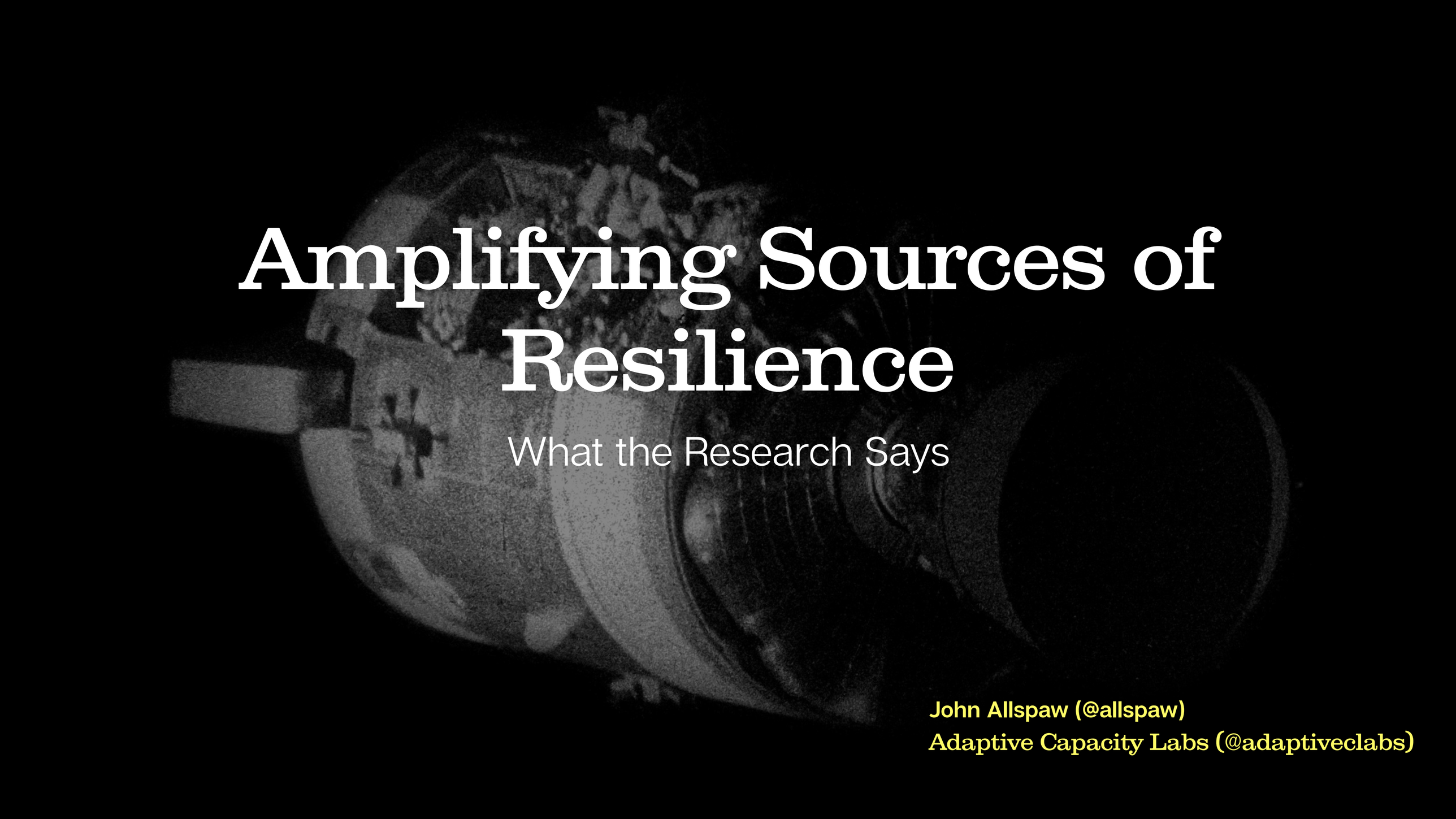


Amplifying Sources of Resilience

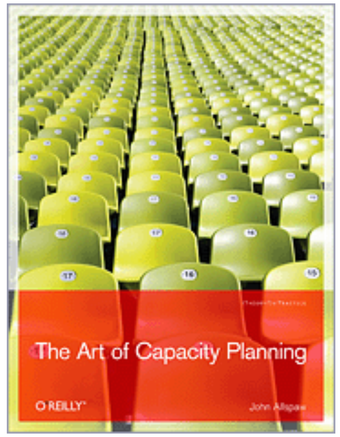
What the Research Says

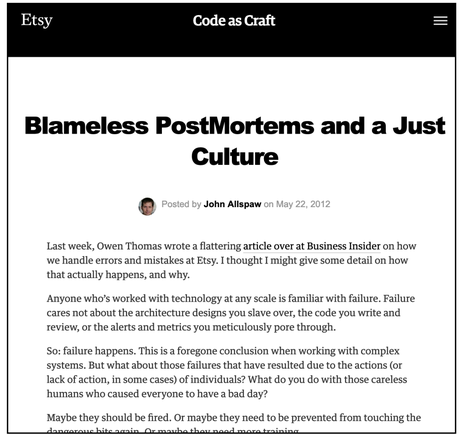
John Allspaw (@allspaw)
Adaptive Capacity Labs (@adaptiveclabs)

me

flickr

Etsy




Etsy Code as Craft

Blameless PostMortems and a Just Culture

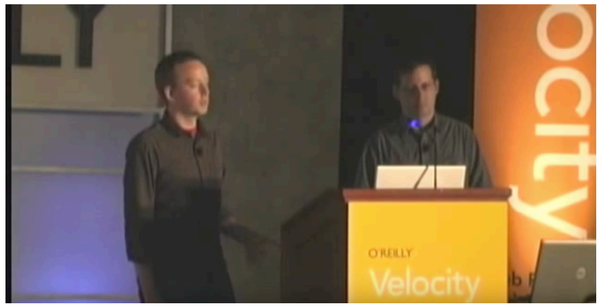
Posted by **John Allspaw** on May 22, 2012

Last week, Owen Thomas wrote a flattering **article over at Business Insider** on how we handle errors and mistakes at Etsy. I thought I might give some detail on how that actually happens, and why.

Anyone who's worked with technology at any scale is familiar with failure. Failure cares not about the architecture designs you slave over, the code you write and review, or the alerts and metrics you meticulously pore through.

So: failure happens. This is a foregone conclusion when working with complex systems. But what about those failures that have resulted due to the actions (or lack of action, in some cases) of individuals? What do you do with those careless humans who caused everyone to have a bad day?

Maybe they should be fired. Or maybe they need to be prevented from touching the



2009 Velocity Conf

LUND UNIVERSITY

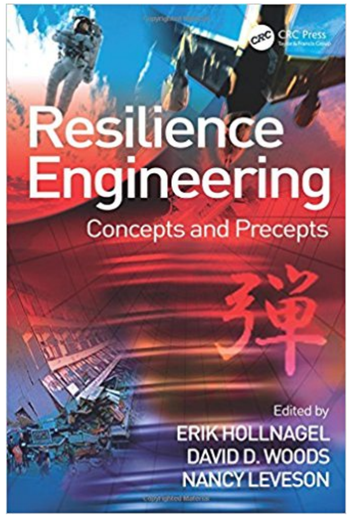
SNAFU CATCHERS

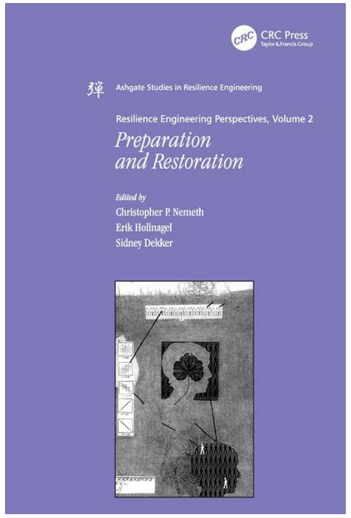
Consortium for Resilient Internet-Facing Business IT

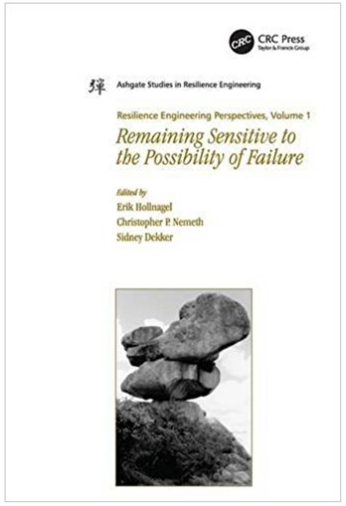
Adaptive Capacity Labs

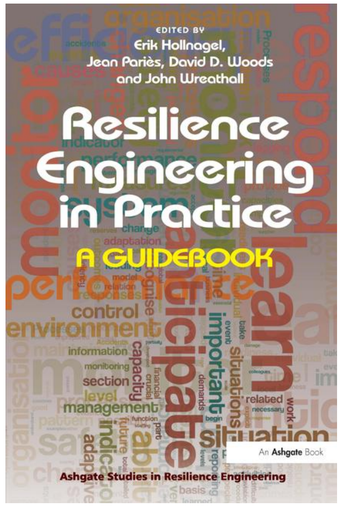
Resilience Engineering Is a Field of Study

