

Распределённая система
near-realtime агрегации
больших объёмов данных на
основе cocaine

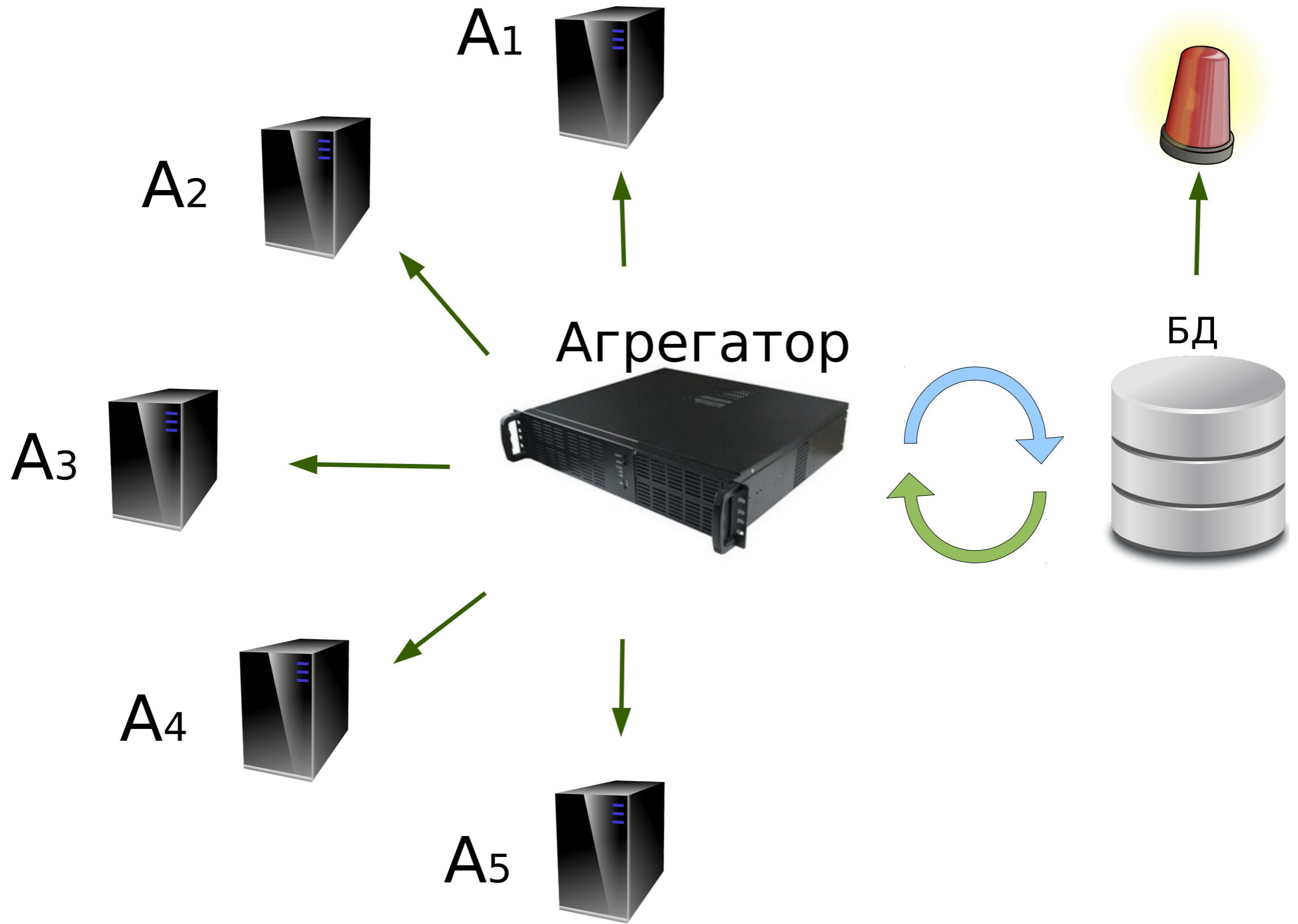
Антон Тюрин

Системный администратор

Я.Субботник в Санкт-Петербурге, 1 декабря 2012

Задача

Как быстро обработать большой
объем данных?



“Простое” решение

I'm working now



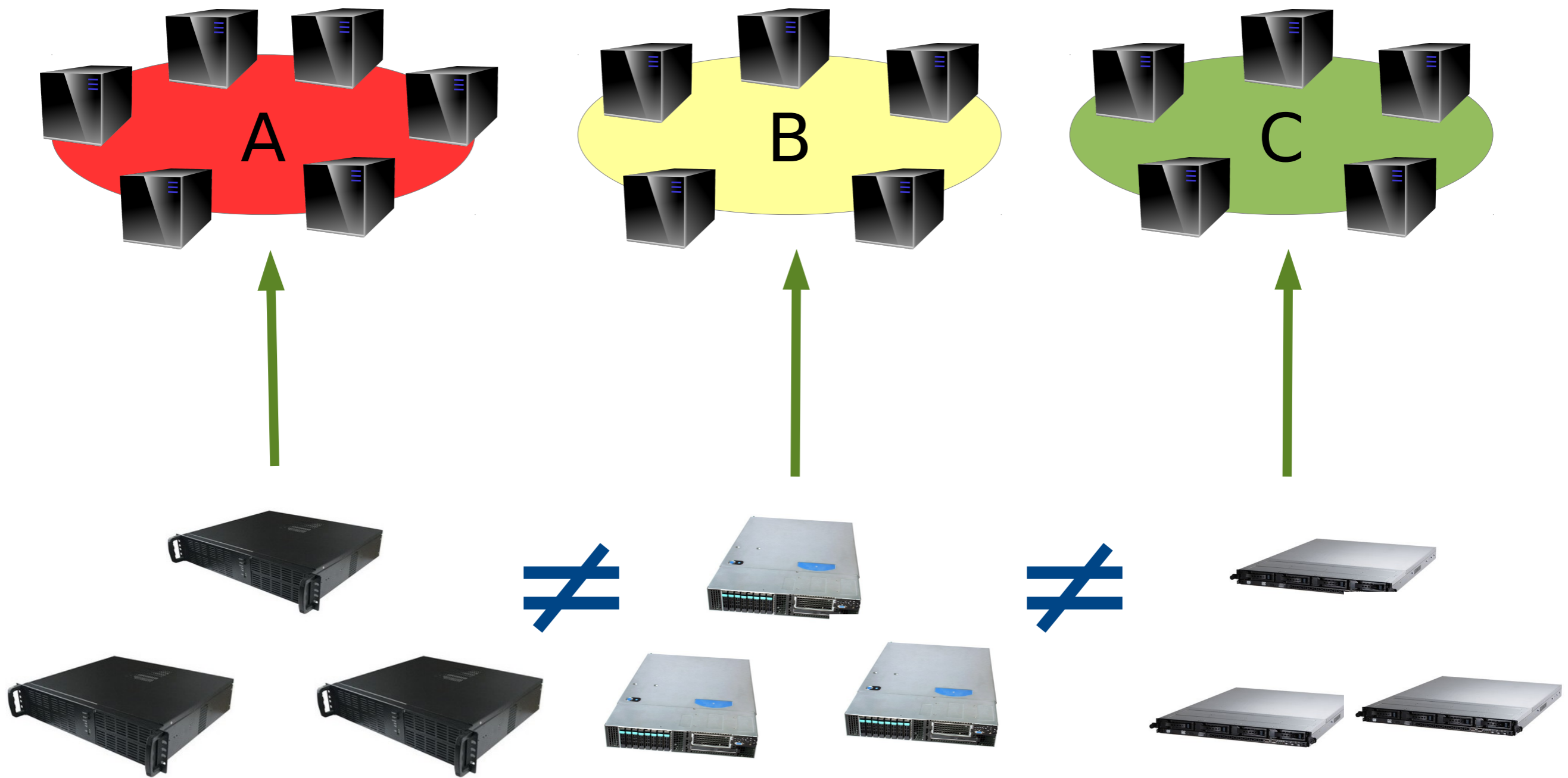
zZZ



zZZ



Обеспечим отказоустойчивость



Каждому кластеру по агрегатору



Данных становится все больше и больше...

