Software Architecture as a Discipline

PV260 Software Quality

Ondřej “Ondra” Krajíček
ondrej.krajicek@ysoft.com
@OndrejKrajicek

What is **Software Architecture**… and **why** should you care?

PV260 Software Quality

Ondřej “Ondra” Krajíček
ondrej.krajicek@ysoft.com
@OndrejKrajicek

What **do you believe** Software Architecture is?

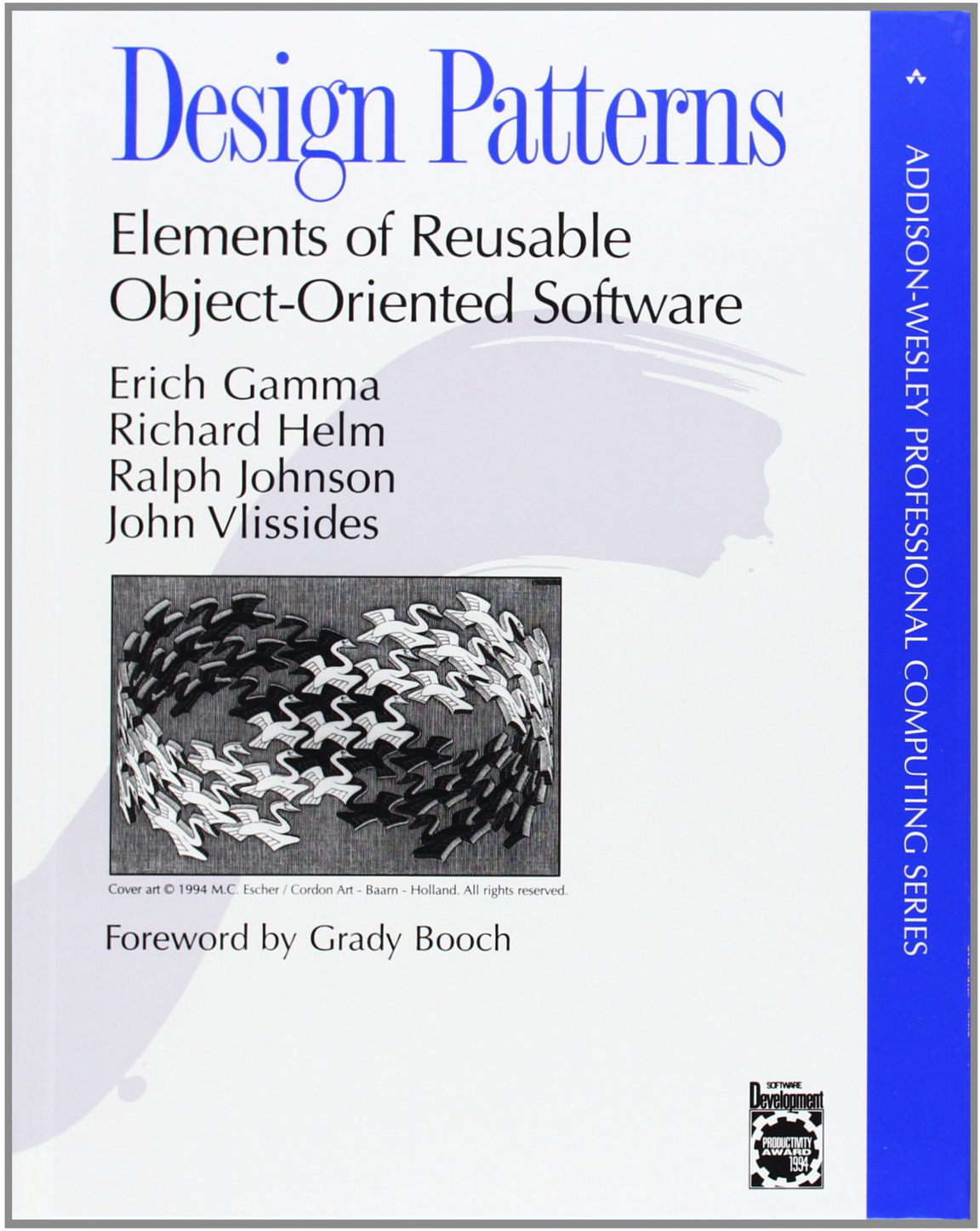
...why do you care?

