



Реализация поддержки асинхронного программирования для фреймворка DSLab

Артём Макогон

20 мая 2023 г.

Введение

Описание предметной области

- Большой объем данных/вычислений
- Надежность
- Масштабируемость



Введение

Описание предметной области

- Большой объем данных/вычислений
 - Надежность
 - Масштабируемость
- ⇓
- > Недетерминированные алгоритмы
 - > Сложное тестирование





Введение

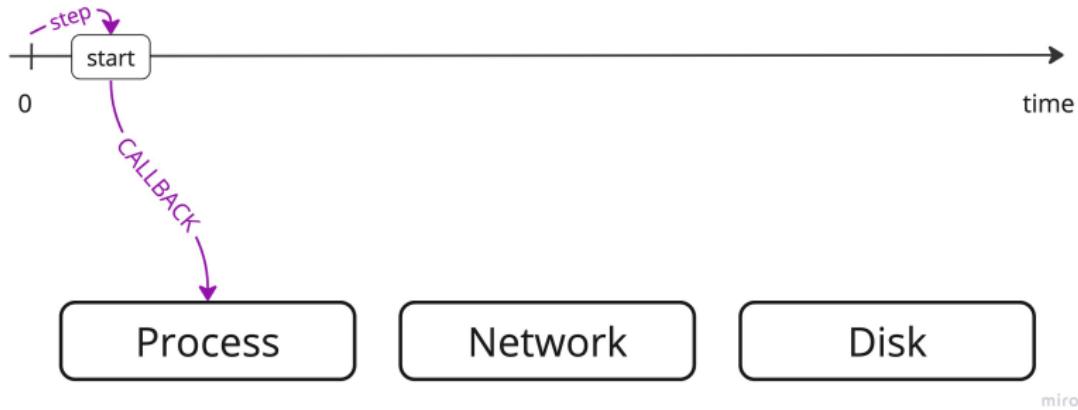
Дискретно-событийное моделирование



Исполнение симуляции

Введение

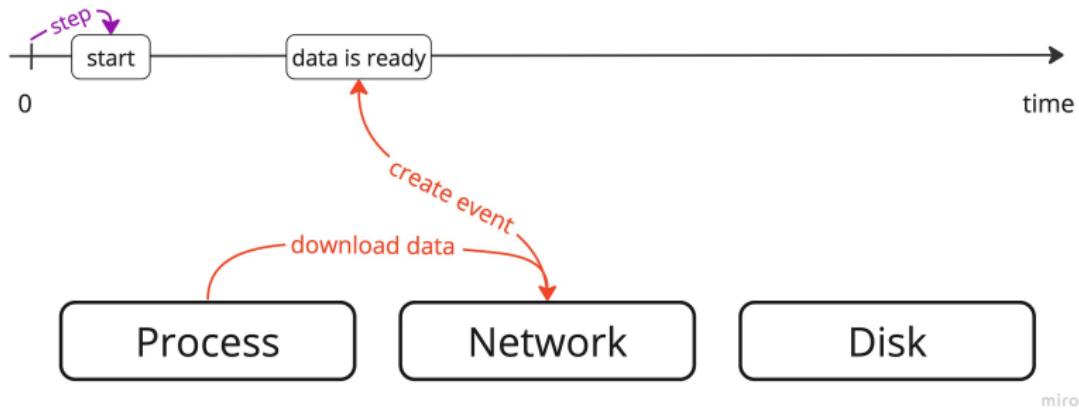
Дискретно-событийное моделирование



Исполнение симуляции

Введение

Дискретно-событийное моделирование

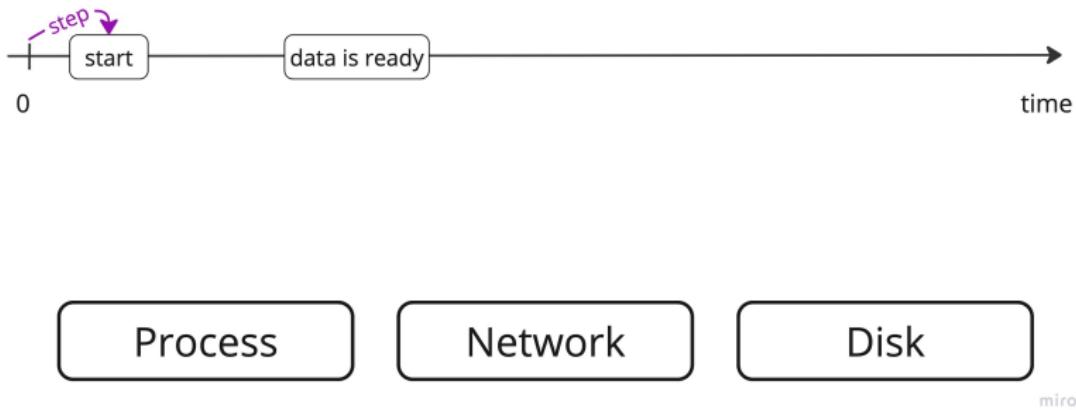


Исполнение симуляции



Введение

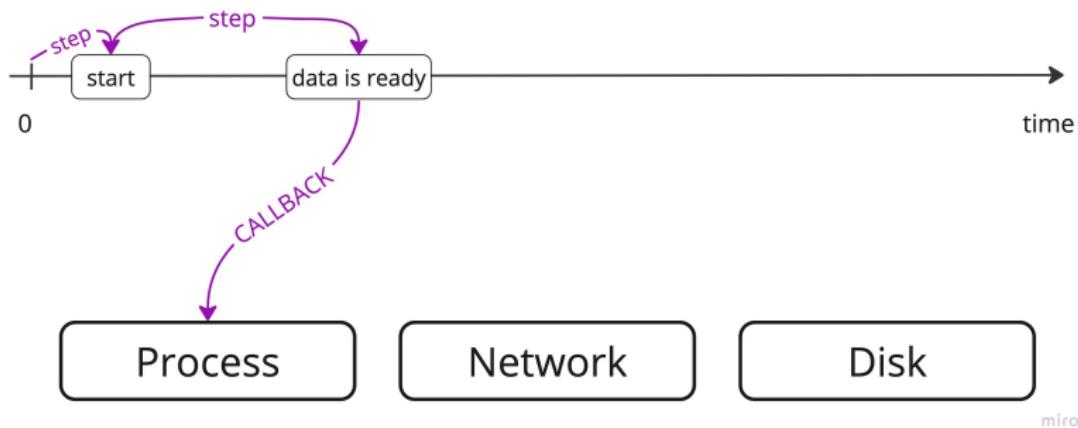
Дискретно-событийное моделирование



Исполнение симуляции

