WORLD OF LOGGING

Welcome

Talk is about Logging in General

We´ll not into specific code

1. Logging in general
2. What we do
3. Who we are
4. What we might do

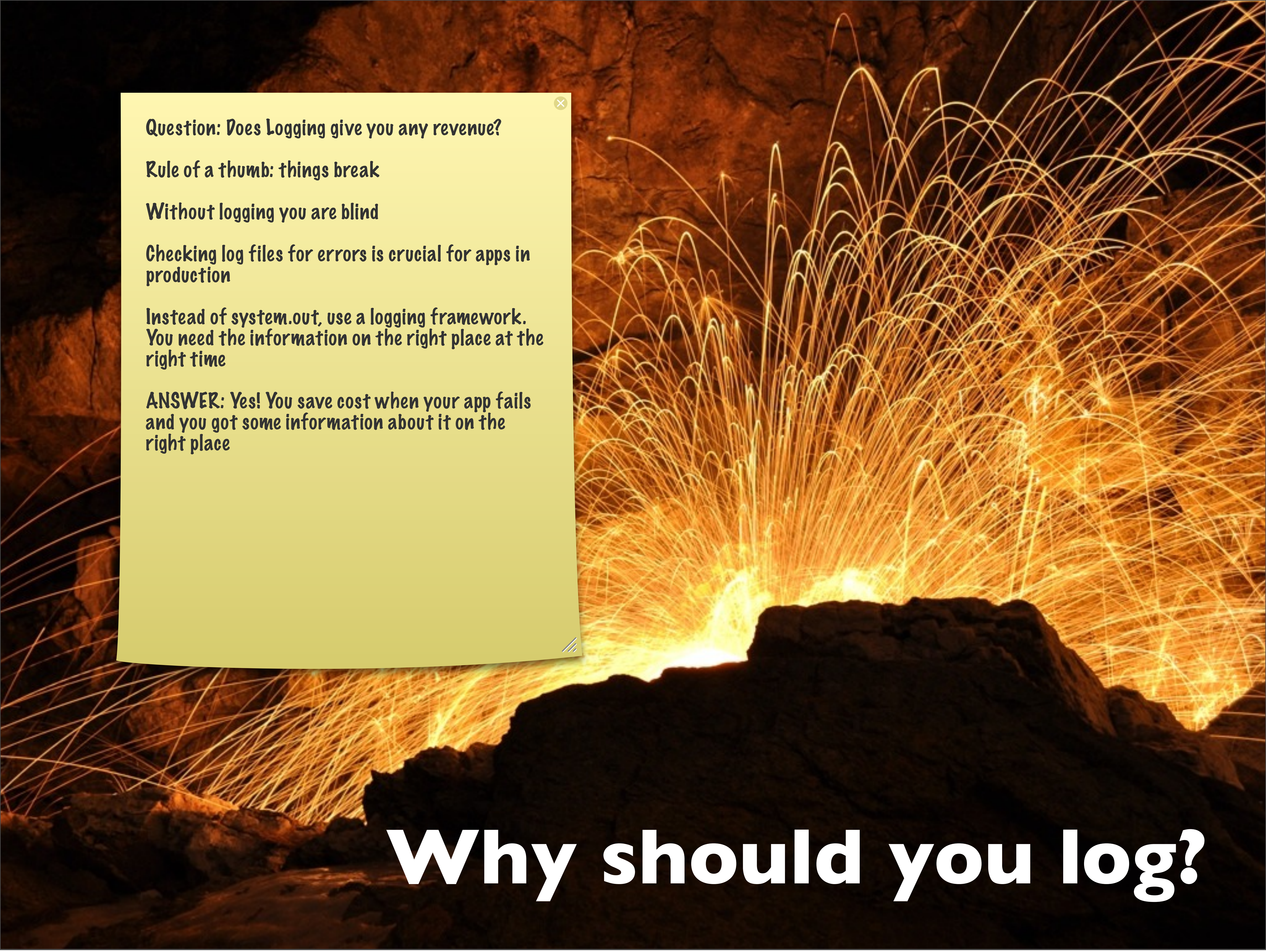
That's me.

Joined logging because I wanted to see a PHP project on ASF

Stay in touch with twitter, if you like

Christian Grobmeier

@grobmeier
http://www.grobmeier.de



Question: Does Logging give you any revenue?

Rule of a thumb: things break

Without logging you are blind

Checking log files for errors is crucial for apps in production

Instead of system.out, use a logging framework. You need the information on the right place at the right time

ANSWER: Yes! You save cost when your app fails and you got some information about it on the right place

Why should you log?

THE BIGGEST CHALLENGES WE HAVE

Logging should not take out (much) application performance

Logging framework need an „off button" to gain more power

SPEED!

Failsafe. Reliable.

Logging Frameworks must not fail. They must not bring your app into trouble

Bad example: log4j 1 has known memory leaks

They must log everything we want - lost log messages are bad

Example: hot reconfiguration - logback looses messages, log4j 2 not. Reason: Config is bound to the appender

Compability.

