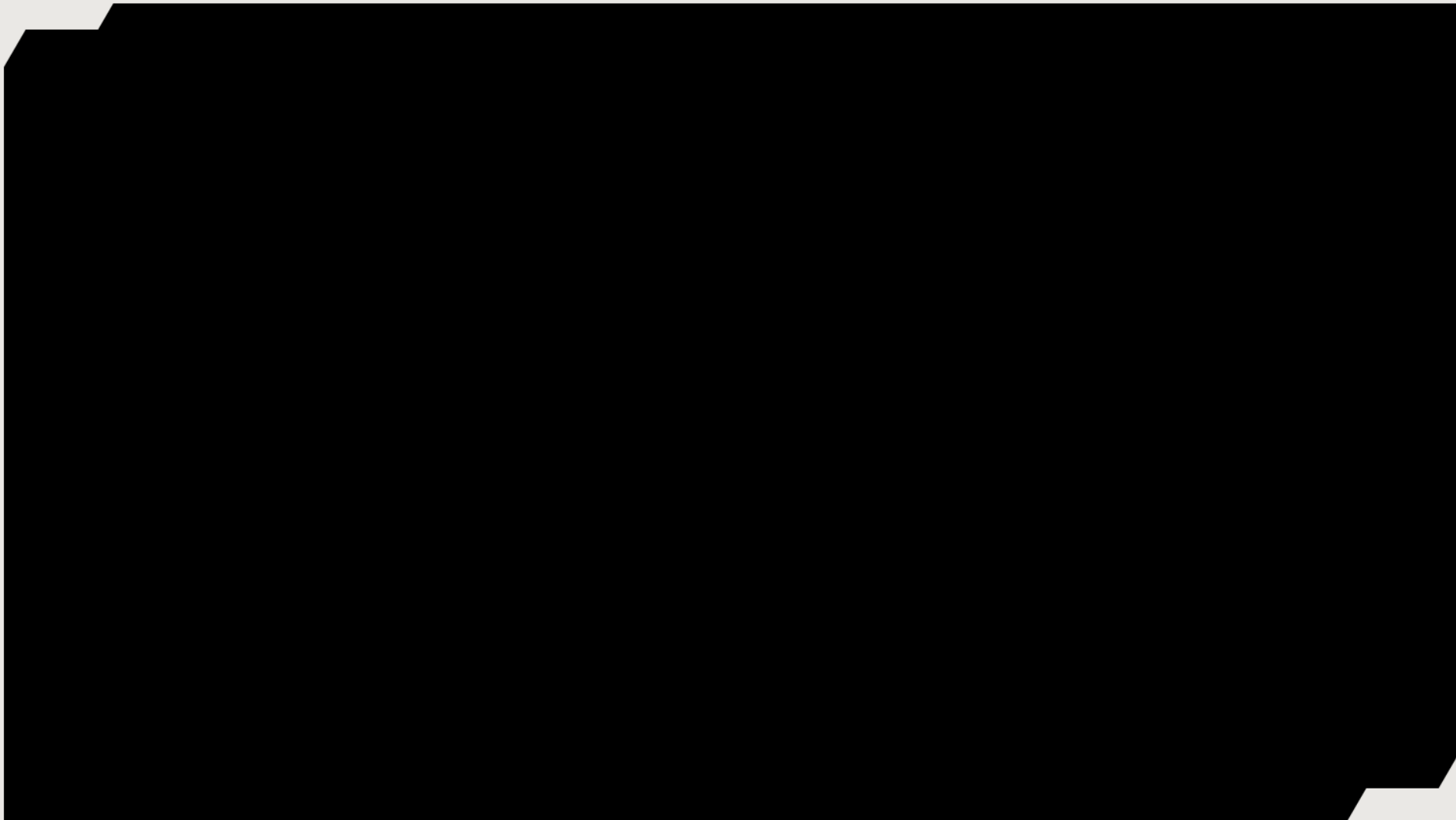# Player Traversal Mechanics In The Vast World Of Horizon: Zero Dawn
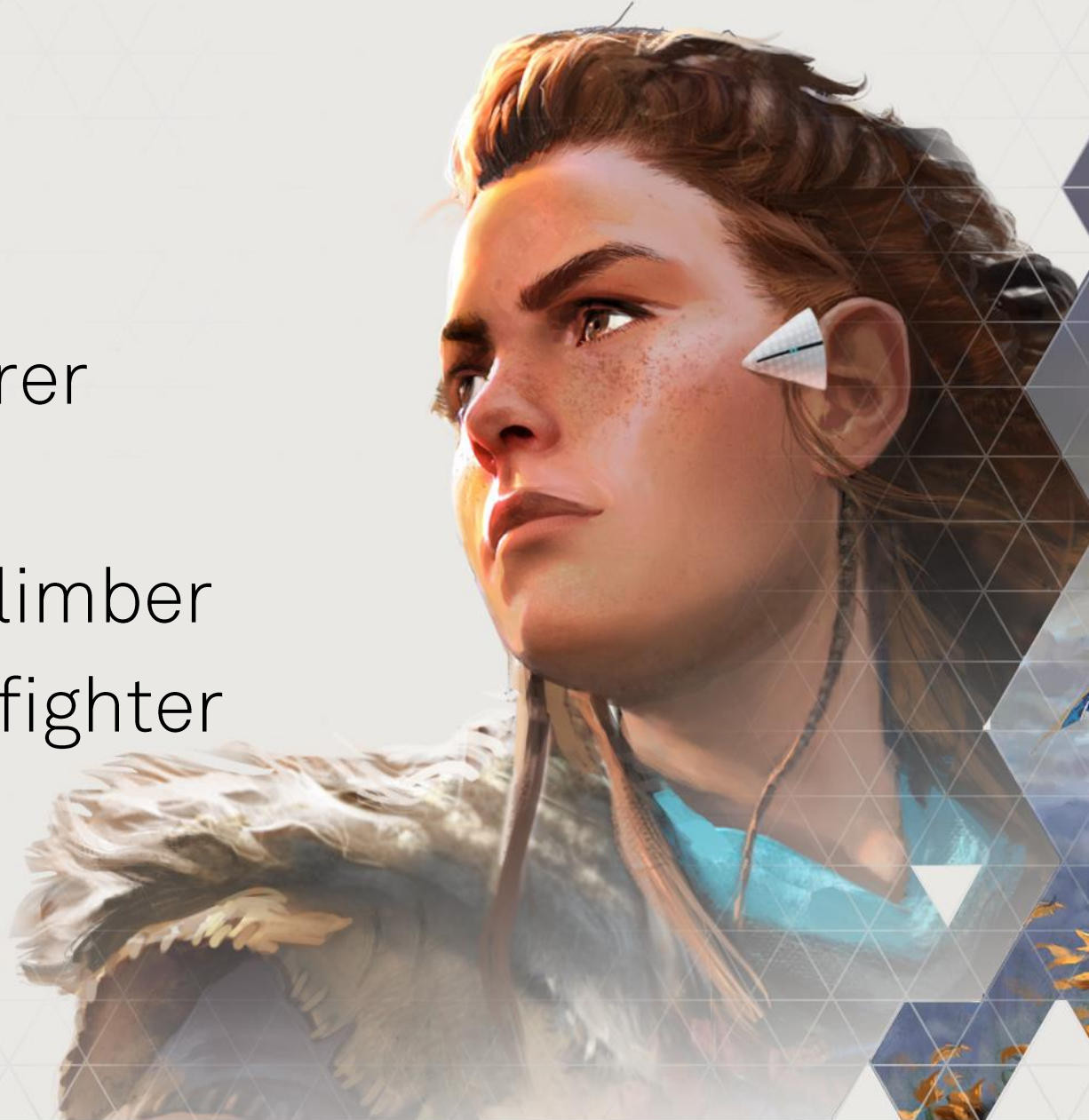
Paul van Grinsven
Game programmer
Guerrilla

Aloy

- Female
- Adventurer
- Agile
- Strong climber
- Tactical fighter
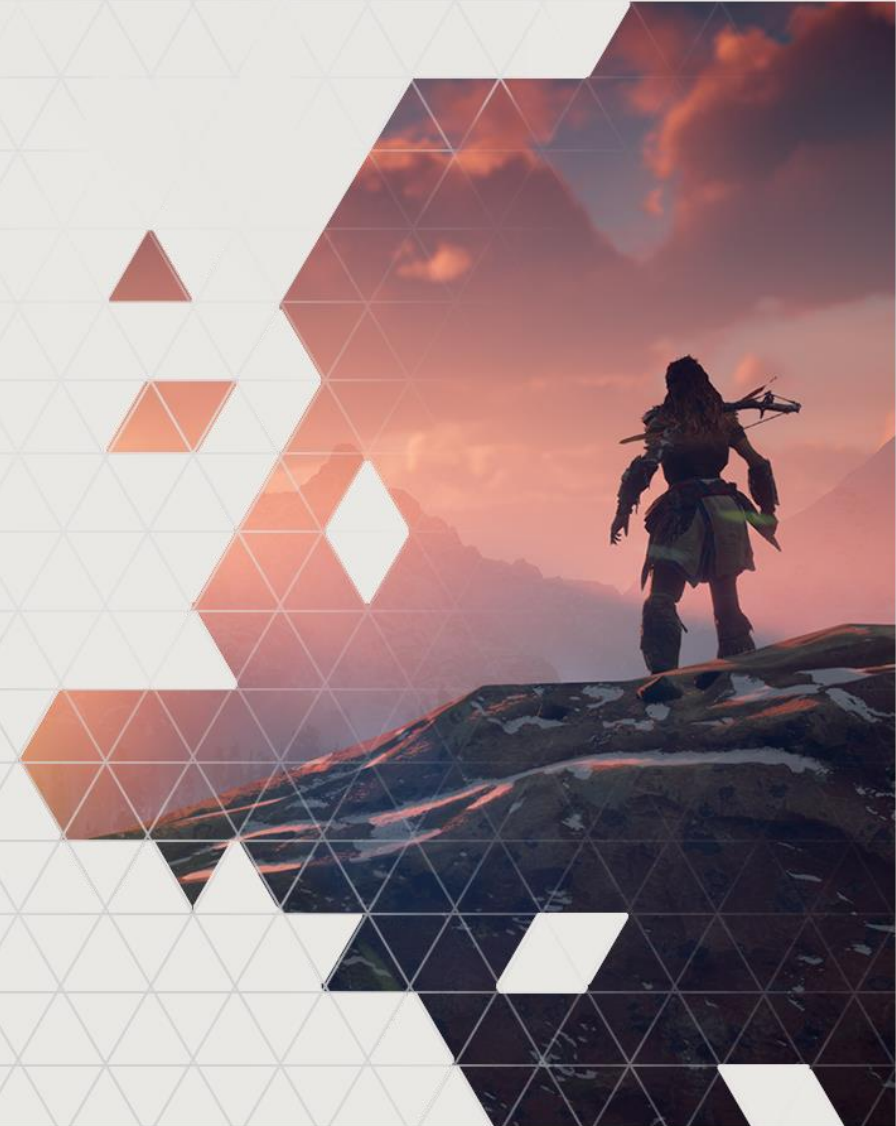- Hunter

The World Of Horizon

- Terrain mostly procedurally generated
- Various ecotopes
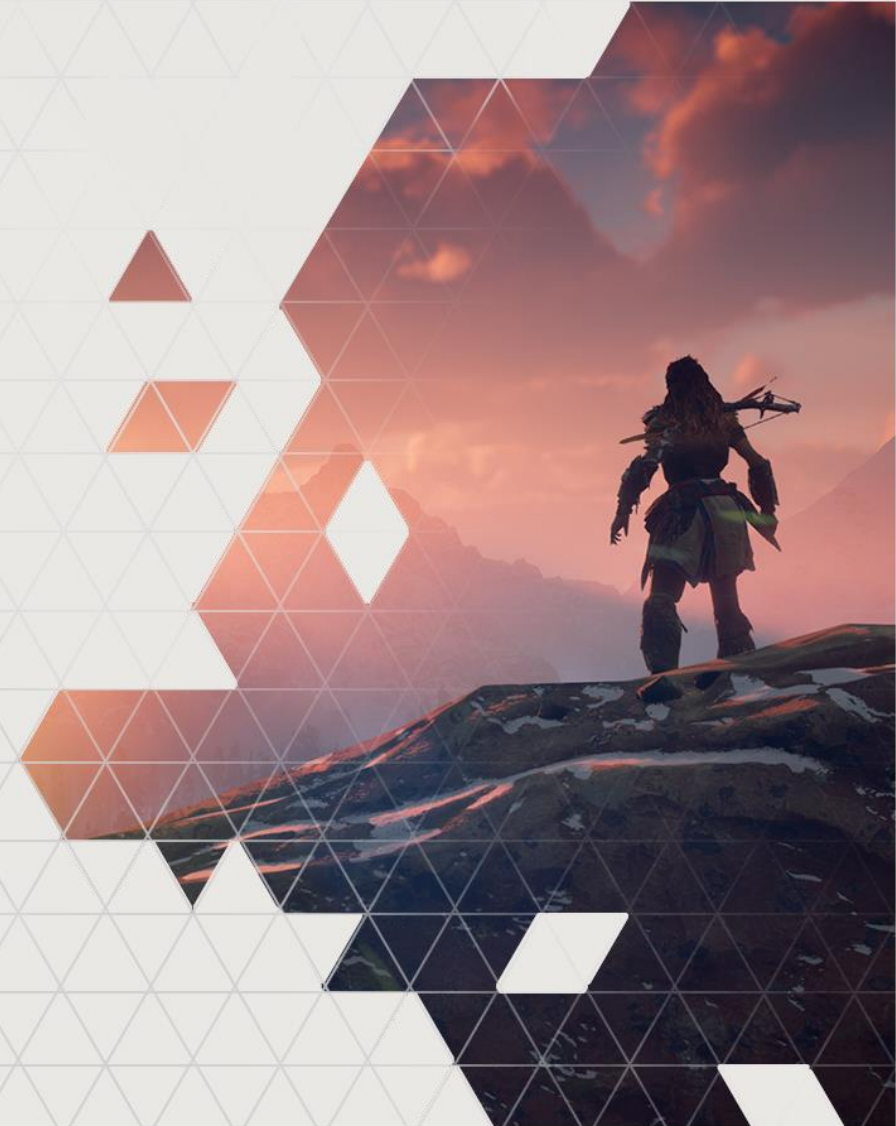- Manmade structures, indoor/outdoor

