AGILE ARCHITECTURE – IDEALS, HISTORY, AND A NEW HOPE

Gary Pedretti
Owner, Sodoto Solutions
SODOTO = See One, Do One, Teach One
Trainer, Coach, Consultant
Professional Scrum Trainer, Scrum.org
Over sixteen years in the software industry as a generalizing specialist: DBA, coder, BA, architect, tester, manager
GaryPedretti.com
@GaryPedretti
http://www.linkedin.com/in/GaryPedretti
http://www.slideshare.net/earl128
SpeakerRate: http://TinyUrl.com/GarysSpeakerRate

GARY PEDRETTI
IDEALS & PRINCIPLES
MISUNDERSTANDINGS
BACKLASH
A NEW HOPE

Agenda
“...the highest level concept of a system in its environment. The architecture of a software system (at a given point in time) is its organization or structure of significant components interacting through interfaces, those components being composed of successively smaller components and interfaces.” – RUP, IEEE

“...shared understanding of the system design...architecture only includes the components and interfaces that are understood by all the developers...architecture is a social construct” – Fowler

“Stuff that’s hard [expensive] to change later.” – Neal Ford and others

BUT FIRST – WHAT IS “ARCHITECTURE”? 
“Architecture is about the important stuff. Whatever that is.” – Fowler

BUT FIRST – WHAT IS “ARCHITECTURE”? 
Neal Ford describes a “Framework-Level Architecture” – important because the unit of reuse in modern languages is the library or framework.

- Application Architecture = building architecture
- Enterprise Architecture = city planning
- Service-Oriented Architecture (SOA) = very interesting as a blend of Application and Enterprise Architectural concerns
A LONG TIME AGO IN A GALAXY FAR, FAR AWAY...

IDEALS & PRINCIPLES
LITTLE KNOWN FACTS ABOUT HAN SOLO, THE IDEALIST
BUDDIES
(BASED ON HAN’S PRINCIPLES AND IDEALS)

Gary@SodotoSolutions.com  @GaryPedretti
This is NOT your capital “A” Architects’ architecture!

AGILE (ARCHITECTURE) HAS IDEALS TOO
Ideals, Principles, Ideas => Techniques, Practices, Patterns

- Agile Manifesto and its Principles
- Kent Beck, Ward Cunningham, Ron Jeffries, et. al. from eXtreme Programming (XP)
- Martin Fowler
- Scott Ambler

AGILE ARCHITECTURE IDEALS, ETC. - SOURCES
- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

That is, **while there is value in the items on the right**, we value the items on the left more.

THE MANIFESTO
Welcome changing requirements, even late in development. Agile processes harness change for the customer’s competitive advantage.

Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

Working software is the primary measure of progress.

Continuous attention to technical excellence and good design enhances agility.

Simplicity - the art of maximizing the amount of work not done - is essential.

The best architectures, requirements, and designs emerge from self-organizing teams.

PRINCIPLES BEHIND THE MANIFESTO
“If you accept the premise that architecture is about things hard to change later, it becomes difficult to allow an architecture to emerge.” – Neal Ford

Jeff Sutherland and Rebecca Wirfs-Brock agree

Words matter – this is about communication… remember architecture is a social construct…

A WORD ON “EVOLUTIONARY” VS. “EMERGENT”
1. Test Everything (and run all the tests)
2. Eliminate Duplication (code)
3. Express All Ideas (that the author wants to express)
4. Minimize Entities (classes and methods)

XP – KENT BECK’S RULES FOR SIMPLE DESIGN
No Big Design Up Front (BDUF)
Replace with Evolutionary Design
Where Evolutionary Design != “code and fix” or Cowboy Hacking

EVOLUTIONARY DESIGN > BDUF
OPTIMISM IS AN OCCUPATIONAL HAZARD OF PROGRAMMING, FEEDBACK IS THE TREATMENT.

Kent Beck
Famous “Who Needs an Architect?” paper and “Is Design Dead?” blog post

Significant emphasis on the untenable split between design and build with software (a technique that can work in engineering, manufacturing, buildings, etc.), with XP practices

But in reality, encourages a healthy middle ground – doesn’t disown UML, patterns, design activities, etc.

Used Pramod Sadalage’s DB work as an example of the “unchangeable” becoming fluid
In his “Agile Modeling,” lays out specific
- Values
- Principles
- Practices

Builds off of core agile principles, and forges middle ground regarding UML, patterns, forethought, etc.

