

Using Dispatcher



() , .

- 3:7 -
- - 30km
- 1~10 -
- Type Type - ,

```

public HigPassGateActor()
{
    id = Guid.NewGuid().ToString();
    logger.Info($"    :{id}");

    ReceiveAsync<string>(async msg =>
    {
        //
        logger.Debug($" {msg}");
    });
}

```

```

public CashGateActor()
{
    rnd = new Random();

    id = Guid.NewGuid().ToString();
    logger.Info($"    :{id}");

    ReceiveAsync<string>(async msg =>
    {
        // 1~10
        int delay = rnd.Next(1000, 10000);
        await Task.Delay(delay);
        logger.Debug($" {msg}-{delay}");
    });
}

```

```

actor.deployment {
    /highpass-gate-pool {
        router = round-robin-pool
        nr-of-instances = 3
    }

    /cashpass-gate-pool {
        router = smallest-mailbox-pool
        nr-of-instances = 7
    }
}

```

.
 . .
 ? ,
 , .
 , Akka Smallest-mailbox .
 • RoundRobin :
 • Smallest-MailBox :

Akka .

: <https://getakka.net/articles/actors/routers.html>

```
#
fast-dispatcher {
    type = Dispatcher
    throughput = 2
}

# (8 7 )
slow-dispatcher {
    type = Dispatcher
    throughput = 7
}
```

```
,
.
, Dispatcher
( ) .
,
: https://doc.akka.io/docs/akka/current/dispatchers.html
```

```
AkkaLoad.RegisterActor(
    "highpass",
    actorSystem.ActorOf(Props.Create<HighPassGateActor>()
        .WithDispatcher("fast-dispatcher")
        .WithRouter(FromConfig.Instance),
        "highpass-gate-pool"
    ));

AkkaLoad.RegisterActor(
    "cashpass",
    actorSystem.ActorOf(Props.Create<CashGateActor>()
        .WithDispatcher("slow-dispatcher")
        .WithRouter(FromConfig.Instance),
        "cashpass-gate-pool"
    ));
```

```
.
?? .
```