

REEF Tutorial

Yunseong Lee

2017/09/21

Goal of this tutorial

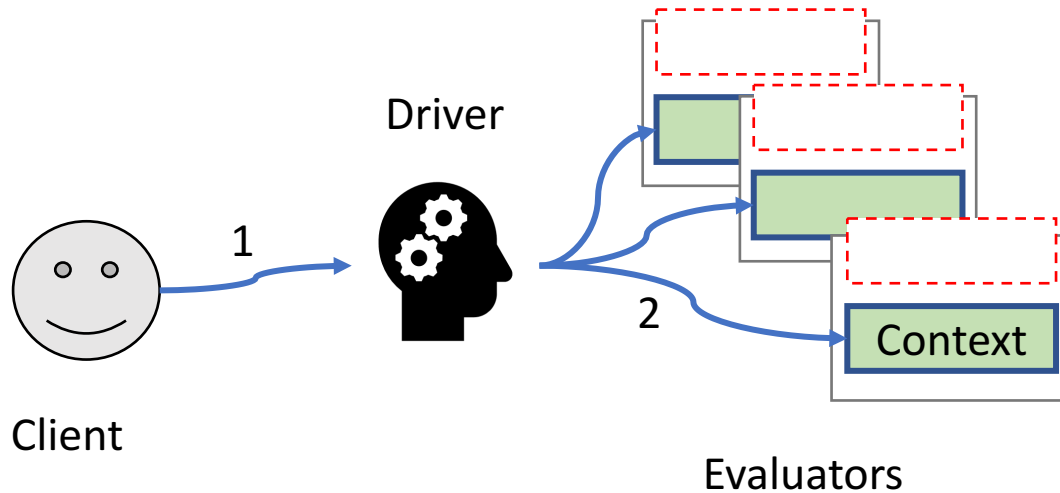
- Help students start programming!
 - What you need to write in code-level
 - Step-by-step instruction of how to build a REEF application
 - Tips and Q & A
- This is **not** a hands-on practice
 - Most effective solution, but we don't have enough time 😞
 - You will see the TA struggle to write code 😊
(very similar example with our assignment)
 - Slides will be uploaded, but the code is NOT!

