

# Perl 6 in 2011

salut o/

masak

Perl 6 historian

# Perl 6 in 2011

and why

you should

be

excited

about it

# Perl 6 in 2011

# Perl 6 in 2011

...for people who are interested in the project  
but don't have the time to get involved  
(have to put food on the family, etc),  
but who don't mind going to a talk  
about how things are going...

# Perl 6 in 2011

...for people who are interested in the project  
but don't have the time to get involved  
(have to put food on the family, etc),  
but who don't mind going to a talk  
about how things are going...

...

# Perl 6 in 2011

...for people who are interested in the project  
but don't have the time to get involved  
(have to put food on the family, etc),  
but who don't mind going to a talk  
about how things are going...

...

...and why they should be excited about it

In development for a long time

First fear:

"Will Perl 6 ever be released?"

now

what I did not know

In 2003

There was a Perl 6 implementation

Quite a bit of features

if unless while until  
for foreach return quotes  
try given/when  
next substr  
last length  
break index  
redo binary ops join  
skip reverse  
continue hyperops print  
warn  
die  
much of regexes exit  
sleep  
time

they were hacking along

but, alas

they ran into a problem

complexity

"The limits of my language mean the limits of my world."

Ludwig Wittgenstein

\o/ enter pmichaud! \o/

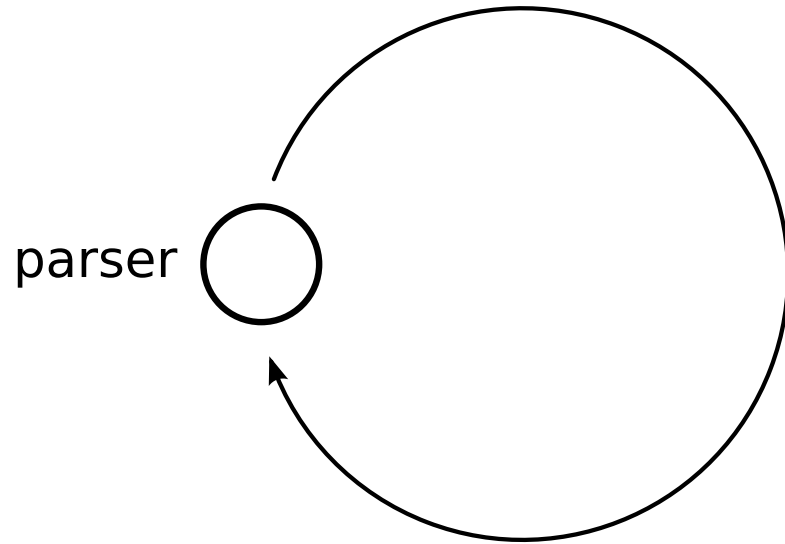
<pmichaud> masak's talks often contain a few  
nuggets of potential embarrassment  
for me :)

<masak> pmichaud: oh!

<pmichaud> "Rakudo: a long List of fail()s" :-P

Perl 6 manager

with a vision



time is cyclic

start over

waterfall

analysis ⇒ design ⇒ implementation ⇒ testing

whirlpool

testdesignplementanalysignplementest...

more specifically

we learn by implementing

we learn by coding

example: cat

Second fear:

"Will they ever stop changing the spec?"