Недостатки:

- Немасштабируемо
- Тяжело администрировать, как сервис
- Неоптимальный расход оборудования

Другой путь – локальный парсинг

А почему бы не отдавать данные в обработанном виде?

- Нецелевая нагрузка на боевой сервер
- Придется раскладывать N парсеров в M групп

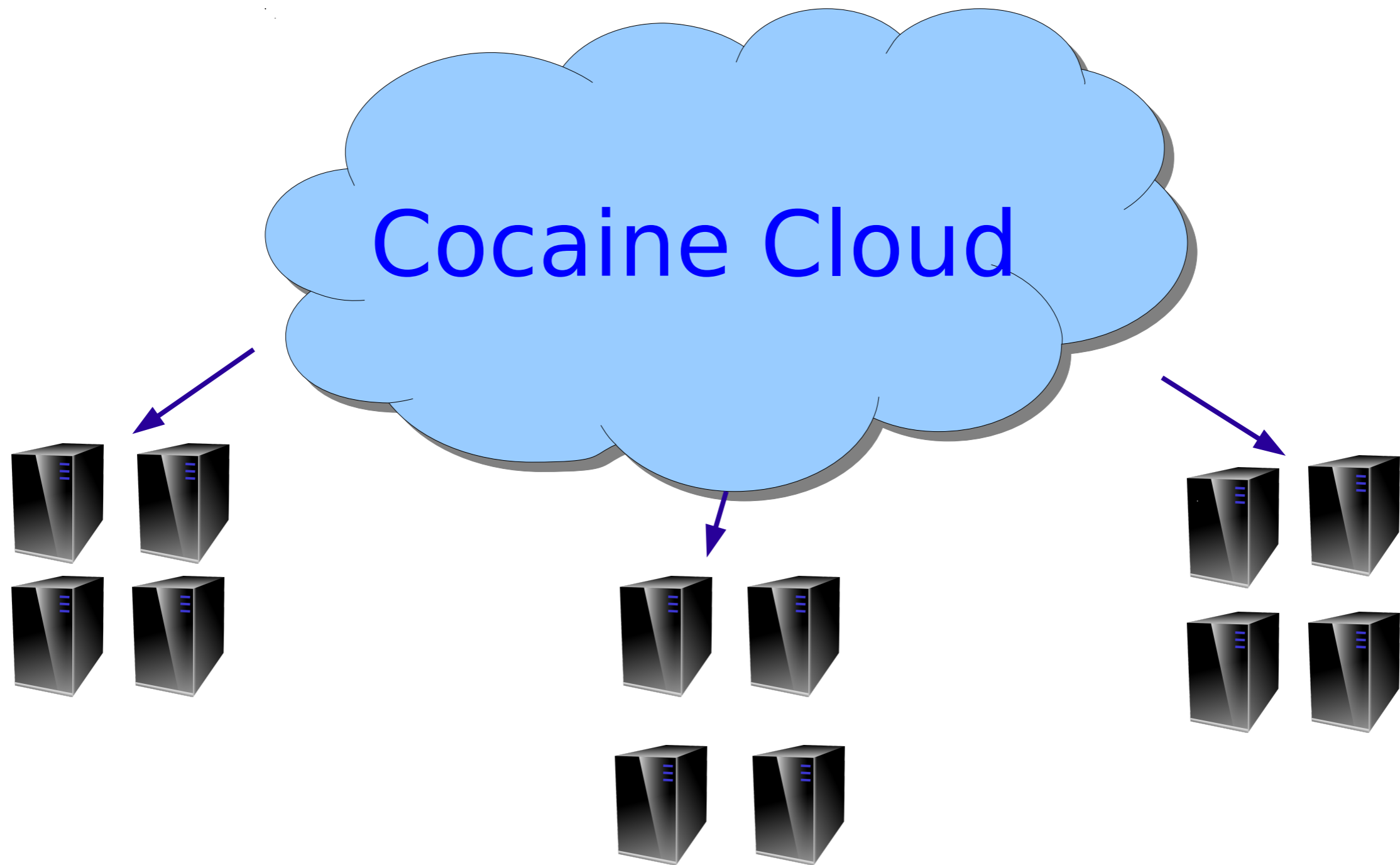
Combaine



1	0	1	0	1
0	1	0	1	0
1	0	1	0	1
0	1	0	1	0
1	0	1	0	1
0	1	0	1	0



В облаке все машины равноправны



Все обслуживаются одним облаком

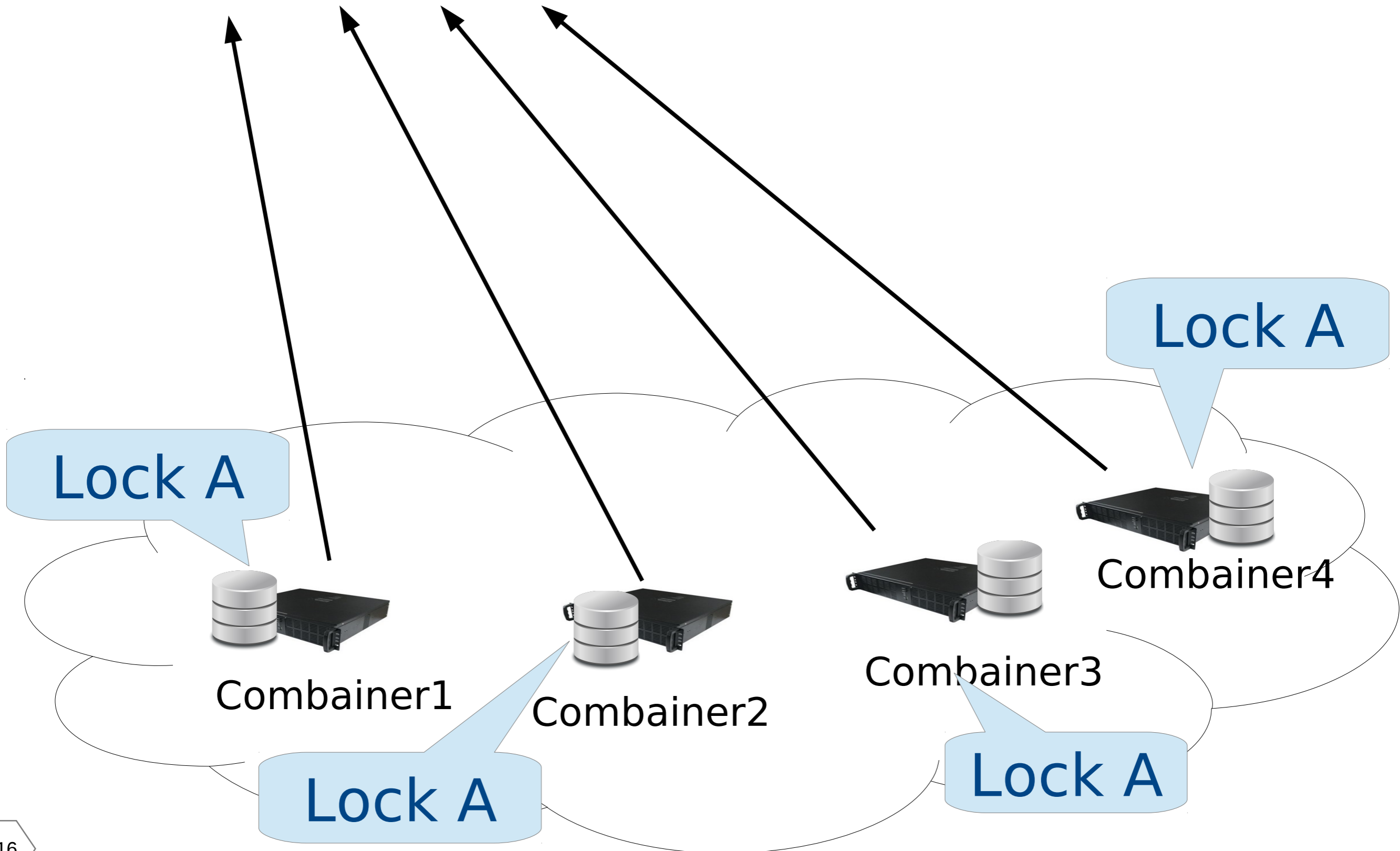
```
{'apps': {
  'aggregate@1': {'drivers': {'aggregate_group':
{'endpoint': 'tcp://host.dev.yandex.net:9505',
  'route': 'host.dev.yandex.net/aggregate@1/aggregate_group',
  'type': 'native-server'}}},
  'queue-depth': 0,
  'slaves': {'busy': 0, 'total': 2},
  'state': 'running'},
  'combainer@1': {'drivers': {'cloudMain': {'interval': 10.0,
  'type': 'recurring-timer'}}},
  'queue-depth': 0,
  'slaves': {'busy': 0, 'total': 1},
  'state': 'running'},
  'parsing@1': {'drivers': {'parsing':
{'endpoint': 'tcp://host.dev.yandex.net:9500',
  'route': 'host.dev.yandex.net/parsing@1/parsing',
  'type': 'native-server'}}},
  'queue-depth': 0,
  'slaves': {'busy': 4, 'total': 9},
  'state': 'running'},
  'jobs': {'pending': 4, 'processed': 4370516},
  'uptime': 950696.05875563622}
```

Combiner — процесс,
руководящий обработкой
данных.

