

Масштабирование брокеров сообщений.

Представим, что наше приложение развивается и нам не хватает мощностей. Мы ранее рассматривали что мощности может не хватать у продюсера/консьюмера и запускали дополнительные инстансы сервисов. Сейчас мы рассмотрим как масштабировать сам брокер сообщений, потому что он тоже ограничен в своей скорости работы и при большом количестве сообщений может требовать доп.ресурсов. Также дополнительные инстансы брокера запускают для отказоустойчивых систем.

Сначала остановимся подробнее на причинах масштабирования.

Что будет, если наш брокер перестанет работать?

- Продюсер не сможет записать сообщения
- Консьюмер не сможет читать сообщения
- Возможна потеря сообщений, если брокер не записывал данные на диск

Что будет, если брокер будет сильно загружен?

- Возможны задержки в обработке сообщений
- Возможны задержки в отправке сообщений (push модели)
- Снижение пропускной способности

Как видите, межсервисное взаимодействие полностью зависит от работы брокера. Чтобы данные передавались быстро и система была отказоустойчивой, необходимо задумываться о масштабировании.

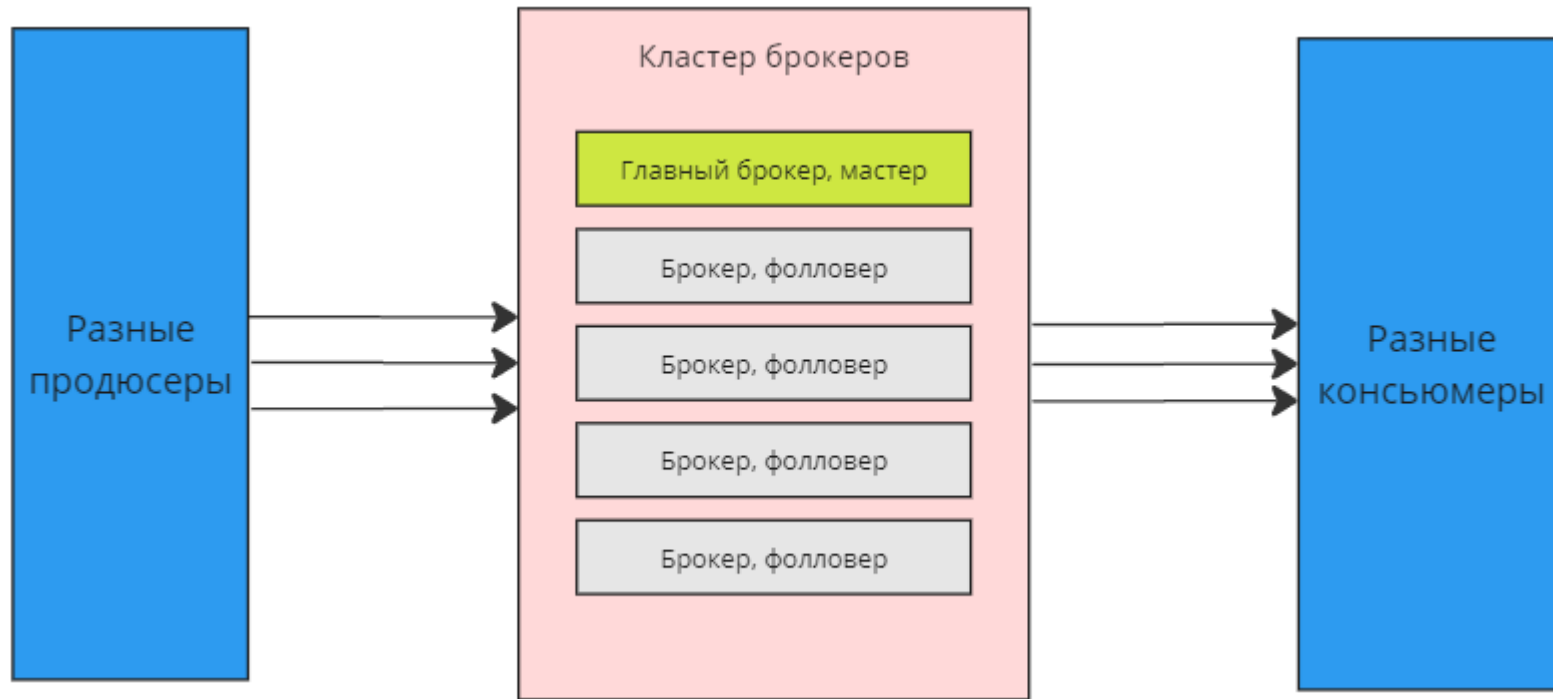
Первый способ - кластеризация.

