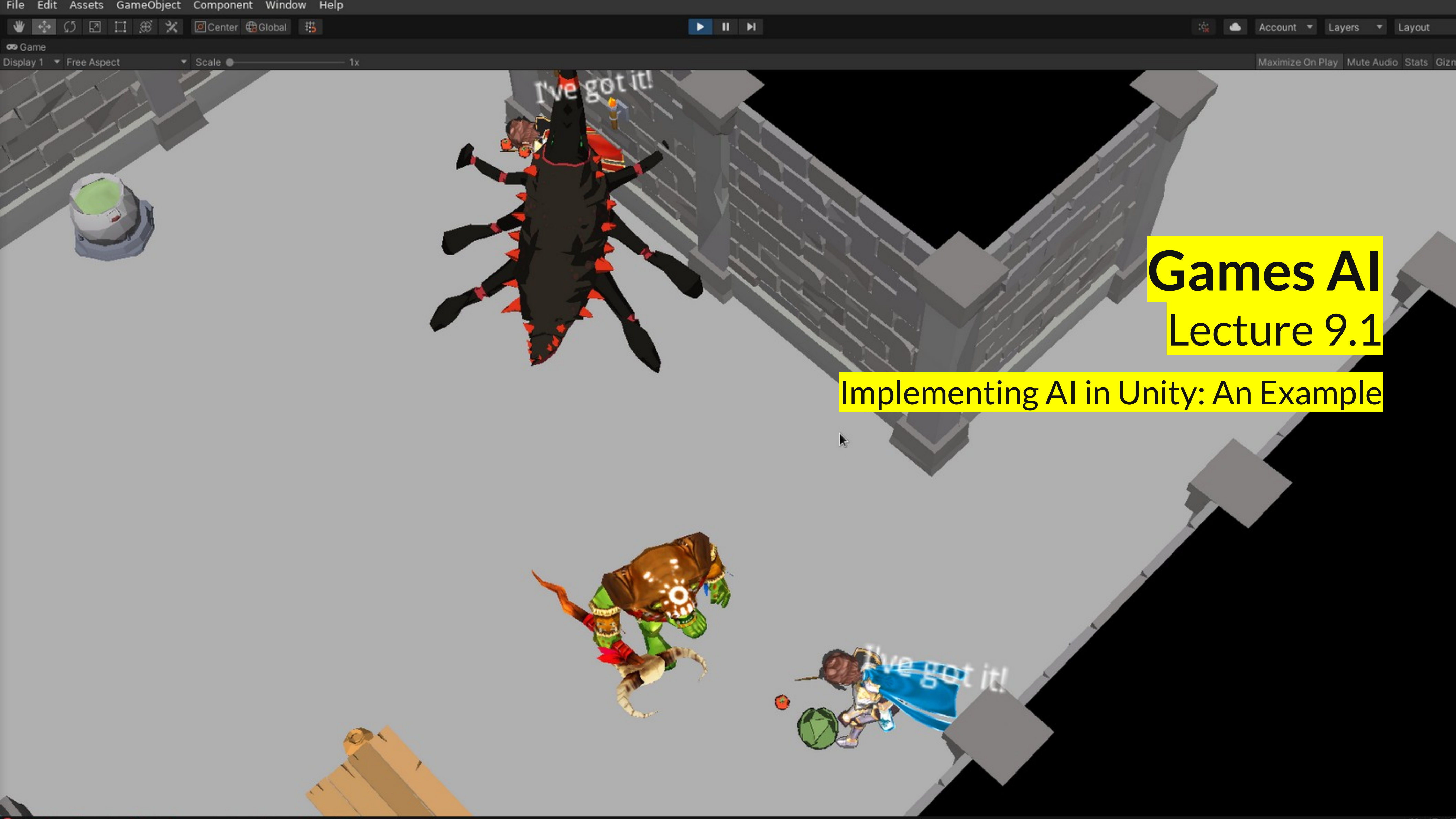


Games AI

Lecture 9.1

Implementing AI in Unity: An Example

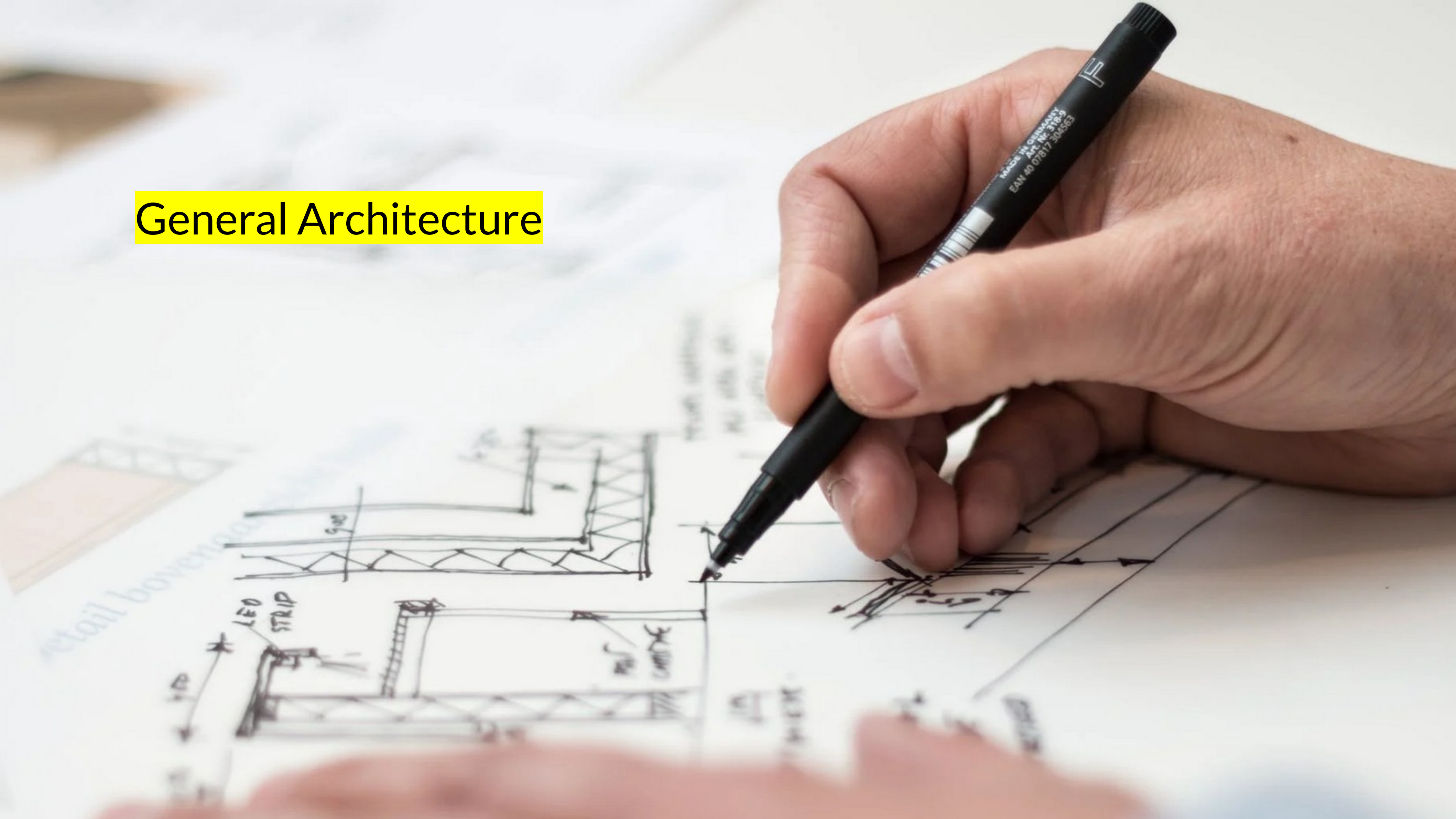
Overview

- Intro to Unity Components
- Representing objects/data
- Representing agents
- Representing actions
- Implementing Utility AI
- Implementing Heirarchical State Machine
- Modular AI

Overview

- Implementation is Hard,
  - not talked about much
  - game dependant
- Disclaimers
  - This is not the only way/best way
  - But it’s useful to see something fleshed out

General Architecture



General Architecture

- Agent
- AgentCommand
- Behaviours
- ObjectData
- Database

General Architecture

- Agent
    - On a GameObject
        - (Which may also have NavMeshAgent, etc.)
    - SetAction
        - Takes ActionCommand
        - Adds a behaviour (component) to the GameObject
        - Removes previous behaviour

General Architecture

- AgentCommand
  - What behaviour to run
  - What targets it should have
  - Any custom properties to set

General Architecture

- Behaviours
    - e.g. AttackBehaviour
    - MoveBehaviour
    - PickupBehaviour
  - Just a Unity Component
    - Start()
    - Update()
    - OnDestroy()

General Architecture

- ObjectData
  - Dictionary(s) of arbitrary data about an object
  - Events currently on DataObject (Unity Component wrapper)

General Architecture

- Database
  - Keeps track of every ObjectData in the game
  - Use a Query to get all matching criteria

Utility AI

Utiltiy AI

- UtilityAgent: Agent
- UtilityAI
- Data classes
  - UtilityInput
  - Consideration
  - UtilityOption

