

Animation 3
WS 2020 / 21

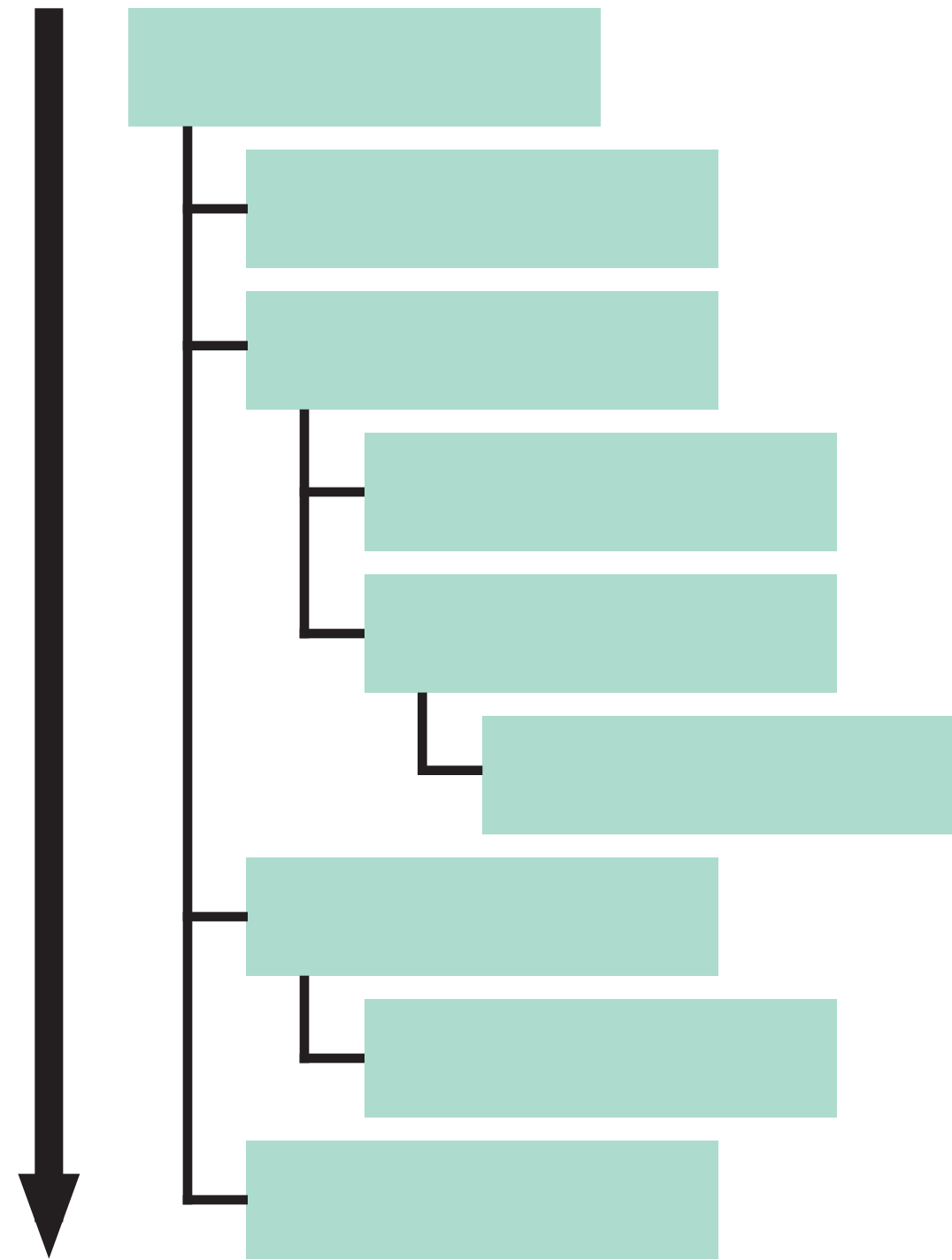
Lehreinheit 3

11.11.2020

