Architecture.Next: Invalidating Old Axioms

Mark Richards
Independent Consultant
Hands-on Software Architect
Published Author | Conference Speaker
ax·iom

a statement or proposition which is regarded as being established, accepted, or self-evidently true
it is possible to draw a straight line from any point to any other point.
it is possible to describe a circle with any center and any radius
(axiom of extensionality)
given any set $A$ and any set $B$, if for every set $X$, $X$ is a member of $A$ if and only if $X$ is a member of $B$, then $A$ is equal to $B$.

\[ \forall A \forall B (\forall X (X \in A \iff X \in B) \implies A = B) \]
software architecture is a separate activity from software development
software architects should adopt and follow best practices in software architecture
software architecture is the stuff that’s hard to change later
software architecture is a separate activity from software development
software architecture

availability
scalability
performance

software development
availability
scarcity
performance

software architecture
software development

leadership

mentoring
“Developers should never take components designed by architects as the last word. Rather, the initial design should be viewed as a first draft, where implementation will reveal more details and refinements.”
software architects should adopt and follow best practices in software architecture
First Law of Software Architecture

“Everything in software architecture is a trade-off”

there are no best practices!
add a new feature to collect, maintain, and redeem reward points
add a new feature to collect, maintain, and redeem reward points
add a new feature to collect, maintain, and redeem reward points
add the ability to apply multiple payment types to pay for an order
add the ability to apply multiple payment types to pay for an order
add the ability to apply multiple payment types to pay for an order
add the ability to apply multiple payment types to pay for an order
add the ability to apply multiple payment types to pay for an order
there are no best practices in software architecture - only trade-offs
software architecture is the foundational structure of a system and therefore should not undergo frequent change

REJECTED
software architecture is the stuff that’s hard to change later
continually refactoring the architecture means you don’t know what you are doing
architecture

business drivers
- innovation
- mergers and acquisitions
- consumer demand
- competition

environment
- CI/CD pipelines
- devops
- cloud infrastructure
- containerization

architecture
- structure
- communication
- dependencies

- asynchronous
- synchronous
how do software architects handle all of this change?

detect change

plan for change

facilitate change
detect change

collaborating with business stakeholders

collaborating with operations stakeholders

continually analyzing architecture characteristics
how do software architects handle all of this change?

detect change

plan for change

facilitate change
plan for change
how do software architects handle all of this change?

detect change  plan for change  facilitate change
facilitate change

business stakeholders

software architect

development team

user stories

architectural stories

technical debt stories

iteration
facilitate change

business stakeholders —— software architect —— development team

user stories

architecture stories

technical debt stories

iteration
how do software architects handle all of this change?

detect change

plan for change

facilitate change
O’Reilly software architecture keynotes are only 20 minutes long
Architecture.Next: Invalidating Old Axioms

Mark Richards
Independent Consultant
Hands-on Software Architect
Published Author | Conference Speaker