Worked with Pramod Sadalage on Database Refactoring

SCOTT AMBLER
MISUNDERSTANDINGS
GOING SOMEWHERE, SOLO?
HAN SHOT FIRST!!
MERCENARY, SWASHBUCKLER, EVERYMAN-FOR-HIMSELF...**SOLO**

Gary@SodotoSolutions.com   @GaryPedretti
AND SMUG, TOO!!
I'M SO MISUNDERSTOOD

Gary@SodotoSolutions.com  @GaryPedretti
BUDDIES
REMINDER: BASED ON HAN’S (SELFLESS) PRINCIPLES AND IDEALS
Agile Principles...

OH, SO MANY MISUNDERSTANDINGS
FIRST, LET’S IMAGINE THE IDEALS ABUSED...
• NO BDUF!!!

• (and skipping the “no Cowboy Hacking, either” part)

ABUSING XP PRINCIPLES
NO DOCUMENTATION NECESSARY!
NO PLANS!

(ignoring the “while there is value in the items on the right” part completely)

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

ABUSING THE MANIFESTO
CODERS ARE GODS NOW – ME ME ME – what’s motivation again? What’s a stakeholder?

WORKING SOFTWARE IS THE ONLY THING THAT MATTERS!

SIMPLE is best – ALWAYS!

Things emerge – I don’t need to think about them!!

ABUSING THE PRINCIPLES BEHIND THE MANIFESTO
"ONE OF THE MORE INSIDIOUS AND PERSISTENT MYTHS OF AGILE DEVELOPMENT IS THAT UP-FRONT ARCHITECTURE AND DESIGN ARE BAD; THAT YOU SHOULD NEVER SPEND TIME UP FRONT...DESIGN FROM NOTHING, ONE TEST-CASE AT A TIME.

PARDON ME, BUT THAT’S HORSE SHIT."

"Uncle Bob" Martin

WOW!

Gary@SodotoSolutions.com   @GaryPedretti
HAVE YOU BEEN HERE BEFORE???
(THE EMPIRE STRIKES BACK?)

THE BACKLASH
GUESS I’M ON MY OWN, HAN...
CAN YOU BLAME HIM?
WE’VE SEEN A LOT OF BACKLASH FROM ALL THOSE MISUNDERSTANDINGS...

- Organizational Snapback
- New Frameworks to “Address the issue of…”
- Broken Trust
▶ “Change Spasm” – Simon Rendl

▶ “Organizational Antibodies” – Michael Sahota

ORGANIZATIONAL SNAPBACK – THREE STEPS FORWARD, TWO STEPS BACK
THERE COMES A POINT AT WHICH EMERGENT DESIGN IS AN INSUFFICIENT RESPONSE TO THE COMPLEXITY OF LARGE-SCALE SYSTEM DEVELOPMENT.

Question: Is there a magical point at which waterfall, command/control, phase-gated, heavily plan-based methodologies suddenly just “start to work”??

From a newer framework
FOR THIS WE NEED SOME INTENTIONAL ARCHITECTURE—A SET OF PURPOSEFUL, PLANNED ARCHITECTURAL INITIATIVES TO ENHANCE SOLUTION DESIGN, PERFORMANCE AND USABILITY—and which provides guidance for inter-team design and implementation synchronization.

Read: Command and Control?? Funny, Mike Cohn used intentional at least back to 2009...

From a newer framework
IN THE CASES WHERE NEW PLATFORMS ARE PARTICULARLY INNOVATIVE, OR IN THE CASE OF ENTIRELY NEW (GREENFIELD) DEVELOPMENT, IT IS COMMON THAT SYSTEM ARCHITECTS PLAY A ROLE IN THE INITIAL DEFINITION AND BUILD OUT OF THE RUNWAY.

Read: Self-Organizing Teams Probably Can’t Be Trusted

From a newer framework
Is there a magical point at which waterfall, command/control, phase-gated, heavily plan-based methodologies suddenly just “start to work”? But...what about nature?

I HAVE A QUESTION
Can people be trusted to make decisions at the last responsible moment?

- Or will they typically conflate responsible with possible?
- Rebecca Wirfs-Brock said “yes” at Agile 2013 – granted, her blog backs this idea with one psychological study.
THE PROBLEM IS THAT EMPOWERED TEAMS ARE STILL HUMAN, THEY DO WHAT THEY ARE INCENTED TO DO. ARE THEY BEING REWARDED FOR QUALITY? OR ARE THEY BEING REWARDED FOR PRODUCTIVITY? HOW MUCH RECOGNITION IS THE TEAM GETTING FOR GOOD CODE QUALITY? HOW MUCH ARE THEY GETTING FOR DELIVERING WORKING FEATURES?