Tips

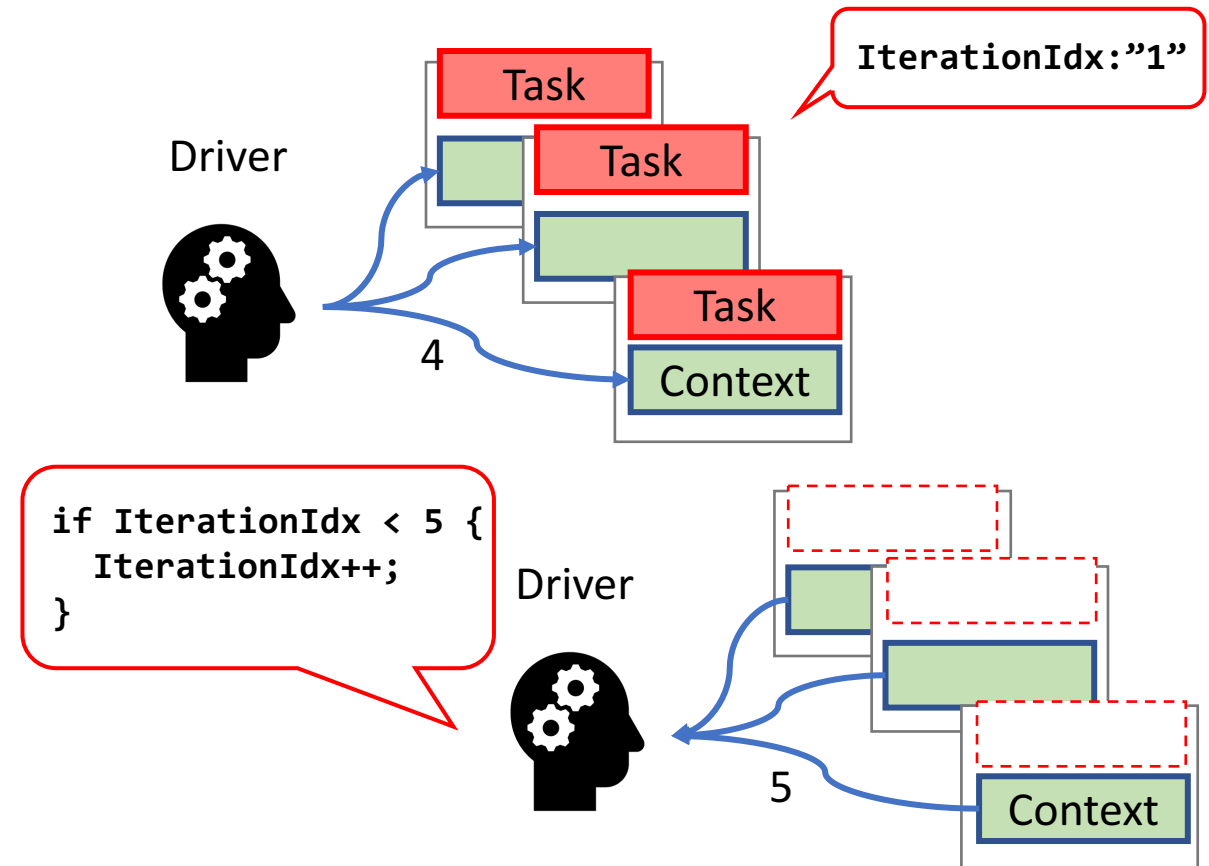
- Coding practices: *checkstyle will not allow you to write a bad code*
 - Add package-info (with a newline at the end)
 - final as possible
 - ...
- REEF beginners' Pitfalls
 - Don't forget the @Inject in the constructors
 - Don't forget @Unit in the Driver
 - It allows injecting inner classes
 - without injectable constructors
- Suspend example will help you understand how memento is used

Welcome to IterativeHelloWorld!

- Scenario



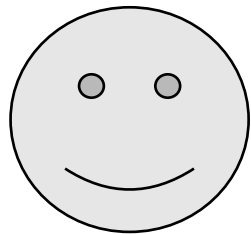
1. Client starts REEF Driver
2. Driver allocated Evaluators
3. Driver submits Context to Evaluators



4. Driver submits Tasks to the Contexts
We will send the iteration idx as Memento
5. Once Tasks completed, check the iteration idx and start the next iteration (go to 4)

Client builds the configurations of a REEF job

REEF provides ConfigurationModuleBuilder



Client

Driver
Config

```
final Configuration driverConf = DriverConfiguration.CONF
    .set(DriverConfiguration.|
    .build();
```

ON_EVALUATOR_FAILED	OptionalImpl<EventHandler<FailedEvaluator>>
CONF	ConfigurationModule
DRIVER_IDENTIFIER	OptionalParameter<String>
DRIVER_JOB_SUBMISSION_DIRECTORY	OptionalParameter<String>
DRIVER_MEMORY	OptionalParameter<Integer>
EVALUATOR_DISPATCHER_THREADS	OptionalParameter<Integer>

