

1С-БИТРИКС



СИСТЕМЫ УПРАВЛЕНИЯ ВЕБ-ПРОЕКТАМИ И КОРПОРАТИВНОЙ ИНФОРМАЦИЕЙ



1С-Битрикс: Управление сайтом 10.0

Веб-кластер



Сколько стоит 1 час?

Пример:

Крупный интернет-магазин с годовым оборотом 1.5 млрд. руб.

210 рабочих дней в году по 10 рабочих часов.

Час простоя крупного интернет-проекта может обойтись владельцам в 0,3 - 1 миллион рублей упущенной выручки.

Более 50 000
проектов



ЕВРОСЕТЬ



ЗЕЛДОРАДО



МЧС России



1С-Битрикс: Веб-кластер

Основные задачи, которые необходимо решить:

1. **Обеспечение высокой доступности сервиса** (так называемые HA - High Availability или Failover кластеры)
2. **Масштабирование веб-проекта в условиях возрастающей нагрузки** (HP - High Performance кластеры)
3. **Балансирование нагрузки, трафика, данных между несколькими серверами.**
4. **Создание целостной резервной копии данных для MySQL.**





1С-Битрикс: Веб-кластер

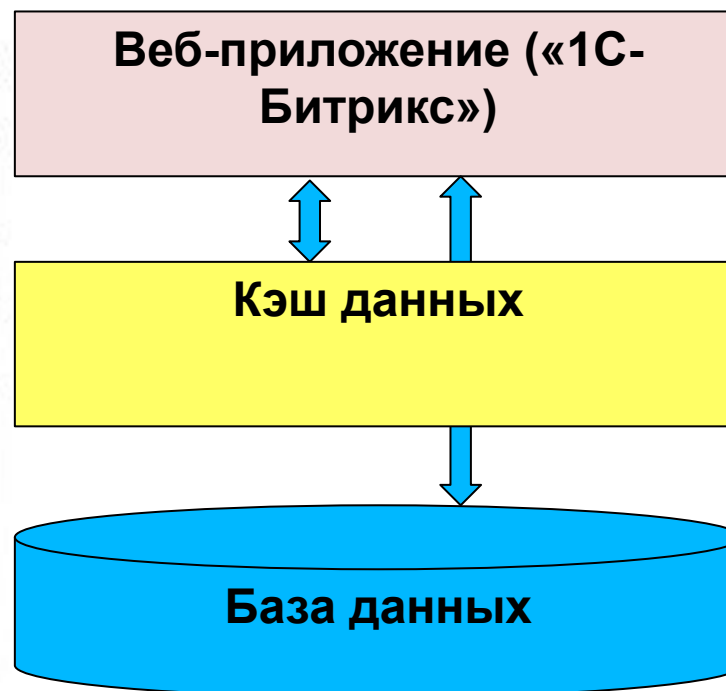
«Веб-кластер» обеспечивает непрерывность бизнеса, отказоустойчивость, масштабирование, распределение нагрузки.

Любой новый или работающий проект на 1С-Битрикс: Управление сайтом 10.0 может быть представлен как веб-кластер взаимозаменяемых серверов.

1. При увеличении посещаемости можно быстро добавить в кластер новые сервера.
2. В случае выхода из строя одного из серверов кластера система продолжает беспрерывно обслуживать Клиентов.
3. Балансирование нагрузки, трафика, данных между несколькими серверами.
4. Система позволяет снимать резервные копии со специально выделенных узлов кластера, не влияя на работу сайта.



Традиционная конфигурация



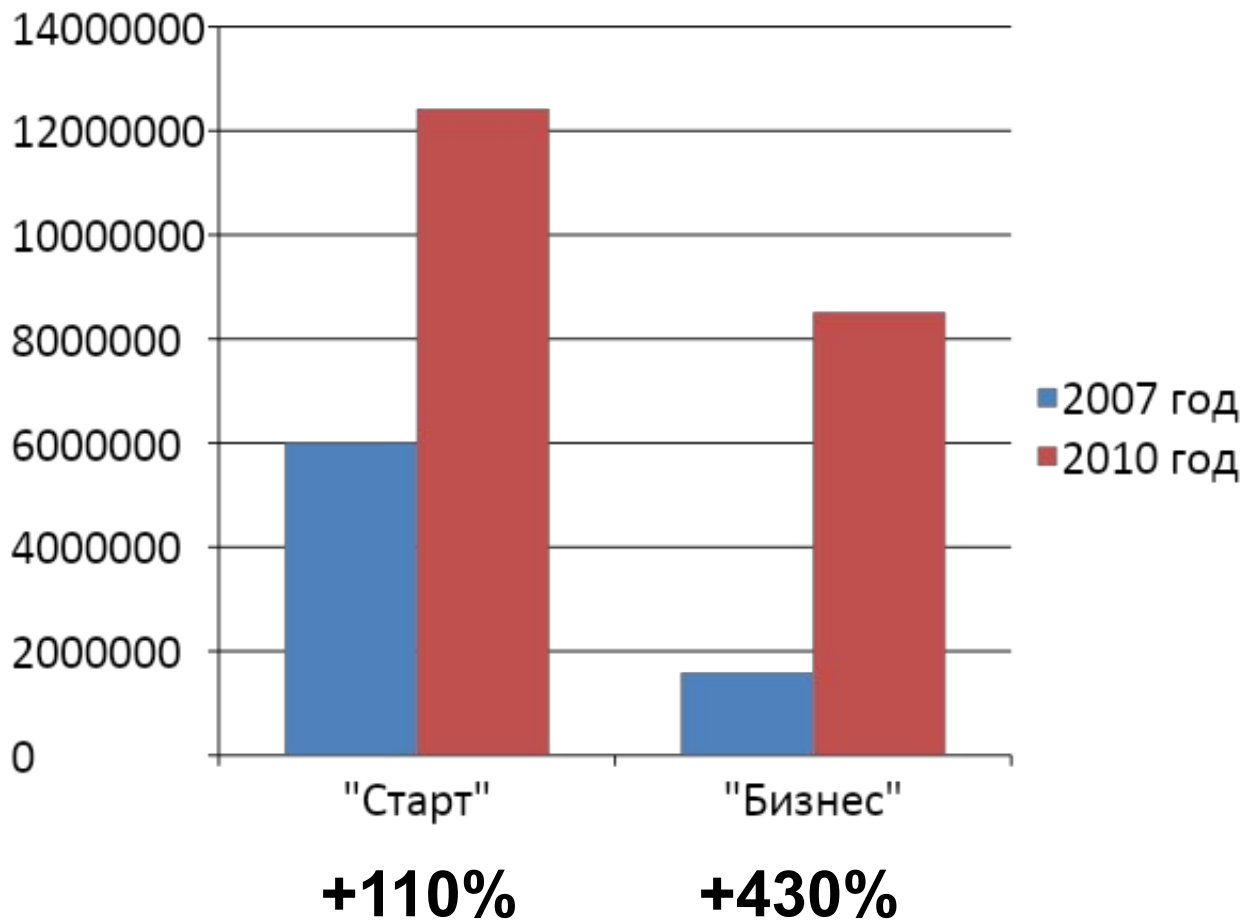


История производительности платформы

- До 2005 года вопросом производительности системно не занимались.
- 2005 год – производительность стала существенной задачей для разработки.
- 2007 год – появление инструментов отладки SQL-запросов. Системная работа над производительностью продукта.
- 2007 год – первое нагрузочное тестирование с QSOFТ (1.5 млн. хитов в сутки на редакции «Бизнес», 6 млн. – на редакции «Старт»).
- 2008-2010 годы – развернуто 4 конфигурации Oracle RAC с 4 серверами.
- 2009 год – «монитор производительности» во всех редакциях продукта.
- 2009-2010 годы – выпущены «1С-Битрикс: Виртуальная машина» и «1С-Битрикс: Веб-окружение».
- 2008-2011 – сертификация хостинг-провайдеров.
- 2010 год – рост производительности – на 430%! Новые нагрузочные тесты: 8.5 млн. хитов – «Бизнес», 12.4 млн. – «Старт», 85 млн. – «HTML кеш».



