ThoughtWorks®

v3.0

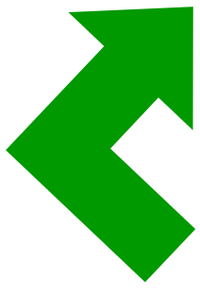
Design Principles Tensions & Synergies

Ilias Bartolini

HOW IT WORKS

- *Divide into pairs*
- *Start setup development environment*
- *Explain the tensions & synergies*
- *Choose a design principle or rule*
- *Game of life*
- *Implement a new user story (45min)*

BREAK



- *Showcase changes*
- *How did it affect other principles? Post-it!!*
- *Repeat*
- *Conclusions*
- *Feedback*

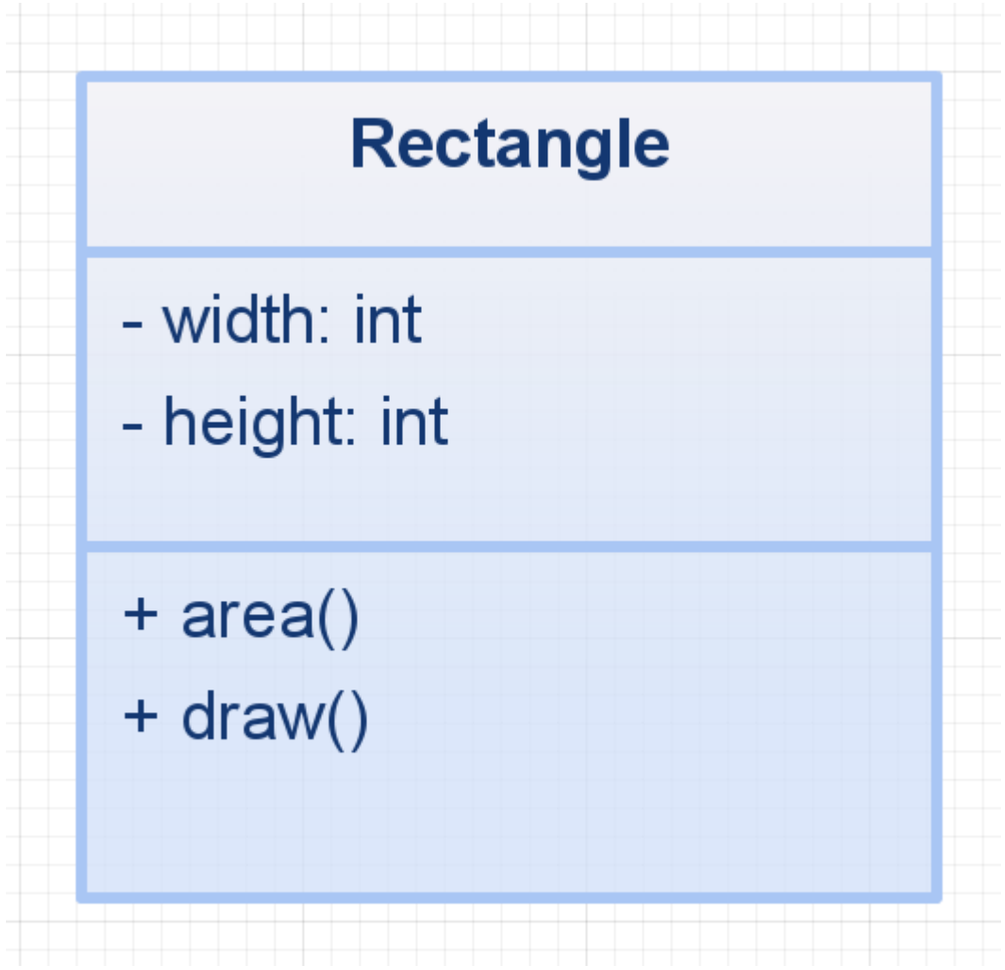
Introductions

http://bit.ly/oo_dojo

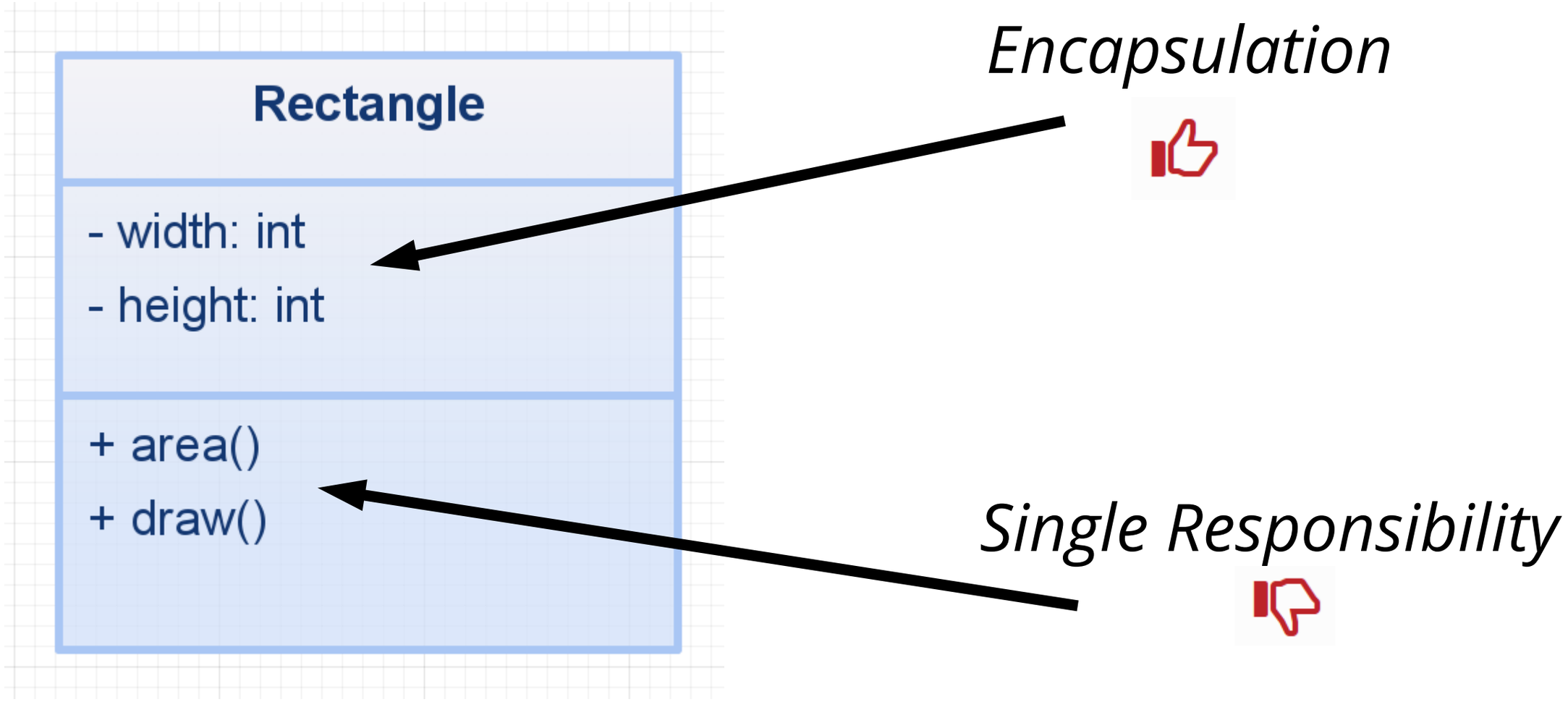
Fork me!

