

Consul - инструмент динамической конфигурации и обнаружения сервисов

Alexey Parkhomenko
17/06/2017



<http://www.devconf.ru>

О чем поговорим?

- Service Discovery как сущности, что, зачем и для чего;

О чем поговорим?

- Service Discovery как сущности, что, зачем и для чего;
- Принципах архитектурного дизайна;

О чем поговорим?

- Service Discovery как сущности, что, зачем и для чего;
- Принципах архитектурного дизайна;
- Динамическом конфигурировании через Key-Value Store;

О чем поговорим?

- Service Discovery как сущности, что, зачем и для чего;
- Принципах архитектурного дизайна;
- Динамическом конфигурировании через Key-Value Store;
- Особенности использования Consul, багах, фичах;

Service Discovery, что и зачем?

- Service Discovery – это автоматическое обнаружение устройств и услуг, предлагаемых этими устройствами в компьютерной сети (из Википедии);



Service Discovery, что и зачем?

- Service Discovery – это автоматическое обнаружение устройств и услуг, предлагаемых этими устройствами в компьютерной сети (из Википедии);
- В контексте существенного количества продакшн серверов заметно облегчает жизнь системным администраторам и программистам;



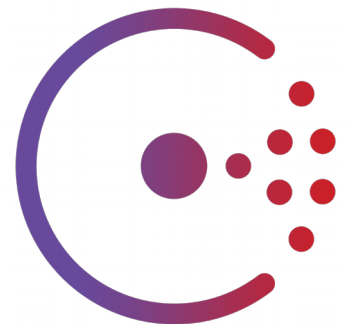
Service Discovery, что и зачем?

- Service Discovery – это автоматическое обнаружение устройств и услуг, предлагаемых этими устройствами в компьютерной сети (из Википедии);
- В контексте существенного количества продакшн серверов заметно облегчает жизнь системным администраторам и программистам;
- Как понятие существует с бородатых 90-х;



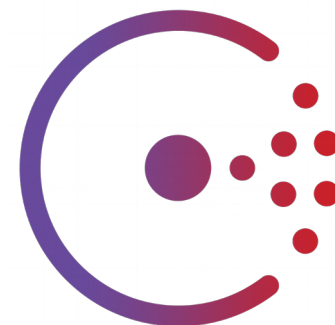
Consul и Service Discovery

- Предоставляет API для регистрации/удаления сервисов;



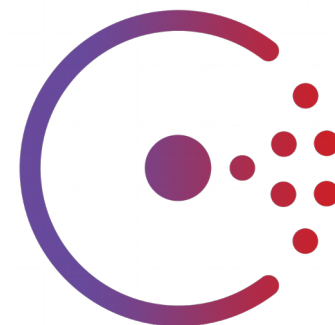
Consul и Service Discovery

- Предоставляет API для регистрации/удаления сервисов;
- Инструмент для управления настройками сервисов;



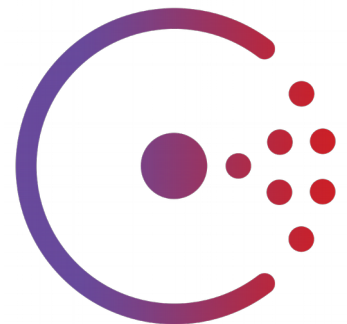
Consul и Service Discovery

- Предоставляет API для регистрации/удаления сервисов;
- Инструмент для управления настройками сервисов;
- Может работать с несколькими ДЦ;



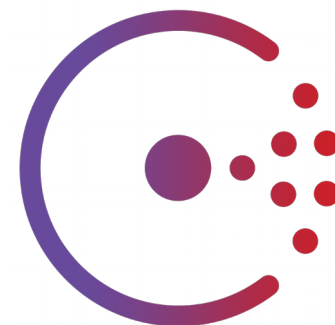
Consul и Service Discovery

- Предоставляет API для регистрации/удаления сервисов;
- Инструмент для управления настройками сервисов;
- Может работать с несколькими ДЦ;
- Проверять доступность сервисов;