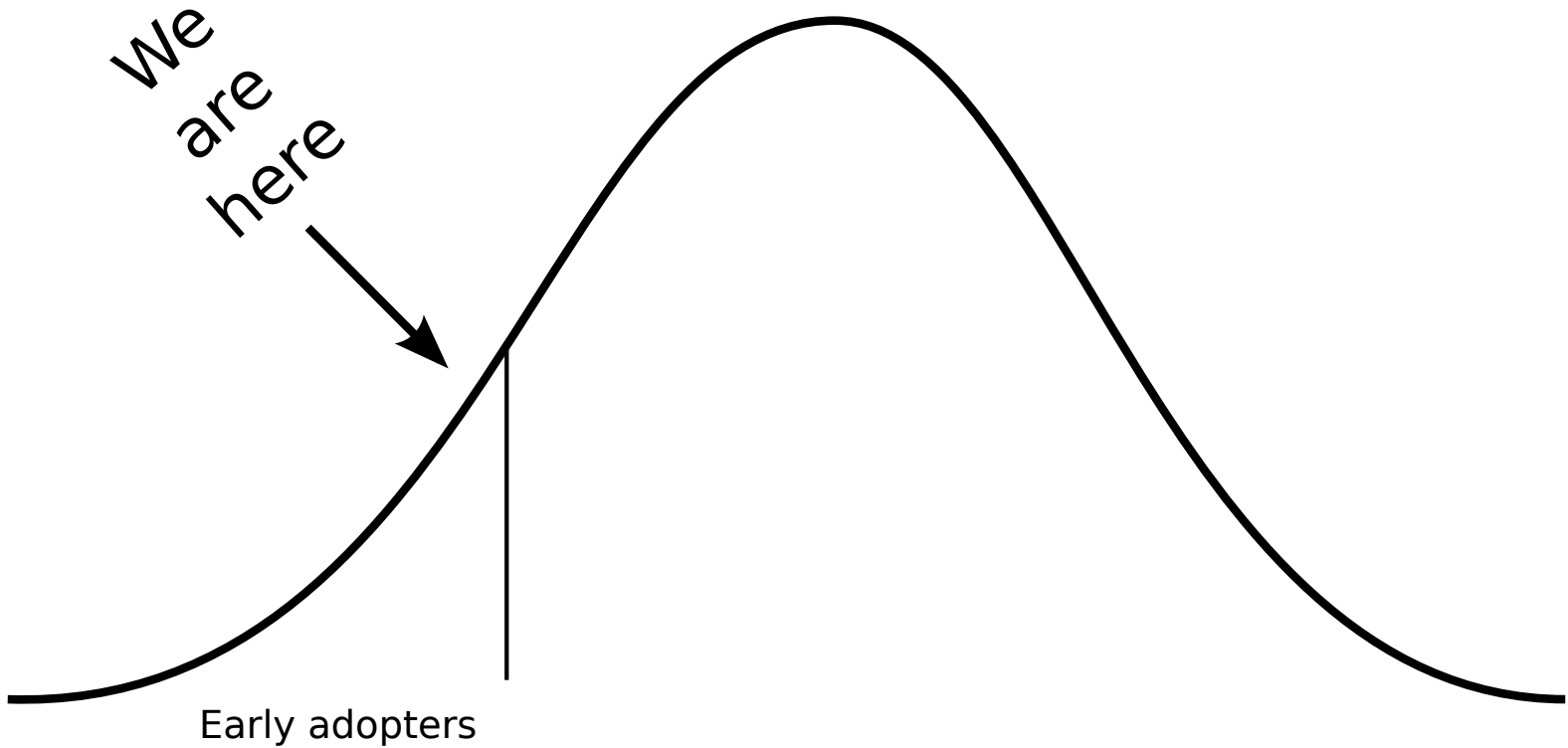
How stable is Perl 6?

<TimToady> it's stable enough to have almost **300 working examples on rosettacode.org** that are expected to continue working for the most part

also

modules

applications



We  
are  
here

Early adopters

The future is here...

...it's just not very evenly distributed.