Why should you listen or (moreover) pay attention?

...and why am I here today?

Software Architecture is the important stuff (whatever that is).

Ralph Johnson



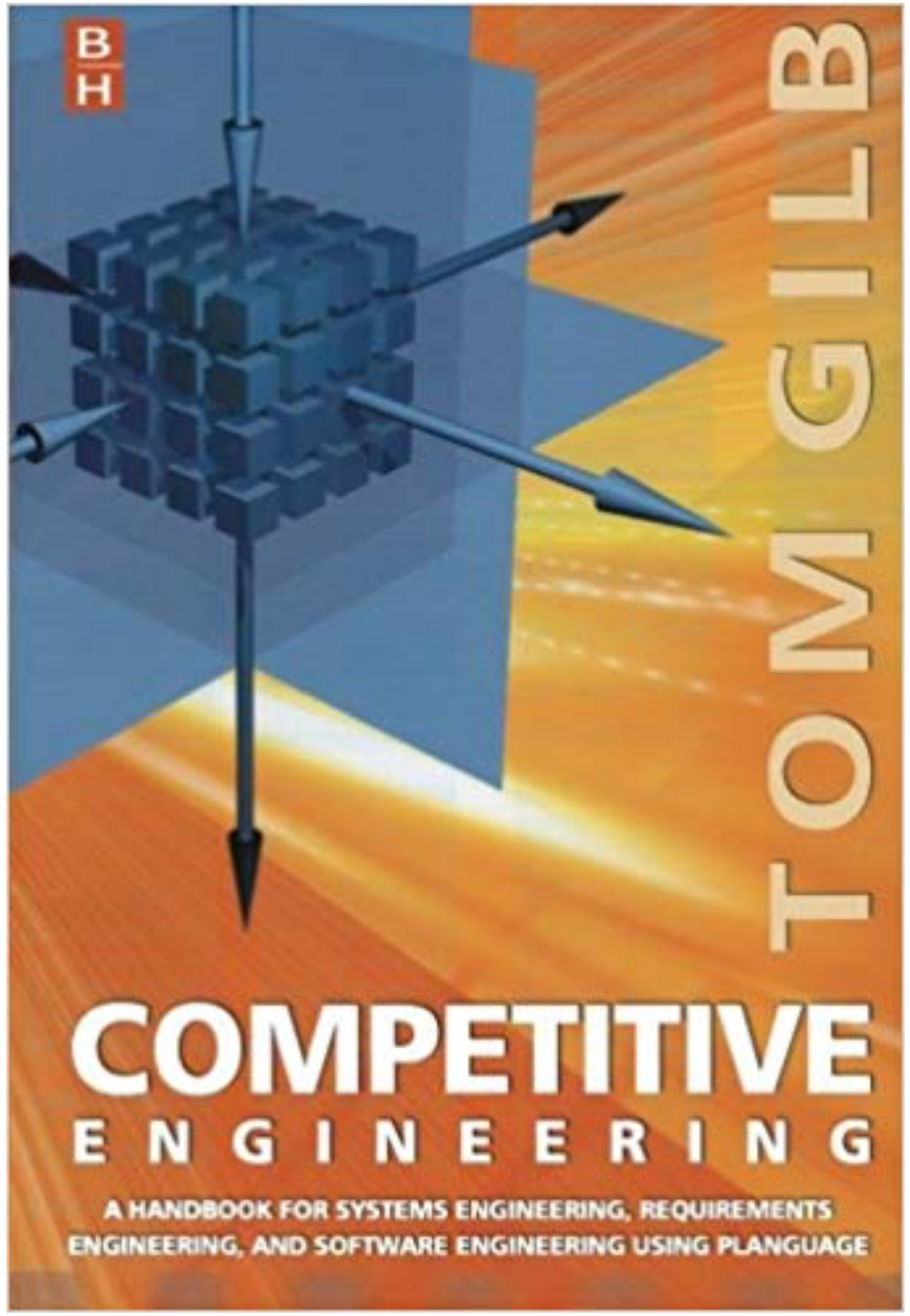
Which stuff is important?