Введем новые термины:

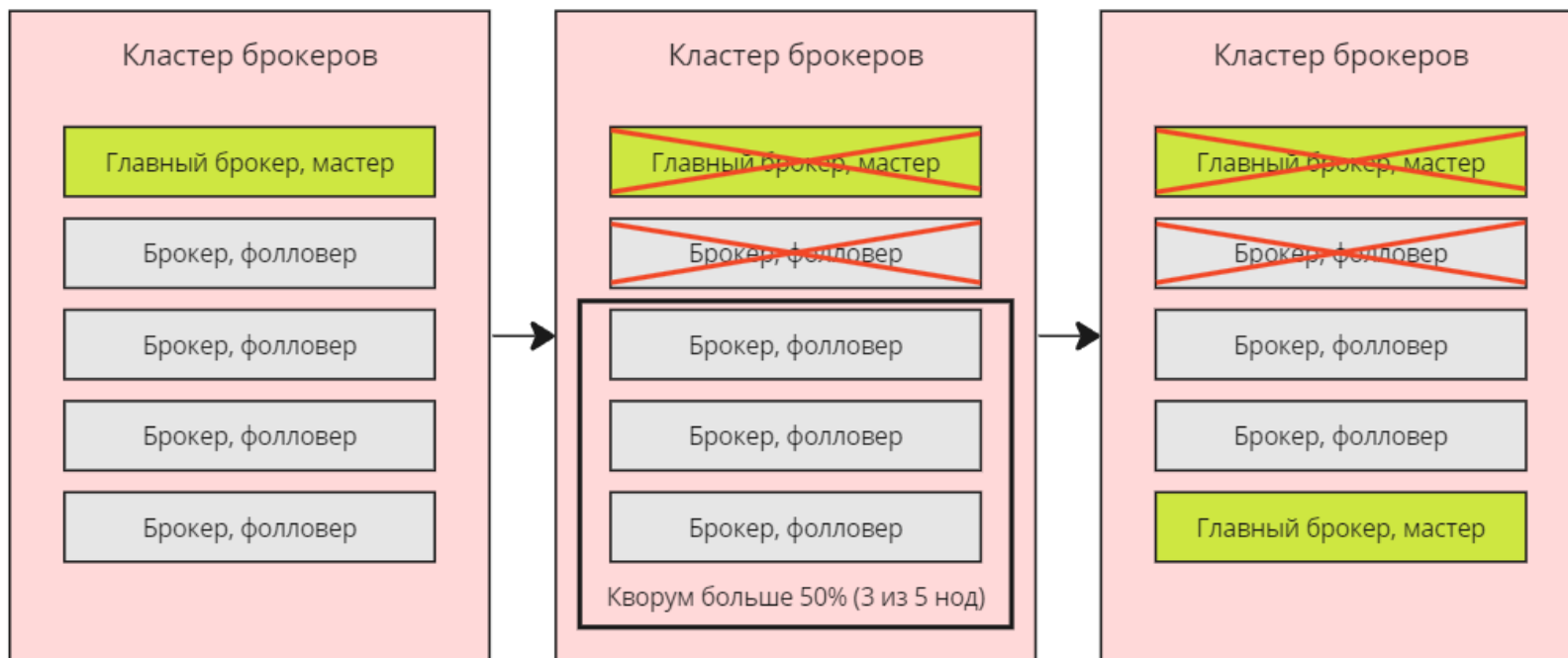
1. Кластер - объединение нескольких идентичных сервисов, инстансов (или нод), в данном случае кластер брокеров это группа идентичных брокеров.

