

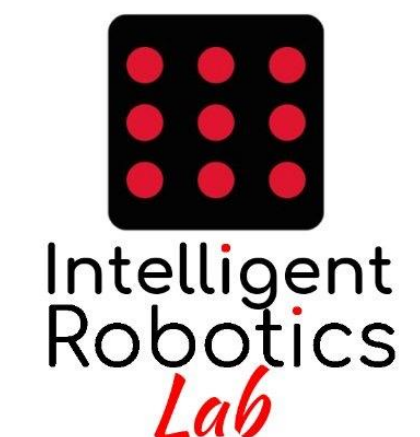


ikerlan

Course of
Robot Programming
with **ROS 2**

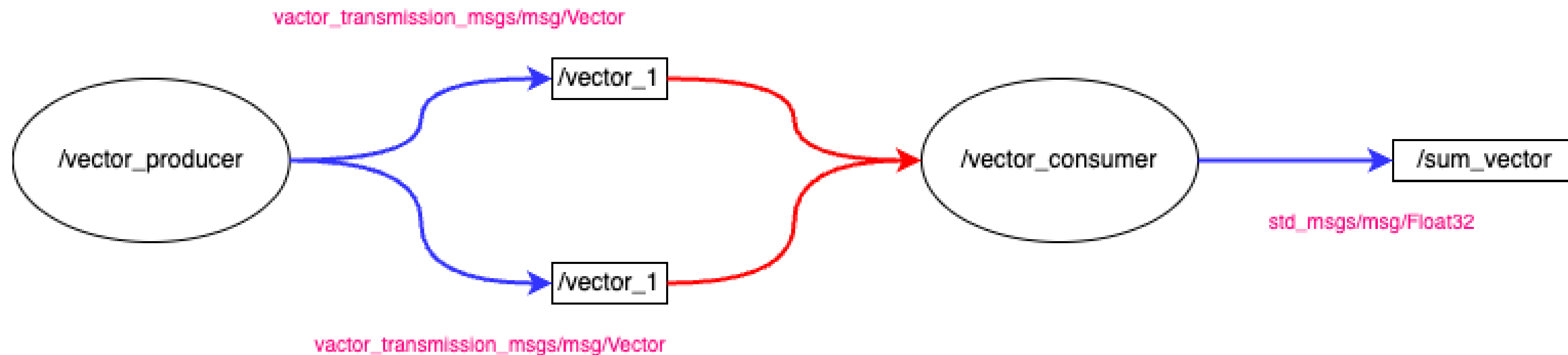
Day 2

3. Node Programming II



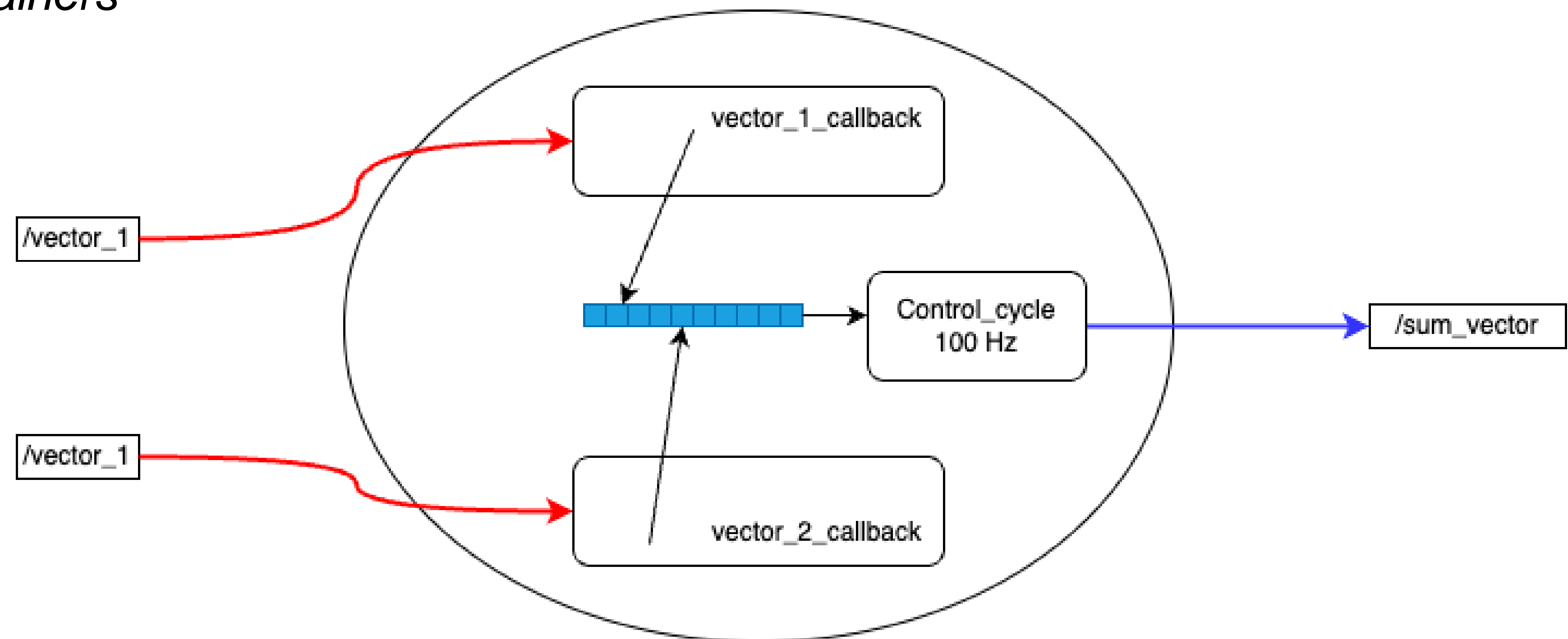
Vector Transmission

- *Executors and Multithreading*
- *Callback Groups*
- *Components and Containers*



Vector Transmission

- *Executors and Multithreading*
- *Callback Groups*
- *Components and Containers*