Version should not conflict as people might use different versions of the same api

configuration should not change as people do upgrade in an instant

People don't want to learn so much about logging. What they know must not be invalidated

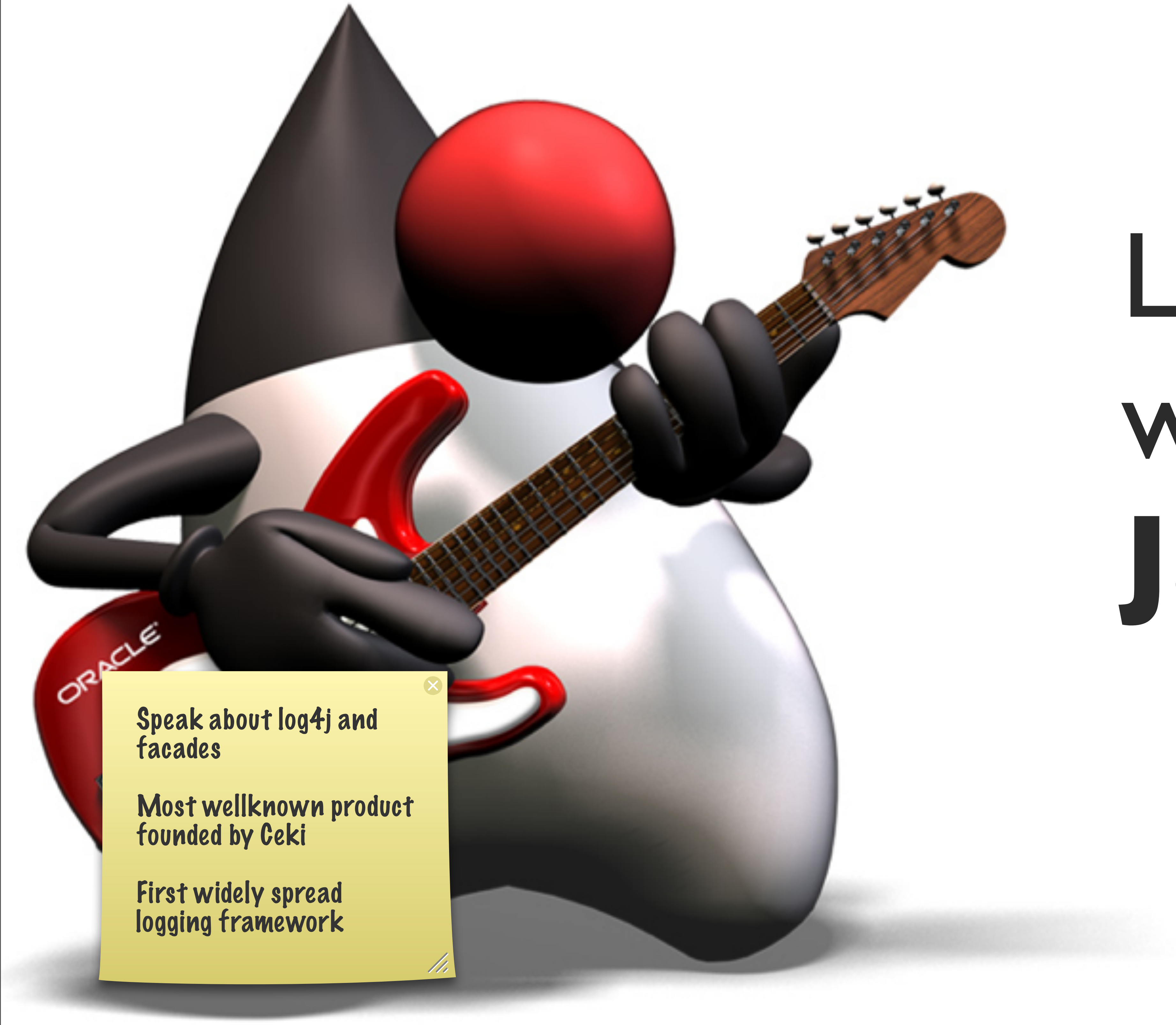
Logging APIs are a matter of taste.

There is no „perfect“ solution. But we developers consider „easy“ solutions as most perfect

We don‘t want to care about logging

Taste.