За три года – на 430% быстрее!





Варианты масштабирования до 10.0

1. Разделение на два сервера: веб-сервер + база данных.
2. Увеличение мощности оборудования (чем мощнее – тем дороже; рост стоимости не пропорционален).
3. Выделение кеша на один внешний сервер через memcached.
4. Переход на Oracle (минимальная лицензия +5000\$ за процессор).
5. Создание Oracle RAC (Real Application Cluster). Проект – около 150 000\$ (оборудование + лицензия + «общая полка»). Очень мало специалистов.

Для большинства клиентов производительности достаточно, но не решены проблемы отказоустойчивости, резервирования, сетевой доступности.



1С-Битрикс: Веб-кластер

«1С-Битрикс: Веб-кластер» - это комбинация технологий:

- **Вертикальный шардинг** (вынесение модулей на отдельные серверы MySQL)
- **Репликация MySQL** (Oracle и MS SQL в дальнейшем) и **балансирование нагрузки между серверами**
- **Распределенный кеш данных** (memcached)
- **Непрерывность сессий между веб-серверами** (**хранение сессий в базе данных**)
- **Кластеризация веб-сервера:**
 - Синхронизация файлов
 - Балансирование нагрузки между серверами



1С-Битрикс: Веб-кластер

Navigation

Region: US West (N. California)

- EC2 Dashboard
- INSTANCES
 - Instances
 - Spot Requests
- IMAGES
 - AMIs
 - Bundle Tasks
- ELASTIC BLOCK STORE
 - Volumes
 - Snapshots
- NETWORKING & SECURITY
 - Security Groups
 - Placement Groups
 - Elastic IPs
 - Load Balancers

My Instances

Launch Instance Instance Actions Reserved Instances Show/Hide Refresh Help

Viewing: All Instances All Instance Types 1 to 5 of 5 Instances

	Name	Instance	AMI ID	Root Device	Type	Status	Security Groups
<input type="checkbox"/>	master.cluster (CentOS 5.4)		ami-6d7f2f28	ebs	m1.large	● running	BitrixCluster
<input type="checkbox"/>	slave.cluster (CentOS 5.4)		ami-6d7f2f28	ebs	m1.large	● running	BitrixCluster
<input type="checkbox"/>	balancer.cluster (CentOS 5.4)		ami-6d7f2f28	ebs	t1.micro	● running	BitrixCluster
<input type="checkbox"/>	HighLoader (CentOS)		ami-6d7f2f28	ebs	m1.large	● terminatec	BitrixCluster
<input type="checkbox"/>	HighLoader (CentOS - 1b zone)		ami-6d7f2f28	ebs	m1.large	● running	BitrixCluster