William Gibson

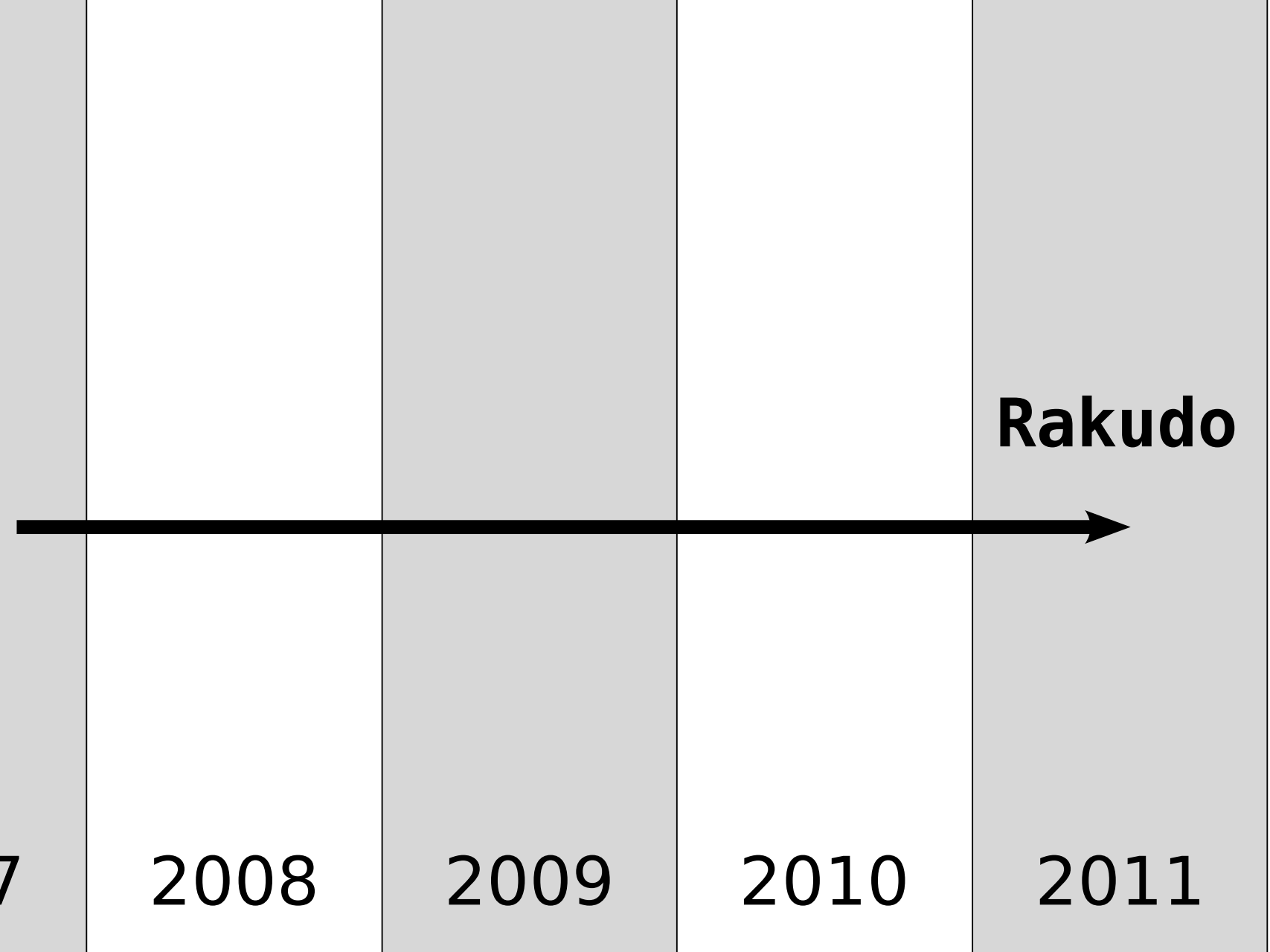
Rakudo

Scalar Array Hash Variables my and our  
sub multi sub signatures with types and  
defaults classes methods submethods  
roles has self is of does handles for loop  
repeat while if elsif else unless given  
when junctions reduction operators  
hyperoperators cross and zip operators  
slicing syntax chained operators whatever  
star in slices closures currying .arity next  
last redo gather/take pointy blocks open  
slurpy files try/CATCH ranges min max  
minmax m// .trans .subst s/// grammars  
.parse pick roll rand list item hash eval  
placeholder parameters short-circuiting