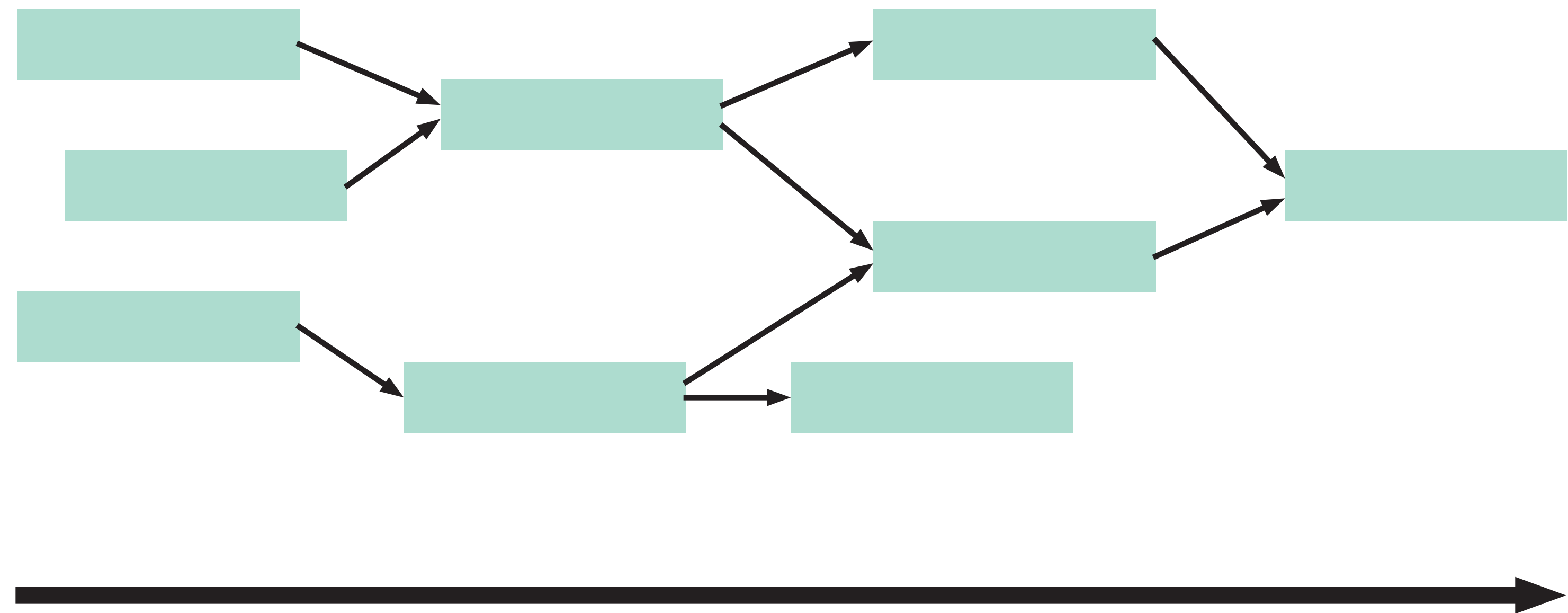
Übersicht:

- Wiederholung
- Joints ausrichten
- Joint-Chains
- 3-Joint IK (Arm/Bein)