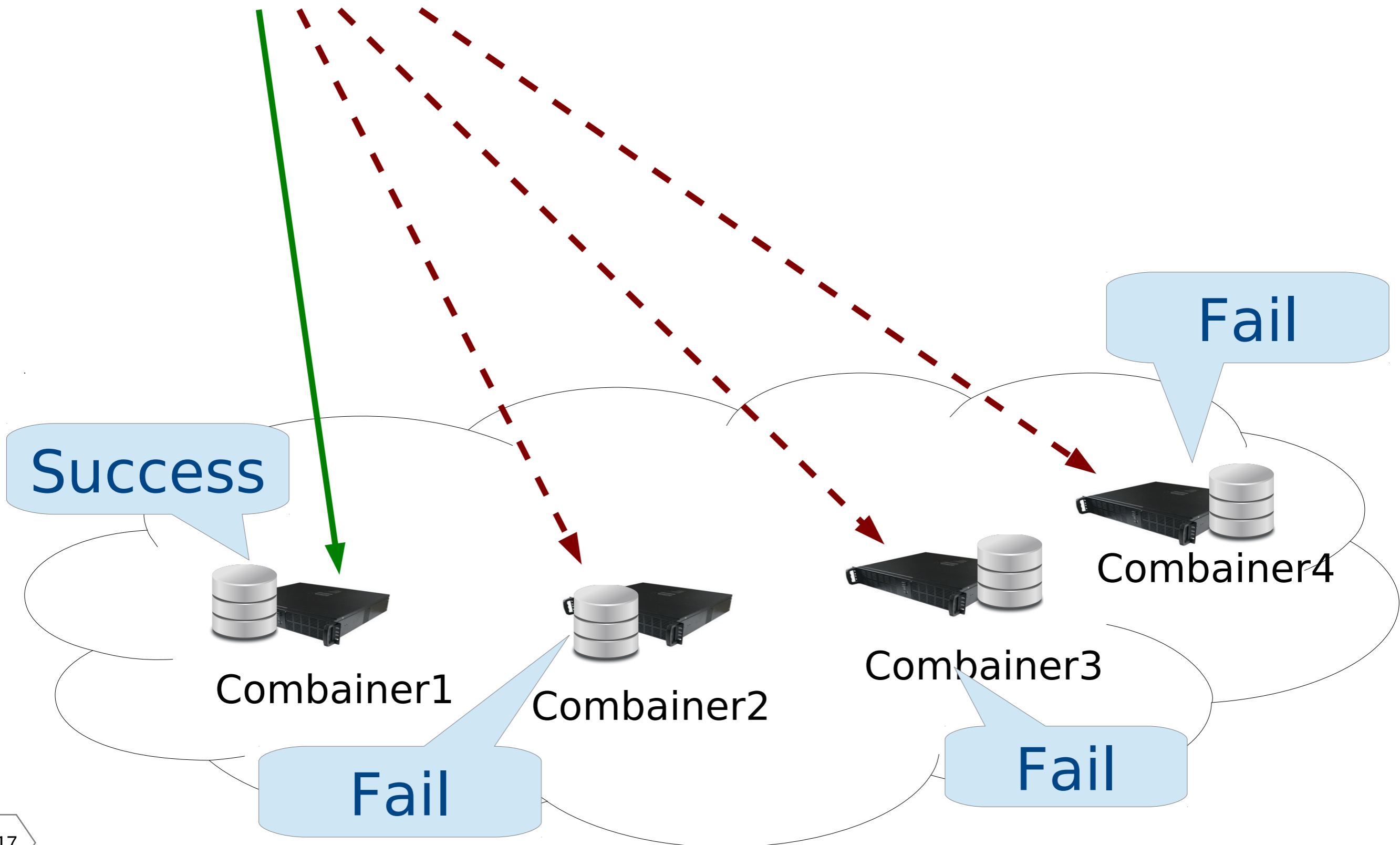
Стадии обработки данных

- Парсинг
- Агрегация по хосту
- Агрегация по группе

Zookeeper



Zookeeper



Zookeeper

I have lock A

Lock B