0 EC2 Instances selected

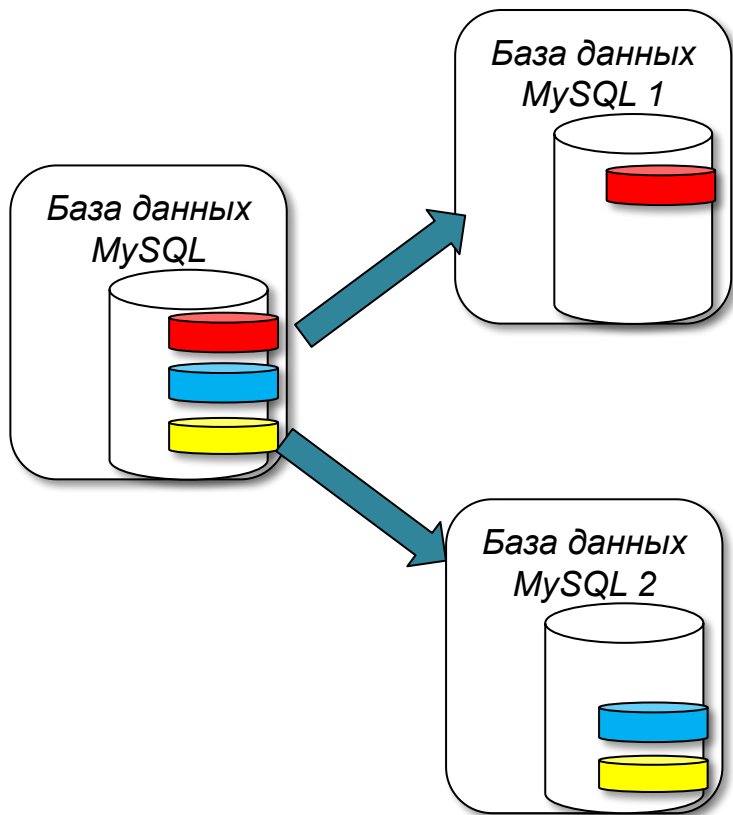
Select an instance above

Тестовый веб-кластер – в «облаке» Amazon

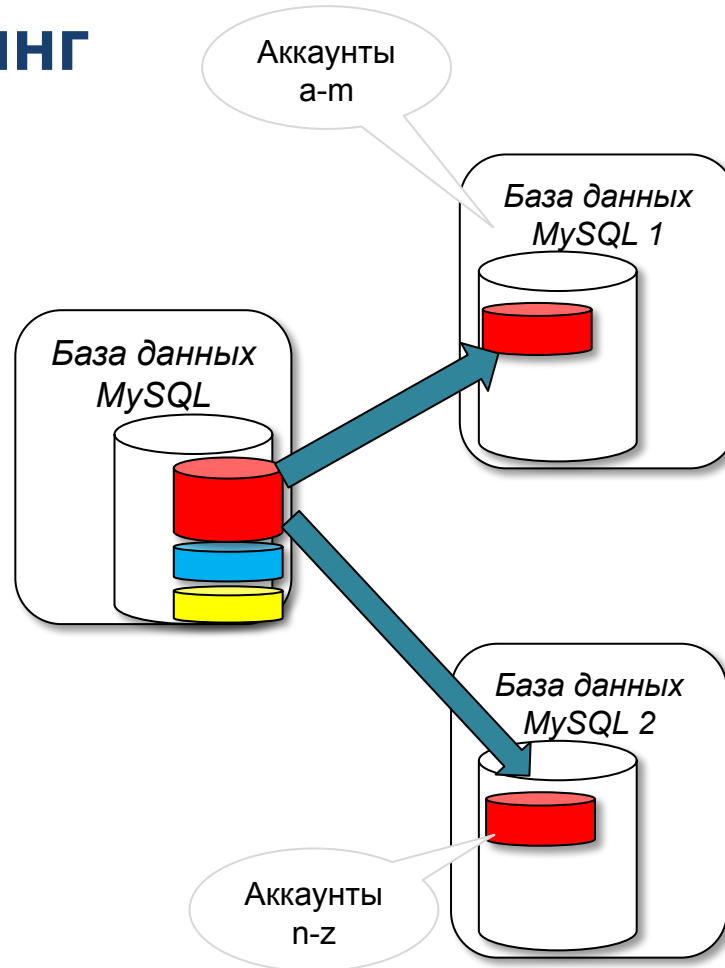




Шардинг



Вертикальный шардинг



Горизонтальный шардинг



Вертикальный шардинг

Разделение одной базы данных веб-приложения на две и более базы данных за счет выделения отдельных модулей, без изменения логики работы веб-приложения:

- Веб-аналитика
- Поиск

1. Эффективное распределение нагрузки.
2. Масштабирование.
3. Разделение больших объемов данных.

Вертикальный шардинг (базы данных модулей)

Рабочий стол > Настройки > Веб-кластер > Шардинг

Добавить новую базу данных | Обновить | Настроить

ID	Состояние	Активность	Статус	Название	Модули
1	● время работы 3 часа	да	ONLINE	Главная база данных	
2	● время работы 5 минут	да	ONLINE	Веб-аналитика	statistic
3	● время работы 5 минут	да	ONLINE	Поиск	search

Выбрано: 3 | Отмечено: 0



Репликация и балансировка нагрузки MySQL

- Гибкая балансировка нагрузки SQL
- Простота администрирования
- Дешевое и быстрое неограниченное масштабирование
- Он-лайн бэкап
- Не требуется доработка логики веб-приложения

Slave базы данных

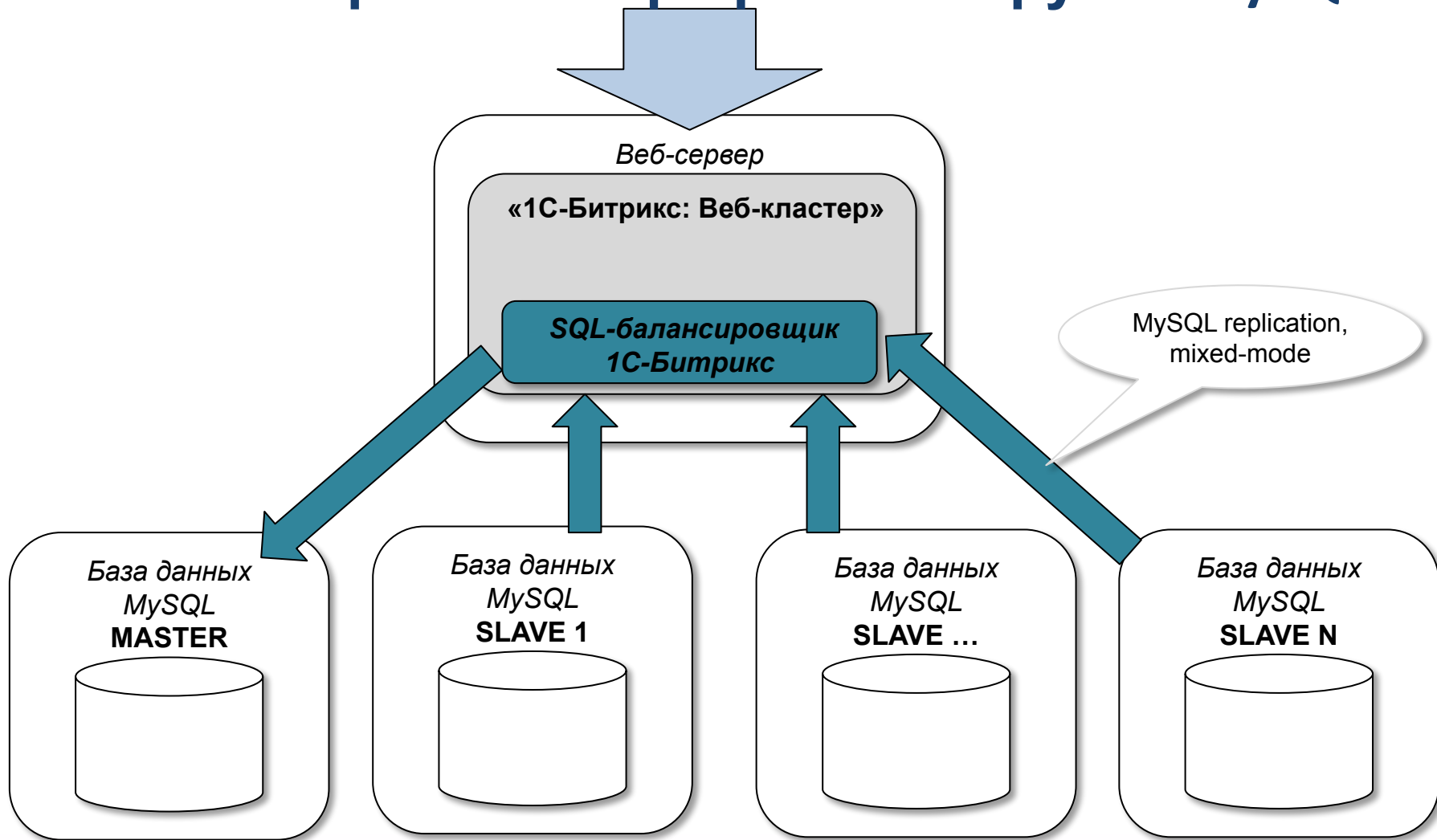
Рабочий стол > Настройки > Веб-кластер > Репликация

Добавить slave базу данных | Обновить | Настроить

ID	Состояние	Название	Отставание (сек)	Статус	Использовать (%)
1	● время работы 3 часа	Главная база данных	0	ONLINE server_id: 1 File: mysql-bin.000029 Position: 17178307 Com_select: 8902 (+10)	50
4	● время работы 19 минут	slave #1	0	ONLINE server_id: 2 Slave_IO_State: Waiting for master to send event Slave_IO_Running: Yes Read_Master_Log_Pos: 17178307 Slave_SQL_Running: Yes Exec_Master_Log_Pos: 17178307 Seconds_Behind_Master: 0 Last_IO_Error: Last_SQL_Error: Com_select: 1409 (+13)	100