Follow README instructions to build & test

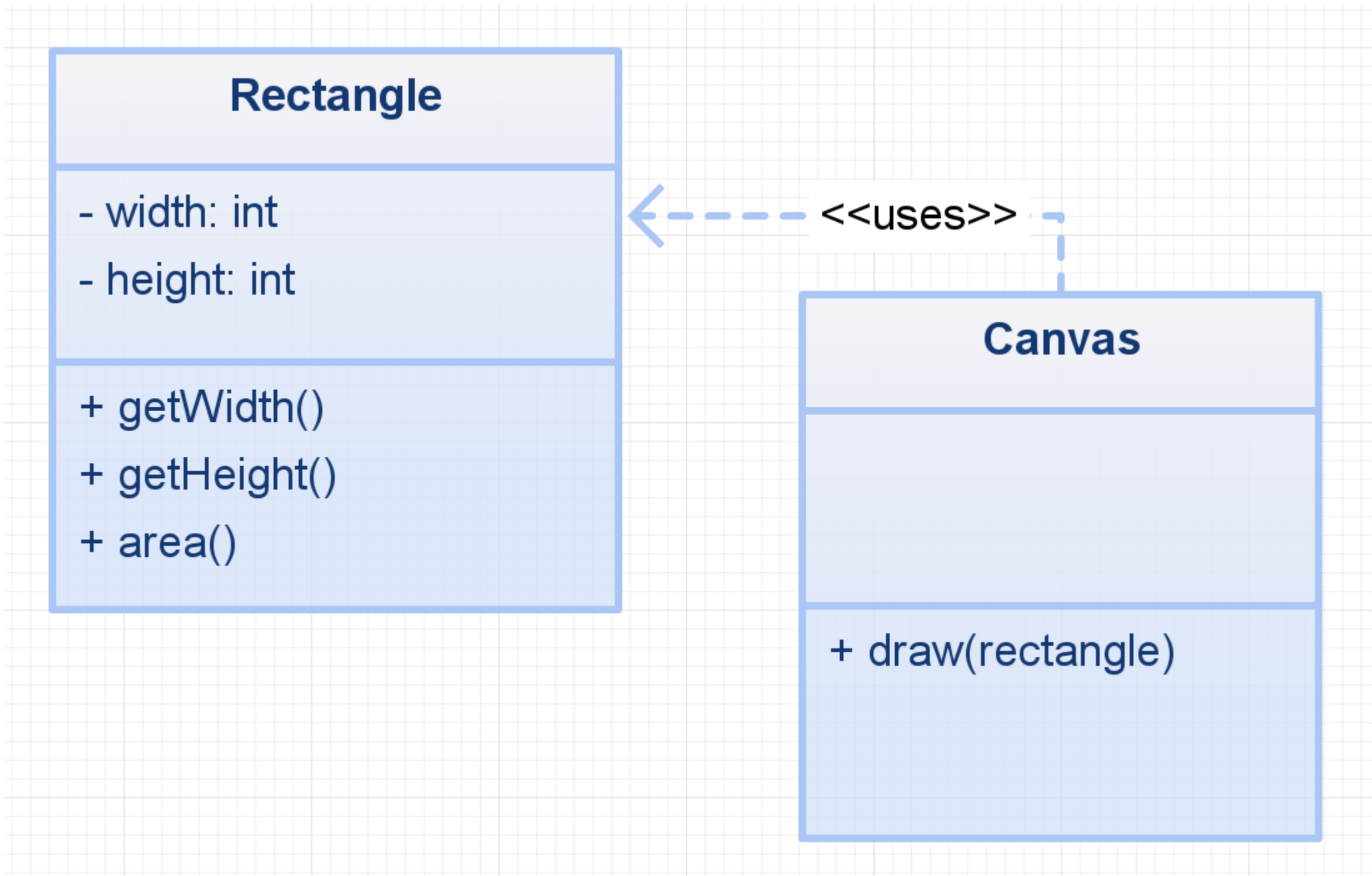
A SIMPLE EXAMPLE



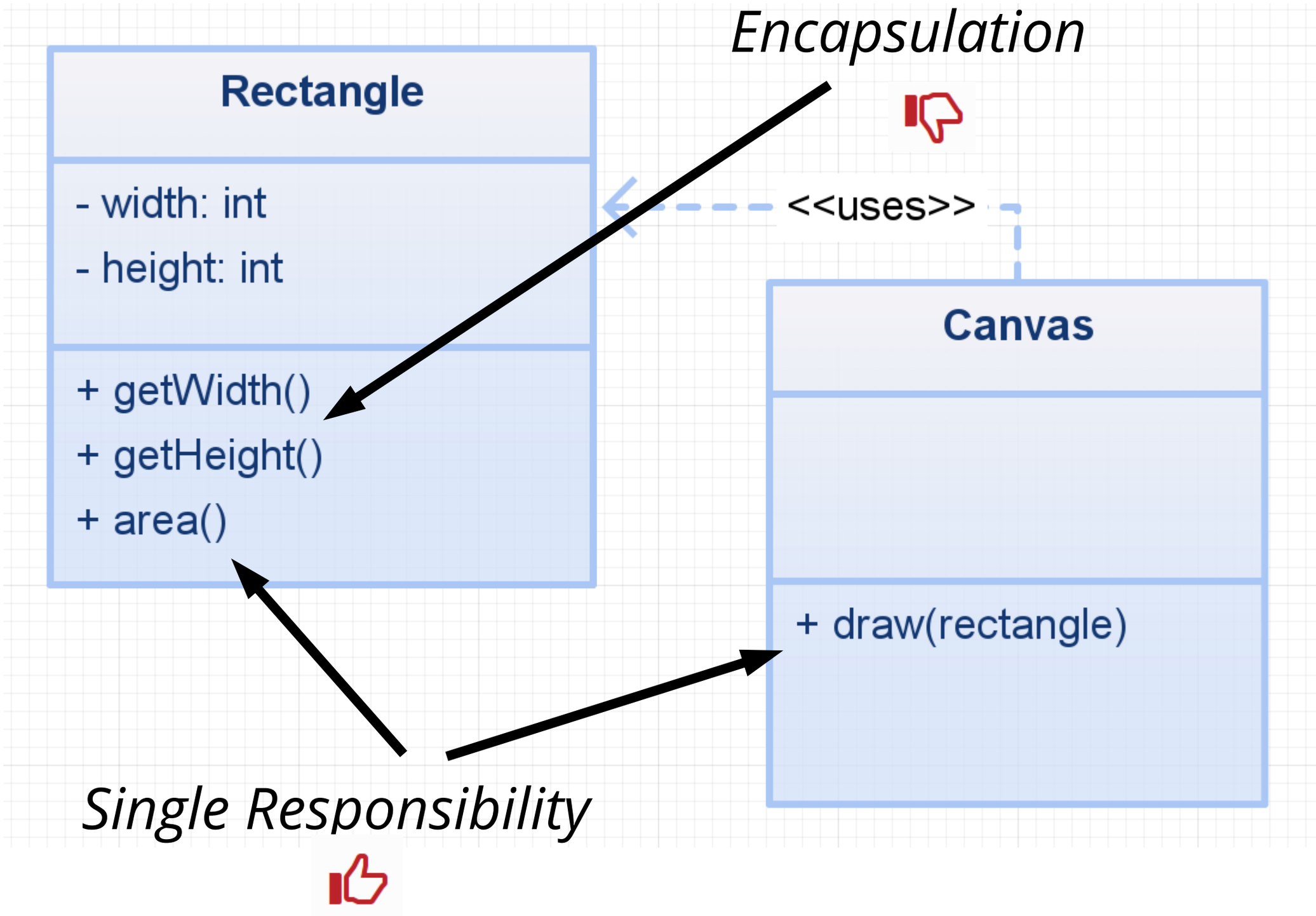
A SIMPLE EXAMPLE



A SIMPLE EXAMPLE



A SIMPLE EXAMPLE



Single Responsibility vs Encapsulation

SOME PRINCIPLES AND RULES...

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	**List of Principles & Rules**		S	O	L	I	D	Enc	Low Cou	High Coh	C over I	DRY	Tell, no ask		LoD
2	S: Single Responsability														
3	O: Open-Close														
4	L: Liskov substitution principle														
5	I: Interface segregation														
6	D: Dependency Inversion														
7	Encapsulation														
8	Low coupling														
9	High Cohesion														
10	Composition over Inheritance														
11	DRY														
12	Tell, don't ask														
13															
14	Law of Demeter														
15															
16	Passes all tests														
17	No duplication														
18	Express intent														
19	Lower # of methods and classes														
20															
21	One assertion per test														
22															

USING PRINCIPLES AND RULES...

