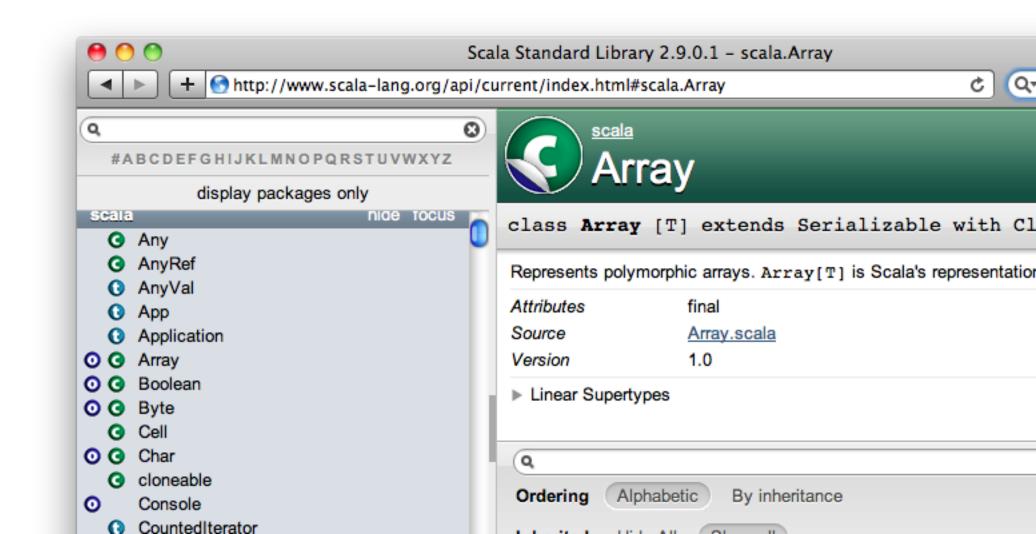
I think code is the answer to a problem. If you don't have the spec or you don't have any documentation, you have to guess what the problem from the answer.

Joe Armstrong

Scaladoc





- I'm a web developer.
- By night, I'm a committer of Scaladoc.
- I prefer Play rather than Lift:)

SLOW JAVASCRIPT

Gilles Dubochet



```
property("parse") = parse(
    "/** One two three */",
    Text("One two three")
)
```

Improve JavaScript performance of Scaladoc

Very nice! I vote to include it.

Aha, much better!

I especially like the alphabetized index of method/val names. Much, much faster, too!

Scaladoc: Incremental search for methods

Just tried it out and my only comment is: Awesome!

That looks really awesome;

indeed: cool!

You're my new hero.

I'm using Chrome and the speed is impressive.

+ 1

December 2010 - January 2011

I got a commit bit on trunk/src/compiler/scala/tools/nsc/doc!

March 2011

[scala-internals]
Upcoming 2.9.0.RC1 (take 2)

Bill Venners, creator of ScalaTest



March 2011

I got a commit bit on trunk!

Donna Malayeri



```
property("Trac #484 - refinements and existentials") = {
 val files = createTemplates("Trac484.scala")
 val lines =
      Itype Bar = AnyRef { type Dingus <: T forSome { type T <: String } }</pre>
      ltype Foo = AnyRef { ... /* 3 definitions in type refinement */ }
      Idef q (x: T forSome { type T <: String }): String</pre>
      ldef h (x: Float): AnyRef { def quux(x: Int,y: Int): Int }
      | Idef hh (x: Float): AnyRef { def quux(x: Int,y: Int): Int }
      ldef j (x: Int): Bar
      ldef k (): AnyRef { type Dingus <: T forSome { type T <: String } }</pre>
    """.stripMargin.trim.lines map (_.trim)
 files("RefinementAndExistentials.html") match {
    case node: scala.xml.Node => {
      val s = node.text.replaceAll("\\s+", " ")
      lines forall (s contains _)
    case _ => false
```

Scala 2.9 is out, sweet. Docs don't make headlines, but my favourite feature is the new Scaladoc.

Lachlan O'Dea