Выбрано: 2 | Отмечено: 0

Масштабирование при росте нагрузки MySQL





Репликация и балансировка нагрузки MySQL

```

root@node2:~
top - 18:31:42 up 5 days, 1:07, 1 user, load average: 0.36, 0.30, 0.22
Tasks: 80 total, 2 running, 78 sleeping, 0 stopped, 0 zombie
Cpu(s): 1.2%us, 0.9%sy, 0.0%ni, 97.2%id, 0.6%wa, 0.0%hi, 0.0%si, 0.1%st
Mem: 7872040k total, 1637524k used, 6234516k free, 310404k buffers
Swap: 0k total, 0k used, 0k free, 653108k cached

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM    TIME+  COMMAND
 26560 mysql    15   0 1022m 147m 5980  S   4.7   1.9   0:10.19  mysqld
   88 root      15   0     0     0     0  S   0.3   0.0   1:04.98  psflus
    1 root      15   0 10348   796   668  S   0.0   0.0   0:01.06  init
    2 root      RT   0     0     0     0  S   0.0   0.0   0:00.47  migration/0
    3 root      34  19     0     0     0  S   0.0   0.0   0:00.00  ksoftirqd/0
    4 root      RT   0     0     0     0  S   0.0   0.0   0:00.00  watchdog/0
    5 root      10  -5     0     0     0  S   0.0   0.0   0:00.08  events/0
    6 root      10  -5     0     0     0  S   0.0   0.0   0:01.02  khelper
    7 root      11  -5     0     0     0  S   0.0   0.0   0:00.00  kthread
    9 root      16  -5     0     0     0  S   0.0   0.0   0:00.00  xenwatch
   10 root      10  -5     0     0     0  S   0.0   0.0   0:00.00  xenbus
   16 root      RT  -5     0     0     0  S   0.0   0.0   0:00.18  migration/1
   17 root      34  19     0     0     0  S   0.0   0.0   0:00.00  ksoftirqd/1
   18 root      RT  -5     0     0     0  S   0.0   0.0   0:00.00  watchdog/1
   19 root      10  -5     0     0     0  S   0.0   0.0   0:00.05  events/1
    
```

```

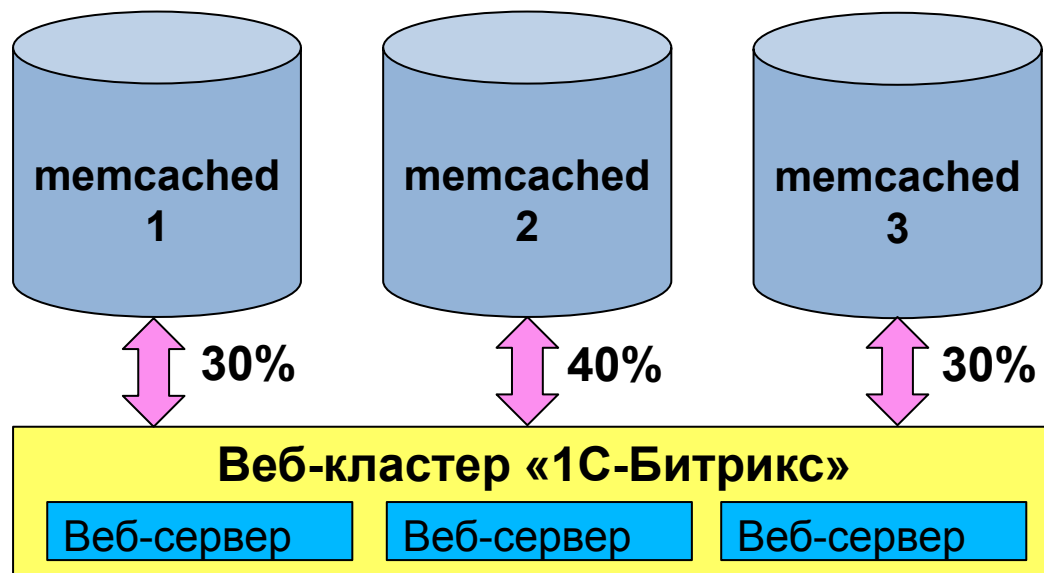
average: 9.48, 3.11, 1.20
stopped, 0 zombie
wa, 0.0%hi, 0.5%si, 22.5%st
free, 388836k buffers
free, 3264376k cached

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM    TIME+  COMMAND
  8 0:14.44  httpd
  8 0:13.80  httpd
  7 0:12.69  httpd
  8 0:13.43  httpd
  7 0:13.08  httpd
  8 0:14.33  httpd
  7 0:12.30  httpd
  7 0:12.59  httpd
  7 0:13.27  httpd
 22401 bitrix   15   0 504m 56m 41m  R  8.2   0.7   0:13.27  httpd
 22404 bitrix   15   0 504m 56m 41m  S  8.2   0.7   0:12.60  httpd
 22406 bitrix   15   0 504m 53m 38m  S  8.2   0.7   0:13.42  httpd
 22410 bitrix   15   0 505m 56m 40m  S  8.2   0.7   0:12.24  httpd
 22407 bitrix   15   0 505m 56m 40m  R  7.9   0.7   0:11.80  httpd
 22402 bitrix   16   0 504m 52m 37m  R  6.3   0.7   0:13.06  httpd
    
```



Распределенный кеш данных (memcached)

- Высокая эффективность - за счет централизованного использования кэша веб-приложением
- Надежность - за счет устойчивости подсистемы кеширования к выходу из строя отдельных компонентов
- Неограниченная масштабируемость - за счет добавления новых memcached-серверов.





Распределенный кеш данных (memcached)

