scruminc.

Agile Contracts
The Foundation of Successful Partnering

Hosts: Alex Brown
          Christine Hegarty

scruminc. Who We Are

Scrum Inc. is the Agile leadership company of Dr. Jeff Sutherland, co-creator of Scrum. We are based in Cambridge, MA.

We maintain the Scrum framework by:
- Capturing and codifying evolving best practices,
- Conducting original research on organizational behavior
- Adapting the methodology to an ever-expanding set of industries, processes and business challenges

We also help companies achieve the full benefits of Scrum through our full suite of support services:
- Training (Scrum Master, Product Owner, Agile Leadership, online courses, etc.)
- Consulting (linking Scrum and business strategy, customizing Scrum)
- Coaching (hands-on support to Scrum teams)
- Publishing and new content development

We run our services company using Scrum as the primary management framework, making us a living laboratory on the cutting edge of "Enterprise Scrum"

Find out more at www.scruminc.com.

scruminc.

Agenda

- The Problem with Traditional Contracts

- Principles of Agile Contracting
  - Align incentives
  - Collaborate and be transparent
  - Contractual and market repercussions for breaking trust
  - Regular discussion and refinement of backlog
  - Track the right metrics and incent continuous improvement
  - Build in flexibility to inspect and adapt
  - Include language to govern contract extensions

- What this allows you to do

- Illustrative case study – Scrum Inc. Web Migration Contract

scruminc.

Motivation: The Purpose of Legal Contracts

Elements of a Contract

- Promise to do something (Pledge)
- Exchange of value (Consideration)
- Agreement of additional terms
- Specifies what happens if one party fails to meet their promise

Value of Formal Contracts
- **Align** both parties on what will be done
- Set **boundaries** on the work, like scope and budget
- Define what constitutes satisfactory **completion**
- Describe what happens if there is a **breakdown in collaboration** between the parties

Contracts are not for day-to-day management! If the contract is being referenced, the project is already off track

scruminc.

Limitations of Traditional Contracts

Traditional contract structures run into many problems, particularly when governing Agile work.