Utiltiy AI

- UtilityInput
  - Where our input comes from
  - In what range to clamp it

Utiltiy AI

- Consideration
  - What input to use
  - What response curve to apply

Utiltiy AI

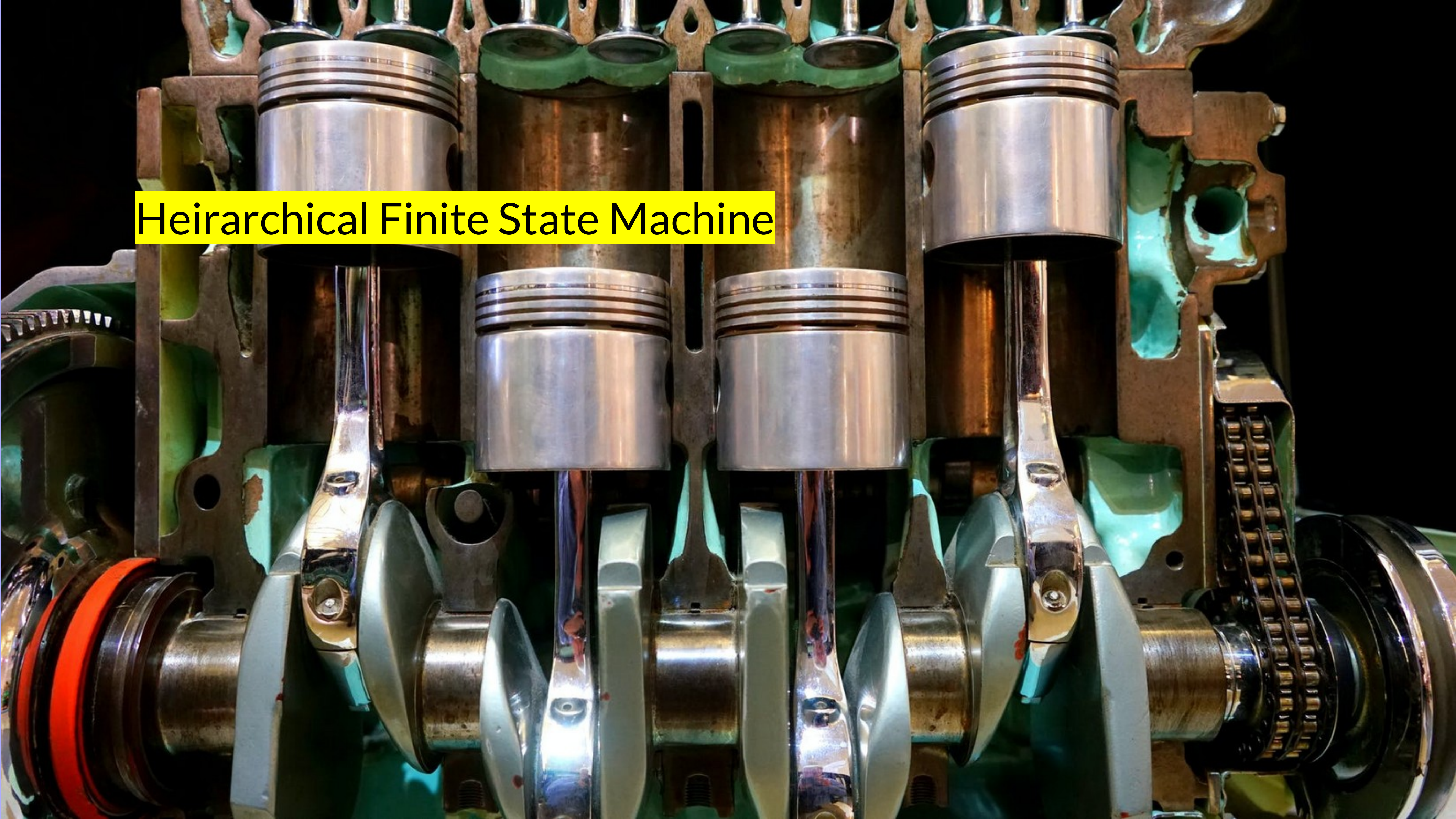
- UtilityOption
  - What behaviour to run
  - What considerations to combine
  - What targets we need to evaluate for
  - Custom properties for the behaviour

Utiltiy AI

- UtilityAgent: Agent
  - Loads possible actions from JSON
    - Dictionary<string, UtilityOption>
  - On Update:
    - Ask UtilityAI to evaluate UtilityOption
    - Ask UtiltiyAI to pick best
    - Uses the Agent code to instantiate that new behaviour

- UtiltyAI
  - Loads utility info from JSON
    - Dictionary<string, UtilityInput>
    - Dictionary<string, Consideration>
  - EvaluateAction
    - Scores UtilityOption for all possible target objects
    - Populates dictionary with AgentCommands
  - PickBest
    - Picks command with highest score