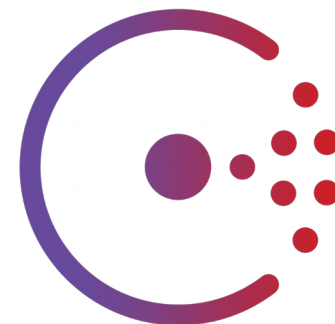
Consul и Service Discovery

- Предоставляет API для регистрации/удаления сервисов;
- Инструмент для управления настройками сервисов;
- Может работать с несколькими ДЦ;
- Проверять доступность сервисов;
- Выполнять какие-либо действия по событию;



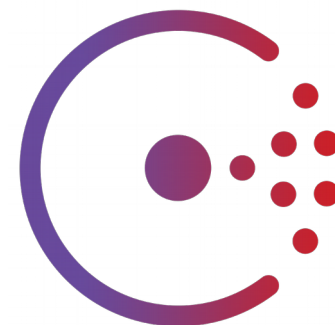
Consul и Service Discovery

- Предоставляет API для регистрации/удаления сервисов;
- Инструмент для управления настройками сервисов;
- Может работать с несколькими ДЦ;
- Проверять доступность сервисов;
- Выполнять какие-либо действия по событию;
- Имеет механизм распределенных блокировок;



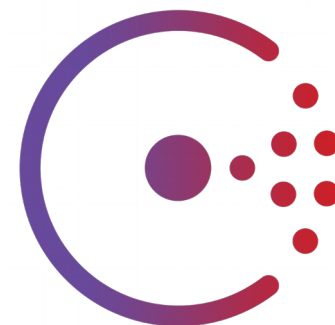
Consul Enterprise и Service Discovery

- Автоматические бекапы своего хранилища;



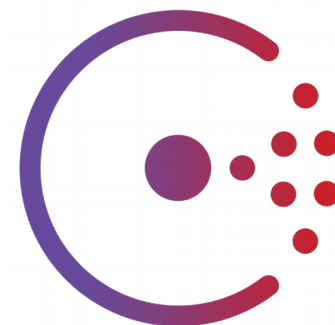
Consul Enterprise и Service Discovery

- Автоматические бекапы своего хранилища;
- Автоматический апгрейд версий;



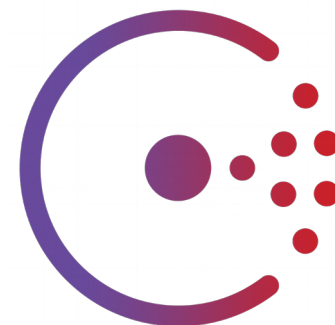
Consul Enterprise и Service Discovery

- Автоматические бекапы своего хранилища;
- Автоматический апгрейд версий;
- Enhanced Read Scalability;