DriverConfiguration

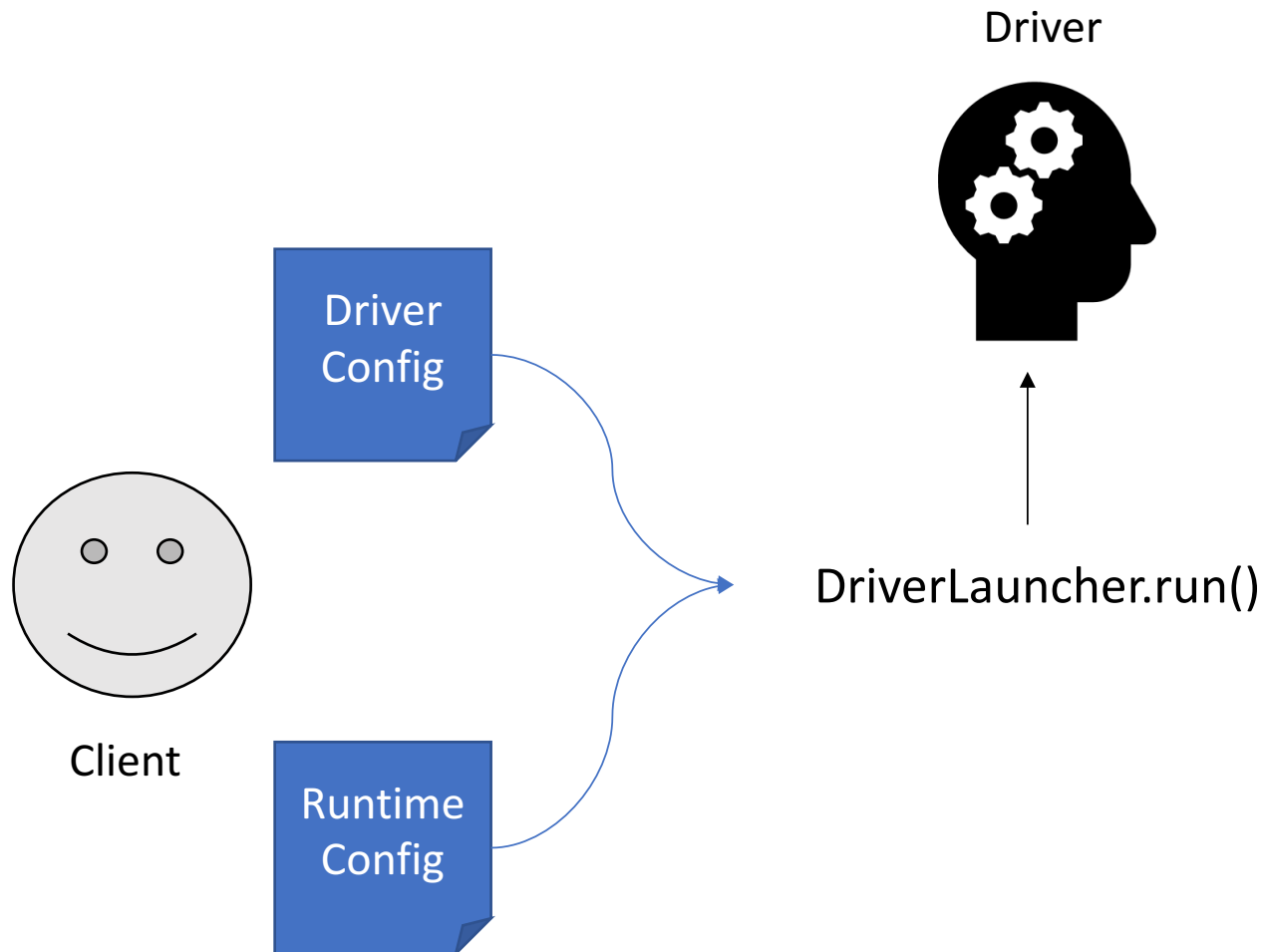
Runtime
Config

```
final Configuration runtimeConf = LocalRuntimeConfiguration.CONF
    .set(LocalRuntimeConfiguration.|
    .build();
```

CONF (= new LocalRuntimeConfig ...)	ConfigurationModule
DRIVER_CONFIGURATION_PROVIDERS ...	OptionalImpl<ConfigurationProvider>
JVM_HEAP_SLACK (= new OptionalParameter<...>	OptionalParameter<Double>
MAX_NUMBER_OF_EVALUATORS (= new Optiona...	OptionalParameter<Integer>

LocalRuntimeConfiguration
YarnRuntimeConfiguration

Launch the Driver!