…and how can we identify the important stuff?

SOFTWARE ARCHITECTURE **IS THE SERVANT OF HIGH-PRIORITY STAKEHOLDER VALUES**.

IS AS SIMPLE AS POSSIBLE, BUT NOT SIMPLER AND IS DESIGNED TO BE REPLACEABLE.

Tom Gilb (Architecture Manifesto)



SERVANT

HIGH-PRIORITY

STAKEHOLDER

VALUES

AS SIMPLE AS POSSIBLE

NOT SIMPLER

REPLACEABLE

SERVANT

HIGH-PRIORITY

STAKEHOLDER

VALUES

AS SIMPLE AS POSSIBLE

NOT SIMPLER

REPLACEABLE

SERVANT

HIGH-PRIORITY

STAKEHOLDER

VALUES

AS SIMPLE AS POSSIBLE

NOT SIMPLER

REPLACEABLE

Software Architecture is **Strategy**

Strategy is a plan how to deliver on your goals. Your goals are defined by high-priority stakeholders.

SOFTWARE ARCHITECTURE **IS THE STRATEGY HOW TO DELIVER HIGH-PRIORITY STAKEHOLDER VALUES**.

SOFTWARE ARCHITECTURE **IS THE STRATEGY HOW TO DELIVER HIGH-PRIORITY STAKEHOLDER VALUES**.

It is still important to **accept change** because no battle plan survives the first contact with the enemy.

Software Architecture is **Risk Mitigation**

What happens when things go wrong? Is it important?

SERVANT

HIGH-PRIORITY

STAKEHOLDER

VALUES

AS SIMPLE AS POSSIBLE

NOT SIMPLER

REPLACEABLE

Software Architecture is **Communication**

What does it mean? How to deliver and protect stakeholder values?

- Constraints

- Constraints
- Rules

- Constraints
- Rules
- Technical Decisions and Decision Logs

- Constraints
- Rules
- Technical Decisions and Decision Logs
- Architecture Models

- Constraints
- Rules
- Technical Decisions and Decision Logs
- Architecture Models
- Design

- Constraints
- Rules
- Technical Decisions and Decision Logs
- Architecture Models
- Design
- Specifications

- Constraints
- Rules
- Technical Decisions and Decision Logs
- Architecture Models
- Design
- Specifications
- Architectural Patterns

- Constraints
- Rules
- Technical Decisions and Decision Logs
- Architecture Models
- Design
- Specifications
- Architectural Patterns
- ?

4 Rules of Simple Design

Software Architect is a **Teacher**

Principles over Patterns

Accountability Problem

Why “traditional" approach fails.