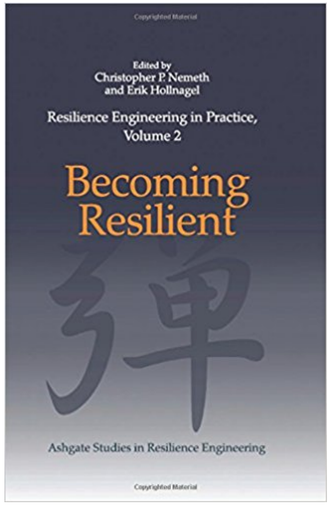
- Emerged from Cognitive Systems Engineering
- Early 2000s, largely in response to NASA events in 1999 and 2000
- 7 symposia over 12 years











Resilience Engineering is a Community

is largely made up of practitioners and researchers from....

Human Factors & Ergonomics

Cognitive Systems Engineering

Complexity Science

Sociology

Cognitive Psychology

Operations Research

Engineering*

Safety Science

Ecology

Cybernetics

Biology

Control Systems

Resilience Engineering is a Community

working in domains such as...

Aviation/ATM

Construction

Mining

Space

Explosives

Surgery

Rail

Pediatrics

Anesthesia

Law Enforcement

Software Engineering

Power Grid & Distribution

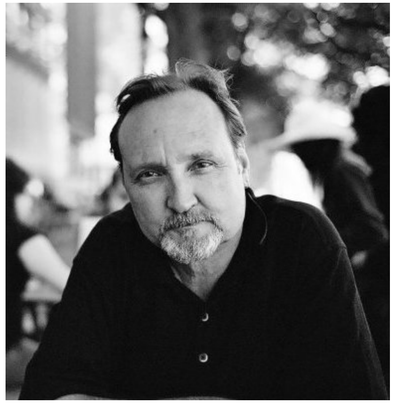
Maritime

Military Agencies

Firefighting

Intelligence Agencies

Some of the cast of characters



David Woods
CSEL/OSU

Shawna Perry
