

Build: 10x Faster

方镇澎

2020/9/1

Outline

1. Background
2. Performance Analysis
3. Optimize Gradle Build Time
4. Optimize Kotlin Compilation Time

Background

Build is slow

Build at least 3 Times:

1. Local
2. Before Merge
3. After Merge

```
BUILD SUCCESSFUL in 13m 40s
```

```
130 actionable tasks: 86 executed, 3 from cache, 41 up-to-date
```

Performance Analysis

Gradle performance analysis

1. `gradle --scan`
2. `gradle --profile`
3. `gradle --info --debug`

gradle --scan



Build Scan



✓ 19-investigate-what...

build

5 Dec 2017 20:30:54 GMT



Summary



Console log



Timeline



Performance



Projects



Dependencies



Plugins



Custom values



Switches



Infrastructure

Build

Configuration

Dependency resolution

Task execution

Daemon

Network activity

Settings and suggestions

Total build time 14.689s

Startup 0.013s

Settings and buildSrc 0.362s

Loading projects 0.003s

Configuration 13.526s

Task execution 0.784s

End of build 0.001s

Total garbage collection time 0.000s

Peak heap memory usage

PS Eden Space 195.24/313.52 MB (62.3%)

PS Survivor Space 18.86/18.87 MB (99.9%)

PS Old Gen 53.67/716.18 MB (7.5%)

gradle --profile

Profile report

Profiled build: :restful:onboarding:onboarding-persistence:build

Started on: 2020/08/31 - 12:46:23

Summary

Configuration

Dependency Resolution

Artifact Transforms

Task Execution

| Description | Duration |
|-----------------------|-----------|
| Total Build Time | 13m40.95s |
| Startup | 0.945s |
| Settings and buildSrc | 6.377s |
| Loading Projects | 0.552s |
| Configuring Projects | 30.921s |
| Artifact Transforms | 0.107s |
| Task Execution | 13m26.14s |

gradle --info (--debug)

```
> Task :restful:onboarding:onboarding-persistence:compileTestKotlin
Build cache key for task ':restful:onboarding:onboarding-persistence:compileTestKotlin' is 5faed36f8271ee5a3c598662dba0aa96
Task ':restful:onboarding:onboarding-persistence:compileTestKotlin' is not up-to-date because:
  Input property 'classpath' file C:\HUE\Workspace\Develop\hue-hr-recruiting\restful\onboarding\onboarding-domain\build\classes\kotlin\main\wap\hr\recruiti
  Input property 'classpath' file C:\HUE\Workspace\Develop\hue-hr-recruiting\restful\onboarding\onboarding-domain\build\classes\kotlin\main\wap\hr\recruiti
  Input property 'classpath' file C:\HUE\Workspace\Develop\hue-hr-recruiting\restful\onboarding\onboarding-domain\build\classes\kotlin\main\wap\hr\recruiti
  Input property 'classpath' file C:\HUE\Workspace\Develop\hue-hr-recruiting\restful\onboarding\onboarding-domain\build\classes\kotlin\main\wap\hr\recruiti
File or directory 'C:\HUE\Workspace\Develop\hue-hr-recruiting\restful\onboarding\onboarding-persistence\src\test\java', not found
File or directory 'C:\HUE\Workspace\Develop\hue-hr-recruiting\restful\onboarding\onboarding-persistence\src\test\java', not found
Using Kotlin/JVM incremental compilation
i: found daemon on port 17997 (762841 ms old), trying to connect
i: connected to the daemon
Options for KOTLIN DAEMON: IncrementalCompilationOptions(super=CompilationOptions(compilerMode=INCREMENTAL_COMPILER, targetPlatform=JVM, reportCategories=[kt, kts, java]), areFileChangesKnown=false, modifiedFiles=null, deletedFiles=null, workingDir=C:\HUE\Workspace\Develop\hue-hr-recruiting\restful\onboarding-persistence\build\kotlin\compileTestKotlin, multiModuleICSettings=MultiModuleICSettings(buildHistoryFile=C:\HUE\Workspace\Develop\hue-hr-recruiting\restful\onboarding\onboarding-persistence\build\kotlin\compileTestKotlin\buildHistoryFile, outputFiles=[C:\HUE\Workspace\Develop\hue-hr-recruiting\restful\onboarding\onboarding-persistence\build\classes\kotlin\test, C:\HUE\Workspace\Develop\hue-hr-recruiting\restful\onboarding\onboarding-persistence\build\classes\kotlin\test\classes, C:\HUE\Workspace\Develop\hue-hr-recruiting\restful\onboarding\onboarding-persistence\build\classes\kotlin\test\resources], incrementalCompilationUseCache=true, incrementalCompilationUseCacheFallback=false, incrementalCompilationUseCacheFallbackReason=NO_SOURCE_FILES_AVAILABLE))
Stored cache entry for task ':restful:onboarding:onboarding-persistence:compileTestKotlin' with cache key 5faed36f8271ee5a3c598662dba0aa96
:restful:onboarding:onboarding-persistence:compileTestKotlin (Thread[Execution worker for ':' Thread 3,5,main]) completed. Took 1.203 secs.
:restful:onboarding:onboarding-persistence:compileTestJava (Thread[Execution worker for ':' Thread 3,5,main]) started.

> Task :restful:onboarding:onboarding-persistence:compileTestJava NO-SOURCE
```