Logging with Duke Java



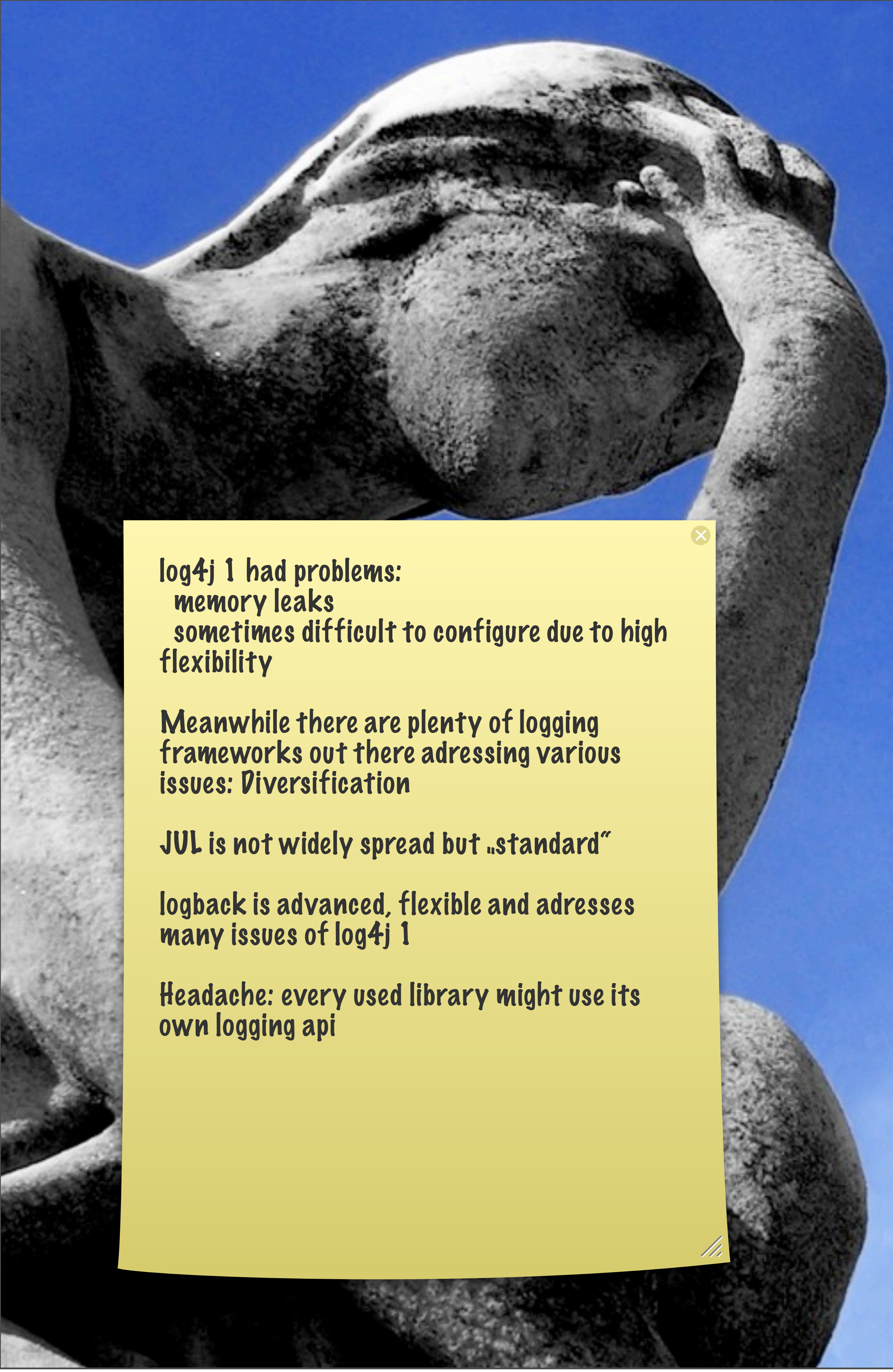
Speak about log4j and facades

Most wellknown product founded by Ceki

First widely spread logging framework

ORACLE

oracle.com/java



log4j 1 had problems:
memory leaks
sometimes difficult to configure due to high flexibility

Meanwhile there are plenty of logging frameworks out there adressing various issues: Diversification

JUL is not widely spread but „standard“

logback is advanced, flexible and adresses many issues of log4j 1

Headache: every used library might use its own logging api

Too much taste... makes headache!

Think on:
log4j, logback, JUL,
TinyLog, AVSL

Pills:

slf4j & Commons Logging

Facades abstract the implementation.

2 known facades

Libs should always use such a facade and never an API directly

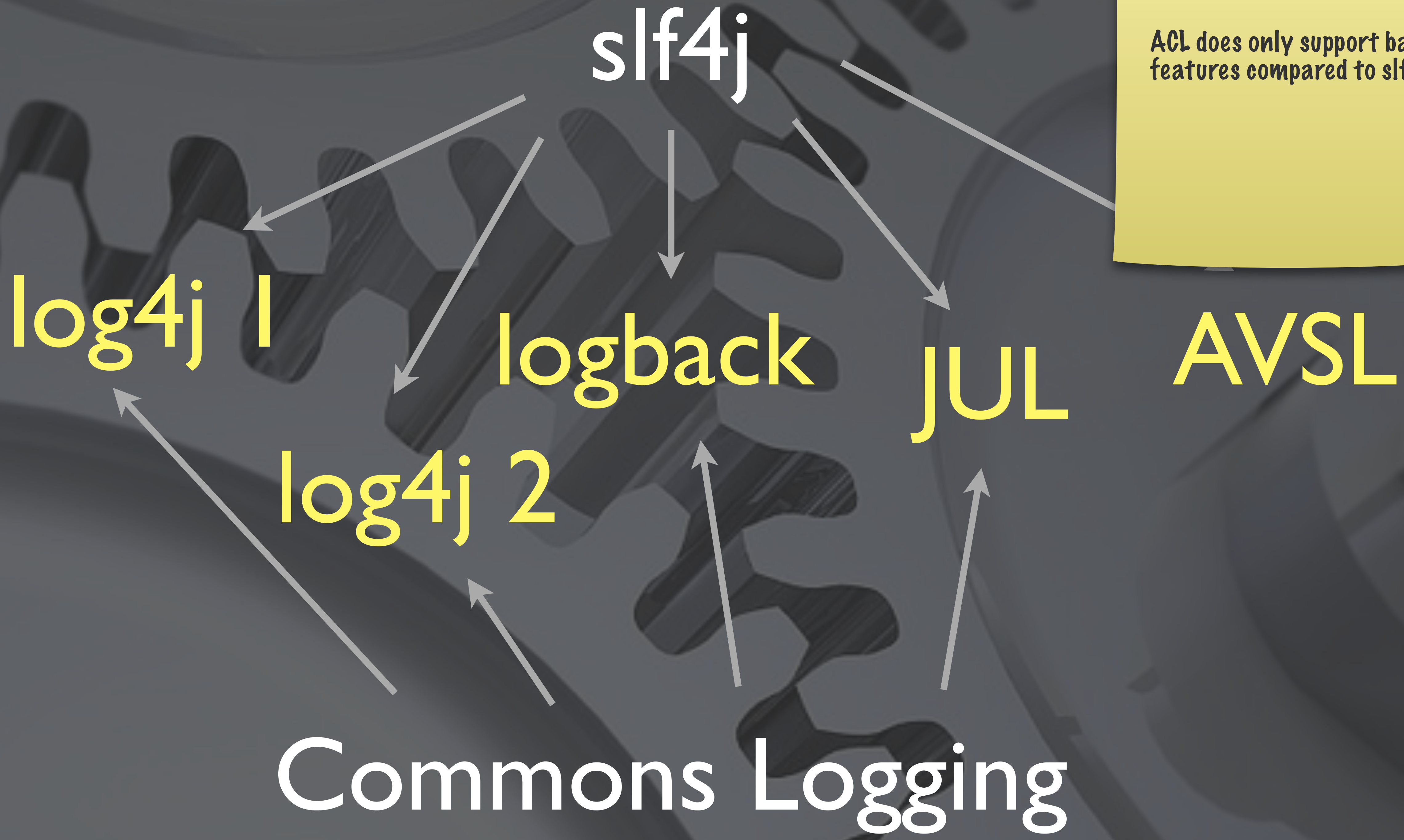
Which pill?