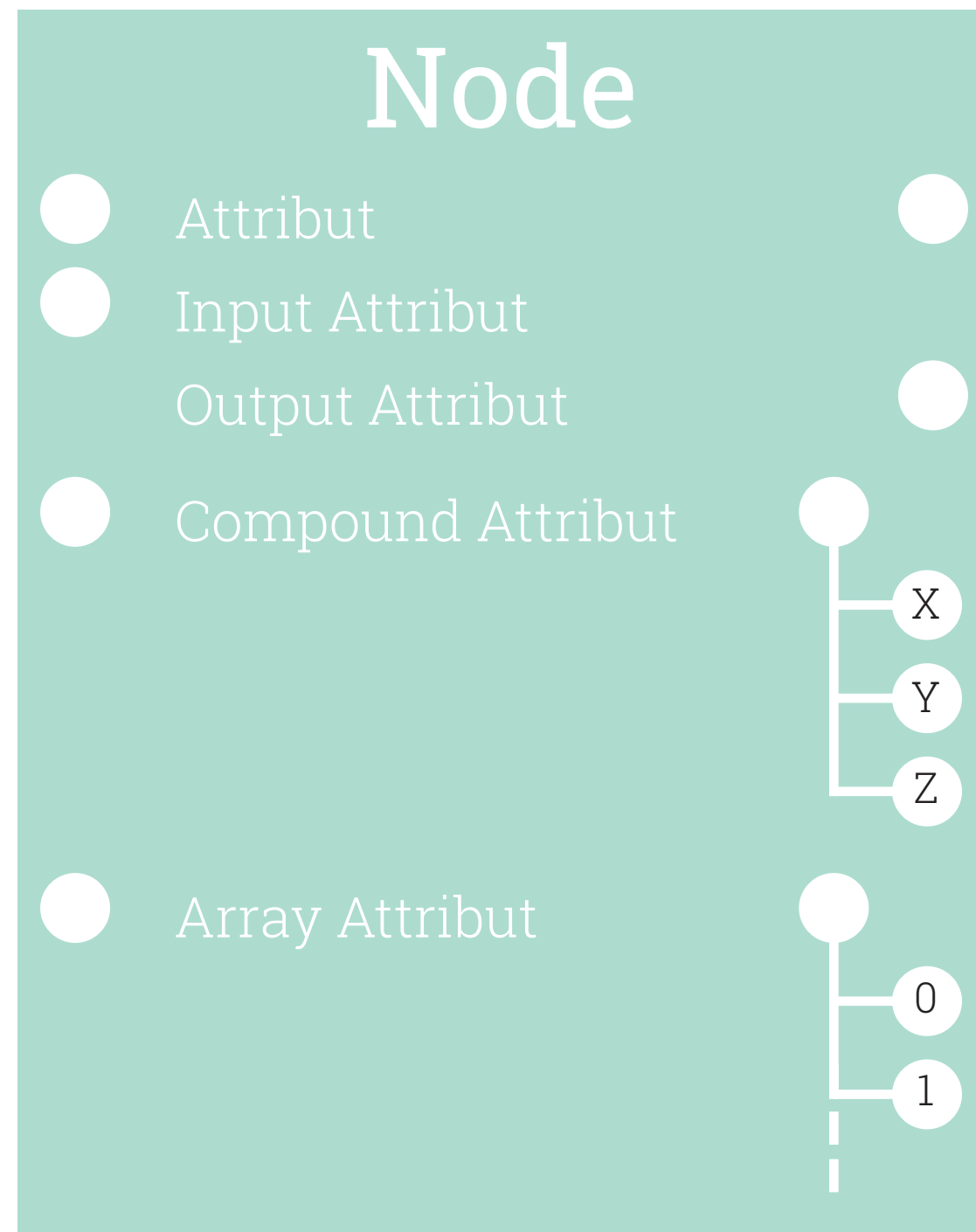
Hierarchy



History



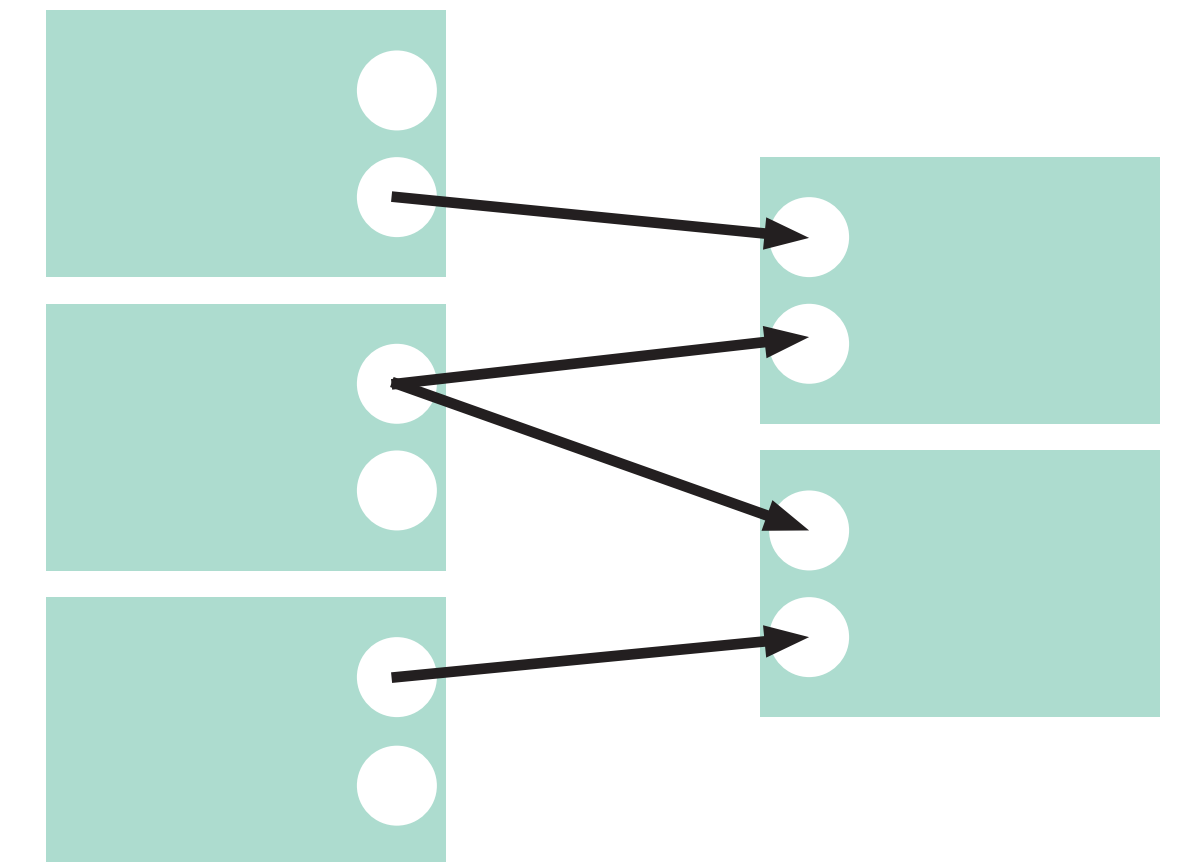
Attribute



- Werte auf einem Node
- verschiedene Arten
 - Boolean (Off, On)
 - Integer (-2, 0, 1, 1024)
 - Enum (Rot, Hase, Rechts, ...)
 - Float (-18.3, 0.56, 1230.0)
 - Vector (3 Float: X, Y, Z)
 - Matrix (4x4 Float)
 - NurbsCurve
 - NurbsSurface
 - Mesh
- Einzel / Compound / Array
- Input und/oder Output
- keyable / lockable / hidden / range

