Microservices: software that fits in your head

Dan North
@tastapod

What is the point of software development?

What is the purpose of software development?

What is the purpose of software development?

create business impact

What is the purpose of software development?

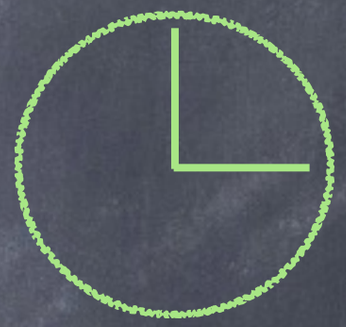
create positive business impact

What is the goal of software development?

What is the goal of software development?

minimise lead time

to business impact

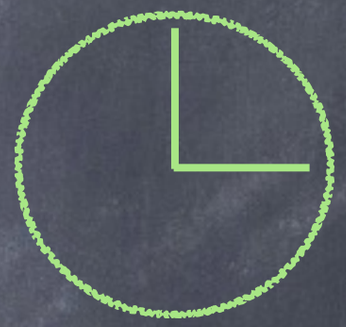


What is the goal of software development?

sustainably

minimise lead time

to business impact



The goal is not to produce software!

produce

produce

production

produce

productive

production

productive

$\neq$

effective

Code is not the asset...

Code is not the asset...

Code is the cost!

Code is not the asset...

- writing code costs

Code is the cost!

Code is not the asset...

- writing code costs
- waiting for code costs

Code is the cost!

Code is not the asset...

- writing code costs
- waiting for code costs
- changing code costs

Code is the cost!

Code is not the asset...

- writing code costs
- waiting for code costs
- changing code costs
- understanding code costs

Code is the cost!

understanding code

understanding code

code I know

understanding code

code I know

code everyone knows

understanding code

code I know

code no-one knows!

code everyone knows

Code should be

stabilised

or

killed off

Code should be

stabilised

or

killed off

fast!

Two complementary patterns

Two complementary patterns

Short Software Half-Life

Two complementary patterns

Short Software Half-Life

Fits In My Head

Short Software Half-Life

Short Software Half-Life

Design considerations

Short Software Half-Life

Design considerations

- write discrete components

Short Software Half-Life

Design considerations

- write discrete components
- define component boundary

Short Software Half-Life

Design considerations

- write discrete components
- define component boundary
- define component purpose

Short Software Half-Life

Design considerations

- write discrete components
- define component boundary
- define component purpose
- define component responsibility

Short Software Half-Life

Stewardship considerations

Short Software Half-Life

Stewardship considerations

- write component tests and docs

Short Software Half-Life

Stewardship considerations

- write component tests and docs
- optimise for replaceability

Short Software Half-Life

Stewardship considerations

- write component tests and docs
- optimise for replaceability
- expect to invest in stabilising

Short Software Half-Life

Stewardship considerations

- write component tests and docs
- optimise for replaceability
- expect to invest in stabilising
- build a stable team

Fits In My Head

Fits In My Head

- multiple dimensions

Fits In My Head

- multiple dimensions
- multiple scales

Fits In My Head

- multiple dimensions
- multiple scales
- "What would James do?"

Fits In My Head

- multiple dimensions
- multiple scales
- "What would James do?"

Contextual Consistency

Fits In My Head

- multiple dimensions
- multiple scales
- "What would James do?"

Contextual Consistency

- agree guiding principles

Fits In My Head

- multiple dimensions
- multiple scales
- "What would James do?"

Contextual Consistency

- agree guiding principles
- agree idioms

Fits In My Head

- multiple dimensions
- multiple scales
- "What would James do?"

Contextual Consistency

- agree guiding principles
- agree idioms
- difference is data

Fits In My Head

- multiple dimensions
- multiple scales
- "What would James do?"

Contextual Consistency

- agree guiding principles
- agree idioms
- difference is data
- familiarity ≠ simplicity

Replaceable Component Architecture

Replaceable Component Architecture

sustainably...

Replaceable Component Architecture

sustainably...

Replaceable Component Architecture

sustainably...