In theory, there are bindings from both pills to most APIs

In practice slf4j is modern and maintained, while ACL is not

De facto standard is slf4j

ACL does only support basic features compared to slf4j



slf4j

log4j 1

logback

JUL

AVSL

log4j 2

Commons Logging

Don‘t forget PAX!

It was long silent on log4j 1 but now there is log4j 2

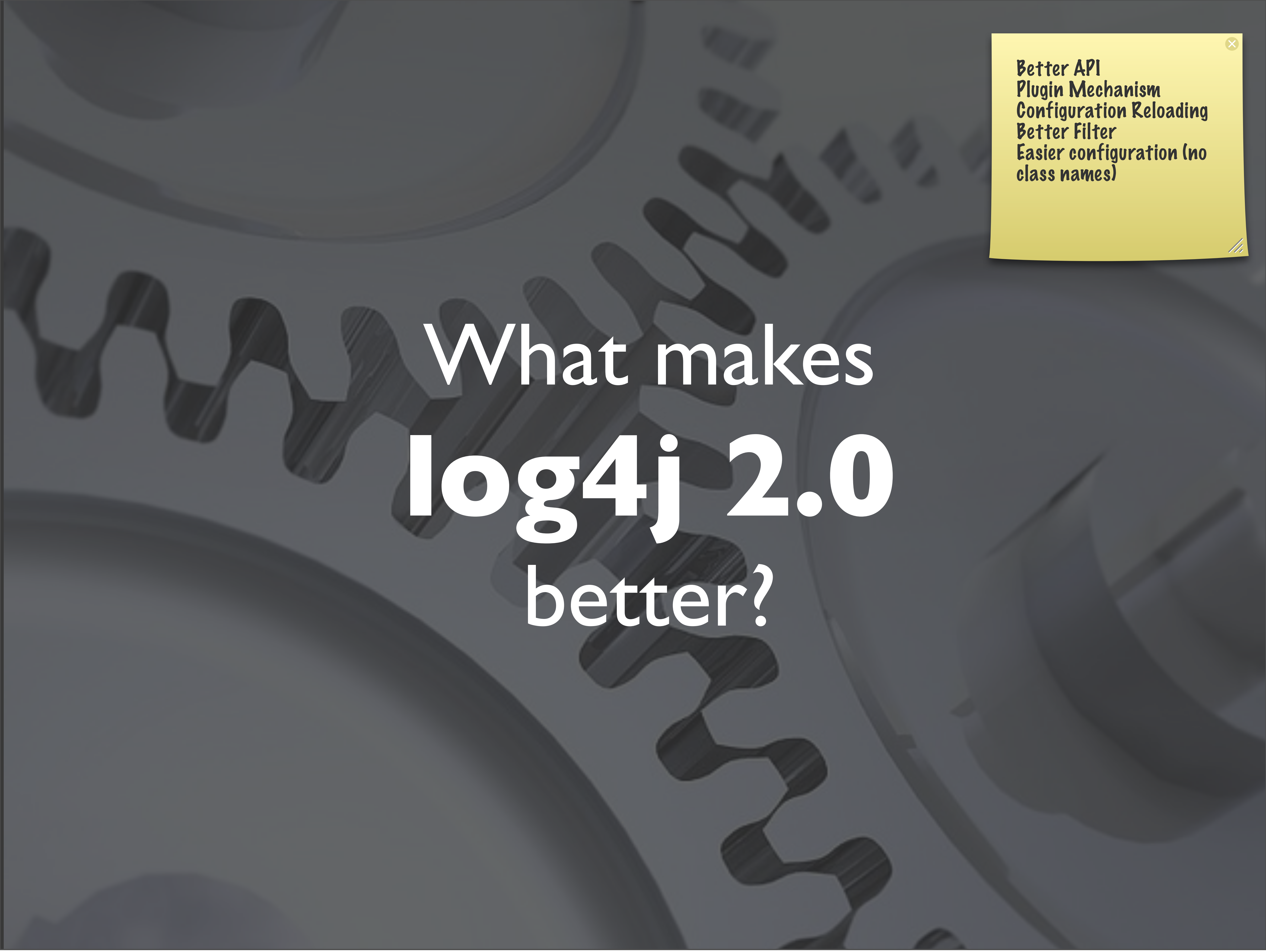
Apache Licensed, Community developed

We will push log4j 2 until it reaches the state to fully replace 1.x series

log4j 2 intergrates well with slf4j, ACL and log4j1

log4j 2.0 is the **future** of **log4j**

Integrates well with:
slf4j
log4j 1.x
Commons Logging



Better API
Plugin Mechanism
Configuration Reloading
Better Filter
Easier configuration (no class names)

What makes **log4j 2.0** better?

Better API
Plugin Mechanism
Configuration Reloading
Better Filter
Easier configuration (no class names)

API++

Instead:

```
if(logger.isDebugEnabled())
   logger.info("Hi, " + u.getA() + “ “ + u.getB());
```

Better API
Plugin Mechanism
Configuration Reloading
Better Filter
Easier configuration (no class names)

API++

Instead:

```
if(logger.isDebugEnabled())
    logger.info("Hi, " + u.getA() + “ “ + u.getB());
```

Write:

```
logger.info("Hi, {} {} ", u.getA(), u.getB());
```

Better Filter

Better API
Plugin Mechanism
Configuration Reloading
Better Filter
Easier configuration (no class names)

