C'
matrixStack.pop(); // C

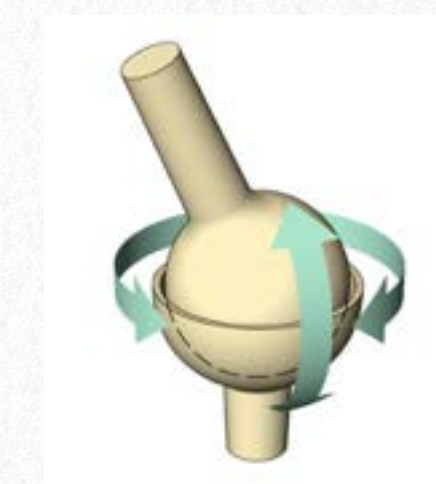
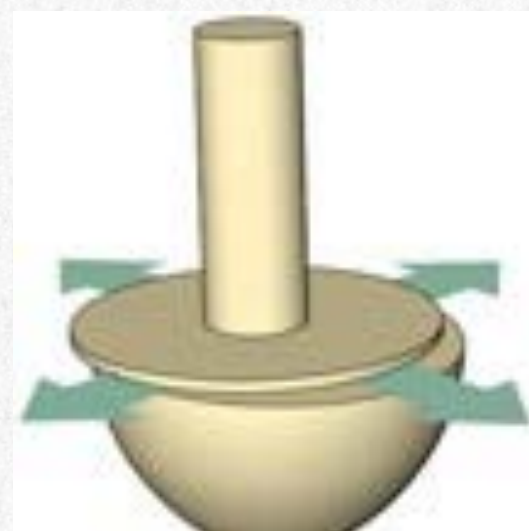
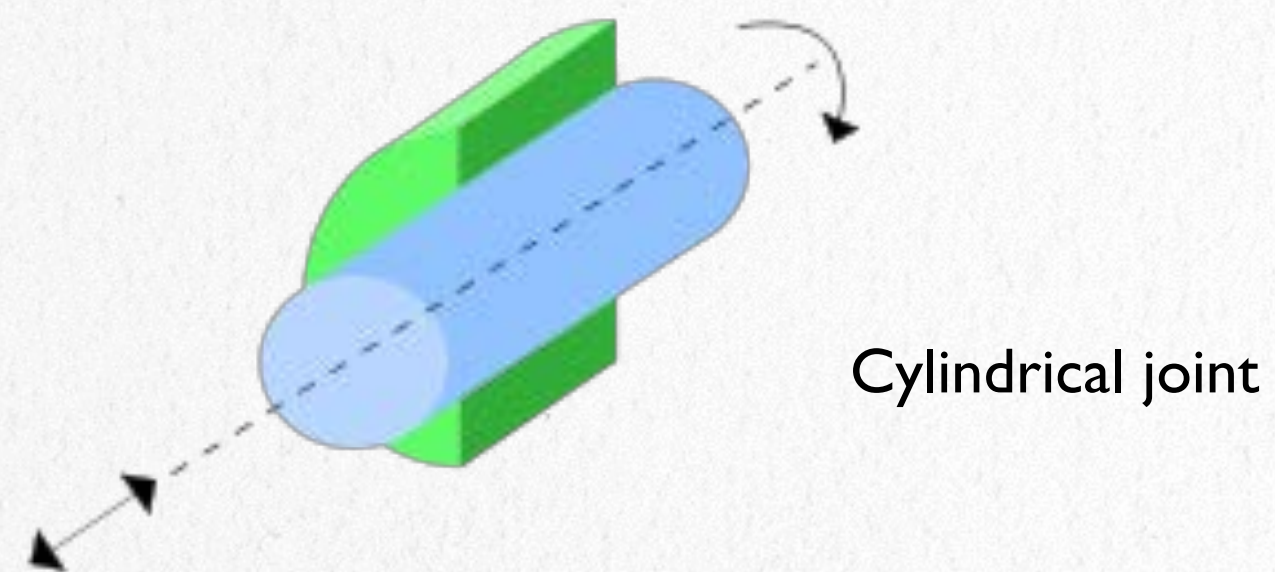
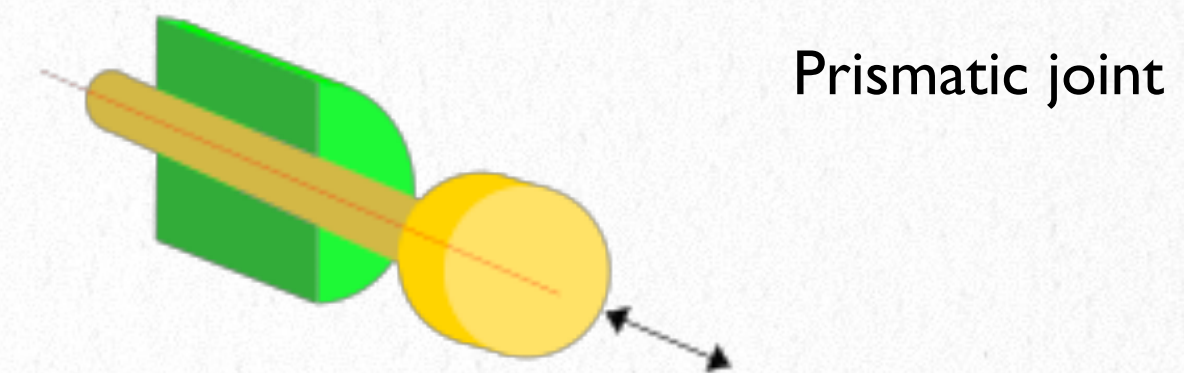
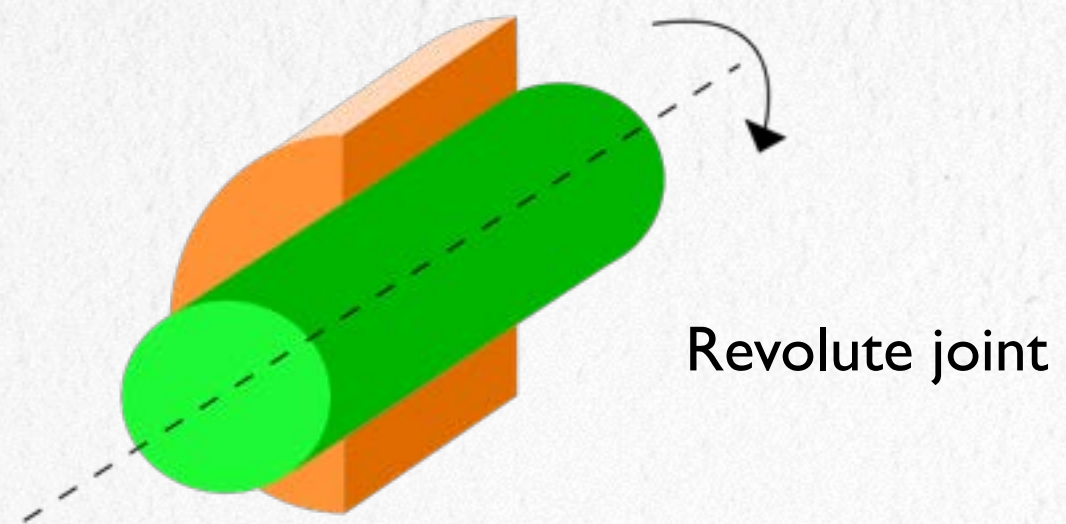
matrixStack.pop(); // B
matrixStack.pop(); // A
// current top matrix is inv(E)*0
// we can now draw another arm
matrixStack.push(F);
...
```



Modelos articulados