Overview

- Goals and constraints
- Tools, workflow and pipeline
- Responsive navigation
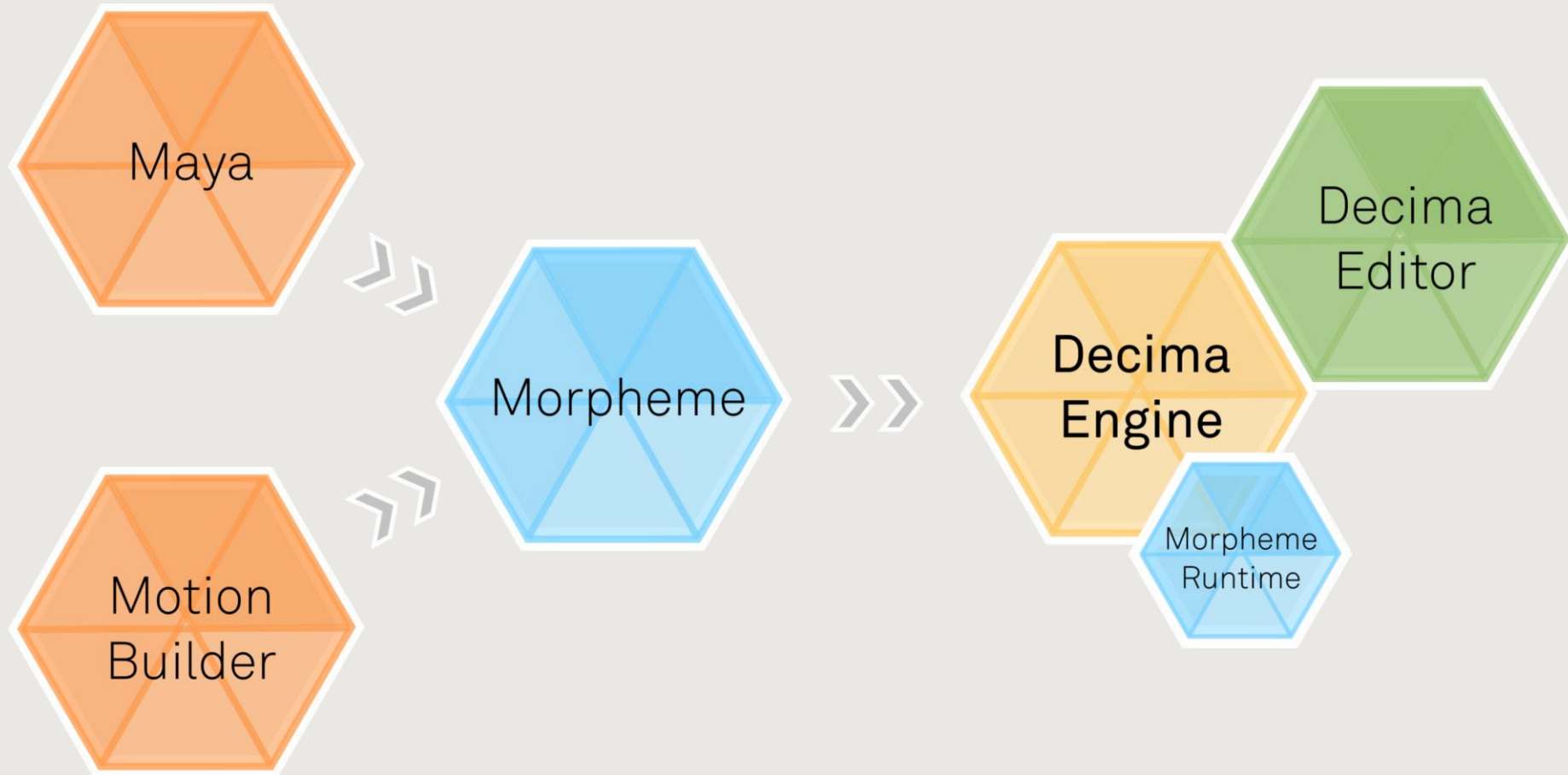- Advanced traversal mechanics
- Future plans

Goals And Constraints

- *Design:*
  - Fluid and responsive
- *Team:*
  - 2 game programmers
  - 3 animators
  - 1 designer
  - 1 producer
- *Technical:*
  - 30Hz frequency updates

Tools, Workflow And Pipeline

Maya

Motion Builder

Morpheme

Decima Engine

Decima Editor

Morpheme Runtime

Vast World Of Horizon: Zero Dawn

Highly Responsive Navigation

- Fluid and responsive
- Left or right foot forward
- Easy to control

Highly Responsive Navigation

- *Problem:*
  - Unresponsive stop to start transition
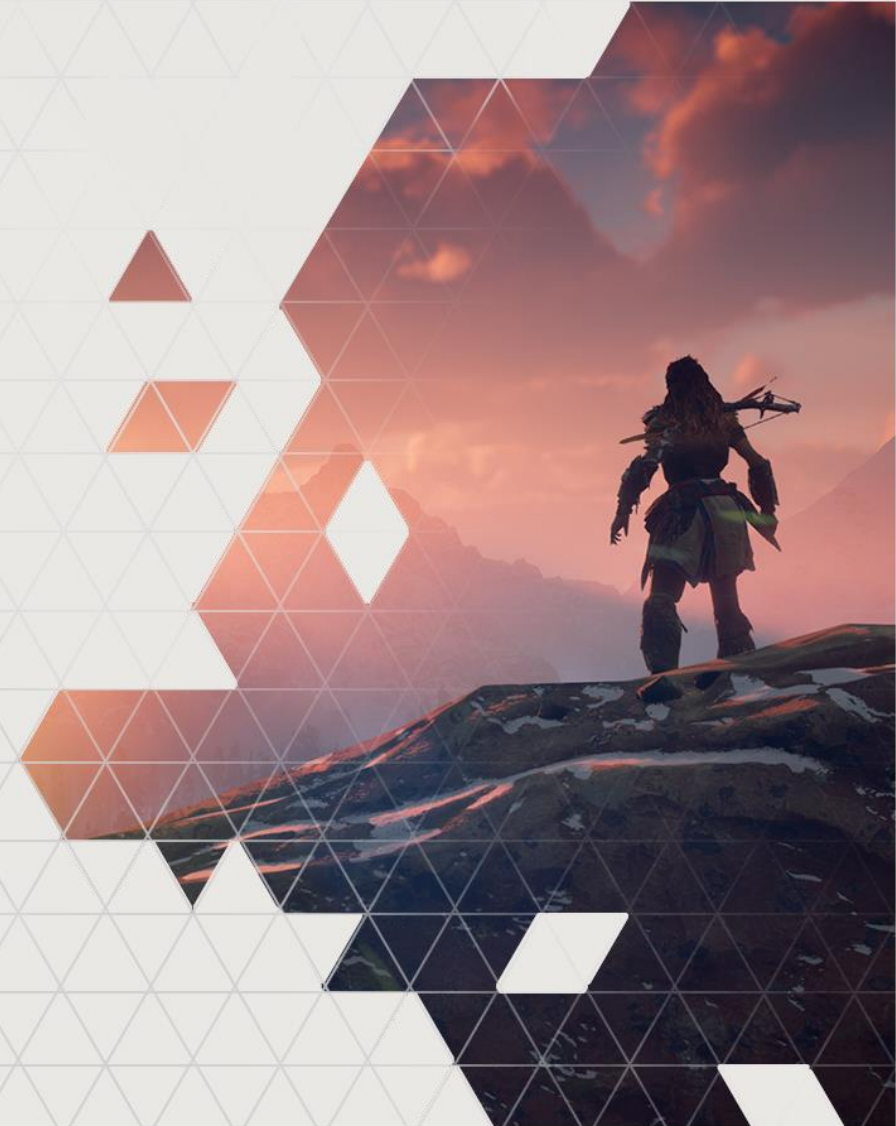 - Waiting for completion of stop animation
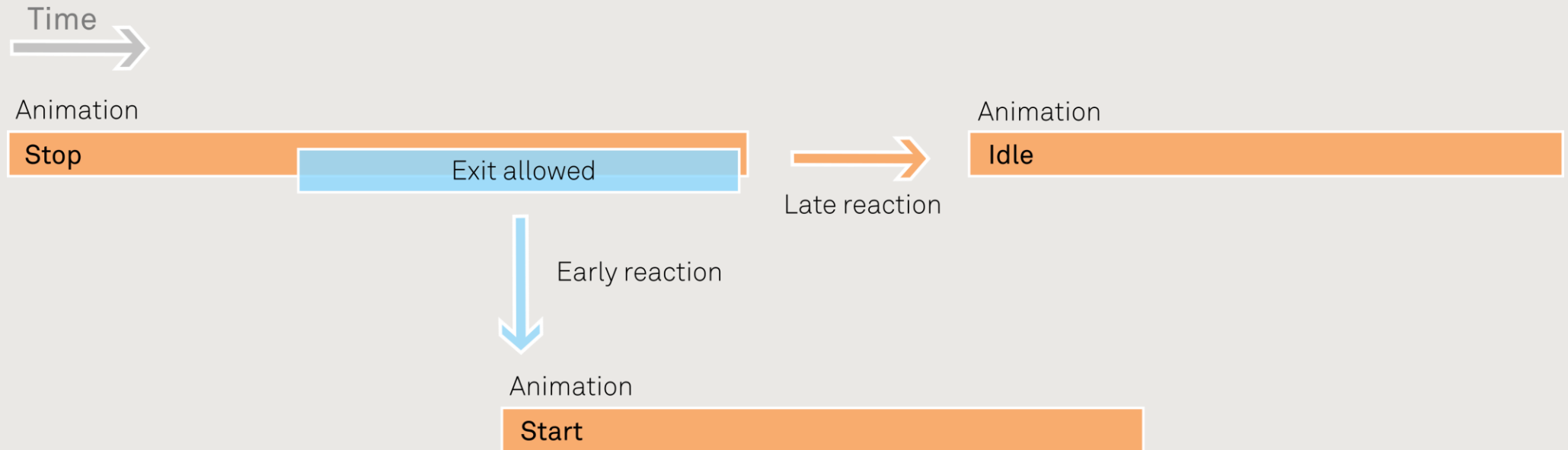
Highly Responsive Navigation

- Animation events
  - Time, duration and ID.
- Provided to the game after each network update