Подключения к memcached

Рабочий стол > Настройки > Веб-кластер > Memcached

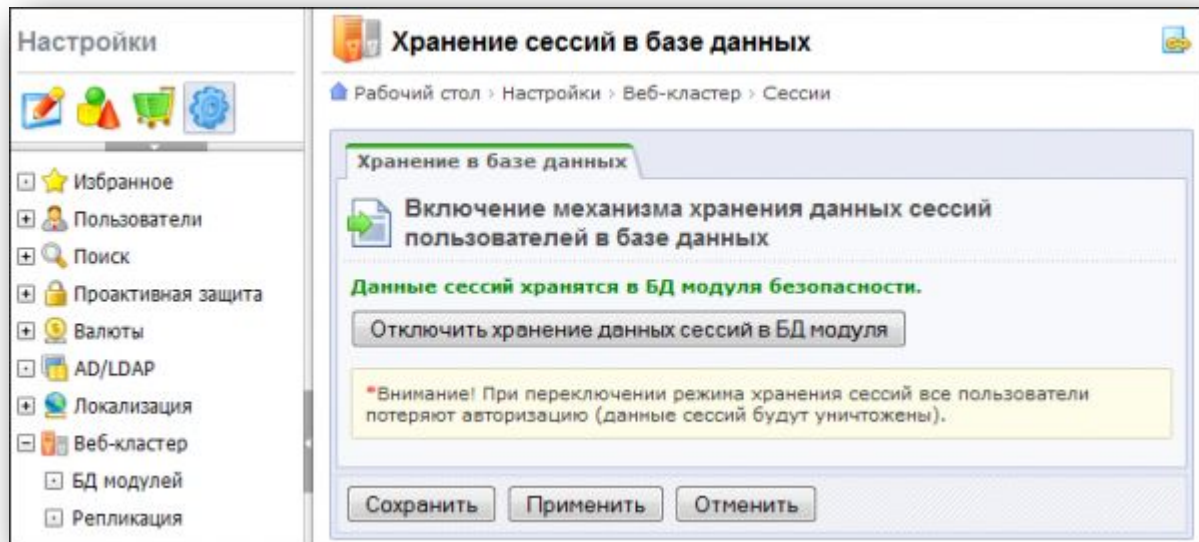
Добавить | Обновить | Настроить

ID	Состояние	Статус	Использовать (%)	Сервер
1	● время работы 3 часа	ONLINE version: 1.4.5 cmd_get: 132 (+12) cmd_set: 22 get_misses: 36 get_hits: 96 (72.73%) limit_maxbytes: 256 Mb using_bytes: 117.1 Kb (0.04%) curr_items: 22 listen_disabled_num: 0	100	node1.demo-cluster.1c-bitrix.ru:11211
2	● время работы 3 часа	ONLINE version: 1.4.5 cmd_get: 7 (+7) cmd_set: 0 get_misses: 7 get_hits: 0 limit_maxbytes: 256 Mb using_bytes: 0 b (0%) curr_items: 0 listen_disabled_num: 0	100	node2.demo-cluster.1c-bitrix.ru:11211

Выбрано: 2 | Отмечено: 0

Непрерывность сессий между веб-серверами

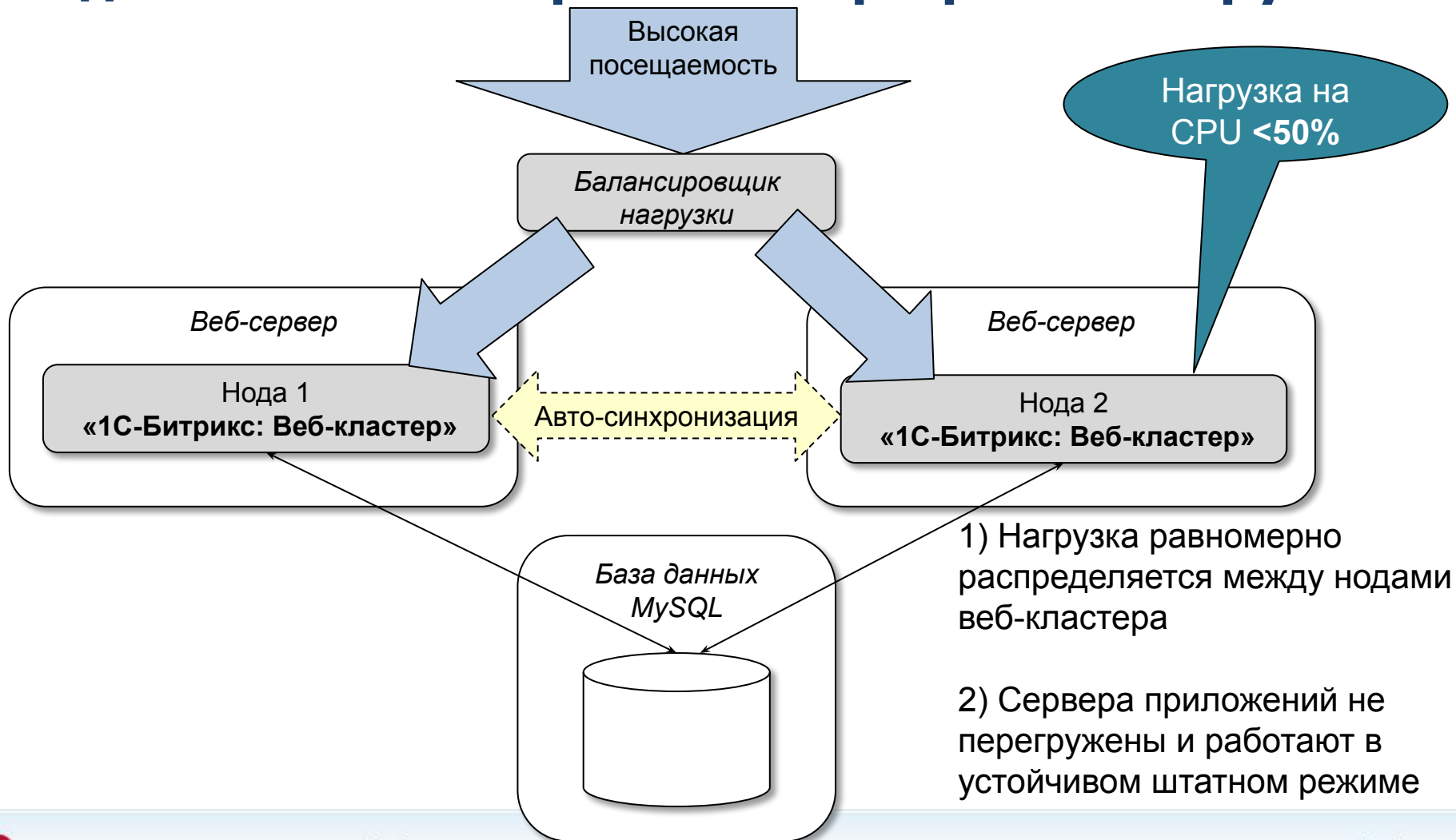
Пользовательская сессия должна быть "прозрачной" для всех серверов веб-кластера.



1. После авторизации на одном из серверов пользователь должен считаться авторизованным и для всех других серверов.
2. И наоборот - окончание сессии на любом сервере должно означать ее окончание на всех серверах сразу.