Vector Transmission

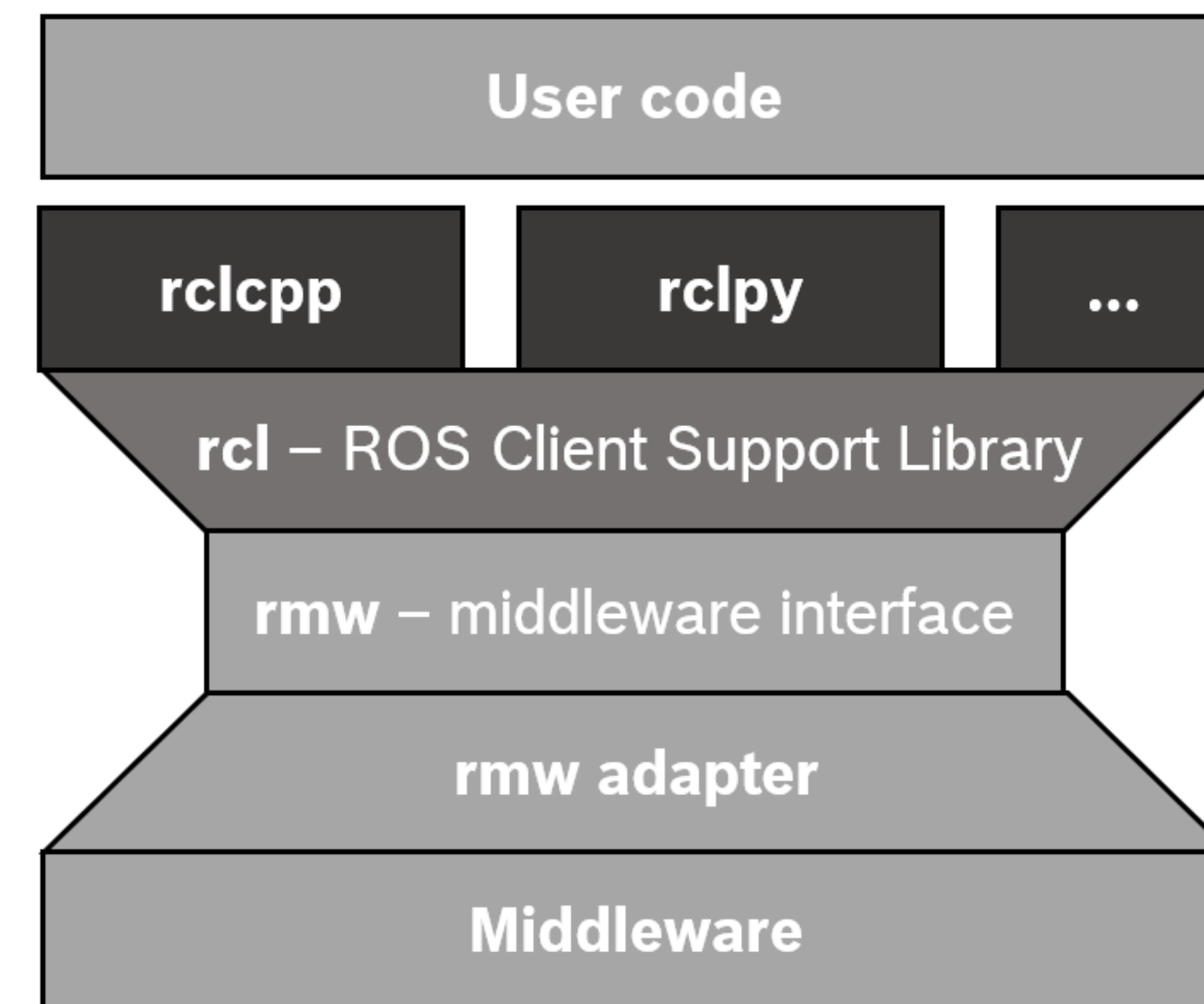
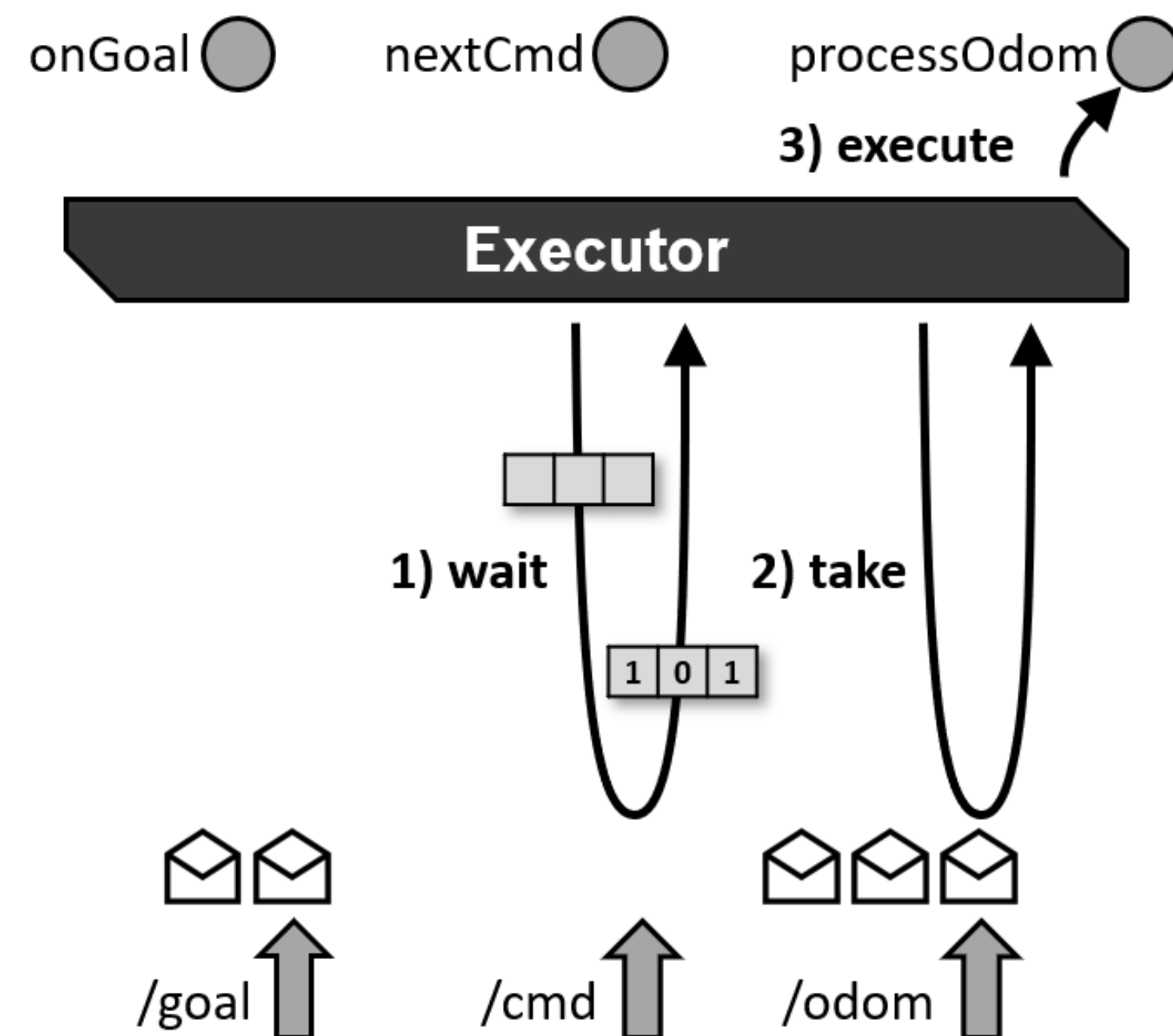
- *Executors and Multithreading*
- *Callback Groups*
- *Components and Containers*

