Polyglot Agile

Aino Vonge Corry, ph.d.
Metadeveloper
@apaipi

GOTO Chicago 2018

Polyglot Agile

A coach’s tale by Aino Vonge Corry

GOTO Chicago 2018

Takeaway message

Agility is a journey

Always inspect and adapt - the process!

It is a continuous investment

Don’t be afraid - there is hope

Once upon a time in Denmark

The heart of being agile

Growth mindset

Inspect and adapt

Flexibility

Examples of organisations

Small startup - agile is in-bred

Big enterprise - agile is decided upon

Another big, but younger enterprise - agile is natural

Examples of processes

Scrum

Kanban

Less

Safe

XP

...

One enterprise

Motivation on all levels

Education

External coaches

User groups

Stage-Gate model - agility in the middle 3 stages

Possibility to run waterfall as well as Scrum

But there be Dragons

The destroyed team

But there be Dragons

The destroyed team

The lack of time to learn

But there be Dragons

The destroyed team

The lack of time to learn

Time spent on OKRs

But there be Dragons

The destroyed team

The lack of time to learn

Time spent on OKRs

The explosion of roles

But there be Dragons

The destroyed team

The lack of time to learn

Time spent on OKRs

The explosion of roles

The inability to react to feedback

But there be Dragons

The destroyed team

The lack of time to learn

Time spent on OKRs

The explosion of roles

The inability to react to feedback

The rest of the business is business as usual

Manifesto for Half-Arsed Agile Software Development

We have heard about new ways of developing software by paying consultants and reading Gartner reports. Through this we have been told to value:

Individuals and interactions over processes and tools

and we have mandatory processes and tools to control how those individuals (we prefer the term 'resources') interact

Working software over comprehensive documentation

as long as that software is comprehensively documented

Customer collaboration over contract negotiation

within the boundaries of strict contracts, of course, and subject to rigorous change control

Responding to change over following a plan

provided a detailed plan is in place to respond to the change, and it is followed precisely

That is, while the items on the left sound nice in theory, we're an enterprise company, and there's no way we're letting go of the items on the right.

Cobbled together one Saturday morning before breakfast by Kerry Buckley (@kerryb), following an article by Ron Jeffries and this suggestion from Eastmad.

Manifesto for Half-Arsed Agile Software Development

We have heard about new ways of developing software by paying consultants and reading Gartner reports. Through

Individuals and interactions over

processes and tools

and we have mandatory processes and tools to control how those individuals (we prefer the term 'resources') interact

That is, while the items on the left sound nice in theory, we're an enterprise company, and there's no way we're letting go of the items on the right.

Cobbled together one Saturday morning before breakfast by Kerry Buckley (@kerryb), following an article by Ron Jeffries and this suggestion from Eastmad.

Manifesto for Half-Arsed Agile Software Development

We have heard about new ways of developing software by paying consultants and reading Gartner reports. Through

Working software over

comprehensive documentation .

As long as that software is comprehensively documented

That is, while the items on the left sound nice in theory, we're an enterprise company, and there's no way we're letting go of the items on the right.

Cobbled together one Saturday morning before breakfast by Kerry Buckley (@kerryb), following an article by Ron Jeffries and this suggestion from Eastmad.

Manifesto for Half-Arsed Agile Software Development

We have heard about new ways of developing software by paying consultants and reading Gartner reports. Through

Customer collaboration over contract negotiation

Within the boundaries of strict contracts, of course, and subject to rigorous change control

That is, while the items on the left sound nice in theory, we're an enterprise company, and there's no way we're letting go of the items on the right.

Cobbled together one Saturday morning before breakfast by Kerry Buckley (@kerryb), following an article by Ron Jeffries and this suggestion from Eastmad.

Manifesto for Half-Arsed Agile Software Development

We have heard about new ways of developing software by paying consultants and reading Gartner reports. Through

Responding to change over following a plan

Provided a detailed plan is in place to the change, and it is followed precisely

That is, while the items on the left sound nice in theory, we're an enterprise company, and there's no way we're letting go of the items on the right.

Cobbled together one Saturday morning before breakfast by Kerry Buckley (@kerryb), following an article by Ron Jeffries and this suggestion from Eastmad.