Univ of Florida
Emergency Medicine

Dr. Richard Cook
Anesthesiologist
Researcher

Ivonne Andrade Herrera
SINTEF

Erik Hollnagel
Univ of S. Denmark

Anne-Sophie Nyssen
University de Liege

Johan Bergström
Lund Universitv

Sidney Dekker
Griffith University

Laura Maguire
CSEL/OSU

Asher Balkin
CSEL/OSU

Some of the cast of characters

David Woods
CSEL/OSU

Shawna Perry
Univ of Florida
Emergency Medicine

Dr. Richard Cook
Anesthesiologist
Researcher

Ivonne Andrade Herrera
SINTEF

Erik Hollnagel
Univ of S. Denmark

Anne-Sophie Nyssen
University de Liege

Johan Bergström
Lund University

Sidney Dekker
Griffith University

Laura Maguire
CSEL/OSU

Asher Balkin
CSEL/OSU

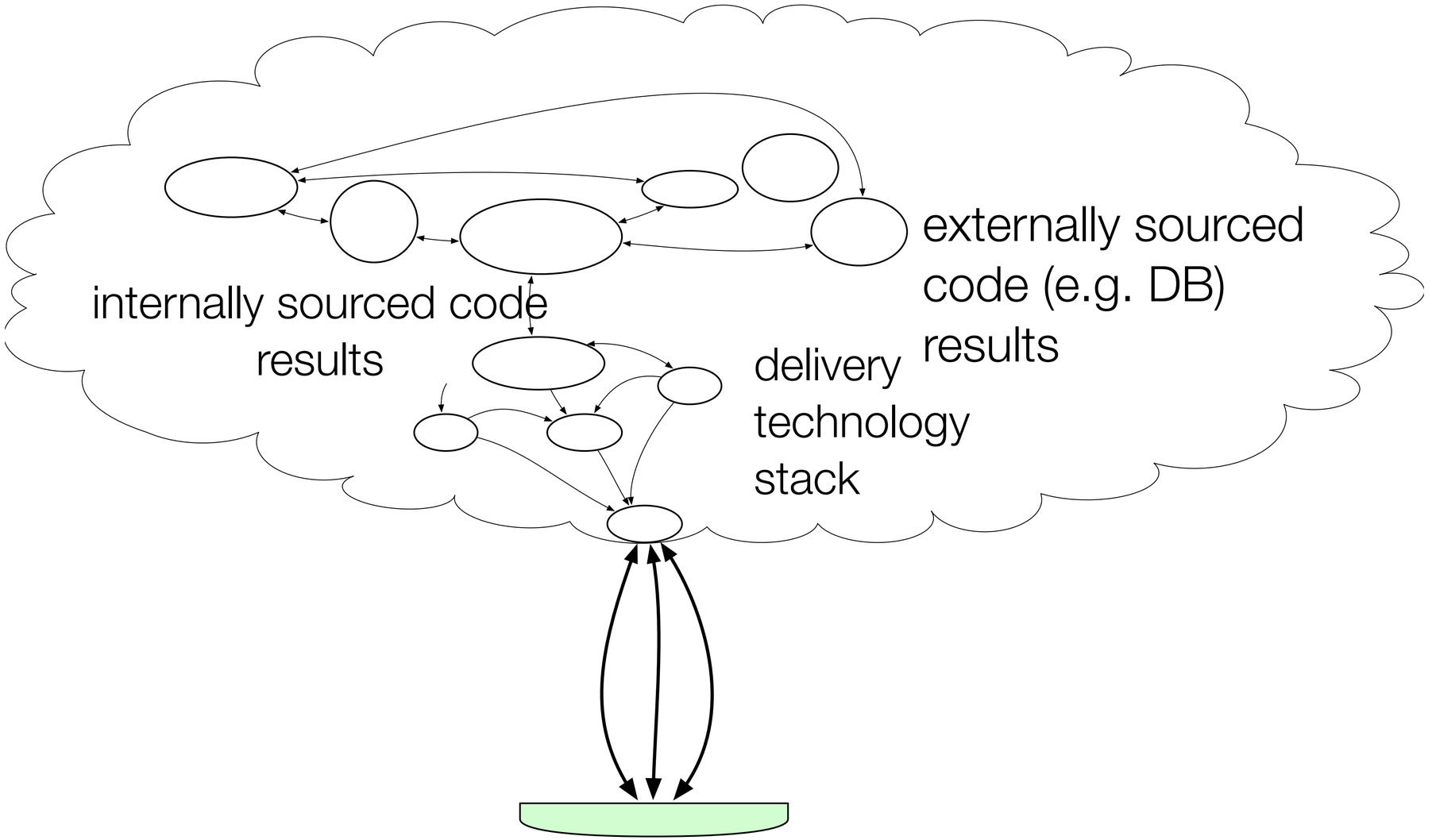
Nora Jones

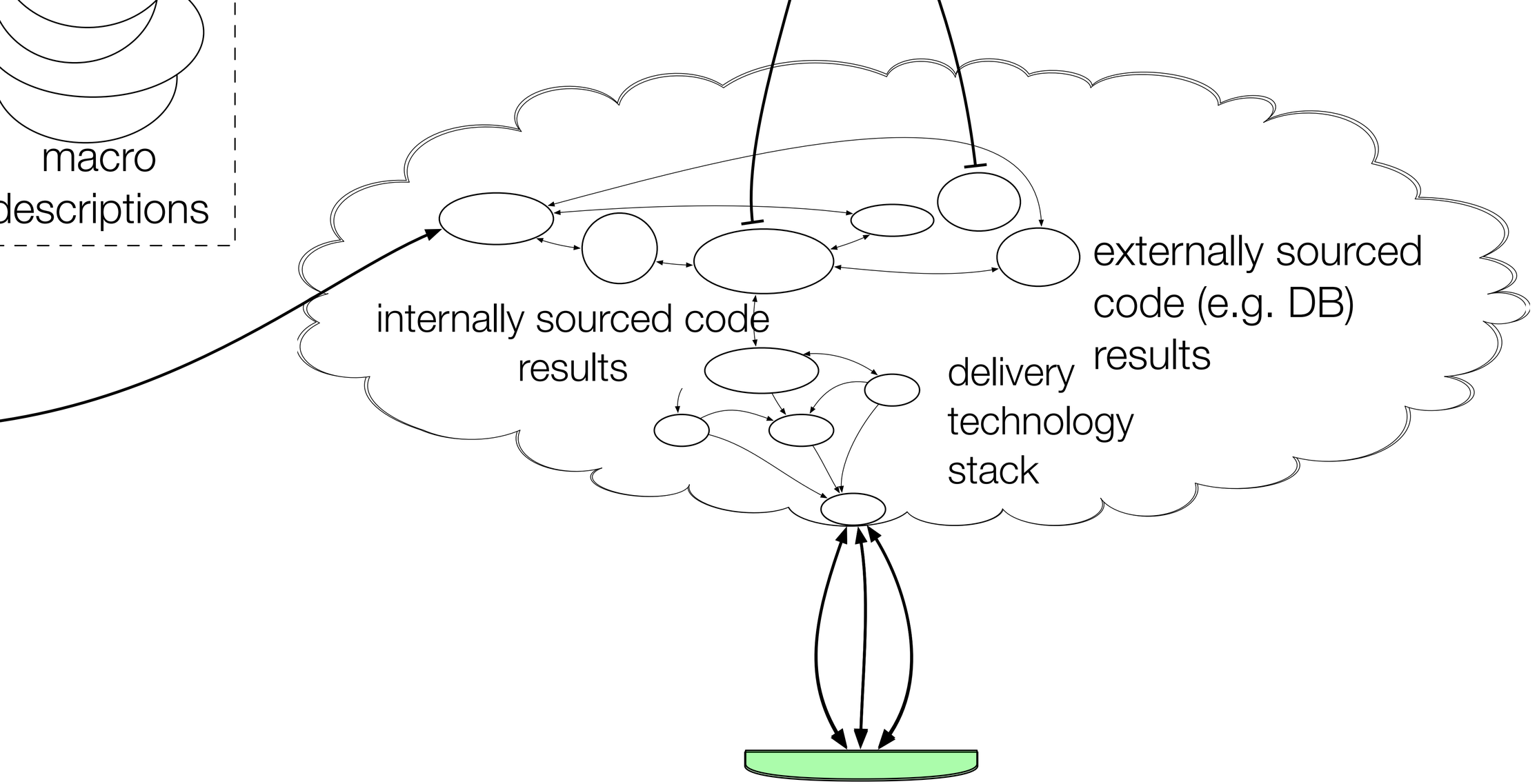
Casey Rosenthal

Jessica DeVita

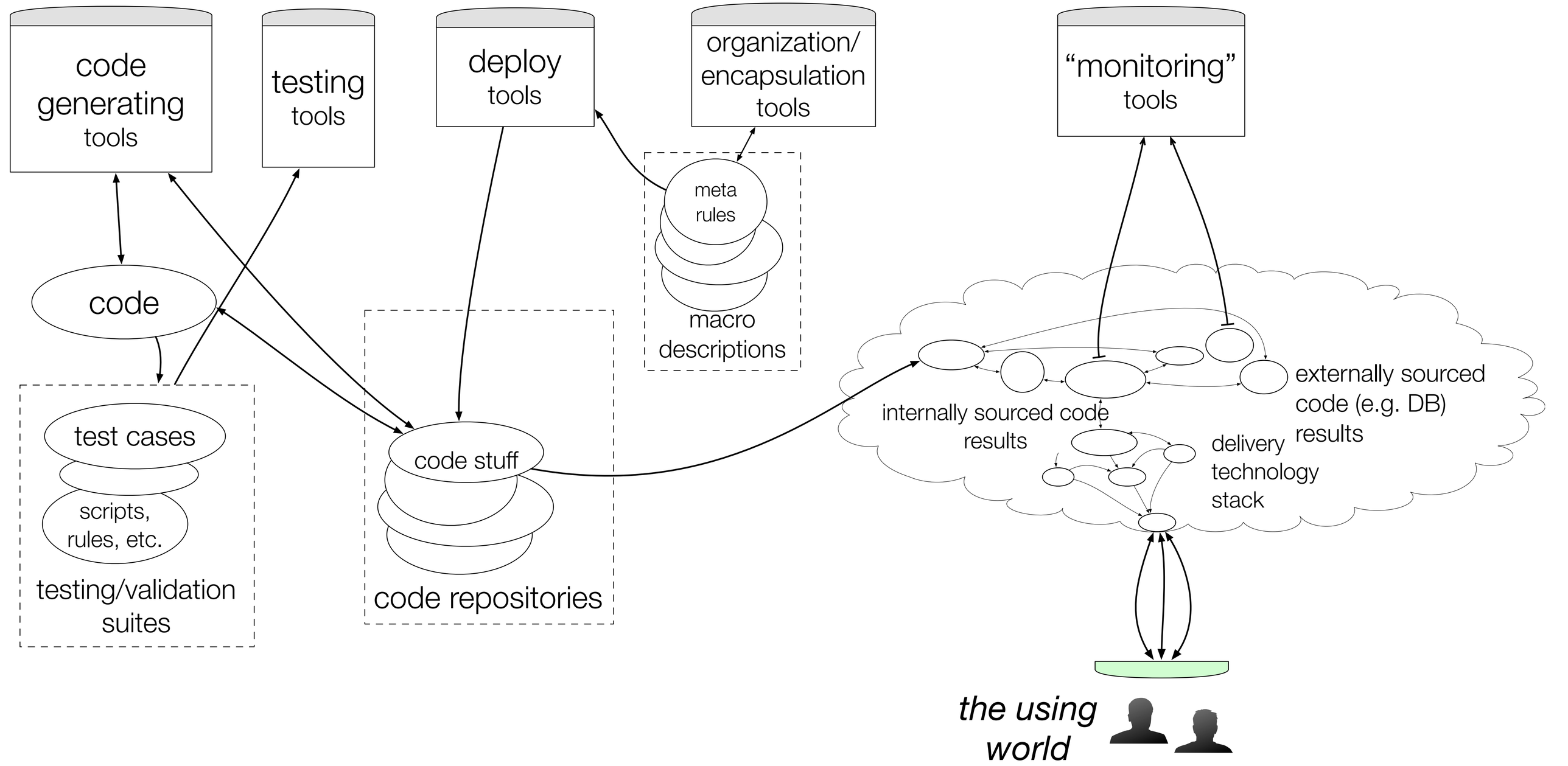
J. Paul Reed

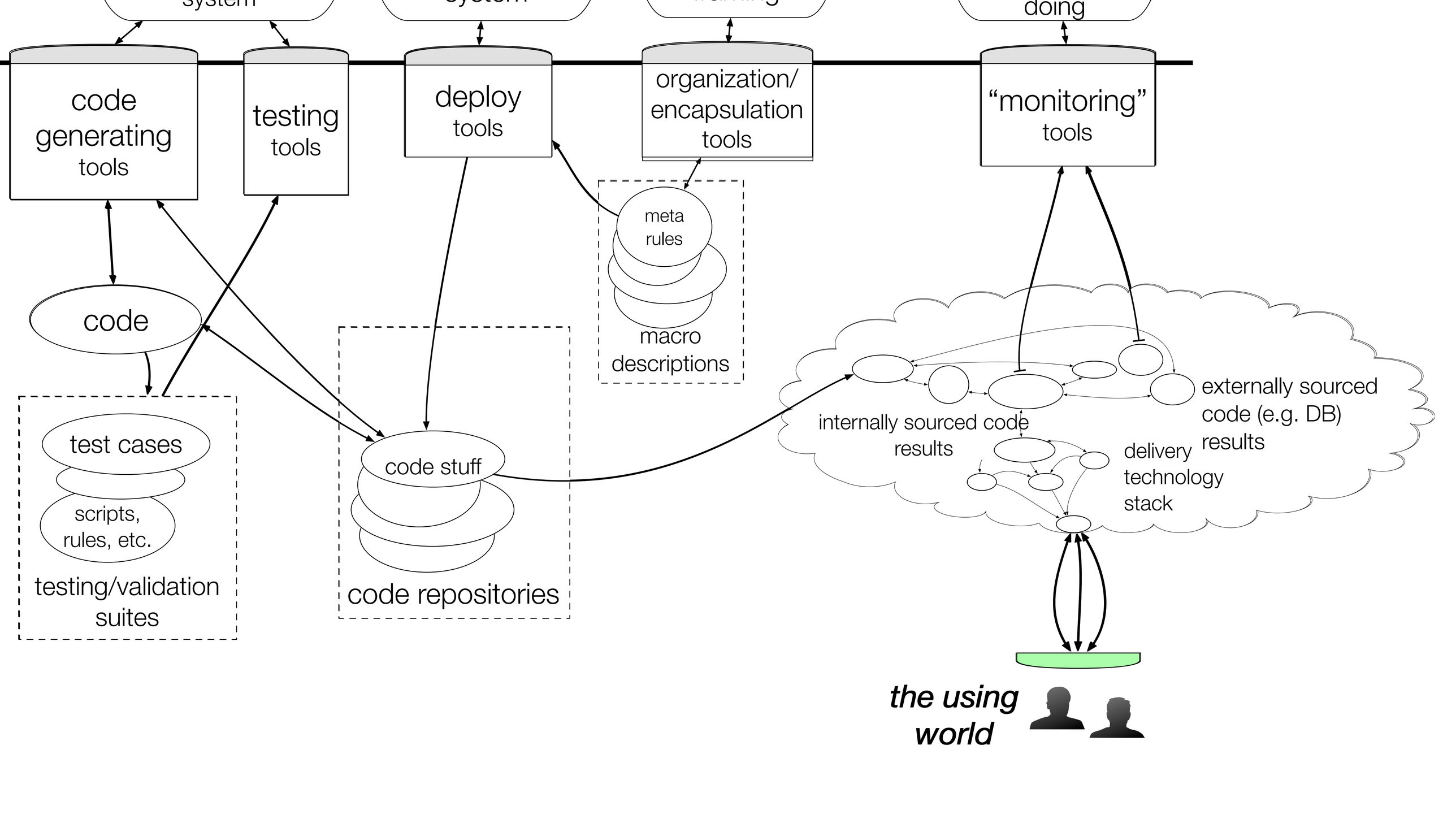
(me)