Combiner1



Combiner2



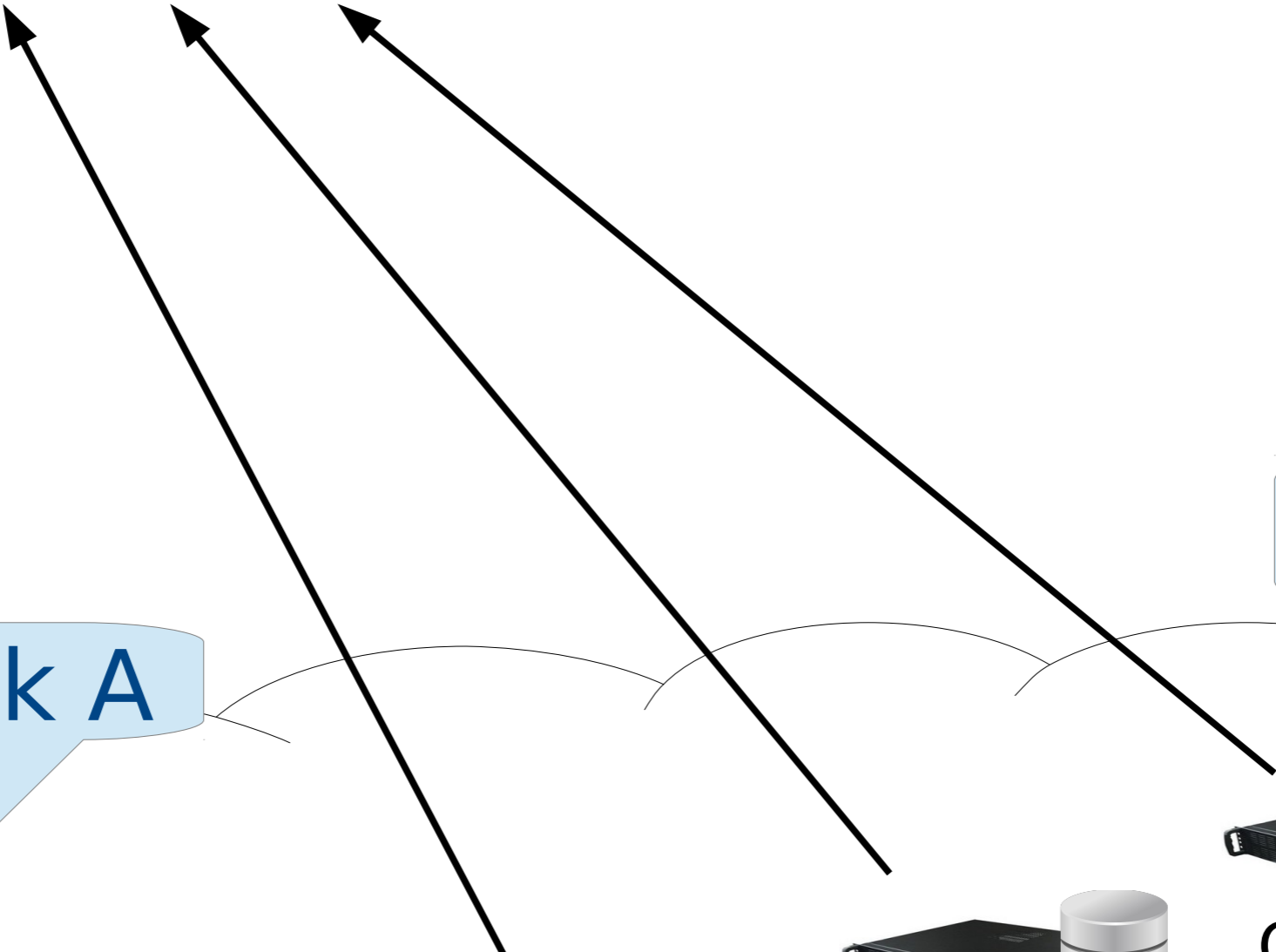
Combiner3



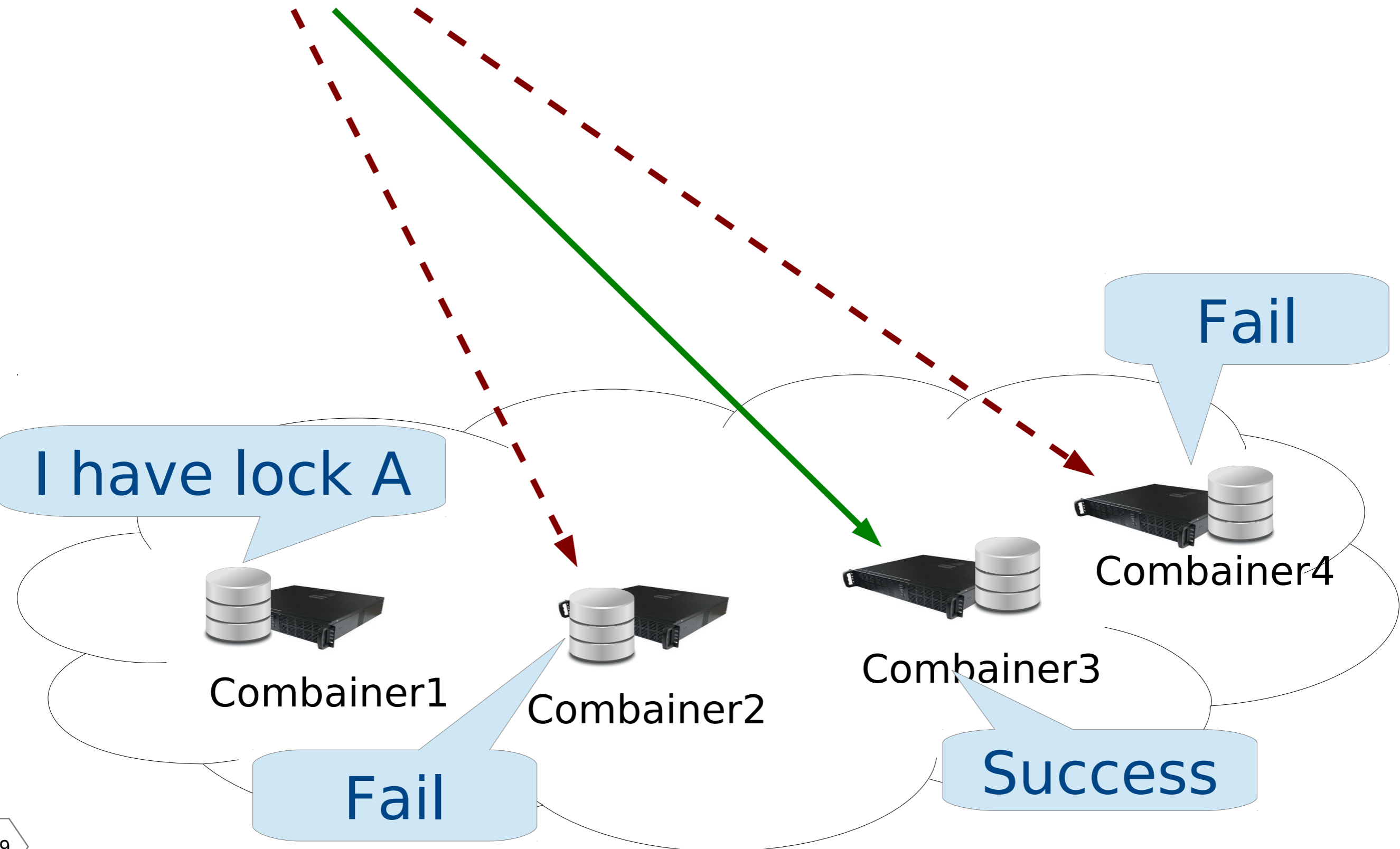
Combiner4

Lock B

Lock B



Zookeeper



I have lock A

Fail