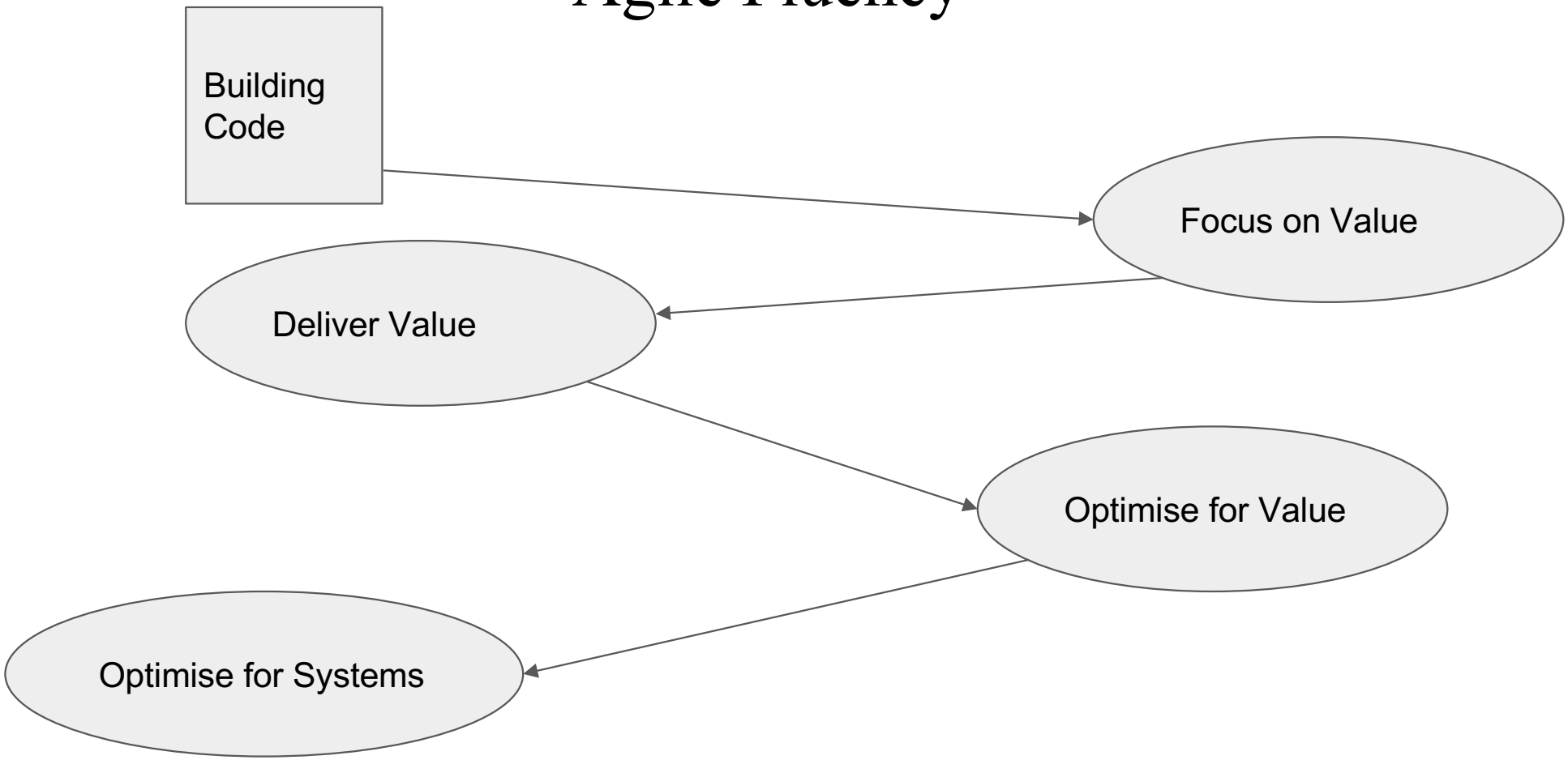
Sad Hobbit

The only consolation was……other people fail as well

The only consolation was……other people fail as well

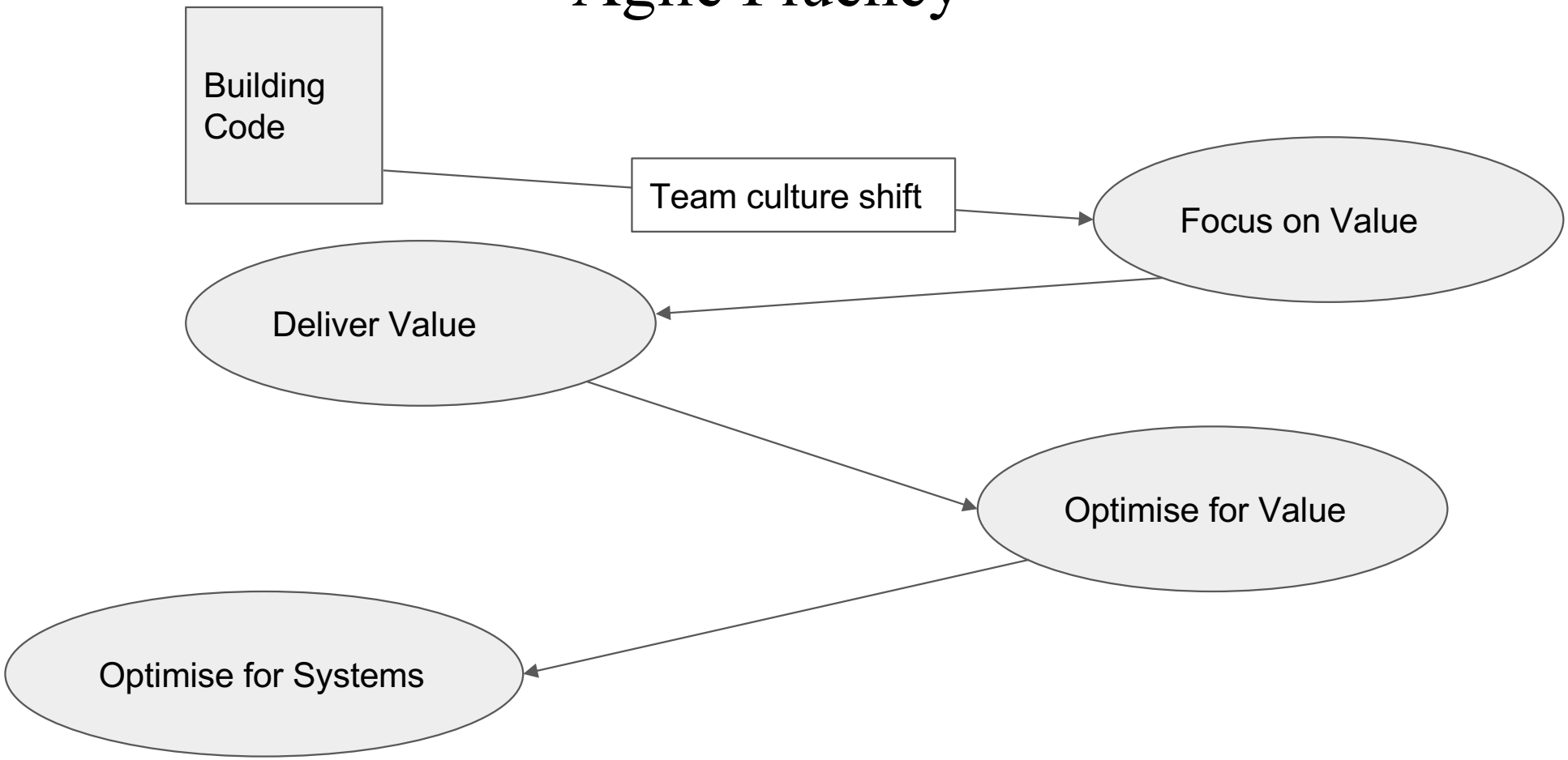
No support for cross-functional teams
Dedicated teams is not possible
Accepting changing priorities Culture and mindset towards waterfall
Stakeholder expectations
Lack of customer focus
Ambiguous definition of "done"
Interruptions
Lack of focus on removing impediments
Low test coverage
Too many discussions without action

