Agile Contracts: Building Trust

Ewan Milne
ewan@calenture.org
Contracts: a necessary evil?

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.

© 2001, see http://agilemanifesto.org
Contracts: a necessary evil?

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.

© 2001, see http://agilemanifesto.org
Contracts: a necessary evil?

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.

© 2001, see http://agilemanifesto.org
Let's blame the lawyers!

Lawyers think differently from us. Creatures of habit, guided by the tried and tested. Trained to be distrustful.

– If not of other parties, at least of anticipated outcomes.

Have a duty to think the unthinkable.

Often view software projects as sequential and relatively predictable processes.
Let's work with the lawyers

Include the legal staff in your cross-functional team.

Educate lawyers in agile development:

• How agile can reduce risk
• How iterative schedules can affect the delivery and payment cycle
• How adaptive planning can replace traditional change management
• How a goal can drive project, eliminating prescriptive lists of deliverables
• How early termination can be positive!
Let's work with the customers

Try to engage customers with relational contract models where goals are aligned, and a relationship based on trust and support can develop.

Ensure that you establish a contract which:

- manages risk appropriately,
- provides flexibility to allow for change,
- provides clarity with regard to obligations and expectations for both parties.
Fixed Price

Very "popular" form of contract.
Designed to protect the customer by transferring risk to the supplier... in theory.

- Fixed Price;
- Fixed Schedule;
- Fixed Scope.

Change Control: RFCs, Off-Specs.
Chance of failure even if the terms of the contract are fulfilled.

Actively erects barriers to collaboration and trust.
The *other* commonly used contract type. Acknowledges uncertainty. Moves risk from supplier to customer. Does **not** incentivize supplier to be efficient (actually, quite the reverse). *Can* be used as the contractual basis for agile development. ...but still fails to establish a collaborative relationship based on trust.
Optional Scope Contract

Proposed by Kent Beck and Dave Cleal back in 1999.

Return to the Iron Triangle: Fixed Price means
Fixed time, fixed cost, fixed scope \(\therefore\) variable quality

Whereas Optional Scope Contract proposes
Fixed time, fixed cost, \textbf{fixed quality} \(\therefore\) variable scope

“Customers can now have what they want at the project end, after they’ve learned, instead of getting what they wanted at the project start.”

Improved alignment of interests.

Ewan Milne | @ewanmilne | Agile Contracts: Building Trust | Agile on the Beach 2013
Incremental Delivery

One way to address inherent risks of Fixed Price. Fixed Price per iteration. Easy to adapt to agile development. Incremental delivery, regular payment. But also, frequent opportunities for either party to abandon. Customer risk mitigated if highest priority features are delivered first. Supplier risk mitigated by proving their value, building relationship.
Price per Story Point

Directly bakes Story Points into contract.

- Short initial period to set up backlog, calibrate velocity.
- Supplier estimates and decides the points.
- Customer sets priorities, pays on acceptance.

But as Allan Kelly would have it, "Just Say No!".

Goodhart's Law; 

*When a measure becomes a target, it ceases to be a good measure*

Gaming of the system.

Misuse of points-based estimation.

Not a very good way to build trust.
Options covered so far fail to fully align the incentives of customer and supplier. Joint effort implies shared risk, and both parties striving for a common purpose. We want contract to act as platform to build partnership. Target cost contract makes total cost a joint responsibility. Define target cost, and therefore price. Any costs above target shared fairly. Any savings under target also shared. “Shared pain, shared gain”.
Money For Nothing, Change For Free

Proposed by Jeff Sutherland.

A variation on Incremental Delivery, with addition of two clauses to specifically support agile development.

- Money for Nothing
- Change for Free

Start with “a standard fixed price contract, which includes time and materials for changes”.

Customer and supplier required to follow Scrum rules, otherwise contract reverts to time and materials.
Money For Nothing

Customer decides on ROI cutoff, when proceeding to next sprint does not offer sufficient business value.

Supplier receives cancellation fee based on remaining profit they would expect from remaining work.

- Supplier is cushioned from effects of early termination.
- Customer does not pay for work that they no longer want.
Product Backlog is reprioritised at end of each iteration. Changes can be included at this point. New features added for free, if low priority items of equal size removed from Product Backlog.

- Allows the ability to respond to change by altering scope within contract terms.
Framework Agreements

Agreement between customer and supplier, defining general terms for specific contracts over period of agreement.

Common in public sector procurement.

Establishing a relationship between customer and suppliers.

However, multi-supplier frameworks tend to lead to lengthy procurement processes.

EU procurement rules are not compatible with agile approach.
Universal Credit – flagship of UK Govt embrace of agile. Agile approach ditched by DWP in a high profile move seen as a failure of agile at scale. But the fundamental problem was procurement:

- Contracts negotiated with major suppliers before adoption of agile;
- Feature set fixed in contract at start of 7 year programme;
- Delivery schedule took incremental but not iterative approach;
- You might call it agile, but if you don’t set up the contract properly, it’s not.
Framework Agreements

UK Government's **Digital Services Framework**. Access to suppliers with right (agile) capabilities. Open to all capable suppliers:

- SMEs
- Agile practices within tier 1 suppliers

Supporting agile methodologies with appropriately balanced terms and conditions.

- Fixed Price
- Time and Materials
- Capped Time and Materials
- Price Per Story Point

Ewan Milne | @ewanmilne | Agile Contracts: Building Trust | Agile on the Beach 2013
Summing up

Do not just settle for Fixed Price – it’s not the only game in town.
Collaborate with your legal staff and your customer or supplier.
Accept that software development is a dynamically complex activity.
Explore the options available to you.
Establish an agreement which aligns interests, and allows the development of a relationship based on trust.
References


Jens Coldewey: Contracting Agile Projects.
(www.cutter.com/content-and-analysis/resource-centers/agile-project-management/sample-our-research/apmu0617.html)

Allan Kelly: Agile Contracts
(www.infoq.com/articles/agile-contracts)

Allan Kelly: Points based contracts? Just Say No.
(allankelly.blogspot.co.uk/2012/05/points-based-contracts-just-say-no.html)

Sourcing For Agile
(www.emergn.com/insights/blogs/insights/thought-papers/sourcing-for-agile)

Jeff Sutherland: Agile 2008 - Money For Nothing
(scrum.jeffsutherland.com/2008/08/agile-2008-money-for-nothing.html)

Kent Beck, Dave Cleal: Optional Scope Contracts
(xprogramming.com/articles(optional-scope-contracts)
Thank you for listening. Any questions?