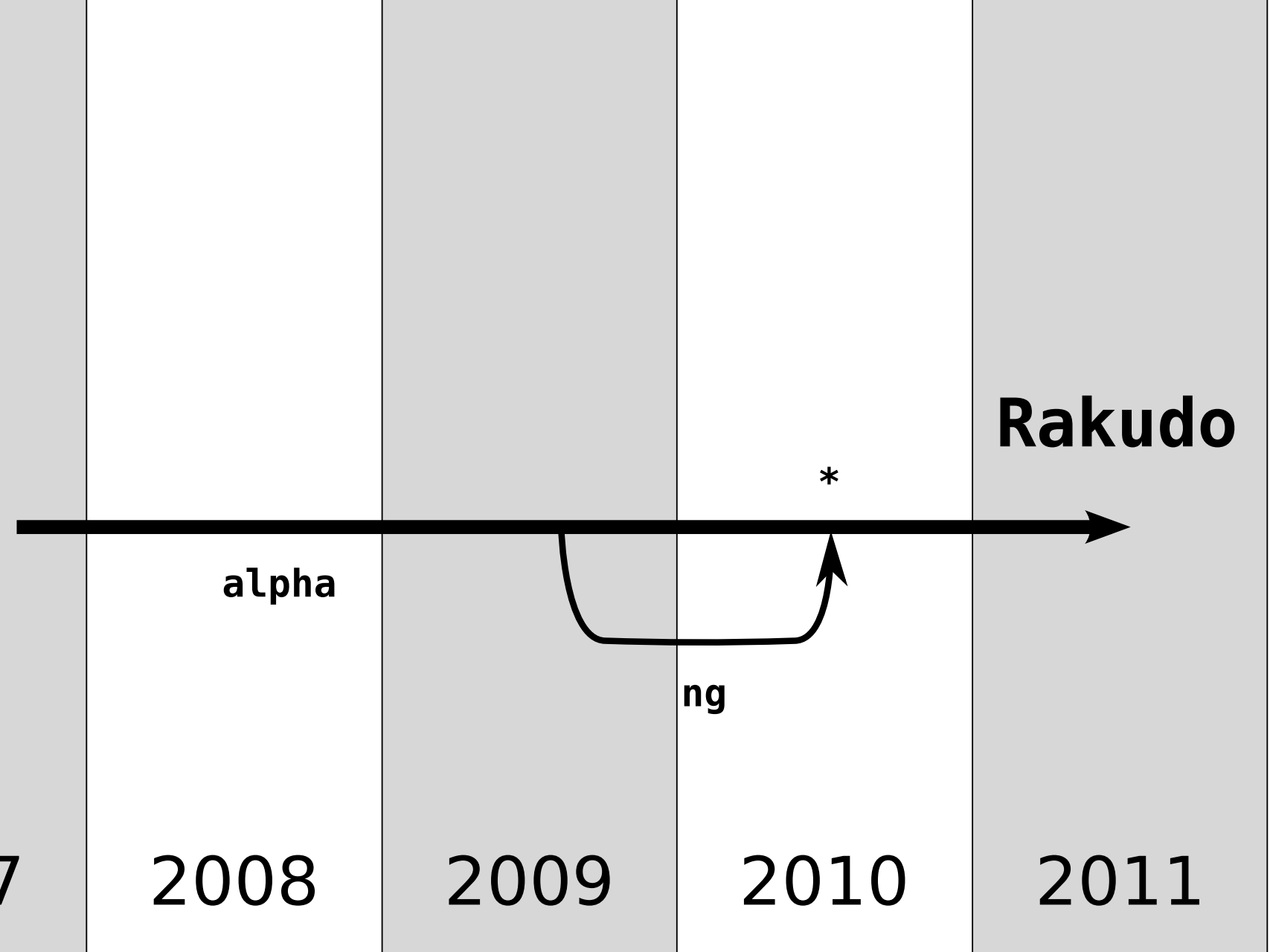
by all accounts

Perl 6 is "here"

it's just... une tortue



Rakudo \*



SO

unlike in 2003

we can build features

but what about speed?

