

# Systemy rozproszone - laboratorium 4

Łukasz Jezapkowicz - 305313

16 kwietnia 2021

# 1 Laboratorium 4

Ten dokument zawiera odpowiedzi do zadań z laboratorium 4 z Systemów Rozproszonych.

## 1.1 Zadanie 1 - Porównaj działanie strategii obsługi błędów stop/resume/restart

### 1.1.1 Stop

```
Z1 main: started
[actorMath-akka.actor.default-dispatcher-6] INFO akka.event.slf4j.Slf4jLogger - Slf4jLogger started
Z1 main: actor system ready
actorMultiply: operation count = 1
actorDivide: operation count = 1
actorMultiply: operation count = 2
actorDivide: operation count = 2
[actorMath-akka.actor.default-dispatcher-7] ERROR akka.actor.typed.Behavior$ - Supervisor StopSupervisor saw failure: / by zero
java.lang.ArithmeticException: Create breakpoint: / by zero
    at Z1.MathActorDivide.onMathCommandDivide(MathActorDivide.java:33)
    at akka.actor.typed.javadsl.BuiltReceive.receive(ReceiveBuilder.scala:213)
    at akka.actor.typed.javadsl.BuiltReceive.receiveMessage(ReceiveBuilder.scala:204)
    at akka.actor.typed.javadsl.Receive.receive(Receive.scala:53)
    at akka.actor.typed.javadsl.AbstractBehavior.receive(AbstractBehavior.scala:64)
    at akka.actor.typed.Behavior$.interpret(Behavior.scala:274)
    at akka.actor.typed.Behavior$.interpretMessage(Behavior.scala:230)
    at akka.actor.typed.internal.InterceptorImpl$$anon$2.apply(InterceptorImpl.scala:57)
    at akka.actor.typed.internal.SimpleSupervisor.aroundReceive(Supervision.scala:124)
    at akka.actor.typed.internal.InterceptorImpl.receive(InterceptorImpl.scala:85)
    at akka.actor.typed.Behavior$.interpret(Behavior.scala:274)
    at akka.actor.typed.Behavior$.interpretMessage(Behavior.scala:230)
    at akka.actor.typed.internal.adapter.ActorAdapter.handleMessage(ActorAdapter.scala:131)
    at akka.actor.typed.internal.adapter.ActorAdapter.aroundReceive(ActorAdapter.scala:107)
    at akka.actor.ActorCell.receiveMessage(ActorCell.scala:579)
    at akka.actor.ActorCell.invoke(ActorCell.scala:547)
    at akka.dispatch.Mailbox.processMailbox(Mailbox.scala:270)
    at akka.dispatch.Mailbox.run(Mailbox.scala:231)
    at akka.dispatch.Mailbox.exec(Mailbox.scala:243) <5 internal calls>
Z1 main: sending second package of messages
Z1 main: messages send
actorMultiply: operation count = 3
actorDivide: operation count = 4
[actorMath-akka.actor.default-dispatcher-7] INFO akka.actor.LocalActorRef - Message [Z1.MathActor$MathCommandDivide] to Actor[akka://actorMath/user/actorDivide#-922094238] was not delivered. [1] dead letters encountered. If this is not an expected behavior then Actor[akka://actorMath/user/actorDivide#-922094238] may have terminated unexpectedly. This logging can be turned off or adjusted with configuration settings 'akka.log-dead-letters' and 'akka.log-dead-letters-during-shutdown'.
[actorMath-akka.actor.default-dispatcher-7] INFO akka.actor.LocalActorRef - Message [Z1.MathActor$MathCommandDivide] to Actor[akka://actorMath/user/actorDivide#-922094238] was not delivered. [2] dead letters encountered. If this is not an expected behavior then Actor[akka://actorMath/user/actorDivide#-922094238] may have terminated unexpectedly. This logging can be turned off or adjusted with configuration settings 'akka.log-dead-letters' and 'akka.log-dead-letters-during-shutdown'.
```

Jak widać, dla strategii **Stop** działanie aktora actorDivide zostało przerwane (actorMultiply dalej działa). Potwierdzenie tego widzimy w dwóch wiadomościach z Loggera mówiących, że wiadomość nie doszła do głównego aktora oraz sugestii, że działanie aktora mogło zostać niespodziewanie zakończone.