Replaceable Component Architecture

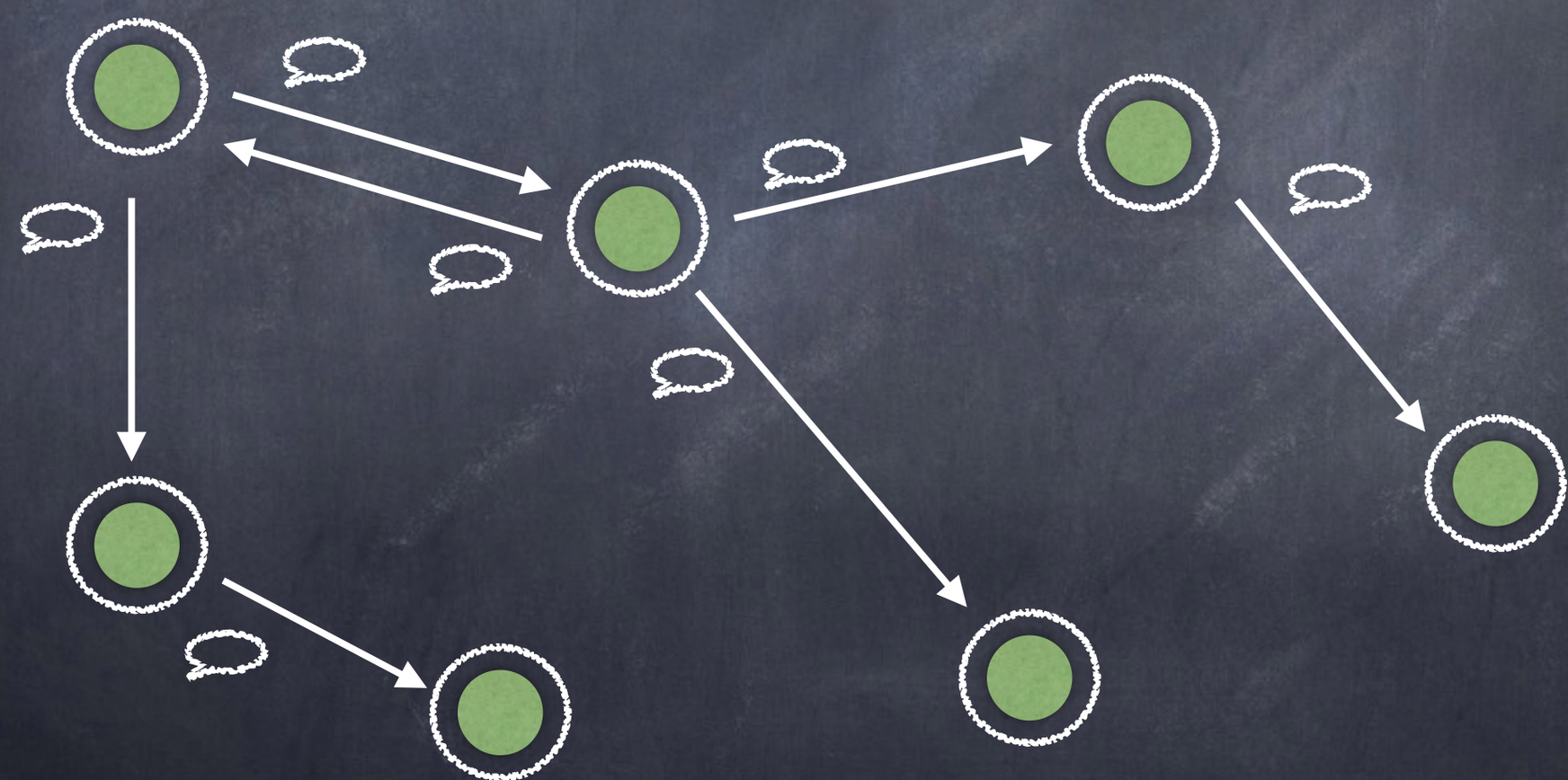
sustainably...

Replaceable Component Architecture

sustainably...