Введение

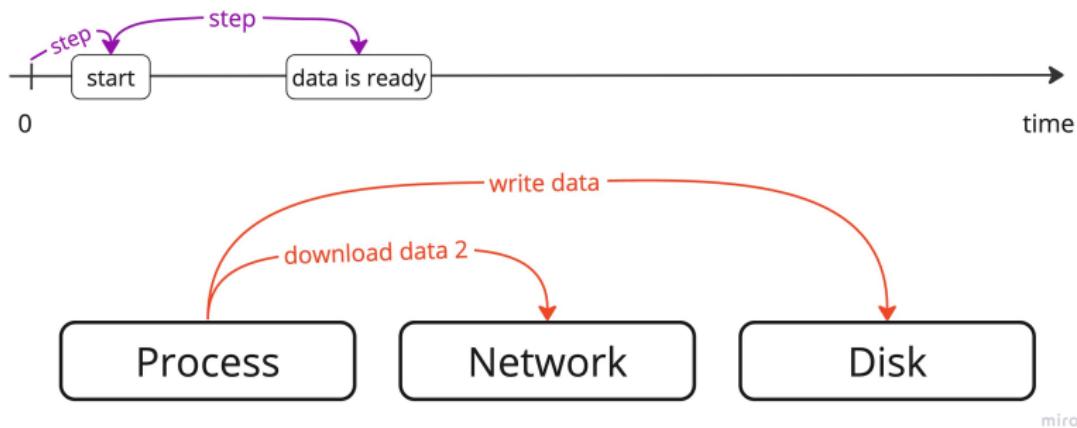
Дискретно-событийное моделирование



Исполнение симуляции

Введение

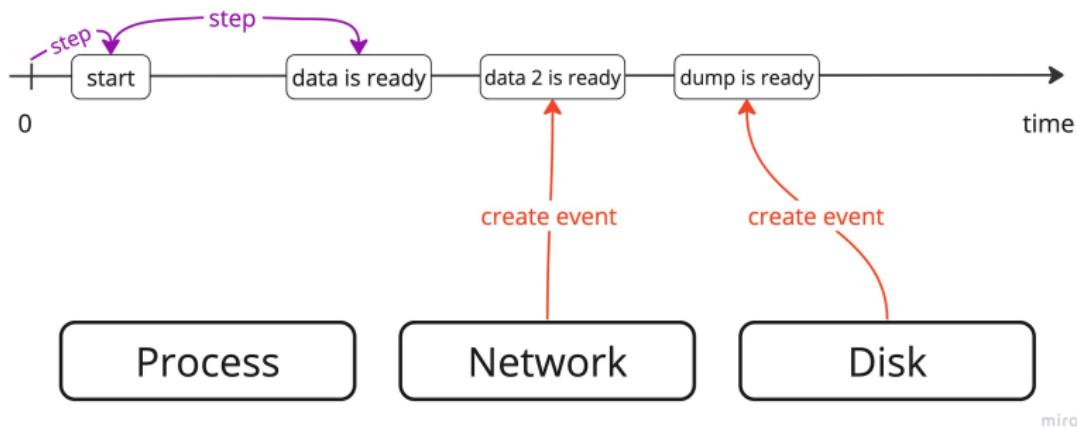
Дискретно-событийное моделирование



Исполнение симуляции

Введение

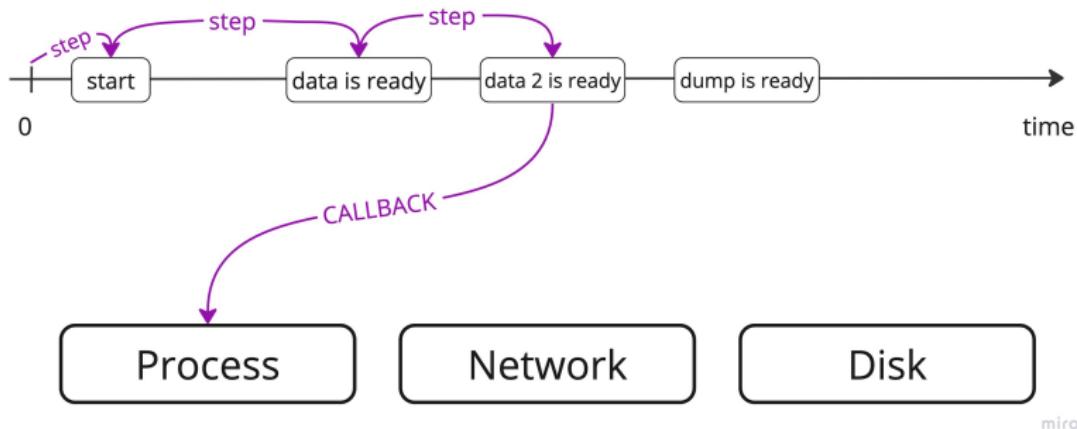
Дискретно-событийное моделирование



Исполнение симуляции

Введение

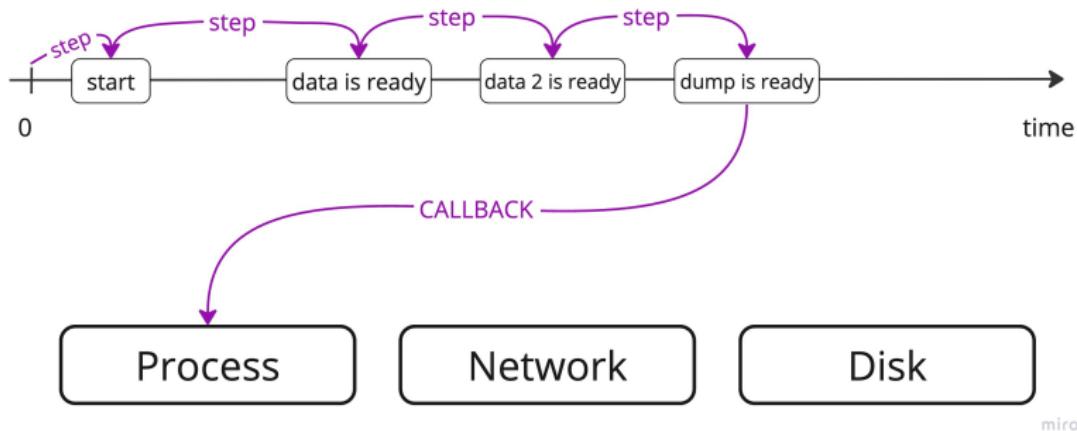
Дискретно-событийное моделирование



Исполнение симуляции

Введение

Дискретно-событийное моделирование

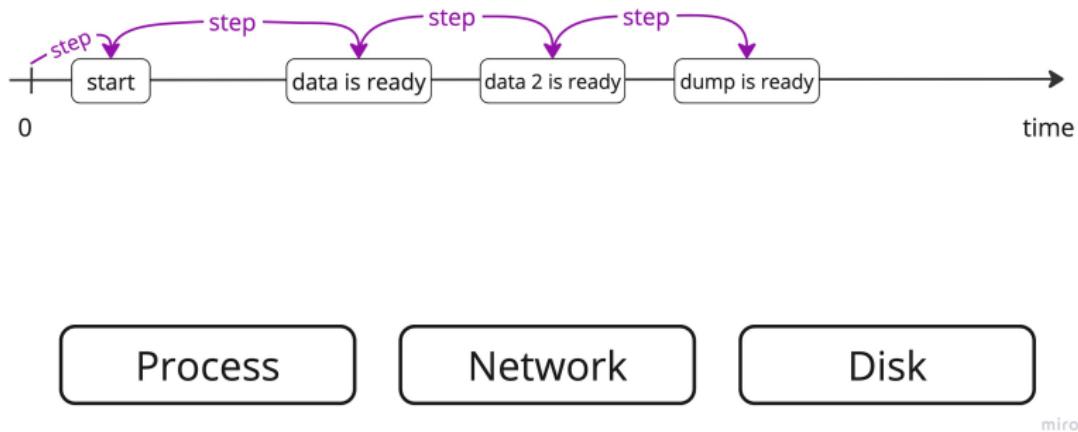


Исполнение симуляции



Введение

Дискретно-событийное моделирование

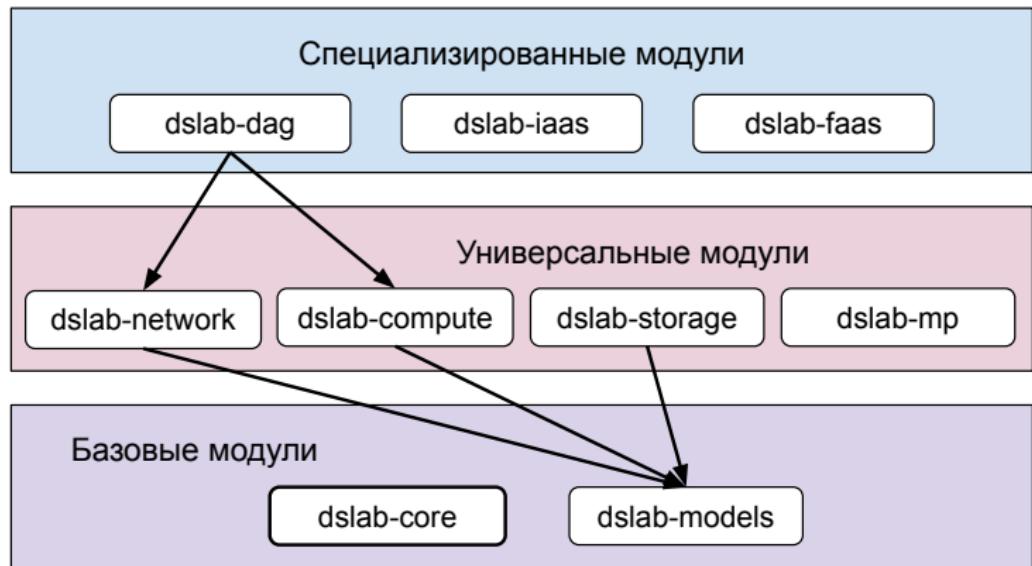


Исполнение симуляции



Введение

Архитектура проекта DSLab





Введение

Callback модель. Реализация Process.

```
fn on(&mut self, event: Event) {  
    cast!(match event.data {  