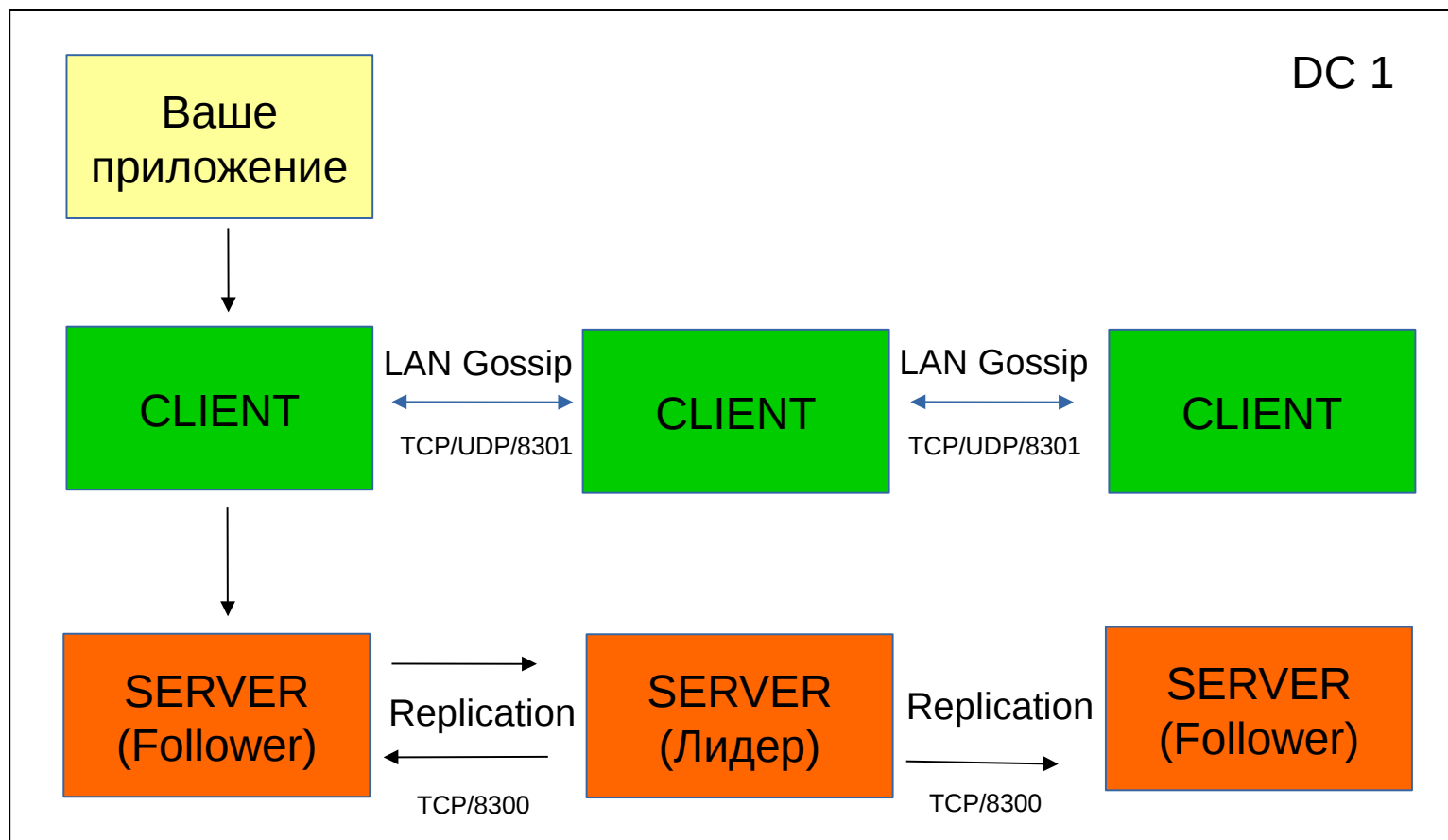


Consul Enterprise и Service Discovery

- Автоматические бекапы своего хранилища;
- Автоматический апгрейд версий;
- Enhanced Read Scalability;
- ... и другое;



Consul изнутри



Что нам все это дает?



Что нам все это дает?

- Состояние всего приложения;

Что нам все это дает?

- Состояние всего приложения;
- Масштабируемость;

Что нам все это дает?

- Состояние всего приложения;
- Масштабируемость;
- ... дофига работы по внедрению, отладке, тестированию кода;

Что нам все это дает?



Принципы архитектурного дизайна

- Одиночка - “как хочу, так и делаю”;

Принципы архитектурного дизайна

- Одиночка - “как хочу, так и делаю”;
- Команда - “как хотим, так и делаем”;

Принципы архитектурного дизайна

- Одиночка - “как хочу, так и делаю”;
- Команда - “как хотим, так и делаем”;
- Монолит с командными соглашениям;

Принципы архитектурного дизайна

- Одиночка - “как хочу, так и делаю”;
- Команда - “как хотим, так и делаем”;
- Монолит с командными соглашениям;
- Монолит с кучей команд и разных соглашений, флоу;

Принципы архитектурного дизайна

- Одиночка - “как хочу, так и делаю”;
- Команда - “как хотим, так и делаем”;
- Монолит с командными соглашениям;
- Монолит с кучей команд и разных соглашений, флоу;
- Сервисный монолит;

Принципы архитектурного дизайна

- Одиночка - “как хочу, так и делаю”;
- Команда - “как хотим, так и делаем”;
- Монолит с командными соглашениям;
- Монолит с кучей команд и разных соглашений, флоу;
- Сервисный монолит;
- SOA - “как хочу, так и делаю”;