Agile Fluency



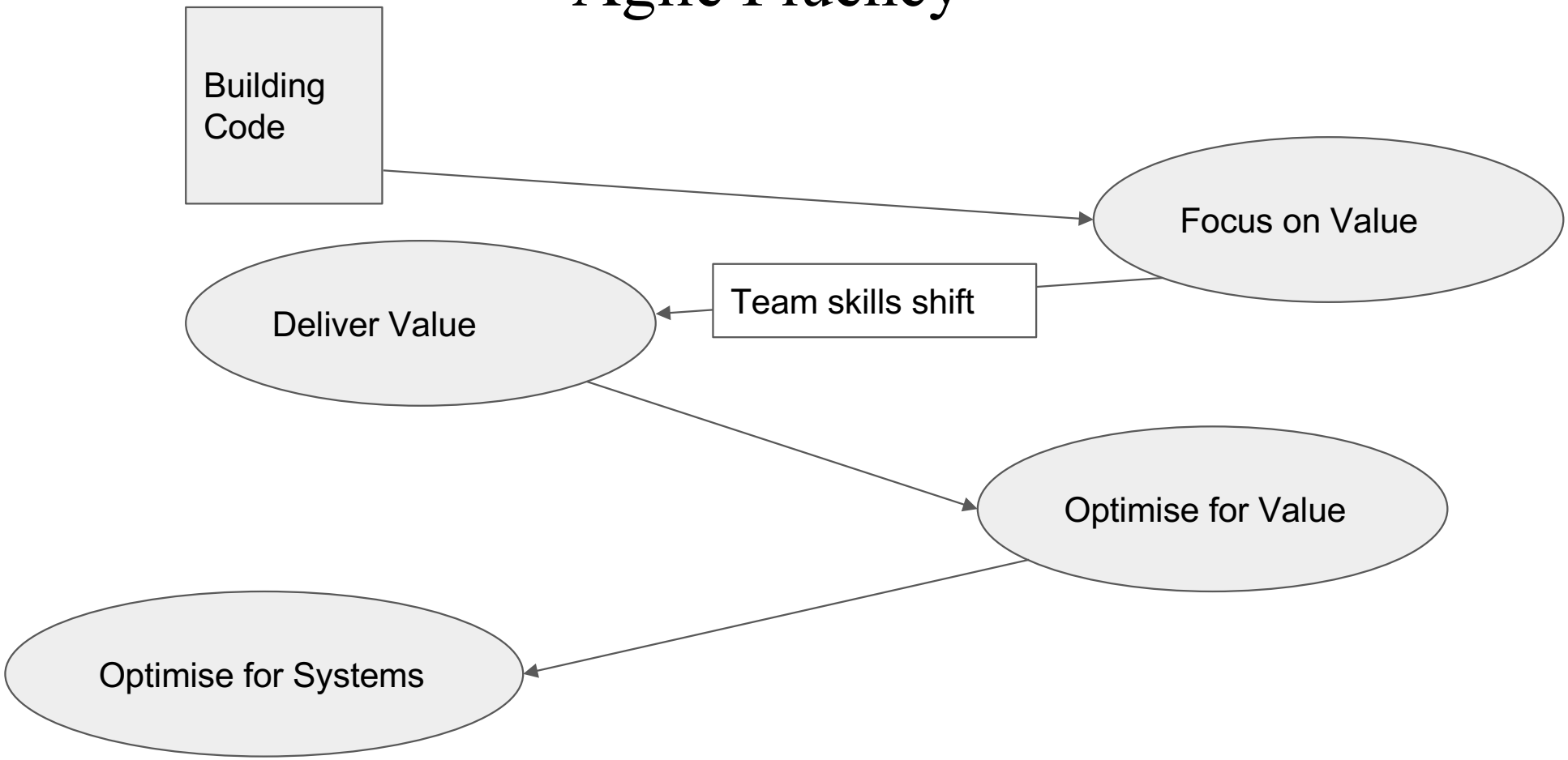
By Diana Larsen and James Shore

Agile Fluency



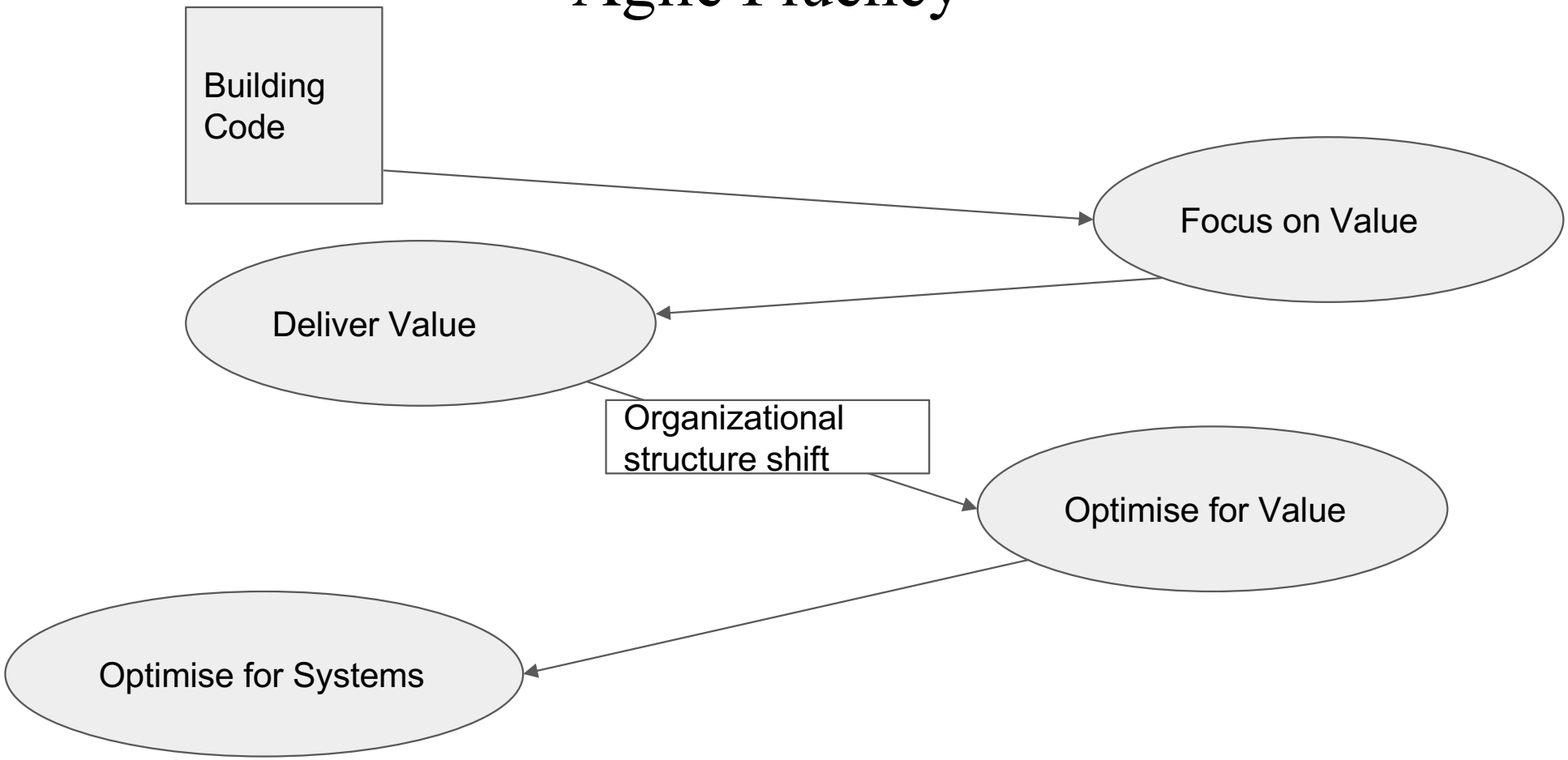
By Diana Larsen and James Shore

Agile Fluency



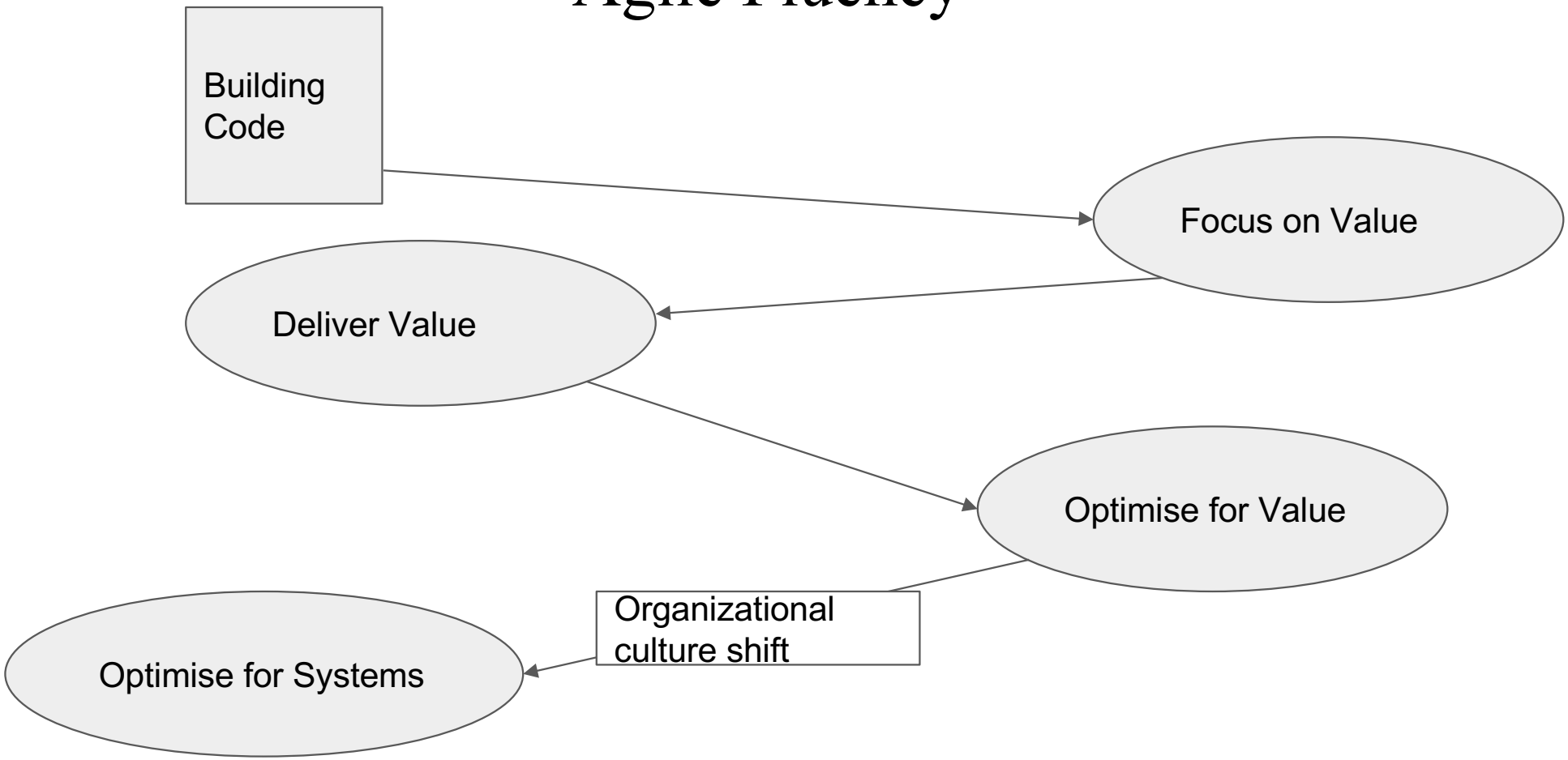
By Diana Larsen and James Shore

Agile Fluency



By Diana Larsen and James Shore

Agile Fluency



By Diana Larsen and James Shore

Happy Hobbit

The heart of being agile

Ability to have:

- Growth mindset
- Inspect and adapt
- Flexibility

The ability lies where?

In the processes?

In the culture?

Where else?

What do we need

Courage and possibility to change

What do we need

Courage and possibility to change:

Code standards

Pair Programming

Commit to trunk

Code reviews

Bug prevention (5 why’s)

Blameless Retrospectives

Culture

Courage and possibility to change:

Code standards

Pair Programming

Commit to trunk

Code reviews

Bug prevention (5 why's)

Blameless Retrospectives

Technical Agile

Courage and **possibility** to change:

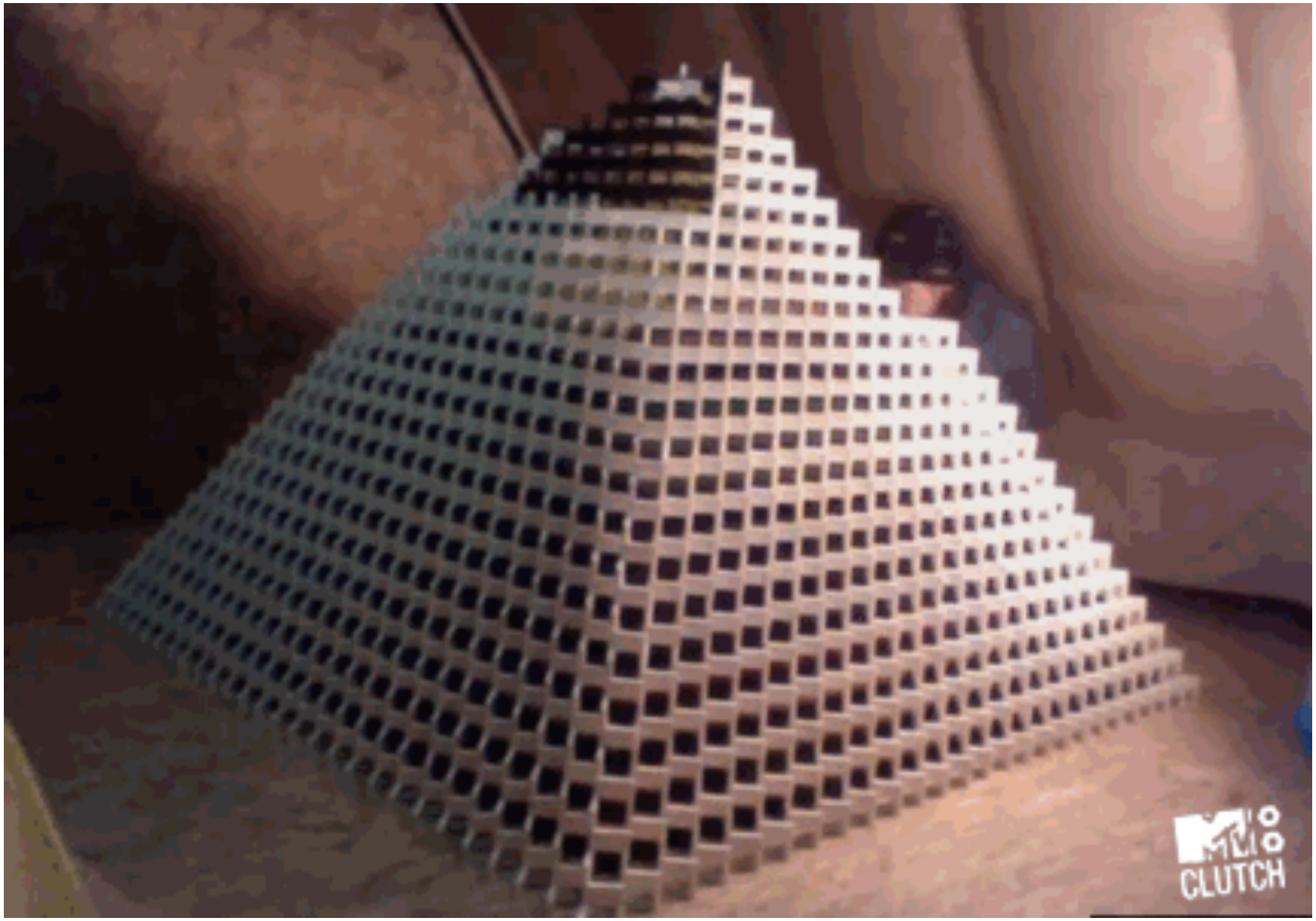
Version Control

Test Suite

Automated Test Suite

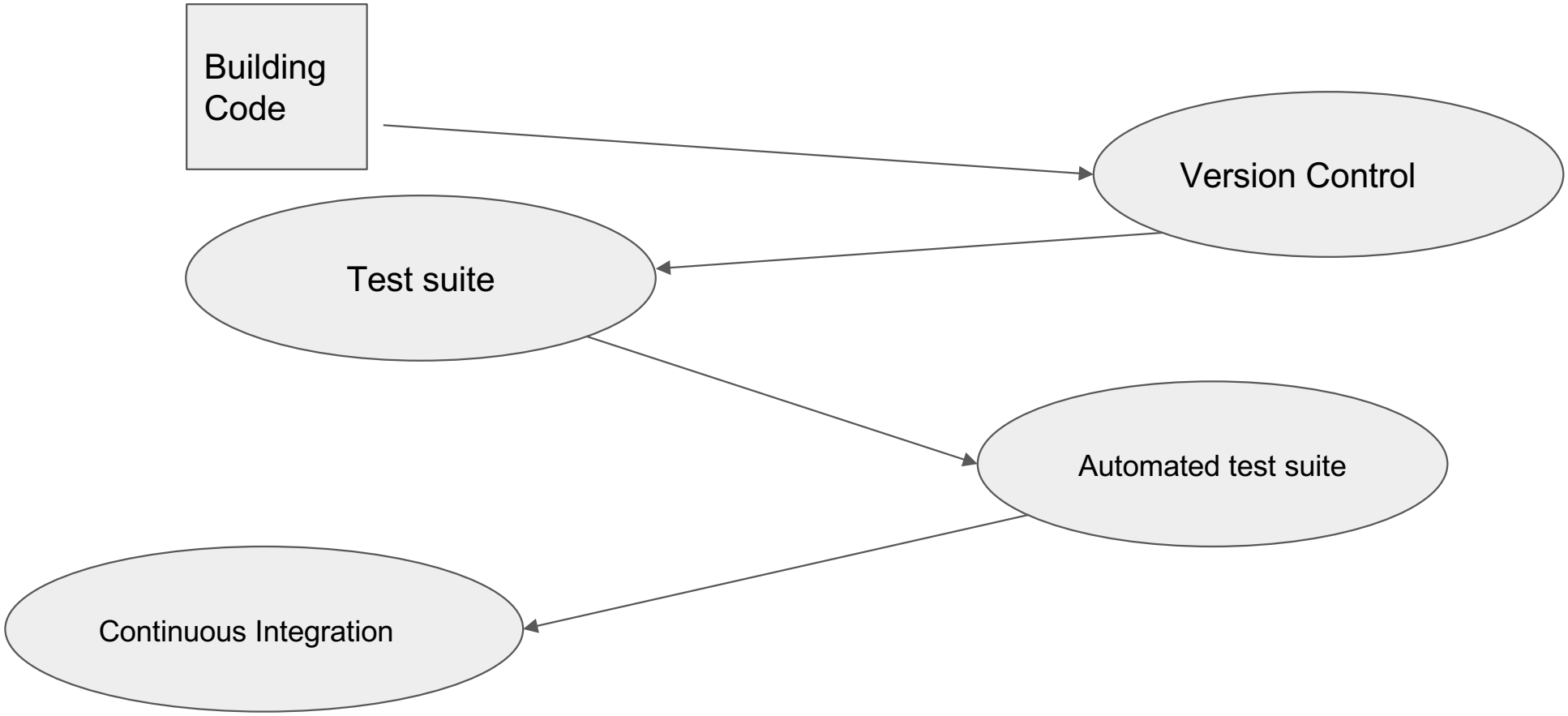
Continuous Integration

Monitoring operations


CLUTCH

Stupid developers, be more agile!

Technical Agile Fluency



What do we need ?

Courage and **possibility** to change

What we need: Culture **and** Technology

And you?

