



---

# Agile 101

---

**Sunil Mundra**  
Principal Consultant

# Software Development Truths

- You can't gather all the requirements upfront
- The requirements you do gather will change
- There is always more to do than time and money will allow

Source: 'The Agile Samurai' by Jonathan Rasmusson

# Software Project Outcomes

	2002	2004	2006	2008	2010
Successful	34%	29%	35%	32%	37%
Challenged	51%	53%	46%	44%	42%
Failed	15%	18%	19%	24%	21%

# Why Agile? (Why Change?)

- Too many projects not delivered
- Software taking too long to get to market
- Requirements not met
- High costs to make changes after delivery
- Having to “get it right” first time/up front
- Too many defects
- Unhappy Customers
- Unhappy Developers

# A Bit Of History.....



# Software Development Methodologies

- Code-fix” (or no process)
- Structured, heavy weight methodologies a.k.a. “Plan Driven Methodologies” and “Waterfall”
  - Largely influenced by *traditional engineering and quality processes in industries*
  - Desire to make software development more *predictable, measurable and efficient*
  - Strong emphasis on *detailed planning and then executing to the plan*

# Why Software Is Different

- **Is not Tangible**
- **Is not based on Mathematics**
- **Needs Knowledge Workers**

# Relevance

- **Heavy weight methodologies are most successful when:**
  - Requirements are stable
  - Technology is well known and mature
  - Everything happens as one would expect
  - We are not taking on anything new or unknown
  - Coding is 'copy and paste'



**Today, projects with these  
characteristics are few and  
far between**

Heavy weight methodologies  
work in some instances, but  
there are **high costs**, and the  
**risk** in using them in  
**dynamic environments** is  
**high**

# Origins Of Agile

## Agile Methods are a reaction to:

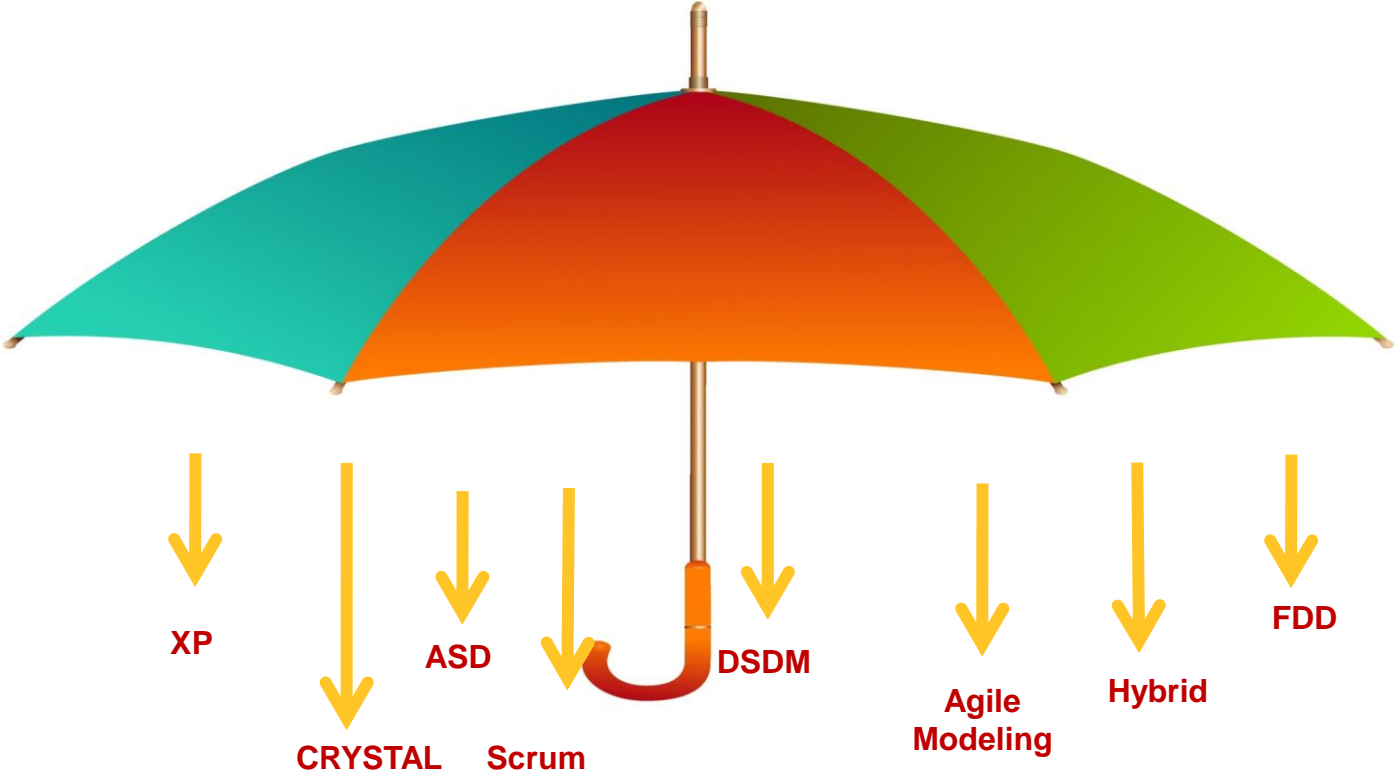
- Rigidity of heavy weight methods
- Bureaucracy introduced by heavy weight methods
- Unpleasant Surprises due to lack of visibility
- The myth that a well defined process is more valuable than the people who use it

