



What has Ruby made Possible?

Heroku
Ruby Association

Yukihiro "Matz" Matsumoto
Ю К И Х И Р О М А Ц У М О Т О
@yukihiro_matz



Karoshi

過勞死

Death by working too hard



Why did they work too hard?



Hard to understand



We work to live, but some may
live to work



We often work too hard



Psychological tendency



Alpha syndrome



Computers are great!



Never be tired



Never be bored



They are fast



We can be more productive
when we work with computers



But



Beware of ALPHA syndrome



ALPHA is a leader of a group



Order or hierarchy



Imagine your pet dog





You pet it



You work for it



You do everything you can for it



It assumes it is ALPHA



It starts to behave bad

ALPHA syndrome



Computers



They are efficient



They are NOT perfect



They are far from perfect



You have to work for them



You start to pay effort for them



You start to think computers are
ALPHA



You gradually become their slave



reverse ALPHA syndrome



Remember you are the master



Don't work too hard



Be lazy



Let computers work for you



The principles behind Ruby



- Productivity
- Development efficiency
- Programmer happiness
- Focusing on humans



Ruby is an Object-oriented
language



Dynamic typing



Duck typing



Duck typing



Duck typing



If it walks like a duck



Quack like a duck

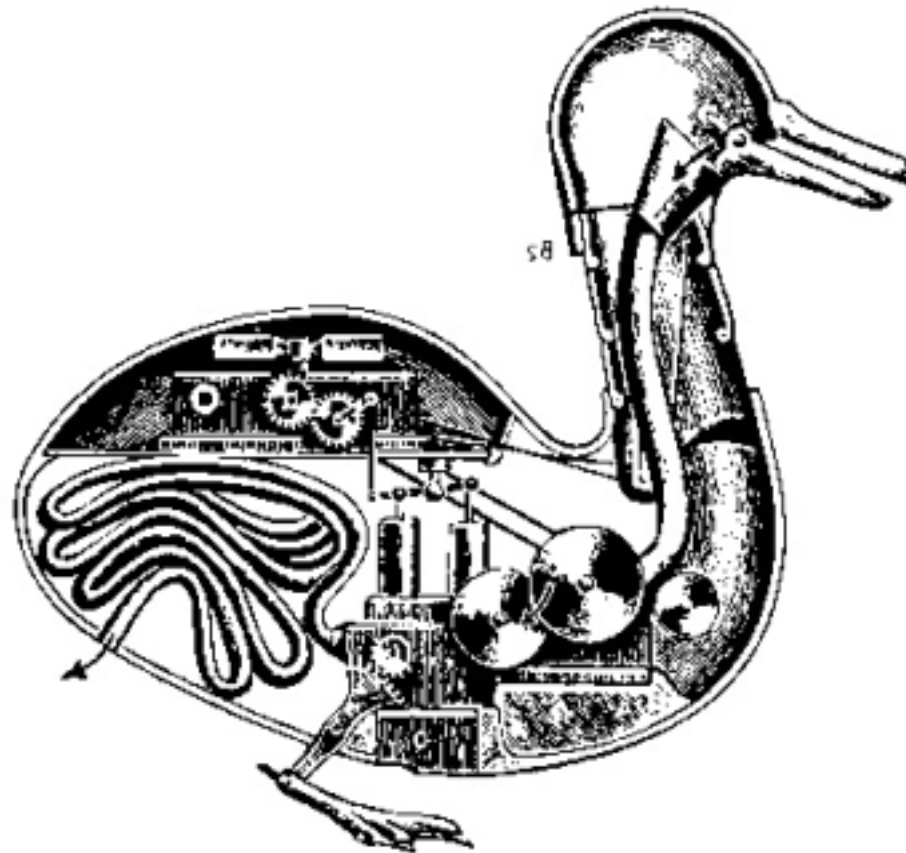


It is a duck



We assume it is a duck





Duck typing



Essence of OOP



Do not classify



Animal · Mammal · Human



Do not check type



Just ask



We don't care about inheritance



We don't care about structure



We just care how it behave



So we can ignore inside detail



So we ask computers to dispatch



So we don't have to work



Let computers work for you



Don't work too hard



Be lazy



DRY



Don't Repeat Yourself



Avoid duplicates



Avoid redundancy



Avoid copy-and-paste



Because it's BAD



You spread bugs all over



Besides that



We are lazy



We are too lazy to maintain
duplicates



In summary,

- 
- Duck typing
 - DRY



Spirit of laziness



3 virtues of a GREAT programmer



- Laziness
- Impatience
- Hubris



Laziness



The quality that makes you go to great effort to reduce overall energy expenditure. It makes you write labor-saving programs that other people will find useful and document what you wrote so you don't have to answer so many questions about it.



Impatience



The anger you feel when the computer is being lazy. This makes you write programs that don't just react to your needs, but actually anticipate them. Or at least pretend to.



Hubris



The quality that makes you write (and maintain) programs that other people won't want to say bad things about.



"Laziness is the most important
virtue"



We should not work too hard



We have to be lazy



We have to enjoy what we do



Enjoy programming



Remember Ruby principles



- Productivity
- Development efficiency
- Programmer happiness
- Focusing on humans



We need more power



To make us happy programmers



To avoid hard working



To avoid ALPHA syndrome



Ruby/Rails give power



That's our ultimate goal



Empowering programmers



I trust you



Great power comes with great
responsibility



I trust you will do great works if
you have power



More power to programmers



Power from computers



What has Ruby made possible?



Laziness



Productivity



Development happiness



Internatinalization

2001



First RubyConf



38 attendees

2006



First RubyKaigi



RubyKaigi



Internatinalization



Ruby leads internationalization of Japanese engineers



I'm proud



Happy Hacking!