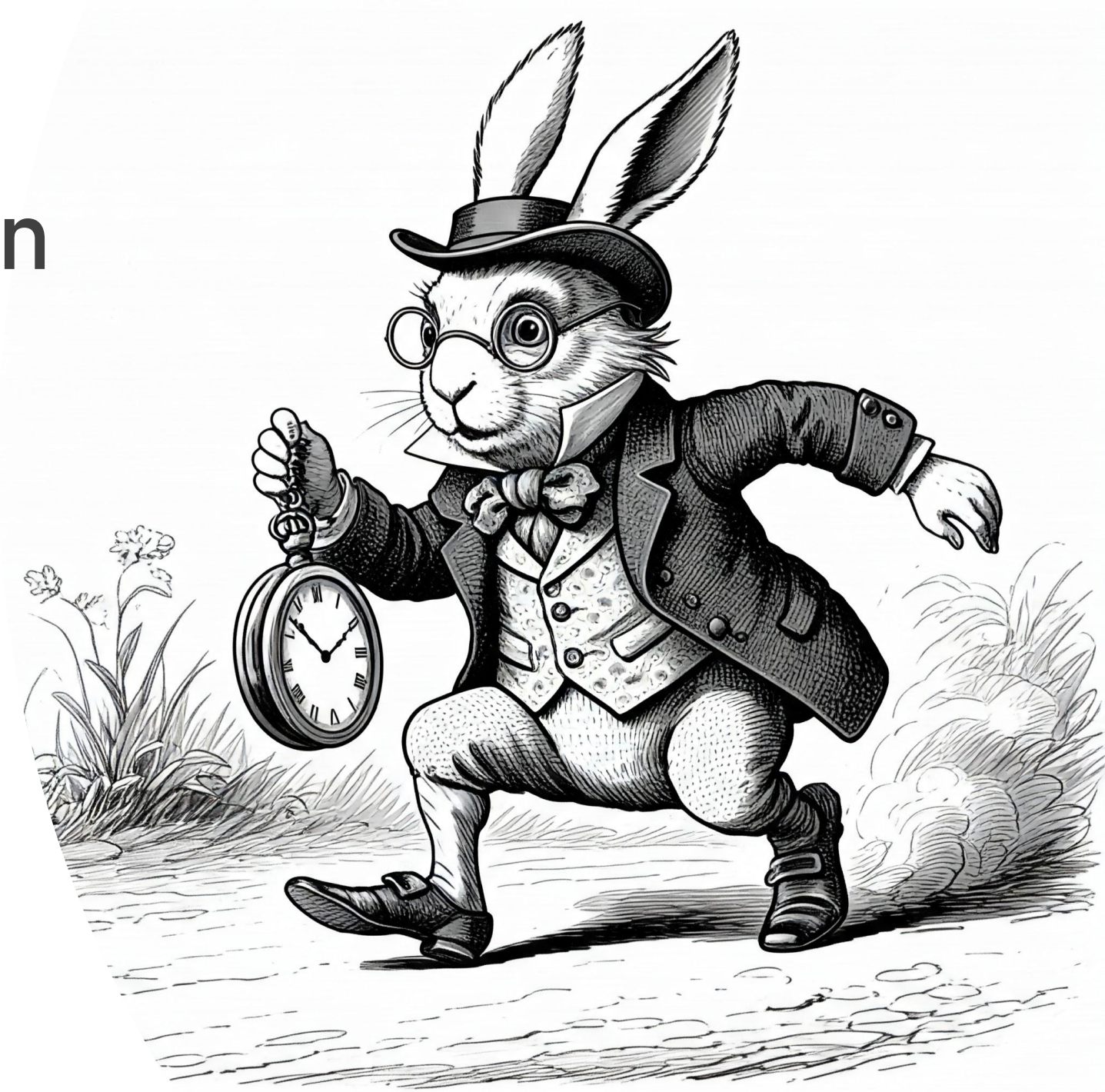
Comme Alice,
osez suivre le lapin blanc et plongez dans la JVM pour comprendre son fonctionnement



Qui suis-je ?

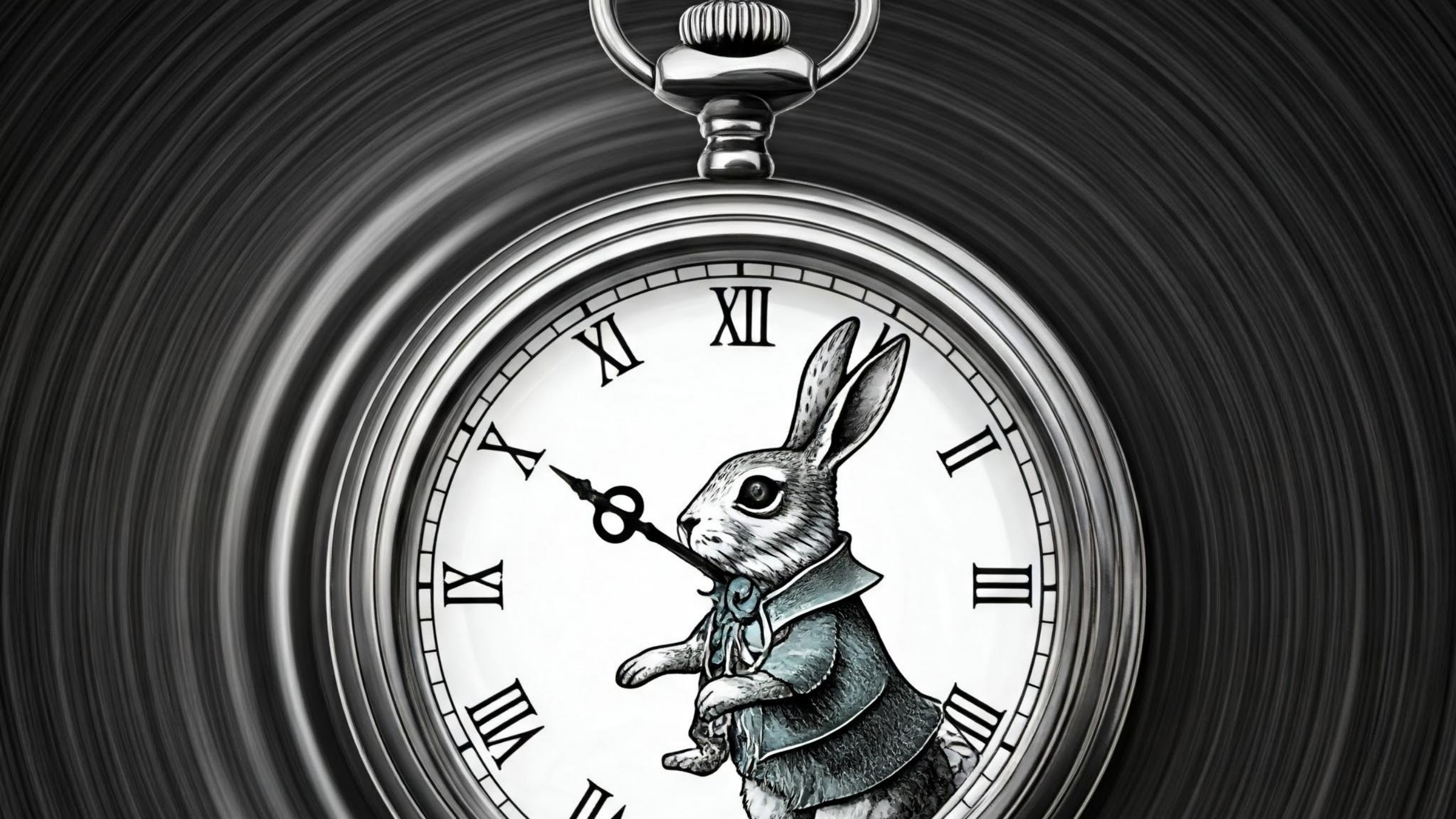
- Antoine DESSAIGNE
  - architecte logiciel

- 

  - éditeur de logiciels français
  - environ 4500 employés

Remerciements

- Aleksey Shipilëv
  - OpenJDK @ AWS
- Nicolai Parlog
  - Java Developer Advocate @ Oracle


XII
I
II
III
IX
X
XI

Qu'est-ce qui se passe ?

- NullPointerException
  - sur 1 benchmark sur 16
  - seulement 50% des exécutions
- Aucun problème sur nos tests
  - environ 11000 tests unitaires
  - environ 6000 tests d'intégration
- Vu 15 jours après avoir migré de 17.0.8 à 21.0.1

Que fait ce benchmark ?

- Il mesure la vitesse de rattrapage
- on simule 30 minutes de données aléatoires
- on mesure le temps pour réabsorber ces 30 minutes (environ 1 minute 30)

Et le code dans tout ça ?

```
final ReadableInterval validTime = value.getValidTime();
if (validTime.getStartMillis() < …) {
    // No exception here
}

…

if (validTime.getStartMillis() < …) {
    // NullPointerException
    // Cannot invoke "ReadableInterval.getStartMillis()"
    // because "validTime" is null
}
```

Un mauvais souvenir de Kafka

```
public void waitObject(Object obj, …) throws InterruptedException {
    synchronized (obj) {
        while (true) {
            if (condition.get())
                return;
            long currentTimeMs = milliseconds();
            if (currentTimeMs >= deadlineMs)
                throw new TimeoutException();
            obj.wait(deadlineMs - currentTimeMs);
            // wait() throws IllegalMonitorStateException
        }
    }
}
```