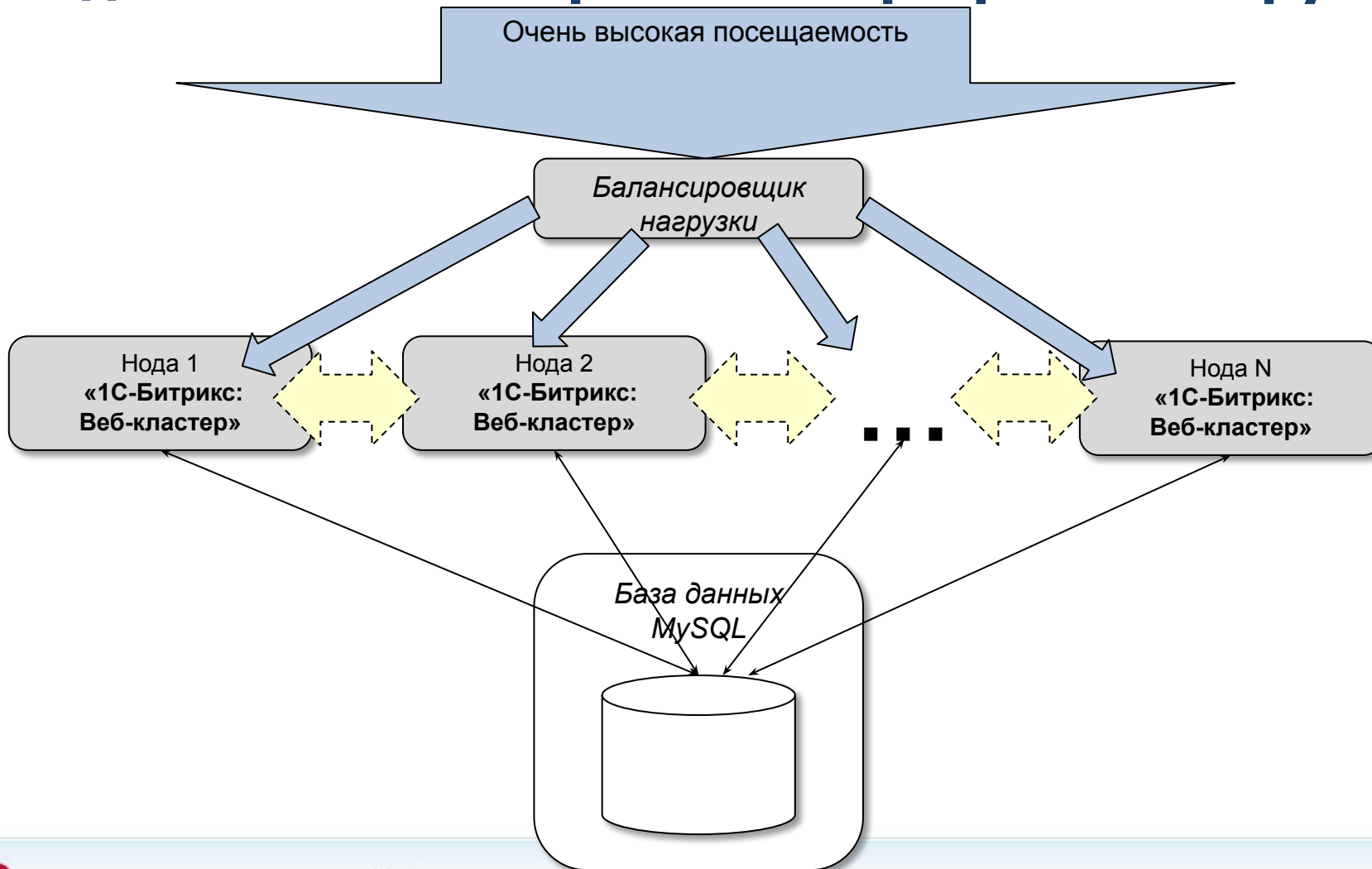


Задача: масштабирование при росте нагрузки



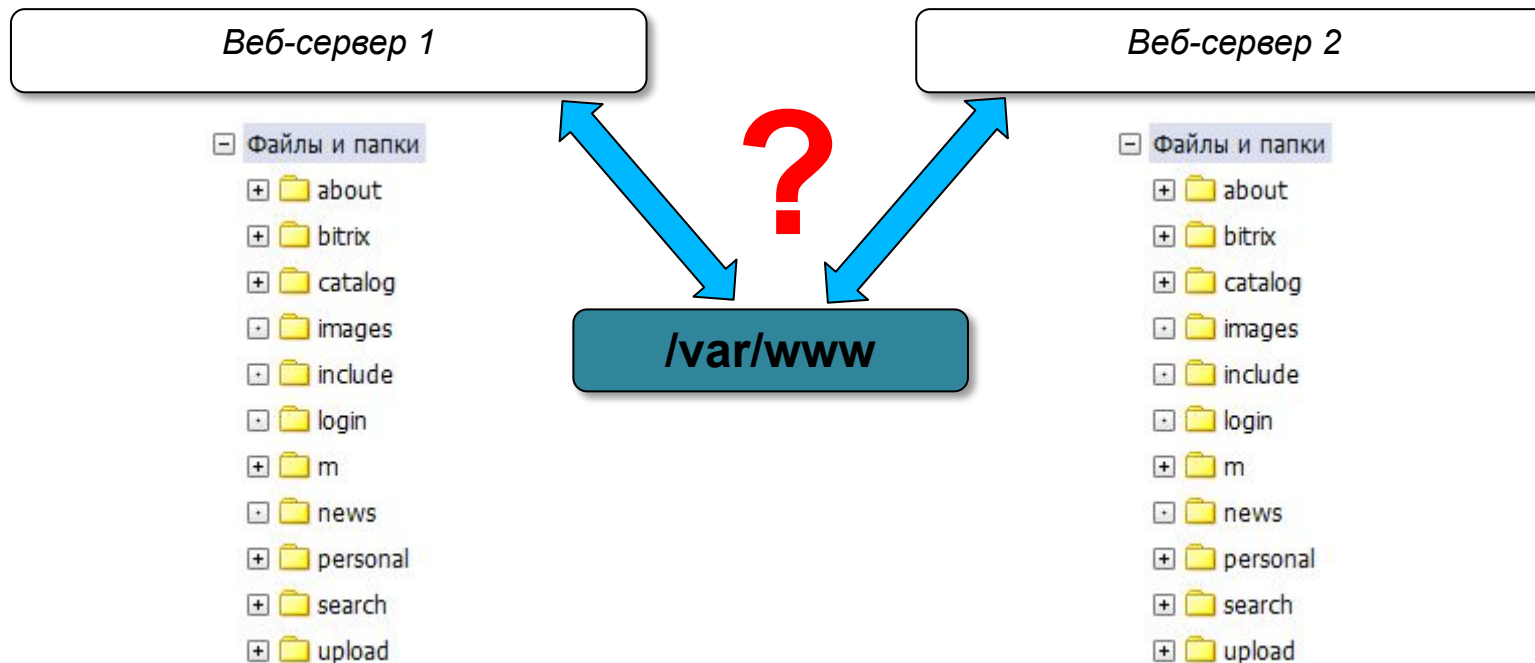


Задача: масштабирование при росте нагрузки





Задача синхронизации файлов





Синхронизация дисковых систем

Два типа:

1. Синхронный:

- Общая «дисковая полка» (дорого, не резервирует данные)
- Сетевые средства – NFS (очень медленно)
- OCFS2
- DRDB

2. Асинхронный (синхронизация локальных дисков)

- rsync
- csync2

Веб-сервера

Рабочий стол > Настройки > Веб-кластер > Веб-сервера

Добавить новый веб-сервер |
 Обновить |
 Настроить

	ID	Состояние	Статус	Название	Сервер
1	1	● время работы 4 часа	Server Version: Apache/2.2.3 (CentOS) Parent Server 0 Generation: Server uptime: 4 hours 12 minutes 35 seconds Total Traffic: 201.8 MB Total accesses: 10758 CPU Usage: u523.78 s46.4 cu0 cs0 - 3.76% CPU load	node 1	node1.demo-cluster.1c-bitrix.ru
2	2	● время работы 4 часа	Server Version: Apache/2.2.3 (CentOS) Parent Server 0 Generation: Server uptime: 4 hours 12 minutes 10 seconds Total Traffic: 5.2 MB Total accesses: 3633 CPU Usage: u93.64 s15.04 cu0 cs0 - .718% CPU load	node 2	node2.demo-cluster.1c-bitrix.ru

