

hl⁺⁺

HighLoad⁺⁺

Приемы разработки высоконагруженных приложений на **Twisted (Python)**

Андрей Смирнов /  /



Задача

- Постоянные соединения
- Много соединений
- "Легкие" запросы
- Низкое CPU time/wall time на один запрос

hl⁺⁺

HighLoad⁺⁺



Twisted Framework

- **Python**
- Ядро: **async IO** (select/poll)
- **Deferred** — асинхронная модель программирования
- Реализация протоколов:
 - DNS, HTTP, IMAP, memcached, AMQP, ...

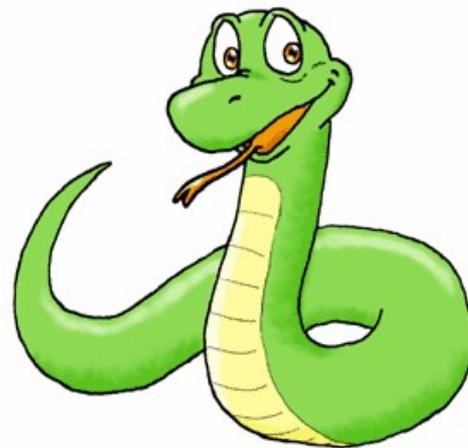
Сервер на Twisted

- Один процесс, одна нить
- **Асинхронный ввод-вывод**
(memcached, nginx, haproxy, ...)
- В одном процессе обрабатывается несколько запросов одновременно



Миф/правда №1

- **Python** не подходит для серверной разработки



Python - медленный

- Примерно **10x** медленнее C
- Но **90%** времени выполнения приходится на **10%** кода
- Cython, Nuitka, Unladen Swallow, ...
- Профайлинг — kcachegrind





Память в Python

- Подсчет ссылок
- Сборщик мусора для разрыва циклов
- Средства анализа — gc, heapu
- Оптимизация: `__slots__`, Cython, ...