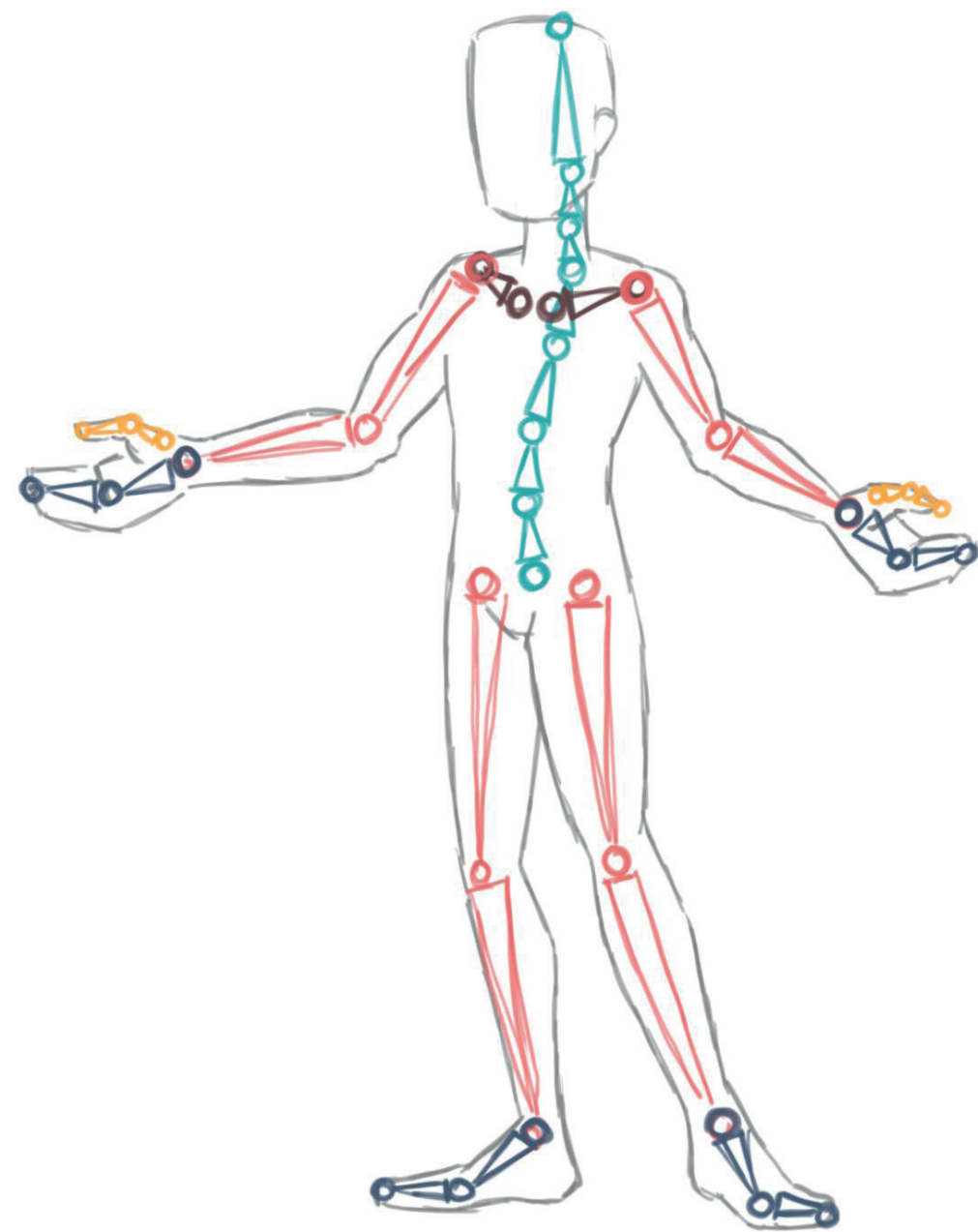
Connections

- Verbindung zwischen Attributen
- Übertragung des Wertes von Quelle zu Ziel
- automatische Konvertierung
- mehrere Ziele pro Quelle möglich
- Eine Quelle pro Ziel möglich

