



Privacyist

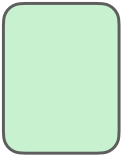
Legend



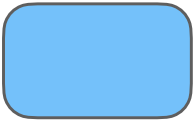
Privacyist: No



Payvist: Noode\*



Playvist: Song Entry



size\_

\_\_\_\_\_

head

tsini

\_\_\_\_\_

**OROV.**



size

stack memory

Heap memory

playivist • obiect





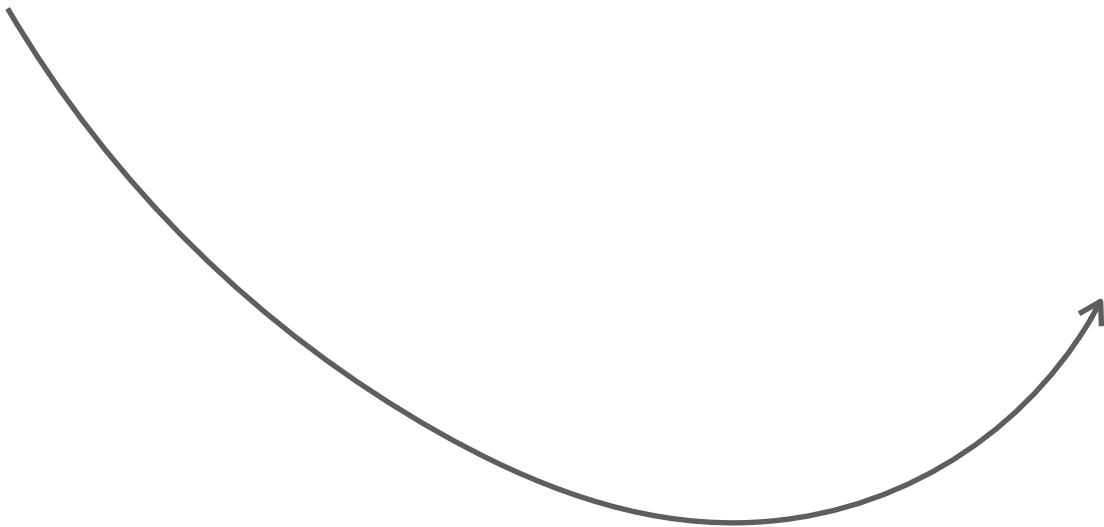


std::string

Let's insert another song:

```
playlist.insert_at_cursor(Playlist::SongEntry(20, "B"));
```







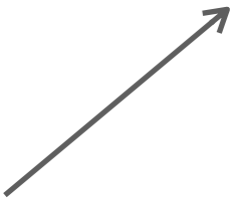
\_id

*-1*

\_name

*HEAD*

\_next



n u i p t r





The diagram illustrates a linked list node structure. It is divided into three horizontal sections by two black lines. The top section is a light blue rounded rectangle containing the text `_id`. The middle section is a light yellow rounded rectangle containing the text `_name`. The bottom section is a light purple oval containing the text `_next`. The entire diagram is enclosed in a thick magenta border.

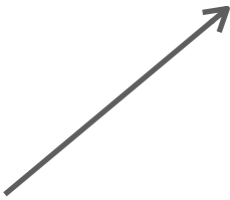
`_id`

`_name`

`_next`

10

A





The diagram illustrates a linked list node structure. It is enclosed in a thick magenta border. The top half is a light green rounded rectangle containing a blue rounded rectangle with the text `_id`. Below this is a yellow rounded rectangle with the text `_name`. A horizontal black line separates this top section from the bottom section, which is a light purple oval containing the text `_next`.