        Start {} => {  
            self.on_start();   
        }  
        DownloadCompleted {data} => {  
            self.on_download_completed(data);   
        }  
        DiskWriteCompleted => {  
            self.on_disk_write_completed();   
        }  
    })  
}
```

Реагирование на события.

```
impl Process {  
    fn /**/ {  
          
    }  
    fn /**/ {  
          
    }  
    fn /**/ {  
          
    }  
}
```

Реализация
callback-ов



Введение

Преимущества асинхронного подхода

```
async fn add_file_to_storage(some_file) {
    send_file_to_all_replicas(some_file);
    result = wait_confirmation_from_all().await;

    if result.has_quorum {
        send_commit_to_replicas(result.nodes);
        wait_commit_confirmation_from(result.nodes).await;

        send_ok_message_to_user();
    } else {
        send_reject_message_to_user();
    }
}
```

Псевдокод асинхронного взаимодействия нод в симуляции



Введение

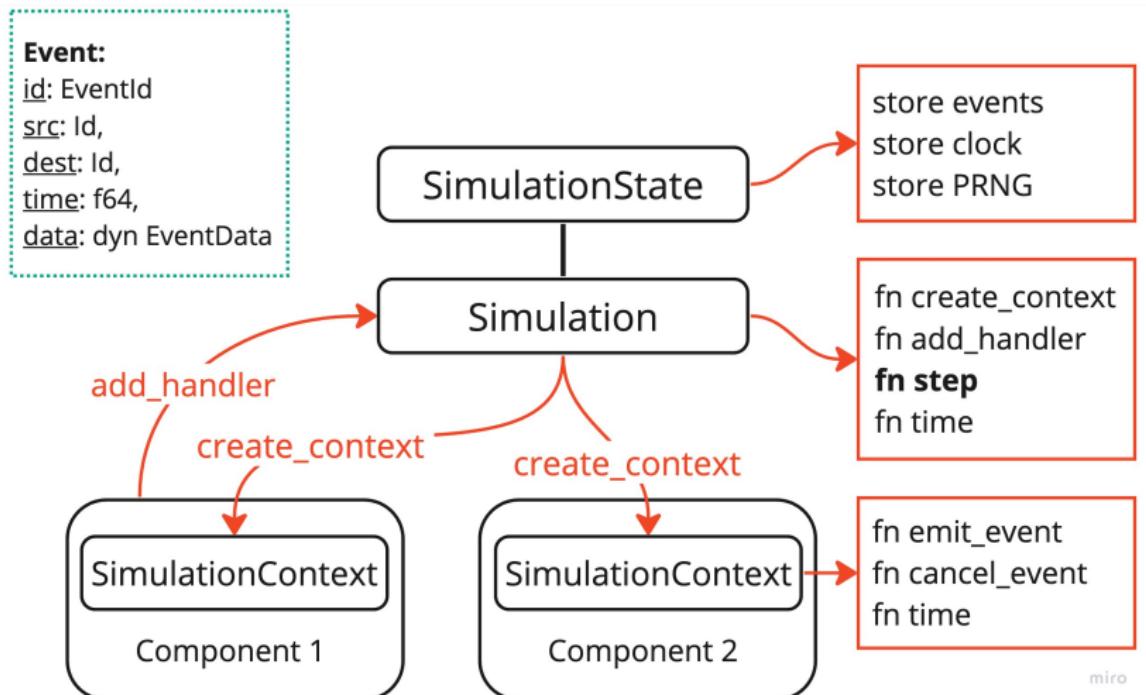
Постановка задачи

Цель проекта – реализация поддержки асинхронного программирования для фреймворка DSLab. Для этого необходимо:

- Реализовать асинхронное расширение для существующего ядра `dslab-core`.
- Добавить примеры использования нового функционала высокоДуровневыми компонентами.
- Написать документацию нового API и покрыть реализацию тестами.

Введение

Дизайн и структура ядра dslab-core



Внутреннее устройство dslab-core

Управление исполнением



```
fn on_start_action(&self) {  
    // do 1  
}  
  
fn on_first_event(&self) {  
    // do 2  
}  
  
fn on_second_event(&self) {  
    // do 3  
}
```

Синхронный код (разрезан на
части разработчиком)