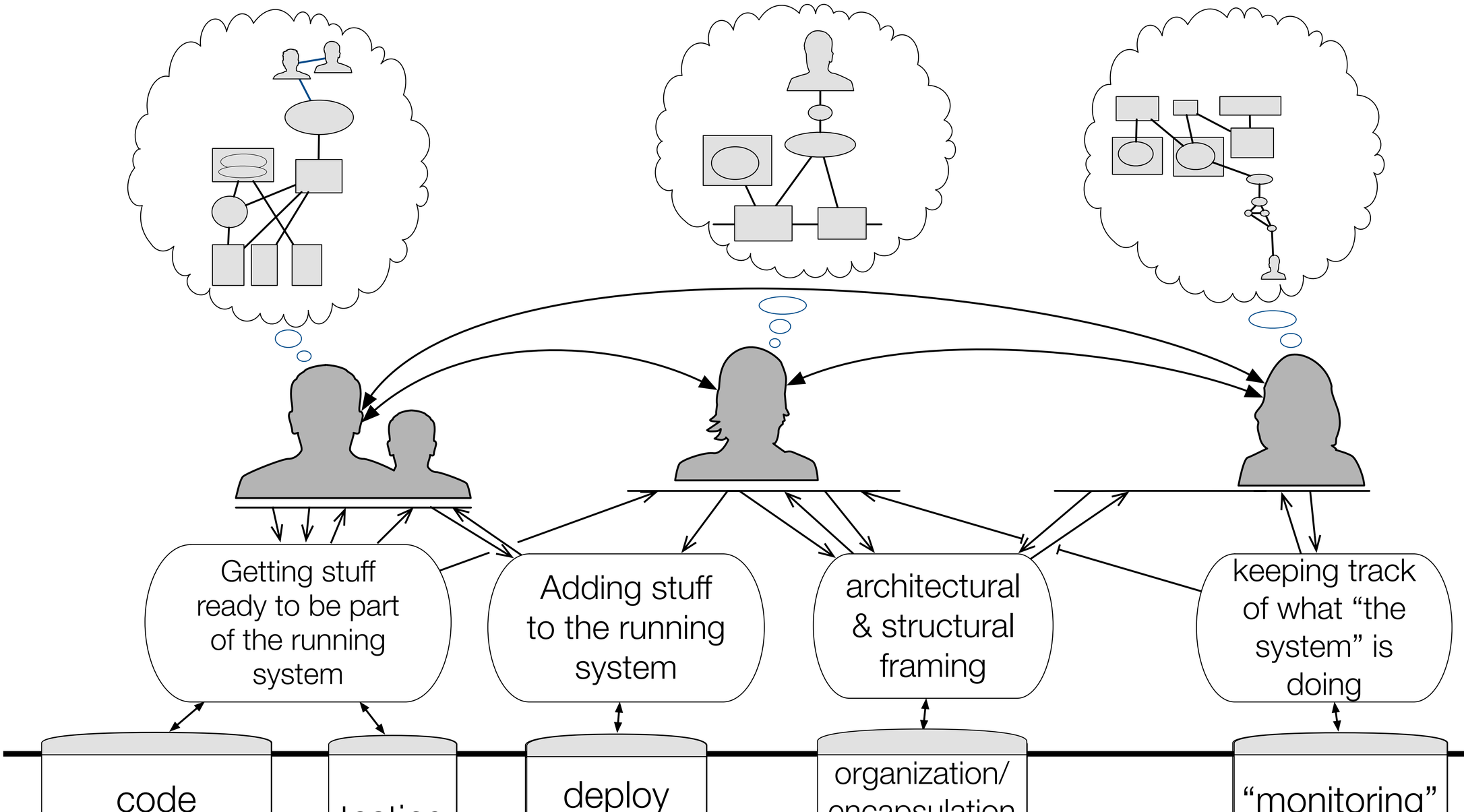

externally sourced
code (e.g. DB)
results
internally sourced code
results
delivery
technology
stack
the using world

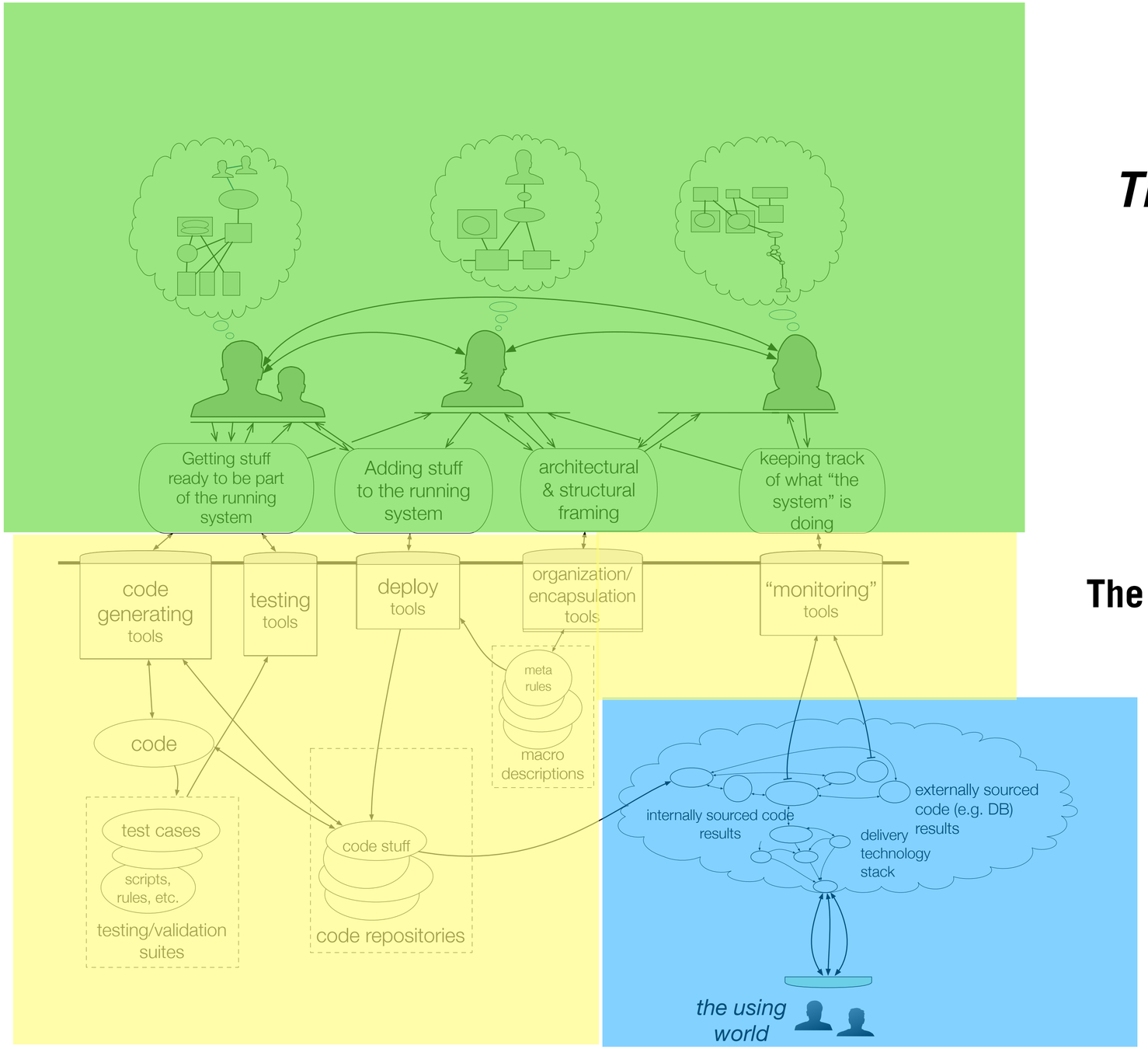


the using world


code generating tools
testing tools
deploy tools
organization/ encapsulation tools
"monitoring" tools
meta rules
macro descriptions
code
test cases
scripts, rules, etc.
testing/validation suites
code stuff
code repositories
internally sourced code results
externally sourced code (e.g. DB) results
delivery technology stack
the using world


doing
code generating tools
testing tools
deploy tools
organization/ encapsulation tools
"monitoring" tools
meta rules
macro descriptions
code
test cases
scripts, rules, etc.
testing/validation suites
code stuff
code repositories
internally sourced code results
externally sourced code (e.g. DB) results
delivery technology stack
the using world