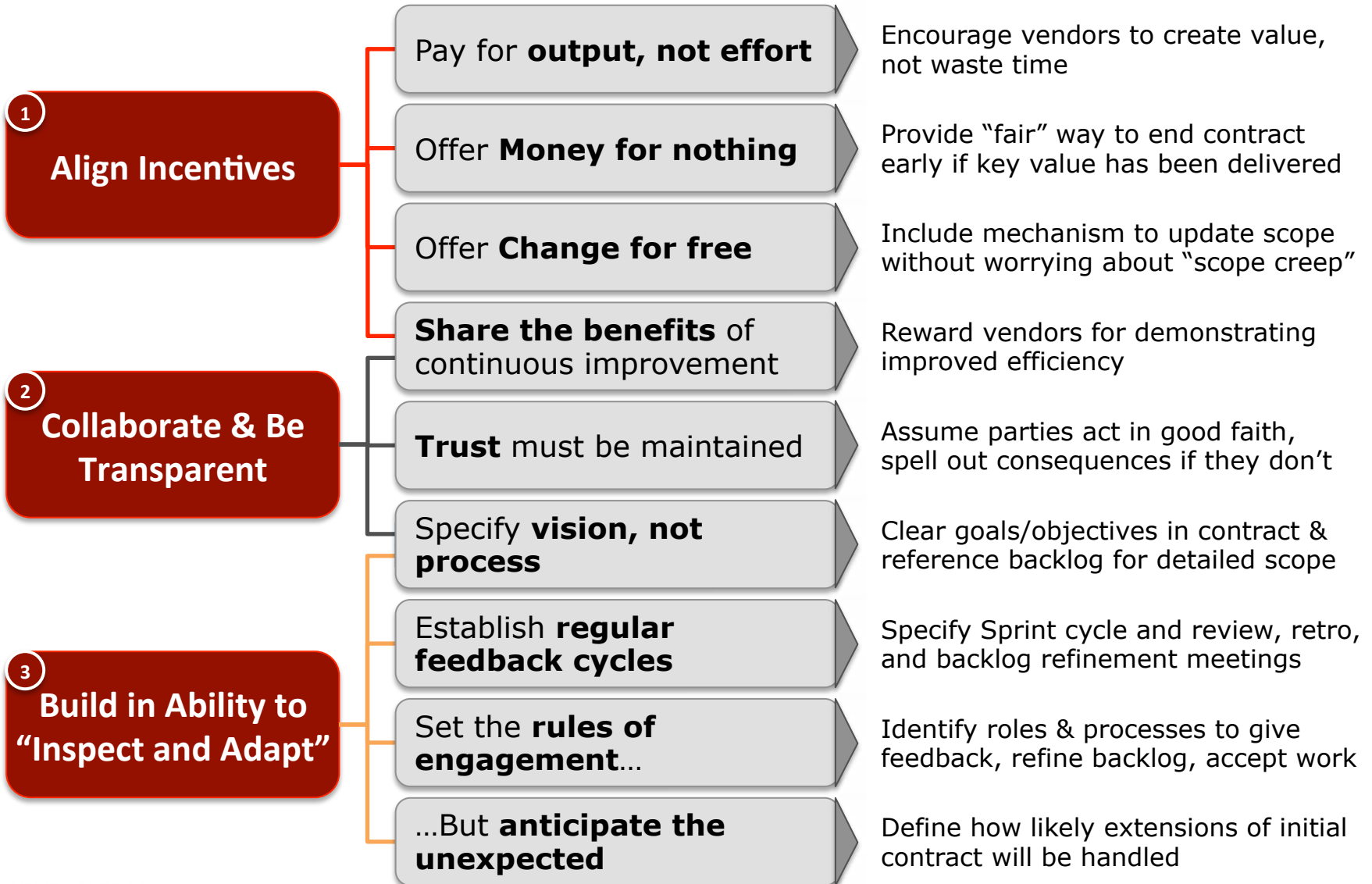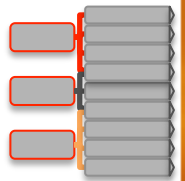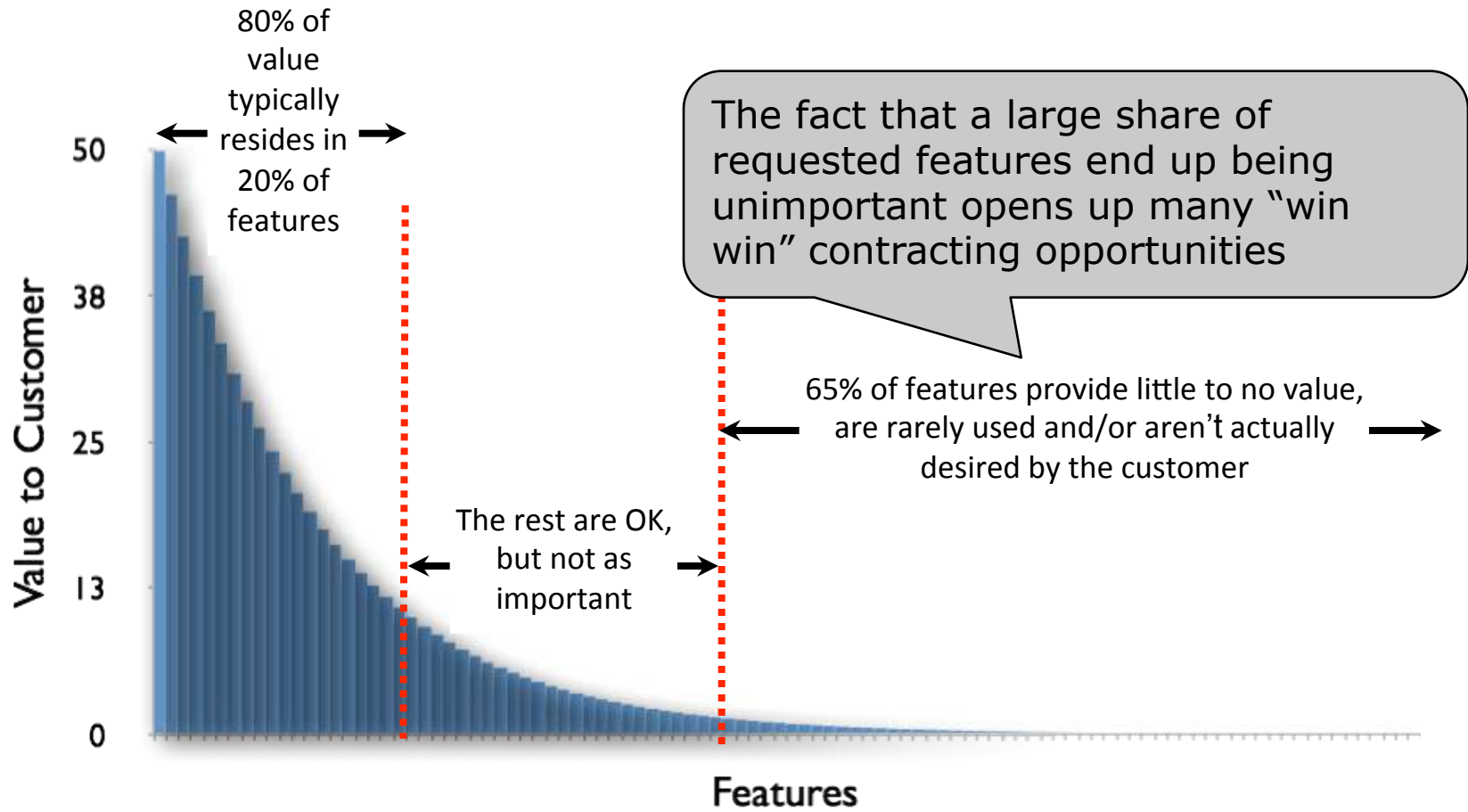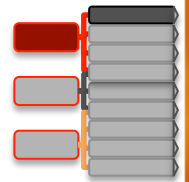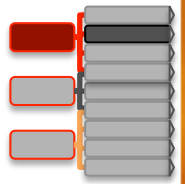|	
Incent Negatives Outcomes	• Assumes lack of trust between parties • Generally reward people for effort over delivering value • Discourages continuous improvement • Drive additional and unnecessary complexity • Often set unreasonable expectations
Erodes Value	• Typically specifies required process rather than clarifying vision • Limits contract partners from being as innovative as they can be • Sets up distracting confrontational relationship over scope • Wastes time and effort to manage complex processes • Prevents customer from getting what they actually want
Discourage Change	• Typically make change expensive to prevent "scope creep" • Vendor avoids hassle of change process • Not easy to incorporate new learning • Difficult to respond to new market conditions