KAFKA-10902

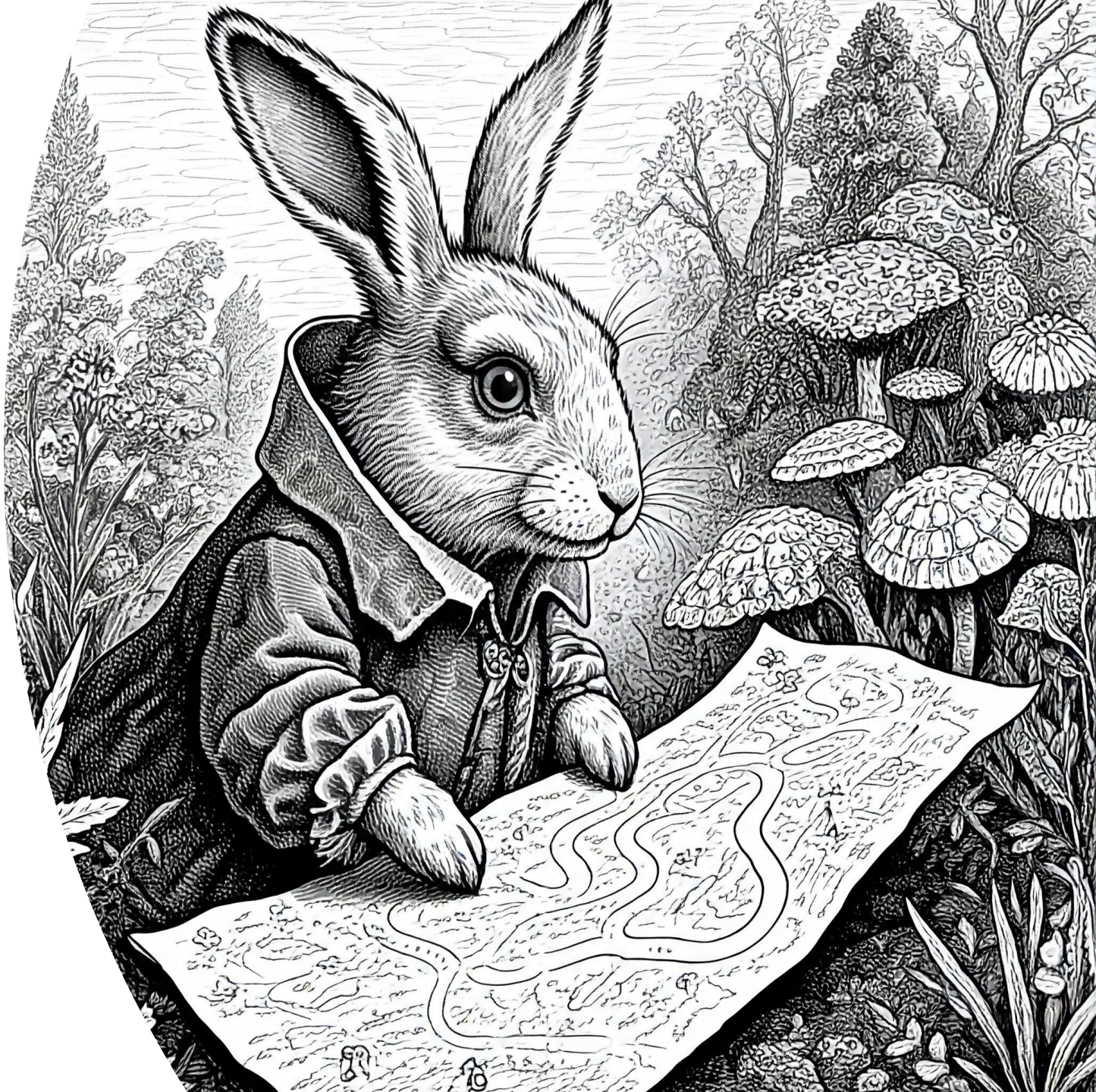
Revenons à nos moutons

```
final ReadableInterval validTime = value.getValidTime();
if (validTime.getStartMillis() < …) {
    // No exception here
}

…

if (validTime.getStartMillis() < …) {
    // NullPointerException
    // Cannot invoke "ReadableInterval.getStartMillis()"
    // because "validTime" is null
}
```

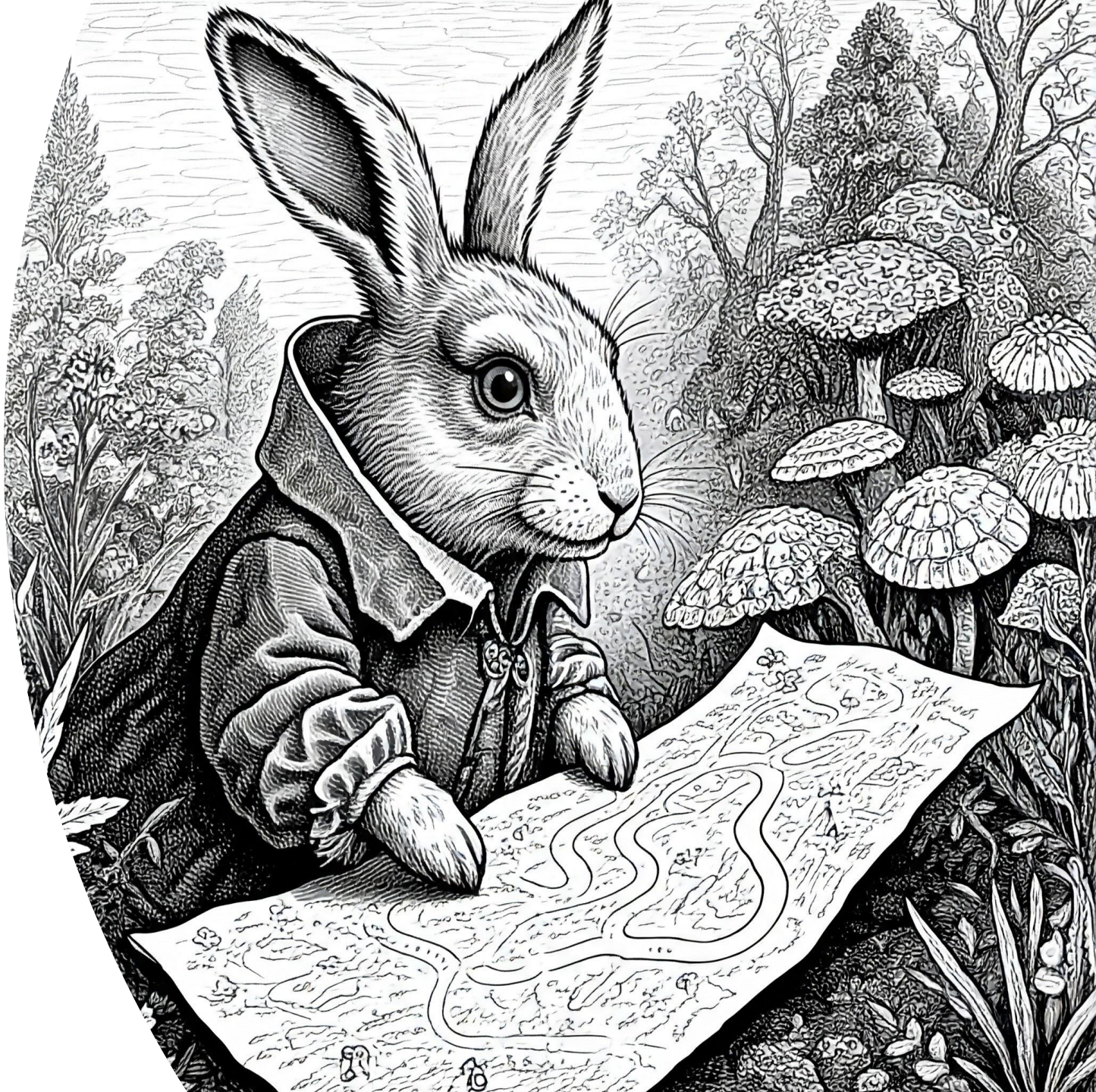
Et je fais quoi maintenant ?!?



Comment je sais si ça marche ?

- Si ça plante c'est facile, mais si ça ne plante pas…
- Idée: 10 exécutions qui marchent alors c'est bon
  - probabilité de 1/1024 de se tromper, ~0,1%

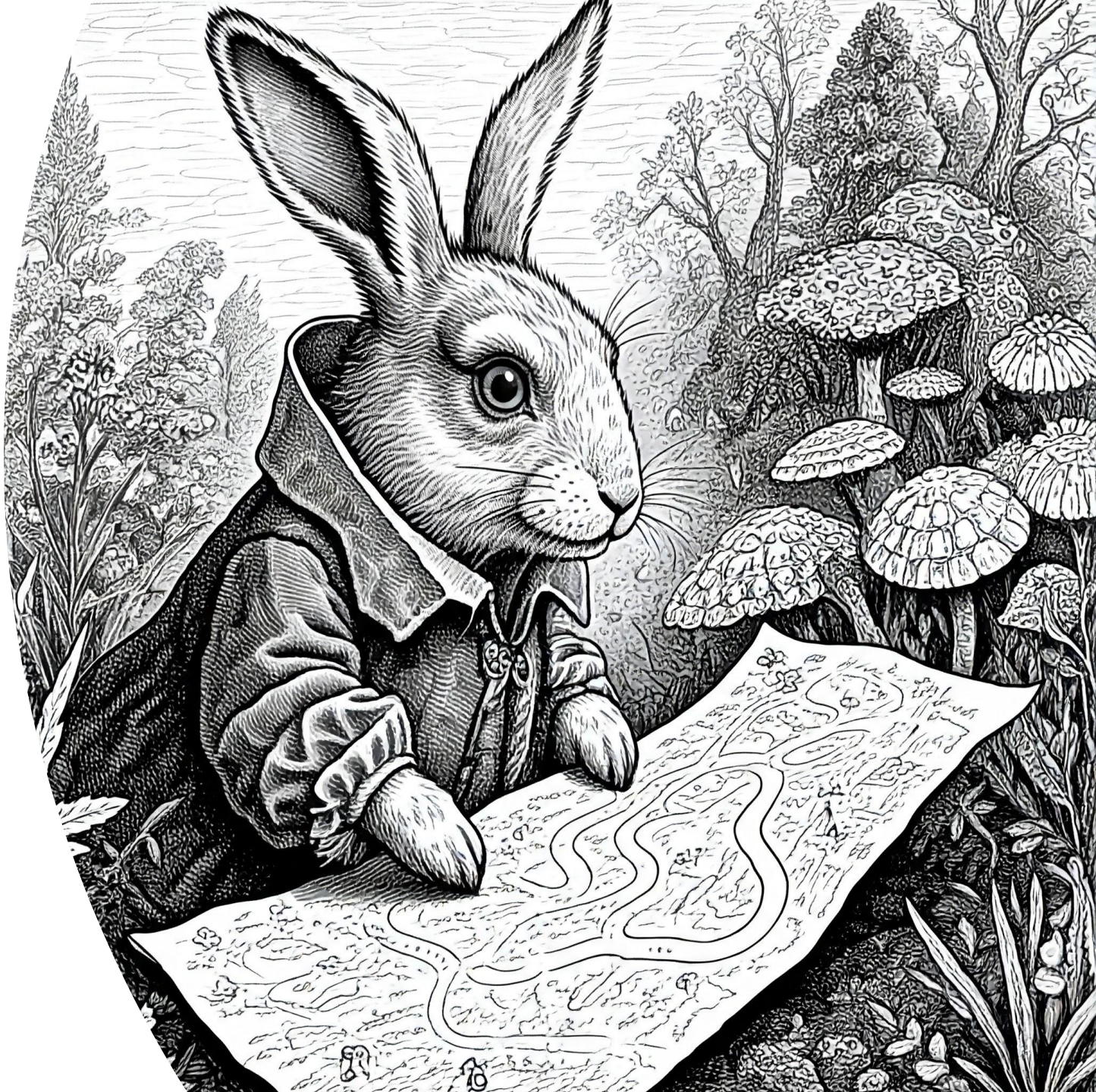
Et je fais quoi maintenant ?!?