Fail

Success



Combainer1



Combainer2

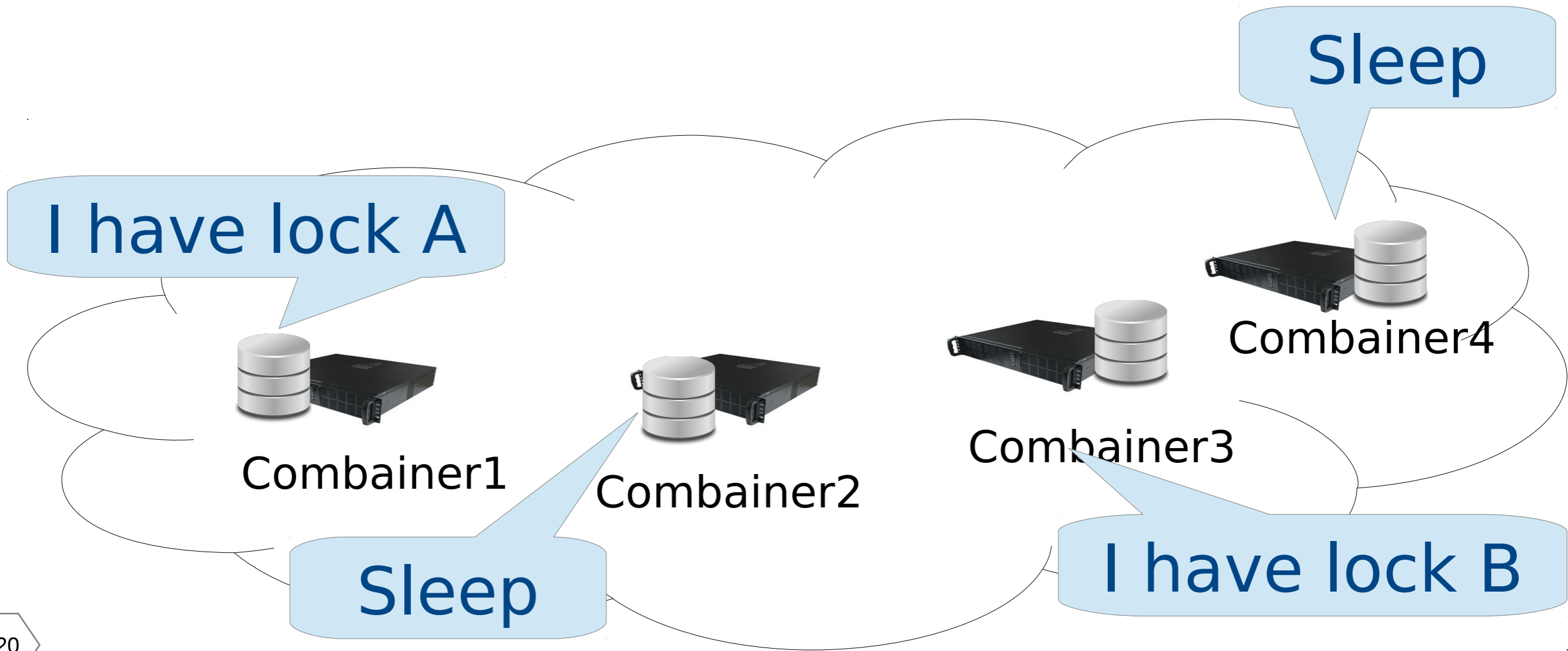


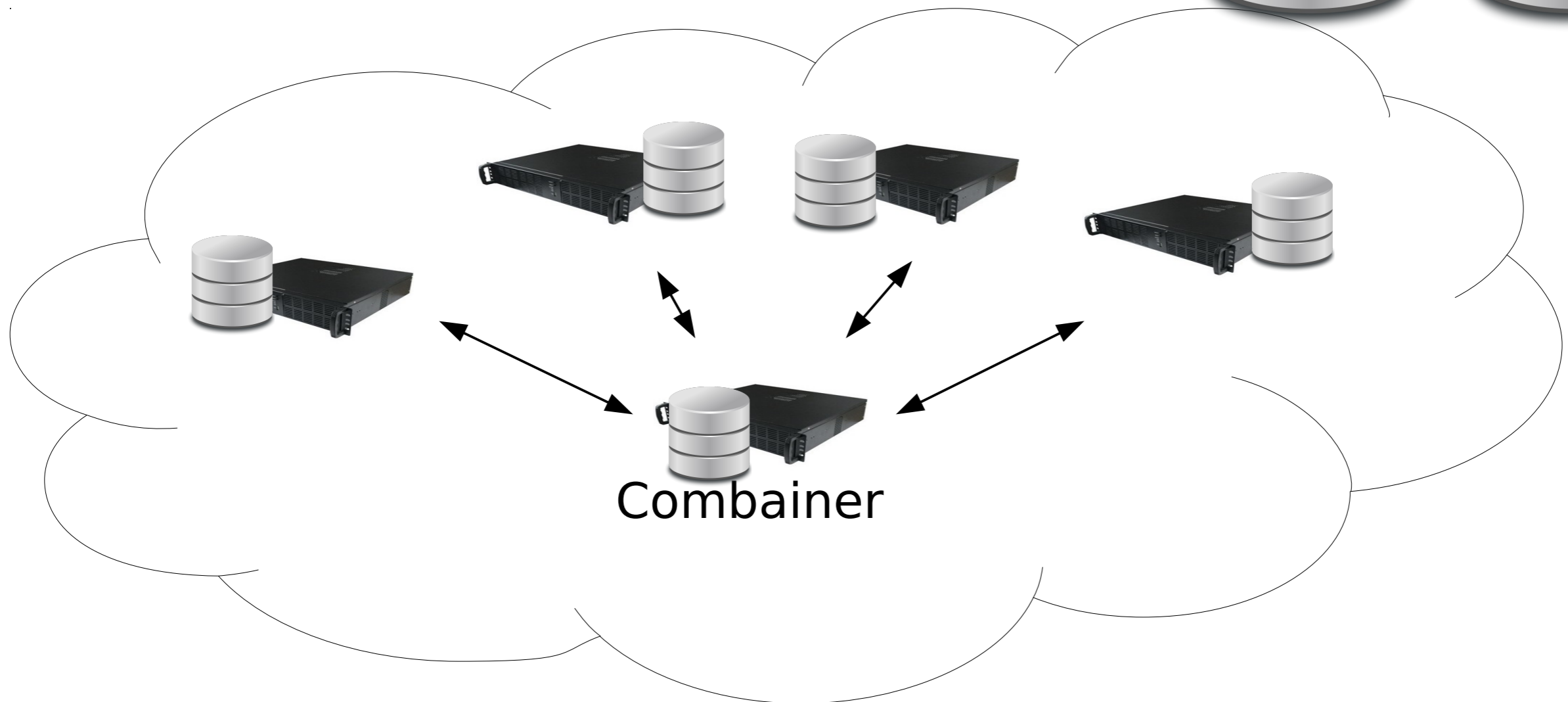
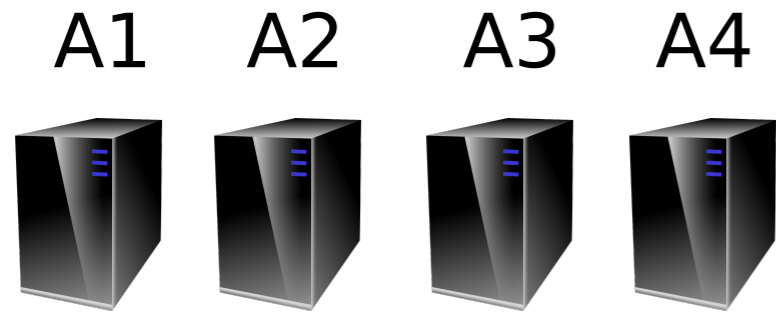
Combainer3