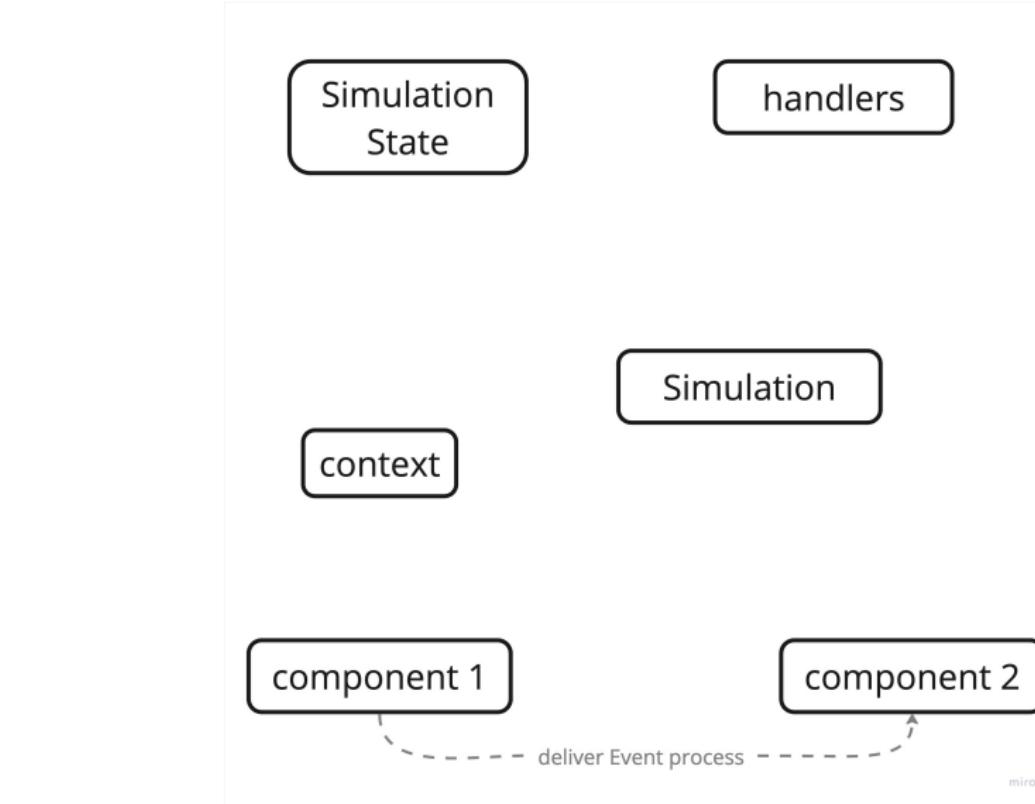


```
async fn action(&self) {  
    // do 1  
  
    wait_for_first_event().await;  
  
    // do 2  
  
    wait_for_second_event().await;  
  
    // do 3  
}
```

Асинхронный код (разрезан на части
компилятором)