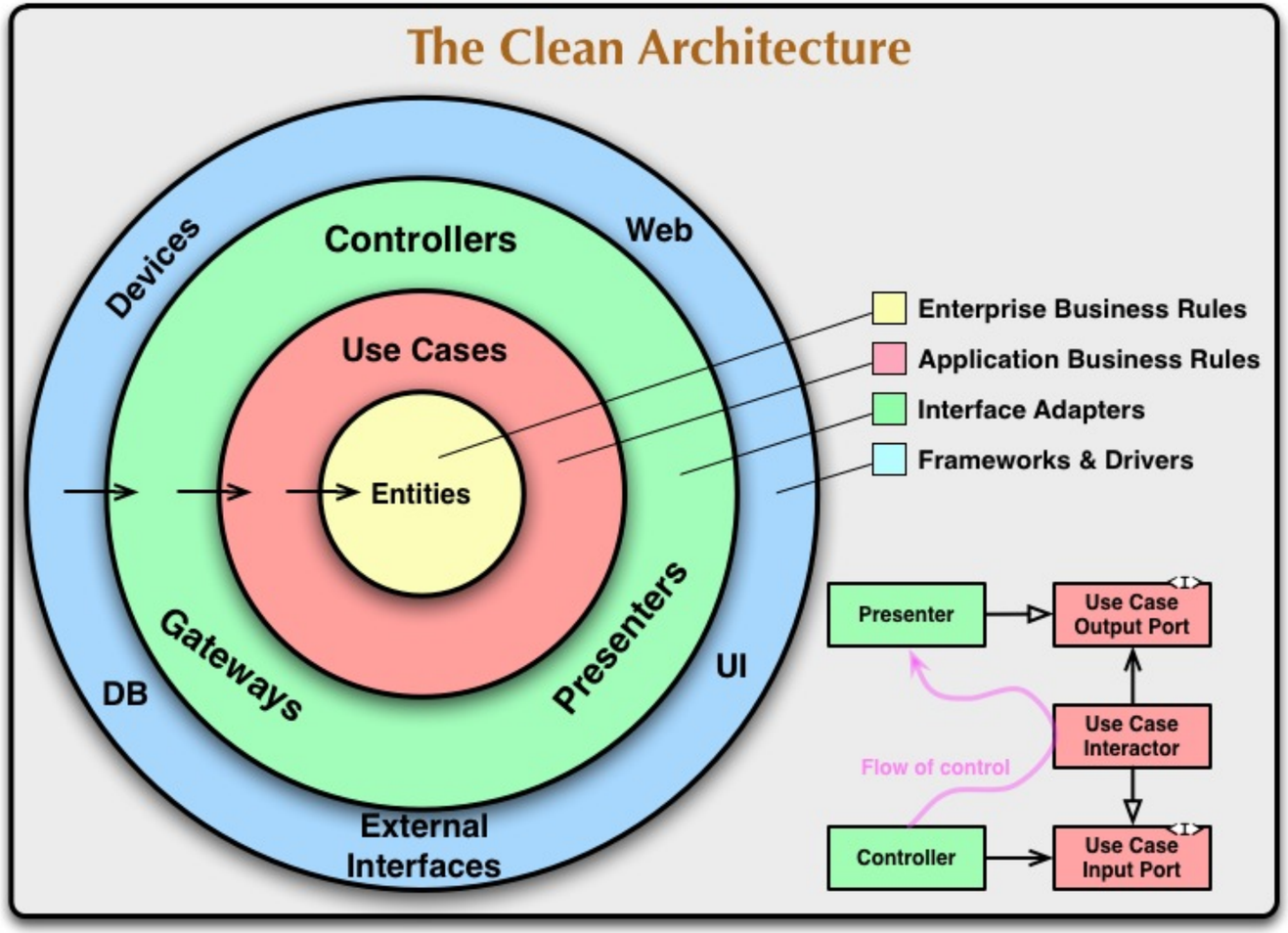
Who **does** software architecture then?

Who **does** software architecture then?

Software Architecture is done by **everyone**.

Principles of Good Architecture

- Independent of Frameworks.
- Testable: all parts and as a whole.
- Independent of UI.
- Independent of the data store / database / object persistence.
- Independent of any external impact.



So what does the architecture of your application *scream*?

When you look at the top level directory structure, and the source files in the highest level package; do they scream: **Health Care System**, or **Accounting System**, or **Inventory Management System**?

Or do they scream: **Rails**, or **Spring/Hibernate**, or **ASP**?

SOLID Principles

SERVANT

HIGH-PRIORITY

STAKEHOLDER

VALUES

AS SIMPLE AS POSSIBLE

NOT SIMPLER

REPLACEABLE

“Perfection is not achieved when there is nothing to add, but when there is nothing to remove.”

Antoine de Saint Exupéry

“Insanity is doing the same thing over and over and expecting different results.”

(attributed to) Albert Einstein

“It has to work.”

IETF RFC 1925

Measurable Quality

How do the changes you deliver map to your stakeholder values / qualities.

***Tag**: Ease of Access.*
***Version**: 11-Aug-2003.*
***Owner**: Rating Model Project (Bill).*
***Scale**: Speed for a defined [Employee Type] with defined [Experience] to get a defined [Client Type] operating successfully from the moment of a decision to use the application.*
***Alternative Scales**: None known yet.*
***Qualifier Definitions**:*
** Employee Type: {Credit Analyst, Investment Banker, …}.*
** Experience: {Never, Occasional, Frequent, Recent}.*
** Client Type: {Major, Frequent, Minor, Infrequent}.*
***Meter Options**:*
** Test all frequent combinations of qualifiers at least twice. Measure speed for the combinations.*
***Known Usage**: Project Capital Investment Proposals [2001, London].*
***Known Problems**: None recorded yet.*
***Limitations**: None recorded yet.*

The moral: introducing changes to a highly dynamic (eco) system can yield unpredictable results.