### 1.1.2 Resume

```
Z1 main: started
[actorMath-akka.actor.default-dispatcher-6] INFO akka.event.slf4j.Slf4jLogger - Slf4jLogger started
Z1 main: actor system ready
actorDivide: operation count = 1
actorMultiply: operation count = 1
actorDivide: operation count = 2
actorMultiply: operation count = 2
[actorMath-akka.actor.default-dispatcher-7] ERROR akka.actor.typed.Behavior$ - Supervisor ResumeSupervisor saw failure: / by zero
java.lang.ArithmeticException: Create breakpoint: / by zero
    at Z1.MathActorDivide.onMathCommandDivide(MathActorDivide.java:33)
    at akka.actor.typed.javadsl.BuiltReceive.receive(ReceiveBuilder.scala:213)
    at akka.actor.typed.javadsl.BuiltReceive.receiveMessage(ReceiveBuilder.scala:204)
    at akka.actor.typed.javadsl.Receive.receive(Receive.scala:53)
    at akka.actor.typed.javadsl.AbstractBehavior.receive(AbstractBehavior.scala:64)
    at akka.actor.typed.Behavior$.interpret(Behavior.scala:274)
    at akka.actor.typed.Behavior$.interpretMessage(Behavior.scala:230)
    at akka.actor.typed.internal.InterceptorImpl$$anon$2.apply(InterceptorImpl.scala:57)
    at akka.actor.typed.internal.SimpleSupervisor.aroundReceive(Supervision.scala:124)
    at akka.actor.typed.internal.InterceptorImpl.receive(InterceptorImpl.scala:85)
    at akka.actor.typed.Behavior$.interpret(Behavior.scala:274)
    at akka.actor.typed.Behavior$.interpretMessage(Behavior.scala:230)
    at akka.actor.typed.internal.adapter.ActorAdapter.handleMessage(ActorAdapter.scala:131)
    at akka.actor.typed.internal.adapter.ActorAdapter.aroundReceive(ActorAdapter.scala:107)
    at akka.actor.ActorCell.receiveMessage(ActorCell.scala:579)
    at akka.actor.ActorCell.invoke(ActorCell.scala:547)
    at akka.dispatch.Mailbox.processMailbox(Mailbox.scala:270)
    at akka.dispatch.Mailbox.run(Mailbox.scala:231)
    at akka.dispatch.Mailbox.exec(Mailbox.scala:243) <5 internal calls>
Z1 main: sending second package of messages
Z1 main: messages send
actorMultiply: operation count = 3
actorMultiply: operation count = 4
actorDivide: operation count = 4
actorDivide: operation count = 5
```

Jak widać, dla strategii **Resume** działanie aktora `actorDivide` nie zostało przerwane (ani zresetowane - jest kontynuowane). Błąd został "zignorowany". Widać, że `operation count` aktora `actorDivide` nie utracił swojej poprzedniej wartości (przed błędem).