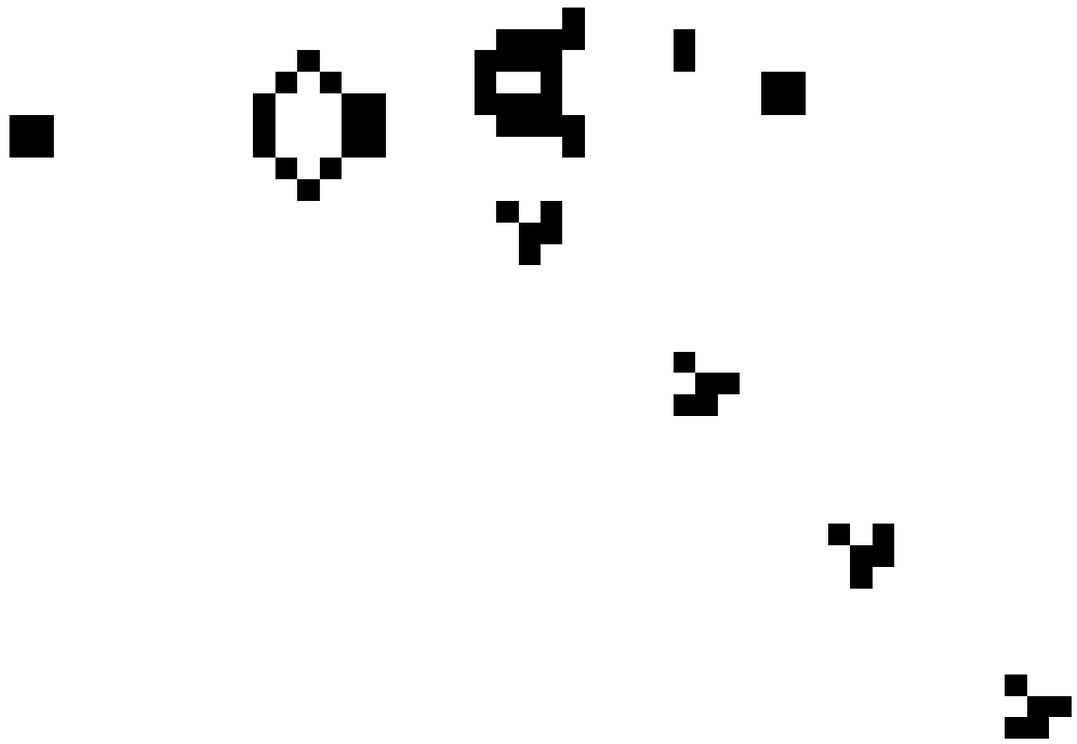
	Novice	Advanced Beginner	Competent	Proficient	Expert
Needs	Monitoring Rule book Instructional Feedback	Guidelines Simple, controlled experiments	Real world exposure	Unrestrained practice Time to reflect	Drive to expand knowledge and experience
How	Checklists Classroom learning "Best Practices"	Simulations Games Q&A	Field trips Case studies Shadowing	Own work allocation Peer-to-peer comparisons Performing mentoring	Exposure to "outside" thinking Time to explore

Patrick Kua - http://www.slideshare.net/thekua/learning-about-learning-with-dreyfus-7924019

Conway's Game of Life

http://en.wikipedia.org/wiki/Conway%27s_Game_of_Life

1. Any **live** cell with **fewer** than **two** live neighbours **dies**,
   as if caused by under-population.
2. Any **live** cell with **two** or **three** live neighbours **lives** on to the next
   generation.
3. Any **live** cell with **more** than **three** live neighbours **dies**,
   as if by overcrowding.
4. Any **dead** cell with **exactly three** live neighbours **becomes a live** cell,
   as if by reproduction.



#5

ThoughtWorks®

As the supreme product owner
I want any cell with >= 4 alive
neighbours to become a zombie
so that apocalypse can begin..

#5

ThoughtWorks®

As the supreme product owner
I want any cell with >= 4 alive
neighbours to become a zombie
so that apocalypse can begin..