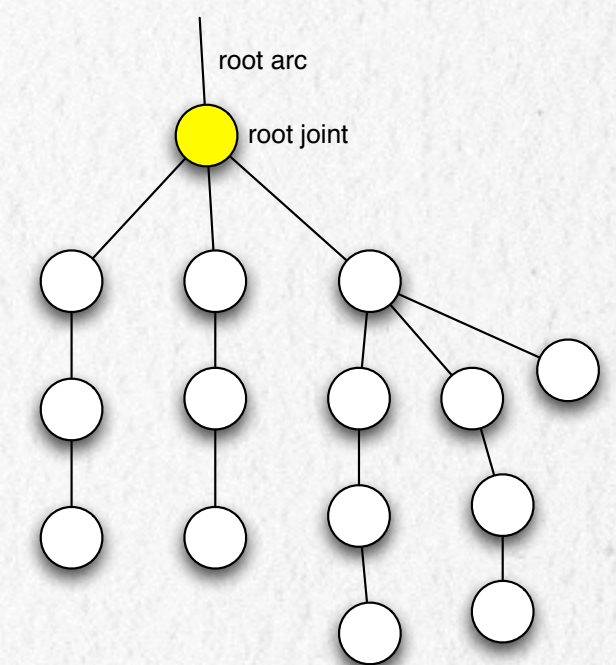
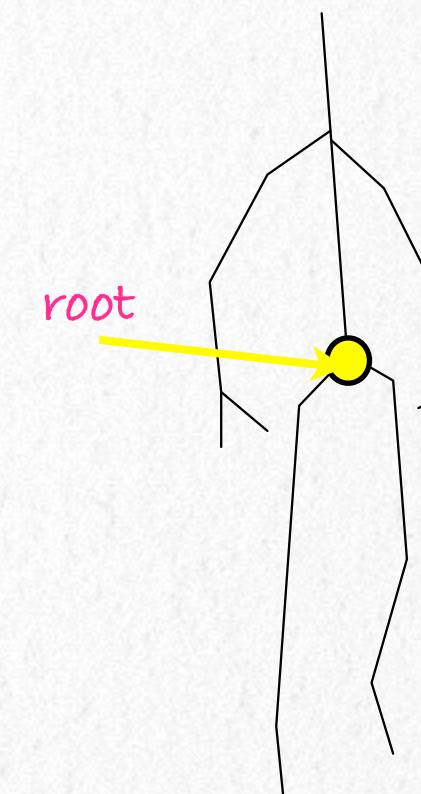
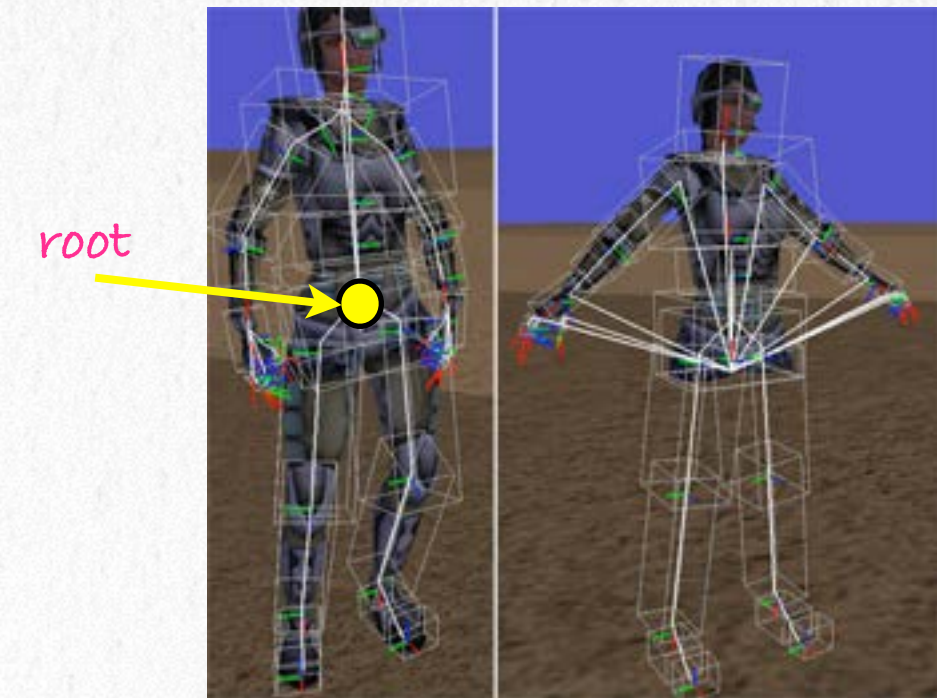
- Objetos que se conectan en sus extremos para formar **cadena articulada**.
- Los objetos rígidos que forman las conexiones entre las juntas (**joints**) se conocen como **links**.
- El objeto libre en el extremo de la cadena de joints y links se conoce como **órgano terminal** o **end-effector**.
- En robótica se manejan diferentes joints, en CG los más comunes son los joints de **rotación** o los joints **ball-and-socket**.
- La dirección permitida de movimiento de un joint se conoce como **grado de libertad (DOF)** del joint.
- Cuando un joint tiene más de un grado de libertad se puede modelar como un conjunto de n joints, cada uno con un grado de libertad conectado por $n-1$ links de tamaño cero.

