

# TPS

1 ~ 60 1 .

, TPS TPS

 TPS

. 1 10TPS 10.

~ 1 .

```
private static final Executor executor = Executors.newFixedThreadPool(450);

private static CompletionStage<String> callApiAsync(Integer param) {
    // CompletableFuture
    return CompletableFuture.supplyAsync(() -> {
        try {

            double dValue = Math.random();
            int iValue = (int)(dValue * 1000);
            Thread.sleep(iValue); // API
            tpsActor.tell("CompletedEvent", ActorRef.noSender()); // TPS
        } catch (InterruptedException e) {
            Thread.currentThread().interrupt();
        }
        return "Response for " + param;
    }, executor);
}
```

TPS AkkaStream .

sink .

```

public void ThrottleTest(int tps, int testCount ) {
    new TestKit(actorSystem) {
        {
            final ActorMaterializerSettings settings = ActorMaterializerSettings.create(actorSystem)
                .withDispatcher("my-dispatcher-streamtest");

            final Materializer materializer = ActorMaterializer.create(settings, actorSystem);
            final TestKit probe = new TestKit(actorSystem);

            tpsActor = actorSystem.actorOf(TpsMeasurementActor.Props(), "TpsActor");
            tpsActor.tell(probe.getRef(), getRef());
            expectMsg(Duration.ofSeconds(1), "done");

            // Source
            Source<Integer, NotUsed> source = Source.range(1, testCount);

            // Flow
            final int parallelism = 450;
            Flow<Integer, String, NotUsed> parallelFlow = Flow.<Integer>create()
                .mapAsync(parallelism, BackPressureTest::callApiAsync);

            // Buffer OverflowStrategy.backpressure
            int bufferSize = 100000;
            Flow<Integer, Integer, NotUsed> backpressureFlow = Flow.<Integer>create()
                .buffer(bufferSize, OverflowStrategy.backpressure());

            AtomicInteger processedCount = new AtomicInteger();

            // Sink
            Sink<String, CompletionStage<Done>> sink = Sink.foreach(s -> {
                //
                processedCount.getAndIncrement();
                if(processedCount.getAcquire() % 10 == 0) {
                    //System.out.println("Processed 10");
                }
            });

            System.out.println("Run backpressureFlow bufferSize:"+bufferSize);

            // RunnableGraph
            source.via(backpressureFlow)
                .throttle(tps, FiniteDuration.create(1, TimeUnit.SECONDS), tps, (ThrottleMode)
ThrottleMode.shaping())
                .via(parallelFlow)
                .to(sink)
                .run(materializer);

            within(
                Duration.ofSeconds(15),
                () -> {
                    // Will wait for the rest of the 10 seconds
                    expectNoMessage(Duration.ofSeconds(10));
                    return null;
                });
        }
    };
}

```

```

my-dispatcher-streamtest {
    type = Dispatcher
    executor = "thread-pool-executor"
    thread-pool-executor {
        fixed-pool-size = 50
    }
    throughput = 1
}

```

```
no usages  psmon
222  @Test
223  @DisplayName("ThrottleTest 200")
224  public void ThrottleTest200(){
225      ThrottleTest( tps: 200, testCount: 5000);
226  }
227
no usages  psmon
228  @Test
229  @DisplayName("ThrottleTest 450")
230  public void ThrottleTest450(){
231      ThrottleTest( tps: 450, testCount: 5000);

```

BackPressureTest.ThrottleTest200

Tests passed: 1 of 1 test - 10 sec 167 ms

BackPressureTest (com.webnori.springweb.akka.stream) 10 sec 167 ms

ThrottleTest200 10 sec 167 ms

Run backpressureFlow bufferSize:100000

```
[INFO ] [2023-12-15 15:18:35,257] [ClusterSystem-akka.actor.default-dispatcher-4] [First Tick]
[INFO ] [2023-12-15 15:18:36,275] [ClusterSystem-akka.actor.default-dispatcher-4] [TPS:376.0]
[INFO ] [2023-12-15 15:18:37,263] [ClusterSystem-akka.actor.default-dispatcher-4] [TPS:189.0]
[INFO ] [2023-12-15 15:18:38,260] [ClusterSystem-akka.actor.default-dispatcher-4] [TPS:207.0]
[INFO ] [2023-12-15 15:18:39,271] [ClusterSystem-akka.actor.default-dispatcher-4] [TPS:201.0]
[INFO ] [2023-12-15 15:18:40,269] [ClusterSystem-akka.actor.default-dispatcher-4] [TPS:204.0]
[INFO ] [2023-12-15 15:18:41,267] [ClusterSystem-akka.actor.default-dispatcher-4] [TPS:199.0]
[INFO ] [2023-12-15 15:18:42,280] [ClusterSystem-akka.actor.default-dispatcher-5] [TPS:199.0]
[INFO ] [2023-12-15 15:18:43,268] [ClusterSystem-akka.actor.default-dispatcher-4] [TPS:196.0]
[INFO ] [2023-12-15 15:18:44,277] [ClusterSystem-akka.actor.default-dispatcher-4] [TPS:210.0]
```

TPS200 ~ .

- API TPS .

: <https://github.com/psmon/java-labs/blob/master/springweb/src/test/java/com/webnori/springweb/akka/stream/BackPressureTest.java#L234>

Next : - API