

# The [Software] Architect

Slaying the dragon of chaos by imposing order

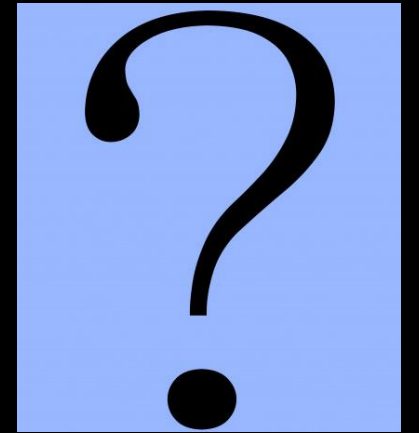
# About Me

- John Rivard
  - Native of Washington State
  - Married 29 years, two kids, two dogs
- University of Washington
  - B.S., Computer Engineering, 1989
  - **Go Huskies!**
- Work at Microsoft Corporation
  - 1991-present – a *loooong* time
  - Mostly on developer tools & languages
    - VB .NET, C#, .NET Framework, Visual Studio, Visual Studio Code
  - Currently Visual Studio services, [Visual Studio Live Share](#)
  - [johnri@microsoft.com](mailto:johnri@microsoft.com)



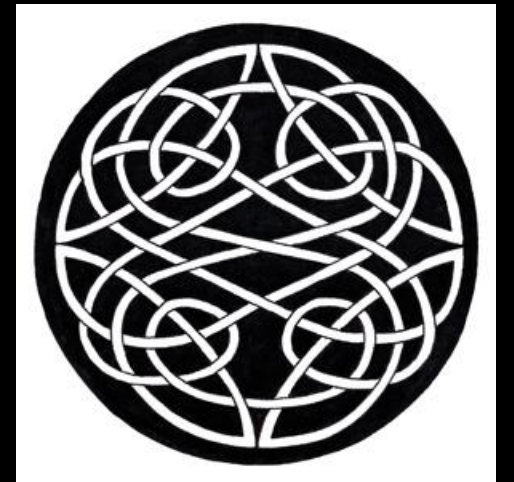
# What can I tell you...

- That you don't already know?
  - You all have several years experience doing software
- That you can't read online?
  - Design principals, techniques, technologies, etc.
- How about my experience "architecting" for a living?
  - What is a software architect?
  - What the heck do I do?



# An architect is more than...

- A craftsman – who cuts code like a pro
- An artisan – who fashions useful and beautiful systems and interfaces
- An engineer – who applies technology to solve problems
- A manager – who gets things done
  
- What is *my* definition an architect?



*An architect brings order out of chaos  
by creating durable structures  
that solve a problem  
for the benefit of humanity*

# Your job is to bring order out of chaotic world

- Create durable structures that last over time, over iterations
  - *Physical* structures – roads, bridges, buildings, cities
  - *Metaphysical* structures – social and political structures
  - *Virtual* structures – software systems
- Solve problems
  - The *big* problems, the *hard* problems
- Benefit humanity (a deliberate overstatement ;)
  - The scope of “architecture” implies big & important
    - To your colleagues, your customers, your business, your industry, your society
  - What you do *matters*! How you do it *matters*!

- You already know how to build software systems, and continue to learn, but...
- this talk is going to focus on the **chaos** bit  
...which feels like a very high percentage of what I *actually* do

# All kinds of chaos...everywhere

*BTW: All of these have happened to me, and then some!*

- Personal chaos
  - A health crisis or relationship crisis going on in your family
  - You're working too much and your sleep or physical health is suffering
- Social chaos
  - Your company makes an erratic decision to defund your project
  - Management wants a string of new features but won't invest to stabilize what has already shipped
  - Your team is full of newbies that don't understand the system
- Technical chaos
  - You're forced to work with legacy technology platforms
  - Your team has inherited code that is "crap"
  - A team member is all excited about the "shiny new thing" and wants to rewrite the world!
  - An industry paradigm shift is making your current solution obsolete

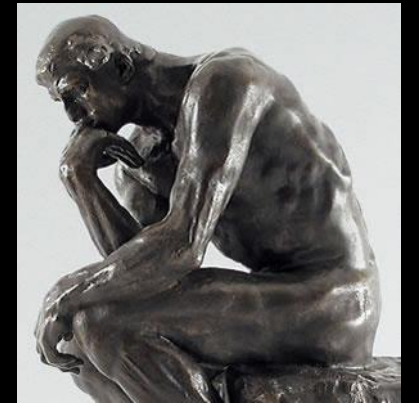


# How does an architect bring order?

- This isn't just a scientific question
- This isn't just a technical question
- It touches who you are and how you interact with others
  
- Yikes!
  - Maybe this sounds easy to you.
  - Or maybe it sounds a little scary like it does to me!