Marker:

```
Marker SQL_MARKER =
    MarkerManager.getMarker("SQL");

logger.debug(
        SQL_MARKER,
         "SELECT * FROM {}", table);
```

Plugins

Better API
Plugin Mechanism
Configuration Reloading
Better Filter
Easier configuration (no
class names)

```
@Plugin(name = "Sandbox",
        type = "Core",
        elementType = "appender")
public class Mine extends AppenderBase {

  private Mine(String name, Filter f) {
    super(name, f, null);
  }

  public void append(LogEvent e) { ... }
```

Plugins

Better API
Plugin Mechanism
Configuration Reloading
Better Filter
Easier configuration (no class names)

```
@PluginFactory
public static Mine create(
    @PluginAttr("name") String n,
    @PluginElement("filters") Filter f) {
    return new Mine(n, f);
}
```

Configuration

Better API
Plugin Mechanism
Configuration Reloading
Better Filter
Easier configuration (no class names)

```
<appenders>
  <Console name="Console"
               target="SYSTEM_OUT">
    <PatternLayout
          pattern="%d{HH:mm} %msg%n"/>
  </Console>
</appenders>
```

Configuration

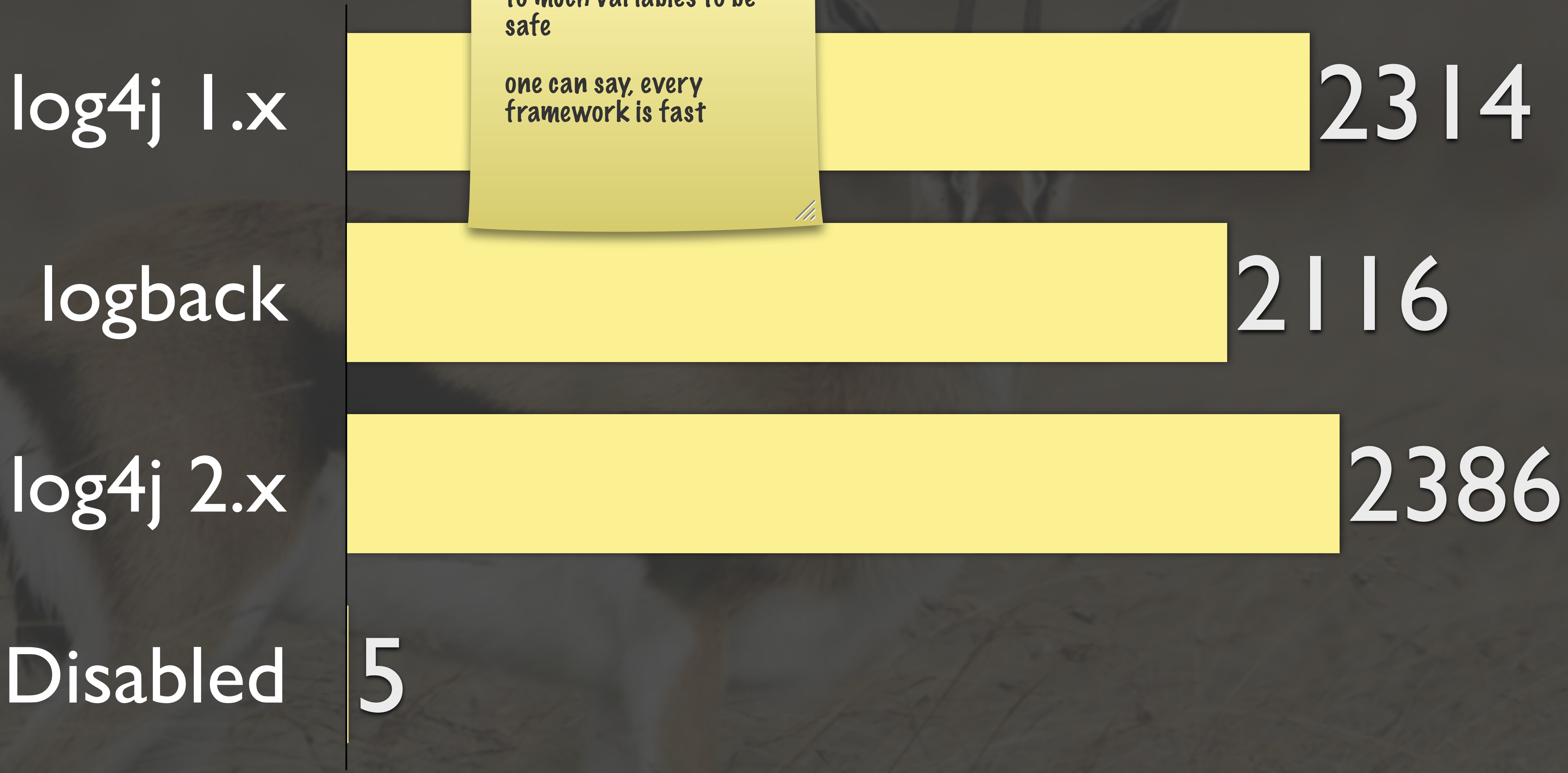
Better API
Plugin Mechanism
Configuration Reloading
Better Filter
Easier configuration (no class names)

```
{ "configuration":
   ...
   "appenders": {
    "Console": {
      "name": "STDOUT",
      "PatternLayout": { "pattern": "%m%n" }
    },
   ...
```