2. Лидер или мастер - главный инстанс, который управляет кластером. Может тоже участвовать в обработке запросов, в зависимости от настроек.
3. Фолловер (follower или slave) - обычный инстанс, который обрабатывает запросы.
4. Перевыбор - процесс голосования между инстансами для выбора нового лидера. Допустим, если лидер "упал" и недоступен.
5. Кворум - необходимое число инстансов для того, чтобы перевыбор состоялся. Что важно запомнить - перевыборы не состоятся, если кворум менее 50% от числа всех инстансов. Поэтому обычно в кластере делают нечетное количество инстансов.

Схематично работа с кластером брокеров будет выглядеть так:



Разные продюсеры передают данные и разные консьюмеры их получают. Внутри кластеры инстансы общаются между собой, поддерживая эффективную работу кластера. Перевыбор схематично будет выглядеть так:



В первом состоянии все в порядке, все инстансы работают. Но в следующем состоянии недоступен мастер и еще один фолловер (отключились по тех.причинам). Начинается перевыбор, т.к. кворума достаточно, и в третьем состоянии получаем нового мастера (который выбран с помощью алгоритмов брокера).

Детали данной концепции:

1. Все брокеры в кластере сильно связаны между собой. Поэтому кластер нужно размещать на одном сервере, чтобы избежать задержек во взаимодействии.
2. Передавать данные продюсер может в любой инстанс, если нет других настроек (возможны разные настройки, в зависимости от брокера). Мы это не отразили на схеме, чтобы пока образно понять как работает данный шаблон масштабирования и сравнить его с другим шаблоном.
3. Возможна репликация данных из одного инстанса в другие. Это значит, что брокер, получивший данные, не только их обработает но и произведет репликацию в другие инстансы.

Плюсы данной концепции:

- При падении инстанса/части инстансов кластер остается рабочим, получаем отказоустойчивость
- Повышаем производительность брокера (увеличиваем пропускную способность)

Минусы:

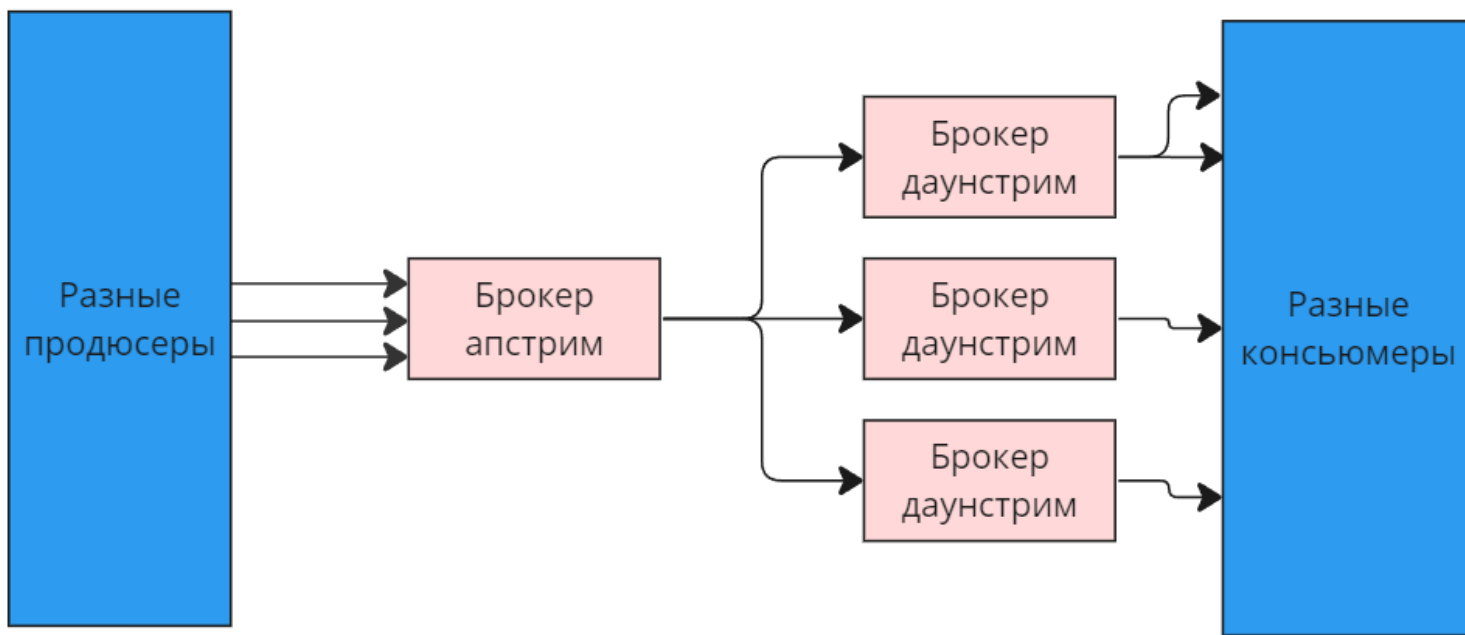
- Сложная настройка и поддержка
- Когда происходят перевыборы брокер теряет производительность
- Инстансы брокеров сильно связаны, надо отслеживать синхронизацию данных