# Bring order to yourself

- Your personal presence must bring order, not chaos
  - Don't bring chaos with you, you'll likely not grow, and you may become despised ☹️
  - Take constructive feedback and criticism seriously, work on it!
- Order your personal life
  - Schedule time for your own physical and mental health
  - Schedule time for your family, your friends, your passions
  - Personal crises *will* come and *will* affect your work and career
  - Get help at the first inkling of chaos, not after things fall apart
- Grow your professional skills
  - Be always curious, be always learning, be always humble
  - *Congratulations! You're here!*

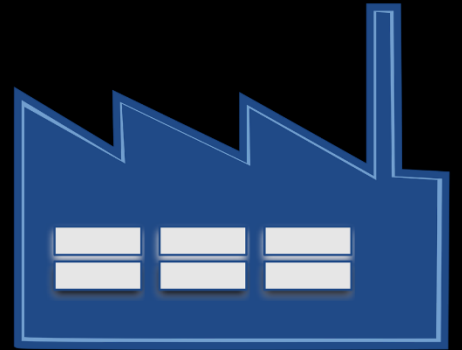


# Bring order to your team



- By bringing clarity
  - Clarify and communicate requirements, constraints, technical direction
  - Clarify “why the \*\*\*\* are we doing this?”
- By providing positive influence
  - Build and maintain trust, cultivate positive relationships
- By fostering processes that maximize order with minimum overhead
  - Simplify your engineering system wherever possible
- By treating your direct reports and colleagues with respect
  - You aren’t an expert at *everything*, everyone else is an expert at *something*
  - Be patient to teach whenever asked
  - Look after and grow others’ careers

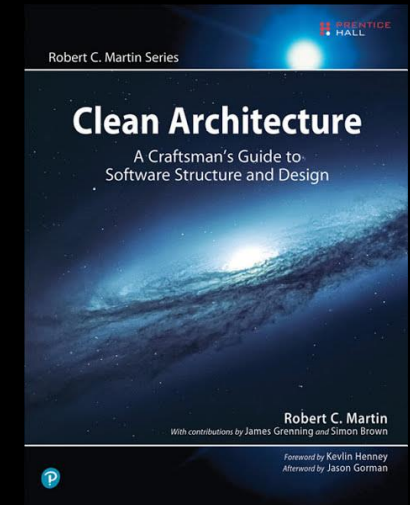
# Bring order to your business



- By working in concert with stated business goals
  - Serve the business gladly or move on if you cannot
  - Know what is important to their bottom line, to your division, to your product
- By being honest and transparent with your management
  - No sugar-coating problems, no avoidance of tough issues
  - Educate them as appropriate on technical and cost trade offs
- By learning what your customers really *need*
  - Not what you *think* they *want*
  - Work with product management to get real data, organize studies
- By honoring partner commitments
  - Communicate clearly with partners
  - Uphold your commitments to the best of your ability

# Bring order to your technology

- By developing a nose for good and bad smells
  - Where have you been bit in the past?
  - What patterns have worked especially well and are transferable?
- By learning and communicating sound architectural principles
  - See, for example, [Clean Architecture](#) by Robert Martin
- By identifying the holes in your tech that will require non-trivial solutions
  - You know it's coming, just a matter of when
- By not “fighting the last war”
  - Past successes bring skills, confidence, and know how
  - But last time isn't necessarily appropriate in your current context
  - Be fearless in trying something new, but prudent in mitigating risk
- Keep your vision on the horizon
  - New opportunities, new paradigms, new technologies