Better API
Plugin Mechanism
Configuration Reloading
Better Filter
Easier configuration (no
class names)

Configuration Reloading

```
<configuration monitorInterval="30">
...
</configuration>
```



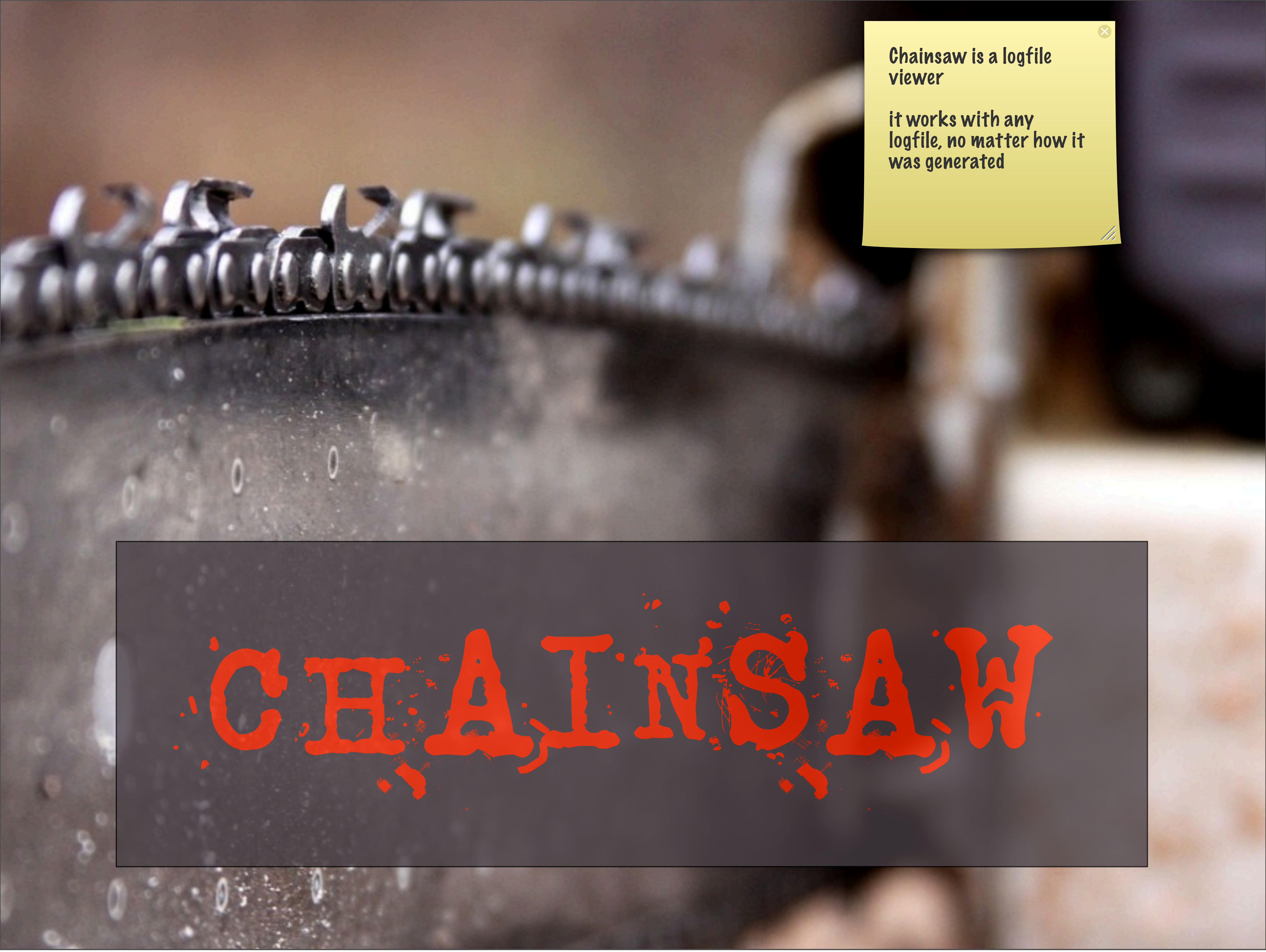
Performance tests have to much variables to be safe

one can say, every framework is fast

SPEED!