2001: 17 Industry leaders met  
and decided to “name” a  
family of methodologies  
**Agile**

# Some Definitions Of Agile

- A new way of thinking about and preparing for the unanticipated
- The capability to successfully effect, cope with and exploit changes in circumstances

# Various Agile Methodologies



# Influences On Agile

## Agile Follows Systems Thinking

- System Thinking is a way of looking how things influence each other as a whole and not as individual parts
- Focus on Flow, not Function
- Look at the end-to-end process and the value we deliver our customers
  - What do our customers value
  - How do we respond to the demands from our customers, as a system

# Influences On Agile (contd.)

## Agile Follows Lean Thinking

- Add nothing but value (eliminate waste)
- Flow value from demand (delay commitment)
- Minimize inventory (minimize intermediate artifacts)
- Optimize across the organization



# The Agile Manifesto

*“We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:*

- *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.”*

# 12 Principles of Agile

- Our highest priority is to satisfy the customer through **early and continuous delivery of valuable software**.
- Welcome **changing requirements**, even late in development.
- **Deliver working software frequently**, with a preference to the shorter timescale.
- **Business people and developers** must work together throughout the project.
- Build projects around **motivated individuals**. Give them the environment and **support** they need, and **trust** them to get the job done.
- The most efficient and effective method of conveying information is **face-to-face conversation**.

# 12 Principles of Agile (contd.)

- **Working software** is the primary measure of progress.
- Agile processes promote **sustainable development**. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
- 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.
- At regular intervals, the team **reflects** on how to become more effective, then tunes and adjusts its behaviour accordingly.