Внутреннее устройство и реализация

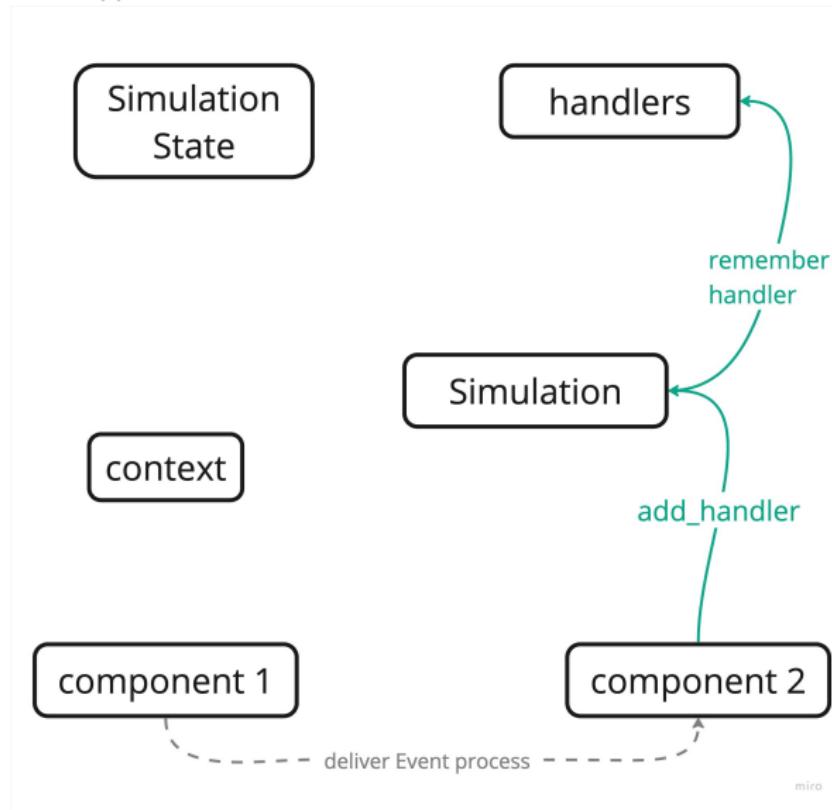
Callback-based модель



miro

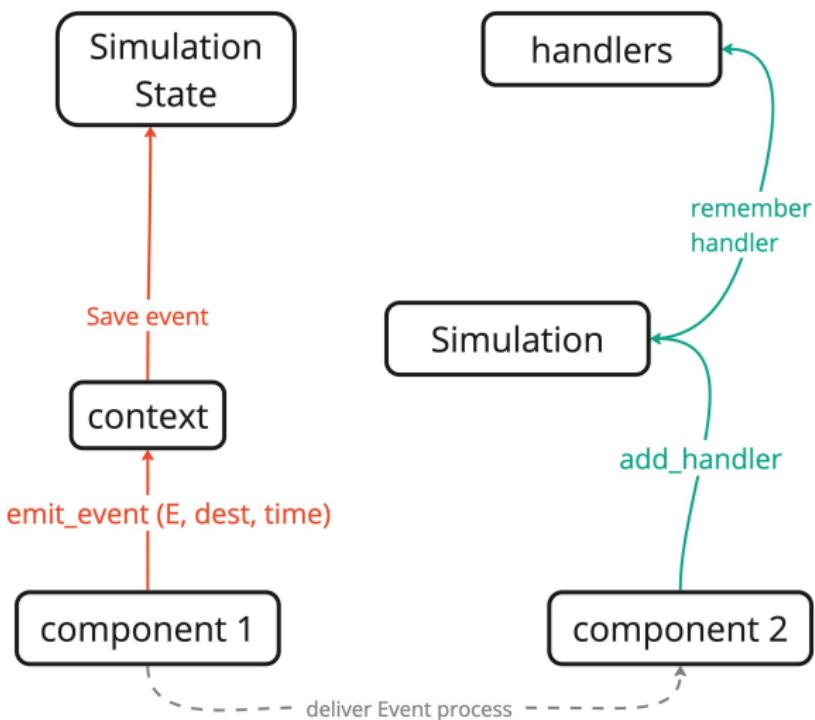
Внутреннее устройство и реализация

Callback-based модель



Внутреннее устройство и реализация

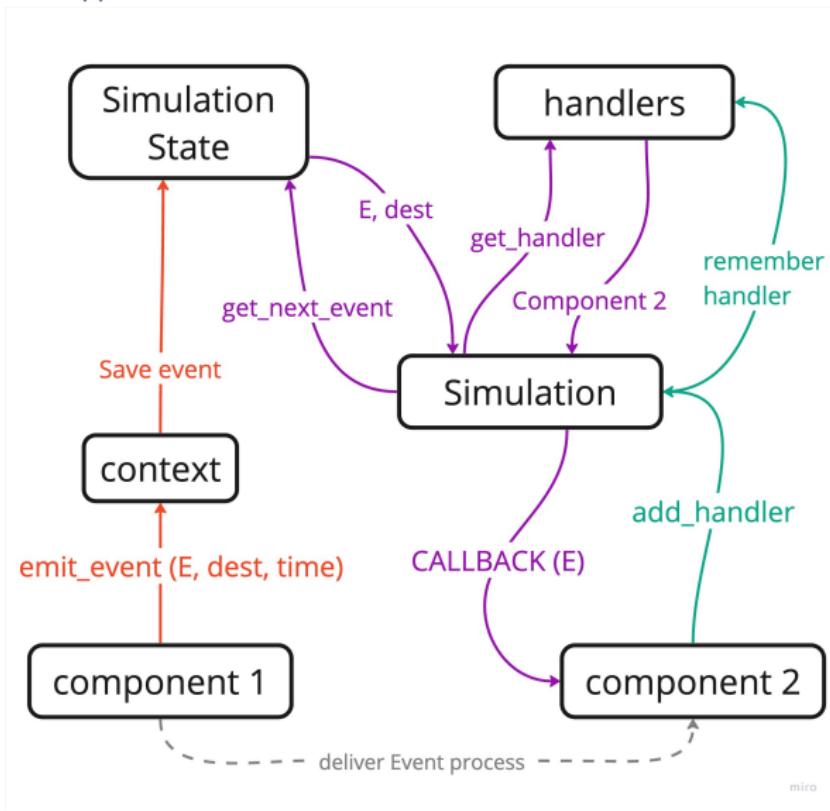
Callback-based модель



miro

Внутреннее устройство и реализация

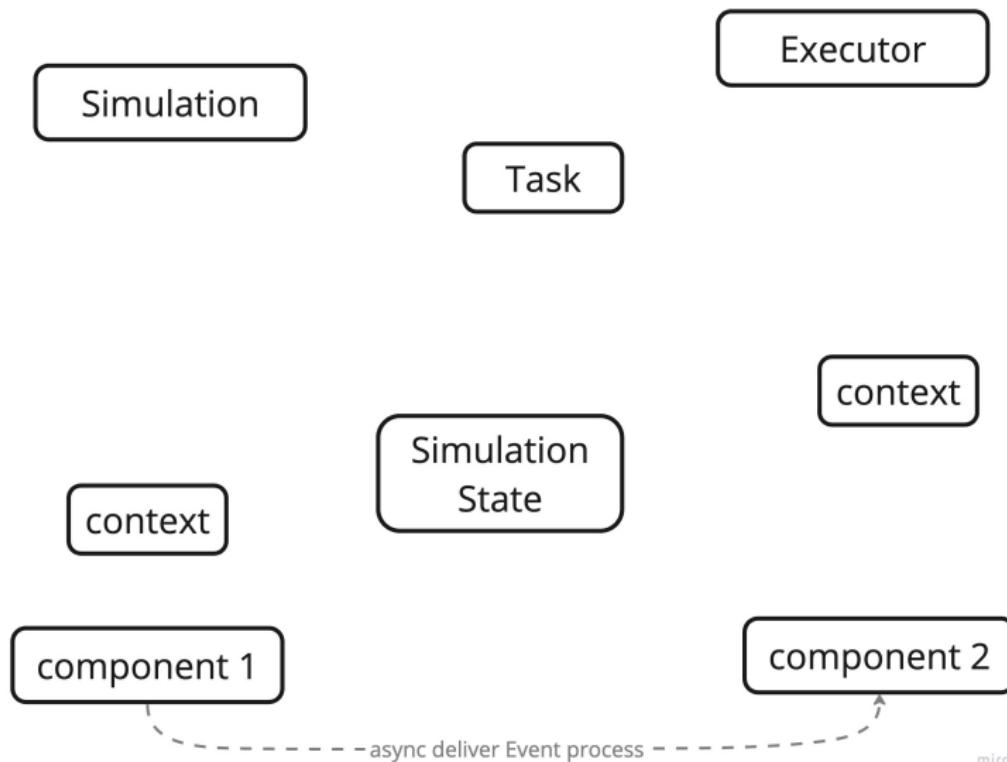
Callback-based модель