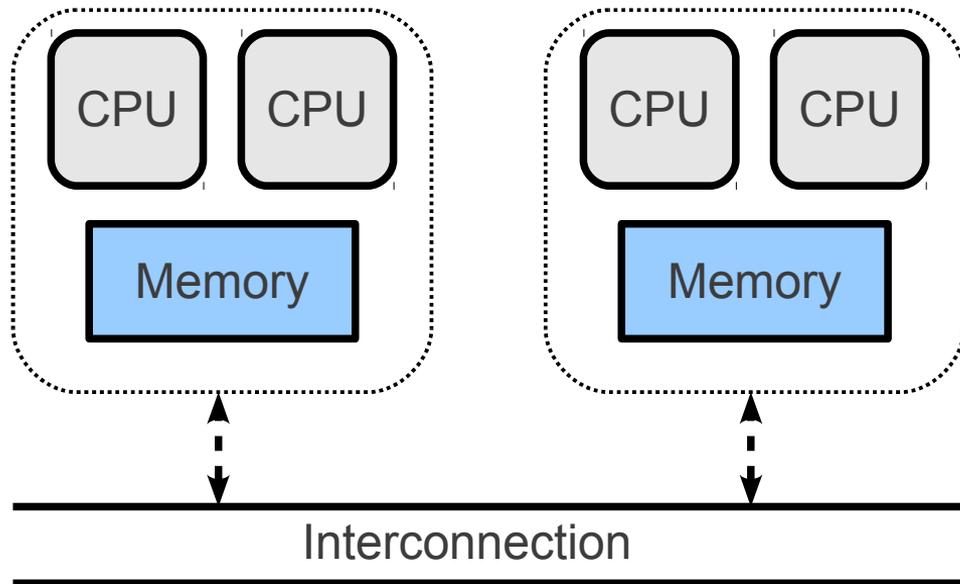
Один поток/процесс

- Многоядерные процессоры
- Локальность, NUMA, привязка к процессору

hl⁺⁺

HighLoad⁺⁺

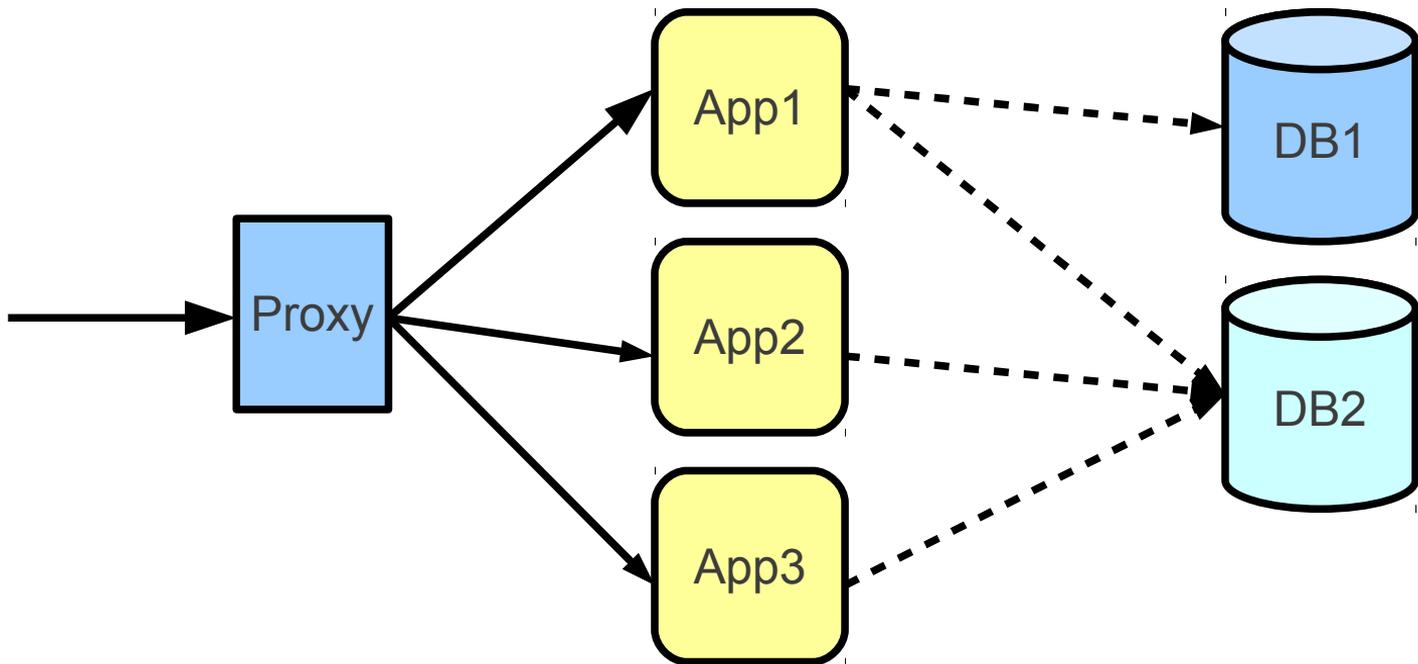
ccNUMA