Performance analysis result

Profile report

Profiled build: :restful:onboarding:onboarding-persistence:build

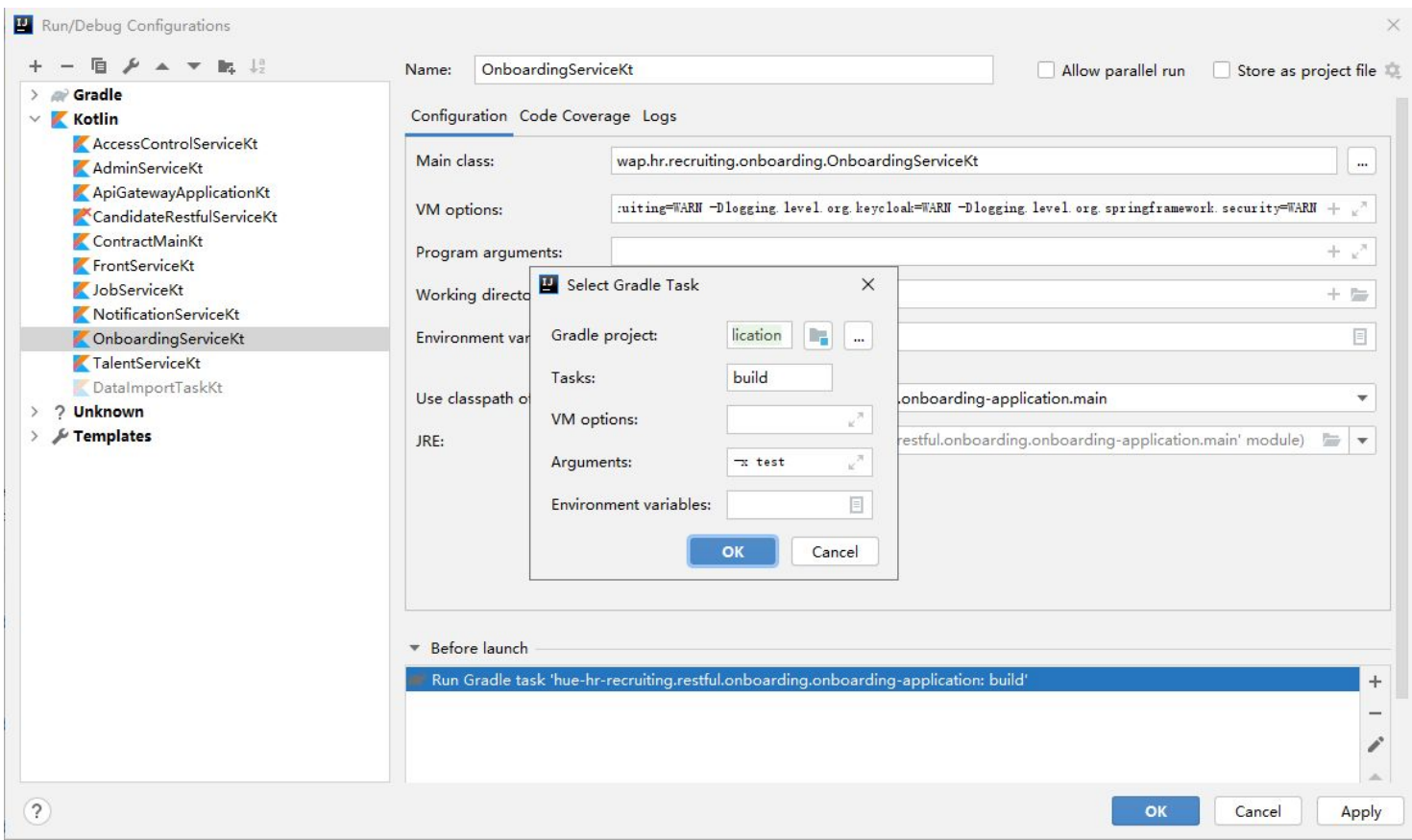
Started on: 2020/08/31 - 12:46:23

- Summary
- Configuration
- Dependency Resolution
- Artifact Transforms
- Task Execution

| Task | Duration | Result |
|--|-----------|---------|
| :restful:onboarding:onboarding-persistence | 11m48.73s | (total) |
| :restful:onboarding:onboarding-persistence:test | 6m13.26s | |
| :restful:onboarding:onboarding-persistence:compileKotlin | 5m31.22s | |