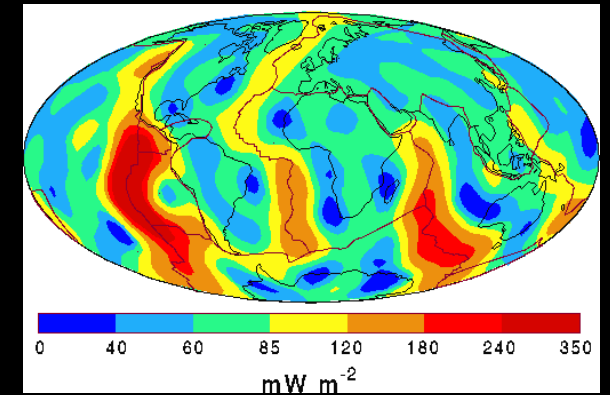
# Manage emotional chaos

- Everyone looks to you when s\*\*t happens!
- Keep calm when others are panicked
  - Take time to manage your own emotions first
  - Resist the temptation to criticize, complain, or blame
- Manage disruption with transparent communication—up, down, out
  - Reassure people who are upset or disillusioned
- Always clarify changes in requirements, constraints, schedules
  - Ask questions, don't assume, don't force your own way
- Negotiate in earnest
  - There will be conflicting interests amongst your team & stakeholders
  - Find a way, find the win-win



# Manage project-level chaos

- It's your job to pay attention to "climate change"
- Anticipate business-level chaos
  - Is the business going through a transition?
  - Are stakeholder requirements shifting?
  - Are there upcoming industry or political requirements? (GDPR anyone?)
  - Monitor changes in customer usage patterns, spikes, scale, regional differences
- Anticipate technology chaos
  - Are any dependencies being deprecated or defunded?
  - Audit and address security vulnerabilities regularly
    - Weak access controls, opened ports, un-patched software...
  - Plan for resilience with external dependencies – eliminate unreliable ones



# Manage chaos in production

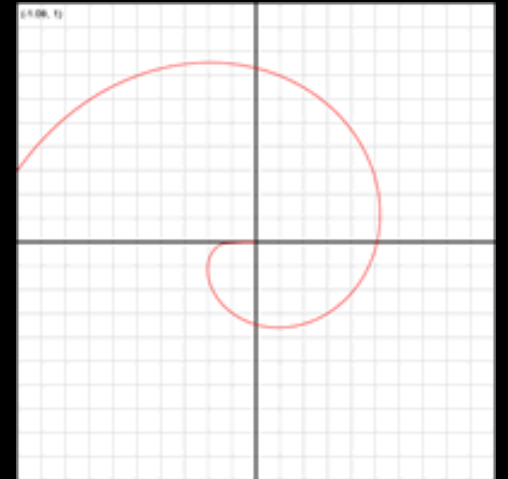
- Plan for outages
  - A natural disaster takes out a data center. What is the failover plan?
  - Human error takes out a critical subsystem. What is the impact?
- Plan for critical ops
  - Who has access to critical resources?
    - Can access be delegated if both “Priya” and “Daniel” are on vacation?
  - Are recovery processes documented?
  - Learn about “devops”!
- Manage rollouts carefully!
  - This includes any change in configuration, in code, or in dependencies
    - *Careful: “docker pull | FROM somelimage” can upgrade a baseline behind your back!*
  - Prefer to roll forwards only
    - Roll-back is often unreliable and leaves you in an unknown state
  - Roll out to test, to one region/datacenter at a time, to limited customer set
  - Take time to observe telemetry signals before proceeding





# Takeaway: It's your job to create order

- Anticipate chaos...from everywhere
  - From yourself, the team, the business
  - From project managers, stakeholders, and customers
  - From the technology itself...even the weather
- Become a person of order
  - Build confidence and trust, repair trust when broken
  - Inventory your own strengths and weaknesses – at least annually
  - Get coaching or counselling for self-improvement
- Bring order by being a person of positive influence
  - Cultivate relationships, positive persuasion, and collaboration
- Bring order by knowing and applying technology wisely
  - Anticipate and solve the problems that aren't obvious to all
- Keep the end in mind as a wedge through the chaos
  - Will your architectural structure be **durable**—enough?
  - Will your architectural structure **solve the *real* problem**?
  - Will your architectural structure provide the **desired benefit** to the customer, to the business?



# Thank you!

- Q & A
- Ask me anything