- CPU binding
- Node binding
- Local alloc

+10-15%

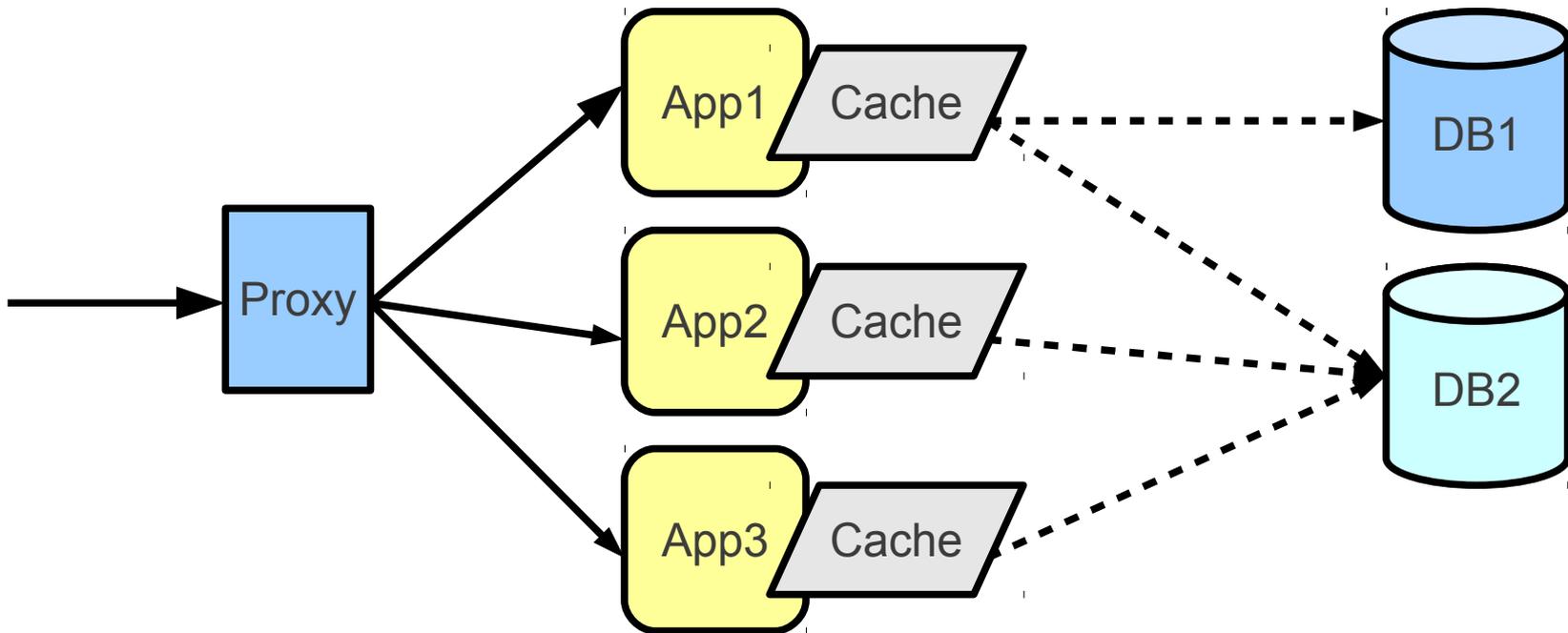
Архитектура сервиса



hl⁺⁺

HighLoad++

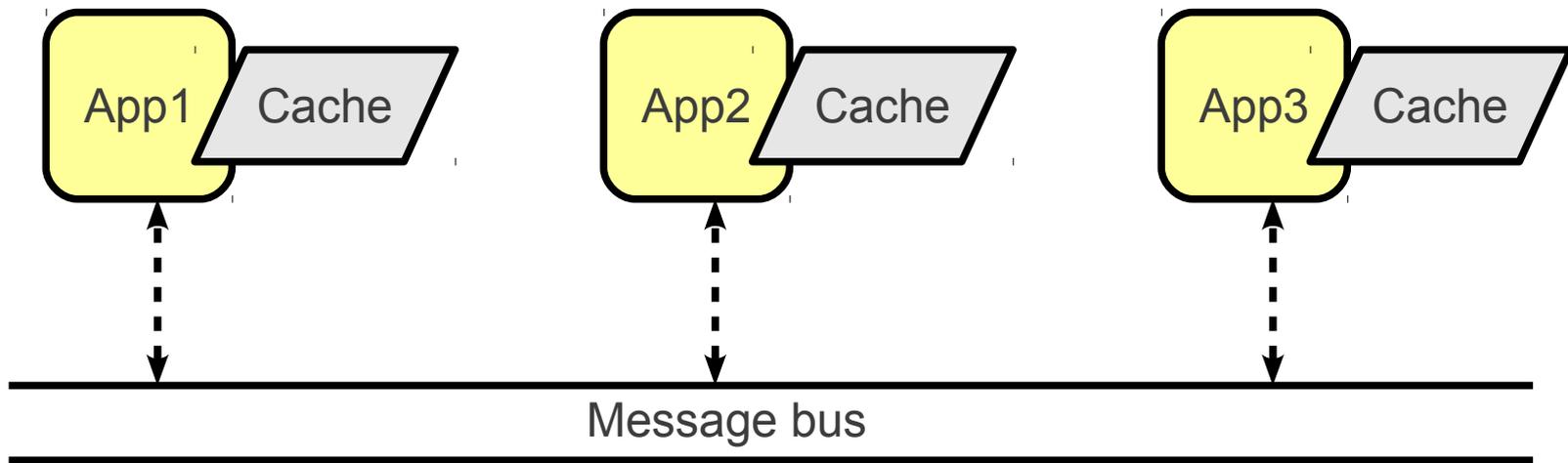
Добавим кэш



hl⁺⁺

HighLoad⁺⁺