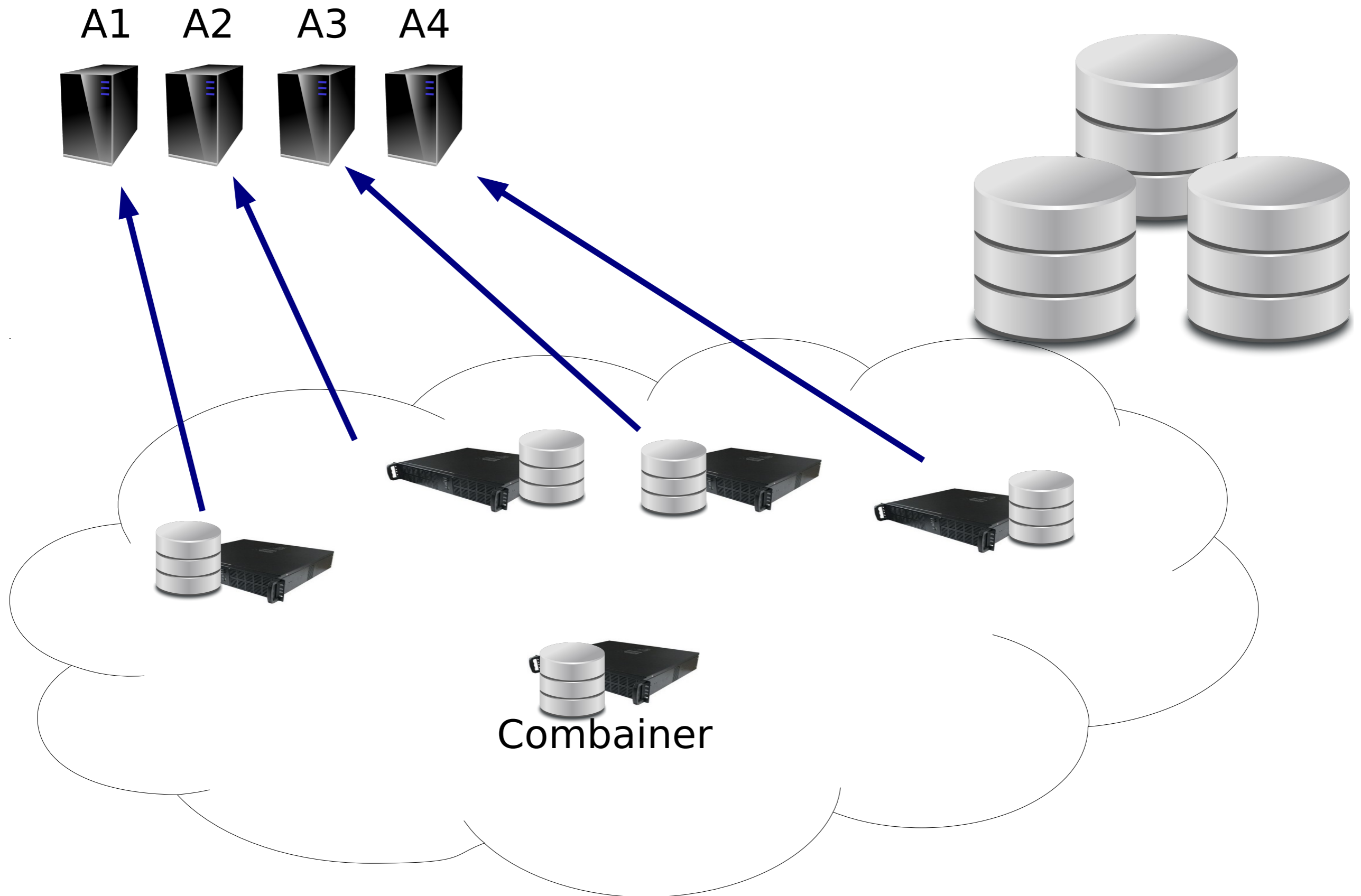


Combainer4

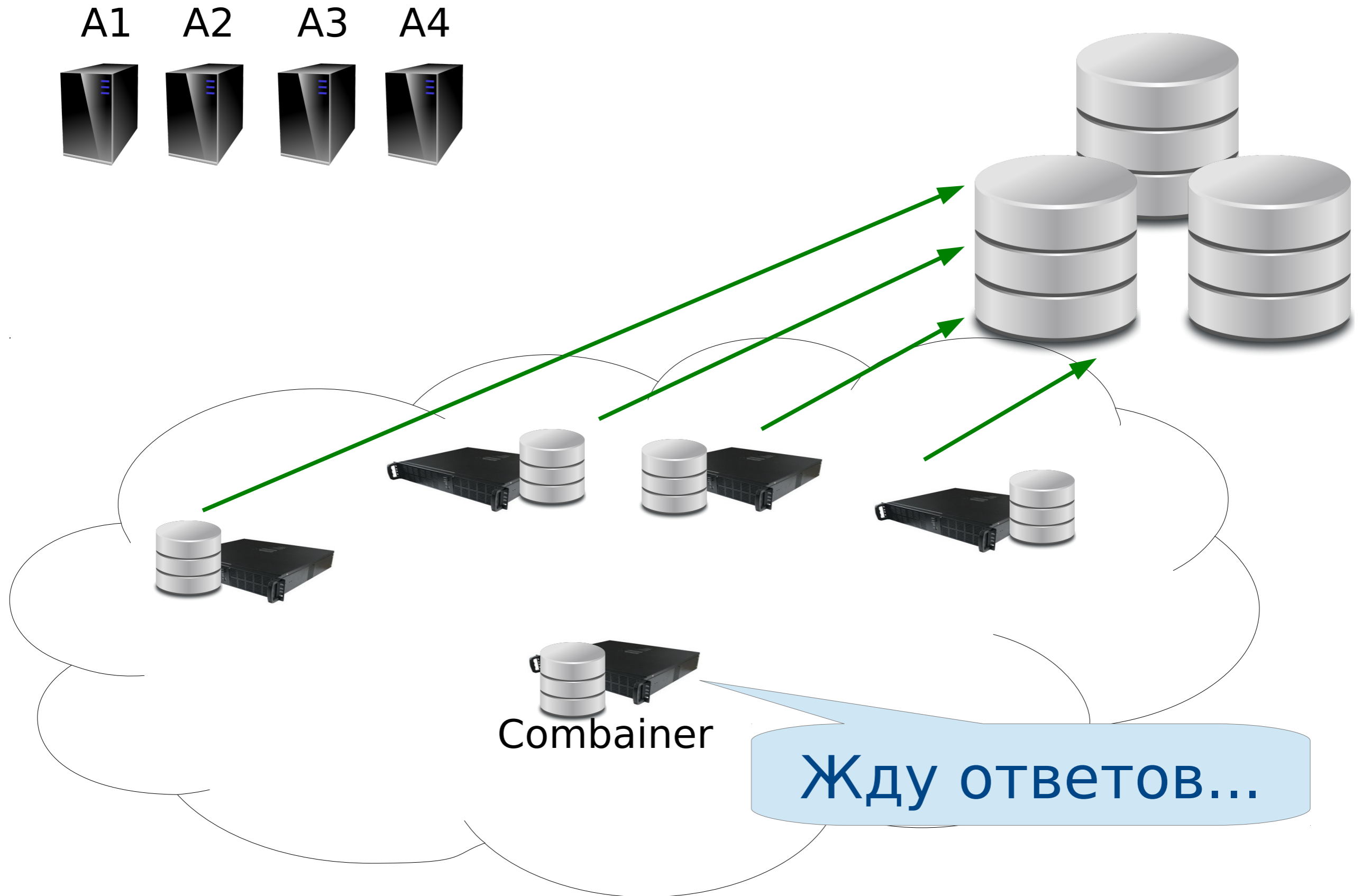
Zookeeper



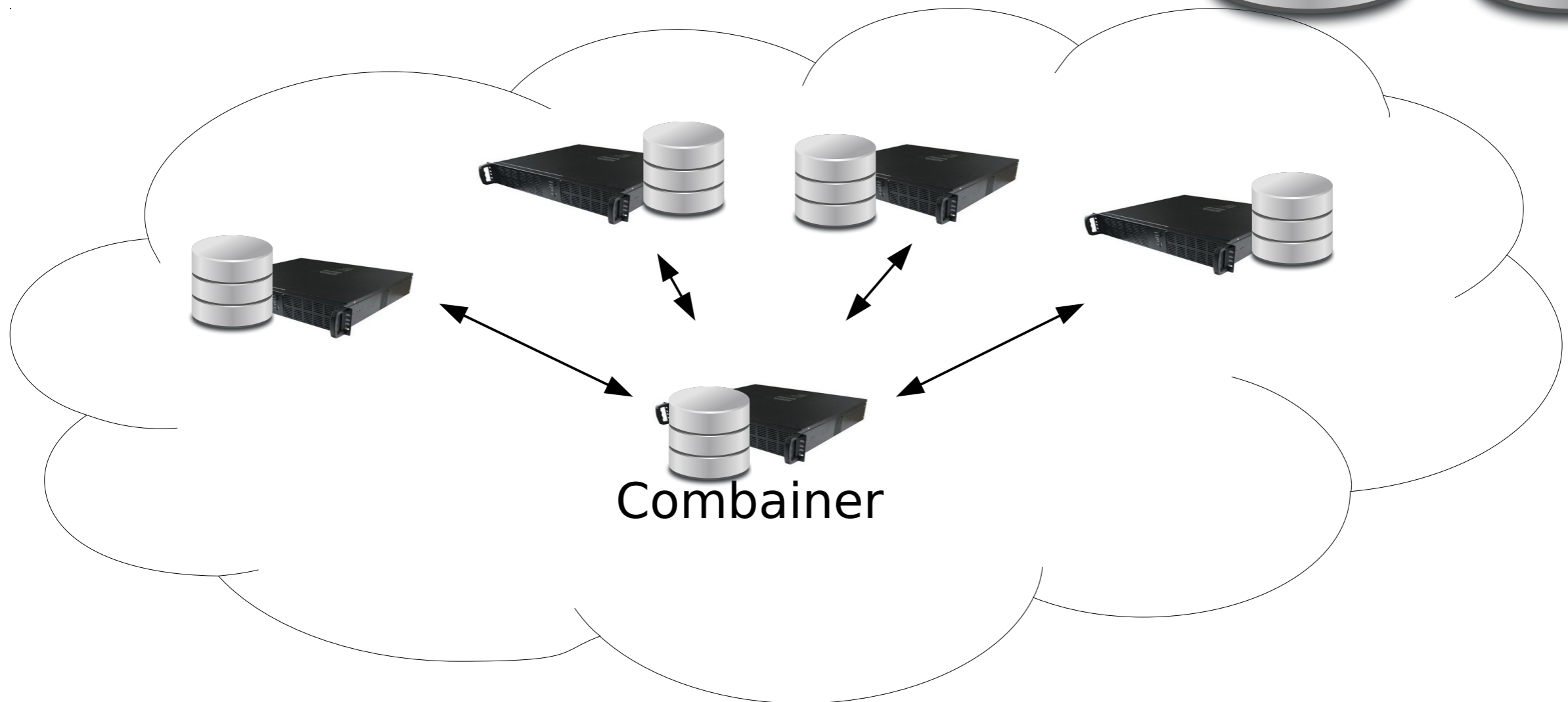