Idée n°1 – Problème de simulateur

- On doit générer un jeu de données invalides
- Je rejoue les données d'un run qui passe, ça plante…
- … et inversement !
- Ce n'est pas un problème de données

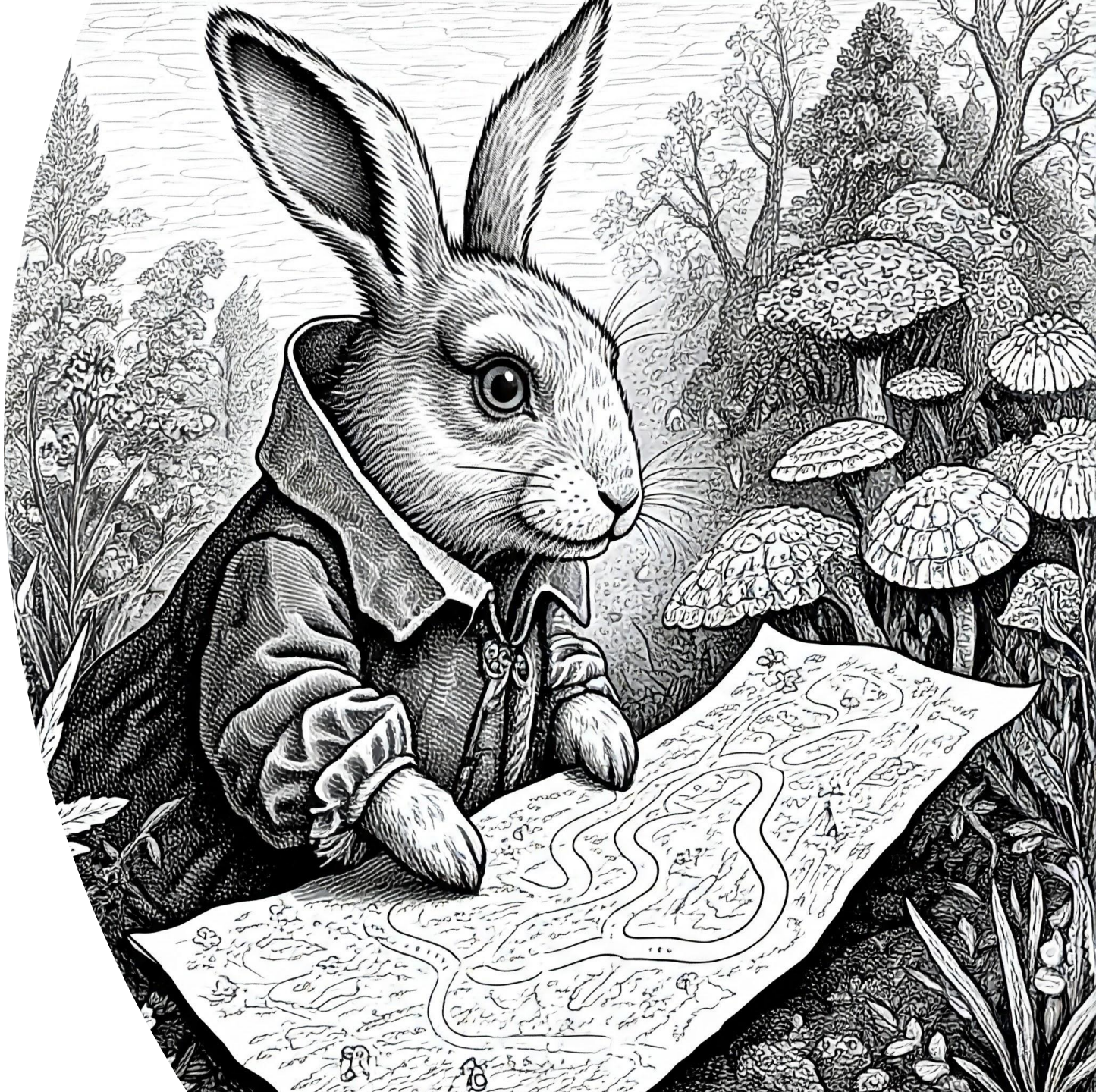
Et je fais quoi maintenant ?!?



Idée n°2 – Problème de serveur

- Le benchmark tourne toujours sur le même serveur
- Si ça se trouve, il y a un problème avec la mémoire, le CPU, le disque, le noyau linux, la version de docker…
- Ce n'est pas un problème de serveur

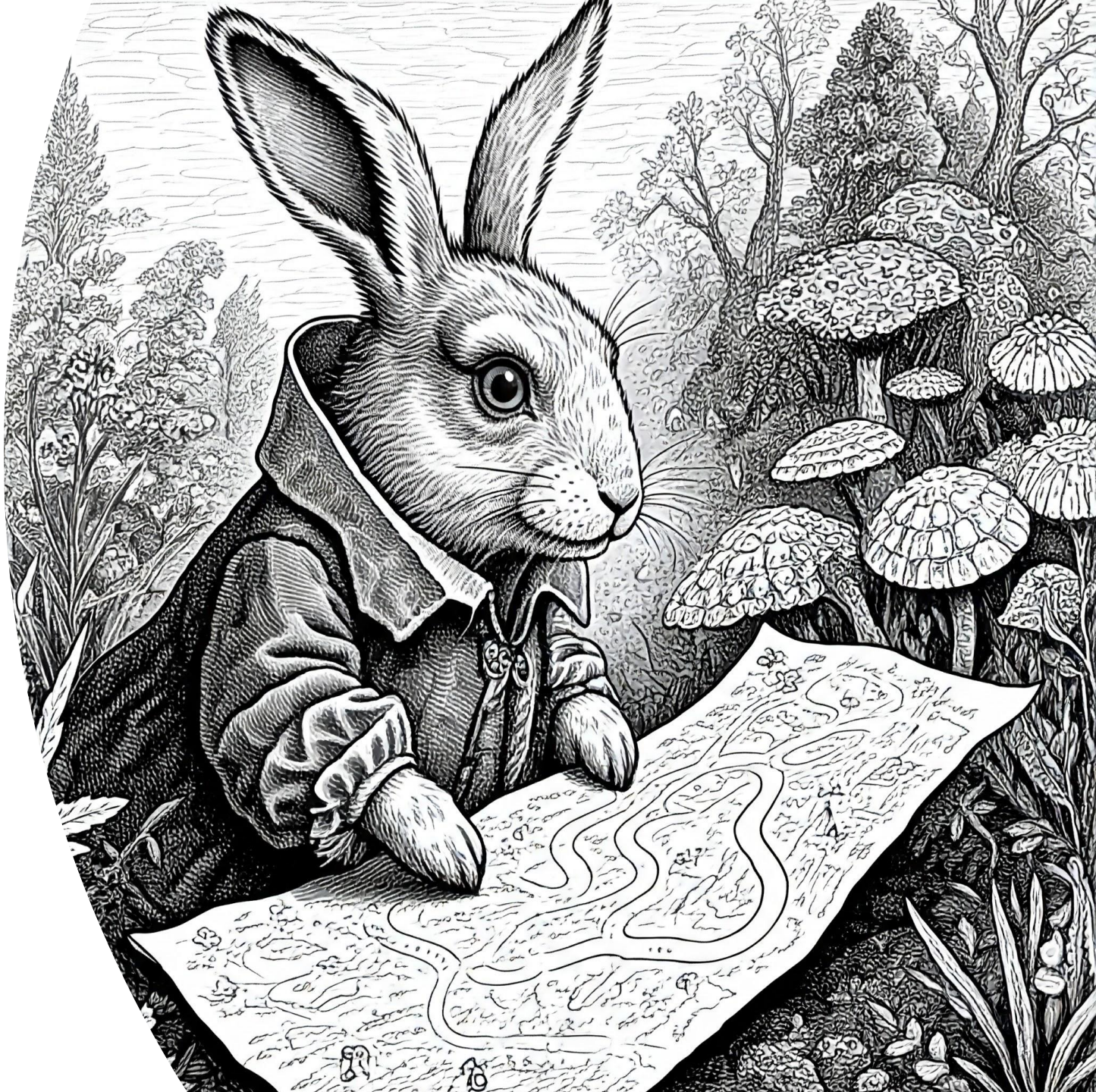
Et je fais quoi maintenant ?!?