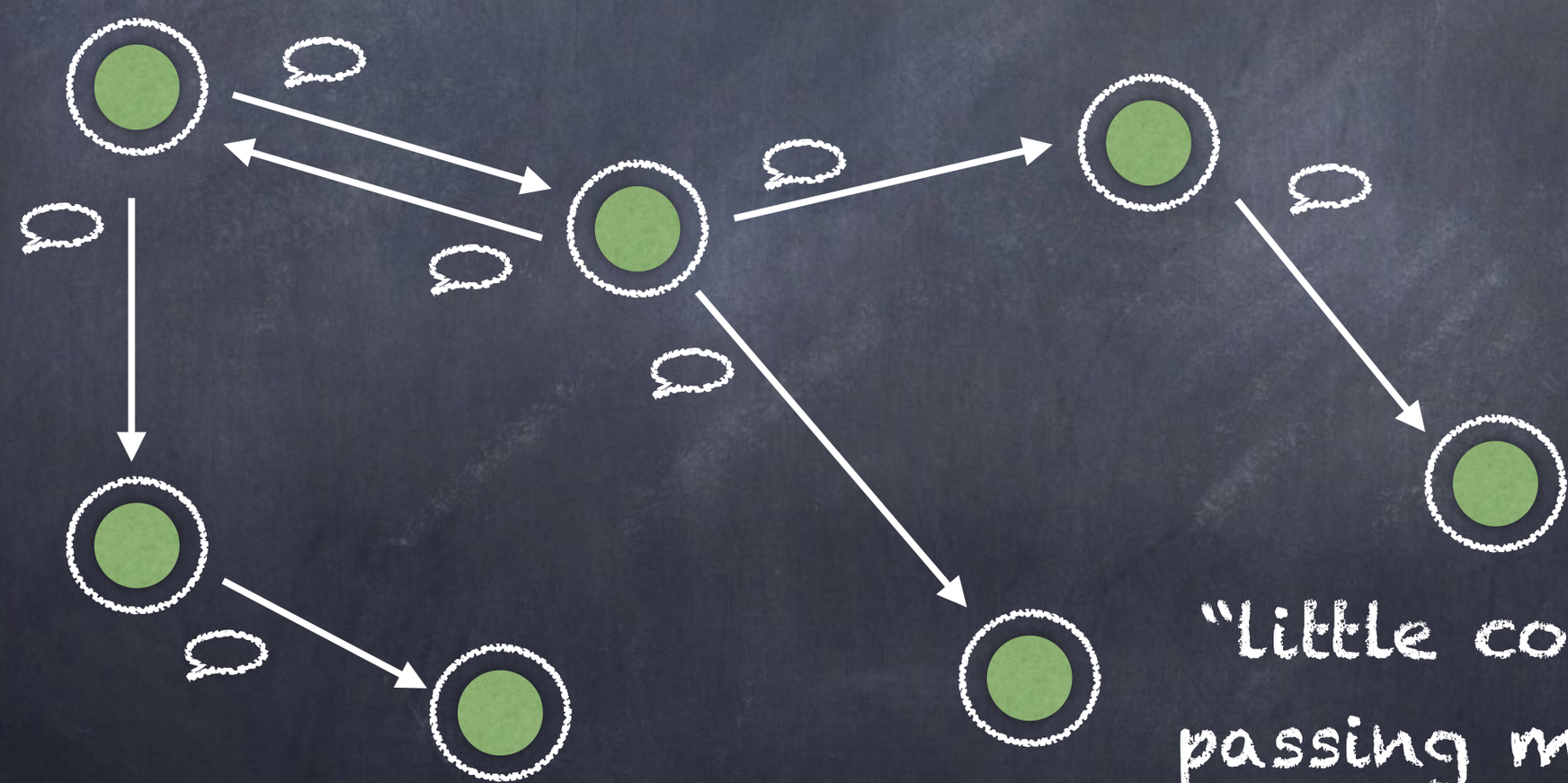
Replaceable Component Architecture

sustainably...