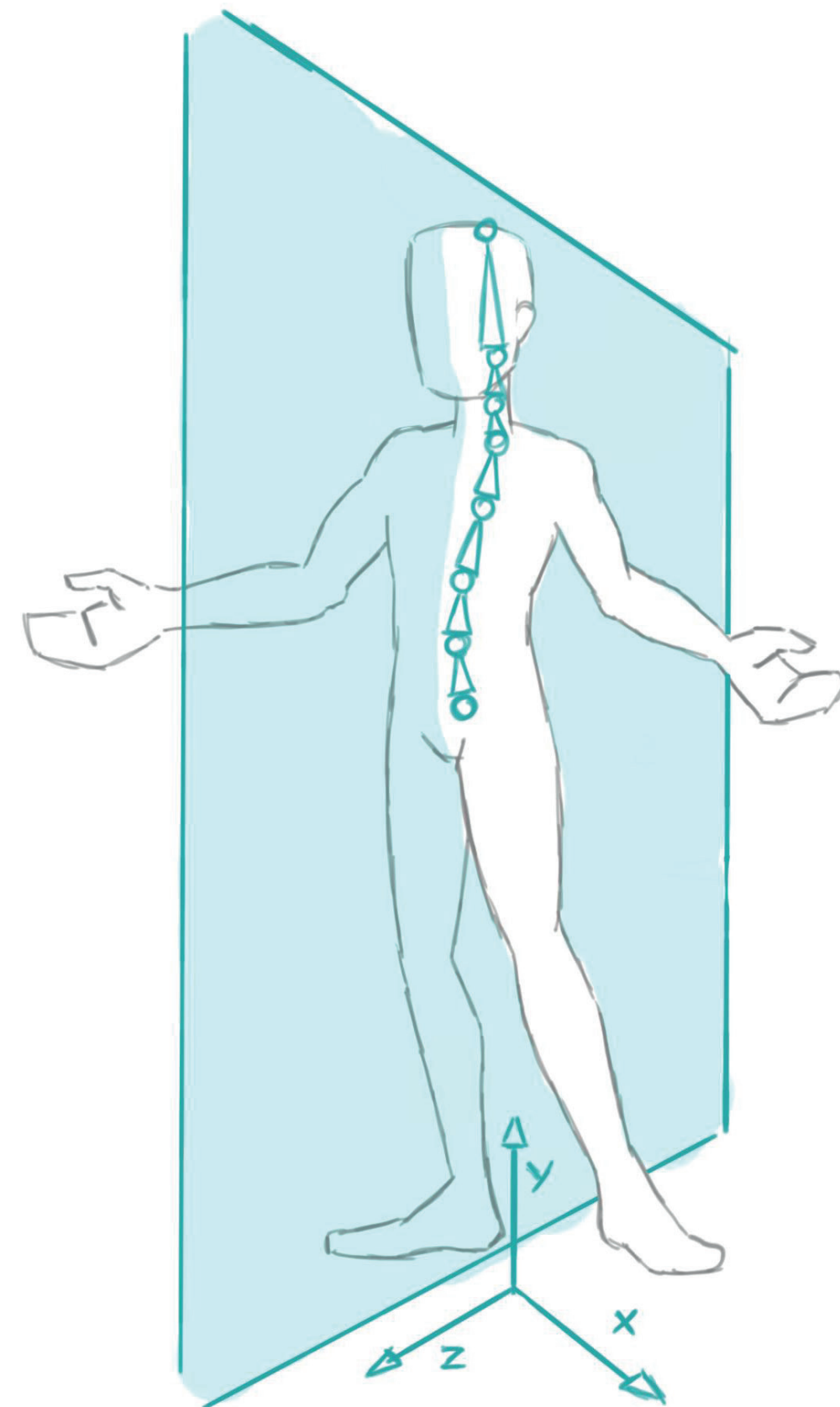


Joints ausrichten

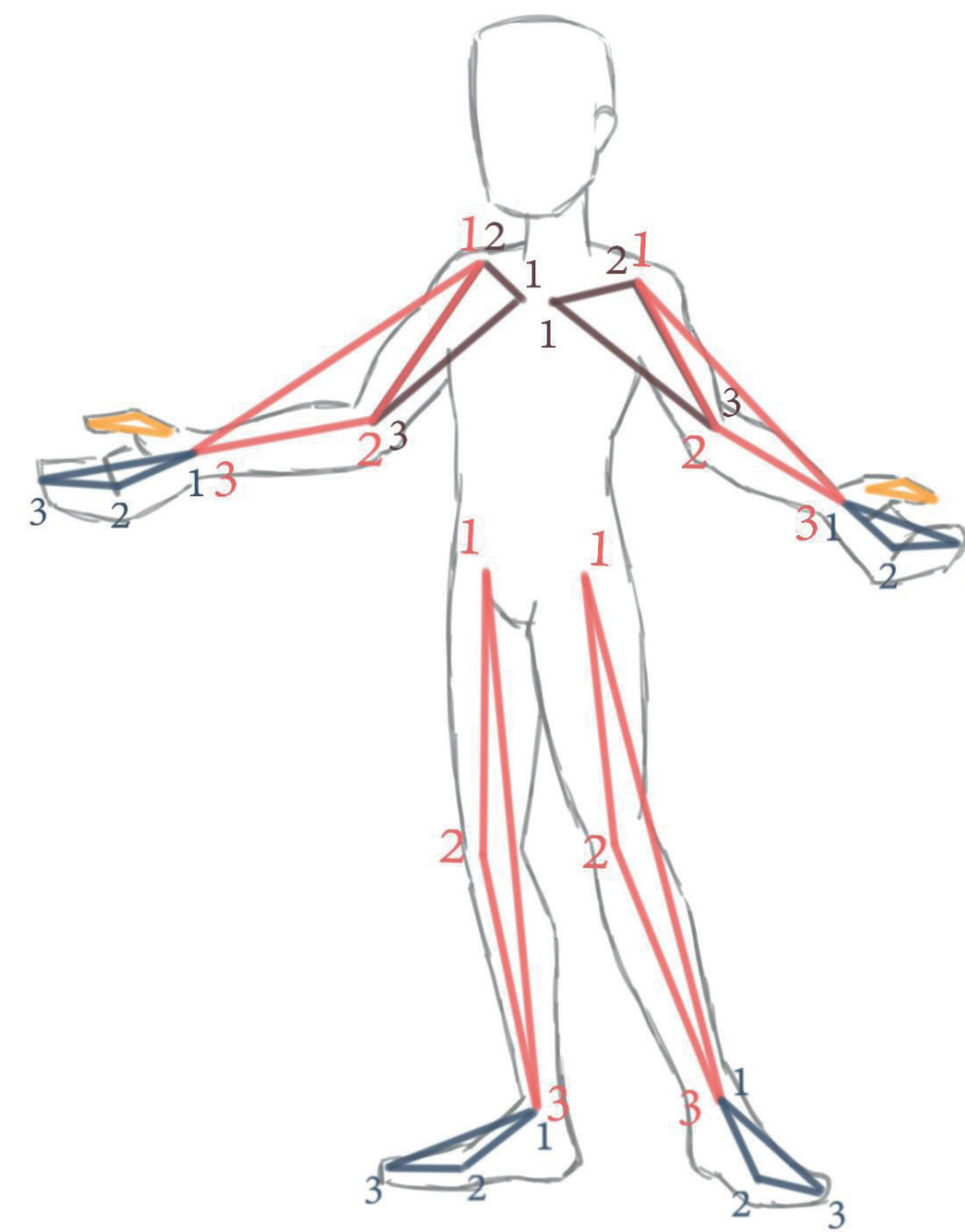
Joint Unterteilungen



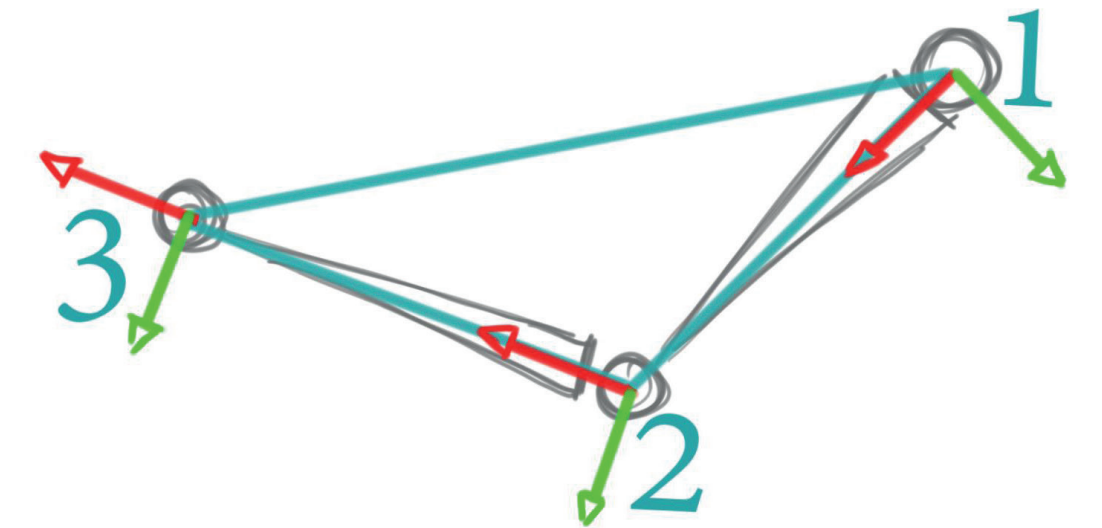
Ebenen-Ausrichtung



Dreiecks-Ausrichtung



Ausrichtung im Dreieck

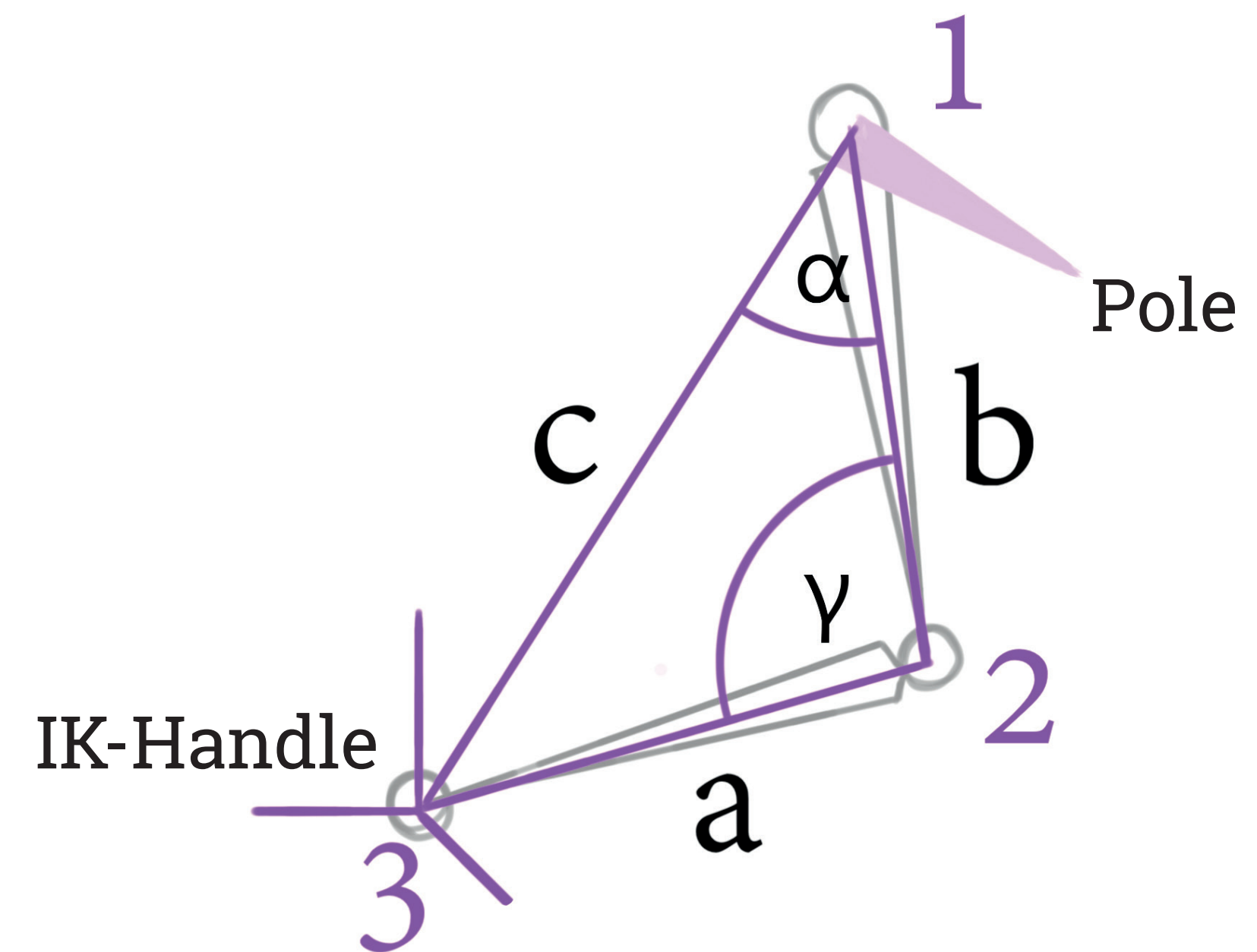


Joint-Chains

- Simple Kette von Joints
- Einfaches aufsetzen für FK
- (IK mittels Spline IK)
- Hilfsmittel:
 - NurbsCurves,
 - Up-Locator
 - JointOrient-Script

3-Joint IK

- Simple IK-Setup zur Steuerung von Joint-Ketten mit einer Länge von 3 (oder mehr) Joints
- Einsatz z.B. in Armen oder Beinen (Hände und Füße nicht inbegriffen!)
- Steuerung per IK-Handle, der die Ziel-Position angibt. Die Rotation der Joints, um mit dem letzten Joint diese Position zu erreichen, wird vom IK-Solver errechnet.
- Berechnung basiert auf fixen Abständen zwischen den Joints und dem variablen Abstand zwischen IK-Handle und Anfangs-Joint.
- Ausrichtung des Setups basiert auf der Ebene, die die drei Joints als Dreieck aufziehen.
- Ausrichtung lässt sich per Pole-Vector verändern.



$$\cos \gamma = c^2 - a^2 - b^2 + 2ab$$

$$\sin \alpha = (a * \sin \gamma) / c$$

Fragen?