Highly Responsive Navigation

- Allowing early exit increases responsiveness

Time

Animation
Stop

Exit allowed

Late reaction

Animation
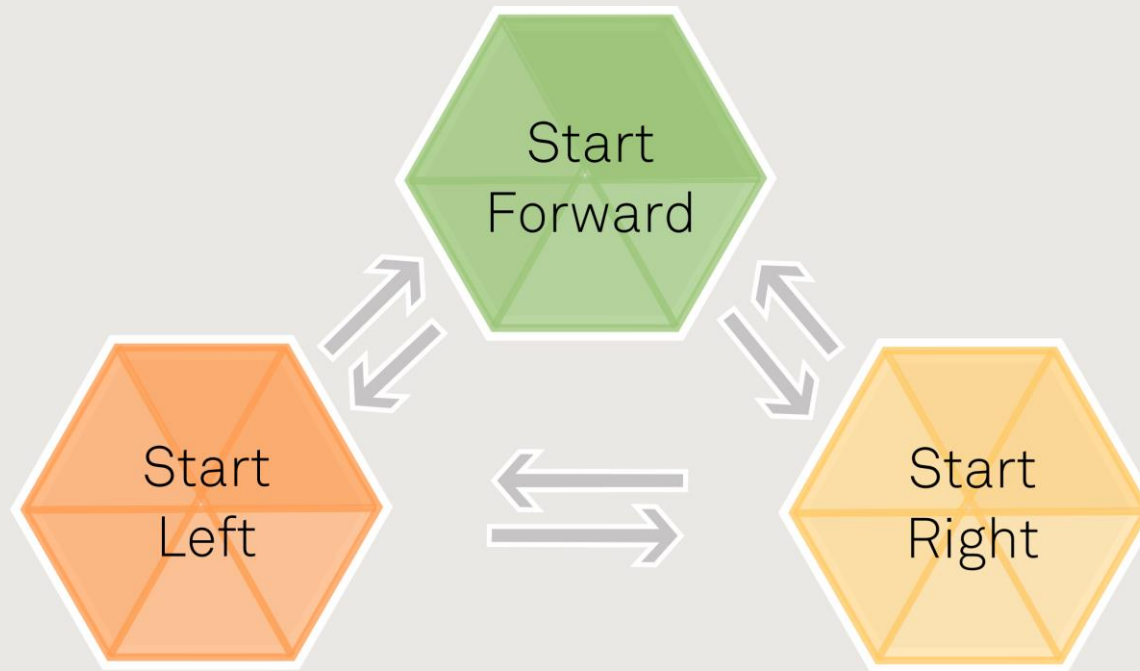Idle

Early reaction

Animation
Start

Highly Responsive Navigation

- Setup of the start system

Highly Responsive Navigation

- Control variables
  - Move (bool), Speed (float) and Heading (float)
- Movement in code is handled with velocity and turn speed
  - Turn speed is evaluated from a curve. (x = linear speed (m/s), y = turn speed (deg/s)
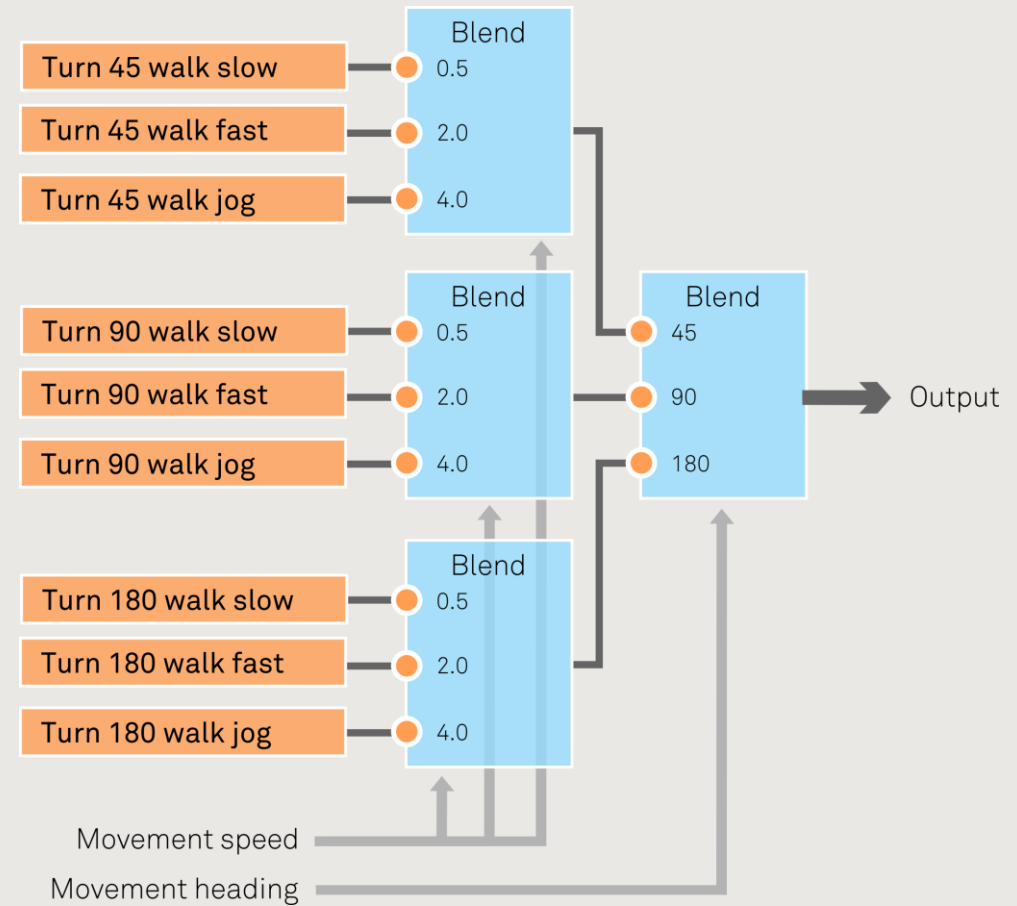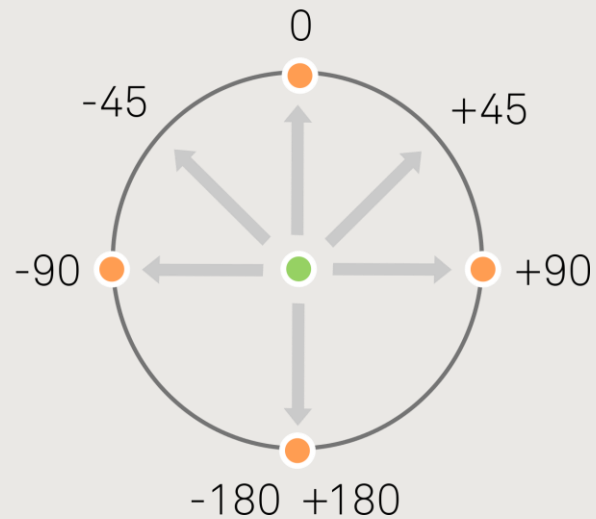 - Low speed = slow turn speed!

Highly Responsive Navigation

- Keep history of controller movement input direction for last 3 frames
    - No input for last 3 frames -> Stop moving
    - Move will always be set for at least 3 frames (0.1 seconds)!
    - Needed for 180 turns during move cycle!
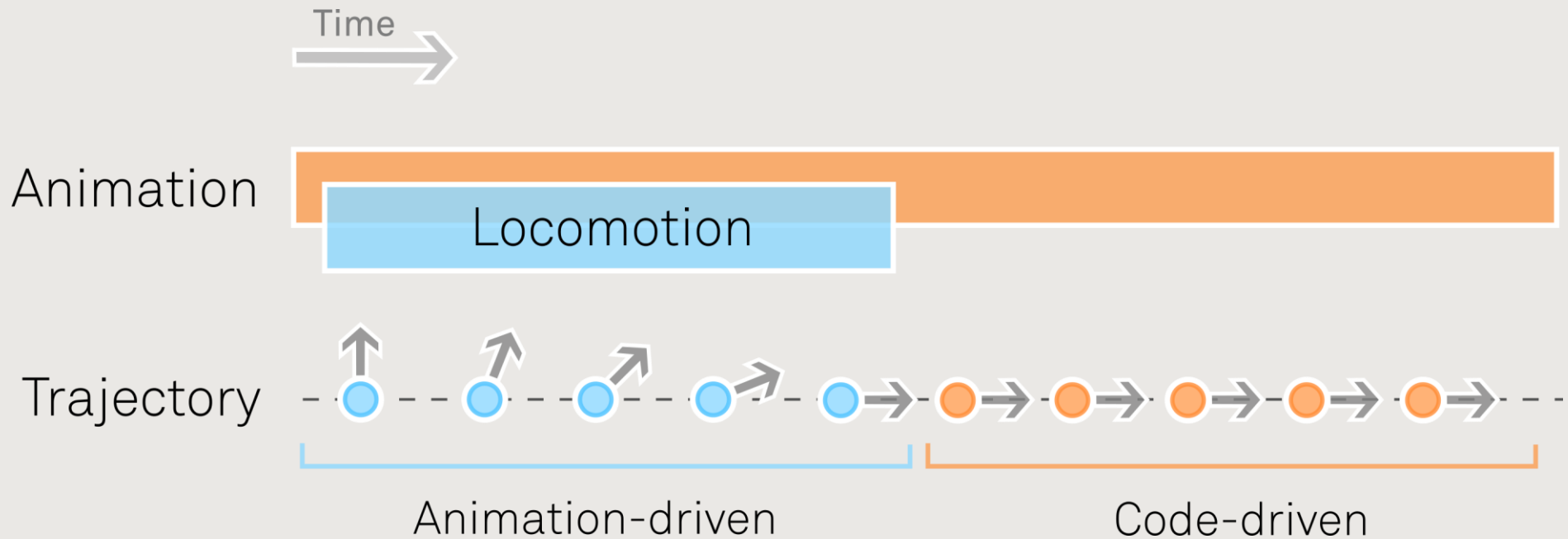
Highly Responsive Navigation

Top view

0

−45   +45

−90   +90

−180 +180

Turn 45 walk slow	Blend
Turn 45 walk fast	0.5
Turn 45 walk jog	2.0
	4.0

Turn 90 walk slow	Blend
Turn 90 walk fast	0.5
Turn 90 walk jog	2.0
	4.0

Turn 180 walk slow	Blend
Turn 180 walk fast	0.5
Turn 180 walk jog	2.0
	4.0

Blend
45
90
180

Output

Movement speed

Movement heading
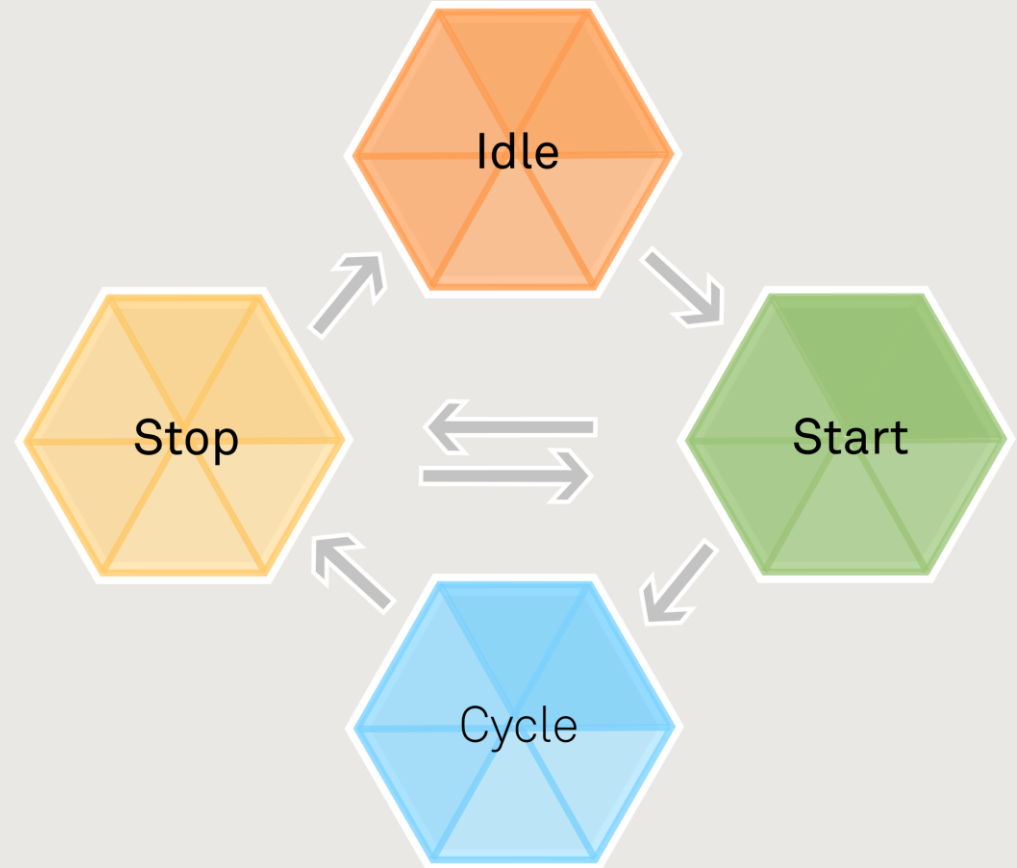
Highly Responsive Navigation

- Controlling responsiveness
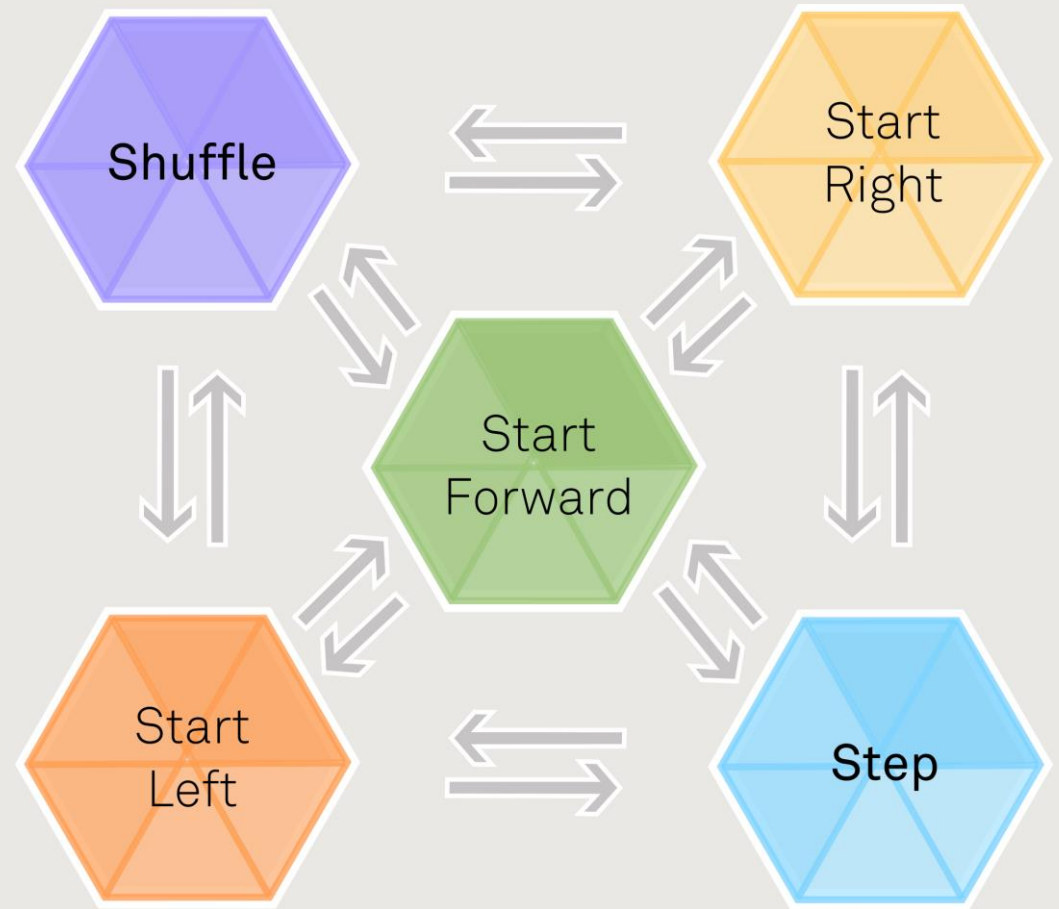
Highly Responsive Navigation

- *Problem:*
  - Uncontrollable when letting go of stick during start!
  - Too much displacement with locomotion in stops.
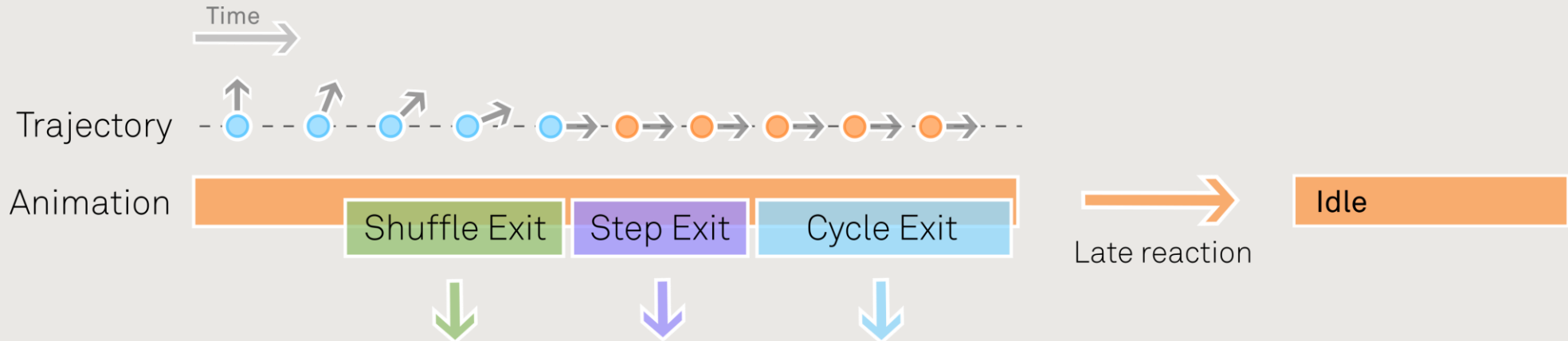
Highly Responsive Navigation

- *Solution:*
  - Shuffles
    - Quick directional motions with small displacement forward
  - Steps
    - Small forward motions that switch footedness

Highly Responsive Navigation

- Move became false
  - Time window in the animation decides when to trigger a step or a shuffle.
- Move still true
  - Early blend to the cycle and eventually a stop

Vast World Of Horizon: Zero Dawn

Complex Terrain Navigation

- No navigation mesh for player!
- Metrics
  - 1.5m high jump, 50 degree max slope, 6m/s max speed
- Capsule-shaped collider
  - Different sizes (max 70cm wide, 1.8m tall)