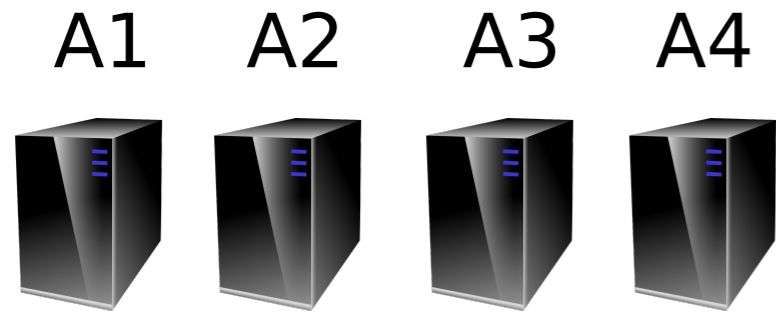




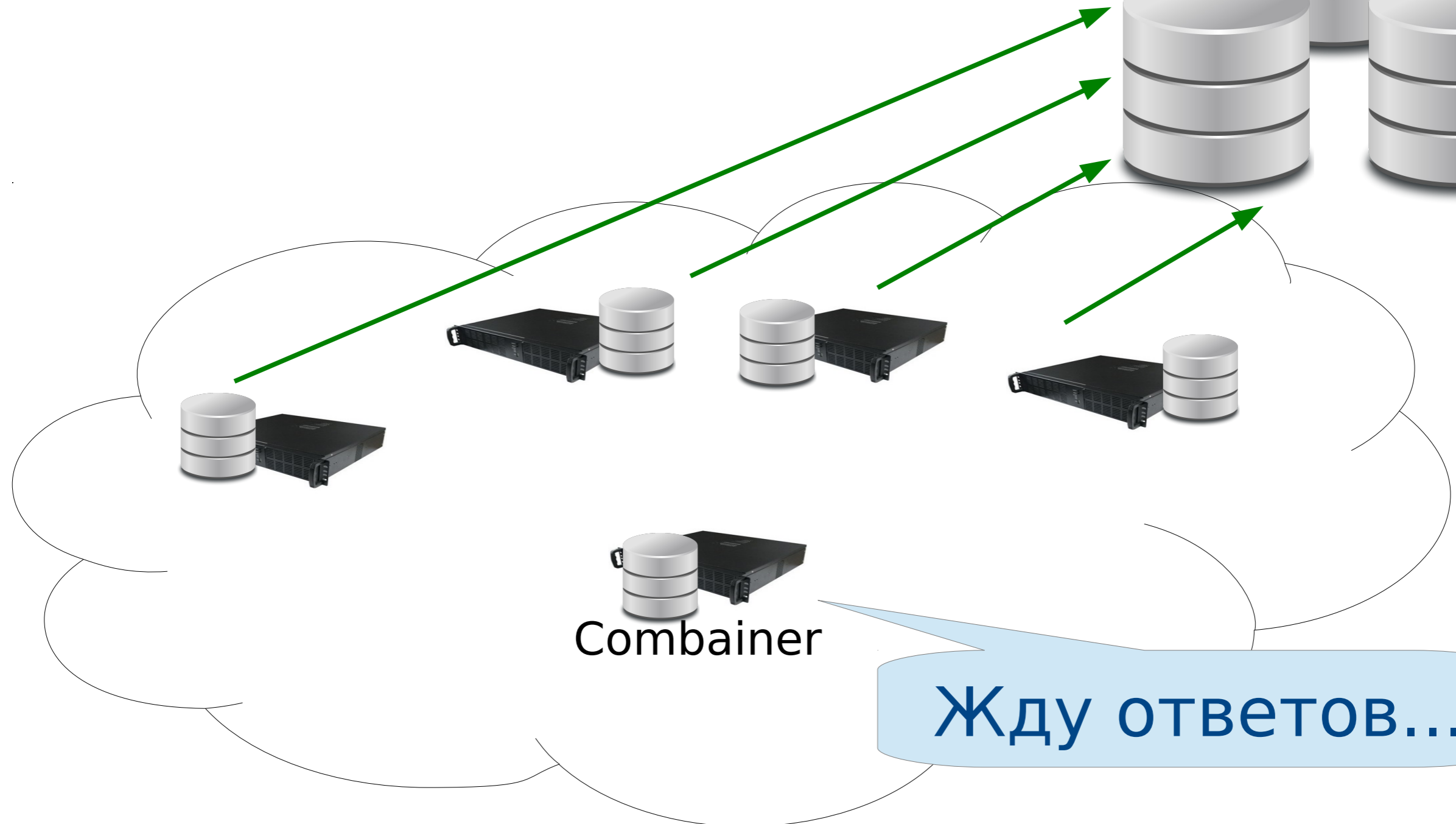
Ноды идут за данными



Агрегируем данные по хосту



A1 A2 A3 A4



Combiner

Жду ответов...