“Uncle Bob” Martin

CAREFUL OF THE INCENTIVES
CAN YOU BE TRUSTED?
A NEW HOPE...?
SO MISUNDERSTOOD...
WE HAVE A NEW HOPE TOO...BACK TO PRINCIPLES
I’ve already covered some typical misuses... including where they deviate and what to concentrate on...
Let’s look at some other things to get us back to principles…
TOOLS – FROM THE OBVIOUS TO THE LEAST-EXPECTED

Gary@SodotoSolutions.com  @GaryPedretti
Play the game with a cross-functional team

1. “Is it really the responsibility of this object to handle this request?”
2. “Is it its responsibility to keep track of all that information?”

“The best architectures, requirements, and designs emerge from self-organizing teams.”

CRC CARDS AND RESPONSIBILITY-BASED MODELING
Mike Brittain and Etsy’s Lean Startup experimentation approach: architecture is irrelevant until a feature is proven to be wanted by some pilot demographic, in which case you rebuild it in a sustainable fashion

Fowler coined the term “Sacrificial Architecture” in October 2014 to describe these throwaways – “often the best code you can write now is code you’ll discard in a couple of years time”

Safely enables YAGNI++

Akin to prototypes, Brooks’ “Build One to Throw Away”

“Simplicity - the art of maximizing the amount of work not done - is essential.”

SACRIFICIAL ARCHITECTURES
“Performance is a feature“ – Jeff Atwood

What does your product vision actually say about size of market, globalization, user appetite for risk, etc.?  
Realize any one answer is temporal…it will evolve

“Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.”

NON-FUNCTIONAL REQUIREMENTS ARE ACTUALLY FEATURES
- "Forward Engineering" – code generation from model
- "Reverse Engineering" – model extraction from code
- These approaches enable Ambler’s Agile Modeling Practices of:
  - Single Source of Information
  - Discard Temporary Models
  - Iterate to Another Artifact
  - Prove it with Code
  - Update Only When it Hurts
- “Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.”

UNIFIED MODELING LANGUAGE (UML) AND MODEL-DRIVEN ARCHITECTURE (MDA)
EVEN MORE TOOLS TO DIG INTO
▪ Communities of Practice
▪ Lunch and Learns
▪ Pairing and Swarming
▪ Remember that quote about Architecture fundamentally being a **social** construct?

▪ “Continuous attention to technical excellence and good design enhances agility.”

MENTORSHIP
“Business people and developers must work together daily throughout the project.”

COMMUNICATION AND HAVING DIFFICULT CONVERSATIONS
SOFTWARE IS NOT LIMITED BY PHYSICS, LIKE BUILDINGS ARE. IT IS LIMITED BY IMAGINATION, BY DESIGN, BY ORGANIZATION. IN SHORT, IT IS LIMITED BY PROPERTIES OF PEOPLE, NOT BY PROPERTIES OF THE WORLD. “WE HAVE MET THE ENEMY, AND HE IS US.

Ralph Johnson

YOUR METAPHORS ARE FUNDAMENTALLY BROKEN
“Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.”

ORGANIZATIONAL TRANSFORMATION
COMMUNICATION, TEAMWORK, HUMOR, HUMILITY...
THANK YOU!

- **Let me know what you think:**  @GaryPedretti
  Gary@SodotoSolutions.com

- **Contact Me regarding Training and Workshops in the Midwest (Chicago, Milwaukee, Madison, Minneapolis, St. Louis)**

- **Watch for a Leadership / Team Dynamics Workshop with me and Esther Derby – Winter 2015/2016, Chicagoland Area**

- **Check out the Scrum.org Booth! Scaling with Nexus! Become a trainer!**

Gary@SodotoSolutions.com  @GaryPedretti
RESOURCES AND CITATIONS

- Neal Ford
- Martin Fowler
  - http://martinfowler.com/articles/evodb.html
  - http://martinfowler.com/articles/designDead.html
- Ron Jeffries
  - http://xprogramming.com/xpmag/expEmergentDesign
- Scott Ambler
  - http://www.agilemodeling.com/practices.htm
- “Uncle Bob” Martin
  - https://sites.google.com/site/unclebobconsultingllc/home/articles/the-scatology-of-agile-architecture
- Rebecca Wirfs-Brock
  - http://wirfs-brock.com/blog/2011/05/12/agile-architecture-myths-3-good-architecture-emerges/
  - http://wirfs-brock.com/blog/2013/08/13/architecture-at-agile-2013/
- Mike Cohn
  - http://www.mountaingoatsoftware.com/blog/agile-design-intentional-yet-emergent