Complex Terrain Navigation

- Need to know ground surface gradient
  - Ignoring high frequency differences!
- *Solution:*
  - Smoothed slope angle calculation
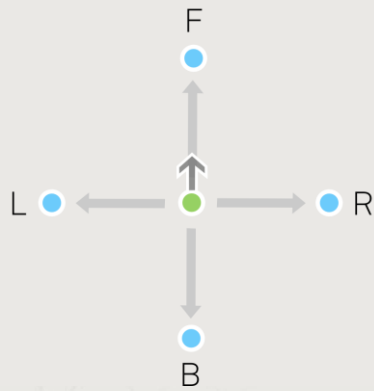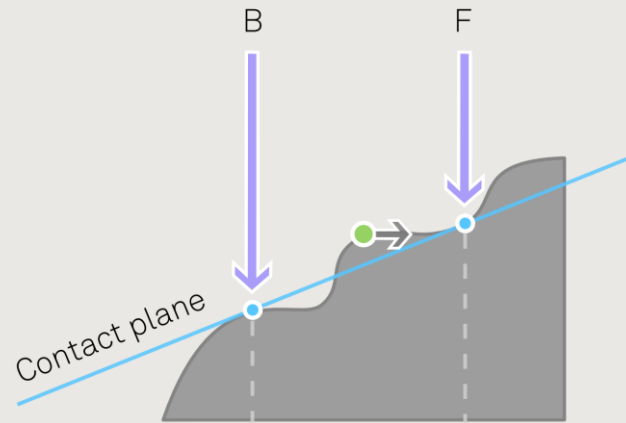   - Through multiple scheduled collision probes

Complex Terrain Navigation
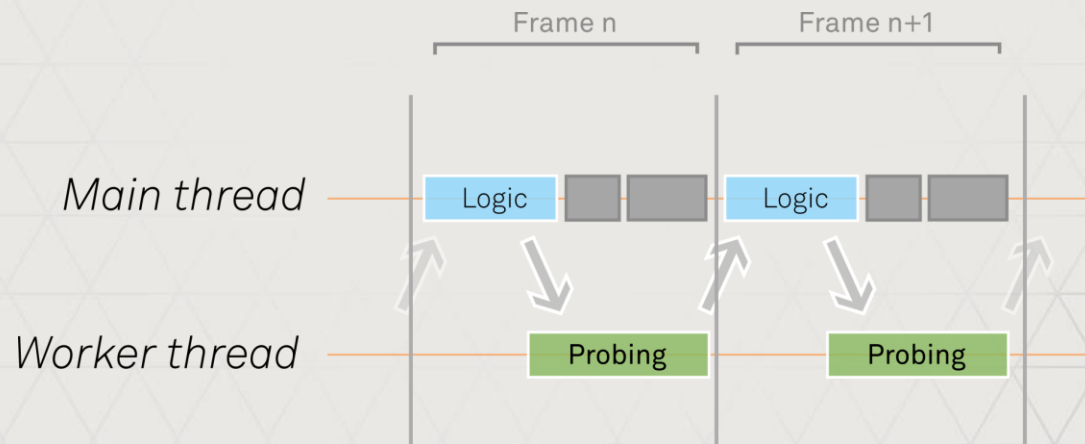
- Constructing a contact plane

Top view	Side view

$$\vec{n} = \left(h\vec{it}_R - h\vec{it}_L \right) \times \left(h\vec{it}_F - h\vec{it}_B \right)$$

Complex Terrain Navigation

- ## Using async probing results from previous frame

  – *Important:* Lagging one frame!
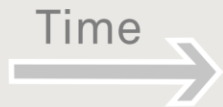
Foot Placement

- Procedural adjustment of legs and feet
- Animation decides when
  - Event-based
- Code decides where
  - Depending on game state and physics probe results

Foot Placement

- Placing feet on uneven terrain
  - Raycast the collision mesh
    - From knee height
  - Two bone IK chain from ankle to knee to hips, with adjustment of pelvis as well
  - Foot rest events in all animations

Time →

Cycle Animation

| Left Foot | Right Foot | Left Foot | Right Foot | Left Foot | Right Foot |

Foot Placement

- Floating foot

Foot Placement

- Solving the "floating foot problem"
  - Additional collision probing
  - Not only apply vertical adjustments

Without feet ser

Vast World Of Horizon: Zero Dawn
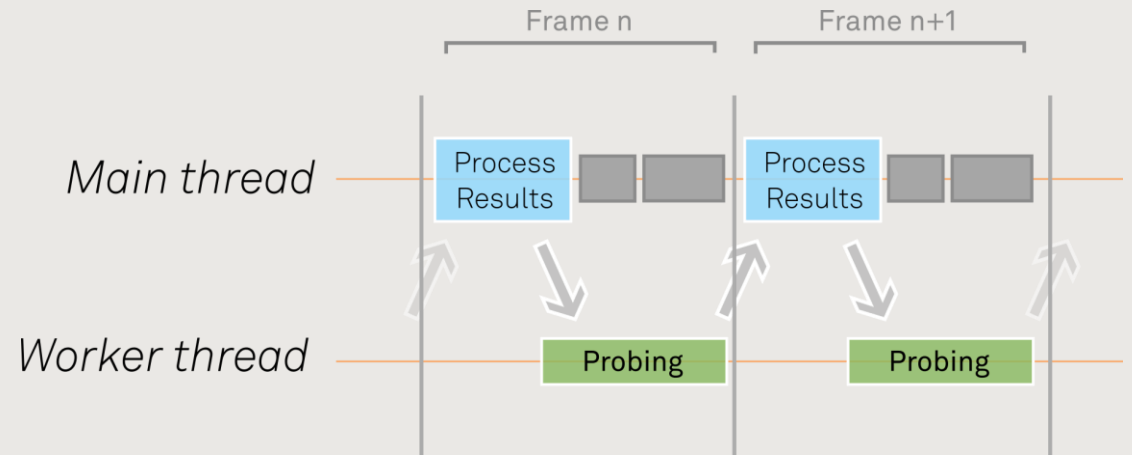
Vaulting

- Handling various surface features
  - Step up, step over, step off
 - Different detection settings per movement context

Vaulting

- Step 1:
  - *Allowed to vault?*
  - *Schedule probes*
- Triggers can disable vaults
- Game assets can disable vaults

Vaulting

- ## Step 2:
  - Process results from previous frame scheduled probe
    - *If:* Collision point high enough? (step up/over)
      - Do raycasts for shape analysis
    - *Else:* Horizontal displacement relative to starting point of swept sphere? (step off)
      - Find valid landing position

Horizontal Displacement
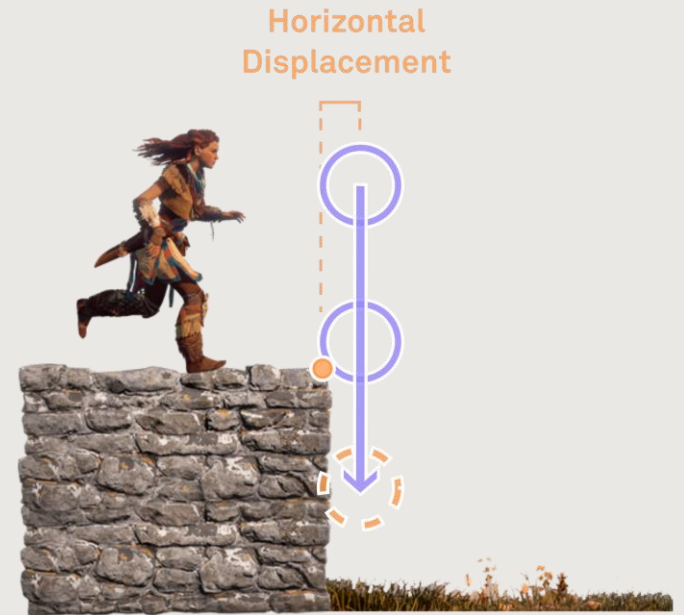
Vaulting

- Step 3:
  - Shape analysis
    - Raycasts with fixed offsets determine depth and height fluctuation
  - Store obstacle metrics
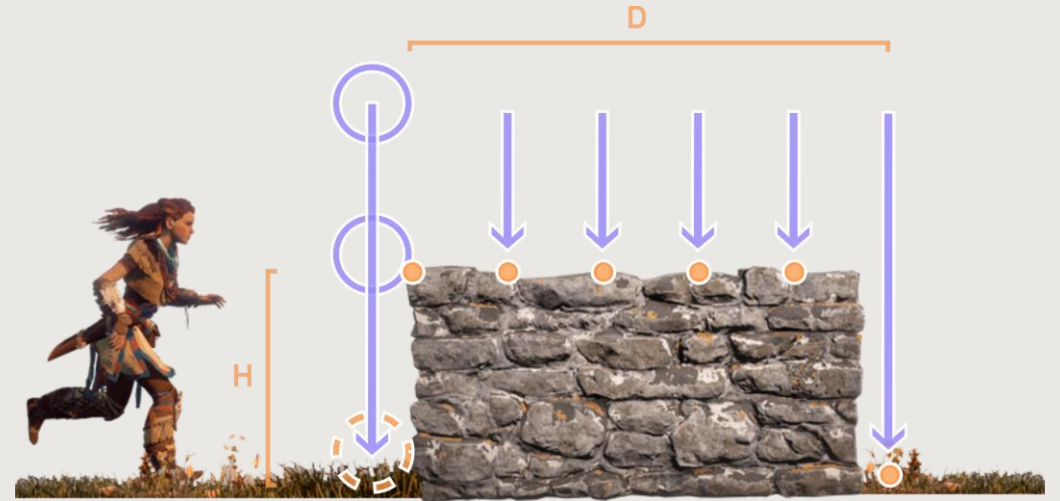
Vaulting

- Transition selection
  - Scoring system
    - Difference between obstacle contact position and animation contact position
    - Add bonus to climbable transitions

Vaulting

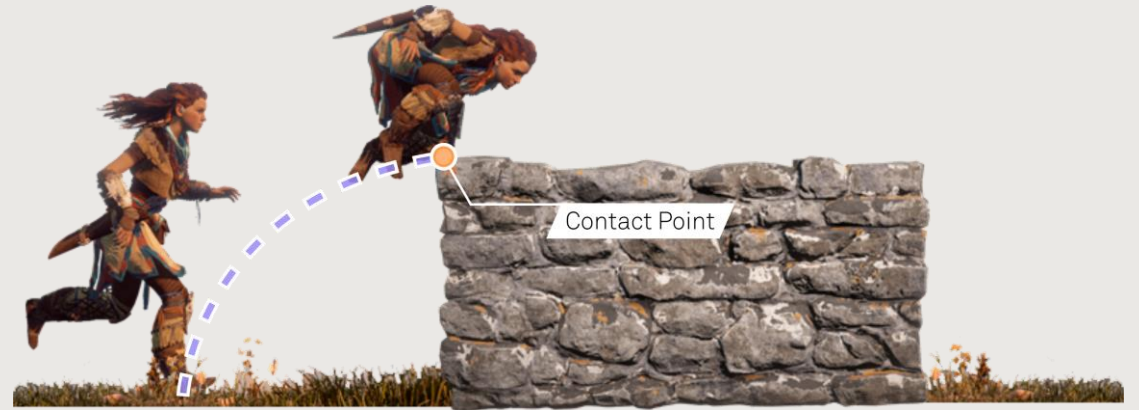- *Problem:* How to match animation with obstacle interaction
- *Solution:* Warp it!

Contact Point

Animation Warping

- Bending and stretching animated motion
    - To reach a specific position at a specific time
- Less animation variations needed
- Destination position can be adjusted during playback

Animation Warping

Animated
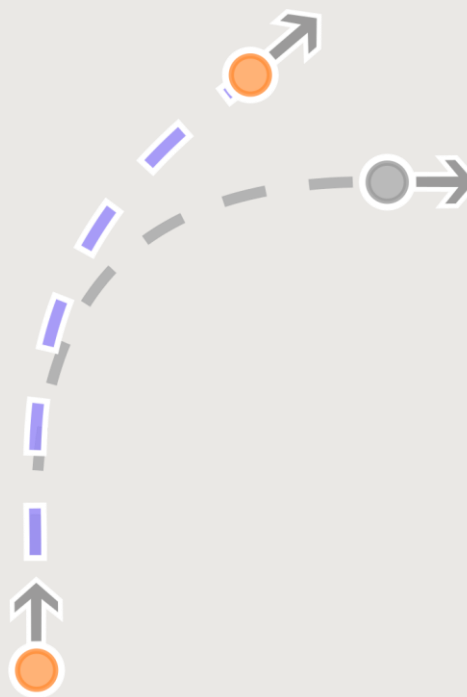
Warped

Animation Warping

- Requires analysis of the animation
  - Need to know the total remaining displacement at any time in the animation