Агрегация по группе

Конфиг cocaine-dealer

```
{  
  "version" : 1,  
  "logger" : {  
    "type" : "STDOUT_LOGGER",  
    "flags" : "PLOG_NONE" },  
  "services" : {  
    "parsing" : {  
      "app" : "parsing@1",  
      "autodiscovery" : {  
        "source" : "my.host/cloud_machines",  
        "type" : "HTTP" }  
    },  
    "aggregate_group" : {  
      "app" : "aggregate@1",  
      "autodiscovery" : {  
        "source" : "my.host/cloud_machines",  
        "type" : "HTTP" }  
    }  
  }  
}
```

Пример конфига парсинга

```
{  
  "groups": ["service-A"],  
  "log_name": "dstat/output.log",  
  "parser": "dstat_parser",  
  "index": ["Time"],  
  "agg_configs": ["dstat_proc", "dstat_mem", "dstat_net", "dstat_disk",  
"dstat_la"]  
}
```

A1.yandex.net; A_dstat; service-A; 1353527992;1353527997

Конфиг парсинга

Период времени

Данные с A1.yandex.net

Задание на парсинг

Пример конфига агрегации

```
{
  "interval_period": 10,
  "data": {
    "free": {
      "host": ".find({'Time': %%})[0]['mem_free']",
      "group": "average"
    },
    "used": {
      "host": ".find({'Time': %%})[0]['mem_used']",
      "group": "average"
    },
    "buffers": {
      "host": ".find({'Time': %%})[0]['mem_buff']",
      "group": "average"
    },
    "cached": {
      "host": ".find({'Time': %%})[0]['mem_cach']",
      "group": "average"
    }
  }
}
```

service-A; A_dstat; dstat_mem; 1353527992;1353527997

Способ агрегации

Конфиг парсинга

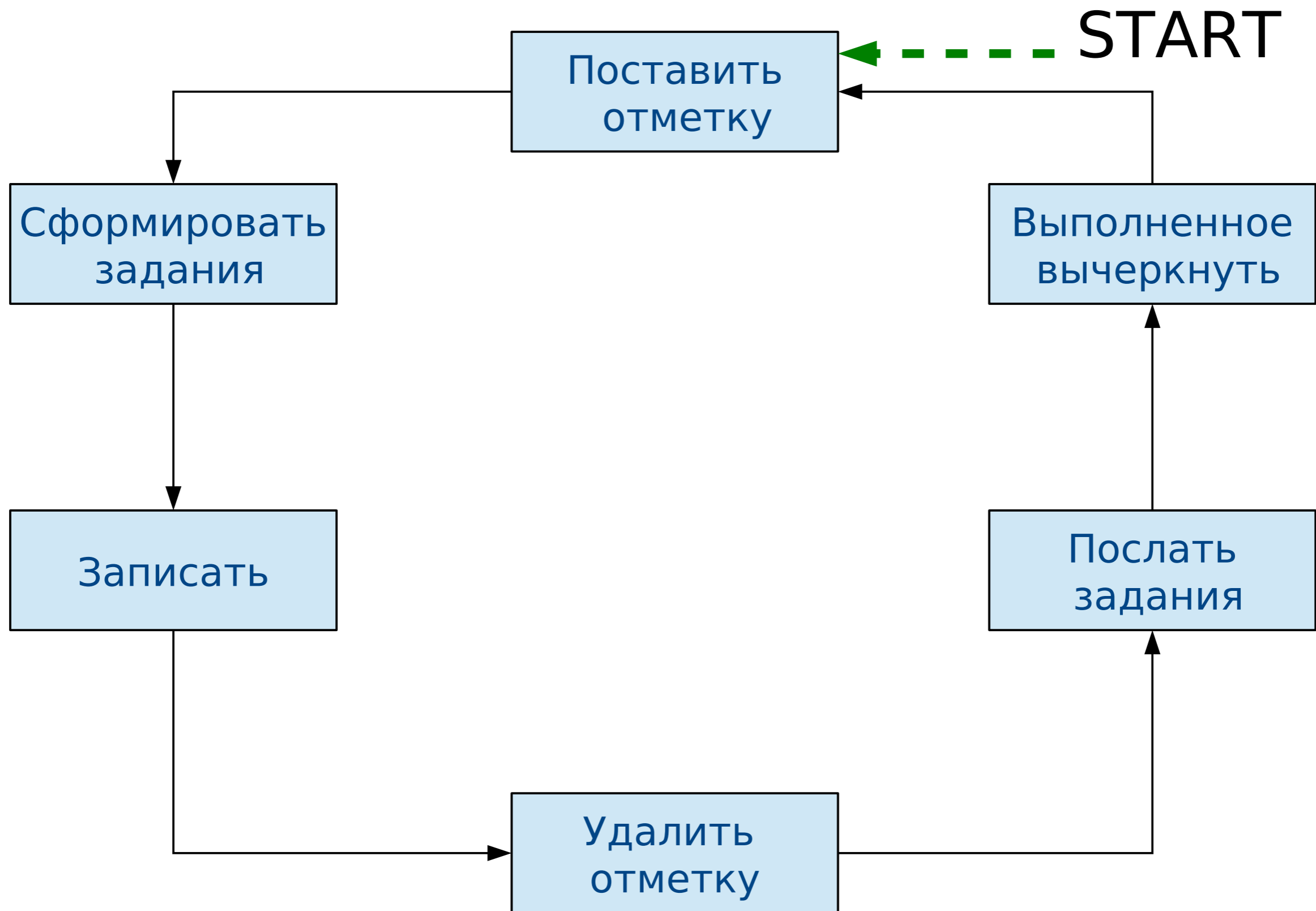
Период времени

Агрегируемая группа

Задание на агрегацию

Эфемерная нода Zookeeper

- 1.** Пропадает при отключении владельца автоматически
- 2.** Проверяется владение при любом опасном действии (`write()`, `send()`, `delete()`)
- 3.** Ставится на каждый конфиг парсинга



Сохранность прогресса внутри цикла

Распараллеливание

- 1.** Каждый combainer управляет обработкой данных с одной группы машин.
- 2.** Обработка данных производится всеми машинами облака.
- 3.** Балансировку нагрузки обеспечивает cocaine-dealer

Достоинства

- Масштабируемая структура
- Одно облако на всех
- Отказоустойчивая структура

Вопросы?



Антон Тюрин

Системный администратор

noxious@yandex.ru

@Noxi0uz