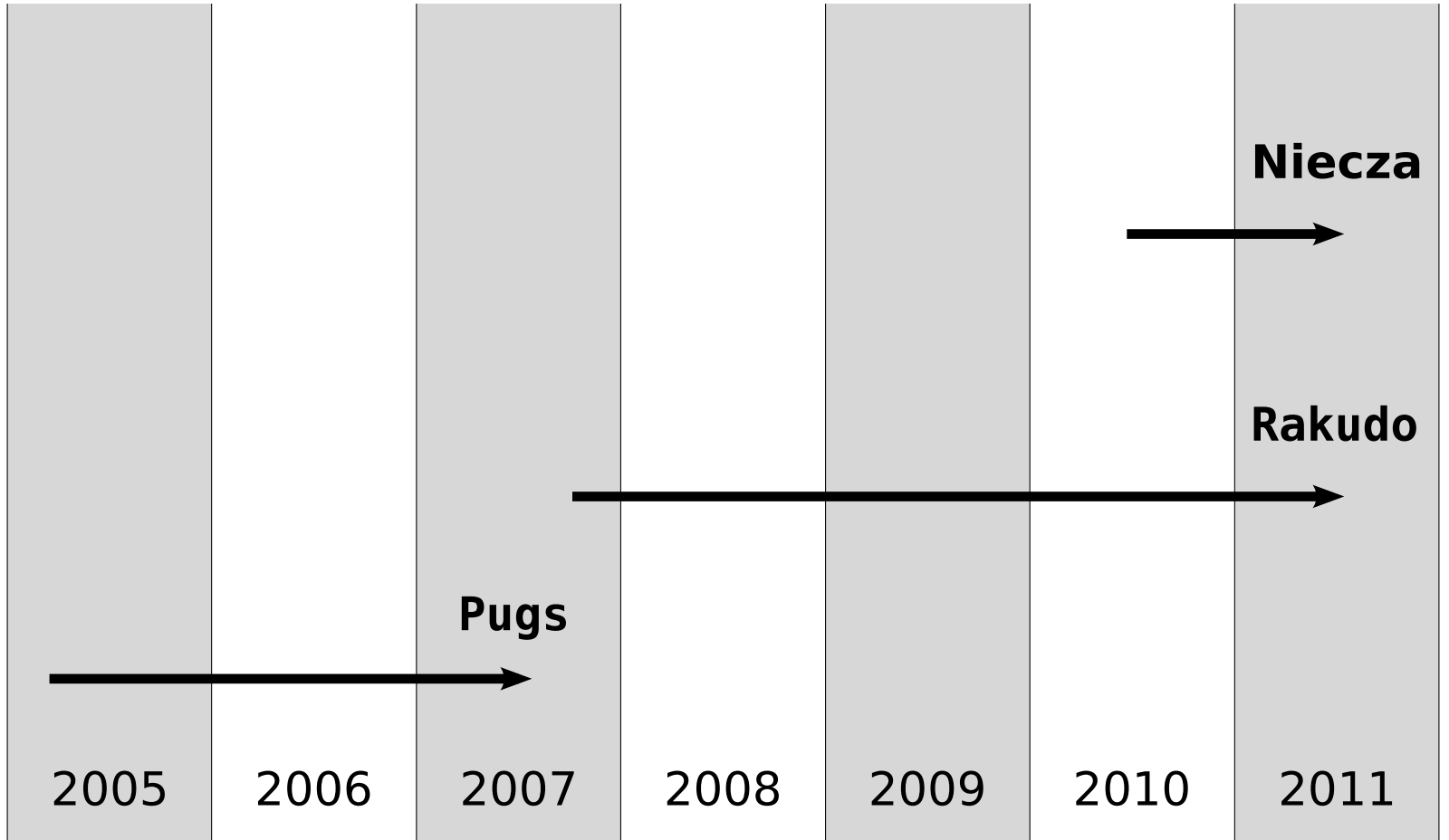
Third fear:

Will Perl 6 ever be performant?