Animation Warping

Displacement per axis in source animation

Animation Warping

Displacement per axis in source animation

Animation Warping

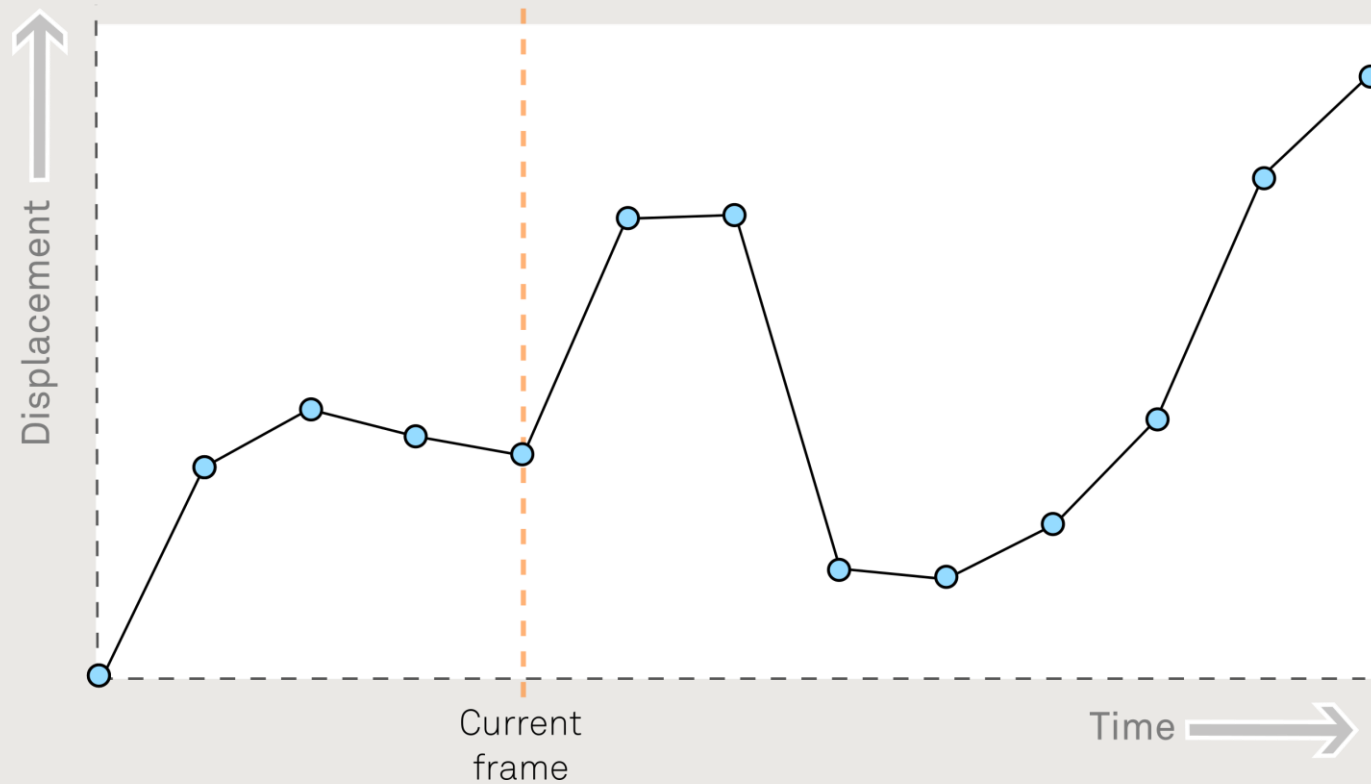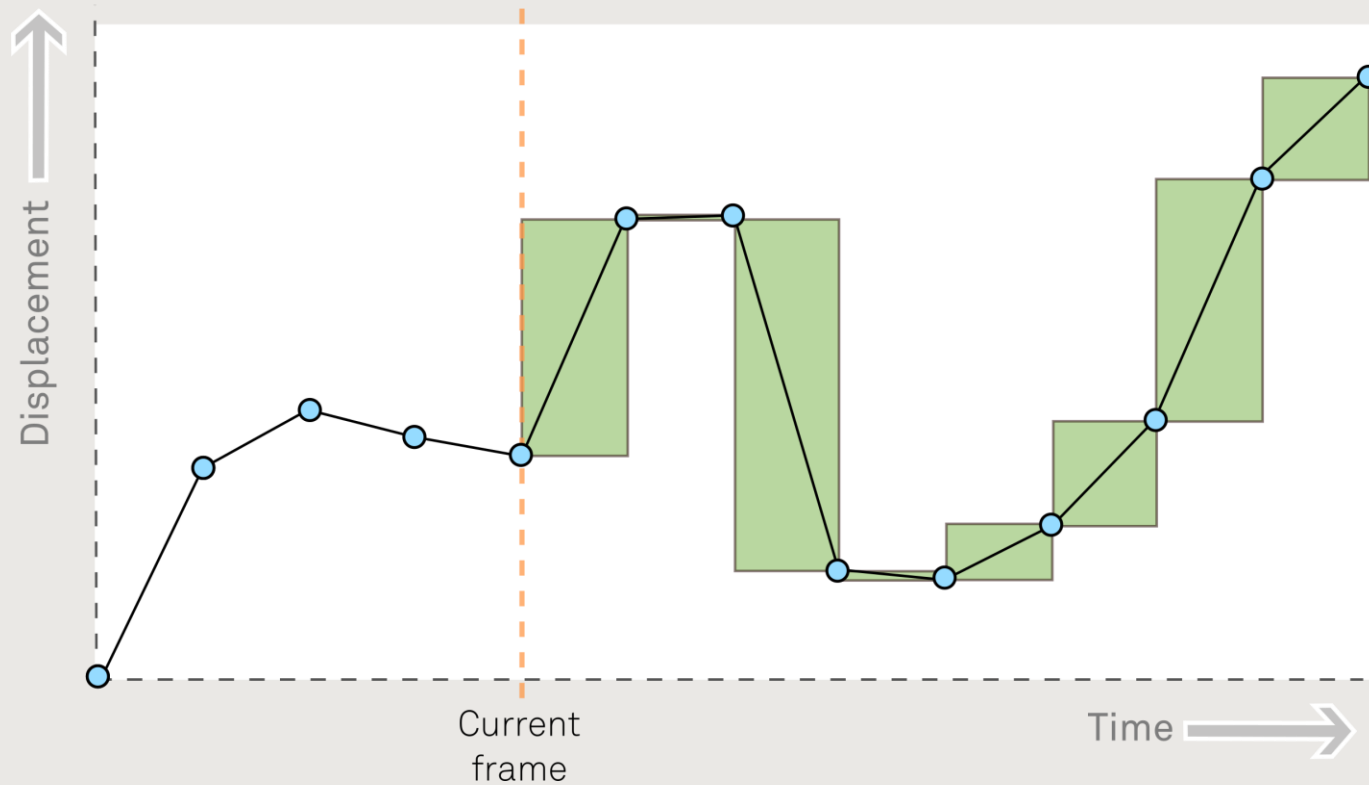Displacement per axis

Remaining displacement

Current displacement

Animation

Current frame — Next frame — Last frame

Distance

Distance to destination

Warped

Current frame — Δx — Next frame — Last frame
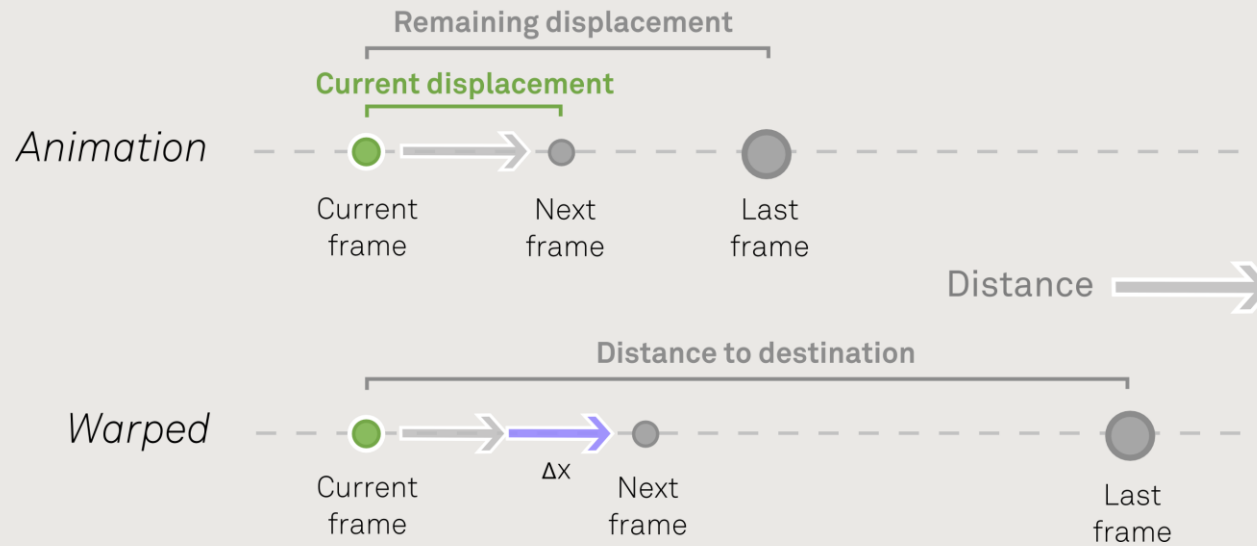
With $\Delta x = \dfrac{D_{current}}{D_{remaining}} D_{destination}$

Animation Warping

- *Enhancements:* Allow any bone to reach the given destination
  - Calculate offset of the guided bone to trajectory bone each frame and subtract this offset from the destination
  - Important: Trajectory is defined by motion of guided bone!

Animation Warping

- *Enhancements:* Only warp during specific time ranges in the animation
    – Warping looks best when feet are not touching ground!
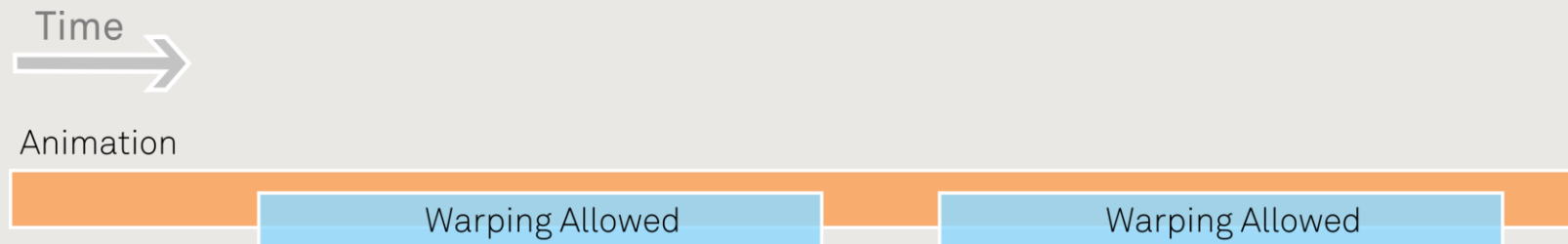
Time

Animation

| Warping Allowed | | Warping Allowed |

Animation Warping

Displacement per axis in source animation
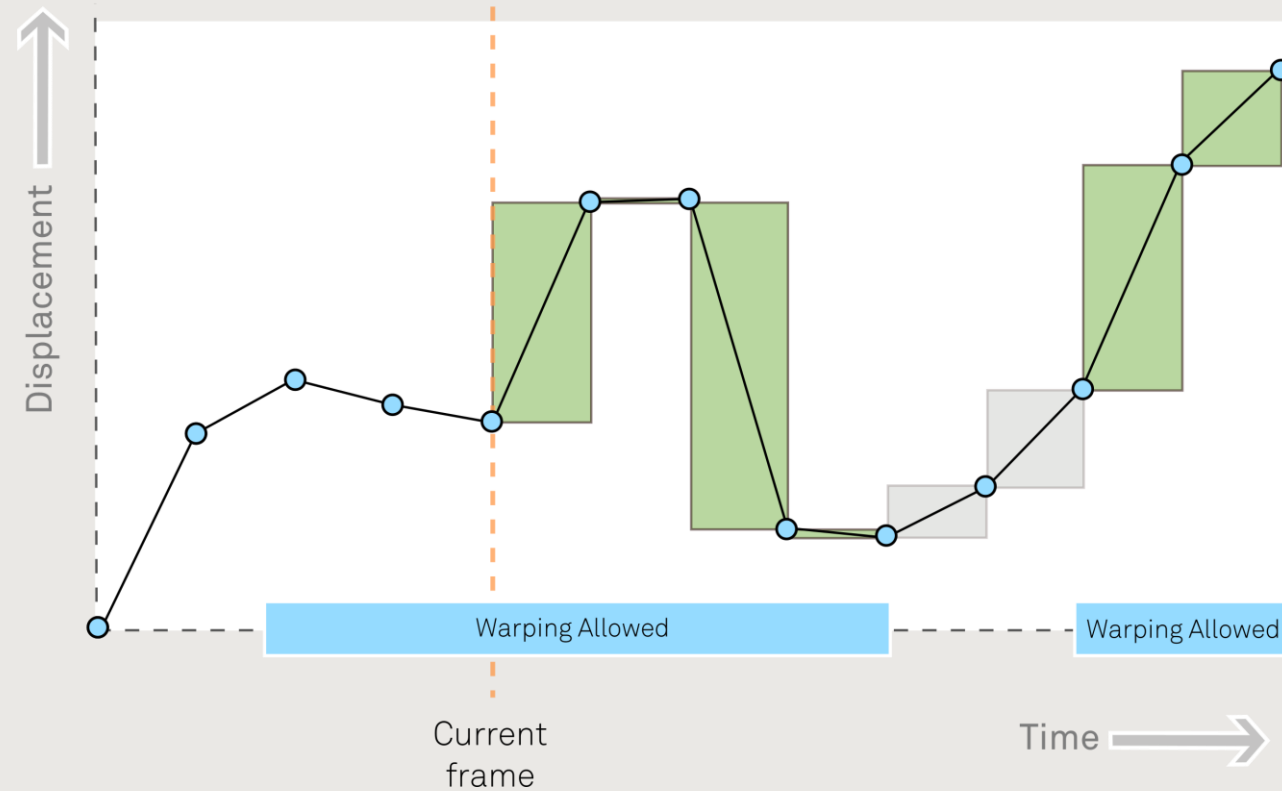
Animation Warping

- *Enhancements:* Specific arrival time
  - *Problem:* Don't always want to reach destination at end of animation
  - *Solution:* Support user defined arrival time
    - Indicated by event in animation
    - Allowing for post arrival motion