Когерентность кэшей

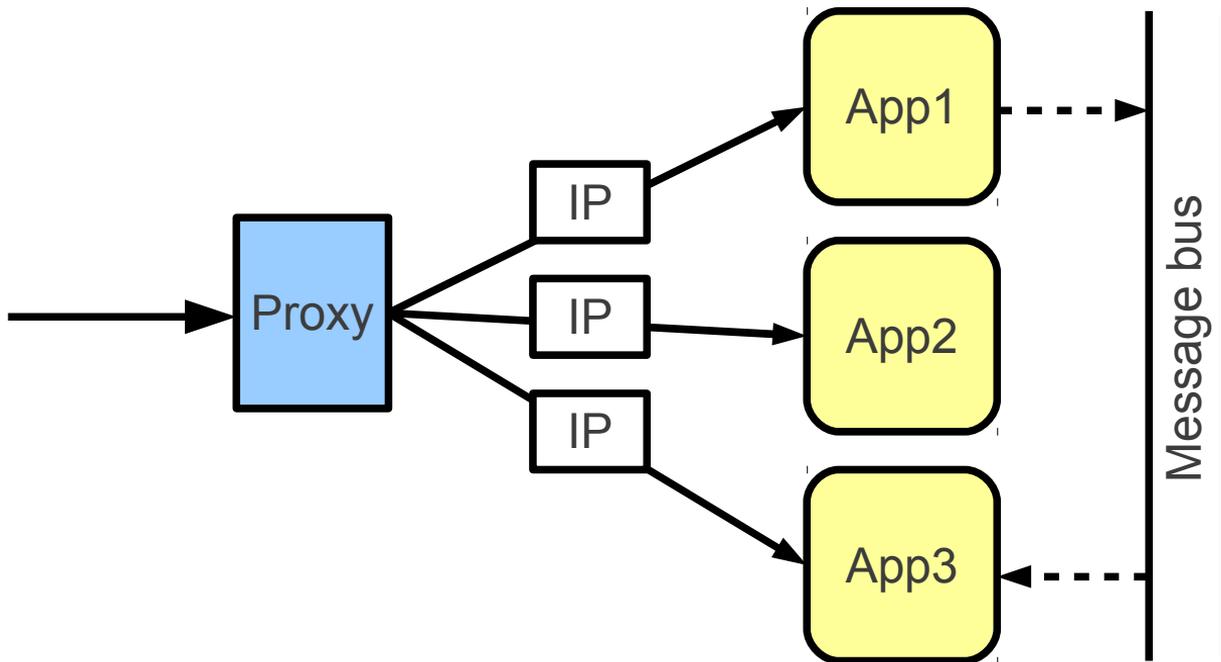


AMQP: шина сообщений

- Варианты доставки:
 - fanout (один ко всем)
 - origin (один к одному)
- RabbitMQ — брокер
- txAMQP — драйвер
- ZeroMQ