### 1.1.3 Restart

```
Z1 main: started
Z1 main: actor system ready
actorMultiply: operation count = 1
actorDivide: operation count = 1
actorMultiply: operation count = 2
actorDivide: operation count = 2
[actorMath-akka.actor.default-dispatcher-3] ERROR akka.actor.typed.Behavior$ - Supervisor RestartSupervisor saw failure: / by zero
java.lang.ArithmeticException: Create breakpoint: / by zero
    at Z1.MathActorDivide.onMathCommandDivide(MathActorDivide.java:33)
    at akka.actor.typed.javadsl.BuiltReceive.receive(ReceiveBuilder.scala:213)
    at akka.actor.typed.javadsl.BuiltReceive.receiveMessage(ReceiveBuilder.scala:204)
    at akka.actor.typed.javadsl.Receive.receive(Receive.scala:53)
    at akka.actor.typed.javadsl.AbstractBehavior.receive(AbstractBehavior.scala:64)
    at akka.actor.typed.Behavior$.interpret(Behavior.scala:274)
    at akka.actor.typed.Behavior$.interpretMessage(Behavior.scala:230)
    at akka.actor.typed.internal.InterceptorImpl$$anon$2.apply(InterceptorImpl.scala:57)
    at akka.actor.typed.internal.RestartSupervisor.aroundReceive(Supervision.scala:268)
    at akka.actor.typed.internal.InterceptorImpl.receive(InterceptorImpl.scala:85)
    at akka.actor.typed.Behavior$.interpret(Behavior.scala:274)
    at akka.actor.typed.Behavior$.interpretMessage(Behavior.scala:230)
    at akka.actor.typed.internal.adapter.ActorAdapter.handleMessage(ActorAdapter.scala:131)
    at akka.actor.typed.internal.adapter.ActorAdapter.aroundReceive(ActorAdapter.scala:107)
    at akka.actor.ActorCell.receiveMessage(ActorCell.scala:579)
    at akka.actor.ActorCell.invoke(ActorCell.scala:547)
    at akka.dispatch.Mailbox.processMailbox(Mailbox.scala:270)
    at akka.dispatch.Mailbox.run(Mailbox.scala:231)
    at akka.dispatch.Mailbox.exec(Mailbox.scala:243) <5 internal calls>
Z1 main: sending second package of messages
Z1 main: messages send
actorMultiply: operation count = 3
actorDivide: operation count = 1
actorMultiply: operation count = 4
actorDivide: operation count = 2
```

Jak widać, dla strategii **Restart** aktor `actorDivide` został zresetowany. Widać, że `operation count` aktora `actorDivide` nalicza się od nowa (ale dopiero po błędzie - dlatego jest 2 a nie 3).

## 1.2 Zadanie 2

### 1.2.1 Z2\_Main

```
[z2main-akka.actor.default-dispatcher-5] INFO akka.event.slf4j.Slf4jLogger - Slf4jLogger started
akka://z2main/user/upper1 registered
akka://z2main/user/upper2 registered
creating receive for text service
request: hello
sending to worker: Actor[akka://z2main/user/upper1#1870539141]
sending to worker: Actor[akka://z2main/user/upper2#717108534]
HELLO
HELLO
```

Jak widać, dla pojedynczego node'a wszystko wykonało się poprawnie (rejestracja aktorów - > wysłanie requesta - > przesłanie go do dwóch aktorów - > dwie odpowiedzi)