`_id`

`_name`

`_next`

20

*B*

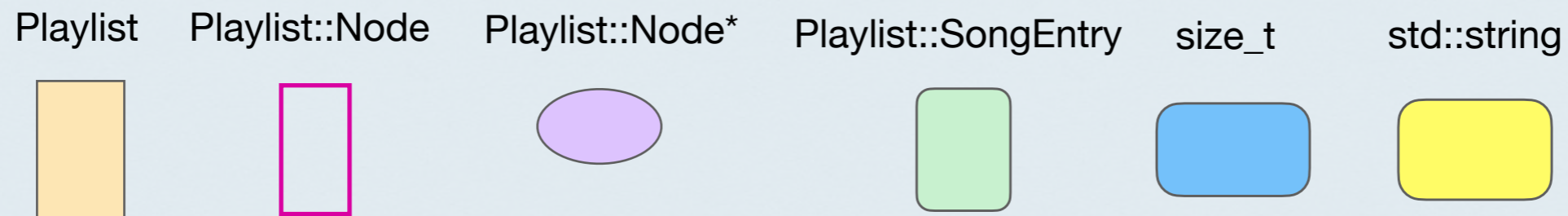
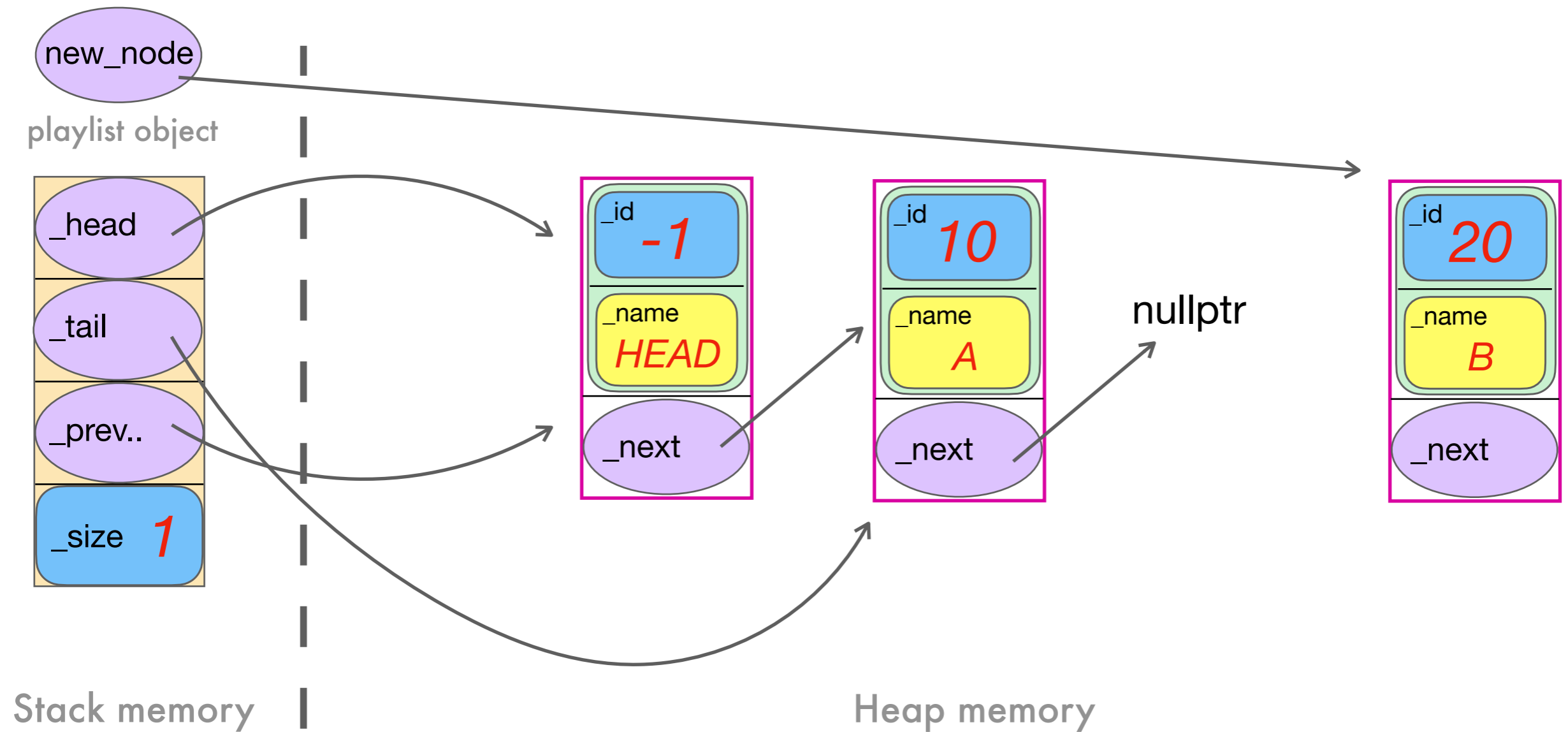