Принципы архитектурного дизайна

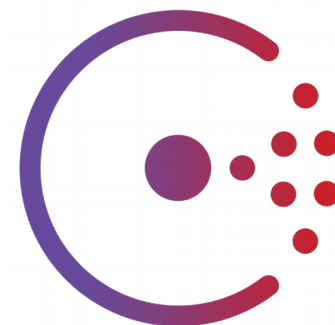
- Одиночка - “как хочу, так и делаю”;
- Команда - “как хотим, так и делаем”;
- Монолит с командными соглашениям;
- Монолит с кучей команд и разных соглашений, флоу;
- Сервисный монолит;
- SOA - “как хочу, так и делаю”;
- SOA с командными соглашениям;

Принципы архитектурного дизайна

- Одиночка - “как хочу, так и делаю”;
- Команда - “как хотим, так и делаем”;
- Монолит с командными соглашениям;
- Монолит с кучей команд и разных соглашений, флоу;
- Сервисный монолит;
- SOA - “как хочу, так и делаю”;
- SOA с командными соглашениям;
- SOA с кучей команд и разных соглашений, флоу;

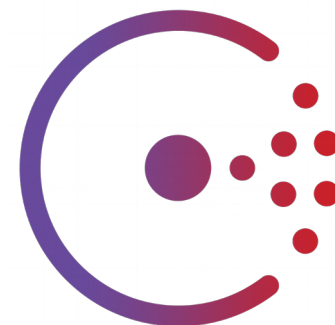
Как интегрировать Consul в это?

- В идеале – начать писать проект изначально с расчетом на Consul;



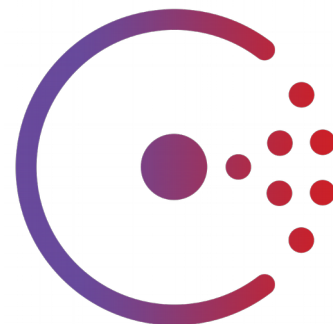
Как интегрировать Consul в это?

- В идеале – начать писать проект изначально с расчетом на Consul;
- В любом случае:
 1. Определить формат именования ключей для KV Store и контролировать объем их добавления;



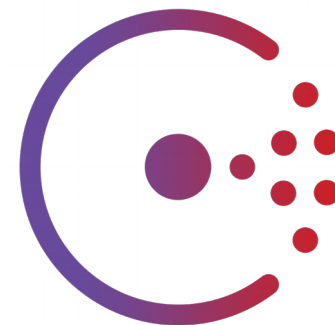
Как интегрировать Consul в это?

- В идеале – начать писать проект изначально с расчетом на Consul;
- В любом случае:
 1. Определить формат именования ключей для KV Store и контролировать объем их добавления;
 2. Вести документацию по добавляемым событиям;



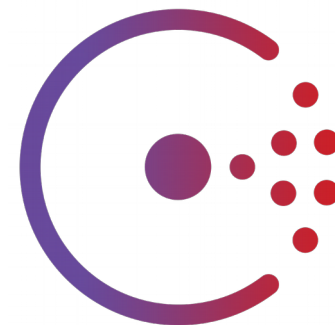
Как интегрировать Consul в это?

- В идеале – начать писать проект изначально с расчетом на Consul;
- В любом случае:
 1. Определить формат именования ключей для KV Store и контролировать объем их добавления;
 2. Вести документацию по добавляемым событиям;
 3. Определить стратегию реконнектов в случае отказа какой-то ноды;



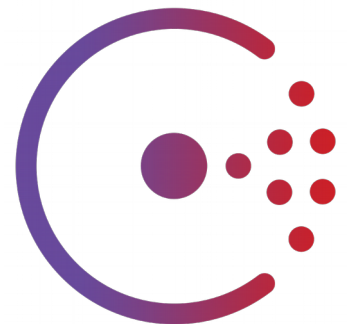
Как интегрировать Consul в это?