Idée n°3 – Le benchmark n'est pas bon

- Le benchmark utilise un agent Java
- Si ça se trouve, l'agent n'aime pas Java 21
- Ce n'est donc pas un problème de benchmark

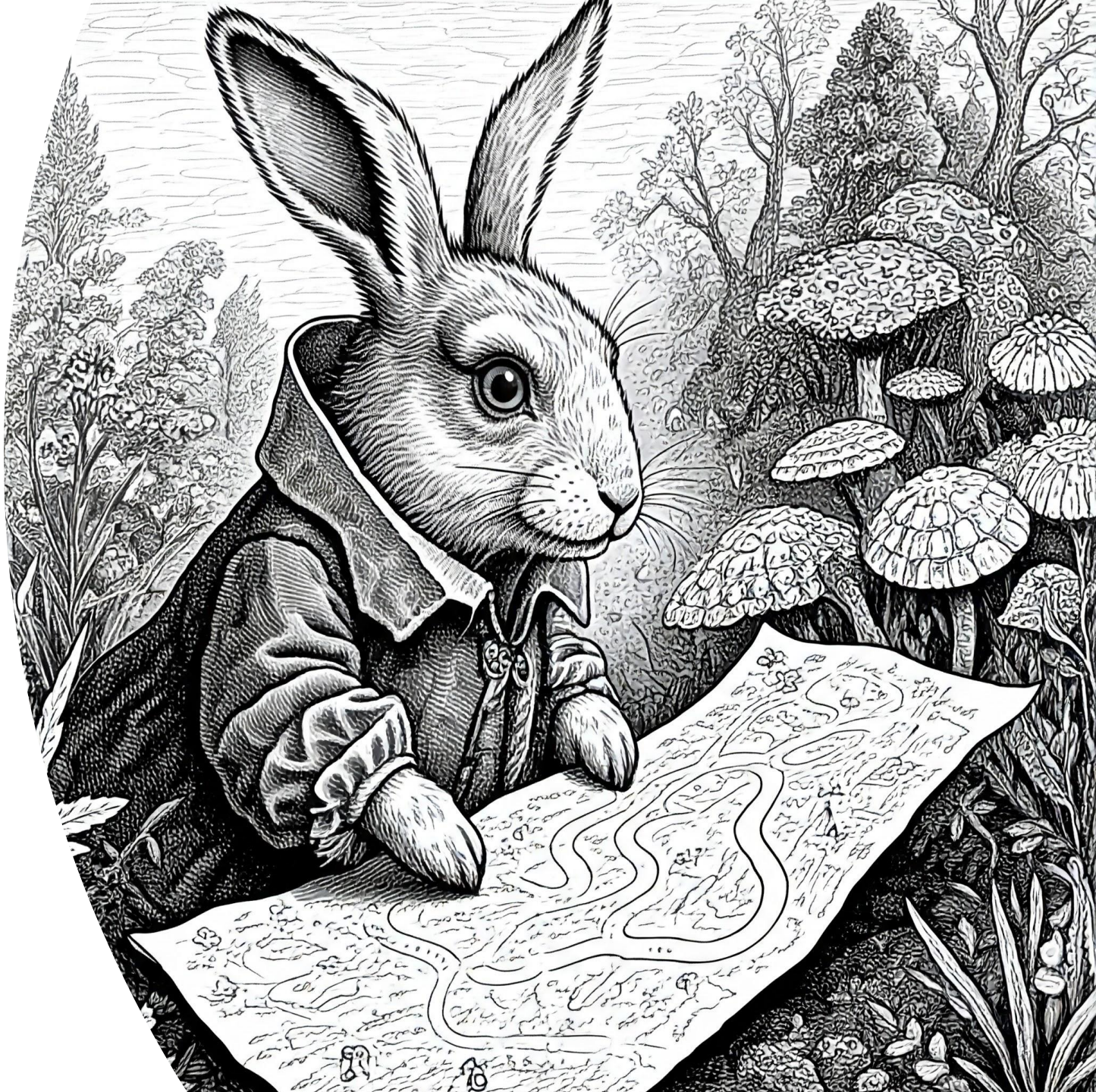
Et je fais quoi maintenant ?!?



Idée n°4 – Un bug de JVM ?

- 17.0.8 ✓
- 21.0.1 ✗
- 21 ✗
- 17.0.9 ✓
- Adoptium 21 nightly ✗
- Latest (23) ✗
- Aleksey Shipilëv 21 builds
  - release ✗
  - fastdebug ✗
  - slowdebug ✗
- 19.0.2 ✓
- Rien sur le bug tracker

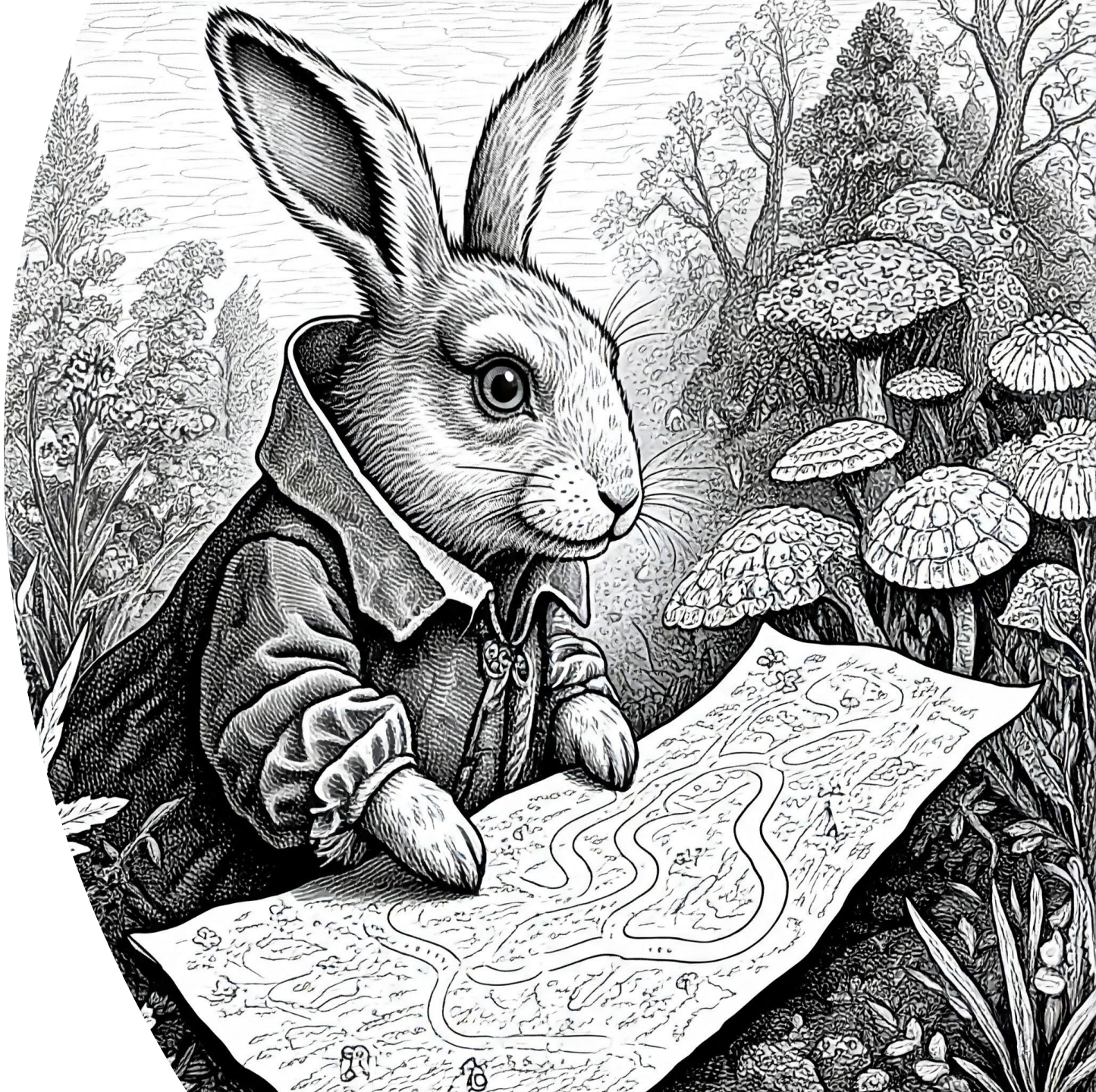
Et je fais quoi maintenant ?!?



Idée n°5 – Le GC libère trop tôt ?

- Si ca se trouve, le GC libère l'objet trop tôt ?
- Tests avec les différents GC
  - -XX:+UseParallelGC ✗
  - -XX:+UseG1GC ✗
  - -XX:+UseZGC ✗
  - -XX:+UseShenandoahGC ✗
  - -XX:+UseSerialGC ✗
- Ce n'est donc pas le GC

Et je fais quoi maintenant ?!?



Idée n°6 – Parfois le JIT génère des bugs

- Comme ça plante vers la fin, ça peut venir du code compilé par le JIT
- Comme ça ne plante pas toujours, le code généré doit être faux 50% du temps
- Je n'ai qu'à comparer 2 exécutions

FAUX !

(mais je ne le sais pas encore)

Comment récupérer le code généré ?

- -XX:+UnlockDiagnosticVMOptions
- -XX:CompileCommand="print package.Class::method"
- -XX:PrintAssemblyOptions=intel
- OpenJDK hsdis
  - disassembly plugin
  - dans le JDK ou en variable d'environnement
  - merci à Chris Newland pour sa compilation de la librairie

Ca ressemble à quoi ?

```
0x0000709395ed0480:   mov    DWORD PTR [rsp-0x14000],eax
0x0000709395ed0487:   push   rbp
0x0000709395ed0488:   sub    rsp,0x130
0x0000709395ed048f:   nop
0x0000709395ed0490:   cmp    DWORD PTR [r15+0x20],0x0
0x0000709395ed0498:   je     0x0000709395ed049f
0x0000709395ed049a:   call   Stub::nmethod_entry_barrier  ;   {runtime_call StubRoutines (final stubs)}
0x0000709395ed049f:   mov    QWORD PTR [rsp+0x80],rsi
0x0000709395ed04a7:   mov    QWORD PTR [rsp+0x70],rdx
0x0000709395ed04ac:   mov    QWORD PTR [rsp+0x88],rcx
0x0000709395ed04b4:   mov    DWORD PTR [rsp+0x68],r8d
0x0000709395ed04b9:   mov    DWORD PTR [rsp+0x98],r9d
0x0000709395ed04c1:   mov    DWORD PTR [rsp+0x6c],edi
0x0000709395ed04c5:   movabs rbx,0x708f09311508
0x0000709395ed04cf:   mov    eax,DWORD PTR [rbx+0x8]
0x0000709395ed04d2:   add    eax,0x2
0x0000709395ed04d5:   mov    DWORD PTR [rbx+0x8],eax
0x0000709395ed04d8:   and    eax,0xffe
0x0000709395ed04dd:   test   eax,eax
0x0000709395ed04df:   je     0x0000709395ed1220
0x0000709395ed04e5:   mov    ebx,DWORD PTR [rdx+0x18]     ; implicit exception: dispatches to 0x0000709395ed1241
…
```

Premières analyses

- Il y a beaucoup trop de lignes
- Les index à gauche ne correspondent pas
- Les pointeurs ne correspondent pas
- Impossible de comparer automatiquement

Les en-têtes de l'assembleur

```
================================ C1-compiled nmethod ================================
---------------------------------- Assembly ----------------------------------

Compiled method (c1) 1107312 37562       2     package.Class::method (530 bytes)
 total in heap  [0x0000709395ecff90,0x0000709395ed2480] = 9456
…
[Disassembly]
…
[/Disassembly]
```

Ça passe !