Grados de libertad en diferentes tipos de joint



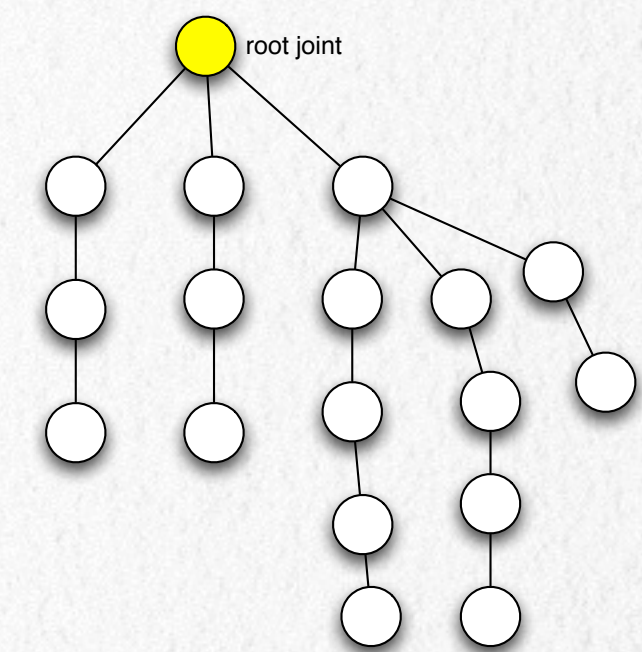
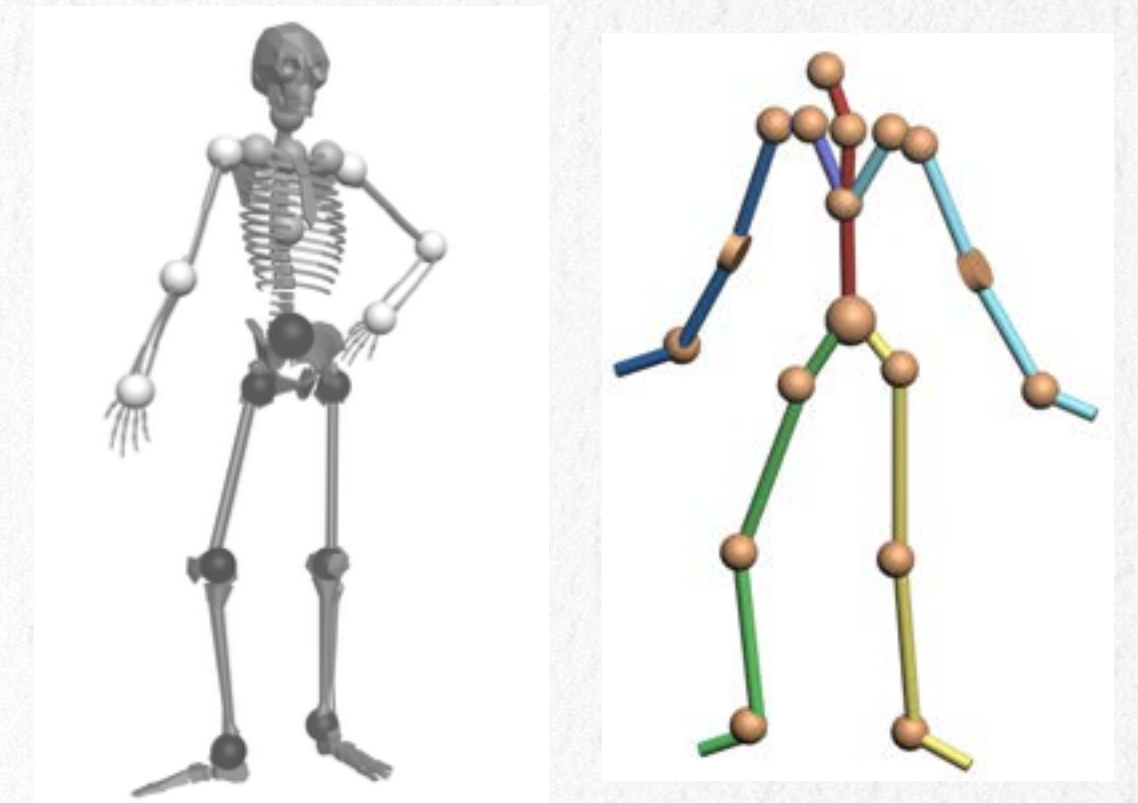
Modelos articulados

- Los links y joints se pueden representar en una estructura de árbol de nodos y arcos.
- El nodo superior en el árbol es el nodo raíz (root) que corresponde al objeto descrito respecto al marco de referencia del wcs o el objeto más alto en la jerarquía.
- La posición de los demás nodos se especifican respecto a su padre comenzando por la raíz.