## 1.2.2 Z2\_NodeA oraz Z2\_NodeB

```
Node A: config: Config(SimpleConfigObject({"akka":{"actor":{"provider":"cluster"},"cluster":{"downing-provider-class":"akka.cluster.sbr.SplitBrainResolverProvider"},"seed-nodes":["akka://ClusterSystem@127.0.0.1:2551","akka://ClusterSystem@127.0.0.1:2552"]},"remote":{"artery":{"canonical":{"hostname":"127.0.0.1","port":2551}}}}}))
[ClusterSystem-akka.actor.default-dispatcher-5] INFO akka.event.slf4j.Slf4jLogger - Slf4jLogger started
[ClusterSystem-akka.actor.default-dispatcher-5] INFO akka.remote.artery.top.ArteryTcpTransport - Remoting started with transport [Artery top]; listening on address [akka://ClusterSystem@127.0.0.1:2551] with UID [-7681288494737835487]
[ClusterSystem-akka.actor.default-dispatcher-5] INFO akka.cluster.Cluster - Cluster Node [akka://ClusterSystem@127.0.0.1:2551] - Starting up, Akka version [2.6.14] ...
[ClusterSystem-akka.actor.default-dispatcher-5] INFO akka.cluster.Cluster - Cluster Node [akka://ClusterSystem@127.0.0.1:2551] - Registered cluster JMX MBean [akka:type=Cluster]
[ClusterSystem-akka.actor.default-dispatcher-5] INFO akka.cluster.Cluster - Cluster Node [akka://ClusterSystem@127.0.0.1:2551] - Started up successfully
[ClusterSystem-akka.actor.default-dispatcher-3] INFO akka.cluster.sbr.SplitBrainResolver - SBR started. Config: strategy [KeepMajority], stable-after [20 seconds], down-all-when-unstable [15 seconds], selfUniqueAddress [akka://ClusterSystem@127.0.0.1:2551#-7681288494737835487], selfDc [default].
akka://ClusterSystem/user/upper2 registered
akka://ClusterSystem/user/upper1 registered
[ClusterSystem-akka.actor.default-dispatcher-16] INFO akka.cluster.Cluster - Cluster Node [akka://ClusterSystem@127.0.0.1:2551] - Received InitJoin message from [Actor[akka://ClusterSystem@127.0.0.1:2552/system/cluster/core/daemon/joinSeedNodeProcess-1#1061670448]], but this node is not initialized yet
[ClusterSystem-akka.actor.default-dispatcher-16] INFO akka.cluster.Cluster - Cluster Node [akka://ClusterSystem@127.0.0.1:2551] - Received InitJoinAck message from [Actor[akka://ClusterSystem@127.0.0.1:2552/system/cluster/core/daemon/joinSeedNodeProcess-1#1061670448]] to [akka://ClusterSystem@127.0.0.1:2551]
[ClusterSystem-akka.actor.default-dispatcher-16] INFO akka.cluster.Cluster - Cluster Node [akka://ClusterSystem@127.0.0.1:2551] - Node [akka://ClusterSystem@127.0.0.1:2551] is JOINING itself (with roles [dc-default], version [0.0.0]) and forming new cluster
[ClusterSystem-akka.actor.default-dispatcher-16] INFO akka.cluster.Cluster - Cluster Node [akka://ClusterSystem@127.0.0.1:2551] - is the new leader among reachable nodes (more leaders may exist)
[ClusterSystem-akka.actor.default-dispatcher-16] INFO akka.cluster.Cluster - Cluster Node [akka://ClusterSystem@127.0.0.1:2551] - Leader is moving node [akka://ClusterSystem@127.0.0.1:2551] to [Up]
[ClusterSystem-akka.actor.default-dispatcher-3] INFO akka.cluster.sbr.SplitBrainResolver - This node is now the leader responsible for taking SBR decisions among the reachable nodes (more leaders may exist).
[ClusterSystem-akka.actor.default-dispatcher-3] INFO akka.cluster.Cluster - Cluster Node [akka://ClusterSystem@127.0.0.1:2551] - Received InitJoin message from [Actor[akka://ClusterSystem@127.0.0.1:2552/system/cluster/core/daemon/joinSeedNodeProcess-1#1061670448]] to [akka://ClusterSystem@127.0.0.1:2551]
[ClusterSystem-akka.actor.default-dispatcher-3] INFO akka.cluster.Cluster - Cluster Node [akka://ClusterSystem@127.0.0.1:2551] - Sending InitJoinAck message from node [akka://ClusterSystem@127.0.0.1:2551] to [Actor[akka://ClusterSystem@127.0.0.1:2552/system/cluster/core/daemon/joinSeedNodeProcess-1#1061670448]] (version [2.6.14])
[ClusterSystem-akka.actor.default-dispatcher-3] INFO akka.cluster.Cluster - Cluster Node [akka://ClusterSystem@127.0.0.1:2551] - Node [akka://ClusterSystem@127.0.0.1:2552] is JOINING, roles [dc-default], version [0.0.0]