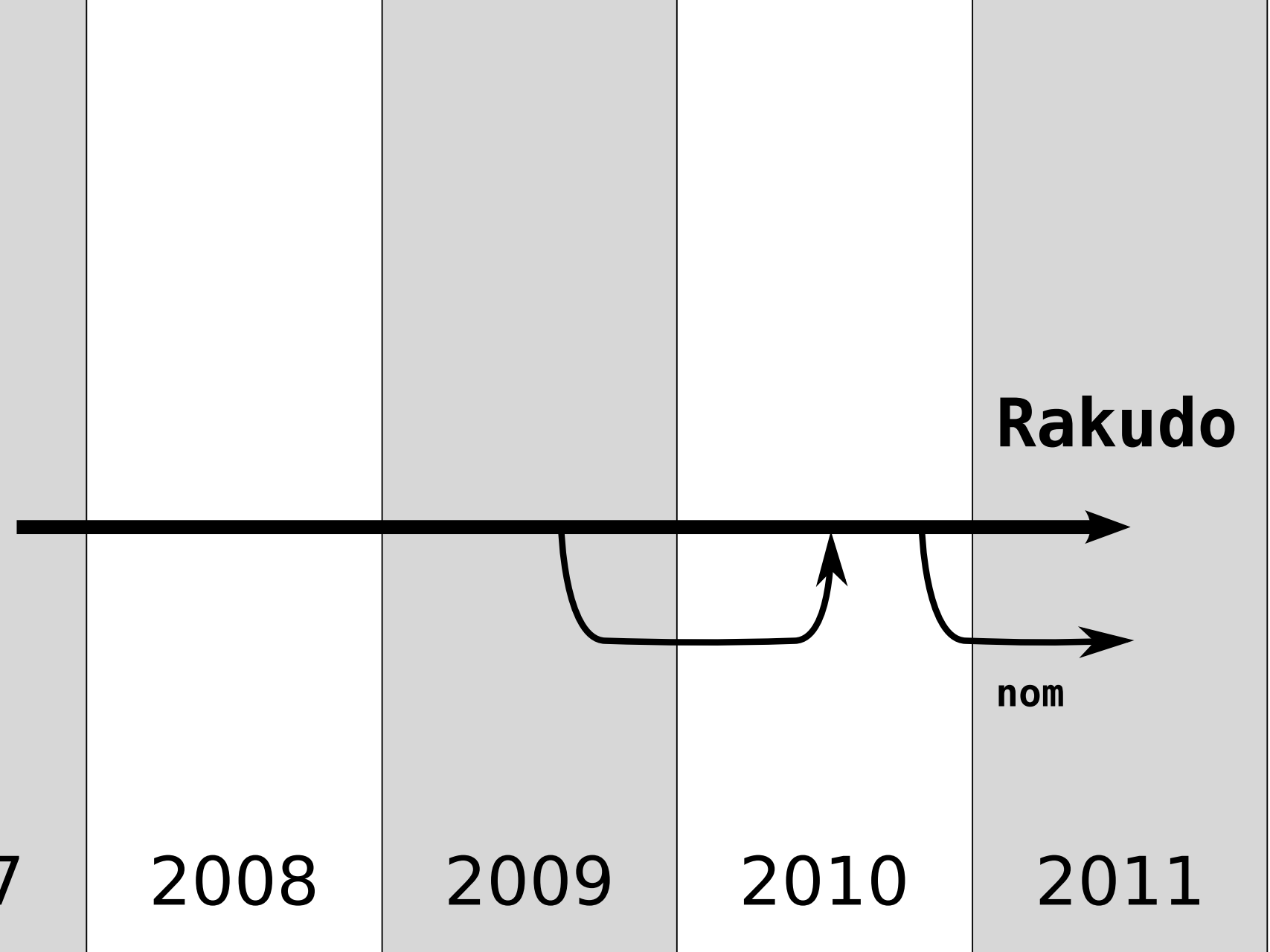
Nieczna

one year old

sorear++



nom



**Rakudo**

nom

2008

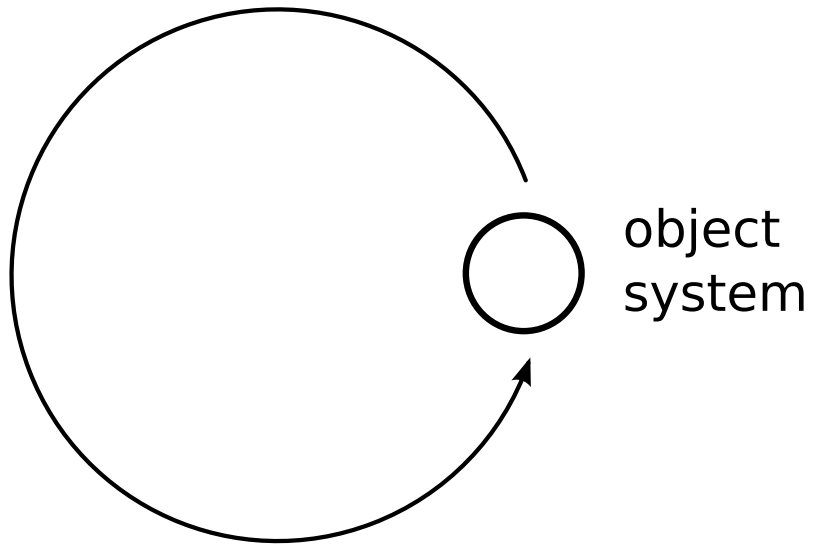
2009

2010

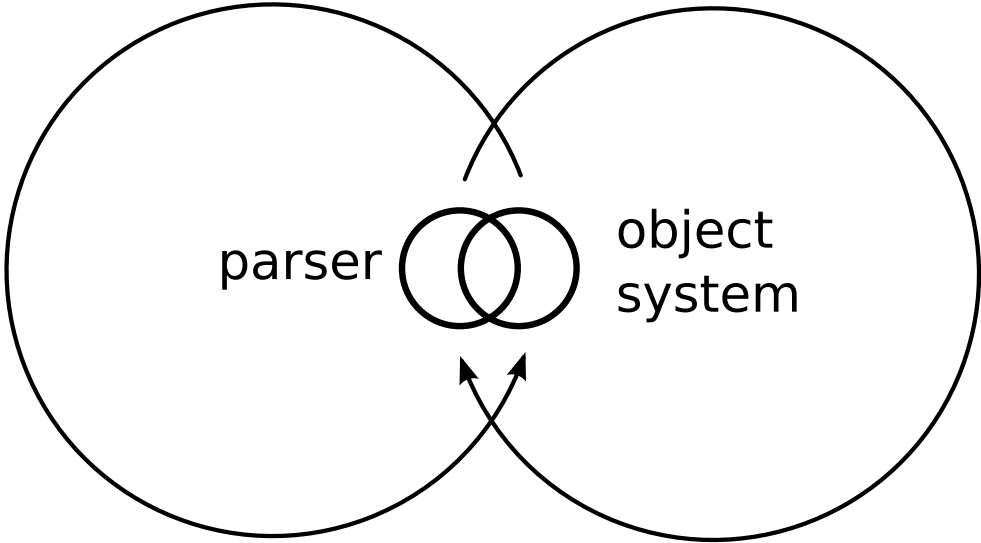
2011

7

MOP



6model

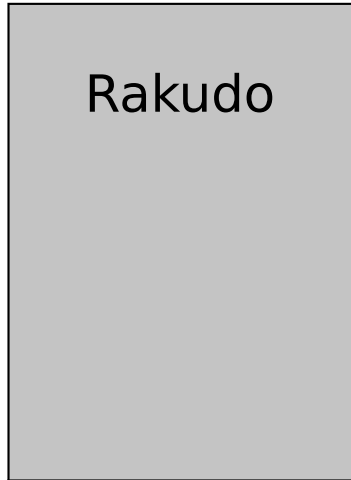


summary

rakudo is shaping up

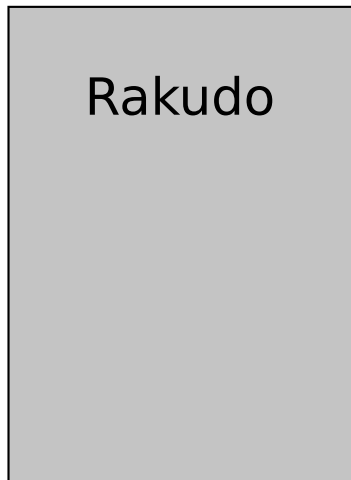
niecza is shaping up

**features**



**speed**

**features**



**speed**

if you haven't tried Perl 6

this is a good time

for you to

join in

download rakudo or niecza

or both

try it out

and tell us what you think

because

things are about to

get very interesting

merci bcp