Les en-têtes de l'assembleur

```
============================= C2-compiled nmethod ==============================
----------------------------------- Assembly -----------------------------------

Compiled method (c2) 1142813 38977       4     package.Class::method (530 bytes)
 total in heap  [0x000070939eece990,0x000070939eed88c8] = 40760
…
[Disassembly]
…
[/Disassembly]
```

Ça plante !

Les en-têtes de l'assembleur

```
============================= C2-compiled nmethod ==============================
----------------------------------- Assembly -----------------------------------

Compiled method (c2) 1142813 38977       4     package.Class::method (530 bytes)
 total in heap  [0x000070939eece990,0x000070939eed88c8] = 40760
…
[Disassembly]
…
[/Disassembly]
```

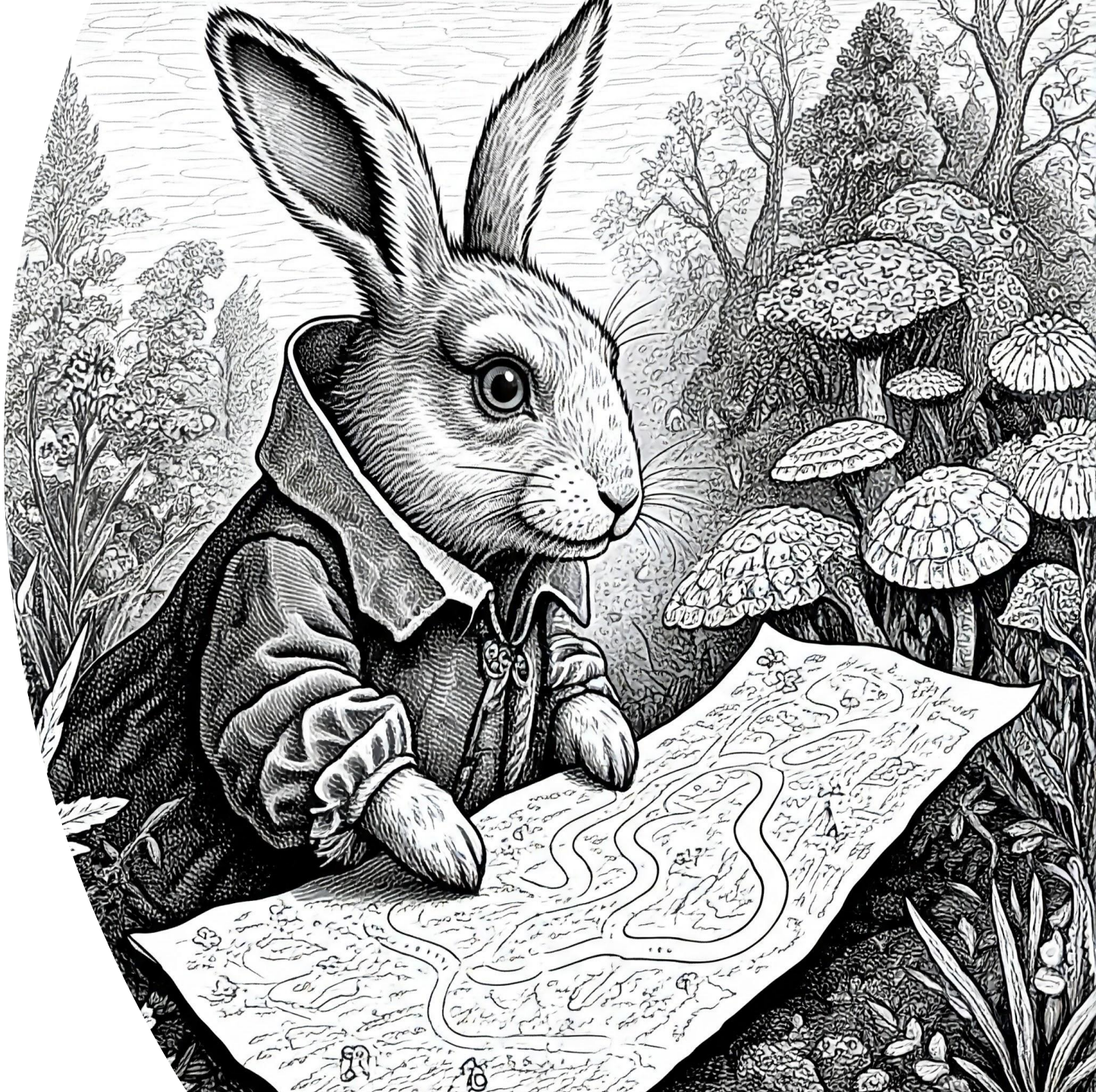
40ko !!!

(moi, je ne compare pas ça)

Les différentes options du JIT

- Aucun JIT
  - -Xint ✓
  - 35 minutes au lieu de 1 minute 30
- C1
  - -XX:TieredStopAtLevel=1 ✓
  - -XX:TieredStopAtLevel=2 ✓
  - -XX:TieredStopAtLevel=3 ✓
- C2 ✗

Et je fais quoi maintenant ?!?



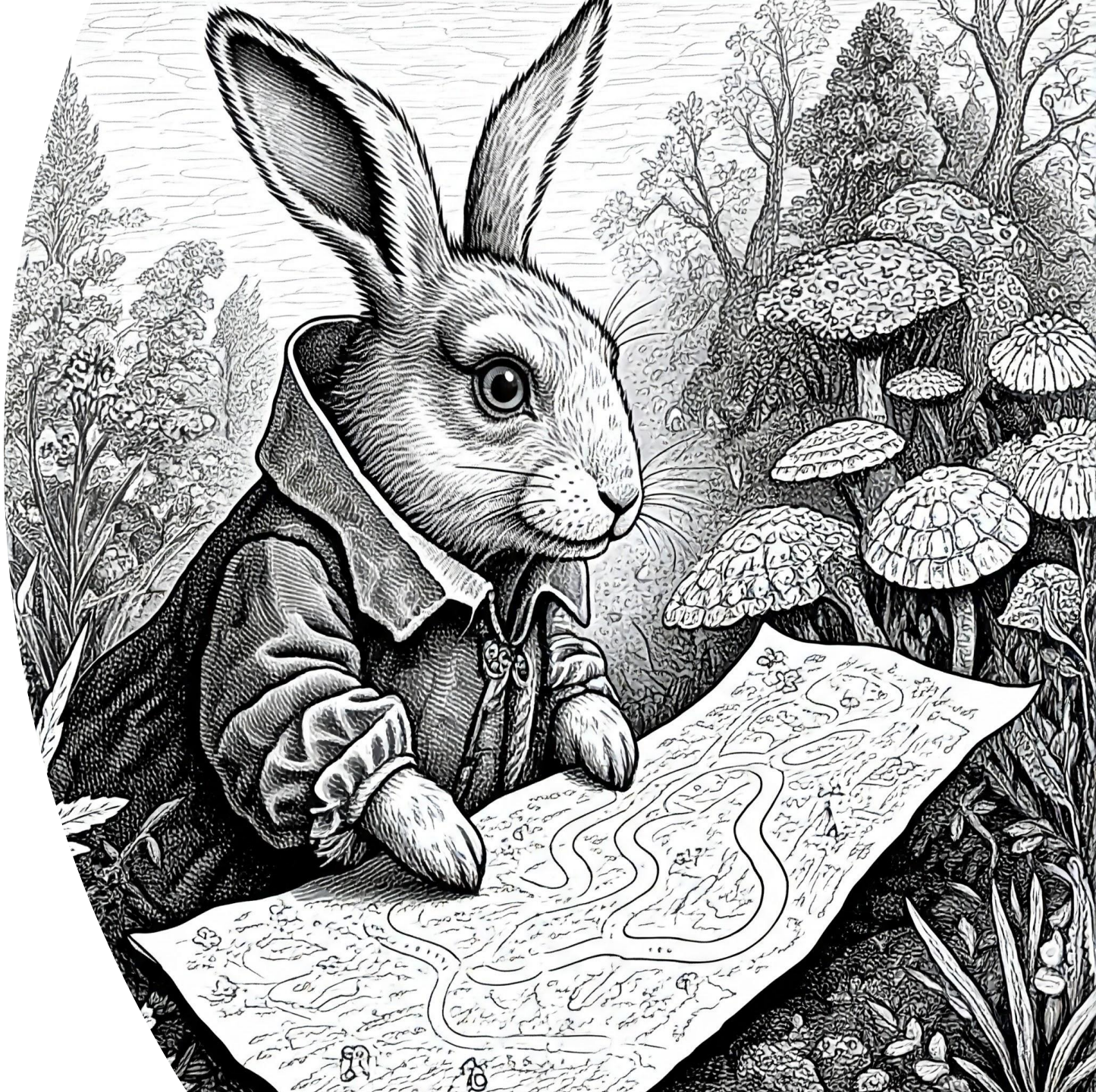
Idée n°7 – Test trop court ?

- Et si le plantage n'était pas aléatoire ?
- Si le benchmark n'avait pas le temps de planter ?
- Au lieu de rejouer 30 minutes de données, je tente 2 heures

Victoire !

(enfin… ça plante à tous les coups)

Et je fais quoi maintenant ?!?



Idée n°8 – Et si je trouvais le commit ?

- Je vais utiliser git bisect pour trouver le commit responsable de la régression
- Ah, mais comment on build Java ?