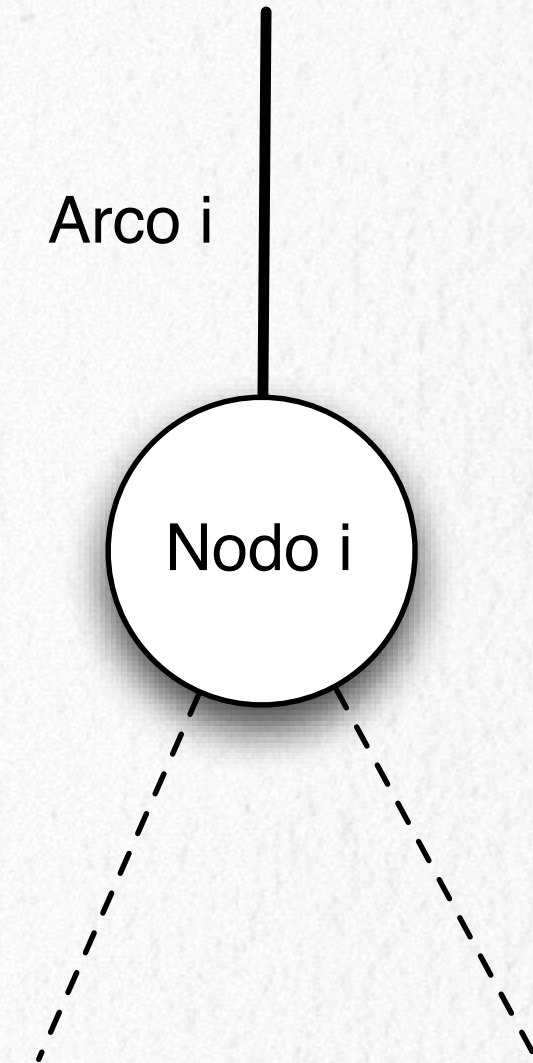
Modelos articulados

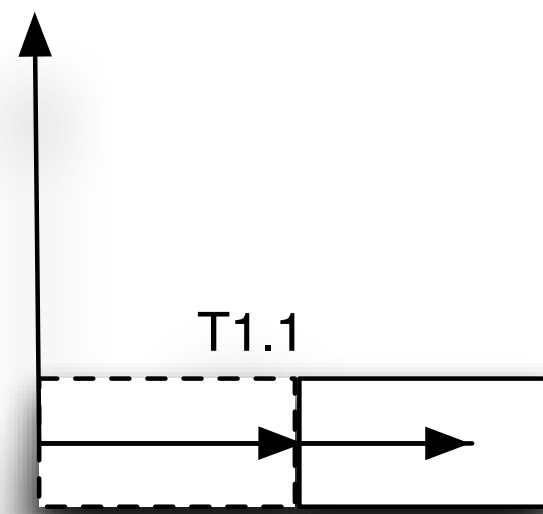
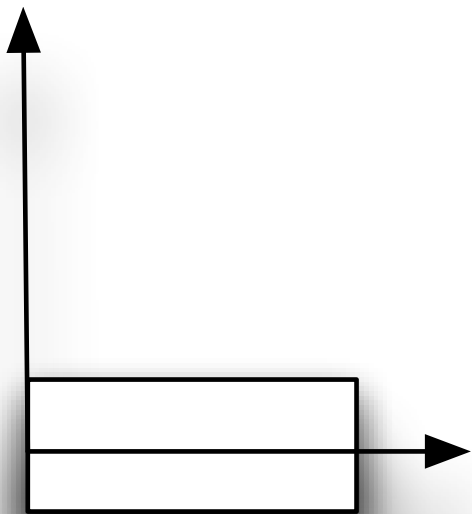
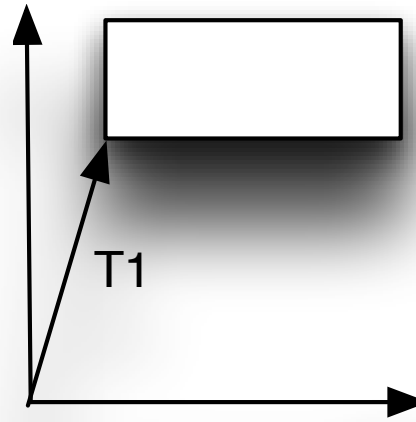
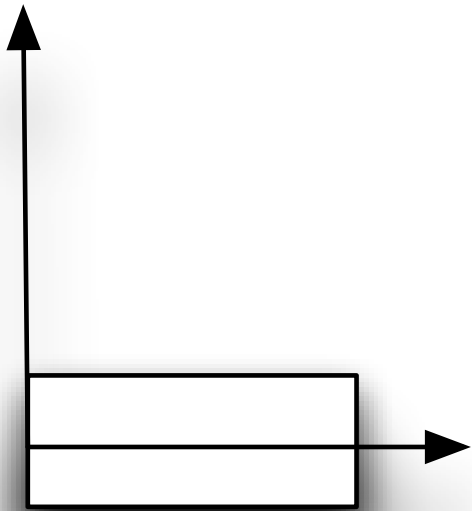
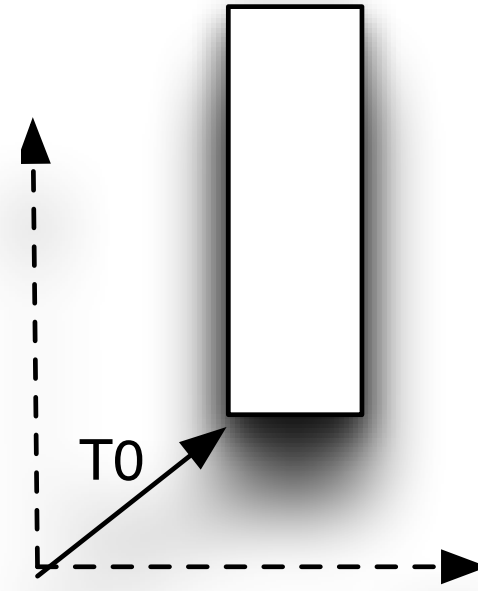
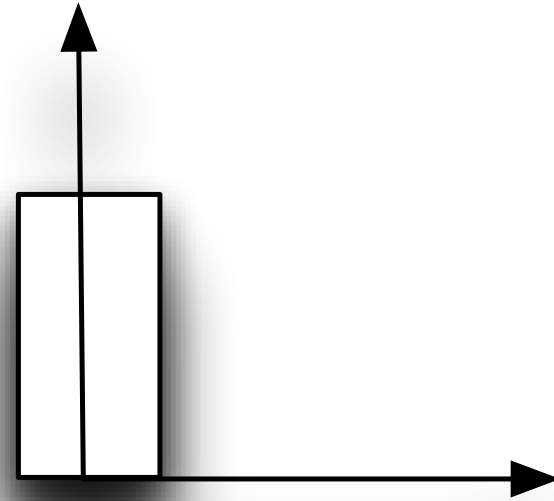
- Un nodo a partir del cual no hay arcos hacia abajo es el nodo hoja.
- “Hacia arriba en la jerarquía” significa más cerca de la raíz.
- Dos nodos conectados por un arco tienen una relación de padre-hijo.
- La transformación jerarquía-árbol relaciona un nodo en el árbol a la información sobre el objeto (link) y el arco del árbol (el joint) a la transformación a realizar a todos sus nodos hijo.