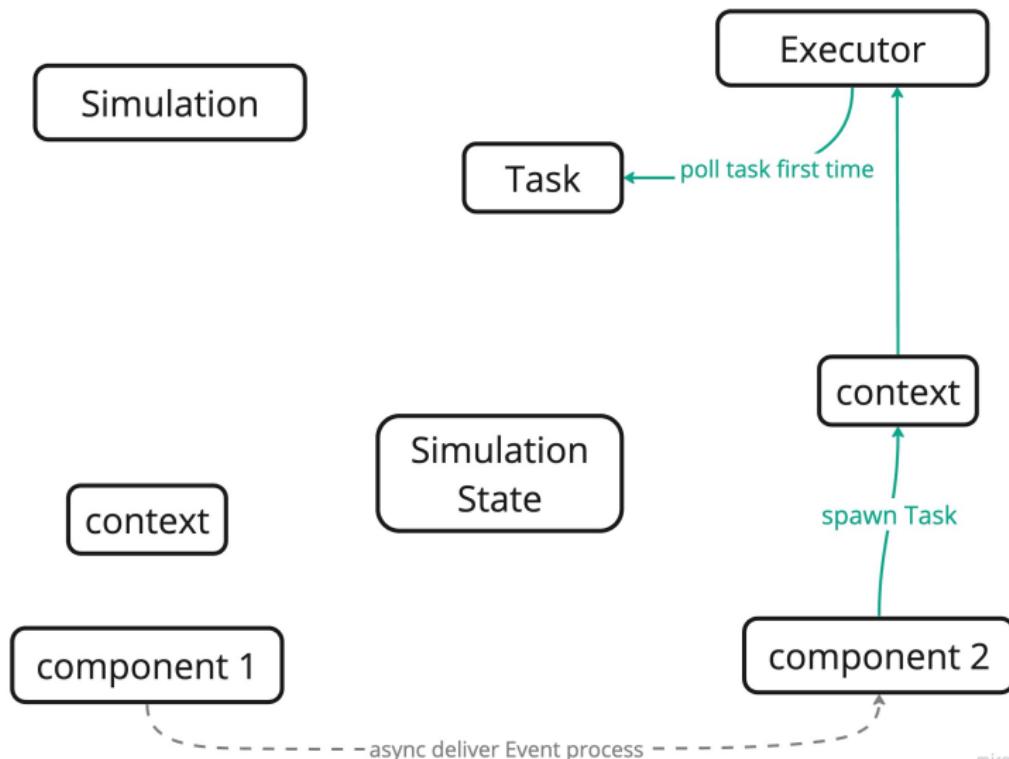
Внутреннее устройство и реализация

Асинхронная модель



Внутреннее устройство и реализация

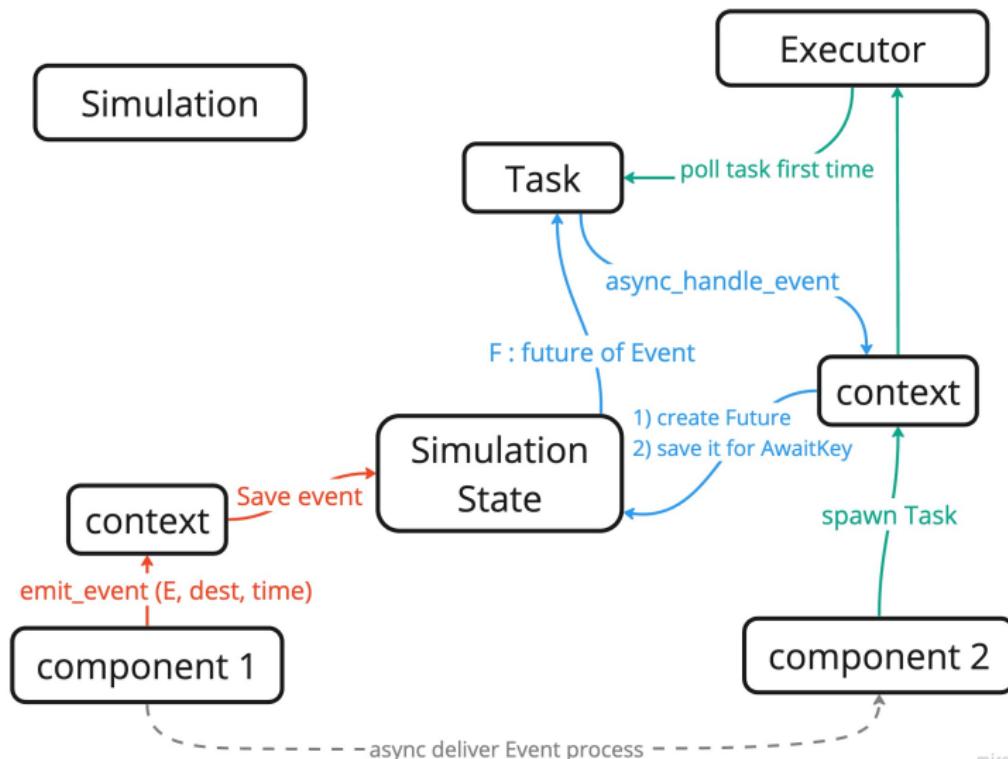
Асинхронная модель



miro

Внутреннее устройство и реализация

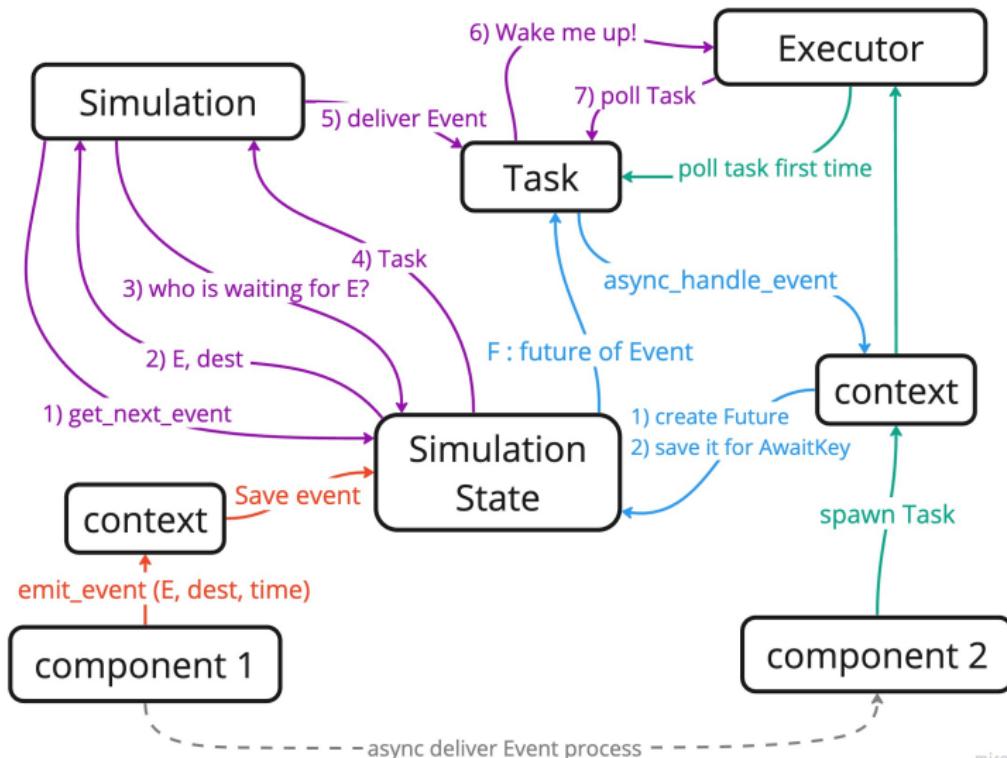
Асинхронная модель



miro

Внутреннее устройство и реализация

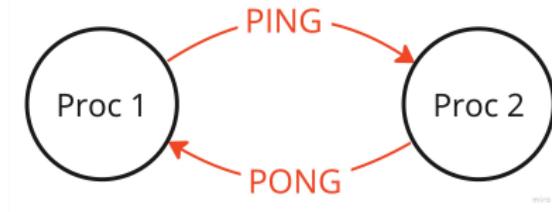
Асинхронная модель



Тестирование и замеры производительности