Comment builder Java ?

```
$ git clone https://github.com/openjdk/jdk.git
Cloning into 'jdk'...

$ cd jdk

$ git checkout jdk-21-ga
Updating files: 100% (27123/27123), done.
```

Comment builder Java ?

```
$ bash configure
…
Autoconf is not found on the PATH
You might be able to fix this by running 'sudo apt-get
install autoconf’.

$ sudo apt-get install autoconf
```

Comment builder Java ?

```
$ bash configure
Using autoconf at /usr/bin/autoconf
…
configure: error: Could not find required tool for FILE

$ sudo apt-get install file
```

Comment builder Java ?

```
$ bash configure
checking for file... /usr/bin/file
…
configure: error: Cannot find GNU make 3.81 or newer!
Please put it in the path, or add e.g.
MAKE=/opt/gmake3.81/make as argument to configure.

$ sudo apt-get install make
$ sudo apt-get install unzip
$ sudo apt-get install zip
```

Comment builder Java ?

```
$ bash configure
…
configure: Could not find a valid Boot JDK. OpenJDK
distributions are available at http://jdk.java.net/.
configure: This might be fixed by explicitly setting
--with-boot-jdk
```

Comment builder Java ?

```
$ bash configure --with-boot-jdk=/opt/jdk-20
…
configure: error: Could not find a C compiler. You might
be able to fix this by running 'sudo apt-get install
build-essential’.

$ sudo apt-get install build-essential
$ sudo apt-get install …
```

Comment builder Java ?

```
$ bash configure --with-boot-jdk=/opt/jdk-20
…
A new configuration has been successfully created in
/home/user/jdk/build/linux-x86_64-server-release
using configure arguments '--with-boot-jdk=/opt/jdk-20’.

Configuration summary:
* Name:              linux-x86_64-server-release
* Debug level:       release
…
```

Comment builder Java ?

```
$ make images
…
Finished building target 'images' in configuration 'linux-
x86_64-server-release’

$ cd build/linux-x86_64-server-release/jdk/bin

$ ./java --version
openjdk 21-internal 2023-09-19
```

Comment builder Java ?

- On installe tout ce dont il a besoin
  - outils, librairies X11, librairies son, …
  - il fournit même les commandes à taper
- 5 minutes pour installer toutes les dépendances
- 5 minutes pour builder Java
- Facile de créer un environnement tout prêt

A la recherche du commit

```
[jdk-21-ga] $ git bisect start

[jdk-21-ga] $ git bisect bad
status: waiting for good commit(s), bad commit known

[jdk-21-ga] $ git bisect good jdk-19-ga
Bisecting: 2441 revisions left to test after this (roughly
11 steps)
[2ccdefc81c0ea2ea5c4380bb045aa82ad1eb8205] 8299470: …
```

A la recherche du commit

```
[2ccdefc81c0] $ make images
[2ccdefc81c0] $ ~/bench/test.sh
Crash!

[2ccdefc81c0] $ git bisect bad
Bisecting: 1219 revisions left to test after this (roughly
10 steps)
[090cdfc7a2e280c620a0926512fb67f0ce7f3c21] 8294726: …
```

A la recherche du commit

```
[090cdfc7a2e] $ make images
[090cdfc7a2e] $ ~/bench/test.sh
OK!

[090cdfc7a2e] $ git bisect good
Bisecting: 609 revisions left to test after this (roughly
9 steps)
[761a4f4852cbb40660b6fb9eda4d740464218f75] 8295788: …
```

A la recherche du commit

```
[761a4f4852c] $ make images
[761a4f4852c] $ ~/bench/test.sh
OK!

[761a4f4852c] $ git bisect good
Bisecting: 304 revisions left to test after this (roughly
8 steps)
[27bbe7be2c43a22e8cf55aa403d8018346ae3e37] 8297976: …
```

A la recherche du commit

```
[27bbe7be2c4] $ make images
[27bbe7be2c4] $ ~/bench/test.sh
OK!

[27bbe7be2c4] $ git bisect good
Bisecting: 152 revisions left to test after this (roughly
7 steps)
[f4caaca100d334b671eed56287dfe7a1009c47d7] 8298852: …
```

A la recherche du commit

```
[f4caaca100d] $ make images
[f4caaca100d] $ ~/bench/test.sh
OK!

[f4caaca100d] $ git bisect good
Bisecting: 73 revisions left to test after this (roughly 6
steps)
[34cdda5b8359cce33c2d4f92a41a620aea4f96e7] Merge
```

A la recherche du commit

```
[34cdda5b835] $ make images
[34cdda5b835] $ ~/bench/test.sh
Crash!

[34cdda5b835] $ git bisect bad
Bisecting: 42 revisions left to test after this (roughly 5
steps)
[e5edb10dc56d9edac8e050e0f8e6c116743975d6] 8299018: …
```

A la recherche du commit

```
[e5edb10dc56] $ make images
[e5edb10dc56] $ ~/bench/test.sh
Crash!

[e5edb10dc56] $ git bisect bad
Bisecting: 17 revisions left to test after this (roughly 4
steps)
[5e678f7500e514f04637c546959613d4688f989c] 8298824: …
```

A la recherche du commit

```
[5e678f7500e] $ make images
[5e678f7500e] $ ~/bench/test.sh
Crash!

[5e678f7500e] $ git bisect bad
Bisecting: 8 revisions left to test after this (roughly 3
steps)
[0ba473489151d74c8a15b75ff4964ac480fecb28] 8287699: …
```

A la recherche du commit

```
[0ba47348915] $ make images
[0ba47348915] $ ~/bench/test.sh
OK!

[0ba47348915] $ git bisect good
Bisecting: 4 revisions left to test after this (roughly 2
steps)
[0eeaeb8e7ba40be5e93eb87c7e3dc94230062746] 8298808: …
```

A la recherche du commit

```
[0eeaeb8e7ba] $ make images
[0eeaeb8e7ba] $ ~/bench/test.sh
Crash!

[0eeaeb8e7ba] $ git bisect bad
Bisecting: 1 revision left to test after this (roughly 1
step)
[3696711efa566fb776d6923da86e17b0e1e22964] Merge
```

A la recherche du commit

```
[3696711efa5] $ make images
[3696711efa5] $ ~/bench/test.sh
Crash!

[3696711efa5] $ git bisect bad
Bisecting: 0 revisions left to test after this (roughly 0
steps)
[f771c56e16a39724712ca0d8c2dd55b9ce260f4d] 8298797: …
```

A la recherche du commit

```
[f771c56e16a] $ make images
[f771c56e16a] $ ~/bench/test.sh
OK!

[f771c56e16a] $ git bisect good
3696711efa566fb776d6923da86e17b0e1e22964 is the first bad
commit
Merge: f4caaca100d f771c56e16a
```

A la recherche du commit

```
[3696711efa5] $ git log --oneline --graph
*   3696711efa5 (HEAD) Merge
|\
| * f771c56e16a 8298797: …
| * …
| * 323e574a505 8298371: …
* | f4caaca100d 8298852: …
* | …
* | 10bc86cc260 Merge
|\|
```

A la recherche du commit

```
[3696711efa5] $ git bisect start
[3696711efa5] $ git bisect bad
status: waiting for good commit(s), bad commit known

[3696711efa5] $ git checkout 10bc86cc260
```

A la recherche du commit

```
[10bc86cc260] $ make images
[10bc86cc260] $ ~/bench/test.sh
OK!

[10bc86cc260] $ git bisect good
Bisecting: 15 revisions left to test after this (roughly 4
steps)
[e41686b4050d6b32fb451de8af39a78ec8bed0fd] 8298710: …
```

A la recherche du commit

```
[e41686b4050] $ make images
[e41686b4050] $ ~/bench/test.sh
Crash!

[e41686b4050] $ git bisect bad
Bisecting: 7 revisions left to test after this (roughly 3
steps)
[0ef353925e645dd519e17aeb7a83e927271f8b95] 8298416: …
```

A la recherche du commit

```
[0ef353925e6] $ make images
[0ef353925e6] $ ~/bench/test.sh
OK!

[0ef353925e6] $ git bisect good
Bisecting: 3 revisions left to test after this (roughly 2
steps)
[3cdbd878e68dc1131093137a7357710ad303ae8c] 8298241: …
```

A la recherche du commit

```
[3cdbd878e68] $ make images
[3cdbd878e68] $ ~/bench/test.sh
Crash!

[3cdbd878e68] $ git bisect bad
Bisecting: 1 revision left to test after this (roughly 1
step)
[4b313b51b1787113961c289a41708e31fa19cacc] 8297798: …
```

A la recherche du commit

```
[4b313b51b17] $ make images
[4b313b51b17] $ ~/bench/test.sh
OK!

[4b313b51b17] $ git bisect good
Bisecting: 0 revisions left to test after this (roughly 0
steps)
[10737e168c967a08e257927251861bf2c14795ab] 8298468: …
```

A la recherche du commit

```
[10737e168c9] $ make images
[10737e168c9] $ ~/bench/test.sh
Crash!

[10737e168c9] $ git bisect bad
10737e168c967a08e2579272518 61bf2c14795ab is the first bad
commit
```

A la recherche du commit

```
commit 10737e168c967a08e257927251861bf2c14795ab
Author: xxx <xxx@openjdk.org>
Date:   Thu Dec 15 19:54:25 2022 +0000

    8298468: Clean up class_loader parameters

    Reviewed-by: yyy, zzz
```

A la recherche du commit

```
src/hotspot/share/cds/cdsProtectionDomain.cpp      |  3
src/hotspot/share/classfile/loaderConstraints.cpp  | 70
src/hotspot/share/classfile/loaderConstraints.hpp  | 13
src/hotspot/share/classfile/systemDictionary.cpp   | 74
src/hotspot/share/classfile/systemDictionary.hpp   |  8
5 files changed, 76 insertions(+), 92 deletions(-)
```

Commit trouvé !