- В идеале – начать писать проект изначально с расчетом на Consul;
- В любом случае:
 1. Определить формат именования ключей для KV Store и контролировать объем их добавления;
 2. Вести документацию по добавляемым событиям;
 3. Определить стратегию реконнектов в случае отказа какой-то ноды;
 4. Обязательно использовать инструменты `consul-template`, `envconsul`, `vault`;



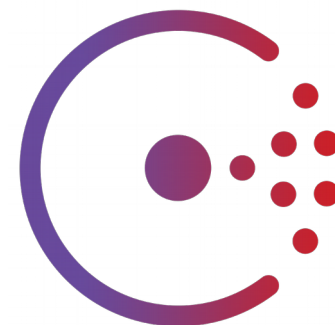
Что надо знать о Key-Value Store?

- Реально удобно менять настройки динамически, а не ждать пока проапдейтятся конфиг-файлы на серваках Puppet'ом или чем-то ещё;



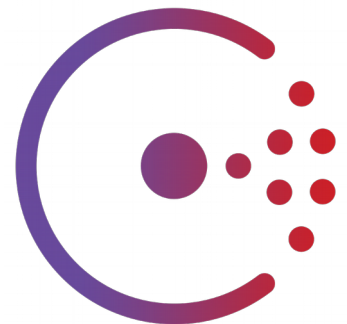
Что надо знать о Key-Value Store?

- Реально удобно менять настройки динамически, а не ждать пока проапдейтятся конфиг-файлы на серваках Puppet'ом или чем-то ещё;
- Это не киллер фича, можно юзать спокойно Redis если только для этих целей или ещё что-то;



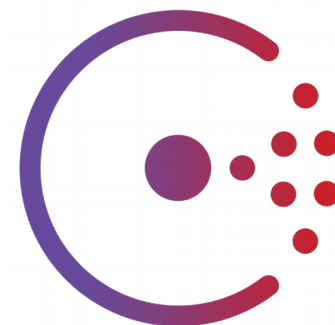
Что надо знать о Key-Value Store?

- Реально удобно менять настройки динамически, а не ждать пока проапдейтятся конфиг-файлы на серваках Puppet'ом или чем-то ещё;
- Это не киллер фича, можно юзать спокойно Redis если только для этих целей или ещё что-то;
- Крутая и опасная штука – подписываться на уведомление об изменении ключа или группы;



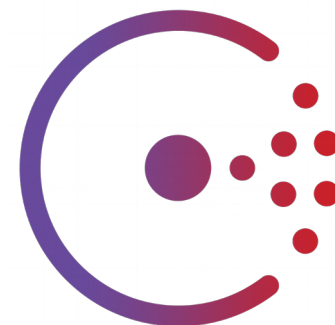
Особенности использования Consul

- Не смотря на богатый набор инструментов не заменяет необходимость держать отдельно полноценные мониторинговые системы;



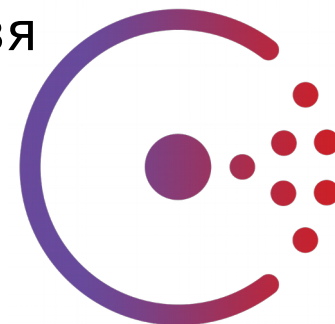
Особенности использования Consul

- Не смотря на богатый набор инструментов не заменяет необходимость держать отдельно полноценные мониторинговые системы;
- Если используется consul-template установите настройку `-max-stale`, чтобы данные брались не только с мастер-ноды;



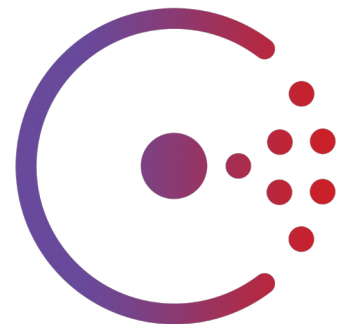
Особенности использования Consul

- Не смотря на богатый набор инструментов не заменяет необходимость держать отдельно полноценные мониторинговые системы;
- Если используется consul-template установите настройку `-max-stale`, чтобы данные брались не только с мастер-ноды;
- Если вы предполагаете в KV пушить много настроек – продумайте мониторинг, сейчас штатно нельзя узнать сколько ключей добавлено и сколько памяти они занимают;