Animation Warping

Displacement per axis in source animation
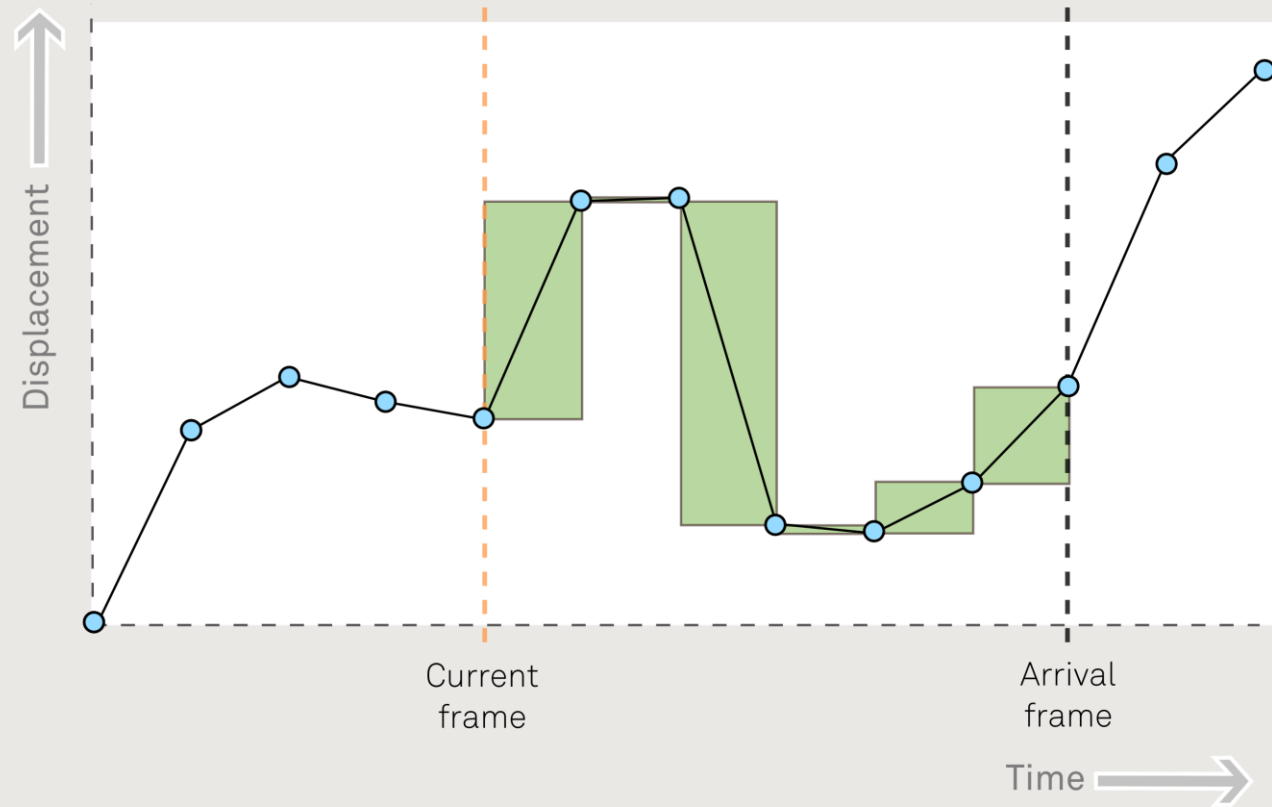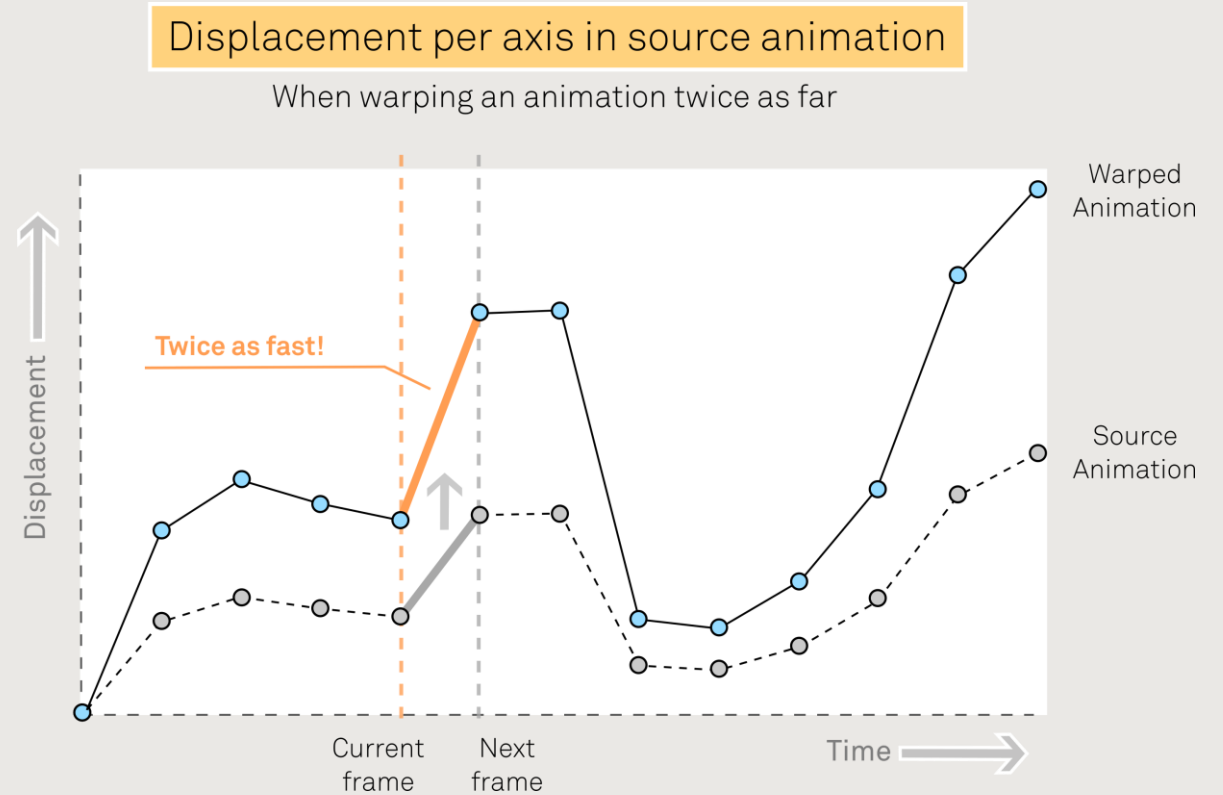
Animation Warping

- *Enhancements:* Preserving velocity by adjusting playback speed
  - *Problem:* Adding displacement can cause unnatural speed ups
  - *Solution:* Adjustment of playback speed allows you to preserve original animation velocity

Animation Warping

- *Example:* Warping an animation twice its own distance
  - Problem: Doubles the velocity

Displacement per axis in source animation

When warping an animation twice as far

Twice as fast!

Warped Animation

Source Animation

Displacement

Current frame

Next frame

Time

Animation Warping

- *Solution:*
  - Modify playback speed to twice as slow
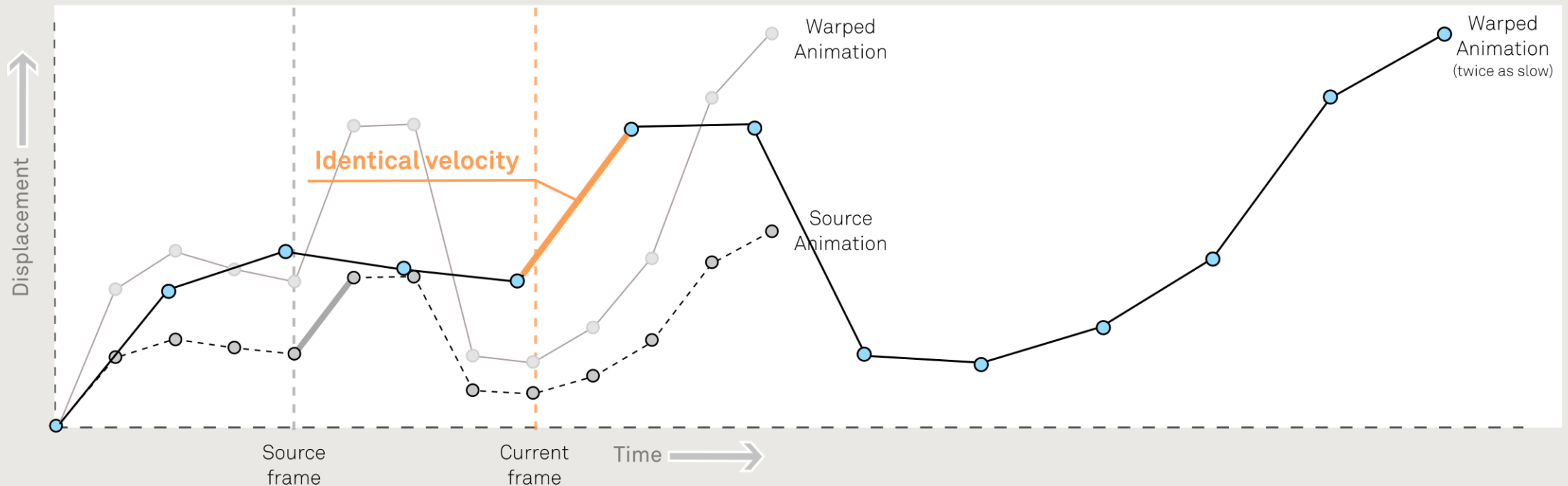
Vaulting

- Used for diving into water
- Used for climbing out of water
- Transition onto annotations

Vast World Of Horizon: Zero Dawn

Environmental Metadata

- Tile-based, stream-able world
    - 512m by 512m tiles
- More than 100 tiles!
    - 3x3 tiles always loaded

Environmental Metadata

- A single tile
  - Collision mesh
  - Water volumes
  - Stealth areas
  - Roads
  - Geometric annotations
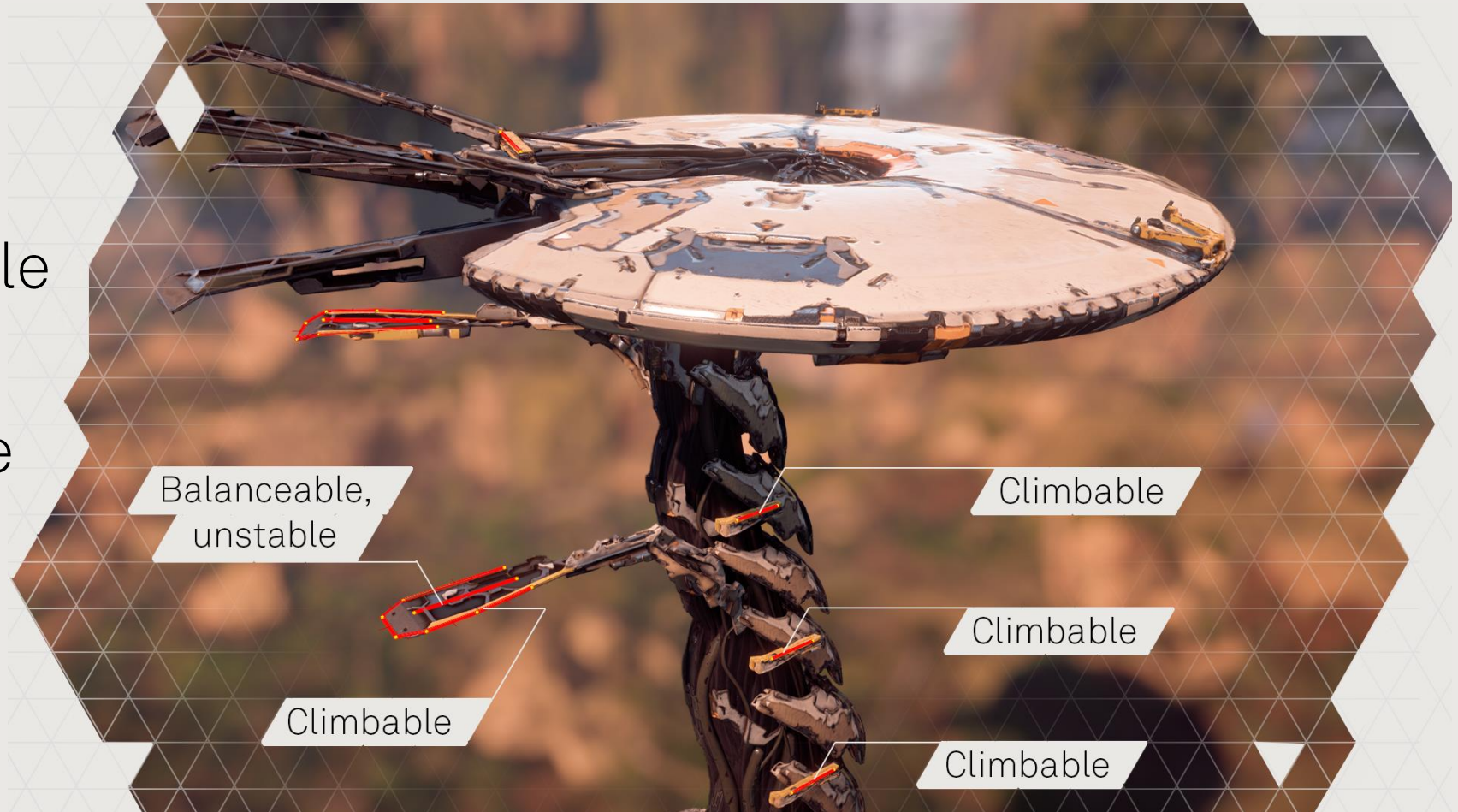
Geometric Annotations

- Points and lines
- Static or dynamic
- Tagged
- Stream-able
- Provides semantics

Geometric Annotations

- Tags
  - Climbable
  - Balanceable
  - Unstable
  - Ziplineable
  - Etc...

Jump System

- Introducing a new jump system
  - Pre apex (animation driven)
  - Post apex (procedurally)
- Improving control
  - In-air steering
  - In-air braking

Jump Guidance

- *Solution:*
  - Predict trajectory
  - Apply small adjustments for guidance
- *Important:*
  - Maintain the illusion of control!

Motion Analysis

- Motion prediction and planning requires animation metadata:
  - Duration, distance, rotation, average speed, etc...
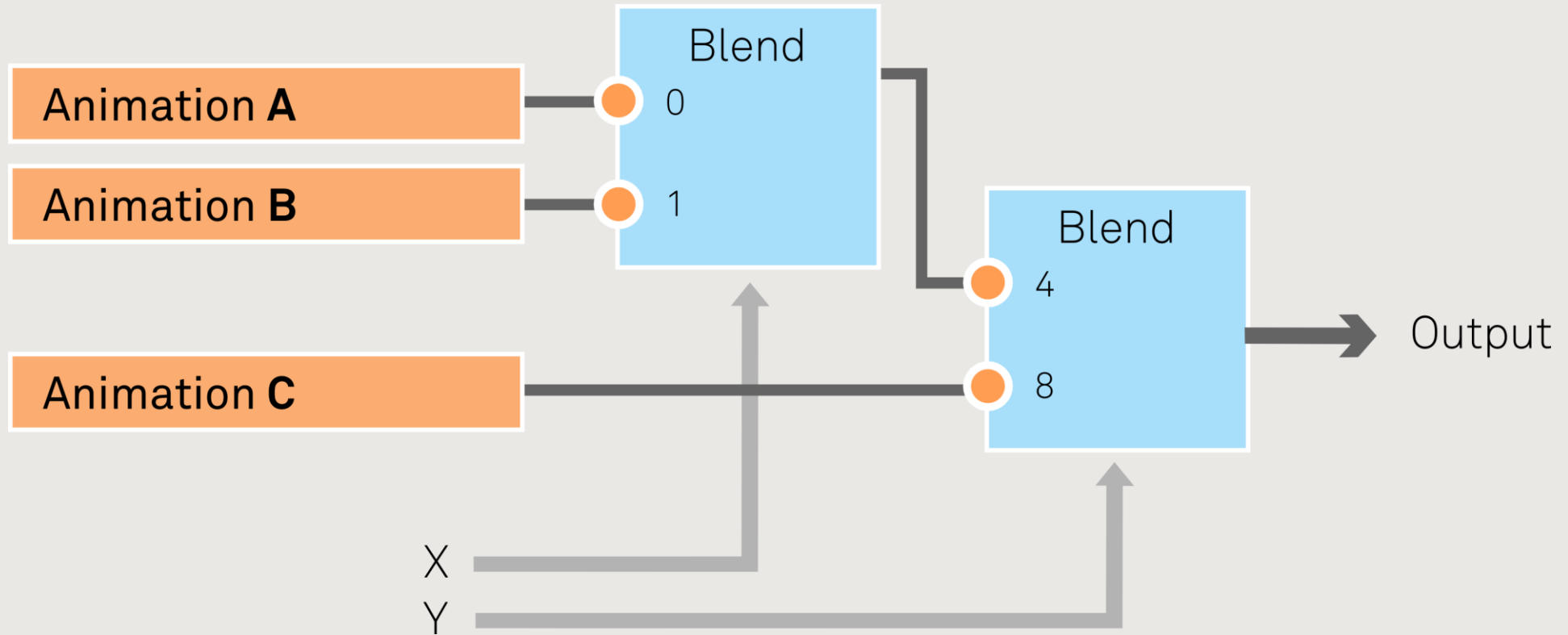- *Solution:*
  - Motion tables