An executor is a where add nodes and spin them all together

```
rcpp::Node::SharedPtr node1 = ...  
rcpp::Node::SharedPtr node2 = ...  
rcpp::Node::SharedPtr node3 = ...  
  
rcpp::executors::StaticSingleThreadedExecutor executor;  
executor.add_node(node1);  
executor.add_node(node2);  
executor.add_node(node3);  
executor.spin();
```

Vector Transmission

- **Executors and Multithreading**
- *Callback Groups*
- *Components and Containers*



Vector Transmission

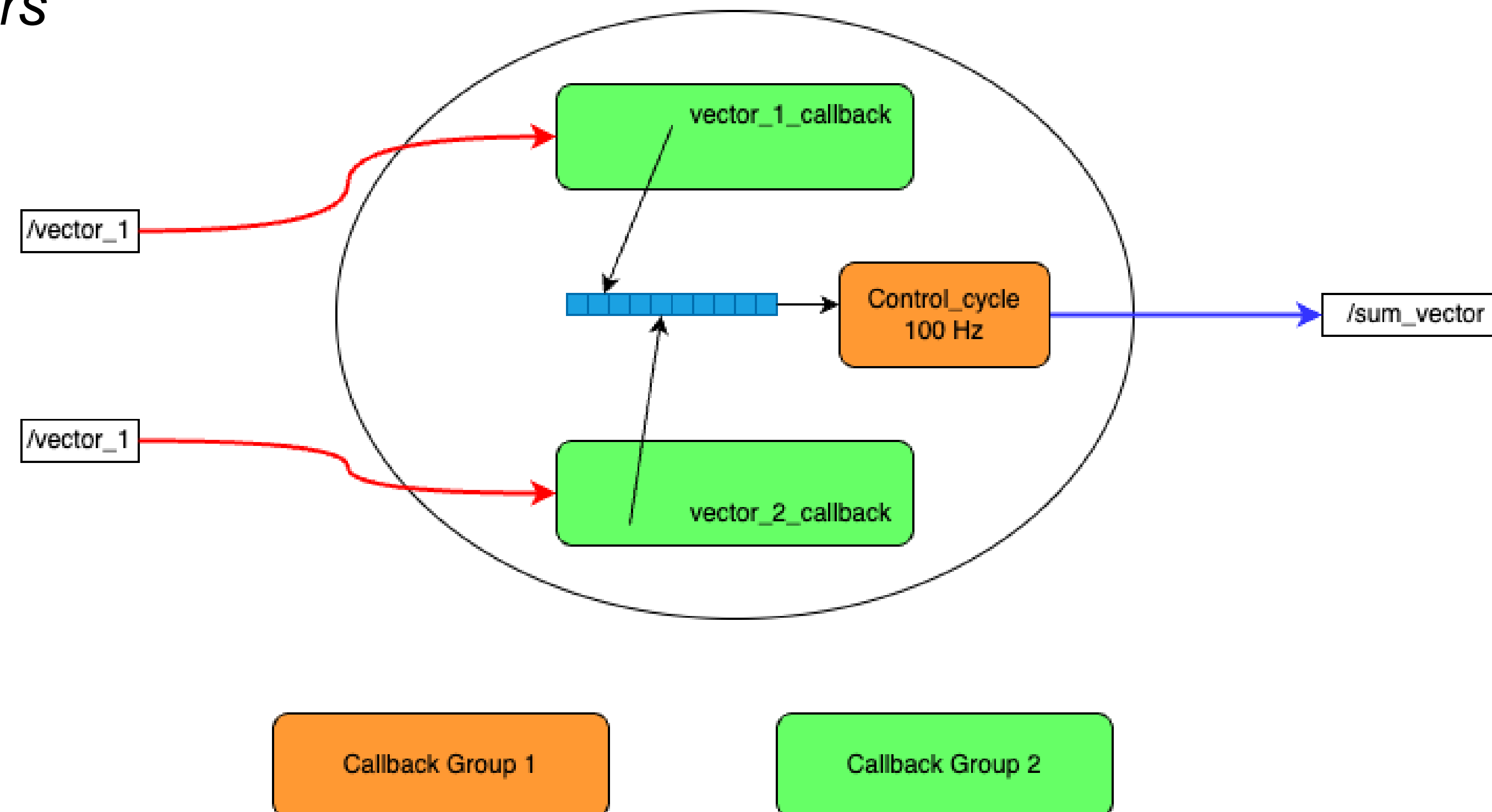
- ***Executors and Multithreading***
- *Callback Groups*
- *Components and Containers*