Principles of Agile Contracting

1 — Align Incentives

Pay for **output, not effort**	Encourage vendors to create value, not waste time
Offer **Money for nothing**	Provide "fair" way to end contract early if key value has been delivered
Offer **Change for free**	Include mechanism to update scope without worrying about "scope creep"

2 — Collaborate & Be Transparent

Share the benefits of continuous improvement	Reward vendors for demonstrating improved efficiency
Trust must be maintained	Assume parties act in good faith, spell out consequences if they don't
Specify **vision, not process**	Clear goals/objectives in contract & reference backlog for detailed scope

3 — Build in Ability to "Inspect and Adapt"

Establish **regular feedback cycles**	Specify Sprint cycle and review, retro, and backlog refinement meetings
Set the **rules of engagement**…	Identify roles & processes to give feedback, refine backlog, accept work
…But **anticipate the unexpected**	Define how likely extensions of initial contract will be handled

scrum**inc.**

Pay for Value Delivery, Not Time Spent:
Not All Features Are Created Equal!

80% of value typically resides in 20% of features

The fact that a large share of requested features end up being unimportant opens up many "win win" contracting opportunities

The rest are OK, but not as important

65% of features provide little to no value, are rarely used and/or aren't actually desired by the customer

Value to Customer

50
38
25
13
0

Features

scruminc.

Agile Contracts: Money for Nothing

1. Projects are usually prioritized by return on investment.

2. Ordering your Product Backlog allows you to prioritize features by return on investment.

3. Since 65% are never or rarely used, during the project it will become evident that the next low priority feature costs more than the value it delivers.

4. Stop the project at that point and deploy the valuable features. Contractor and customer split remaining budget.

5. All projects should deliver early and save money.

Value to Customer: 50, 38, 25, 13, 0

Features

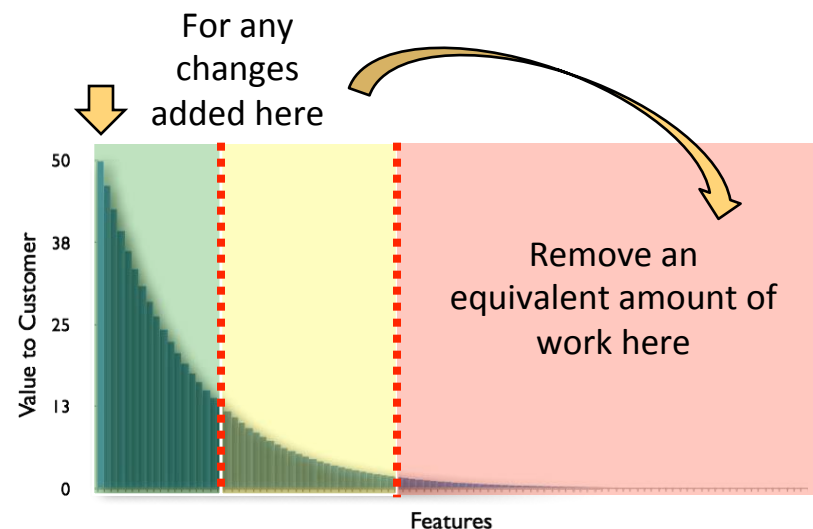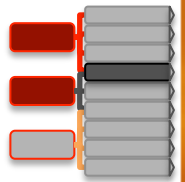End project here

Split this remaining budget
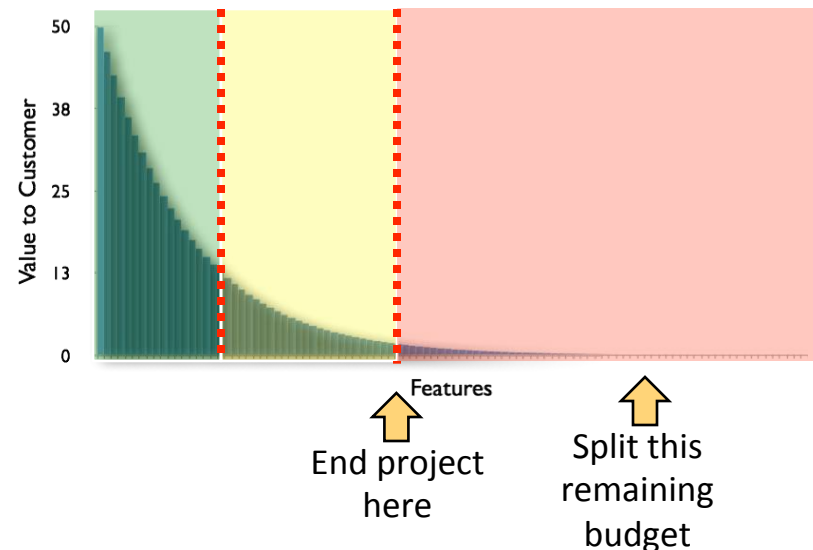
Agile Contracts: Change for Free
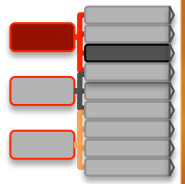
1. Create a prioritized backlog of work to be done with highest business value items first.