On Driver's Start!

- An EventHandler configured to DriverConfiguration.*ON_DRIVER_STARTED* is called

Driver



```
public final class StartHandler implements EventHandler<StartTime> {  
    @Override  
    public void onNext(final StartTime startTime) {  
        // Do what you want  
    }  
}
```

- TODO: Request N Evaluators via EvaluatorRequestor
(Hint: inject the object in the Driver's constructor)

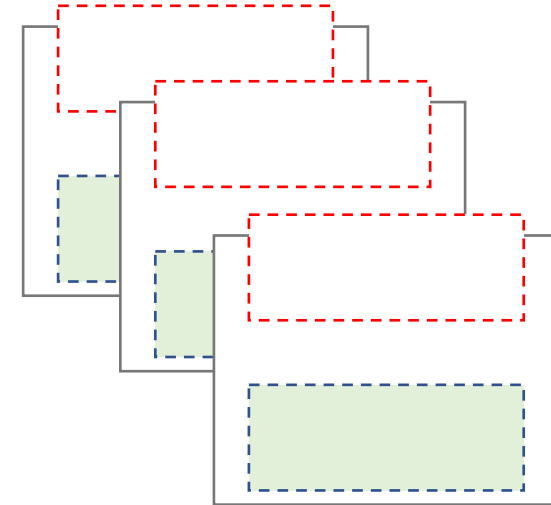
On Allocated Evaluator

- An EventHandler configured to `DriverConfiguration.ON_EVALUATOR_ALLOCATED` is called

Driver



```
public final class AllocatedEvaluatorHandler implements EventHandler<AllocatedEvaluator> {  
    @Override  
    public void onNext(final AllocatedEvaluator allocatedEvaluator) {  
        // Do what you want.  
    }  
}
```