Replaceable Component Architecture

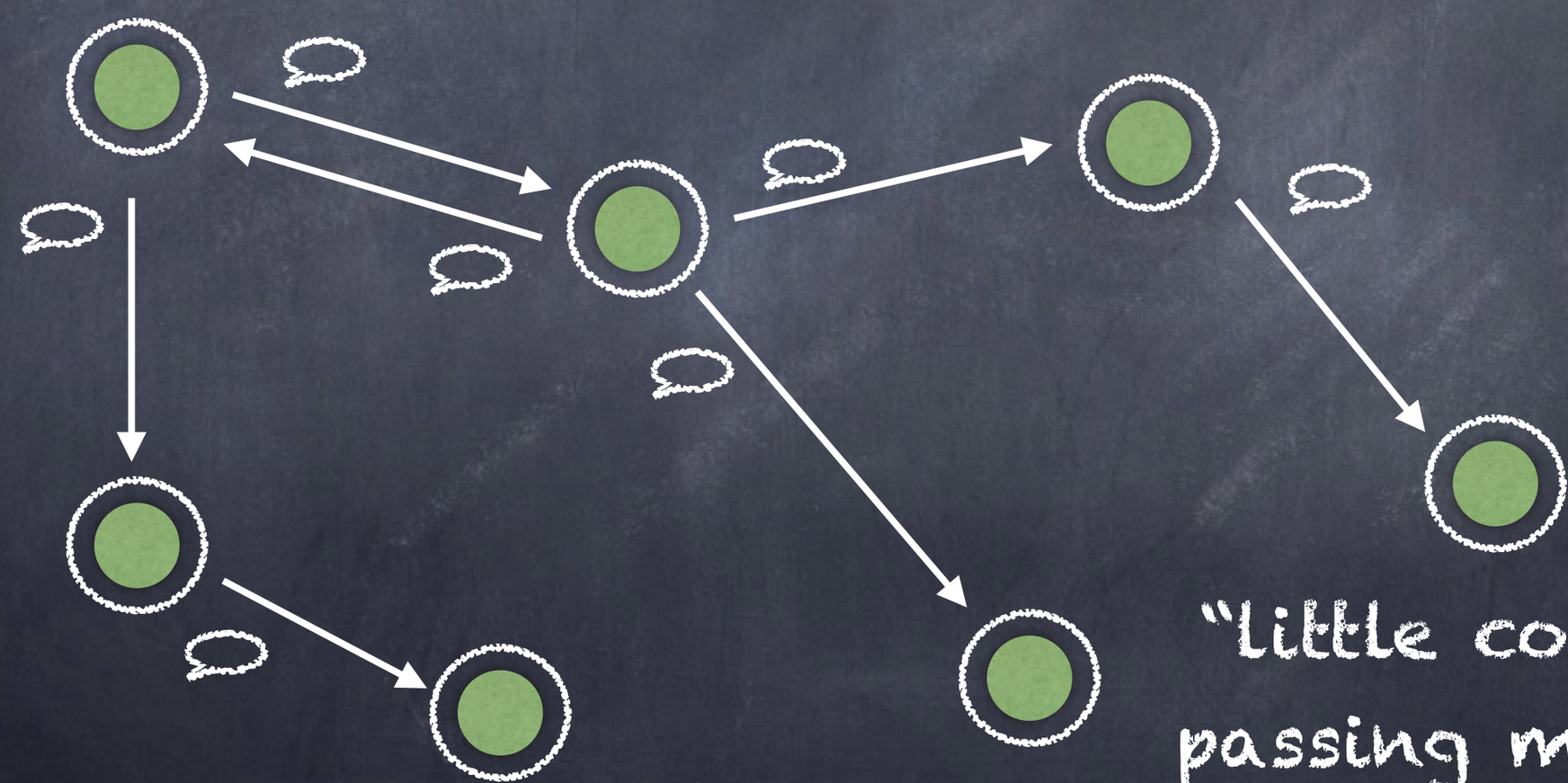
sustainably...



"little computers passing messages"