Modelos jerárquicos

- El Nodo i contiene:
 - la transformación que debe aplicarse a la posición del objeto de tal manera que su punto de rotación sea el origen.
 - atributos del objeto.
- El Arco i contiene:
 - la transformación de rotación y translación del Link i a su posición neutral relativa al Link $n-1$.
 - las variables responsables de articular el joint.

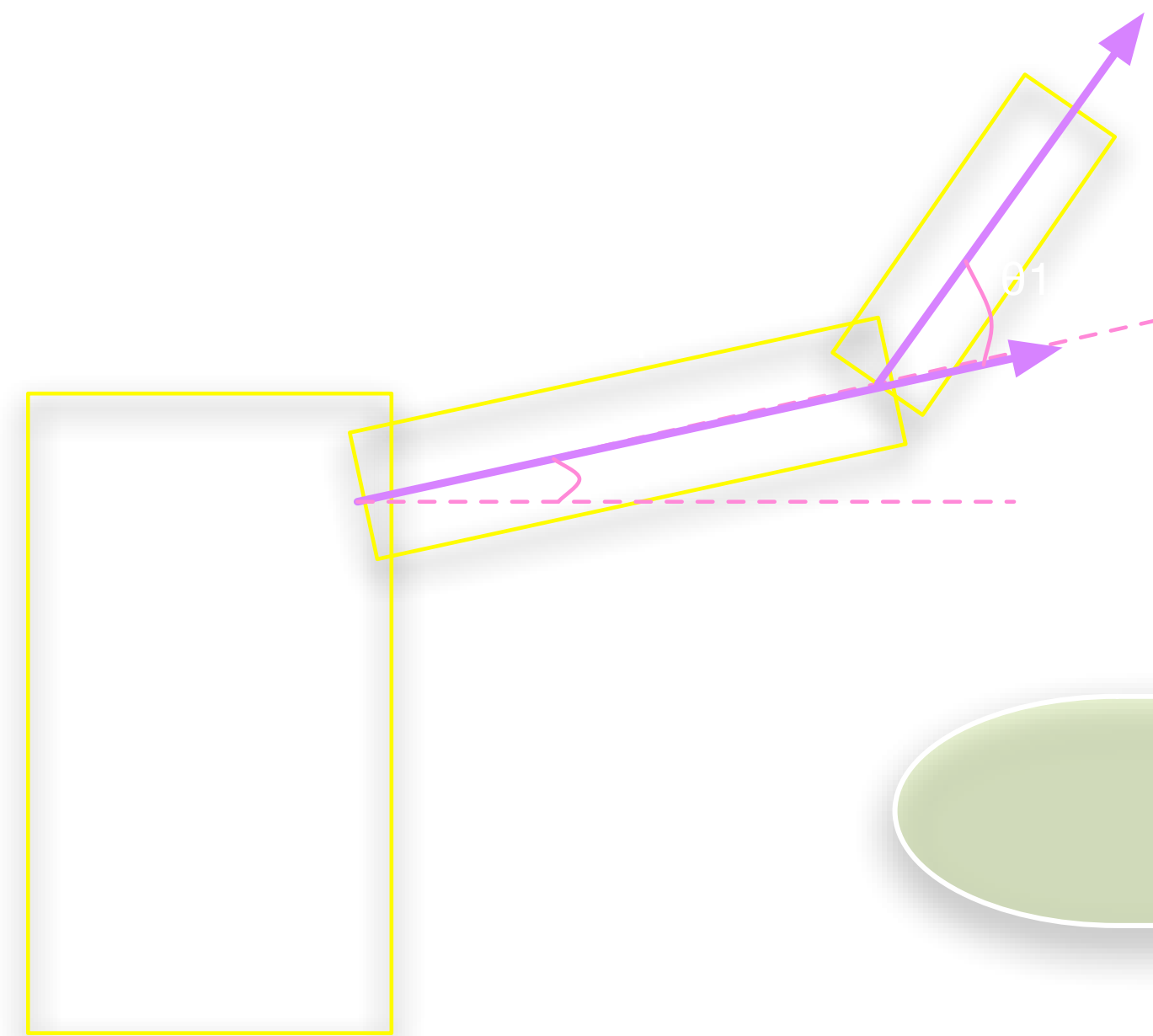




datos para el Link 0 (raíz)

datos para el Link 1

datos para el Link 1.1



datos para el Link 0 (raíz)

datos para el Link 1

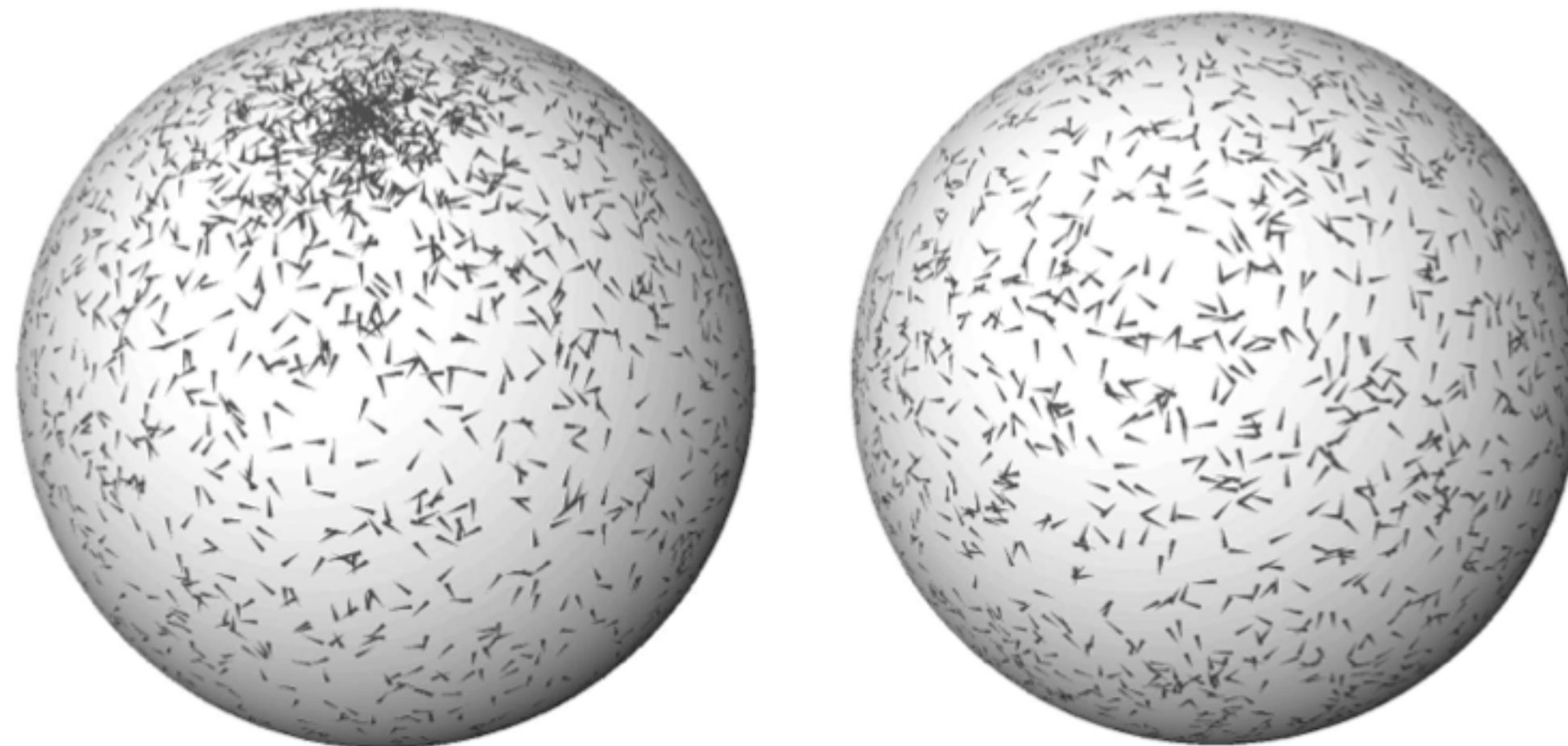
datos para el Link 1.1

Cinemática directa

- Para evaluar una jerarquía de links y joints hacemos un recorrido depth-first en el árbol correspondiente.