Repository test? gradle build -x test

Skip test in IntelliJ IDEA



Optimize Gradle Build Time

Common optimization parameters

```
org.gradle.caching=true  
org.gradle.parallel=true  
org.gradle.jvmargs=-Xmx2048m -Duser.language=en-us
```

```
59 actionable tasks: 3 executed, 56 up-to-date
```

Using Kotlin/JVM incremental compilation

i: found daemon on port 17997 (726872 ms old), trying to connect

i: connected to the daemon

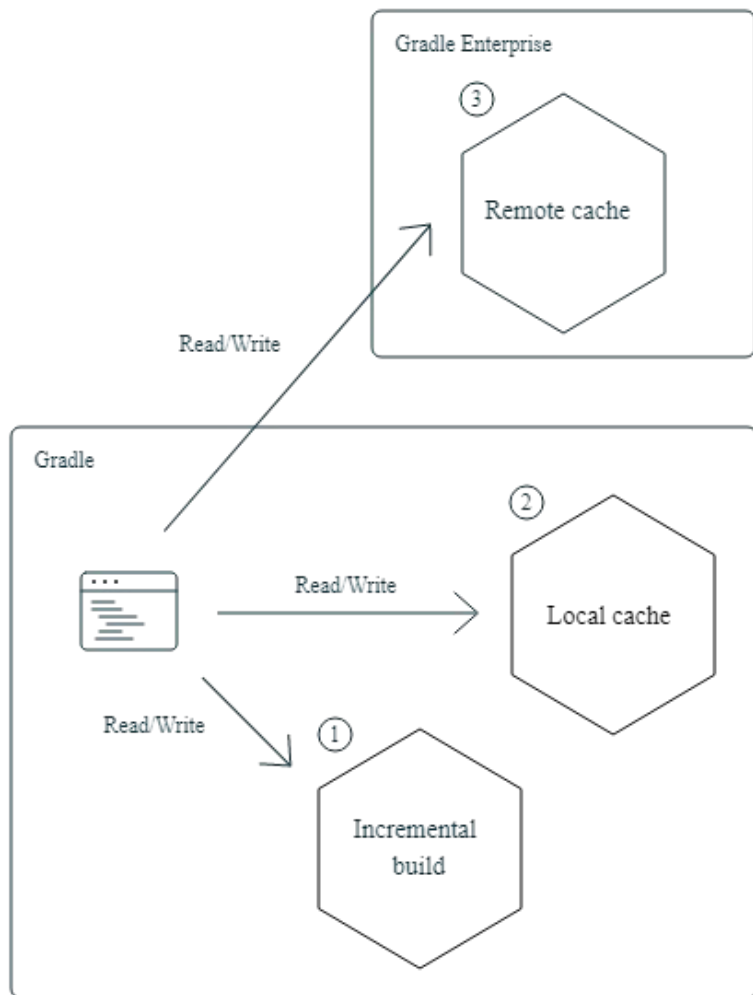
Options for KOTLIN DAEMON: IncrementalCompilationOptions(super=Compilat

