Statecharts model of a CD player

Systeemsimulatie 2002

The assignment

Draw (in your preferred drawing tool) and explain (in plain English), by means of a model in the Statechart formalism, the design of the dynamics of a CD Player software application. The application consists of three main parts: the static GUI part (widgets), the dynamic part (Statechart), and the hardware part (which has a given software interface). This assignment is about the Statechart part which will obviously have links with the other two parts.

Application Requirements

1. The application shall be closed by closing its window (no explicit Quit button).

2. There shall be seven buttons named: Play/Pause, Stop, Previous, Next, Forward, Reverse, and Eject.

3. There shall be a Time text field to display time information.

4. There shall be a Track text field to display the current track.

5. When there is no disc in the CD player or when the CD player drawer is open, the system shall be in a state named `No CD Loaded`.

6. When the CD player drawer is closed and a CD is in the CD player, the system shall be in one of three possible states: `CD stopped`, `CD Playing`, or `CD Paused`.

7. When in the `No CD Loaded` state, with the drawer open and a CD in the drawer, pressing the Eject button shall cause the drawer of the CD player to close and the system shall enter the `CD Stopped` State.

8. When in the `No CD Loaded` state, with the drawer open and no CD in the drawer, pressing the Eject button shall cause the drawer of the CD player to close and the system shall remain in the `No CD Loaded` state.

9. When in the `No CD Loaded` state, pressing the Play/Pause, Stop, Previous, Next, Forward and Reverse buttons shall have no effect.

10. The `CD Playing` State is entered from the `CD Stopped` state by a user clicking the Play/Pause button.

11. The `CD Stopped` State is entered from the `CD Playing` state by a user clicking the Stop button.

12. The `CD Paused` State is entered from the `CD Playing` state by a user clicking the Play/Pause button.

13. The `CD Playing` State is entered from the `CD Paused` state by a user clicking the Play/Pause button.

14. When in the `CD Stopped` state, the Time field shall display 00:00 and the Track field shall display [track 1].

15. When in the `CD Stopped` state, the Stop button shall be disabled.

16. When in the `CD Stopped`, `CD Playing` or `CD Paused` states, clicking the Next button when the current track is not the last track on the CD will cause the CD player to move to the next track, the Time field will display 00:00 and the Track field shall display the track number.

17. When in the `CD Stopped`, `CD Playing` or `CD Paused` states, clicking the Next button when the current track is the last track on the CD will cause the CD player to move to the first track, the Time field will display 00:00 and the Track field shall display [track 1] and the `CD Stopped` state will be entered.

18. When in the `CD Stopped`, `CD Playing` or `CD Paused` states, clicking the Previous button when the current track is not the first track on the CD will cause the CD player to move to the previous track, the Time field will display 00:00 and the Track field shall display the track number.

19. When in the `CD Stopped`, `CD Playing` or `CD Paused` states, clicking the Previous button when the current track is the first track on the CD will cause the CD player to move to the start of the first track, the Time field will display 00:00 and the Track field shall display [track 1].

20. When in the `CD Stopped`, `CD Playing`, or `CD Paused` states, pressing the Forward button will cause the CD to stop playing and the CD to step forwards in one-second intervals. Each step will take (in real-time) no more than 0.1 seconds. Fore each step, the Time field will display the current track time and the Track field will display the current track number. The application will stop stepping through the CD when the user stops holding down the Forward button, or the end of the CD is reached. If the end of the CD is reached, the `CD stopped` state will be entered, the Time field will display 00:00 and the Track field shall display [track 1].

21. When in the `CD Stopped`, `CD Playing`, or `CD Paused` states, pressing the Reverse button will cause the CD to stop playing and the CD to step backwards in one-second intervals. Each step will take (in real-time) no more than 0.1 seconds. Fore each step, the Time field will display the current track time and the Track field will display the current track number. The application will stop stepping through the CD when the user stops holding down the Reverse button, or the start of the CD is reached.

22. While in the `CD Playing` state, the Time field shall be updated every second with the elapsed playing time of the current track and the Track field shapll display the current track number N as [track N].

23. When in the `CD Paused` state, the values in the Time and Track fields will be displayed initially and then after one second will be hidden (blank). After another second, they will be displayed again. This displaying/hiding cycle will continue as long as the system is in the `CD Paused` state.

You may need to add/modify requirements as you construct the statechart model. This may be due to the under-specification or even inconsistencies in the requirements.

A possible concrete User Interface (static part, consisting of a collection of widgets) may look like below.

It is assumed that appropriate events are generated by the widgets. For example, pressing the Eject button will generate an `Eject` event. You must explicitly link these mappings between the static part of the model (the widgets) and your Statechart. At an implementation level, this is often called "binding" of GUI widget events to callbacks implementing the dynamics (here described in the form of a Statechart).

Similarly, it assumed the Statechart has access to the widgets. For example, the Statechart is able to "disable" a button by invoking an appropriate method of such a button. You must explicitly list the methods of graphical entities you presume are present.

Software interaface to CD player hardware

- `open_drawer()`
  Causes the CD player drawer to open.
- `close_drawer()` → Boolean
  Causes the CD player drawer to close. When the drawer has been succesfully closed, the funtion returns True. Closing and already closed drawer also returns True.
- `stop()`
  Causes the CD player to stop and curren track and time information to be lost.
- `play(track_nr, time)`
  Starts playing the CD at the specified track number and time (in seconds) within that track. If the arguments are invalid, nothing will happen. When the end of the CD is reached during playing, the CD player will stop playing the CD. If the CD player was paused, calling play() will cause the CD to resume playing at the point where it was paused.
- `pause()`
  Causes the CD playing to pause. The track number and time within the track are remembered.
- `cd_loaded()` → Boolean
  Returns True if a Cd is physically in the machine, False otherwise.
- `current_track()` → Number
  Returns the current track number. Returns 0 if the CD is not playing or there is no CD in player.
- `current_time()` → Number
  Returns the elapsed time in the current track. Returns 0 if the CD is not playing or there is no CD in player.
- `previous_track()`
  Causes the CD player to step backwards one track from the current track. If the current track is the first track, this call will cause the player to move to the last track. The current time will be set to 0 whenever this method is called. If the CD is playing when `previous_track()` is called, it will continue playing after the step.
- `next_track()`
  Causes the CD player to step forward one track from the current track. If the current track is the last track, this call will cause the player to move to the first track. The current time will be set to 0 whenever this method is called. If the CD is playing when `next_track()` is called, it will continue playing after the step.
- `last_track()` → Boolean

3

Returns True if the current track is the last track.

- `forward_one_sec()`
  Causes the CD player to move forward one second from the current track and time. If the end of the CD is reached, the player will move to the start of the first track. If the CD is playing when this method is called, it will continue playing after the step.
- `back_one_sec()`
  Causes the CD player to move back one second from the current track and time. If the start of the first track of the CD is reached, the player will move to the end of the last track. If the CD is playing when this method is called, it will continue playing after the step.