

Текстовая расшифровка видео:

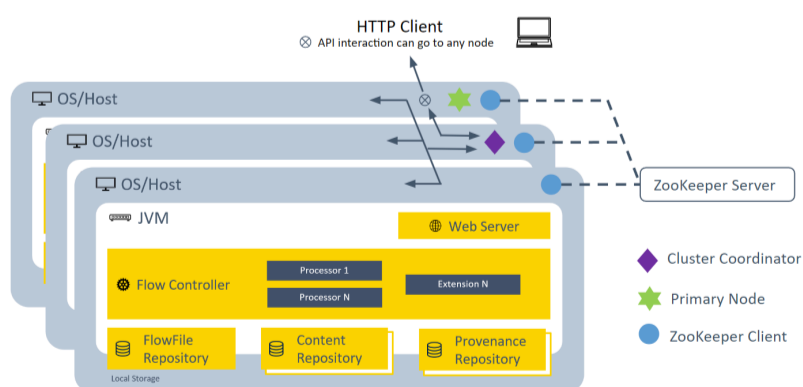
## РЕЖИМ КЛАСТЕРА

План:

- Режим кластера;
- Разбор примера на консоли;
- О режиме кластера.

### Режим кластера

NiFi может работать в режиме кластера. На данном изображении представлены три ноды, которые координируются программой «ZooKeeper»:



**ZooKeeper** – это открытая служба, координирующая распределенные системы и основанная на резидентных базах данных класса «ключ-значение». ZooKeeper назначает Primary Node и координатора.

Координатор отвечает за сбор данных о состоянии других нод и за подключение/отключение их от кластера. По HTTP к UI или Rest API можно обратиться к любой ноде.

**Общие сведения:**

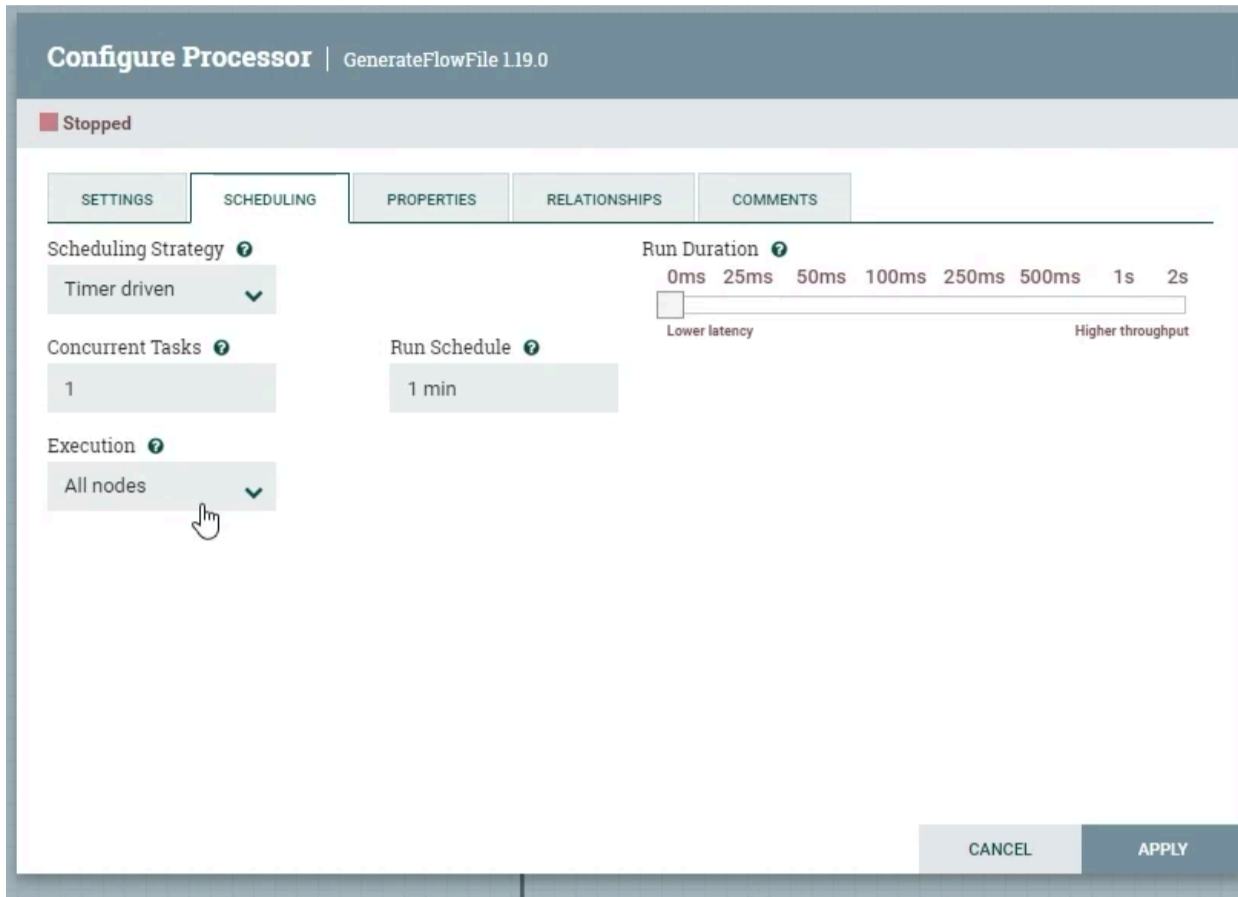
- Работа в режиме кластера обеспечивает отказоустойчивость и дает возможность горизонтально масштабировать нагрузку.