Caching for faster build

Gradle logically has 3 layers of reuse that allow potentially expensive tasks from being executed:

1. Up-to-date checks for task outputs in the workspace.
2. A lookup for task outputs in the local cache.
3. A lookup for task outputs in a remote cache, such as the remote cache provided out of the box in [Gradle Enterprise](#).

Caching for faster build



Incremental Compilation

- Gradle will recompile all classes *affected* by a change.
- A class is *affected* if it has been changed or if it depends on another affected class. This works no matter if the other class is defined in the same project, another project or even an external library.
- A class's dependencies are determined from type references in its bytecode.
- Since constants can be inlined, any change to a constant will result in Gradle recompiling all source files. For that reason, you should try to minimize the use of constants in your source code and replace them with static methods where possible.
- Since source-retention annotations are not visible in bytecode, changes to a source-retention annotation will result in full recompilation.
- You can improve incremental compilation performance by applying good software design principles like loose coupling. For instance, if you put an interface between a concrete class and its dependents, the dependent classes are only recompiled when the interface changes, but not when the implementation changes.
- The class analysis is cached in the project directory, so the first build after a clean checkout can be slower. Consider turning off the incremental compiler on your build server.

Compile Avoidance

<https://blog.gradle.org/incremental-compiler-avoidance>

Unfortunately, all features are already used in our project.

Optimize Kotlin Compilation Time

Why is compilation so slow?

.class

| | | | |
|--|-----------------|----------|--------|
| ThinProspectiveEmployeeForCJKReadOnly.class | 2020/8/31 12:53 | CLASS 文件 | 372 KB |
| EmployeeFamilyMemberInfoListReadWrite.class | 2020/8/31 12:50 | CLASS 文件 | 164 KB |
| EmployeeBasicInfoReadWrite.class | 2020/8/31 12:48 | CLASS 文件 | 124 KB |
| EmployeeCareerHistoryListReadWrite.class | 2020/8/31 12:48 | CLASS 文件 | 60 KB |
| EmployeeAddressInfoListReadWrite.class | 2020/8/31 12:48 | CLASS 文件 | 49 KB |
| ThinProspectiveEmployeeRepositoryKt.class | 2020/8/31 12:53 | CLASS 文件 | 38 KB |
| ThinProspectiveEmployeeRepositoryKt\$searchProspectiveEmployeesSpec\$1.class | 2020/8/31 12:53 | CLASS 文件 | 34 KB |
| ThinProspectiveEmployeeCustomRepository.class | 2020/8/31 12:53 | CLASS 文件 | 32 KB |
| EmployeeContractInfoReadWrite.class | 2020/8/31 12:48 | CLASS 文件 | 29 KB |
| OnboardingTemplateReadWrite.class | 2020/8/31 12:50 | CLASS 文件 | 29 KB |
| ProspectiveEmployeeReadWrite.class | 2020/8/31 12:50 | CLASS 文件 | 29 KB |
| EmployeeCommuteRouteInfoReadWrite.class | 2020/8/31 12:48 | CLASS 文件 | 26 KB |
| EmployeeQualificationListReadWrite.class | 2020/8/31 12:50 | CLASS 文件 | 25 KB |
| EmployeePersonalIndividualNumberListReadWrite.class | 2020/8/31 12:50 | CLASS 文件 | 23 KB |

372kb? 4 minutes to compile, it is the cause.

