Domain modeling & the real world

Carl Mäsak
GPW 2016-03-09

https://www.flickr.com/photos/tir_na_nog/534634
so, I'm a father
baby changing zone
baby food production pipeline
baby nourishment table zone
baby's office
baby amusement area
living room
couch
secondary baby rest area
baby rest area
this makes one think
“what is software development?”
two views
completely at odds with each other
ok, that could make a decent talk
hi, I'm masak
I write Perl 6
professionally
(and on my spare time)
so, two views of software dev
first view: complete clarity
your core domain has no dependencies
example: domain events
example: Git
example: Unix pipes
second view: complete chaos
code as organism
code as organism
technical debt
we don't plan, we measure
less physics
more biology
resources
example: FPGA + evolutionary programming

Fig. 18. The pruned circuit diagram: cells and wires are only drawn if there is a connected path through which they could possibly affect the output.

Fig. 19. The functional part of the circuit. Cells not drawn here can be clamped to constant values without affecting the circuit's behaviour.
example: exceptions in Argentina
example: injection in SOAP::Lite

```perl
# if nothing there, then get it from envelope itself
$request->match((ref $request)->method);
($method_uri, $method_name) = ($request->namespaceuriof || '', $request->dataof unless $method_name;

$self->on_action->(my $action = $self->action, $method_uri, $method_name);

# check to avoid security vulnerability: Protected->Unprotected::method(@param
# see for more details: http://www.phrack.org/phrack/58/p58-0x09
die "Denied access to method ($method_name)\n" unless $method_name =~ /^\w+$/;
```
unintended consequences
alright, so who wins?
my take: we can push niceness far...
...but not all the way
Design Beyond Human Abilities

Richard P. Gabriel
we need to be both physicists and biologists
to handle both the beauty and the chaos
thank you