Heirarchical Finite State Machine



HFSM

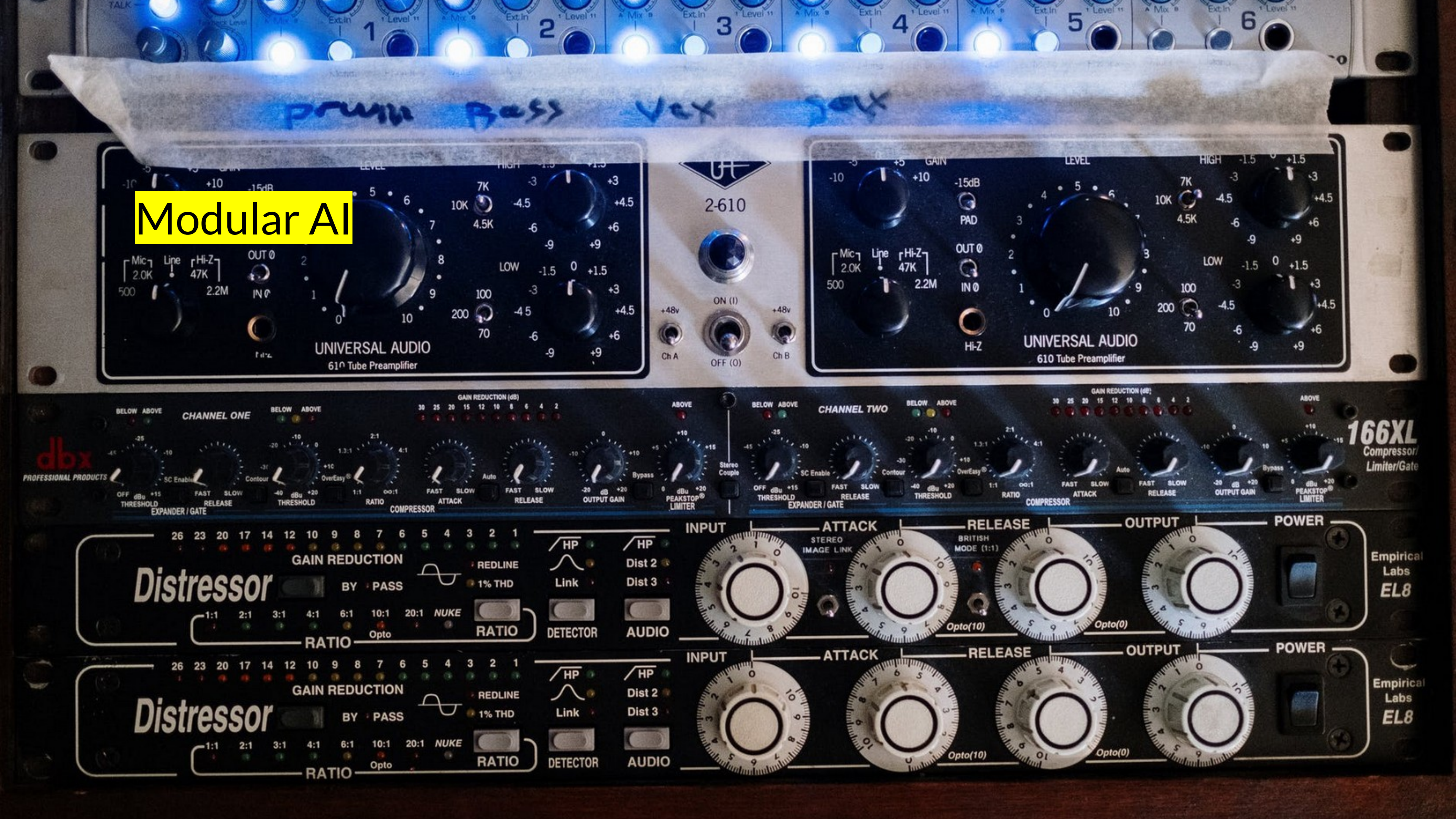
- StateAgent: Agent
- StateMachine

HFSM

- StateAgent: Agent
  - Holds StateMachine
  - Get events from DataObject
  - Send them to StateMachine
  - Listen to updates from StateMachine
  - Converts them to AgentCommands for SetAction on Agent

HFSM

- StateMachine
  - Implements Heirarchical Finite State Machine
  - Recursive
  - Recieves events from StateAgent
    - Can transition? Do so
    - Can’t? Pass down to child StateMachine
    - If changed state, get innermost state for StateAgent


UNIVERSAL AUDIO
610 Tube Preamplifier
2-610
dbx
PROFESSIONAL PRODUCTS
CHANNEL ONE
CHANNEL TWO
166XL
Compressor/
Limiter/Gate
Distressor
Empirical
Labs
EL8

Modular AI

Modular AI

- Sensing
  - Sensor Components
- Deciding
  - Agent
- Acting
  - Behaviours