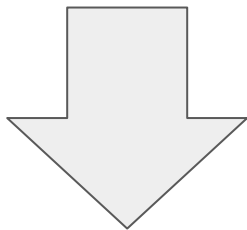
`arrow.syntax.function.pipe?`

pipe -> let,

372kb->369kb

ThinProspectiveEmployeeForCJKRepository.kt

```
private fun ThinProspectiveEmployeeForCJKEntity.toBankAccountInfo() = bankAccountInfo?.pipe { it: EmployeeBankAccountInfoEntity
    EmployeeBankAccountInfo(
        employeeId = ProspectiveEmployeeId(id),
        paymentType = it.paymentType?.pipe { BankAccount.PaymentType(it) },
        transferType = it.transferType?.pipe { BankAccount.TransferType(it) },
        bankCode = it.bankCode?.pipe { BankMasterDataCode(it) },
        accountType = it.accountType?.pipe { BankAccount.AccountType(it) },
        accountNumber = it.accountNumber?.pipe { TextInput(it) },
        accountHolderName = it.accountHolderName?.pipe { TextInput(it) }
    )
}
```



800 lines -> 300 lines
Reduce 2 minute compilation time
.class reduce to 160kb+

```
EmployeeBankAccountInfoReadWrite.run { this: EmployeeBankAccountInfoReadWrite
    bankAccountInfo?.toDomain()
},
```

Decompilation: Kotlin to Java

```
package wap.hr.recruiting.onboarding.dao
```

```
public object OnboardingReminderSettingReadWrite : wap.hr.recruiting.persistence.adapter.ReadWrite<wap.hr.recruiting.onboarding.domain.task.OnboardingReminderSettingInfo, wap.hr.recruiting.onboarding.domain.task.OnboardingReminderSettingInfo.Id, wap.hr.recruiting.onboarding.model.OnboardingReminderSettingEntity, kotlin.Long> {  
    public open fun wap.hr.recruiting.onboarding.domain.task.OnboardingReminderSettingInfo.getKey(): wap.hr.recruiting.onboarding.domain.task.OnboardingReminderSettingInfo.Id { /* compiled code */ }  
  
    public open fun wap.hr.recruiting.onboarding.model.OnboardingReminderSettingEntity.toDomain(): wap.hr.recruiting.onboarding.domain.task.OnboardingReminderSettingInfo { /* compiled code */ }  
  
    public open fun wap.hr.recruiting.onboarding.domain.task.OnboardingReminderSettingInfo.Id.toDtoKey(): kotlin.Long { /* compiled code */ }  
}  
  
    public open fun wap.hr.recruiting.onboarding.domain.task.OnboardingReminderSettingInfo.toMutableDto(): wap.hr.recruiting.onboarding.model.OnboardingReminderSettingEntity { /* compiled code */ }  
}
```



Kotlin to Java decompiler

0.6.0 Freddie Wang



Tool/Kotlin/show Kotlin ByteCode

Decompilation

```
OnboardingReminderSettingInfo(
```

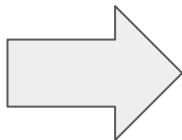
```
.....
```

```
assigneeAssignmentChangedMailTemplate =  
assigneeAssignmentChangedMailTemplat@.pipe {  
MailTemplateId(it) }
```

```
.....
```

```
)
```

```
.pipe .let .map if()else().....
```



```
$this$pipe$iv = var10014;  
var34 = var10013;  
var33 = var74;  
var32 = var10011;  
var31 = var73;  
var30 = var72;  
var29 = var10008;  
var28 = var10007;  
var27 = var10006;  
var26 = var71;  
var25 = var10004;  
var24 = var70;  
var23 = var10002;  
var22 = var10001;  
var21 = var10000;  
$i$f$pipe = false;  
var6 = false;  
var35 = new MailTemplateId($this$pipe$iv);  
var10000 = var21;  
var10001 = var22;  
var10002 = var23;  
var70 = var24;  
var10004 = var25;  
var71 = var26;  
var10006 = var27;  
var10007 = var28;  
var10008 = var29;  
var72 = var30;  
var73 = var31;  
var10011 = var32;  
var74 = var33;  
var10013 = var34;  
var75 = var35;
```