Выбрано: 2 Отмечено: 0

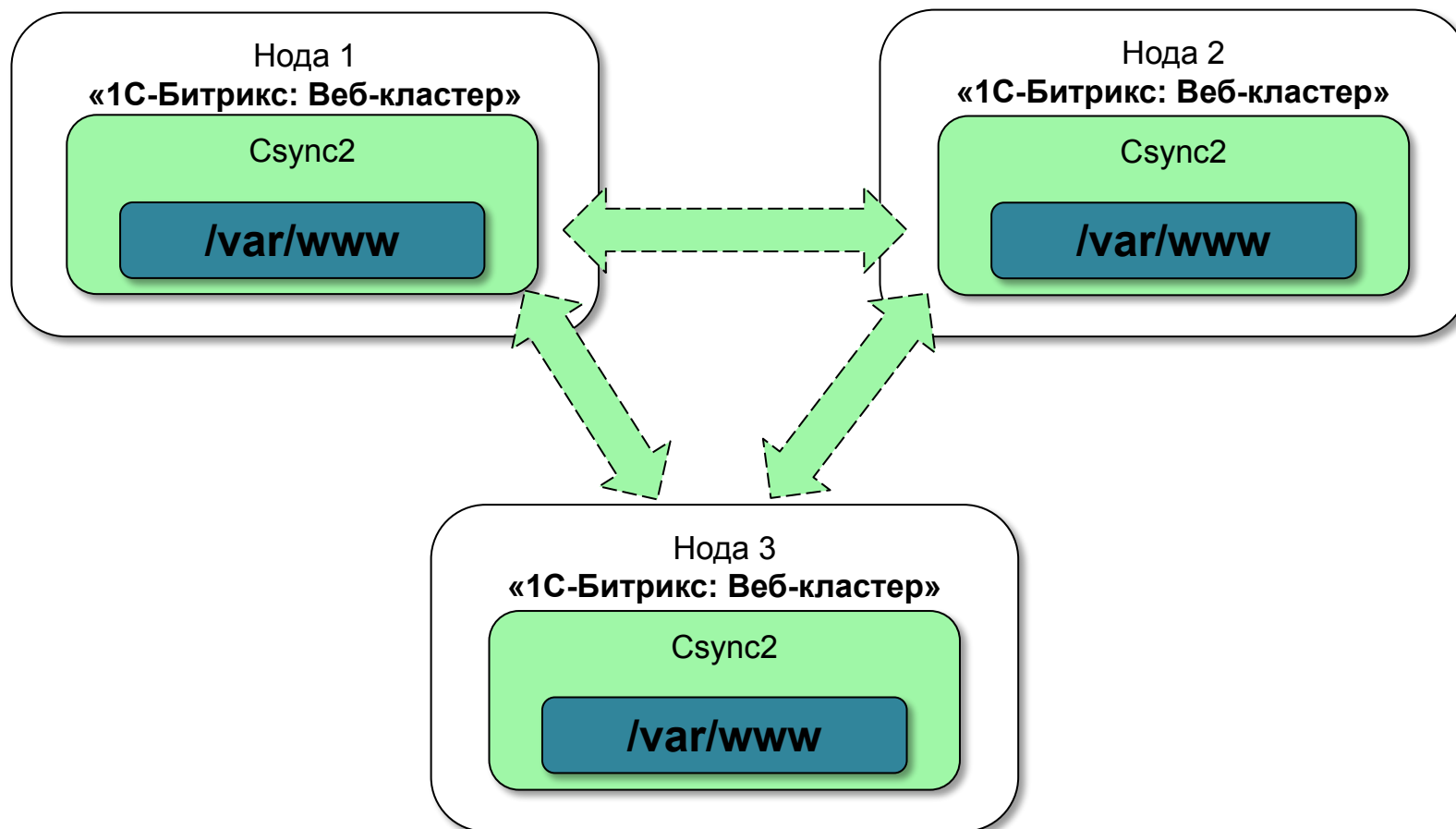


Почему мы выбрали csync2?

- **Быстрый доступ** к файлам приложения за счет использования локальных хранилищ.
- **Высокая скорость работы.**
- **Низкое потребление ресурсов** (CPU, дисковые операции). Два этих фактора позволяют запускать процесс синхронизации максимально часто, поэтому данные на серверах становятся идентичными практически в "реальном времени".
- **Простота настройки** для обмена данными между **любым количеством серверов.**
- **Возможность синхронизации удаления файлов.**
- **Защищенный обмен данными** между хостами (SSL).