src/hotspot/share/classfile/loaderConstraints.cpp +31 -39

```
@@ -47,11 +47,11 @@ class LoaderConstraint : public CHeapObj<mtClass> {
47  47    // not class loaders.
48  48    GrowableArray<ClassLoaderData*>*  _loaders;               // initiating loaders
49  49   public:
50      -  LoaderConstraint(InstanceKlass* klass, oop class_loader1, oop class_loader2) :
    50  +  LoaderConstraint(InstanceKlass* klass, ClassLoaderData* loader1, ClassLoaderData* loader2) :
51  51       _klass(klass) {
52  52    _loaders = new (mtClass) GrowableArray<ClassLoaderData*>(10, mtClass);
53      -    add_loader(class_loader1);
54      -    add_loader(class_loader2);
    53  +    add_loader_data(loader1);
    54  +    add_loader_data(loader2);
55  55    }
56  56    LoaderConstraint(const LoaderConstraint& src) = delete;
57  57    LoaderConstraint& operator=(const LoaderConstraint&) = delete;
@@ -61,7 +61,7 @@ class LoaderConstraint : public CHeapObj<mtClass> {
61  61    InstanceKlass* klass() const     { return _klass; }
62  62    void set_klass(InstanceKlass* k) { _klass = k; }
```

Commit trouvé !

src/hotspot/share/classfile/loaderConstraints.cpp +31 -39

```
164  157    // Either add it to an existing entry in the table or make a new one.
165         - void LoaderConstraintTable::add_loader_constraint(Symbol* name, InstanceKlass* klass, oop class_loader1, oop class_loader2) {
       158  + void LoaderConstraintTable::add_loader_constraint(Symbol* name, InstanceKlass* klass,
       159  +                                                    ClassLoaderData* loader1, ClassLoaderData* loader2) {
166    160      assert_lock_strong(SystemDictionary_lock);
167         -    LoaderConstraint* constraint = new LoaderConstraint(klass, class_loader1, class_loader2);
       161  +    LoaderConstraint* constraint = new LoaderConstraint(klass, loader1, loader2);
168    162
169    163      // The klass may be null if it hasn't been loaded yet, for instance while checking
170    164      // a parameter name to a method call.  We impose this constraint that the
                @@ -256,22 +250,22 @@ void LoaderConstraintTable::purge_loader_constraints() {
256    250    }
257    251
258    252    void log_ldr_constraint_msg(Symbol* class_name, const char* reason,
259         -                              Handle class_loader1, Handle class_loader2) {
       253  +                              ClassLoaderData* loader1, ClassLoaderData* loader2) {
260    254      LogTarget(Info, class, loader, constraints) lt;
```

Commit trouvé !

src/hotspot/share/classfile/loaderConstraints.cpp +31 -39

```
265  259                      class_name->as_C_string(),
266       -                   ClassLoaderData::class_loader_data(class_loader1())->loader_name_and_id(),
267       -                   ClassLoaderData::class_loader_data(class_loader2())->loader_name_and_id(),
     260  +                   loader1->loader_name_and_id(),
     261  +                   loader2->loader_name_and_id(),
268  262                      reason);
269  263        }
270  264      }
271  265
272  266    bool LoaderConstraintTable::add_entry(Symbol* class_name,
273       -                                        InstanceKlass* klass1, Handle class_loader1,
274       -                                        InstanceKlass* klass2, Handle class_loader2) {
     267  +                                        InstanceKlass* klass1, ClassLoaderData* loader1,
     268  +                                        InstanceKlass* klass2, ClassLoaderData* loader2) {
275  269
276  270      LogTarget(Info, class, loader, constraints) lt;
277  271      if (klass1 != NULL && klass2 != NULL) {
     @@ -282,35 +276,35 @@ bool LoaderConstraintTable::add_entry(Symbol* class_name,
```

Le ticket associé

JDK / JDK-8298468

Clean up class_loader parameters

Resolved

Export

Details

Type:	Enhancement	Resolution:	Fixed
Priority:	P4	Fix Version/s:	21
Affects Version/s:	21		
Component/s:	hotspot		
Labels:	classloading		

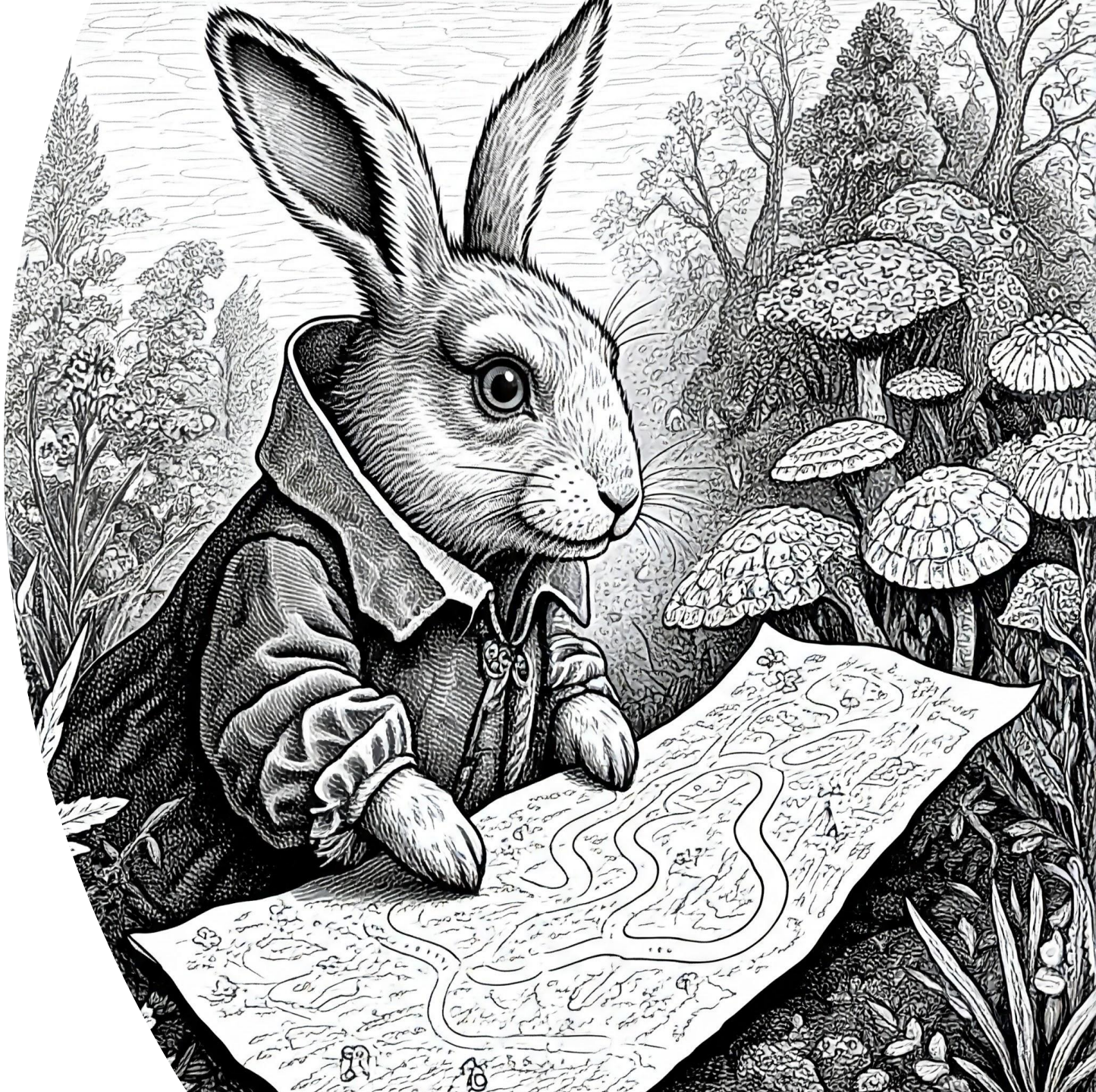
Description

The LoaderConstraintTable stores ClassLoaderData, but passes class_loader objects. There's a lot of unnecessary conversion back and forth that can be cleaned up.

Issue Links

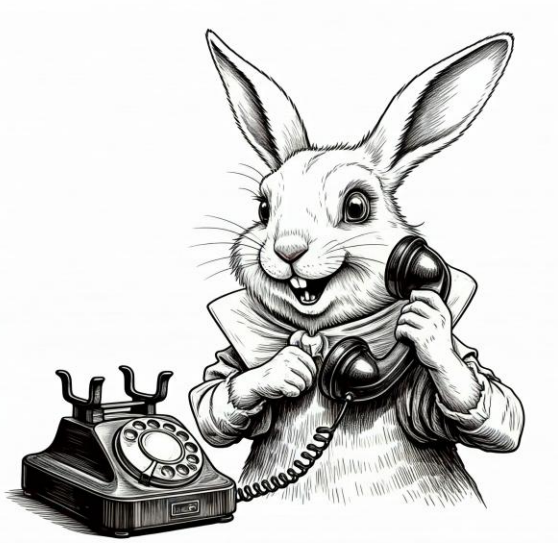
links to

Et je fais quoi maintenant ?!?