Neal Ford, Building Evolutionary Architectures

How to protect our system from dynamic / complex changes?

Protect your system with fitness functions.

YAGNI - You Ain’t Gonna Need It

Forget YAGNI. **You ain’t gonna need it.**

SERVANT

HIGH-PRIORITY

STAKEHOLDER

VALUES

AS SIMPLE AS POSSIBLE

NOT SIMPLER

REPLACEABLE

Osaka Castle

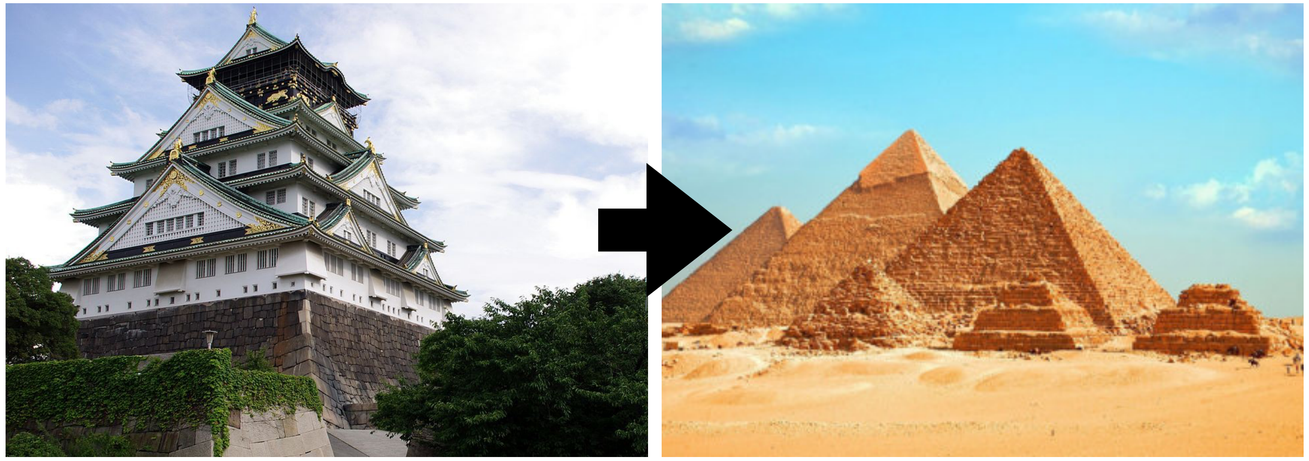
Giza Pyramids

- Built to last: hundreds and thousands of years.

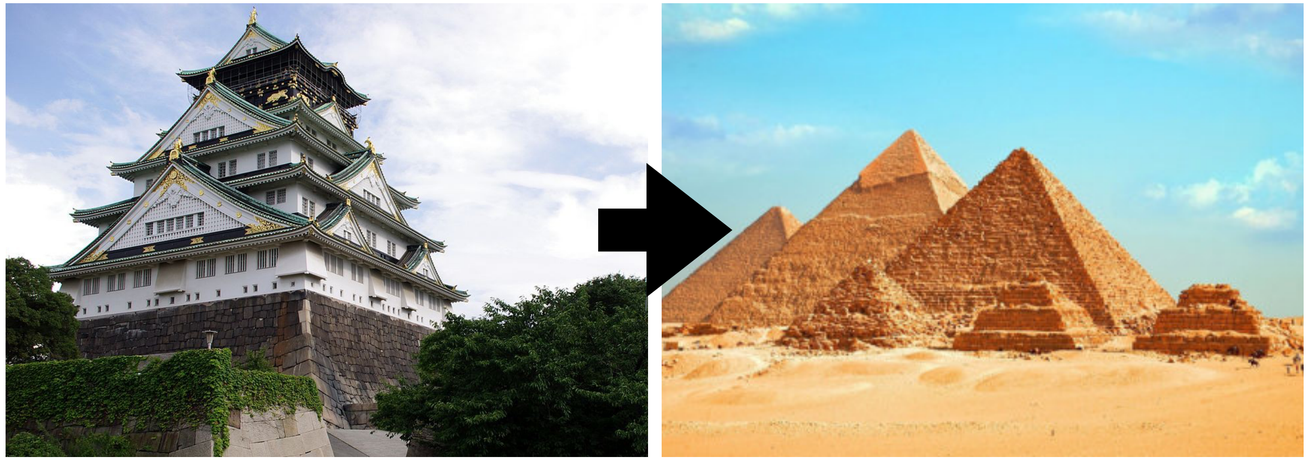
- Built to last: hundreds and thousands of years.
- Built to survive natural disasters, especially earthquakes (shinbashira).