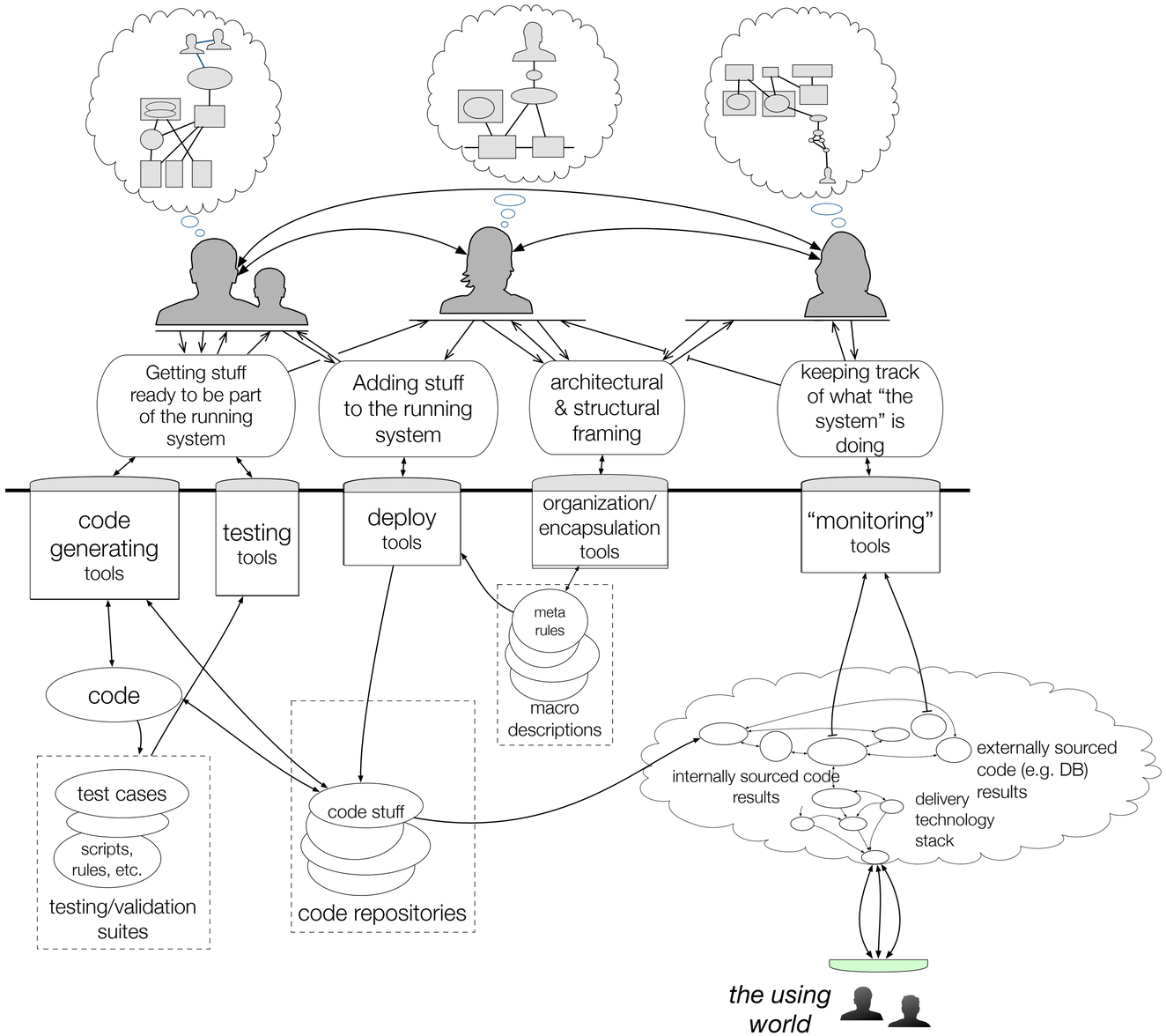

Getting stuff ready to be part of the running system
Adding stuff to the running system
architectural & structural framing
keeping track of what “the system” is doing
code
deploy
organization/ encapsulation
“monitoring”