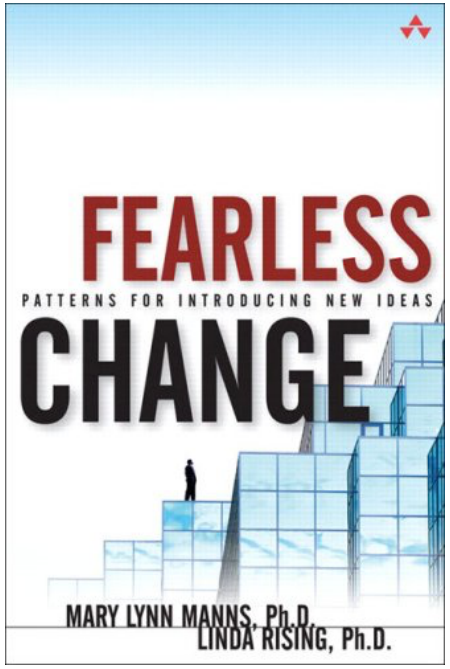
Reflect on, whether agile is possible where you are?

One minute challenge

Share with your neighbour

Listen to your neighbour

“Fearless Change” -Linda Rising and Mary Lynn Manns



Patterns I often use

Evangelist

Wake-up call

Brown Bag

Do Food

Shoulder to cry on

Patterns I use a bit

Just do it

Study group

Stay in touch

Champion skeptic

Patterns I often forget...

Test the waters

Time for reflection

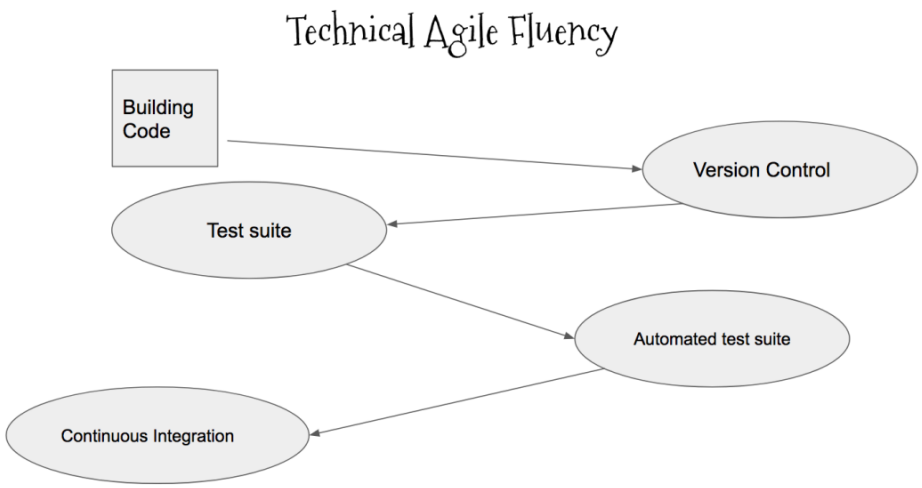
Big Jolt

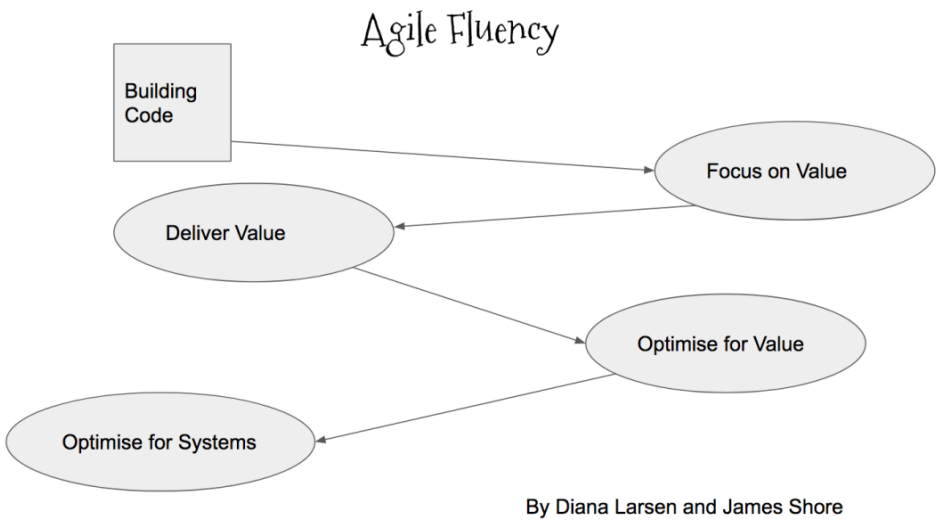
Small successes

Step by step

Fear less

Remember - agility is a journey





Thank you for your time

Agility is a journey

Always inspect and adapt - the process!

It is a continuous investment

Don’t be afraid - there is hope