2. Implement in short sprints, always less than a month.

3. When higher priority requirements emerge, put them in the next sprint.

4. Cut an equivalent amount of the lowest priority items out of the project to the work added. These features are unlikely to be used anyway.

5. Change for free allows you to meet your budget and deliver on time.

For any changes added here

Remove an equivalent amount of work here

Value to Customer

50
38
25
13
0

Features

scruminc.

Share the Benefits of Continuous Improvement

Traditional Incentive Structure	Common Undesired Outcome	Better Incentive Structure
Paid by Hour	• Rewards consumption of time • Actively discourages productivity improvement	• Pay per point delivered
Paid by Lines of Code	• Rewards volume of output, not quality • Discourages clean, concise code (E.g. Hospital Medical Errors)	• Pay by business value delivered
Partially Completed Functionality	• Incents enormous amounts of Work in Progress (E.g. EVM)	• Only count releasable functionality in business value delivered
Fixed Date <u>and</u> Scope	• Encourages teams to cut corners to deliver on time (E.g. HealthCare.gov Web Launch)	• Pick one or the other!
Overly Complex Approval Process	• Slows delivery • Limits vendor's ability to experiment with process or product improvements	• Have a single point of approval (like a Product Owner) • Have enough of a process to guarantee quality, and no more

scruminc.

Trust, Vision and Regular Feedback Cycles
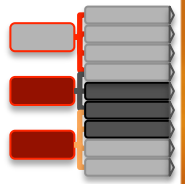
Trust must be maintained

"Scope of work will be mutually negotiated through backlog."

"Vendor will estimate point value of all features in the backlog."

"For projects ended because sufficient value has been delivered, the vendor will receive 25% of remaining contract price. However this will not apply to projects ended due to breach of trust."

Specify **vision, not process**

"Currently, the website is divided among 3 different platforms with different levels of functionality. Our goal is to integrate these into a single platform with functionality that is editable by staff with limited need for on-going external support and maintenance."

"Backlog will reflect the prioritized list of features in the upcoming sprint."

Establish **regular feedback cycles**

"The vendor will meet with the purchaser for a weekly review meeting on Mondays to demonstrate newly created functionality."

"Purchaser agrees to provide feedback on the demonstrated feedback in a timely manner."

Flexibility to Inspect and Adapt:
The US Constitution as an Agile Contract
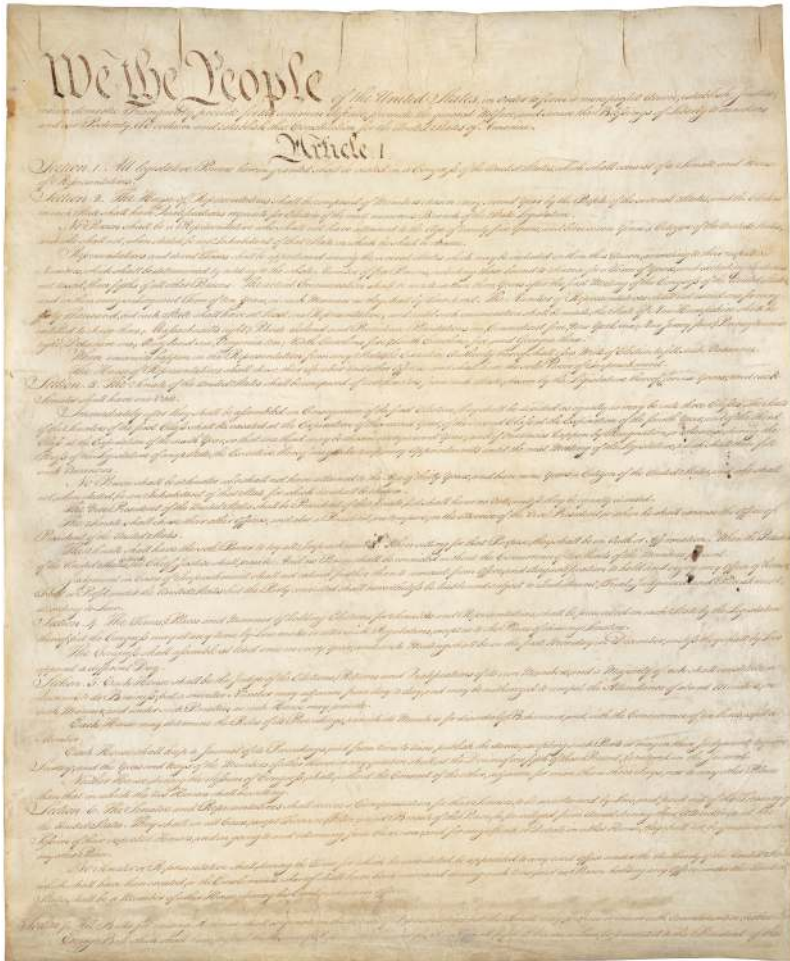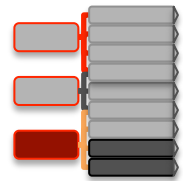
Sets the rules of engagement