- Built to last: hundreds and thousands of years.
- Built to survive natural disasters, especially earthquakes (shinbashira).
- Both have very different architecture.

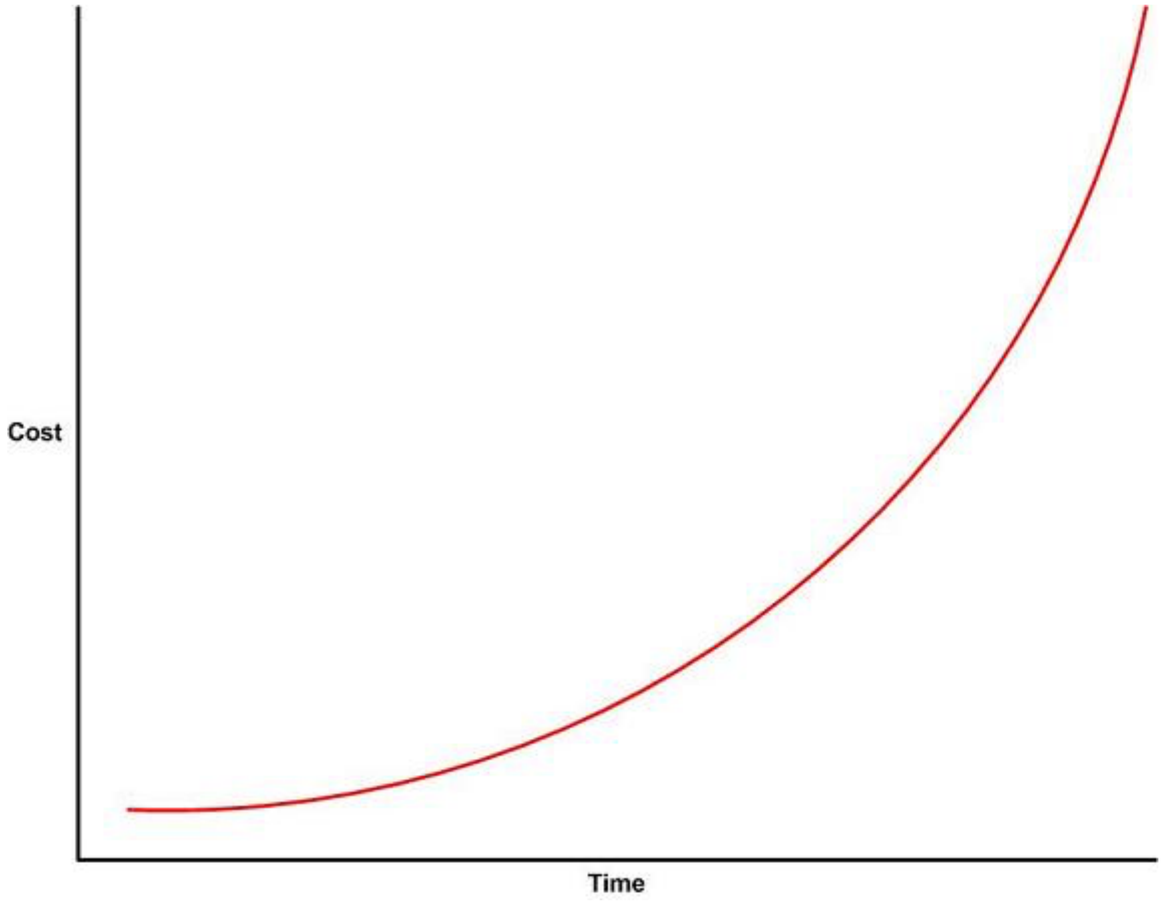
- Built to last: hundreds and thousands of years.
- Built to survive natural disasters, especially earthquakes (shinbashira).
- Both have very different architecture.
- You cannot replace one with the other.