new\_node

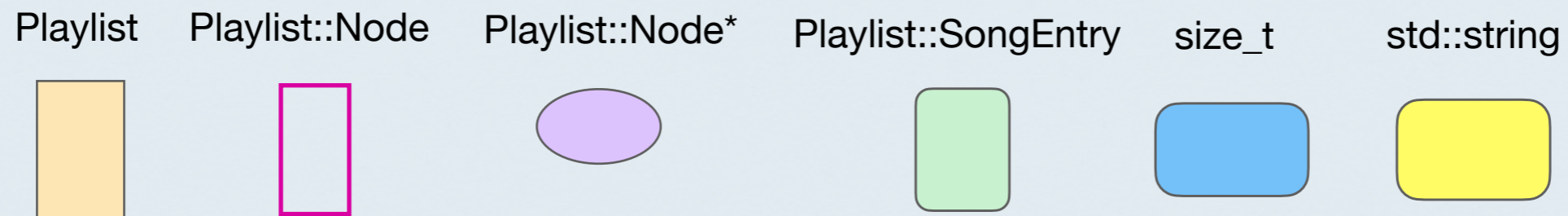
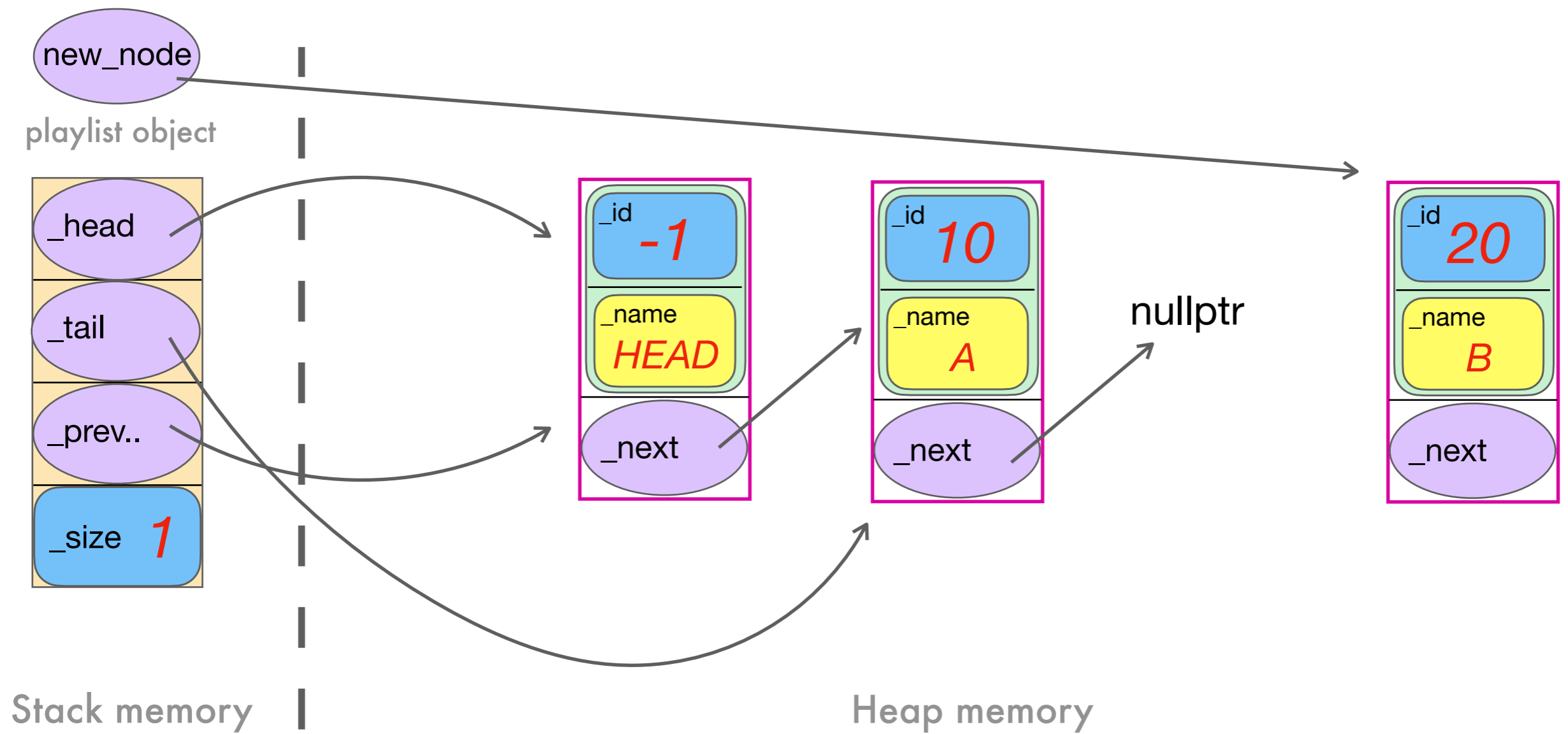


Let's insert another song:

```
playlist.insert_at_cursor(Playlist::SongEntry(20, "B"));
```



Legend



Legend