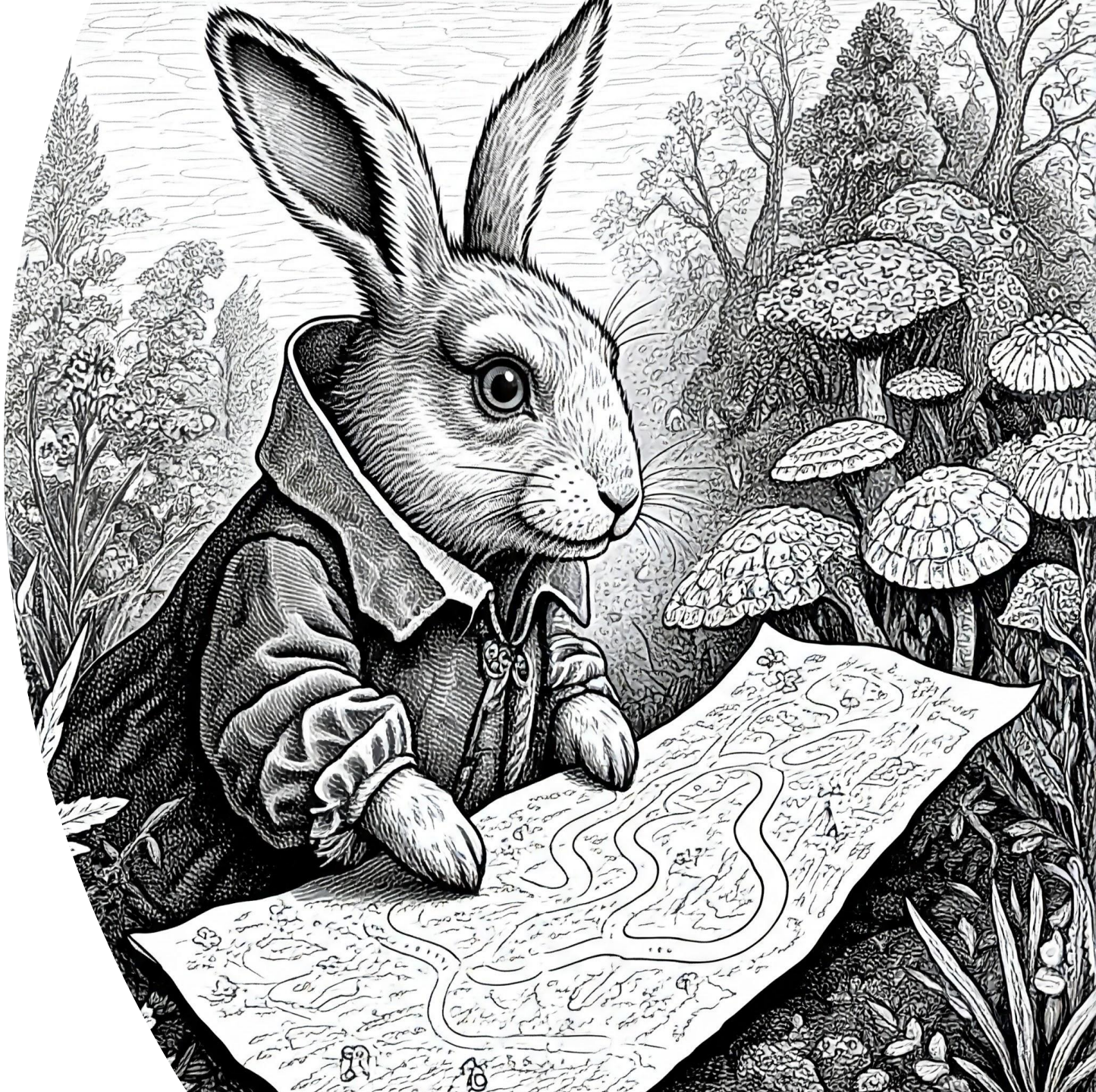


Appeler à l'aide

- Commit author
- Aleksey Shipilëv
- Nicolai Parlog
- Rémi Forax
- Mailing list java



Et je fais quoi maintenant ?!?



Relecture du commit

src/hotspot/share/classfile/loaderConstraints.cpp +31 -39

```
348  340       } else if (pp1 == NULL) {
349       -      pp2->extend_loader_constraint(class_name, class_loader1, klass);
          341 +    pp2->extend_loader_constraint(class_name, loader1, klass);
350  342       } else if (pp2 == NULL) {
351       -      pp1->extend_loader_constraint(class_name, class_loader2, klass);
          343 +    pp1->extend_loader_constraint(class_name, loader1, klass);
352  344       } else {
353  345         merge_loader_constraints(class_name, pp1, pp2, klass);
354  346       }
          @@ -359,7 +351,7 @@ bool LoaderConstraintTable::add_entry(Symbol* class_name,
359  351     // return true if the constraint was updated, false if the constraint is
360  352     // violated
361  353     bool LoaderConstraintTable::check_or_update(InstanceKlass* k,
362       -                                            Handle loader,
          354 +                                        ClassLoaderData* loader,
363  355                                               Symbol* name) {
364  356       LogTarget(Info, class, loader, constraints) lt;
365  357       LoaderConstraint* p = find_loader_constraint(name, loader);
```

Relecture du commit

- Ça ressemble à une typo
- Si je mets loader2, est-ce que ca marche ?

Victoire !

(reste plus qu'à l'intégrer)

Création de la pull request

- Signer le Oracle Contributor Agreement
  - partage de la propriété intellectuelle avec Oracle
- Il faut créer un bug sur le tracker d'abord
- J'ai demandé à Aleksey Shipilëv de le faire
  - JDK-8322282

Création du bug

JDK / JDK-8322282

Incorrect LoaderConstraintTable::add_entry after JDK-8298468

Closed

Details

Type:	Bug	Resolution:	Fixed
Priority:	P2	Fix Version/s:	23
Affects Version/s:	21, 22, 23		
Component/s:	hotspot		

Description

As reported here by Antoine Dessaigne:
https://mail.openjdk.org/pipermail/hotspot-compiler-dev/2023-December/071086.html

There is a regression introduced by ~~JDK-8298468~~:
https://github.com/openjdk/jdk/commit/10737e168c967a08e257927251861bf2c14795ab#diff-49d18b7f77db80ff67872c7db1e4c87e98a126e63fff5f63353e4406db332b19R343

...which looks like a typo.

Création de la pull request

4 src/hotspot/share/classfile/loaderConstraints.cpp

Viewed

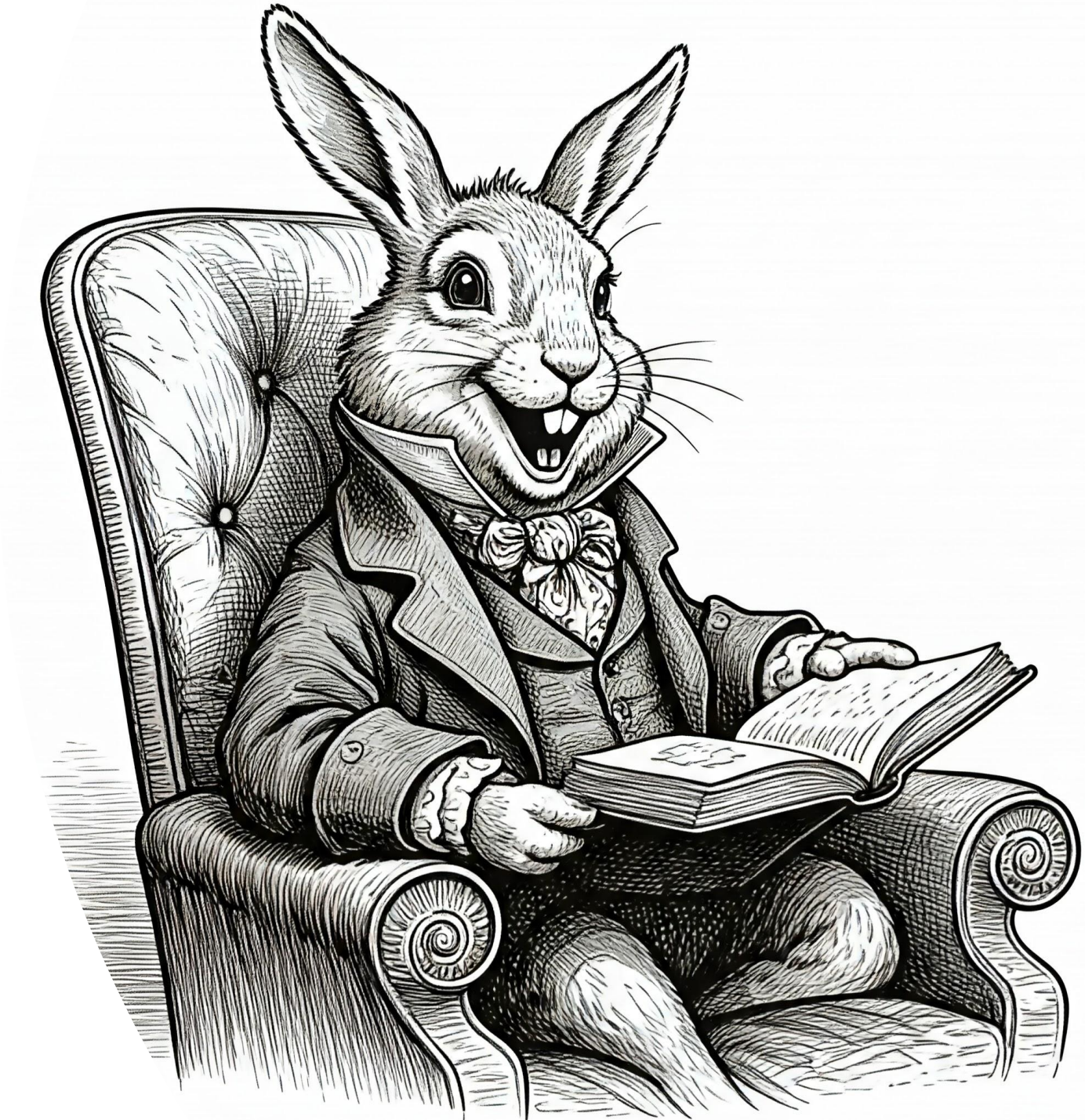
```
...  ...   @@ -1,5 +1,5 @@
 1    1    /*
 2         - * Copyright (c) 2003, 2023, Oracle and/or its affiliates. All rights reserved.
      2    + * Copyright (c) 2003, 2024, Oracle and/or its affiliates. All rights reserved.
 3    3     * DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS FILE HEADER.
 4    4     *
 5    5     * This code is free software; you can redistribute it and/or modify it
      @@ -387,7 +387,7 @@ bool LoaderConstraintTable::add_entry(Symbol* class_name,
387  387      } else if (pp1 == nullptr) {
388  388        pp2->extend_loader_constraint(class_name, loader1, klass);
389  389      } else if (pp2 == nullptr) {
390       -    pp1->extend_loader_constraint(class_name, loader1, klass);
     390  +    pp1->extend_loader_constraint(class_name, loader2, klass);
391  391      } else {
392  392        merge_loader_constraints(class_name, pp1, pp2, klass);
393  393      }
```

Corrigé en 21.0.3

Bilan

- J'étais perdu mais en fait c'était simple
- C'est très facile de compiler Java
  - Je m'en suis servi une seconde fois
- J'ai appris plein de choses au passage
  - les différents niveaux de JIT
  - comment voir l'assembleur généré
- Attention aux commits de refactor
  - même s'il y a une code review…

Merci pour votre écoute



Antoine DESSAIGNE

Slides