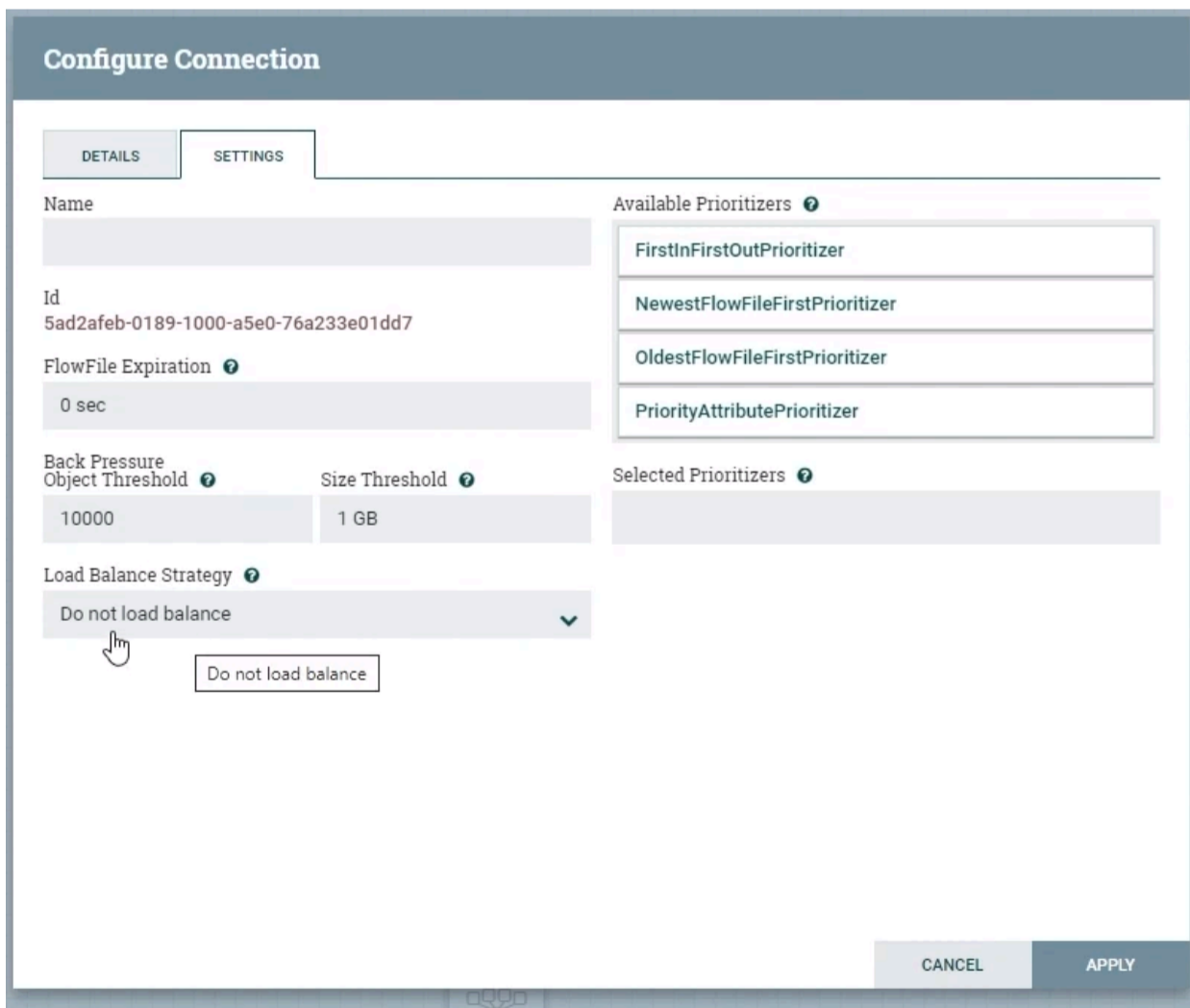
- Статично зафиксированной мастер-ноды нет. Вместо этого Apache Zookeeper выбирает одну ноду в качестве координатора и одну в качестве primary.
- Primary-нода для запуска изолированных процессоров, которые не должны запускаться на всех нодах одновременно.