---

# The Process

---

# Agile Project Lifecycle

**Inception**

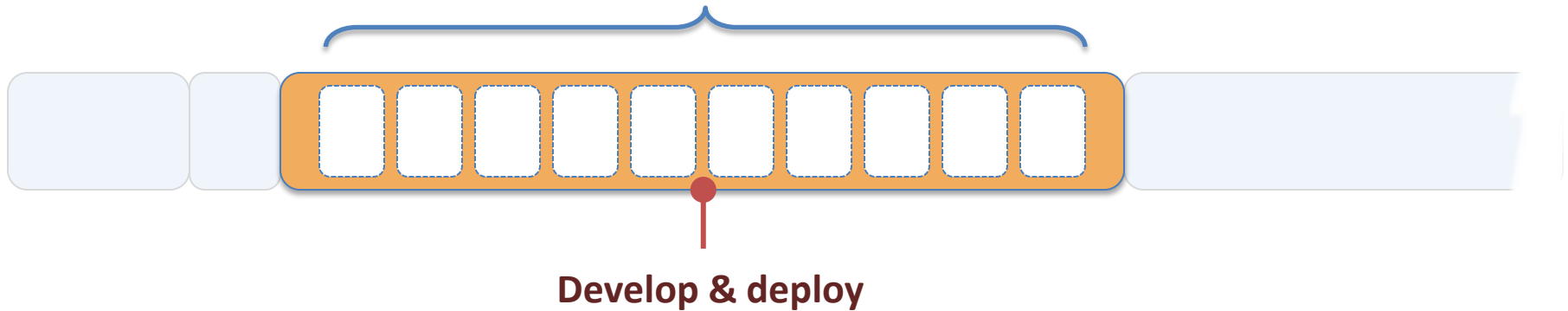
**Initiation**

**Develop &  
deploy**

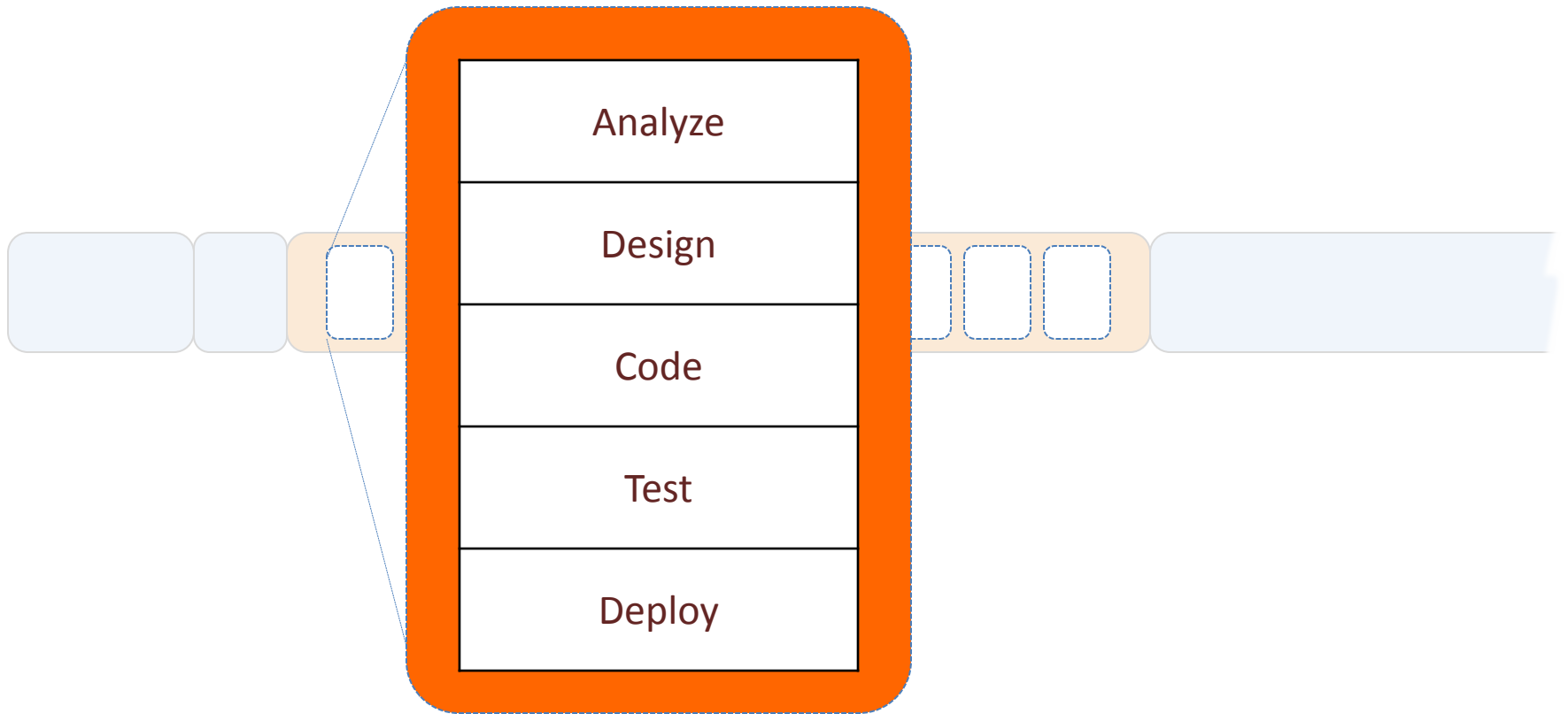
**Evolve /  
handover**

# Agile Project Lifecycle

Many time-boxed iterations

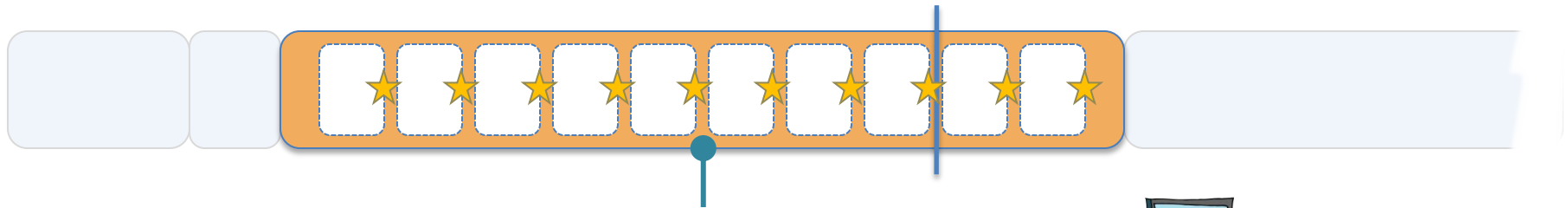


# The Basics Still Apply



# Agile Project Lifecycle

Working software showcased to the business



Develop & deploy





# Agile Is Iterative....



Credits: Jeff Patton

# And Incremental

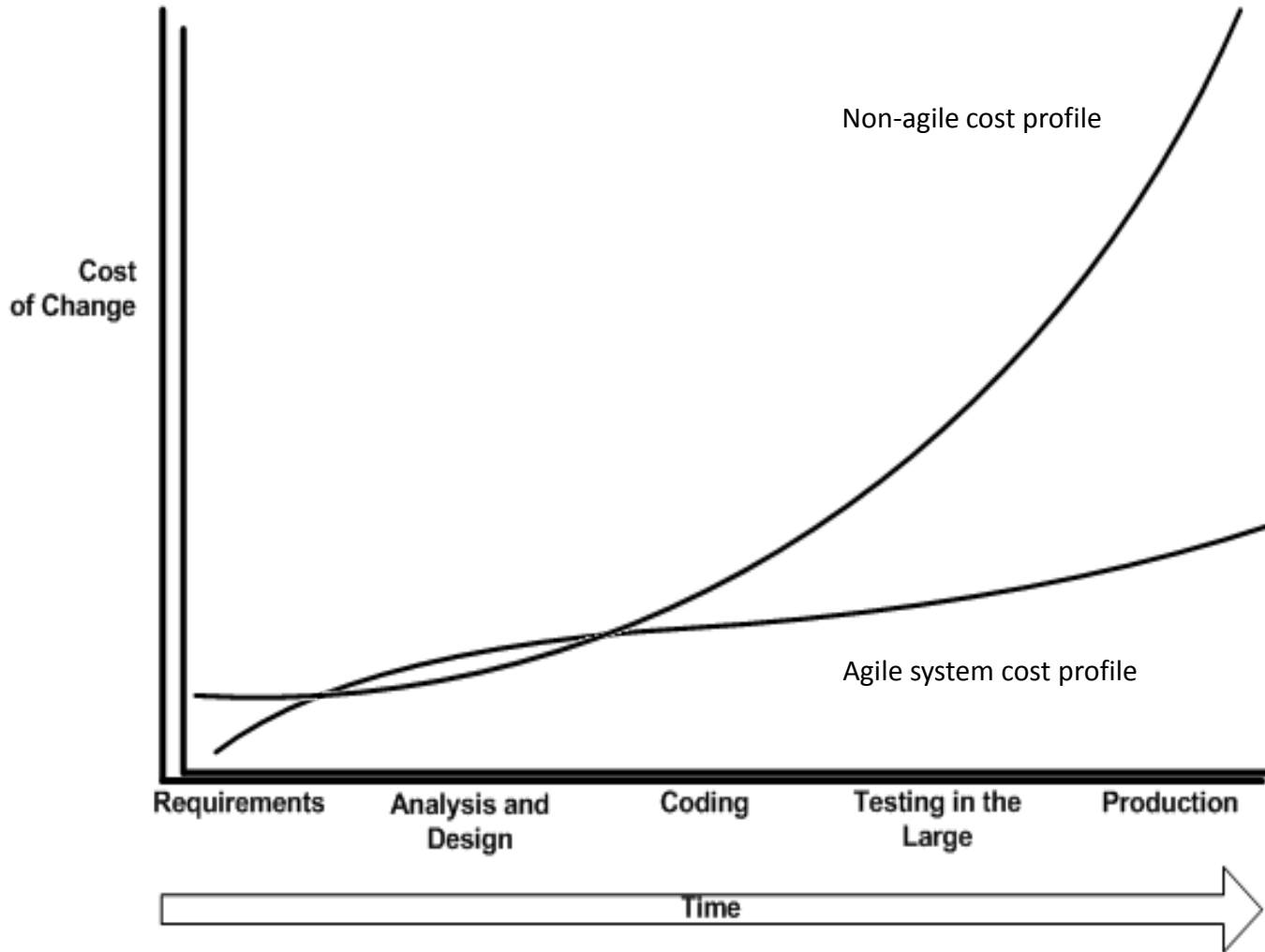


Credits: Jeff Patton

# Benefits of Agile

- Handles Changing Requirements and Priorities
- Lowers Cost of Change
- Better Visibility into Project Progress
- Reduces Risk
- Delivers Business Value Early and Often
- Maximizes Return on Investment (ROI)
- Encourages Higher Quality and Simpler Code

# Lower Cost of Change



# Agile Myths

- No Planning
- No Documentation
- Lacks Discipline
- Limited to Co-Located Teams
- Open Ended

# Making the Right Choice

**Agile**

**Waterfall**



**Volatile  
Requirements**

**Stable  
Requirements**

**Quality Focus From Start**

**Quality Focus Towards End**

**Customer Readily  
Accessible**

**Customer Not  
Accessible**

**Business Value  
Driven**

**Task Driven**

**Time to market  
matters**

**Fixed scope  
matters**

# Don't Be Afraid Of Agile!

“Fear leads to anger,  
anger leads to hate,  
hate leads to  
suffering”

(Yoda: Star Wars Episode 1)



# **Thank You**

**Email: [sunilrm@thoughtworks.com](mailto:sunilrm@thoughtworks.com)**

**Blog: [suniltalksagile.wordpress.com](http://suniltalksagile.wordpress.com)**