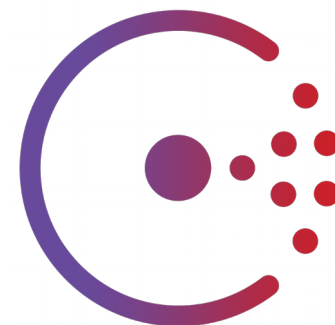
Особенности использования Consul

- Генерируемые события между агентами доставляются как Peer-To-Peer и их доставка не гарантируется в отличие от записи в KV;



Особенности использования Consul

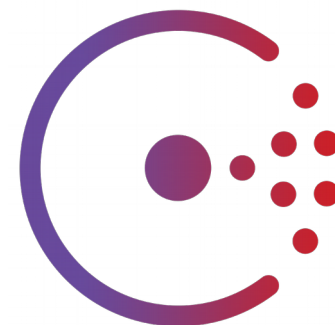
- Генерируемые события между агентами доставляются как Peer-To-Peer и их доставка не гарантируется в отличие от записи в KV;
- Согласно CAP теореме при проектировании архитектуры консул выбрал CP, поэтому штатная ситуация затыпы при пуше на большом количестве нод в кворуме;



Особенности использования Consul

- Генерируемые события между агентами доставляются как Peer-To-Peer и их доставка не гарантируется в отличии от записи в KV;
- Согласно CAP теореме при проектировании архитектуры консул выбрал CP, поэтому штатная ситуация затыпы при пуше на большом количестве нод в кворуме;

(В Enterprise решается через Enhanced Read Scalability, путем отключения обязательного согласования)



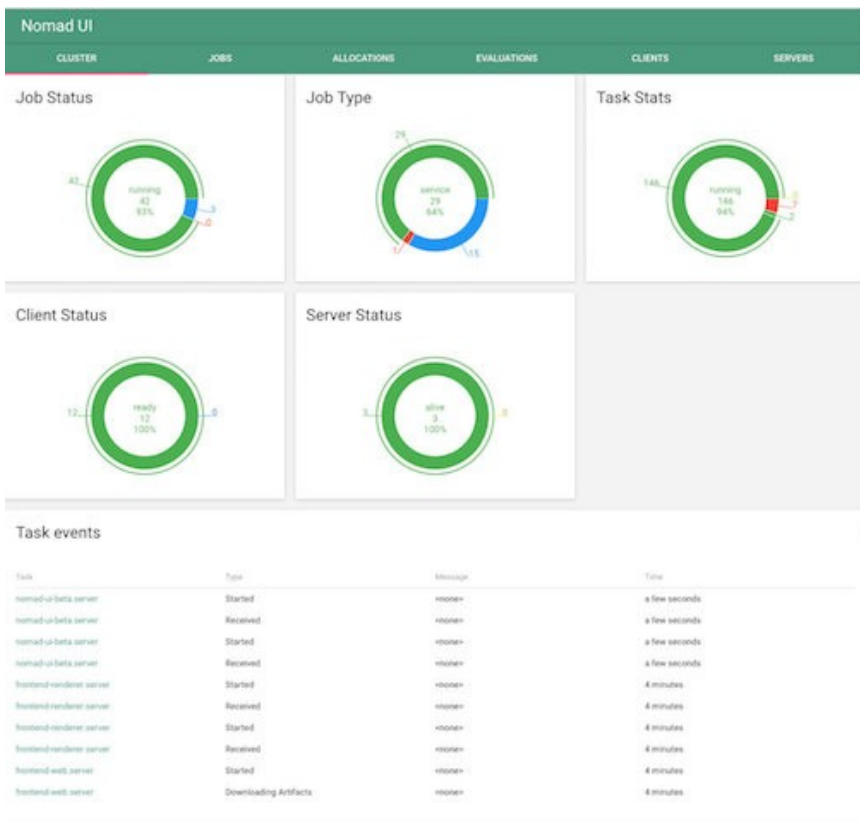
Какие подобные решения есть?

- Etcd
- Apache Zookeeper
- Doozerd
- SkyDNS
- Eureka

... и т.д.



Модные UI для Consul и Vault



VAULT - UI

Enter username

Enter password

<https://github.com/jippi/hashui>

<https://github.com/djenriquez/vault-ui>

<http://www.devconf.ru>

Спасибо. Вопросы?