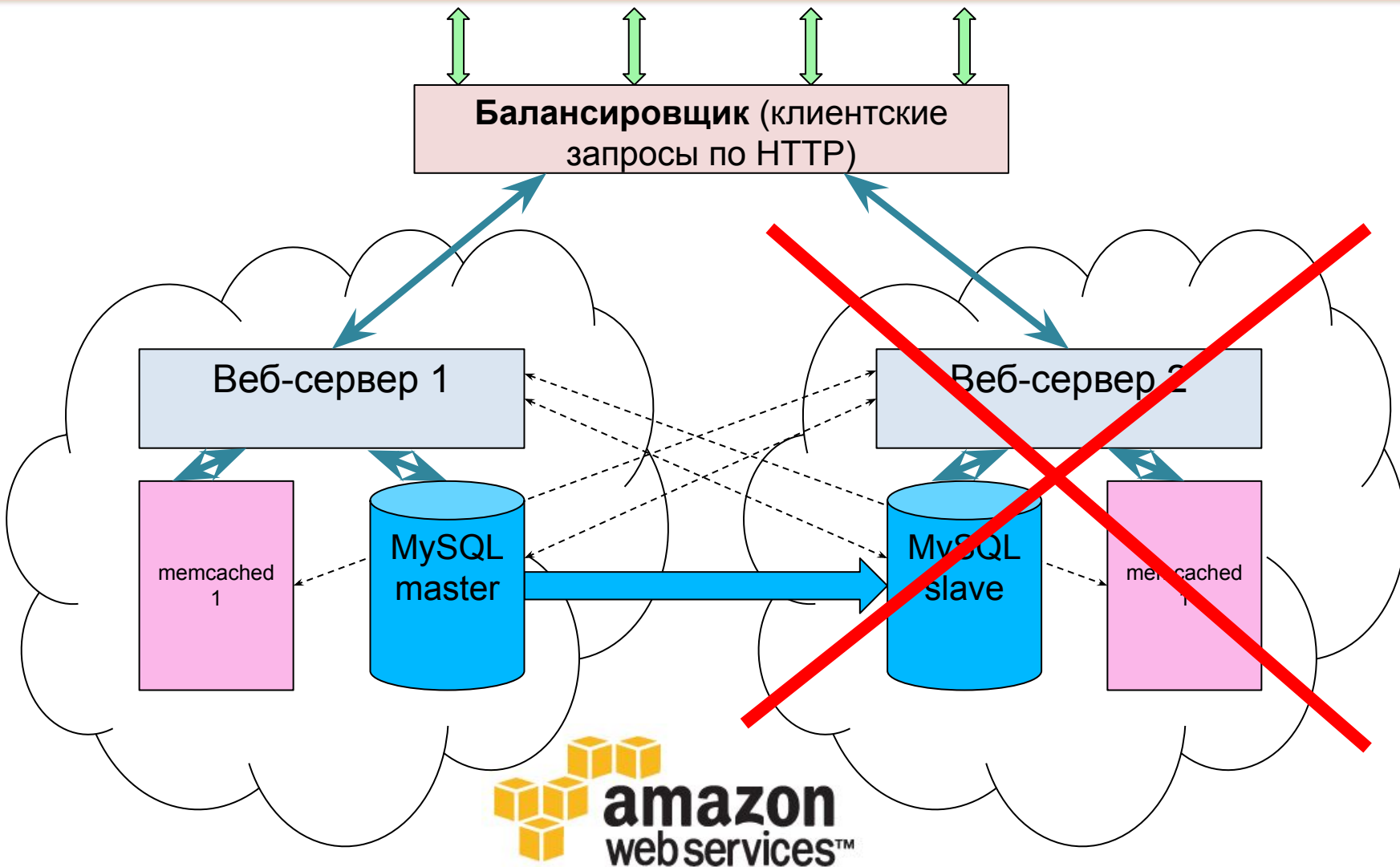
Тип 2: синхронизация локальных дисков





Способы балансирование нагрузки

- DNS сервер с несколькими записями типа A и разными IP адресами и коротким TTL
- NGINX на отдельном оборудовании
- Аппаратный маршрутизатор с балансированием нагрузки
- Балансировка силами дата центра (Amazon EC2)





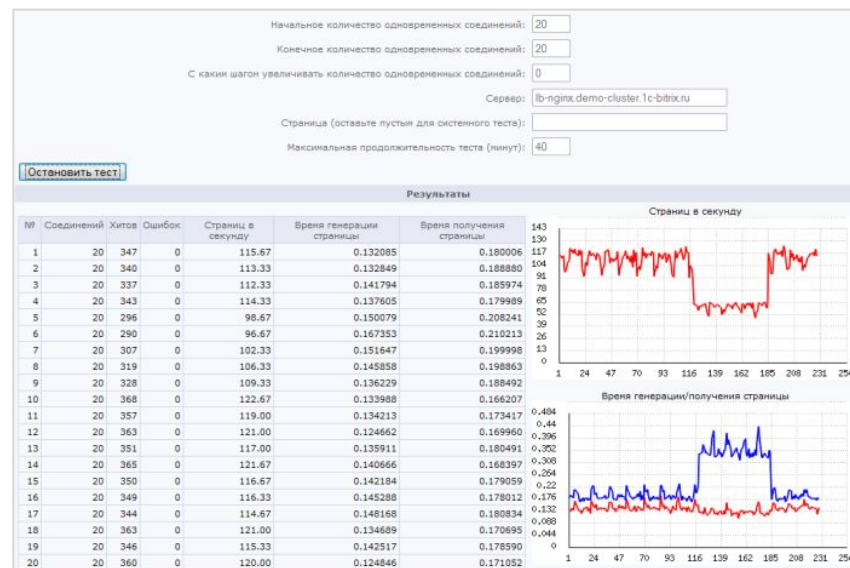
Устойчивость системы при выключении узлов веб-кластера

Нагрузочный тест – отключение одного из узлов кластера

При отключении узлов кластера система не прерывает обслуживание клиентов.

Увеличивается очередь (растет время отдачи страниц клиентам), однако в целом система сбалансирована по нагрузке.

Обратное добавление узла веб-кластера пропорционально увеличивает производительность системы.

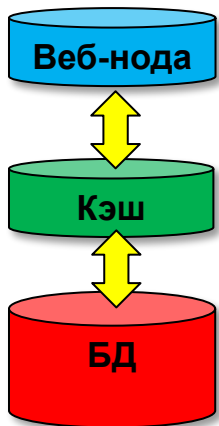




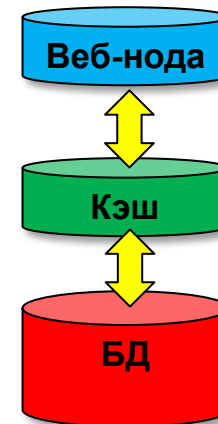
Мы работаем над...

круговой, асинхронной,
master-master
репликацией для обеспечения работы
географически распределенных веб-
кластеров 1С-Битрикс

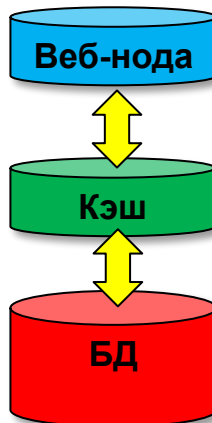
«1С-Битрикс: Веб-
кластер»,
ДЦ в Москве



«1С-Битрикс: Веб-
кластер»,
ДЦ в Новосибирске



«1С-Битрикс: Веб-кластер»,
ДЦ в Нью-Йорке

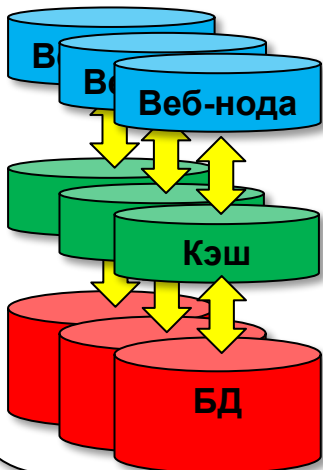




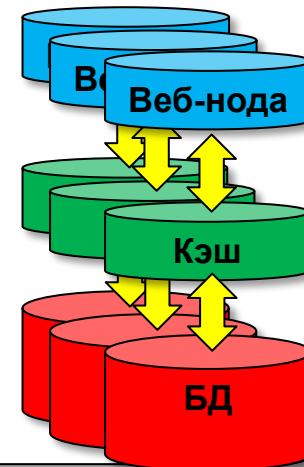
Мы работаем над...

круговой, асинхронной,
master-master
репликацией для обеспечения работы
географически **распределенных веб-кластеров** 1С-Битрикс

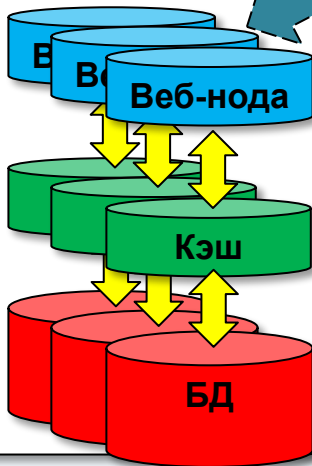
«1С-Битрикс: Веб-кластер»,
ДЦ в **Москве**



«1С-Битрикс: Веб-кластер»,
ДЦ в **Новосибирске**



«1С-Битрикс: Веб-кластер»,
ДЦ в **Нью-Йорке**





1С-БИТРИКС



СИСТЕМЫ УПРАВЛЕНИЯ ВЕБ-ПРОЕКТАМИ И КОРПОРАТИВНОЙ ИНФОРМАЦИЕЙ

Руководство по настройке и администрированию «1С-Битрикс: Веб-кластер»

http://www.1c-bitrix.ru/download/manuals/ru/web-cluster_guide.pdf



1С-БИТРИКС



СИСТЕМЫ УПРАВЛЕНИЯ ВЕБ-ПРОЕКТАМИ И КОРПОРАТИВНОЙ ИНФОРМАЦИЕЙ

Спасибо за внимание! Вопросы?