Motion Analysis

- Calculate and store the metadata in a database for all valid permutations of input variable values for the animation state

- When querying metadata from the database blend several results together, based on the given values for the input variables
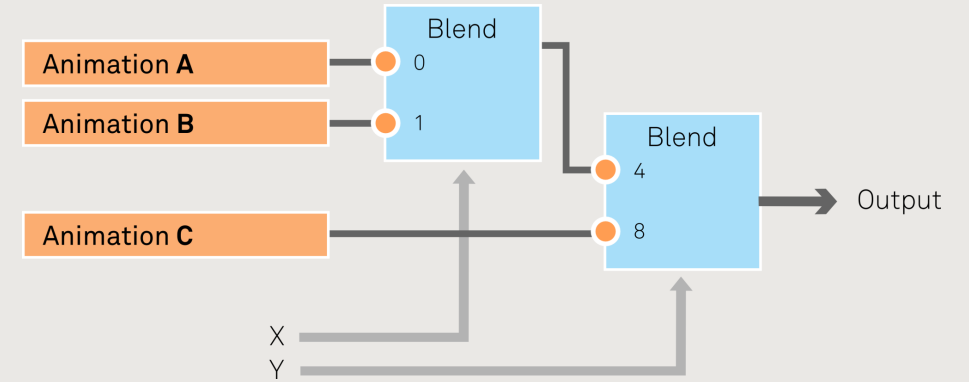
Motion Analysis: Construction

- Input samples:
  - X: 0 or 1 and Y: 4 or 8
- Store metrics for 3 permutations of input values

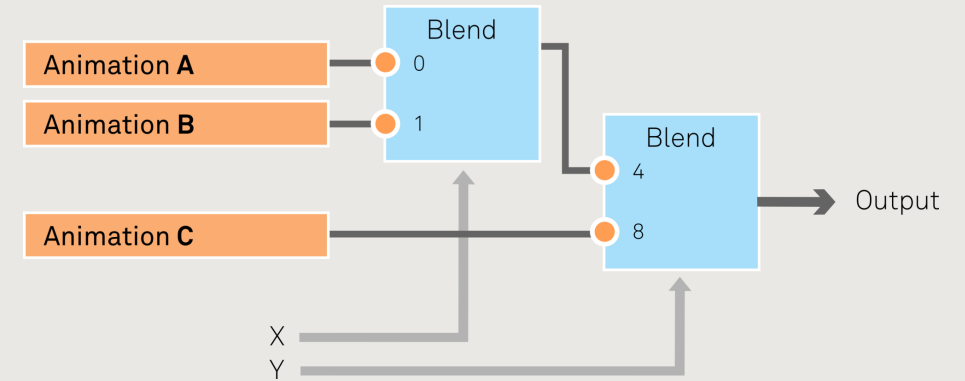# Motion Analysis: Querying

- ## Metrics for (X=0.25, Y=6)?
  - 0.5*0.75*SampleA +
    0.5*0.25*SampleB +
    0.5*SampleC

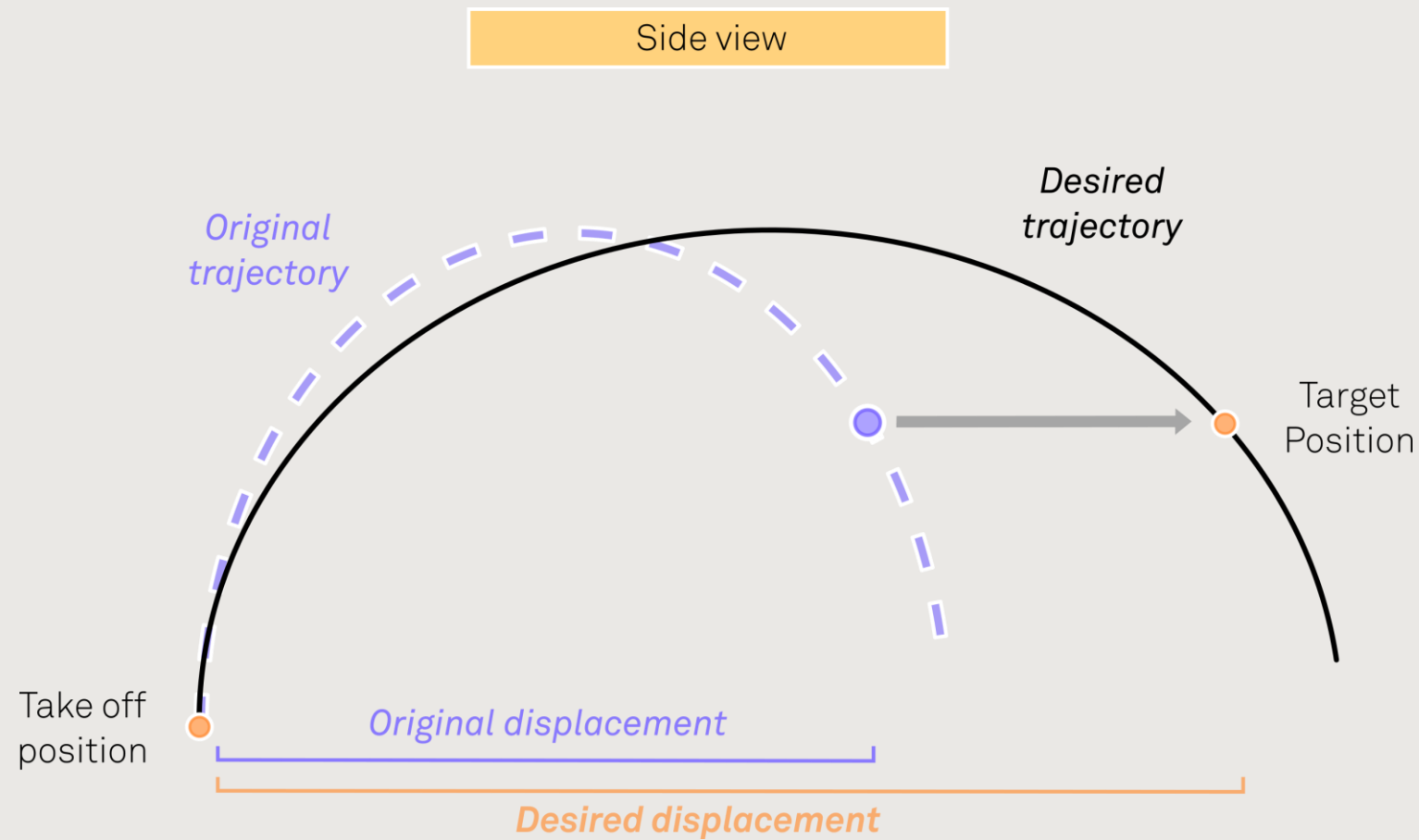| Sample	X	Y
Sample A	0	4
Sample B	1	4
Sample C	–	8

Constructing The Trajectory

- ## Apex position:
  - – Take off position + jump motion displacement
- ## Initial fall velocity:
  - – Velocity from the last frame of the jump motion

Side view

Apex position

Take off position
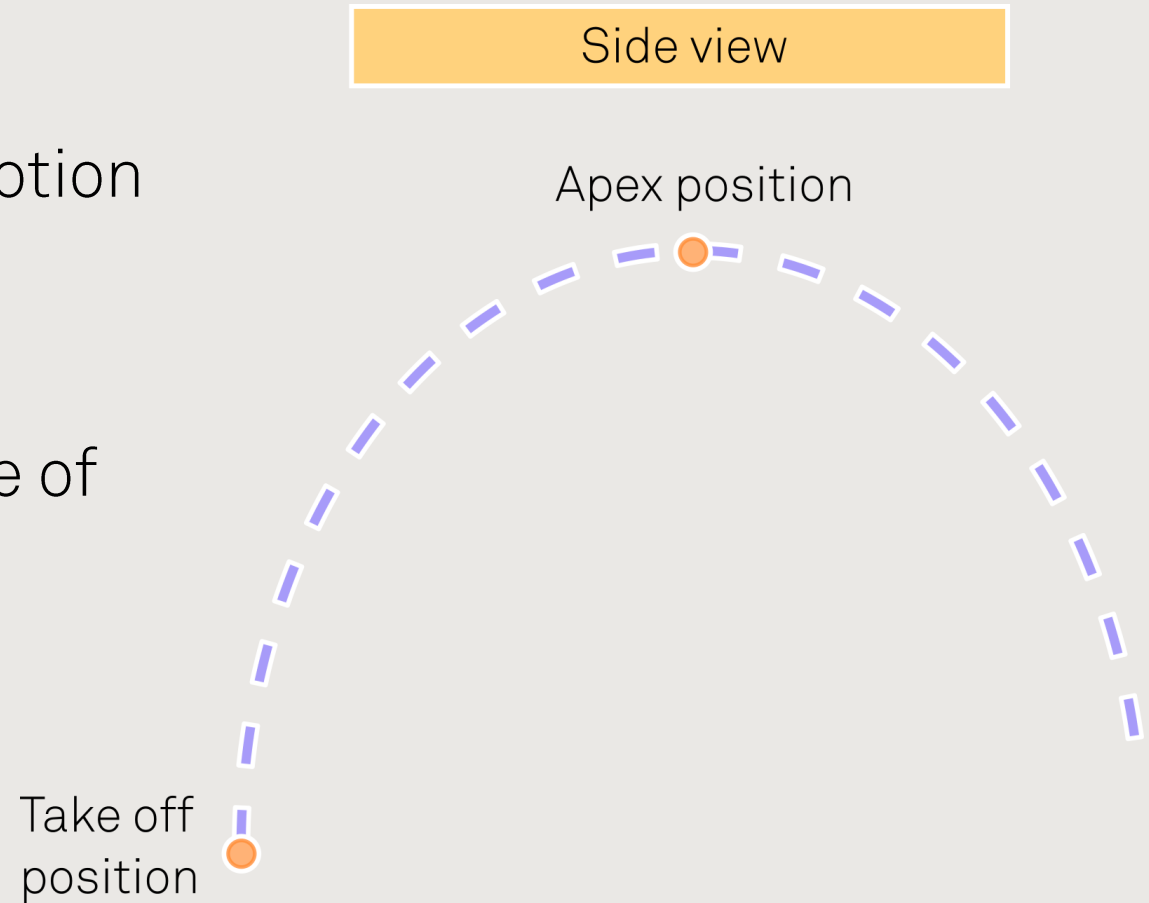
Jump Guidance: Forward Speed

Side view

Original trajectory

Desired trajectory
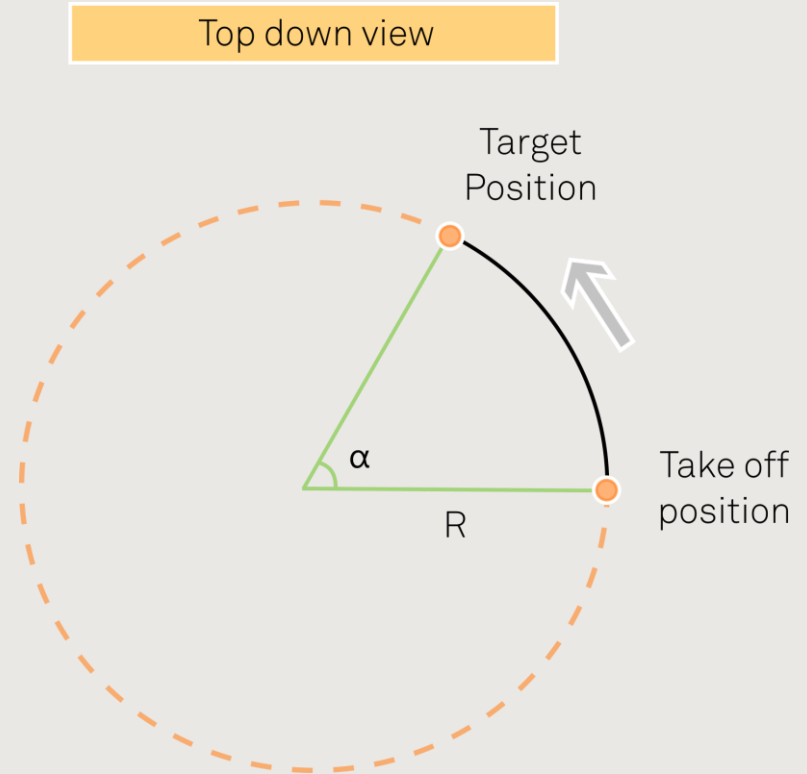
Target Position

Take off position

Original displacement

Desired displacement

$$c_v = \frac{D_{desired}}{D_{original}}$$

Jump Guidance: Turn Speed

- Calculate the circle that gets us there, in 2D

Target Position

Take off position

R

α

$$R = \frac{x^2 + y^2}{2|x|}$$

Vast World Of Horizon: Zero Dawn

Climbing

- Relying on geometry annotations
- One attach point on the annotation

Climbing

- Transition selection
  - Find nearest annotation in the direction of movement stick
  - Find best transition by matching conditions, ranges and check for blocking collision.
  - Trigger transition and apply warping

Climbing: Guiding Limbs

- Directing hands and feet while climbing
  - Hands: holding onto geometry
    - Current or destination
  - Feet: against geometry if possible, otherwise dangling

Climbing: Guiding Limbs

- Collision probes
  - Through the palm of the hand
  - Two sets for the feet
- Individual control of all four limbs
- Limb rest events in all animations
  - Locks the limb as well – sticks it to the geometry!