- Cuando un arco se visita hacia abajo en la jerarquía, sus transformaciones se concatenan a la transformación de su nodo padre.
- Cuando un arco se visita hacia arriba en la jerarquía, la transformación de ese nodo debe restablecerse antes de continuar hacia abajo por otra rama.
- La estructura de datos para cinemática: pila de matrices de transformación.
- Un conjunto de parámetros de rotación y translación (DOF) que resulta de especificar completamente la posición y orientación de cada link de conoce como configuración o pose.
- Una pose se especifica generalmente con un vector con el mismo número n que los n grados de libertad.

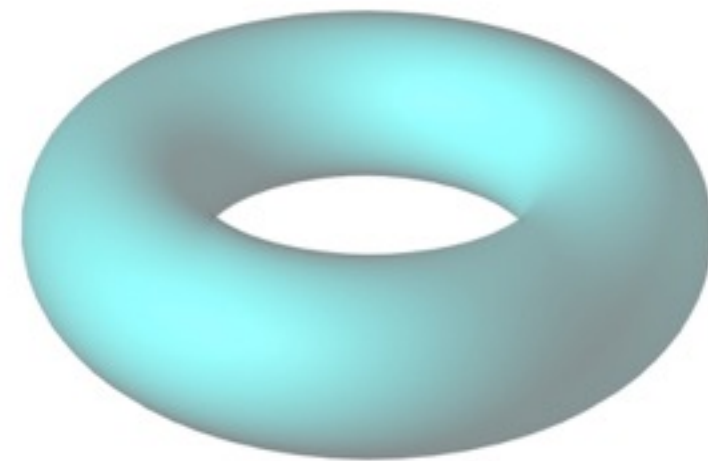
CS de cuerpos rígidos

- Cada configuración de un cuerpo rígido libre de rotar relativamente a un marco de referencia inercial (fijo), puede identificarse con un $R \in SO(3)$ único.
- Bajo esta identificación, se refiere al grupo de rotaciones $SO(3)$ como al **espacio de configuraciones CS** del sistema.



CS de cuerpos rígidos

- La trayectoria del sistema es una curva $R(t) \in SO(3)$ para $t \in [0, T]$.
- Más generalmente llamamos al conjunto **CS** **espacio de configuración** del sistema si cada elemento $R(t) \in SO(3)$ corresponde a una configuración válida del sistema y se puede identificar con un elemento único de **CS**.



$$\dot{x} = \frac{1}{12} \dot{y}$$