[ClusterSystem-akka.actor.default-dispatcher-16] INFO akka.cluster.Cluster - Cluster Node [akka://ClusterSystem@127.0.0.1:2551] - Leader is moving node [akka://ClusterSystem@127.0.0.1:2552] to [Up]
HELLO
HELLO
```

```
Node B: config: Config(SimpleConfigObject({"akka":{"actor":{"provider":"cluster"},"cluster":{"downing-provider-class":"akka.cluster.sbr.SplitBrainResolverProvider"},"seed-nodes":["akka://ClusterSystem@127.0.0.1:2551","akka://ClusterSystem@127.0.0.1:2552"]},"remote":{"artery":{"canonical":{"hostname":"127.0.0.1","port":2552}}}}}))
[ClusterSystem-akka.actor.default-dispatcher-14] INFO akka.remote.artery.top.ArteryTcpTransport - Remoting started with transport [Artery top]; listening on address [akka://ClusterSystem@127.0.0.1:2552] with UID [413824359916154646]
[ClusterSystem-akka.actor.default-dispatcher-14] INFO akka.cluster.Cluster - Cluster Node [akka://ClusterSystem@127.0.0.1:2552] - Starting up, Akka version [2.6.14] ...
[ClusterSystem-akka.actor.default-dispatcher-14] INFO akka.cluster.Cluster - Cluster Node [akka://ClusterSystem@127.0.0.1:2552] - Registered cluster JMX MBean [akka:type=Cluster]
[ClusterSystem-akka.actor.default-dispatcher-14] INFO akka.cluster.Cluster - Cluster Node [akka://ClusterSystem@127.0.0.1:2552] - Started up successfully
[ClusterSystem-akka.actor.default-dispatcher-3] INFO akka.cluster.sbr.SplitBrainResolver - SBR started. Config: strategy [KeepMajority], stable-after [20 seconds], down-all-when-unstable [15 seconds], selfUniqueAddress [akka://ClusterSystem@127.0.0.1:2552#413824359916154646], selfDc [default].
creating receive for text service
[ClusterSystem-akka.actor.default-dispatcher-6] INFO akka.cluster.Cluster - Cluster Node [akka://ClusterSystem@127.0.0.1:2552] - Received InitJoin message from [Actor[akka://ClusterSystem@127.0.0.1:2551/system/cluster/core/daemon/firstSeedNodeProcess-1#-409234290]], but this node is not initialized yet
[ClusterSystem-akka.actor.default-dispatcher-6] INFO akka.cluster.Cluster - Cluster Node [akka://ClusterSystem@127.0.0.1:2552] - Received InitJoin message from [Actor[akka://ClusterSystem@127.0.0.1:2551/system/cluster/core/daemon/firstSeedNodeProcess-1#-409234290]], but this node is not initialized yet
[ClusterSystem-akka.actor.default-dispatcher-6] INFO akka.cluster.Cluster - Cluster Node [akka://ClusterSystem@127.0.0.1:2552] - Received InitJoin message from [Actor[akka://ClusterSystem@127.0.0.1:2551/system/cluster/core/daemon/firstSeedNodeProcess-1#-409234290]], but this node is not initialized yet
[ClusterSystem-akka.actor.default-dispatcher-6] INFO akka.cluster.Cluster - Cluster Node [akka://ClusterSystem@127.0.0.1:2552] - Received InitJoin message from [Actor[akka://ClusterSystem@127.0.0.1:2551/system/cluster/core/daemon/firstSeedNodeProcess-1#-409234290]], but this node is not initialized yet
[ClusterSystem-akka.actor.default-dispatcher-6] INFO akka.cluster.Cluster - Cluster Node [akka://ClusterSystem@127.0.0.1:2552] - Received InitJoinAck message from [Actor[akka://ClusterSystem@127.0.0.1:2551/system/cluster/core/daemon/firstSeedNodeProcess-1#-409234290]] to [akka://ClusterSystem@127.0.0.1:2552]
[ClusterSystem-akka.actor.default-dispatcher-6] INFO akka.cluster.Cluster - Cluster Node [akka://ClusterSystem@127.0.0.1:2552] - Welcome from [akka://ClusterSystem@127.0.0.1:2551]
request: hello
sending to worker: Actor[akka://ClusterSystem@127.0.0.1:2551/user/upper1#-1005349975]
sending to worker: Actor[akka://ClusterSystem@127.0.0.1:2551/user/upper2#-1428491292]
```

Jak widać, dla dwóch node'ów akcje wykonały się zgodnie z założeniem:

- *NodeA* (odpowiedzialny za workerów): rejestracja aktorów — > dwie odpowiedzi
- *NodeB* (odpowiedzialny za text service): wysłanie requesta — > przesłanie go do dwóch aktorów