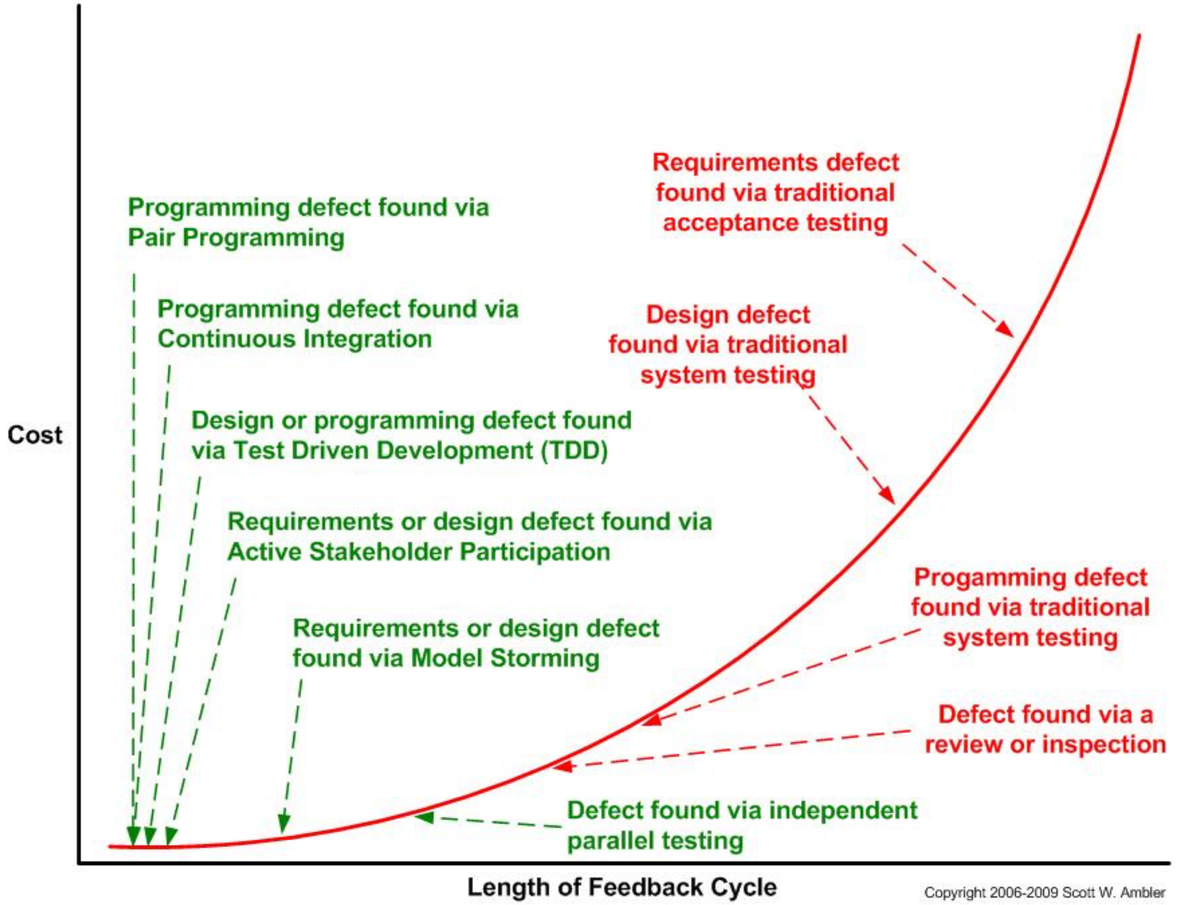


- Built to last: hundreds and thousands of years.
- Built to survive natural disasters, especially earthquakes (shinbashira).
- Both have very different architecture.
- You cannot replace one with the other.
- Why would you?