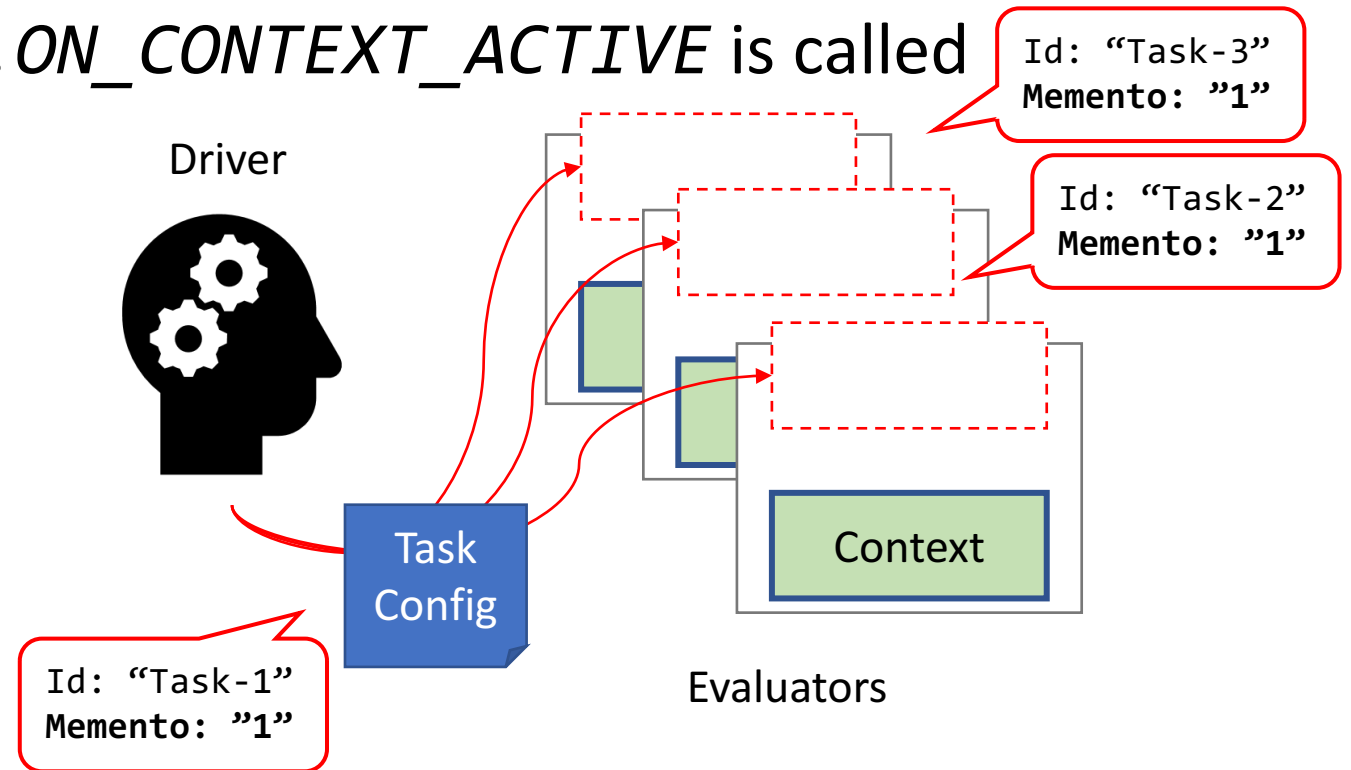


Evaluators

- TODO: Submit Context!
(Hint: Use ContextConfiguration)

On Active Context

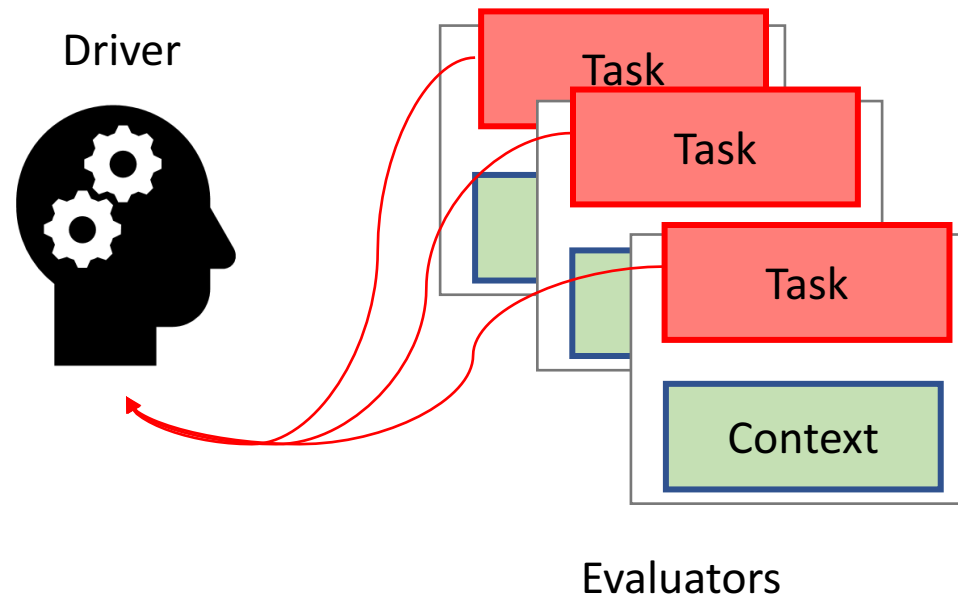
- An EventHandler configured to `DriverConfiguration.ON_CONTEXT_ACTIVE` is called



- **TODO: Submit Task!**
Let's set iteration idx as Memento

On Task Completed

- An EventHandler configured to `DriverConfiguration.ON_TASK_COMPLETE` is called



- TODO: Check `IterationIdx` and repeat!

Q&A?