**Single-Threaded
Executors**

**Multi-Thread
Executor**

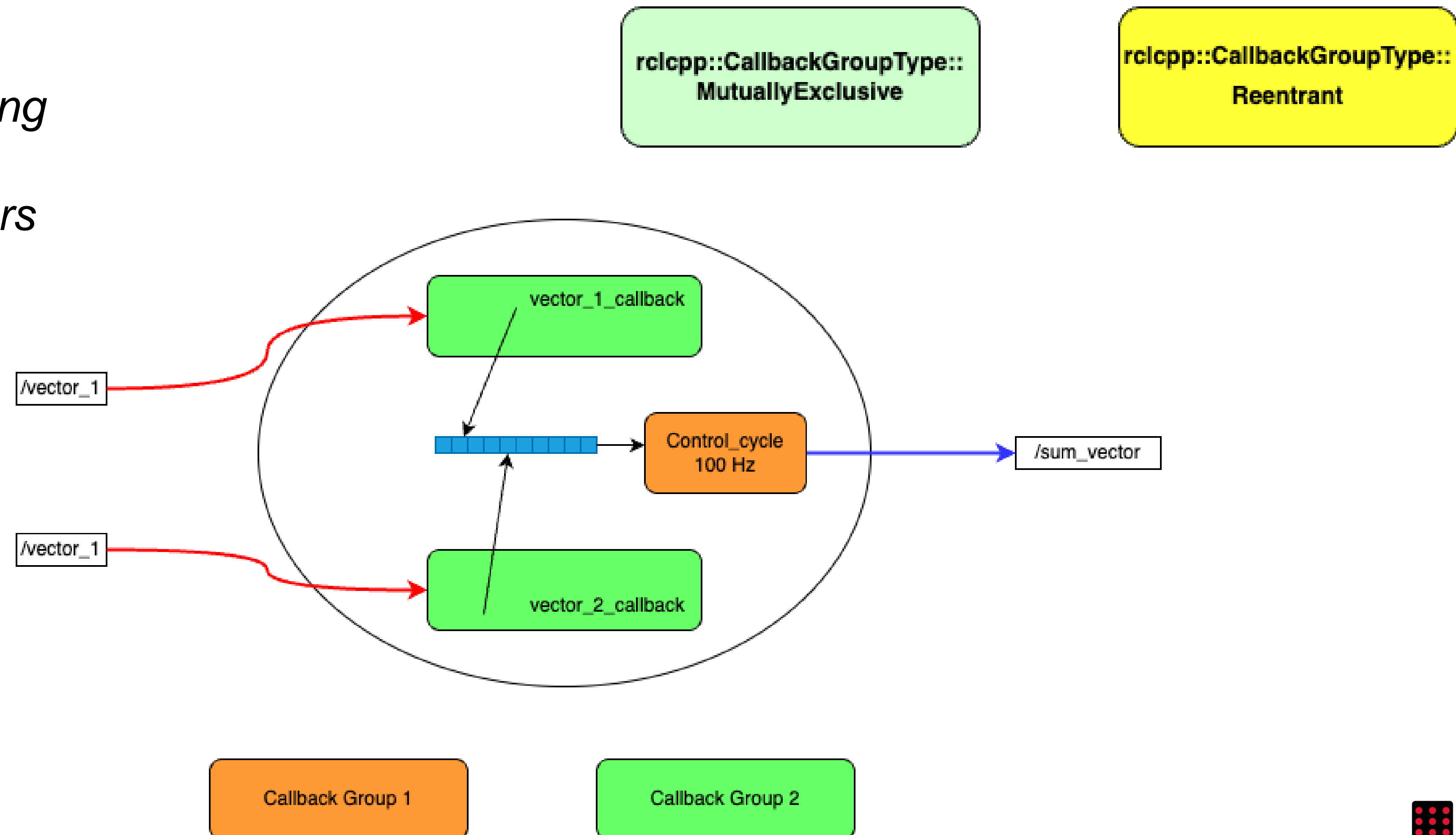
Vector Transmission

- *Executors and Multithreading*
- **Callback Groups**
- *Components and Containers*



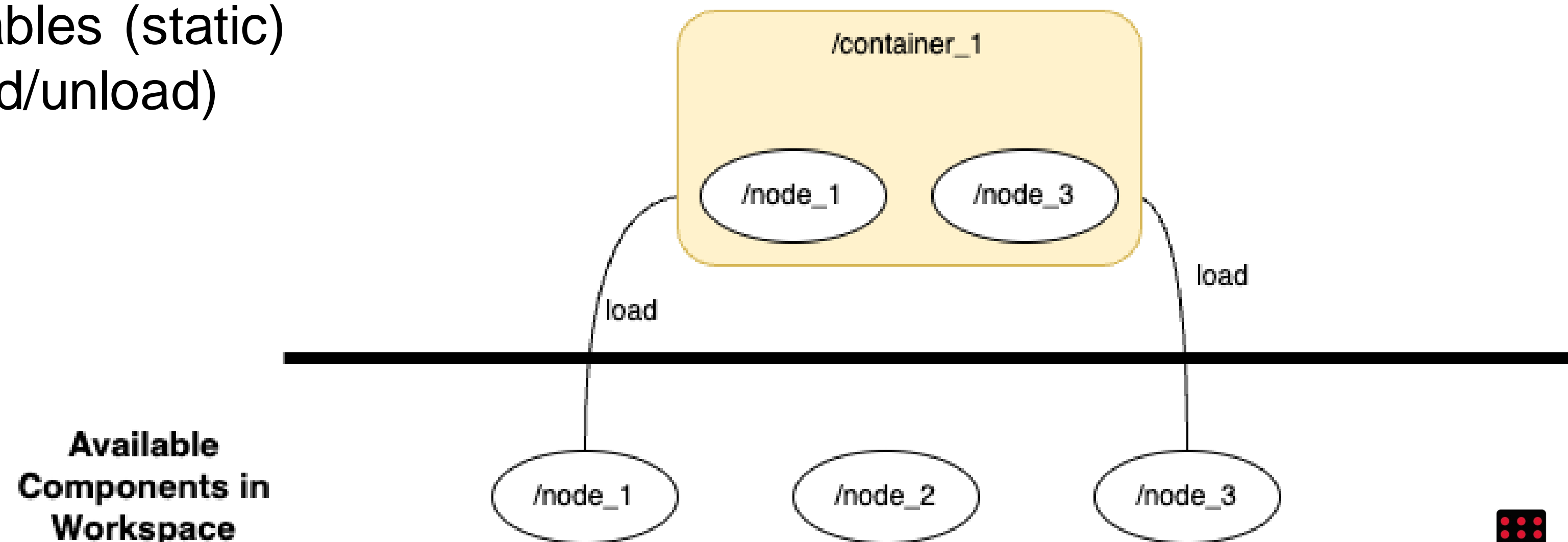
Vector Transmission

- *Executors and Multithreading*
- **Callback Groups**
- *Components and Containers*



Vector Transmission

- *Executors and Multithreading*
 - *Callback Groups*
 - ***Components and Containers***
-
- Nodes are usually compiled in libraries
 - Nodes are instantiated in executables (static)
 - ... or in containers (dynamic: load/unload)



Vector Transmission

- *Executors and Multithreading*
- *Callback Groups*
- ***Components and Containers***

Let's go to code!!!