Climbing: Dynamic Objects

- Dynamic annotations
  - Handling different update frequencies
  - Taking displacement and rotation into account
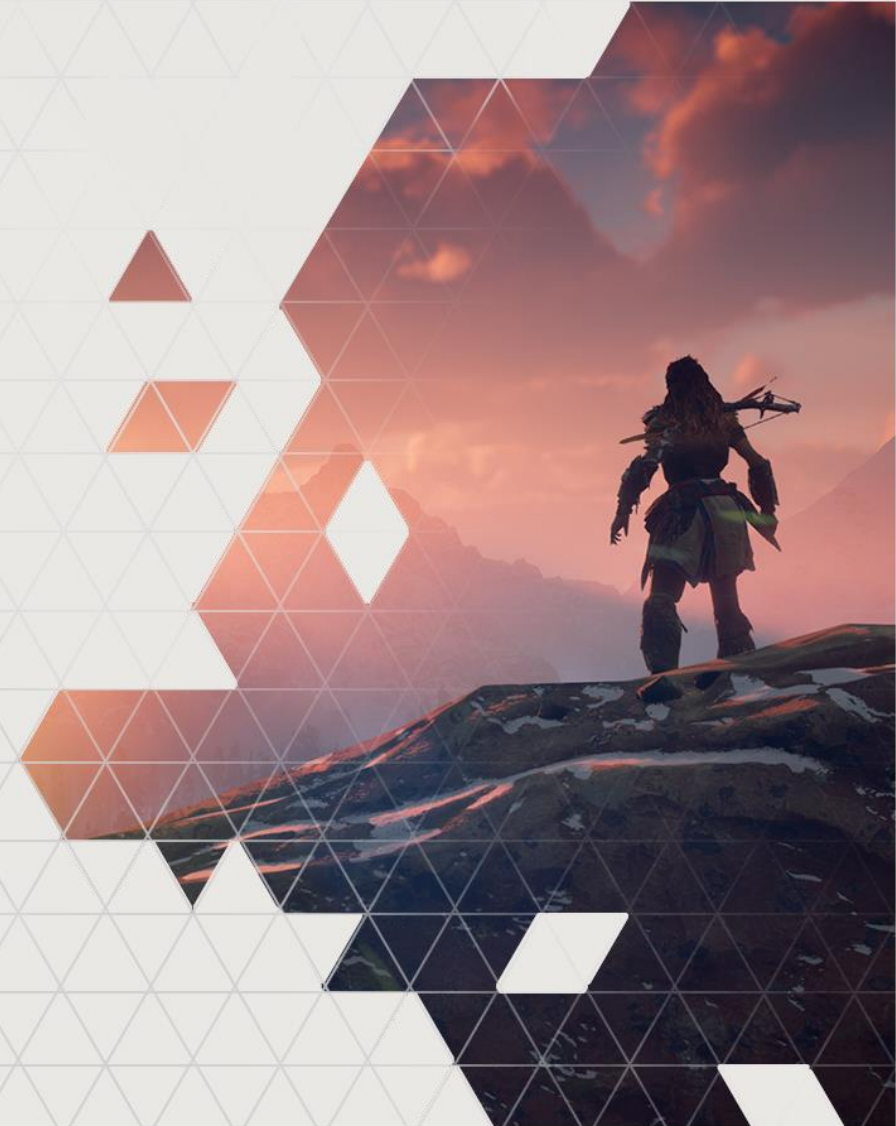- Dynamic IK calculation

Post Mortem

- What worked well?
  - Animation queuing based on events
    - Controllable responsiveness
  - Geometry annotations
  - Tweakable jump guidance
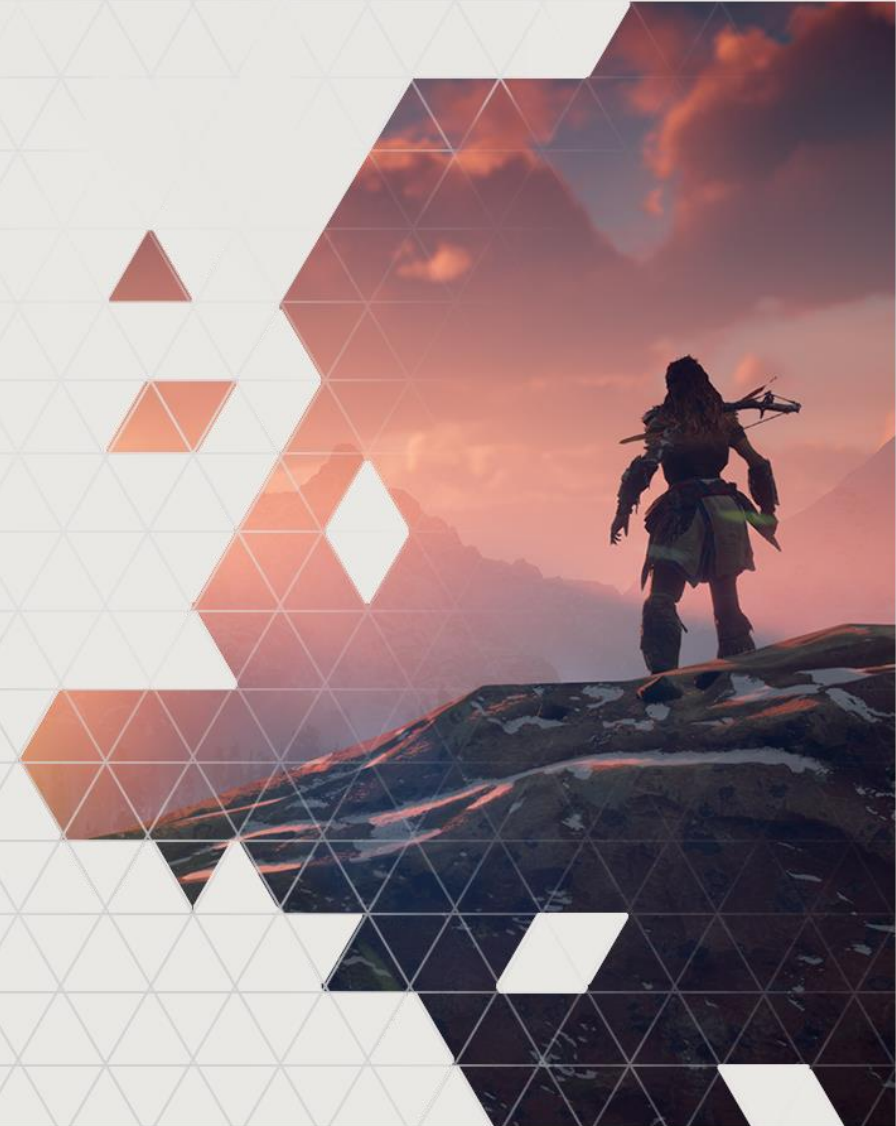   - Snappiness vs realism

Post Mortem

- ## What didn't work that well?
  - Lack of good editor/visualization tools caused level designers to not focus on traversal
    - Tools were still in development
    - Hard for level designers to build traversal paths
  - Vault system
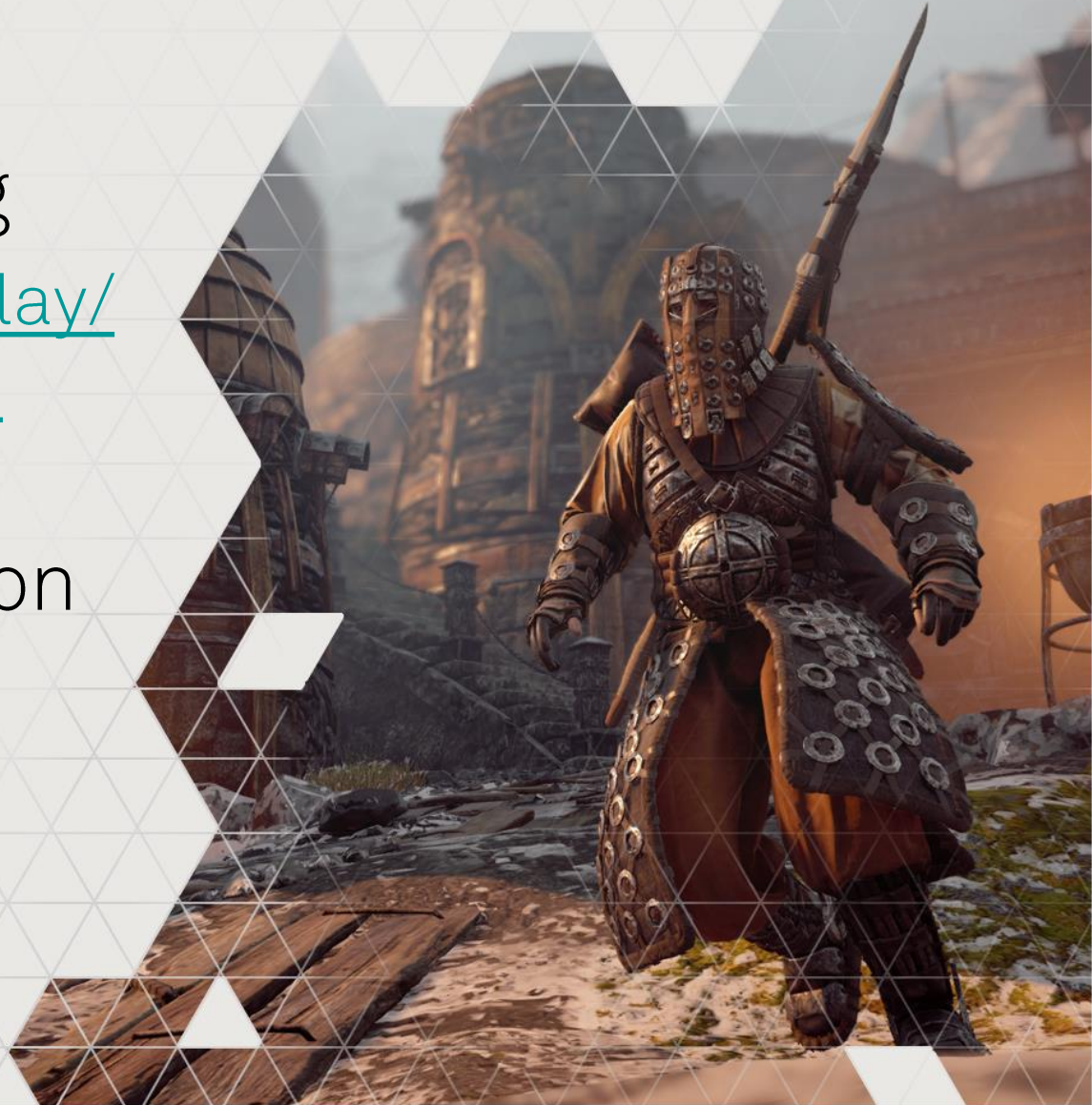 - No navigation mesh for player

Post Mortem

- What didn't work that well?
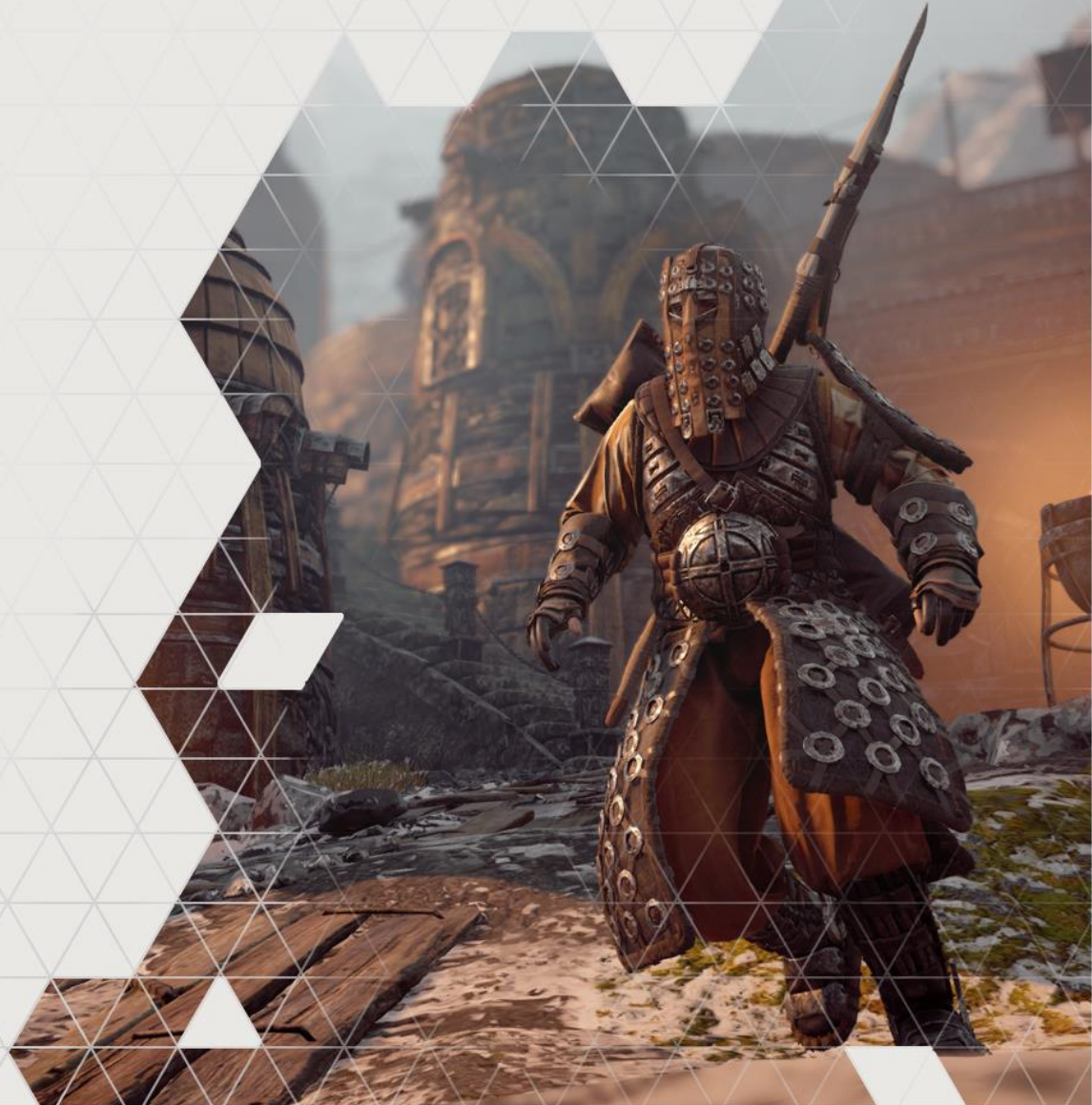 - Very complex animation networks

Future Plans

- Research motion matching
  - http://www.gdcvault.com/play/1022985/Motion-Matching-and-The-Road
- Investigate player navigation through use of navigation mesh
- Full body IK

Acknowledgement

- Guerrilla
- Team Traversal
- Thijs Kruithof

Thank you!

THANK YOU!

Q&A

- Questions?
- Also feel free to mail us!
  - paul.vangrinsven@guerrilla-games.com
  - thijs.kruithof@guerrilla-games.com