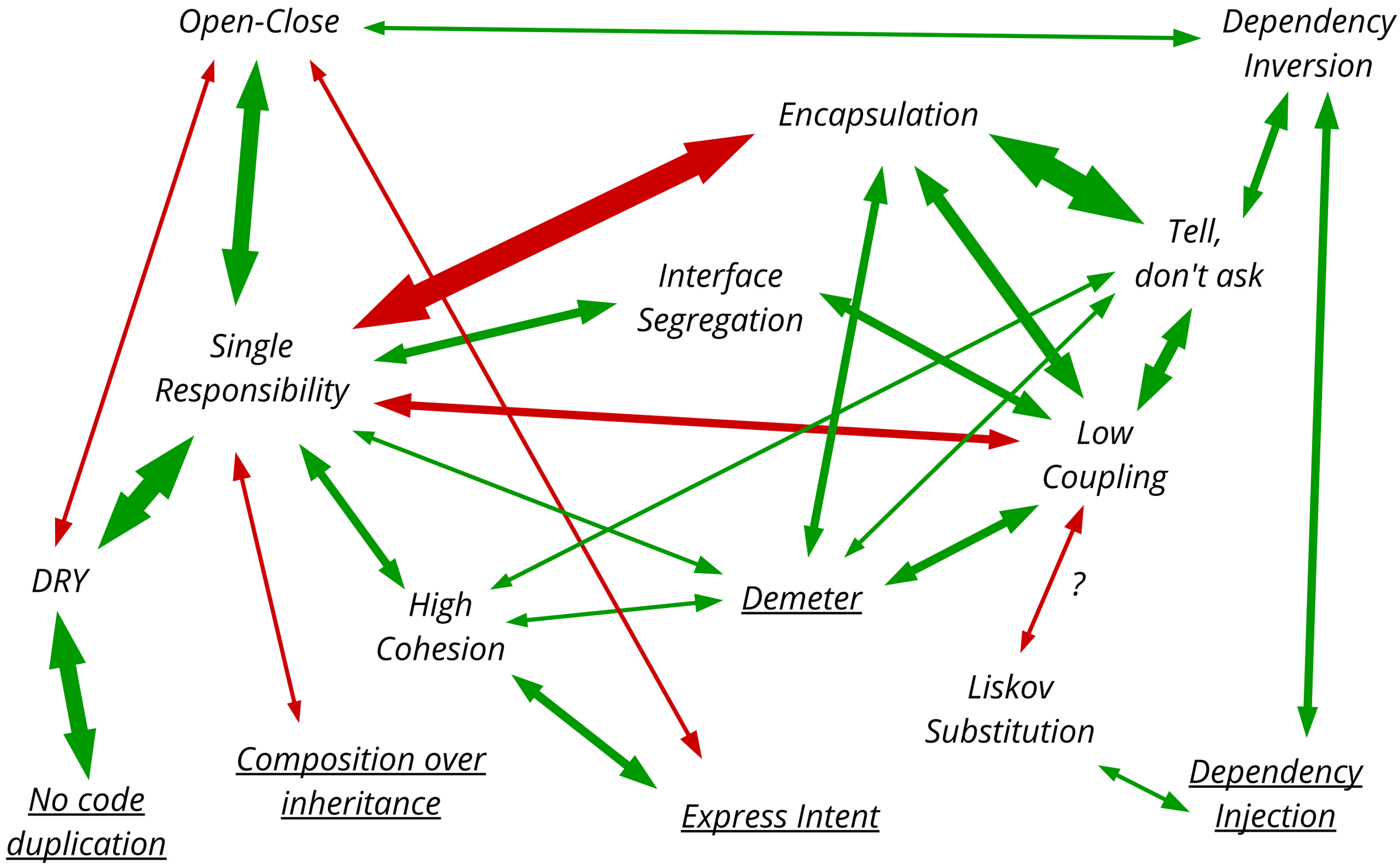
(ps. Zombies will be forever
zombie, cannot die or become alive
again)

Code showcase

Don't worry,

be crappy :-)

TENSIONS AND SYNERGIES (v3.0)
WHAT WE LEARNED SO FAR... ??



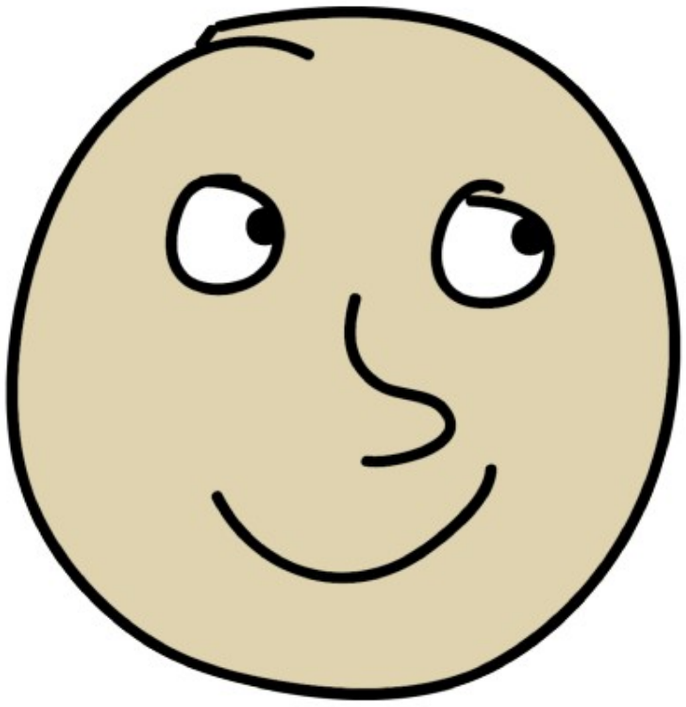
https://docs.google.com/spreadsheets/d/1jiqS5soI0PCo8Rm4M6LFJ20L3I6yJeyuOjLAK23RDWY/edit