Decompilation

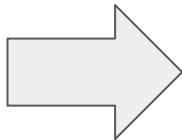
```
OnboardingReminderSettingInfo(
```

```
.....
```

```
    assigneeAssignmentChangedMailTemplate =  
    assigneeAssignmentChangedMailTemplate@.pipe {  
        MailTemplateId(it) }  
  
    .....
```

```
)
```

```
.pipe .let .map if()else().....
```



```
val assigneeAssignmentChangedMailTemplate =  
    assigneeAssignmentChangedMailTemplate@.pipe {  
        MailTemplateId(it) }  
  
.....
```

```
OnboardingReminderSettingInfo(
```

```
.....
```

```
    assigneeAssignmentChangedMailTemplate =  
    assigneeAssignmentChangedMailTemplate
```

```
.....
```

```
)
```

Performance result

| | | | |
|--|-----------------|----------|-------|
| EmployeeFamilyMemberInfoListReadWrite.class | 2020/8/31 16:23 | CLASS 文件 | 91 KB |
| ThinProspectiveEmployeeForCJKReadOnly.class | 2020/8/31 16:24 | CLASS 文件 | 88 KB |
| EmployeeBasicInfoReadWrite.class | 2020/8/31 16:23 | CLASS 文件 | 70 KB |
| EmployeeCareerHistoryListReadWrite.class | 2020/8/31 16:23 | CLASS 文件 | 38 KB |
| EmployeeAddressInfoListReadWrite.class | 2020/8/31 16:23 | CLASS 文件 | 37 KB |
| ThinProspectiveEmployeeRepositoryKt.class | 2020/8/31 16:24 | CLASS 文件 | 37 KB |
| ThinProspectiveEmployeeRepositoryKt\$searchProspectiveEmployeesSpec\$1.class | 2020/8/31 16:24 | CLASS 文件 | 34 KB |
| ThinProspectiveEmployeeCustomRepository.class | 2020/8/31 16:24 | CLASS 文件 | 32 KB |
| EmployeeContractInfoReadWrite.class | 2020/8/31 16:23 | CLASS 文件 | 29 KB |
| OnboardingTemplateReadWrite.class | 2020/8/31 16:23 | CLASS 文件 | 27 KB |
| EmployeeCommuteRouteInfoReadWrite.class | 2020/8/31 16:23 | CLASS 文件 | 26 KB |
| EmployeeQualificationListReadWrite.class | 2020/8/31 16:23 | CLASS 文件 | 25 KB |
| ProspectiveEmployeeReadWrite.class | 2020/8/31 16:23 | CLASS 文件 | 24 KB |
| EmployeePersonalIndividualNumberListReadWrite.class | 2020/8/31 16:23 | CLASS 文件 | 23 KB |
| EmployeeFamilyIndividualNumberListReadWrite.class | 2020/8/31 16:23 | CLASS 文件 | 23 KB |
| EmployeeEducationHistoryListReadWrite\$toDomain\$1.class | 2020/8/31 16:23 | CLASS 文件 | 21 KB |

Performance result 331s->36s

Profile report

Profiled build: -x :restful:onboarding:onboarding-persistence:test :restful:onboarding:onboarding-persistence:build

Started on: 2020/08/31 - 16:23:33

[Summary](#)[Configuration](#)[Dependency Resolution](#)[Artifact Transforms](#)[Task Execution](#)

| Task | Duration | Result |
|--|----------|------------|
| :restful:onboarding:onboarding-persistence | 37.393s | (total) |
| :restful:onboarding:onboarding-persistence:compileKotlin | 36.111s | |
| :restful:onboarding:onboarding-persistence:compileTestKotlin | 1.203s | |
| :restful:onboarding:onboarding-persistence:processResources | 0.063s | UP-TO-DATE |
| :restful:onboarding:onboarding-persistence:checkstyleMain | 0.005s | NO-SOURCE |
| :restful:onboarding:onboarding-persistence:jar | 0.004s | UP-TO-DATE |
| :restful:onboarding:onboarding-persistence:checkstyleTest | 0.003s | NO-SOURCE |
| :restful:onboarding:onboarding-persistence:inspectClassesForKotlinIC | 0.002s | UP-TO-DATE |
| :restful:onboarding:onboarding-persistence:processTestResources | 0.001s | UP-TO-DATE |
| :restful:onboarding:onboarding-persistence:testClasses | 0.001s | UP-TO-DATE |

Thank you