Локализация клиентов



hl⁺⁺

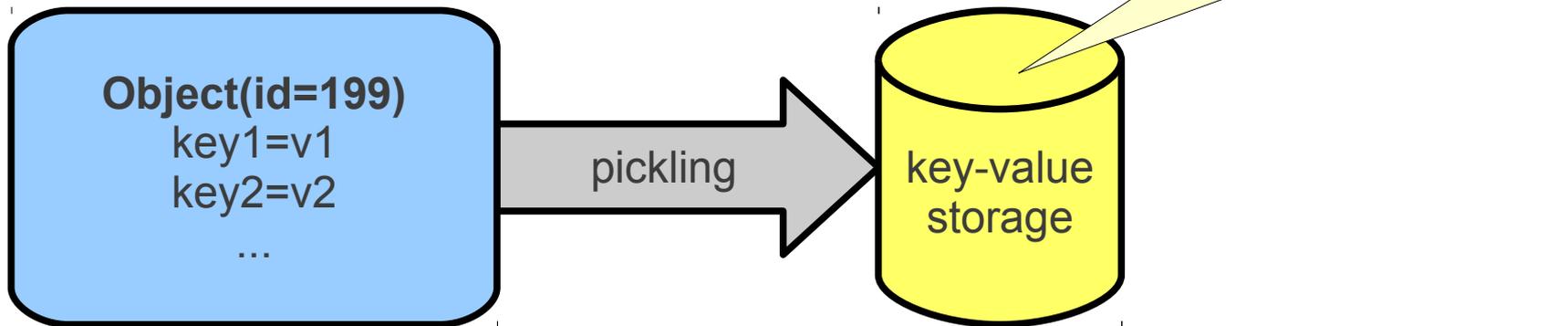
HighLoad⁺⁺

Key-value хранилища

- Memcached — встроенный
 - memcached, MemcacheDB, MemcacheQ, membase, ...
- txRedis
- txMongo
- ...



Хранение объектов

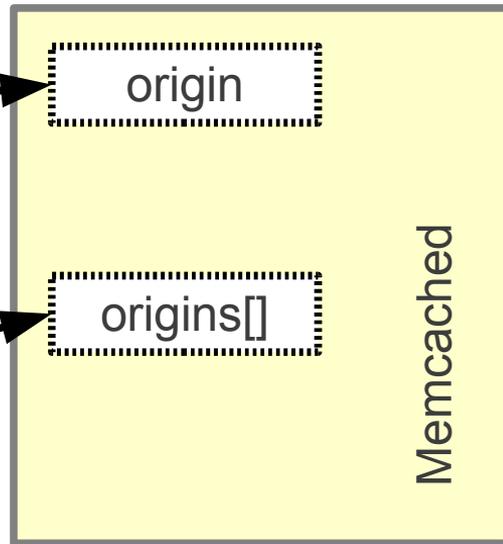


Сортированный список



Нумерация экземпляров

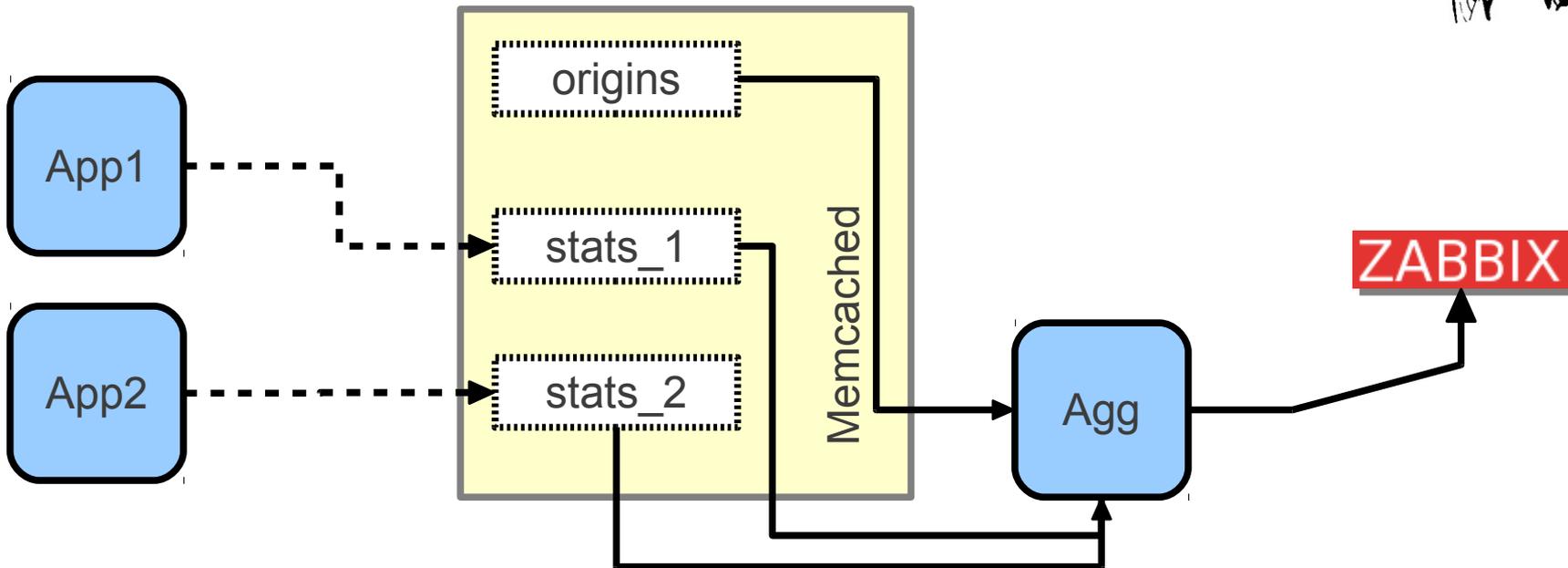
- "Свой" номер
 - increment/add
- Список всех экземпляров
 - append/add