Design decisions are trade-offs

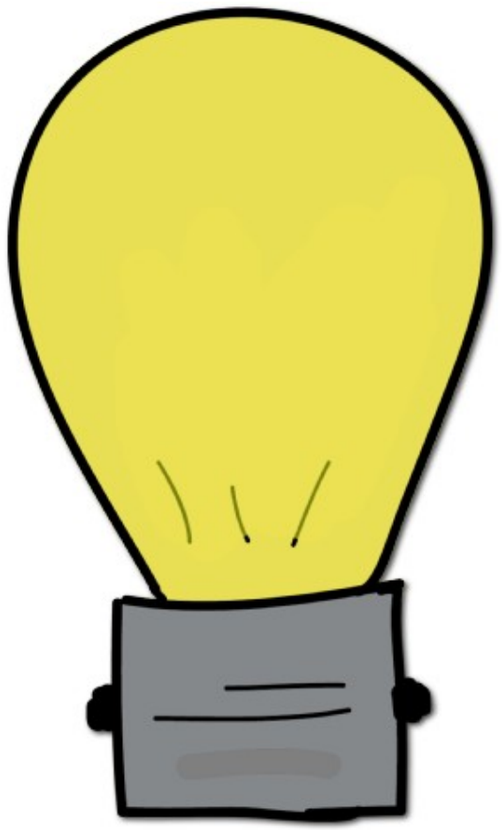
Recap

Questions?

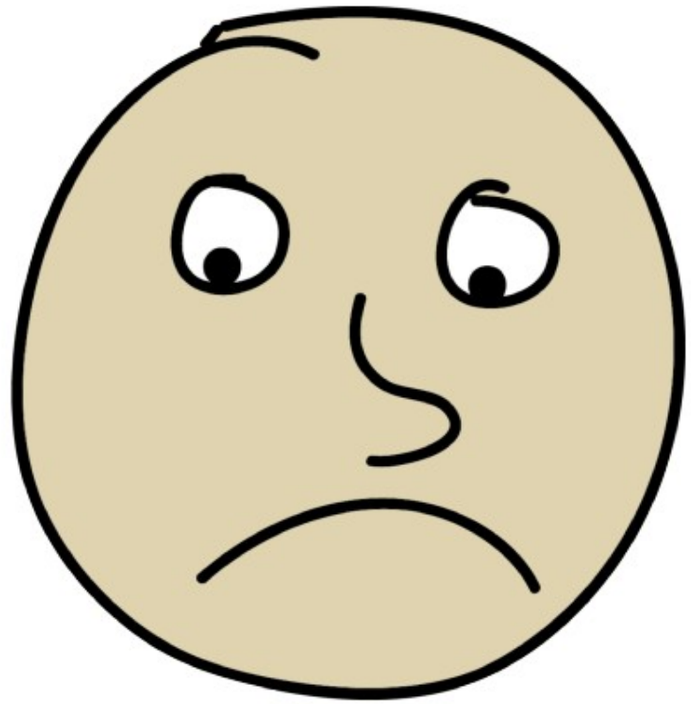
Feedback



LIKED



IDEAS



DISLIKED

THANKS

Modify and reuse CC-BY-SA

https://github.com/iliasbartolini/design_principles_dojo_facilitator_guide

Contributions from:

Luca Minudel

Matteo Vaccari

CC photo credits:

https://www.flickr.com/photos/torek/4444673930

ThoughtWorks®