Logging Performance in ns

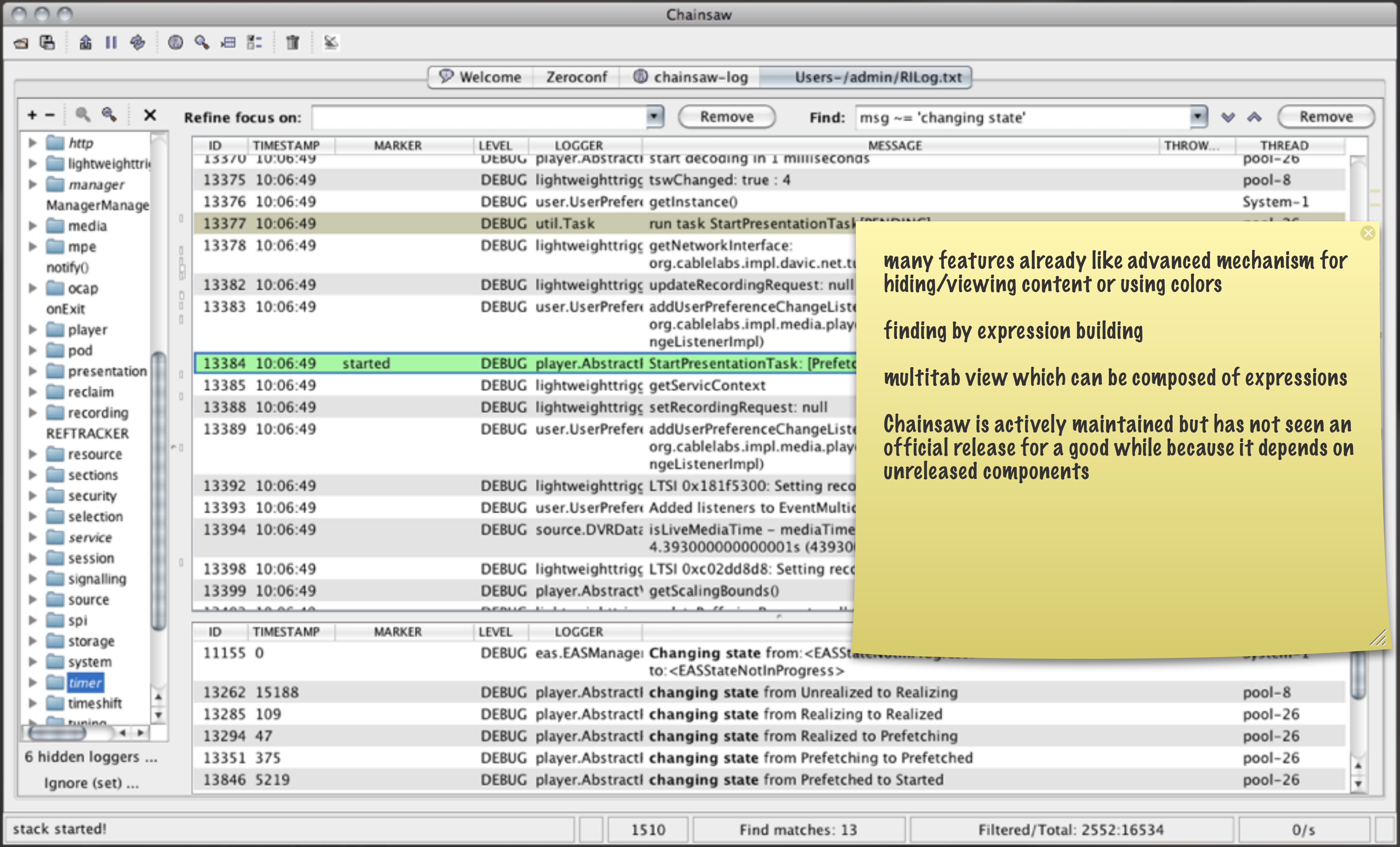
There is more sweetness



CHAINSAW

Chainsaw is a logfile viewer

it works with any logfile, no matter how it was generated



```
ID     TIMESTAMP   MARKER    LEVEL  LOGGER            MESSAGE                                  THROW...  THREAD
13370  10:06:49              DEBUG  player.AbstractI  start decoding in 1 milliseconds                    pool-26
13375  10:06:49              DEBUG  lightweighttrigg  tswChanged: true : 4                                 pool-8
13376  10:06:49              DEBUG  user.UserPrefere  getInstance()                                        System-1
13377  10:06:49              DEBUG  util.Task         run task StartPresentationTask [PENDING]
13378  10:06:49              DEBUG  lightweighttrigg  getNetworkInterface:
                                                      org.cablelabs.impl.davic.net.tu
13382  10:06:49              DEBUG  lightweighttrigg  updateRecordingRequest: null
13383  10:06:49              DEBUG  user.UserPrefere  addUserPreferenceChangeListe
                                                      org.cablelabs.impl.media.play
                                                      ngeListenerImpl)
13384  10:06:49   started    DEBUG  player.AbstractI  StartPresentationTask: [Prefetc
13385  10:06:49              DEBUG  lightweighttrigg  getServiceContext
13388  10:06:49              DEBUG  lightweighttrigg  setRecordingRequest: null
13389  10:06:49              DEBUG  user.UserPrefere  addUserPreferenceChangeListe
                                                      org.cablelabs.impl.media.play
                                                      ngeListenerImpl)
13392  10:06:49              DEBUG  lightweighttrigg  LTSI 0x181f5300: Setting reco
13393  10:06:49              DEBUG  user.UserPrefere  Added listeners to EventMultic
13394  10:06:49              DEBUG  source.DVRData    isLiveMediaTime - mediaTime
                                                      4.393000000000001s (43930
13398  10:06:49              DEBUG  lightweighttrigg  LTSI 0xc02dd8d8: Setting reco
13399  10:06:49              DEBUG  player.AbstractV  getScalingBounds()

ID     TIMESTAMP   MARKER    LEVEL  LOGGER            
11155  0                     DEBUG  eas.EASManager    Changing state from:<EASSta...
                                                      to:<EASStateNotInProgress>             System-1
13262  15188                 DEBUG  player.AbstractI  changing state from Unrealized to Realizing        pool-8
13285  109                   DEBUG  player.AbstractI  changing state from Realizing to Realized          pool-26
13294  47                    DEBUG  player.AbstractI  changing state from Realized to Prefetching        pool-26
13351  375                   DEBUG  player.AbstractI  changing state from Prefetching to Prefetched      pool-26