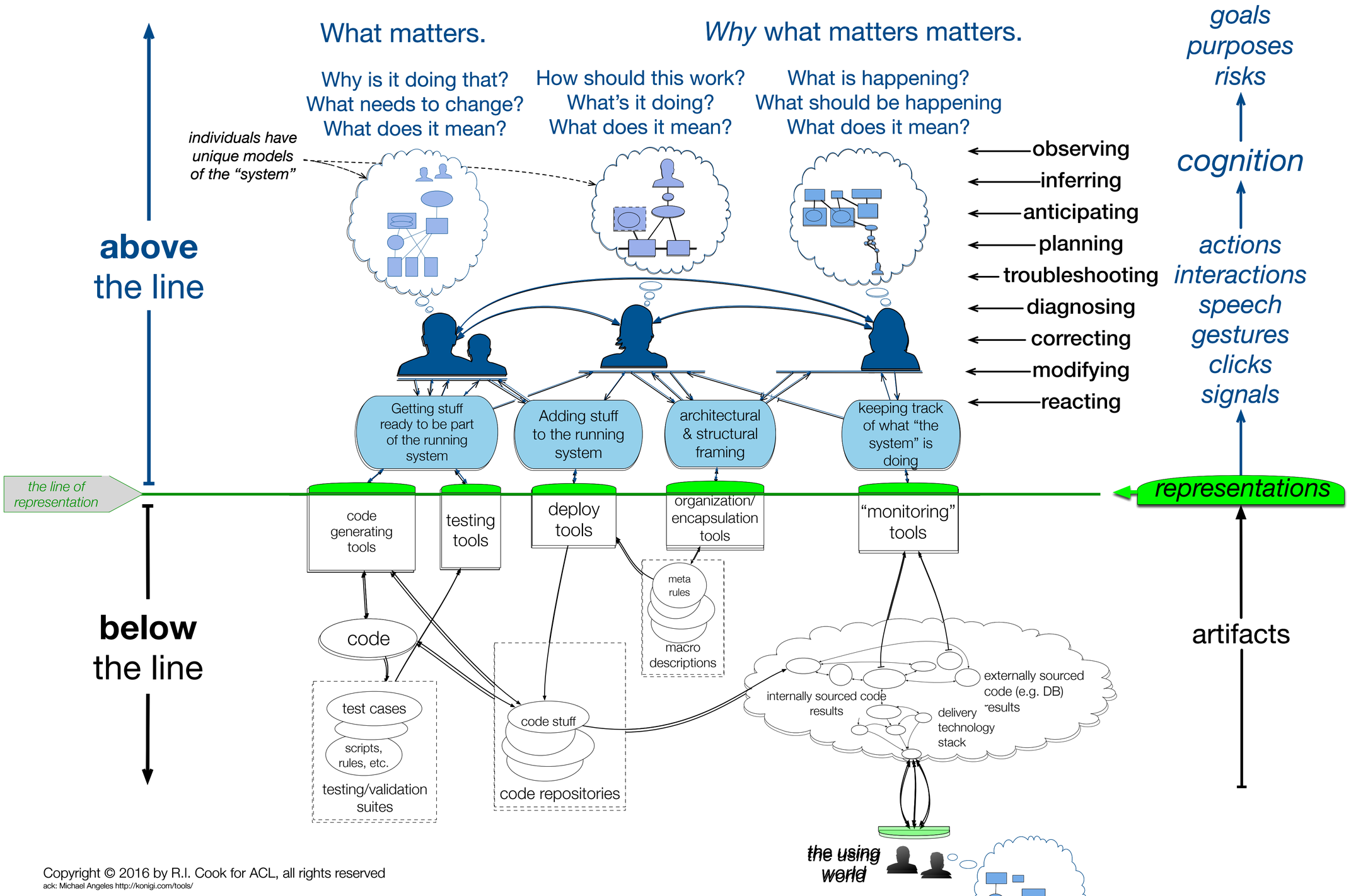


The Work Is Done Here

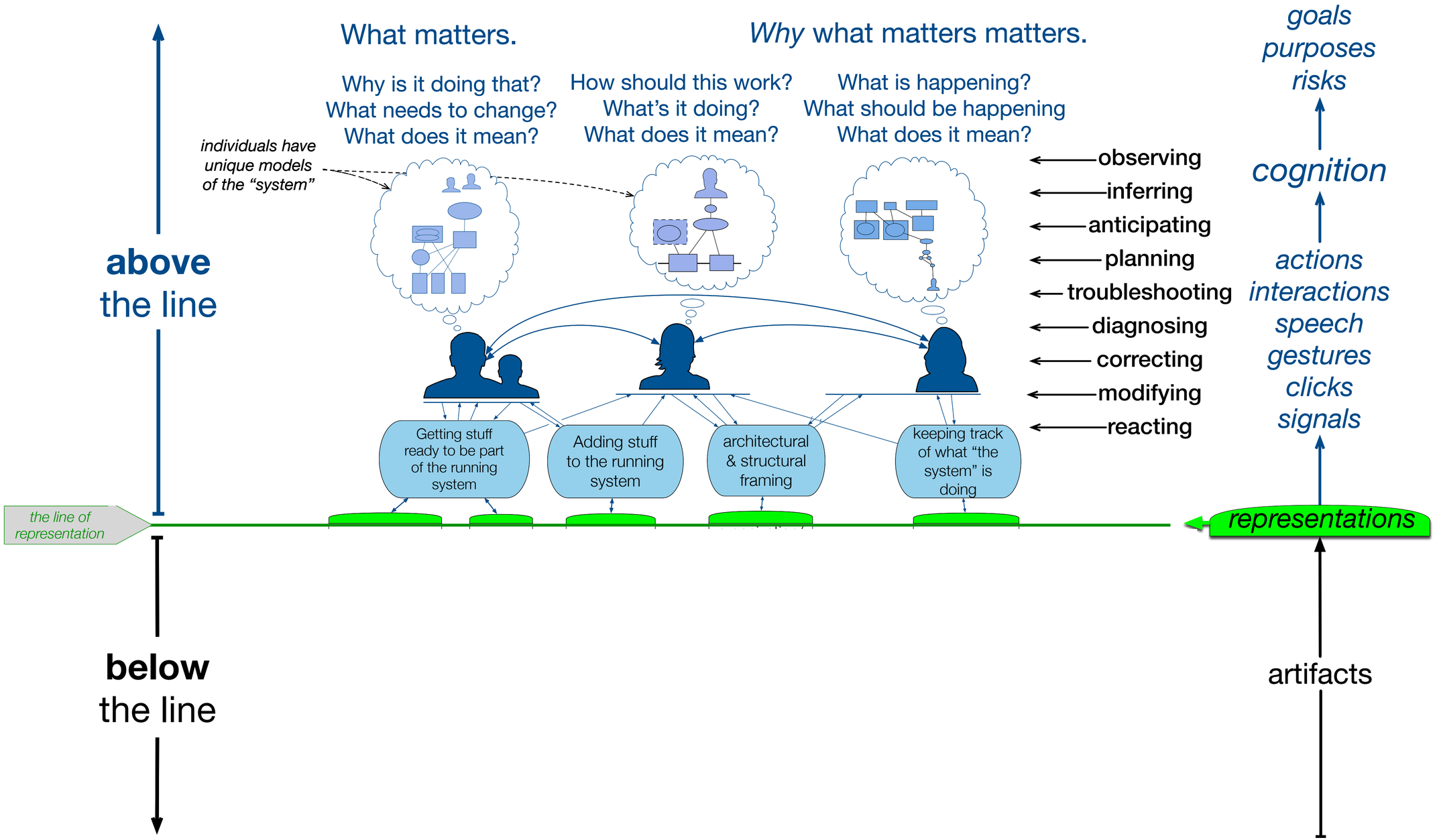
The Stuff You Build and Maintain With

Your Product Or Service


Getting stuff ready to be part of the running system
Adding stuff to the running system
architectural & structural framing
keeping track of what "the system" is doing
code generating tools
testing tools
deploy tools
organization/ encapsulation tools
"monitoring" tools
meta rules
macro descriptions
code
test cases
scripts, rules, etc.
testing/validation suites
code stuff
code repositories
internally sourced code
results
externally sourced code (e.g. DB) results
delivery technology stack
the using world



ack: Michael Angeles http://konigi.com/tools/



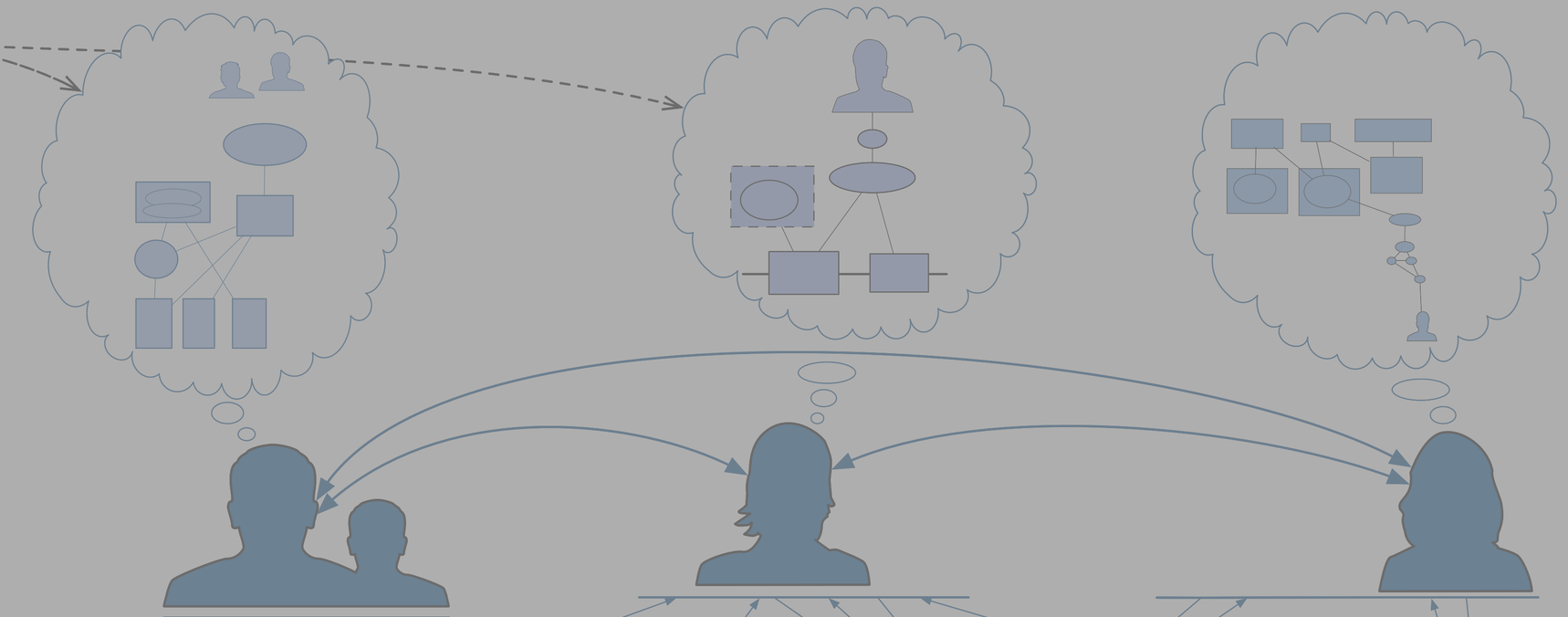
What matters.

Why what matters matters.

Why is it doing that?
at needs to change?
What does it mean?

How should this work?
What's it doing?
What does it mean?

What is happening?
What should be happening
What does it mean?