- Vision for system of government
- Three branches of government
- Branch powers and responsibilities
- How new laws are made
- Interactions between stakeholders
- When it comes into force

But leaves room to inspect & adapt

- No actual laws included
- Provision for amendment
- Regular election cycle every 2-6 years

Same social contract still relevant ~250 years later

What Agile Contracts Allow You to Do

Satisfy the purpose of a legal contract

Avoid unpleasant surprises, "black hole" projects and the potential consequences of a chaotic breakdown in collaboration

Get things done faster without penalizing the vendor

Faster development time allows you to get new innovations to market first and crush the competition

Change direction easily without raising cost

Incorporate insights as you go to consistently build a successful product that customers actually want to buy

Unleash your contract partner's creative potential

Informed partners can make better decisions and may deliver innovative solutions you never could have imagined

Illustrative Case Study:
Scrum Inc. Web Migration – The RFP

Specified that the contract would be AGILE, and what that means

What we <u>did</u> specify	What we <u>didn't</u> specify

Vision and context

Not → Exactly what needed to be done

Initial backlog of features
- With expectation we will refine it further together before contracting

Not → A fixed and defined scope

Rules of engagement
- Partner must be willing to use Scrum
- We will manage backlog together
- We expect "Change for Free"
- Demo every Monday at our Review
- We need access to demo site to test

Not → An elaborate set of procedures for approval, cross-checking and verifying contract compliance

Offered "Money for Nothing" to sweeten the deal

Actual RFP available to view on ScrumLab

Illustrative Case Study:
Scrum Inc. Web Migration – Selecting a Vendor

Collection

Screening

Evaluation

Selection

1 RFP Posted online, tweeted, and e-mailed to a targeted group of 15 local service providers

2 Initial expression of interest and request for further information from 150 companies

3 Initial screening based on:
1. Demonstrated projects
2. Familiarity with likely platforms
3. Past experience with or willingness to learn Scrum

Resulted in 9 short-list candidates

4 Live meetings to discuss project and assess whether vendors are truly agile. Focus on **HOW** we can work together more than **WHAT** they have proposed

5 Team selection of best qualified vendor

Seattle-based *Portal Integrators* unanimously selected as preferred partner

scruminc.

Illustrative Case Study:
Scrum Inc. Web Migration – <u>Our</u> Agile Contract

The contract has 6 sections:

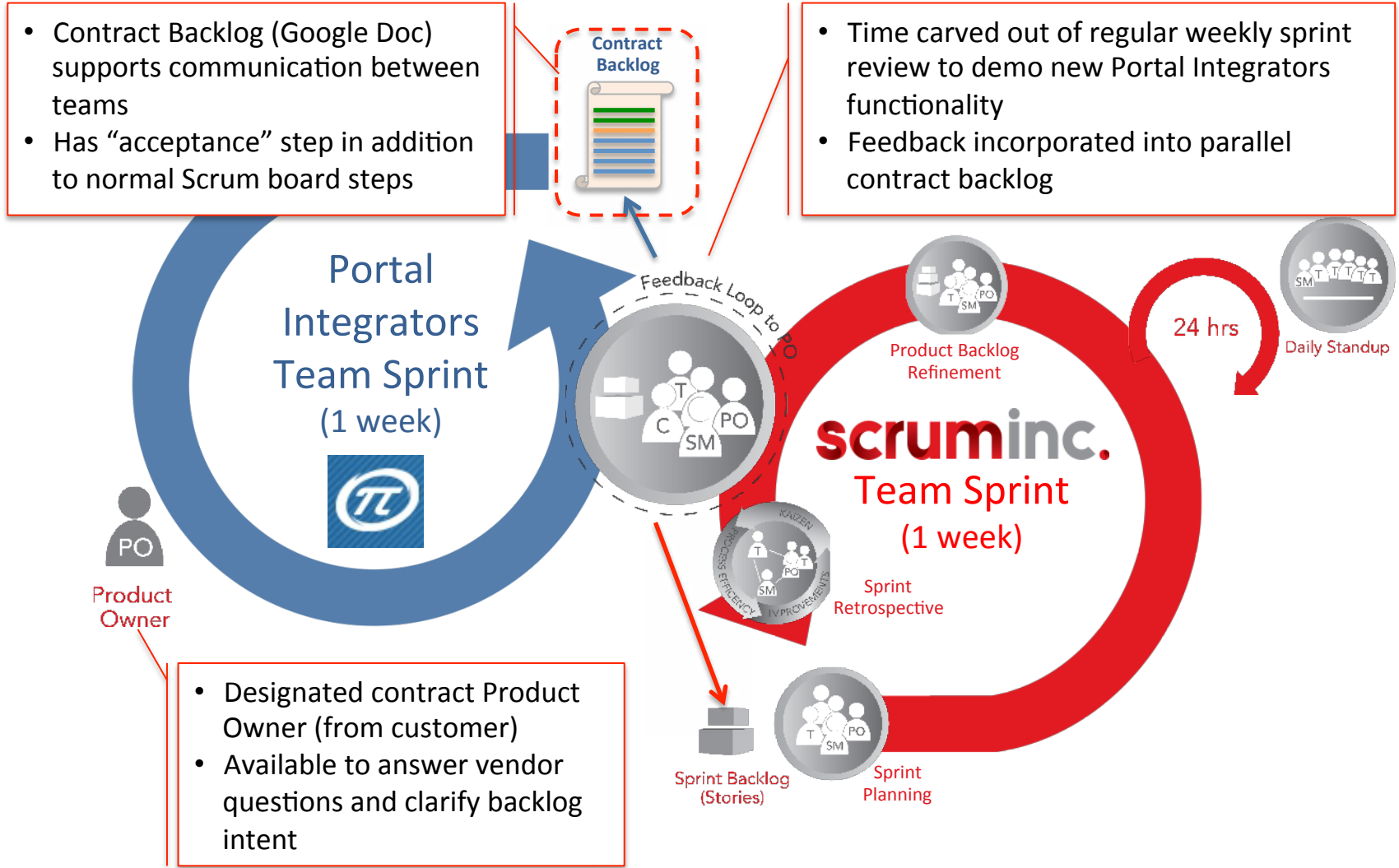|	
Period of Performance	• 1-week sprints, until ended by either party
Engagement Resources	• A Dev Team in the Philippines with a SM & Proxy PO • A customer Product Owner to answer any questions
Scope of Work	• Clear goals for first 3 sprints (proof of concept) • After that, contract backlog will define scope
Client & Contractor Responsibilities	• Client provides feedback & maintains prioritized backlog • Contractor provides transparency and weekly demo
Fee Schedule	• Contractor is paid at the end of each 1-week sprint
Termination/ Renewal	• Both parties can decide to renew or cancel after each sprint review

scruminc.