Второй способ - федерация.

Введем новые термины:

1. Федерация - объединение нескольких идентичных сервисов, инстансов (или нод), аналогично кластеру, но в слабосвязную систему.
2. Апстрим - брокер, в который поступают сообщения от продюсеров. Стоит "первый" в очереди.
3. Даунстрим - брокеры, к которым попадают данные из апстрима. Передают данные консьюмерам. Стоят после апстрима.

Схематично работа федерации будет выглядеть так:



Есть разные продюсеры, не важно какие и сколько, они передают данные в апстрим. Апстрим передает данные в даунстримы, которые в свою очередь передают или позволяют забирать сообщения разным консьюмерам. На схеме приведен "чистый" вид федерации, но например к любому даунстриму может подключиться какой-либо продюсер для записи данных.

В чем же отличие по сравнению с кластеризацией?

Если вы заметите, то дополнительные брокеры не зависят от главного, к ним может прийти даже отдельный продюсер и начать передавать данные. Это и есть главное отличие. Каждый инстанс работает независимо, и также настраивается, независимо. Нет главного и "подопечных". Таким образом мы уменьшаем связность инстансов между собой.

Плюсы данной концепции:

- Можно легко распределить нагрузку согласно нашей бизнес-логике (например, мы знаем, что какой-либо продюсер может отправлять большое количество сообщений, тогда из апстрима передаем его сообщения на два даунстрима и более, а сообщения других продюсеров передаем на третий даунстрим)
- Более простая настройка
- Нет перевыборов

- Слабая связность, брокеры могут иметь своих продюсеров и свои данные, очереди или топики

Минусы:

- При падении апстрима может встать вся работа системы
- Настройка более простая, но каждый брокер надо настраивать отдельно (требуется больше времени)
- Сложнее обнаруживать ошибки, надо анализировать всю цепочку событий (федерации могут быть огромными)

Чтобы выбрать паттерн масштабирования, можете для начала ориентироваться на эти вопросы:

1. В системе брокеров должна быть независимость (настройки, данные)?

1. Да (федерация)

2. Нет (кластеризация)

2. Данные в системе брокеров должны быть согласованы? (или у каждого компонента могут быть свои данные)

1. Да (кластеризация)

2. Нет (федерация)

3. Могут потребоваться индивидуальные клиенты (продюсеры) или потребители (консьюмеры) для некоторых инстансов?

1. Да (федерация)

2. Нет (кластеризация)

Подчеркнем, что нет "лучшего" решения, все зависит от требований к вашей системе. Оба паттерна используются с целью повышения производительности и отказоустойчивости. Ключевое отличие шаблонов масштабирования:

- Федерация подразумевает независимость брокеров, сотрудничество и обмен данными по необходимости. Могут быть свои клиенты и потребители
- Кластеризация, напротив, объединяет брокеры в единую сущность, у которых цель - это согласованность данных и эффективная работа для одних и те же продюсеров и консьюмеров