- observing
- inferring
- anticipating
- planning
- troubleshooting
- diagnosing
- correcting
- modifying
- reacting

Getting stuff ready to be part of the running system

Adding stuff to the running system

architectural & structural framing

keeping track of what "the system" is doing

code

testing

deploy

organization/ encapsulation

"monitoring"

cogn

act

intera

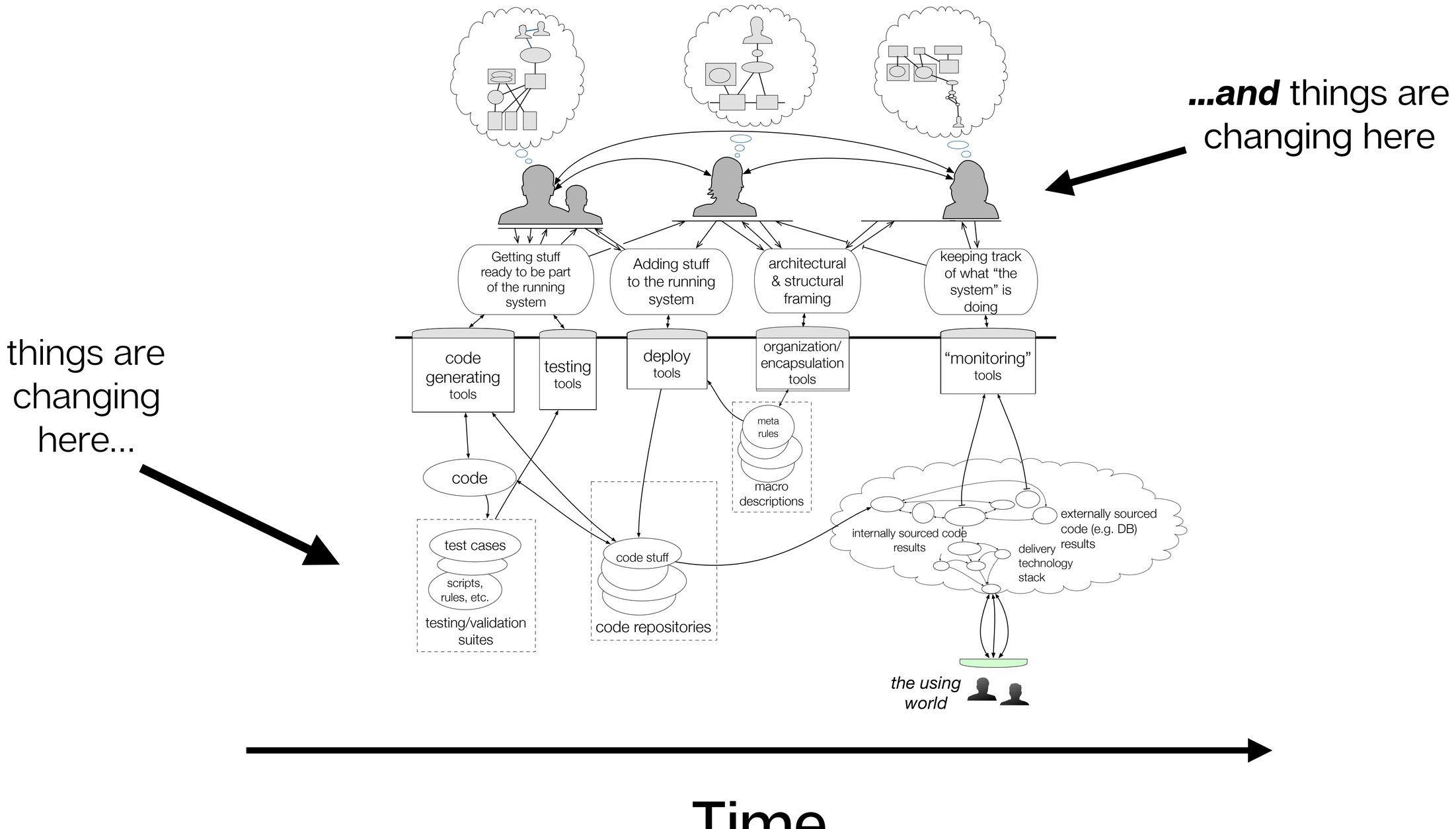
spe

ges

cl

sig

represe


...and things are changing here
things are changing here...
Getting stuff ready to be part of the running system
Adding stuff to the running system
architectural & structural framing
keeping track of what "the system" is doing
code generating tools
testing tools
deploy tools
organization/ encapsulation tools
"monitoring" tools
meta rules
macro descriptions
code
test cases
scripts, rules, etc.
testing/validation suites
code stuff
code repositories
internally sourced code results
externally sourced code (e.g. DB) results
delivery technology stack
the using world
Time

can’t amplify something if you can’t find it first

Amplifying Sources of Resilience

What the Research Says

clarifications about what this actually is

John Allspaw
Adaptive Capacity Labs

what we find when we look closely at incidents in software


tracing/
observability tools
alerts
what's been happening
in the world
(news, service provider
outages, etc.)
recent
changes in
existing tech
time-series
data
new
dependencies
what has been
investigated thus far
status of other
ongoing work
observations and
hypotheses others
share
logs
time of year
day of year
time of day
who is on
vacation, at a
conference,
traveling, etc.

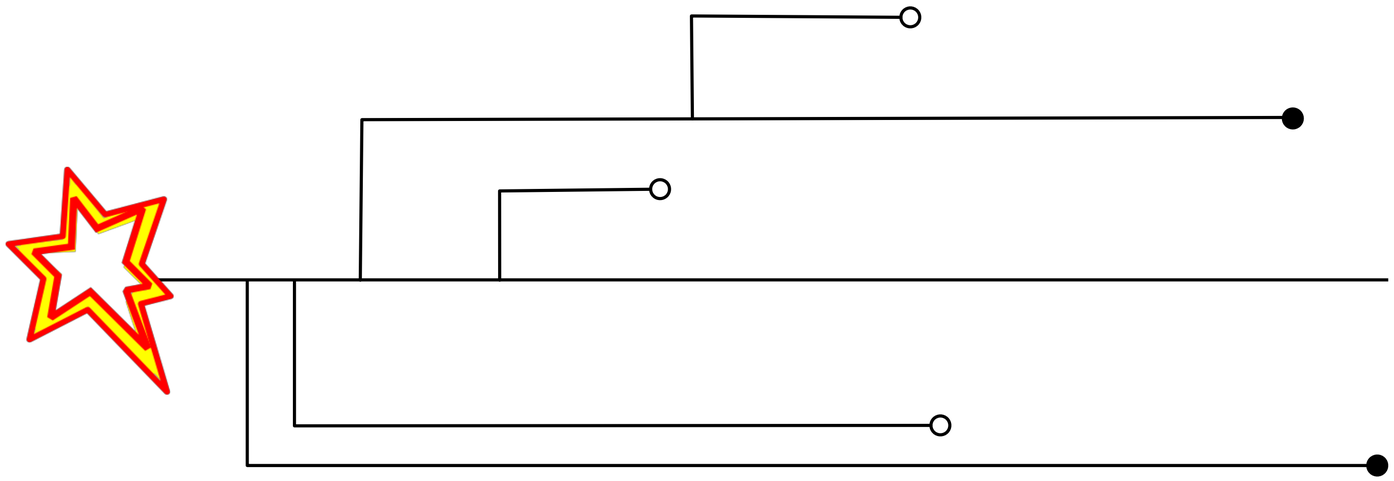
- tenure
- domain expertise
- past experience with details