Illustrative Case Study:
Scrum Inc. Web Migration – Working Together

- Contract Backlog (Google Doc) supports communication between teams
- Has "acceptance" step in addition to normal Scrum board steps

Contract Backlog

- Time carved out of regular weekly sprint review to demo new Portal Integrators functionality
- Feedback incorporated into parallel contract backlog

Feedback Loop to PO

Portal Integrators Team Sprint
(1 week)

Product Owner

scruminc.
Team Sprint
(1 week)

Product Backlog Refinement

24 hrs

Daily Standup

Sprint Retrospective

Sprint Backlog (Stories)

Sprint Planning

- Designated contract Product Owner (from customer)
- Available to answer vendor questions and clarify backlog intent

scruminc.

Illustrative Case Study:
Scrum Inc. Web Migration – The Results So Far

What has worked well	Challenges

What has worked well

- Vendor makes excellent suggestions to improve product
  - E.g. plug-ins that provided significant functionality rapidly

- Contract backlog makes clear:
  - What vendor is working on
  - What will be covered at the coming review
  - What functionality vendor believes is done
  - What has been officially accepted

- Review meetings provide collaborative dialogue
  - Manage "discovered work" together

- Starting to develop release plans

Challenges

- <u>Significant</u> effort to verify that vendors **actually are** agile
  - Many say they are…but aren't

- Discipline needed for internal stakeholders to interact regularly with the draft product to provide feedback

- Contract backlog separate from main team backlog requires greater ongoing PO effort to coordinate

scruminc.

Agile Contracts are Coming

Section 804 "Implementation of new acquisition process for information technology systems" added to the 2010 Defense Acquisition Bill. These are the rules that the Department must follow when purchasing anything.

The key language is:
2. Contracts must be designed to include…
    A. Early and continual involvement of the user;
    B. Multiple, rapidly executed increments or releases of capability;
    C. Early, successive prototyping to support an evolutionary approach; and
    D. A modular, open-systems approach.

4. Enable Agile IT
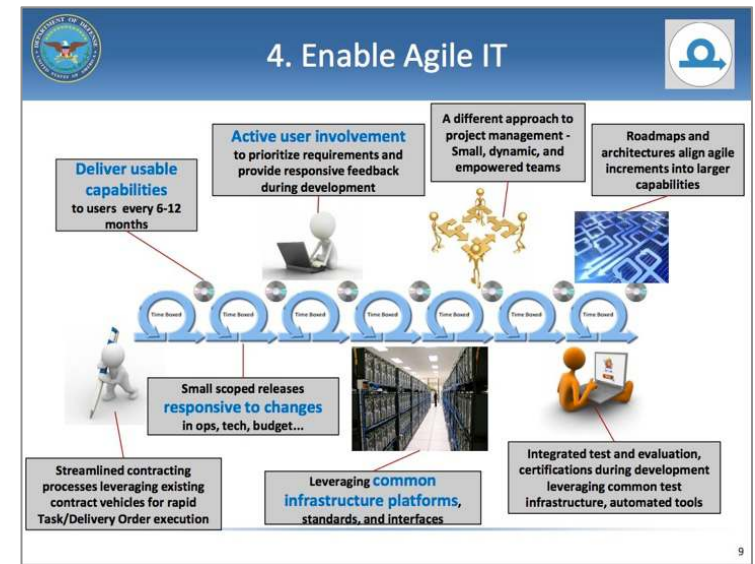
Basically, DoD now requires Agile Contracting!

Conclusion

- At some point, traditional procurement becomes an impediment to most Scrum teams

- Agile contracts can offer a better way to work with suppliers

- These methods are in early stages of adoption…

- …but they are being used successfully and increasingly popular

Questions?

Stay Connected

Our Website
- check in for announcements, new content and services, book releases, and more!
- www.scruminc.com

ScrumLab
- join the conversations on our forums with the scrum community and your class.
- coming soon: articles, videos, papers on all things scrum
- scrumlab.scruminc.com

Blog
- scrum.jeffsutherland.com

Webinars
- advance your learning with our interactive webinars. visit the scrumlab store to view upcoming topics.

Twitter, Facebook, and G+
- @jeffsutherland, scrum and scrum inc.

scruminc.