13846  5219                  DEBUG  player.AbstractI  changing state from Prefetched to Started          pool-26
```

stack started! | 1510 | Find matches: 13 | Filtered/Total: 2552:16534 | 0/s

many features already like advanced mechanism for hiding/viewing content or using colors

finding by expression building

multitab view which can be composed of expressions

Chainsaw is actively maintained but has not seen an official release for a good while because it depends on unreleased components

The story of **log4php**

Entered Incubation 2004
Retired and reenabled in 2007
While retirement discussion in 2009, Christian stepped up
2010: log4php graduated
2010: Ivan joined

Multiple releases
Aiming for a next big release
First PHP project on ASF

2004 2007 2009 2010

What‘s with...

log4net

log4cxx

log4net has little activity

log4cxx has no activity

Community

(Finland)
Ísland
(Iceland)
(Ukraine)
Kazak
(Afghanistan)
(Pakistan)
Venezuela
Colombia
Perú
(Peru)
Bolivia
Chile
Nigeria
Ethiopia
Kenya
DR Congo
Tanzania
Angola
Namibia
Botswana
Madagasikara
(Madagascar)
South Africa
South Atlantic Ocean

Good times, **bad times**

2000 Founded
2005 Ceki left
2009 Near death
2010 log4php revived
2011 log4j 2.0 arrived

2000 2009 2010 2011

We need your HELP

Currently only 4 regular code contributors.

We still have a problem:
more people = more fun =
more code

But we **FIGHT!**

6 Releases
~750 Commits
~600 dev Mails

in 2012 only!

Seriously.
Isn‘t logging just
BORING?

Well... how do you log on your mobile/ cloud apps?

With new innovatio
new logging challenge
come up.

Logging needs innovation too.

Look at Apache Flume.

Flume is an advanced framework to collect log data

log4j2 has an Flume appender

logphoria
loggly.com
do commercial products

Apache MAYHEM™

Upcoming at the Incubator
Guice Components

Apache MAYHEM™

Instead:

```
Logger logger = Logger.getLogger( „test“);
```

Apache MAYHEM™

Instead:

```
Logger logger = Logger.getLogger( „test“);
```

Write:

```
@InjectLogger
Logger logger;
```

Remember: Logging is mission critical.

Don‘t go without.

We are hiring!

use it!
ask questions!
spread the love!
send patches!

general@logging.apache.org

Thank you!

Christian Grobmeier

@grobmeier
http://www.grobmeier.de

Follow me, if you dare!

Image Credits

Earth (Nasa)
Vulcano (Flickr: Martin Barland)
Tiger (Flickr: TeryKats)
Gazelle (Flickr: fwooper)
Golden Gate Bridge (Flickr: TimeLapseBlog.com)
Pens (Flickr: JD | Photography)
Wineglass (Flickr: Willia4)
Headache (Flickr: Threephin)
Pills (Dr. Michael Günther)
Wheelgear (Flickr: ralphbijker)
Grapes (Flickr: RVWithTito)
Chainsaw (Flickr: Dave Hosford)
Elephant (Flickr: Werner Vermaak)
Haunted House (Flickr: barb_ar)
SOS (Flickr: Daquella Manera)
Bathtime muscles (Flickr: mollypop)
Yawning Animal (Flickr: robef)
Mobile (Flickr: twicepix)
Clouds (Flickr: karindalziel)
Fight (Flickr: KellBailey)