Ping-pong

Arg	Value
Hosts	100000
Peers per host	100
Iterations	100



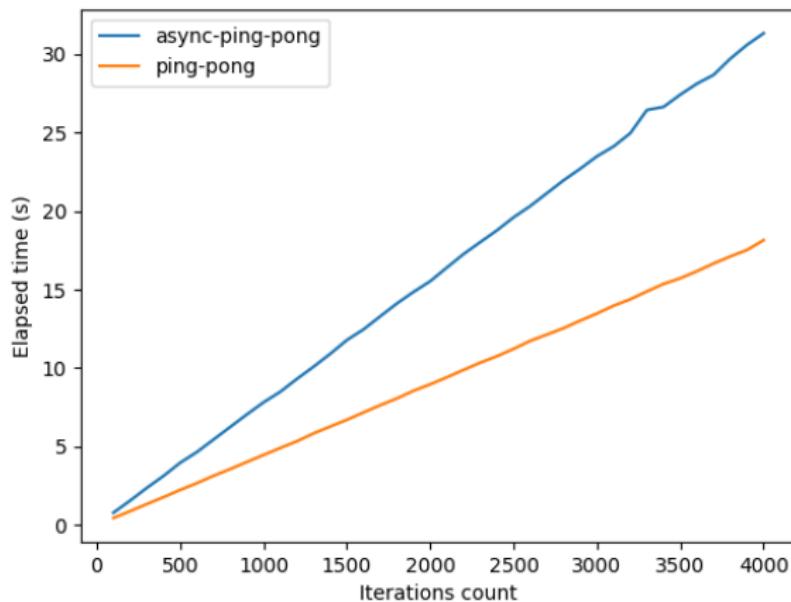
Аргументы

Example	Elapsed time	Events/s	Iterations/s
async-ping-pong	16.45s	1234103	6.08
ping-pong	8.20s	2452670	12.20

Сравнение производительности

Тестирование и замеры производительности

Ping-pong



Сравнение производительности

Тестирование и замеры производительности



Master-workers