Cost of Change Curve





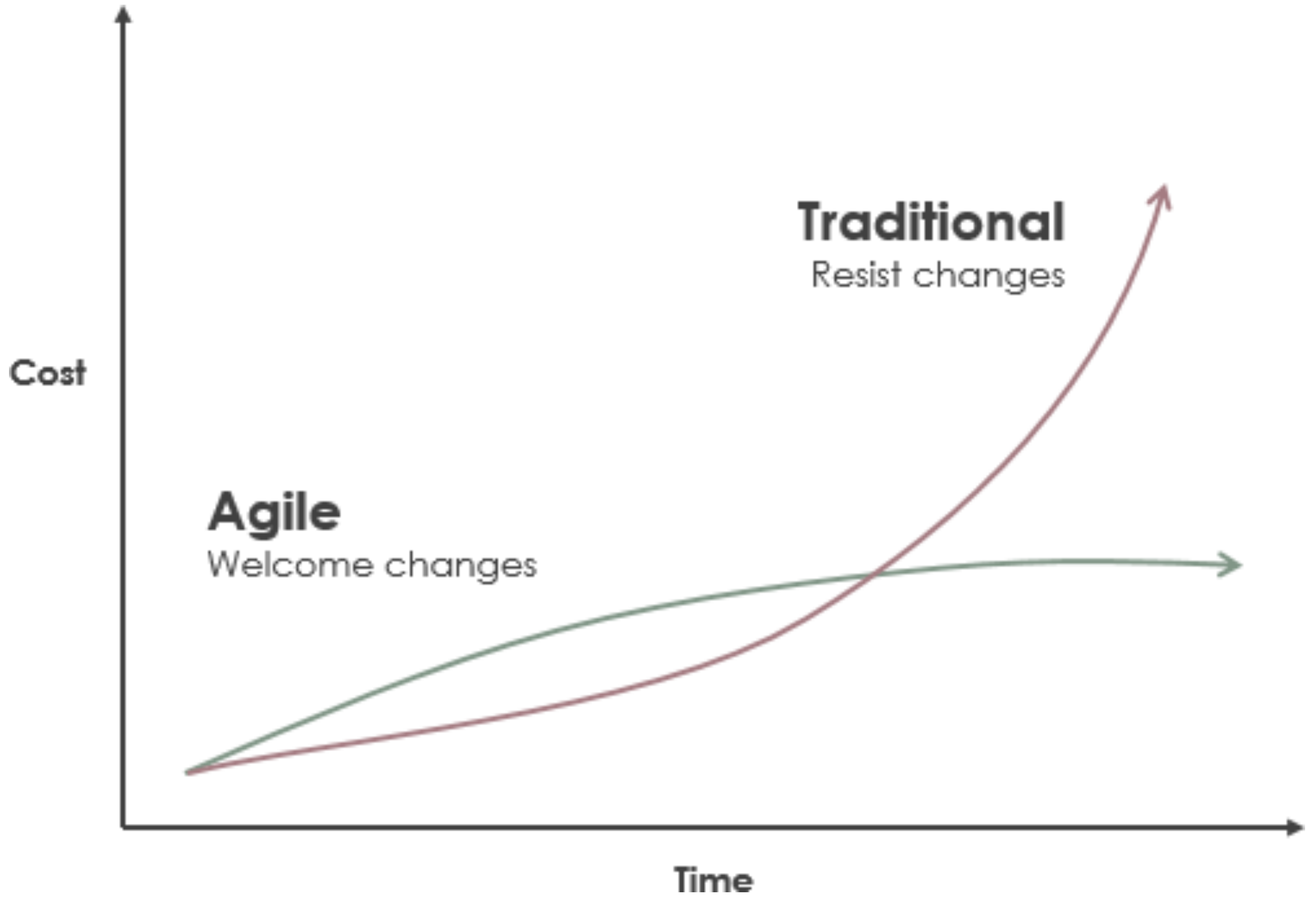
Cost of Change Curve

Replaceable as in ***Having rather low Cost of Change***

How to decrease Cost of Change?

Agile Frameworks

- How to build architecture incrementally?
- How to build architecture in iterations?



These two questions will be discussed in our seminar next week: **April 23rd, 14:00 - 18:00**.

And there will be (free) pizza and (free) architectural kata. Not for the faint of heart, because this time, we will have **a real discussion**.