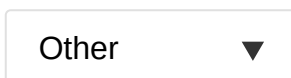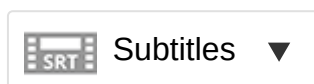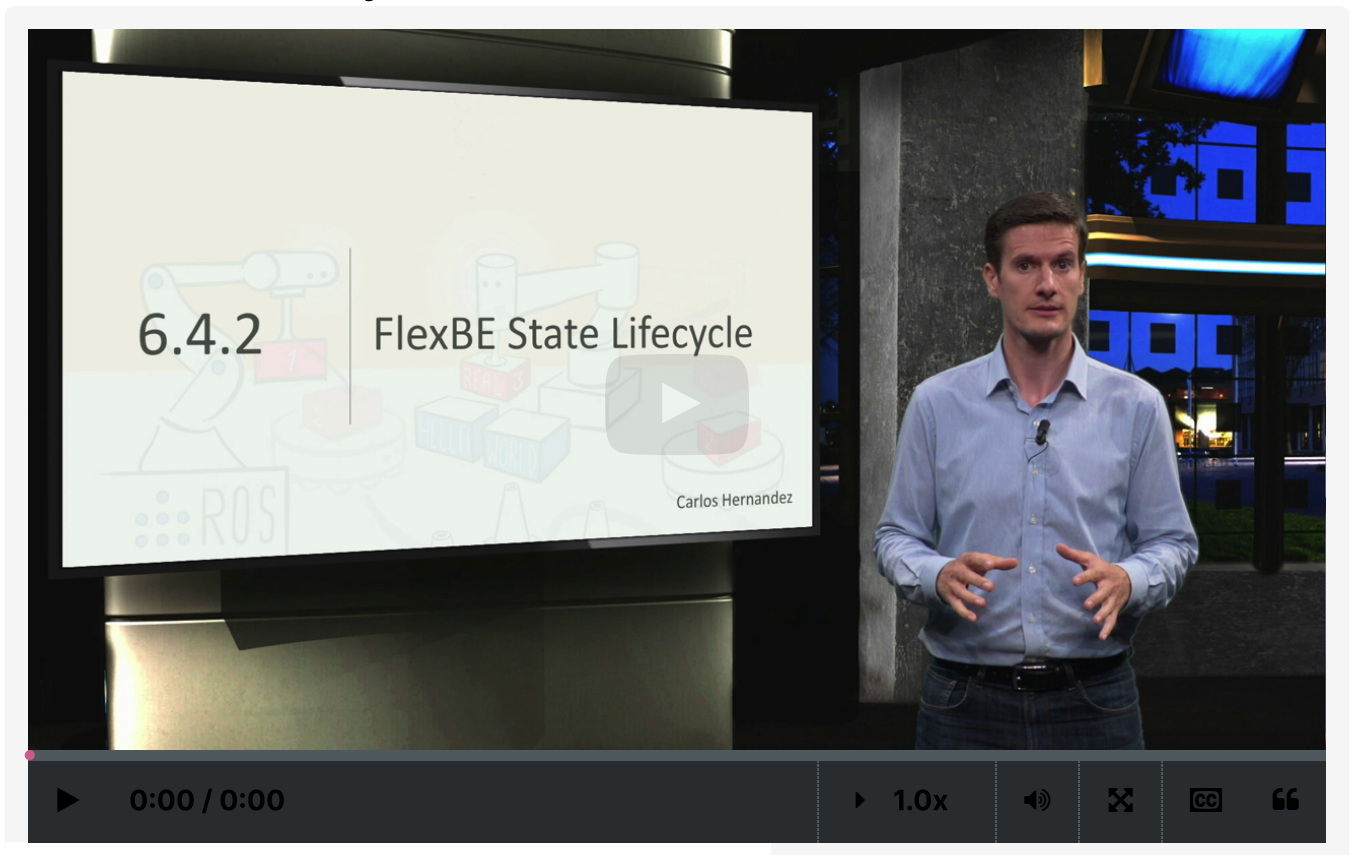In this lecture, we will learn about the lifecycle of FlexBE states. That is, what is happening in each of the state internals when we execute a behavior?

FlexBE States Lifecycle

| ▶ | 0:00 / 0:00 | ▶ 1.0x | 🔊 | ⛶ | CC | 66 |

| 🎞 Video ▼ | | 🎞 Subtitles ▼ |

| Other ▼ |

In this video we have learned the LifeCycle of FlexBE states, that is, what is happening within each state of the state machine when we execute a behavior.

FlexBE State Lifecycle:

- The FlexBE State Lifecycle is a sequence of function calls inside a state implementation, defined by the `EventState` Python class.

- The functions in the FlexBE Lifecycle are associated to events in the state machine, triggered by the behavior engine.

Sequential execution of the Lifecycle functions:

- `_init_` : when the state machine is instantiated, the `_init_` function of all the states is executed (it is the constructor of the states).

- `on_start` : this function is executed for all the states when the behavior starts executing (this provides a good place to initialize the resources used by the states)

- `on_enter` : executed only once when a state becomes active (typically the most used function, it is where any action or service request should be sent).

- `execute` : after `on_enter`, the `execute` method is called periodically while the state is active. Its main purpose is to check state conditions and trigger a corresponding outcome if the corresponding condition is met.

- other functions are: `on_exit`, `on_stop`

Note: parallel state execution (so multiple states being active at the same time) is supported by FlexBE, but outside of the scope of this course.

In the next video we will use the Lifecycle functions to program our own state implementation.

Question 1

1 point possible (ungraded)
Select the true statement.

The FlexBE State Lifecycle:

○ Defines the sequential call of functions inside the implementation of the FlexBE states during behavior execution.

◯ Defines the sequence of state activations in the behavior state
    machine.

Submit