hl⁺⁺

HighLoad++

Статистика



hl++

```
>>> from guppy import hpy
>>> hp = hpy()
>>> h = hp.heap()
>>> h
```

Partition of a set of 217545 objects. Total size = 29877016 bytes.

Index	Count	%	Size	% Cumulative	% Kind (class / dict of class)
0	95449	44	9085280	30	9085280 30 str
1	52060	24	4394208	15	13479488 45 tuple
2	4067	2	2250440	8	15729928 53 dict (no owner)
3	13588	6	1630560	5	17360488 58 types.CodeType
4	619	0	1586440	5	18946928 63 dict of module
5	13030	6	1563600	5	20510528 69 function
6	1640	1	1476848	5	21987376 74 type
7	1640	1	1239872	4	23227248 78 dict of type
8	876	0	850464	3	24077712 81 dict of class
9	3121	1	638008	2	24715720 83 unicode

<675 more rows. Type e.g. '_.more' to view.>

```
>>> h[0]
```

Partition of a set of 95449 objects. Total size = 9085280 bytes.

Index	Count	%	Size	% Cumulative	% Kind (class / dict of class)
0	95449	100	9085280	100	9085280 100 str

```
>>> █
```



C100k+

- Firewall в stateless mode
- twistd: reactor=epoll (kqueue)
- Минимизируем активность соединения
- Анализируем объем памяти на соединение



Масштабирование

- Каждый новый клиент — какие ресурсы потребляет?
- Какой процент ресурсов "сервера" и остальных серверов (в т.ч. разделяемых)?

hl⁺⁺

HighLoad⁺⁺

Масштабирование

Ресурсы "своего" сервера

Другие
сервера

Разделяемые
ресурсы



Оптимизация

- Минимизация межсерверного взаимодействия
 - Агрегация сообщений
- Используем высокопроизводительные КОМПОНЕНТЫ
 - memcached, MongoDB, RabbitMQ, cPickle, ...

Цифры

- JSON-RPC, "hello world"
 - 890 req/s @ 0.022 sec
 - 530 req/s @ 0.006 sec

Цифры

- Реальная система (кластер)
 - **JSON-RPC**: 710 req/sec
 - **memcachedX**: 11833 req/sec
 - **AMQP**: 800 req/sec
 - **connections**: 600K+ (670 reconnects/sec)

hl⁺⁺

HighLoad⁺⁺



Контакты

- Андрей Смирнов
 - Руководитель web-разработки Qik
- me@smira.ru, smira@qik.com
- <http://www.smira.ru/>, [@smira](#)