Replaceable Component Architecture

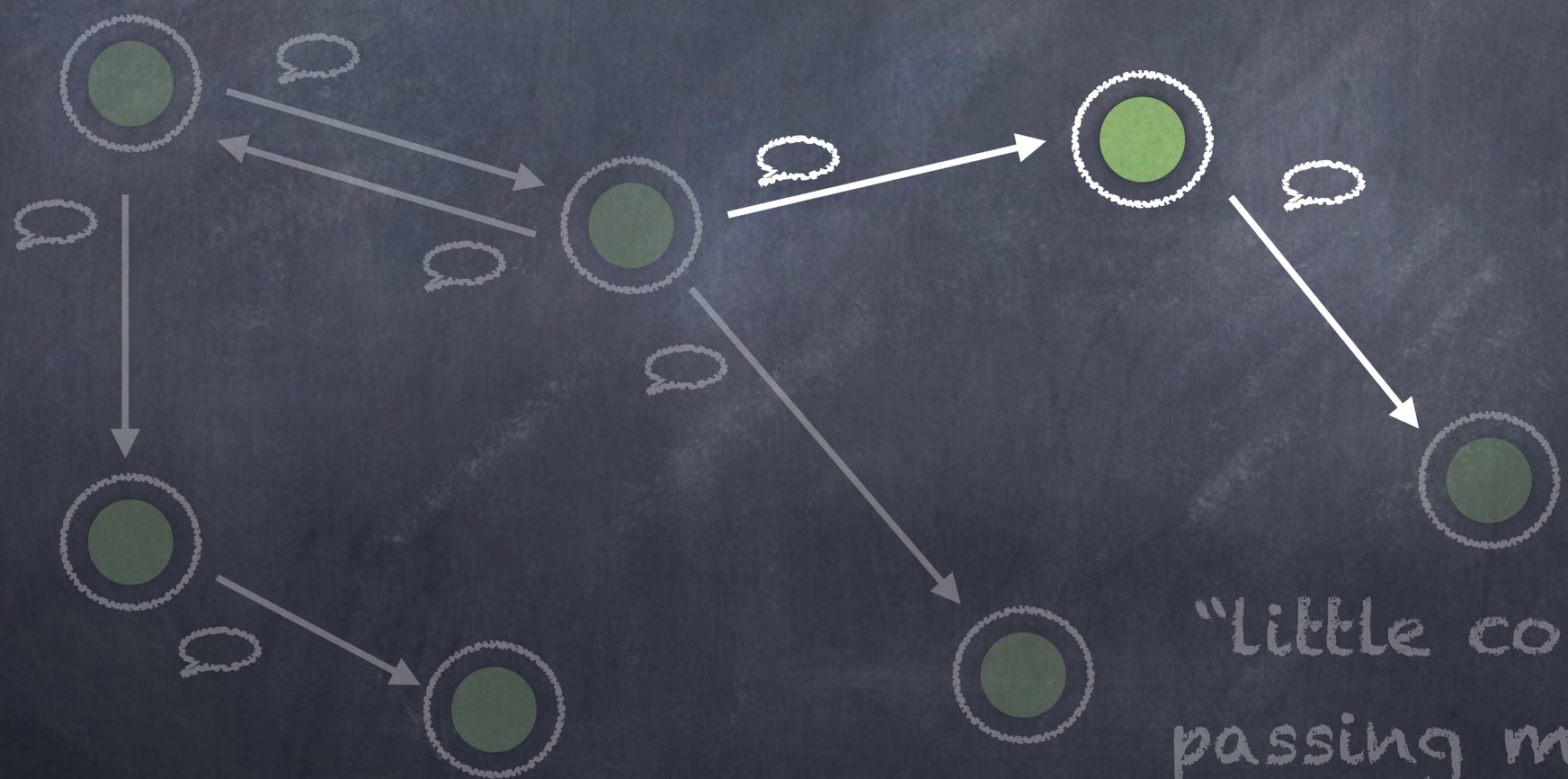
sustainably...



"little computers passing messages"
– Alan Kay

Replaceable Component Architecture

sustainably...



"Little computers passing messages"
— Alan Kay

Microservices can be a Replaceable Component Architecture

Microservices can be a Replaceable Component Architecture

- if you choose to optimise for replaceability and consistency

Microservices can be a Replaceable Component Architecture

- if you choose to optimise for replaceability and consistency
- smaller is not necessarily better

Microservices can be a Replaceable Component Architecture

- if you choose to optimise for replaceability and consistency
- smaller is not necessarily better
- more replaceable is better

Kill code fearlessly!

Kill code fearlessly!

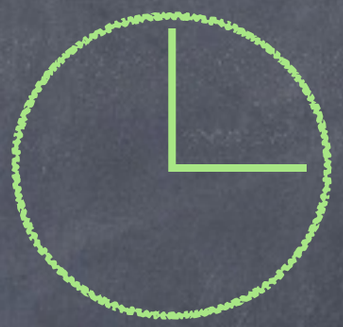
code I know

code no-one knows!

code everyone knows

Kill code fearlessly!

code I know



code everyone knows

Thank you!

Dan North

@tastapod

http://dannorth.net

http://leanpub.com/software-faster