### Разбор примера на консоли

У нас есть процессор, который стоит вначале потока:



В его свойствах (в разделе «Scheduling») мы можем выбрать в Execution либо «All nodes», либо «Primary Node». На кластере выбираем «Primary Node» (на single server влияния не оказывает). По дефолту выбираем «All nodes». После этого поток стартанет на первой ноду, и следующий процессор после настройки сделает запрос к базе данных только с первой ноды, после чего данные попадут в очередь:



В свойствах очереди мы можем распределить нагрузку по нодам через балансировщик:

У нас есть выбор:

- Выбрать без балансировки;
- Сделать балансировку, основываясь на атрибуте (на его значении);
- Выбрать Round robin, который поможет случайным образом распределить потоки между нодами (Single node означает, что балансировка не будет произведена, данные пойдут сразу на Primary Node).

### О режиме кластера

Если вы работаете с кластером, то в глобальном меню появится пункт:

#### NiFi Cluster

NODES	SYSTEM	JVM	FLOWFILE STORAGE	CONTENT STORAGE	PROVENANCE STORAGE	VERSIONS
Displaying 2 of 2						
Displaying 2 of 2						
Filter	by address					
Node Address	Content Repository	Total Space	Used Space	Free Space	Utilization	
192.168.182.16:8081	repo0	45,03 GB	14,32 GB	30,71 GB	32.0%	
192.168.182.15:8081	repo0	45,03 GB	16,48 GB	28,55 GB	37.0%	

Через него вы можете попасть в окно, где видны все ресурсы кластера:

- С точки зрения нод можно посмотреть их расположение в системе, в каком статусе находятся, активные они или неактивные, кто из них primary/координатор.
- С точки зрения системы можно посмотреть на какой системе расположен кластер.
- С точки зрения Java-виртуальной машины можно посмотреть количество потребляемой нодой памяти.
- С точки зрения репозитория можно посмотреть сколько дискового пространства забирает каждая нода.

**Важно:** следите за тем, чтобы место на дисках не закончилось.

Как вам урок?



Изучил, далее >