multiple perspectives on

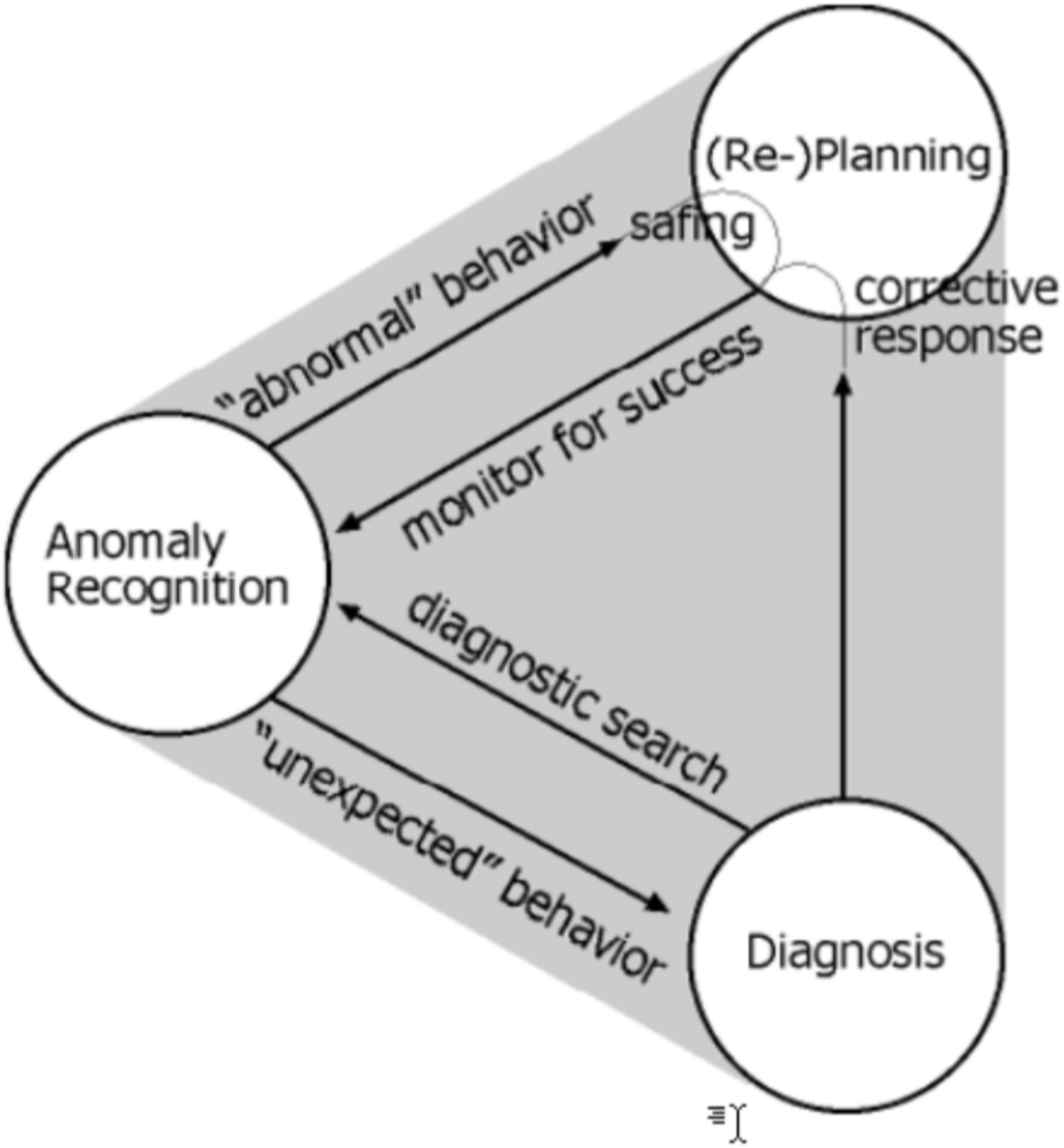
- what “it” is that is happening
- what can – and what *cannot* – be done to “stem the bleeding” or “reduce the blast radius”
- who has authority to take certain actions
- what *shouldn’t* be tried to mitigate or repair

multiple threads of activity

some productive
some unproductive



- problem detection
- generating hypotheses
- diagnostic actions
- therapeutic actions
- sacrifice decisions
- coordinating
- (re) planning
- preparing for potential escalation/cascades



time pressure
high consequences

this is not

“debugging”
“troubleshooting”

therefore…“resilience”?

Ward Cunningham

1992

software development may incur future liability in order to achieve short-term goals

THANK YOU, WARD!

resilience is *not:*

- preventative design
- fault-tolerance
- redundancy
- Chaos Engineering
- stuff about software or hardware
- a property that a system ***has***

unforeseen

unanticipated

unexpected

fundamentally surprising

“things that have never happened before happen all the time”

–Scott Sagan “The Limits of Safety”

resilience is:

- proactive activities aimed at **preparing to be unprepared** – *without an ability to justify it economically!*
- sustaining the potential for future adaptive action when conditions change
- something that a system ***does,*** *not what it* ***has***

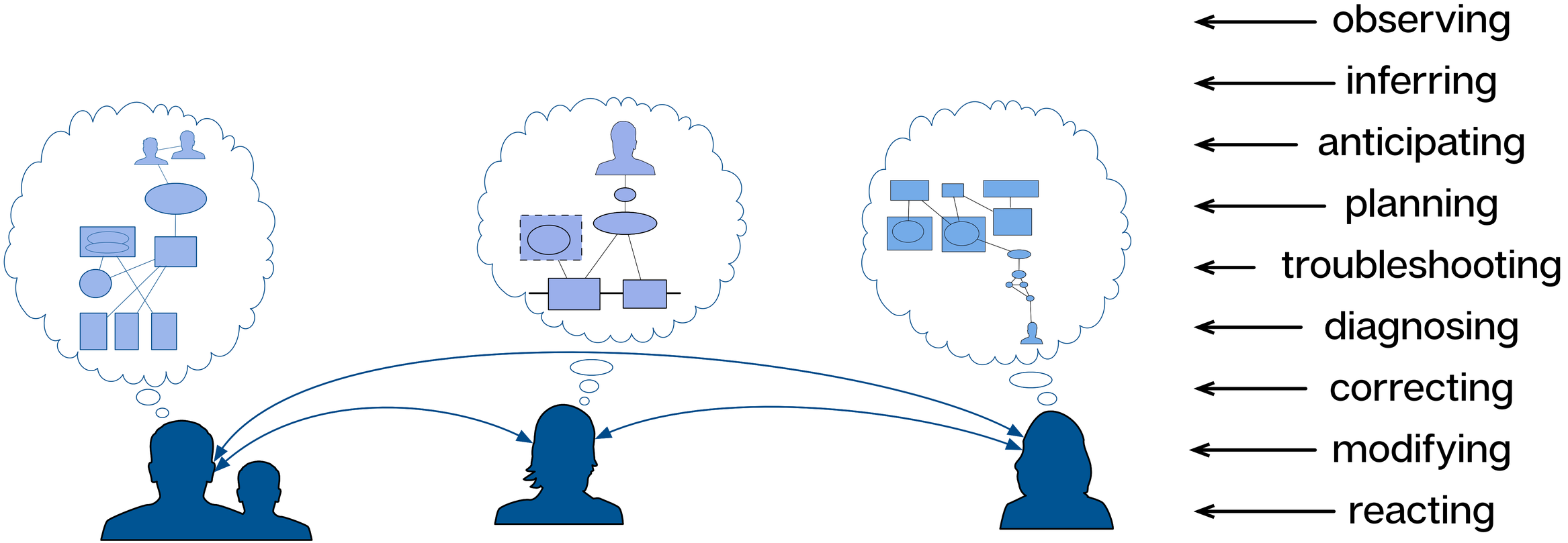
sustained
adaptive capacity

sustained
adaptive capacity

Poised To Adapt

1. Knowing what the platform is supposed to do
2. Knowing how the platform works
3. What the platform's behavior means
4. Being able to devise a change that addresses 1, 2, & 3
5. Being able to predict the effects of that change
6. Being able to force the platform to change in that way
7. Being prepared to deal with the consequences

Dr. Richard Cook, Velocity Conf 2016 Santa Clara, CA



Finding sources of resilience means finding and understanding *cognitive work*.

all incidents can be worse

what are things (people, maneuvers, knowledge, etc.) that went into
preventing it from being worse?

How can I find this “adaptive capacity”?

Find incidents that have:

- high degree of **surprise**
- whose consequences were **not severe**
- and look closely at the details about what went into making it not nearly as bad as it could have been
- protect and acknowledge explicitly the sources you find

indications of surprise and novelty

<murphy> wtf happened here

<steve> I have no idea what is going on

<laurie> well that's terrifying

indications about contrasting mental models

<jeremy> I'm still a bit confused why B and A are different if A got to 0 and B is still at 3099

<lisa> wait wait, i thought the X table was small

<laurie> so you want to rebuild {server01} first?

<laurie> neither box has been touched yet

<laurie> and im a tad nervous to do both at once

<tim>: oh I see.. the retry interval is pretty aggressive

why not look at incidents with ***severe*** consequences?

- scrutiny from stakeholders with face-saving agenda tend to block deep inquiry
- with “medium-severe” incidents the cost of getting details/descriptions of people’s perspectives is low relative to the potential gain
- “Goldilocks” incidents are the ideal

some (contextual) sources

- esoteric knowledge and expertise in the organization
- flexible and dynamic staffing for novel situations
- authority that is **expected** to migrate across roles
- a “*constant sense of unease*” that drives explorations of “normal” work
- capture and dissemination of ***near-misses***

Summary!

- Resilience is something a system **does,** not what a system **has.**
- Creating and sustaining adaptive capacity in an organization while being unable to justify doing it specifically = resilient action.
- How people (the flexible elements of The System™) cope with surprise is the path to finding sources of resilience

Resilience is the story of the outage that didn’t happen.

Thank You!


@allspaw


@AdaptiveCLabs

Resilience Is A Verb (Woods, 2018)
http://bit.ly/ResilienceIsAVerb

Stella Report
http://stella.report

How Complex Systems Fail (Cook, 1998)
http://bit.ly/ComplexSystemsFailure

SRE Cognitive Work
(chapter in Seeking SRE, O'Reilly Media)
http://bit.ly/SRECognitiveWork

https://www.adaptivecapacitylabs.com/blog