

Agile Contracts *Truth & Options*



Agile 4 Agencies,
October 2014

Allan Kelly

[Twitter: @allankelly.net](https://twitter.com/allankelly.net)

allan@softwarestrategy.co.uk

<http://www.softwarestrategy.co.uk>



Allan Kelly...

- Provide advice on software development & business strategy
- Training in Agile, Coaching, Consulting

Author

- **Xanpan:** Team Centric Agile Software Development <https://leanpub.com/xanpan>
- Business Patterns for Software Developers (2012, Wiley - ISBN: 978-1119999249)
- Changing Software Development: *Learning to be Agile* (2008, Wiley)

Xanpan

- Team centric agile software development
- Combining Kanban and XP - inspiration for creating your own hybrid



Allan Kelly

Business Patterns for Software Developers

Allan Kelly

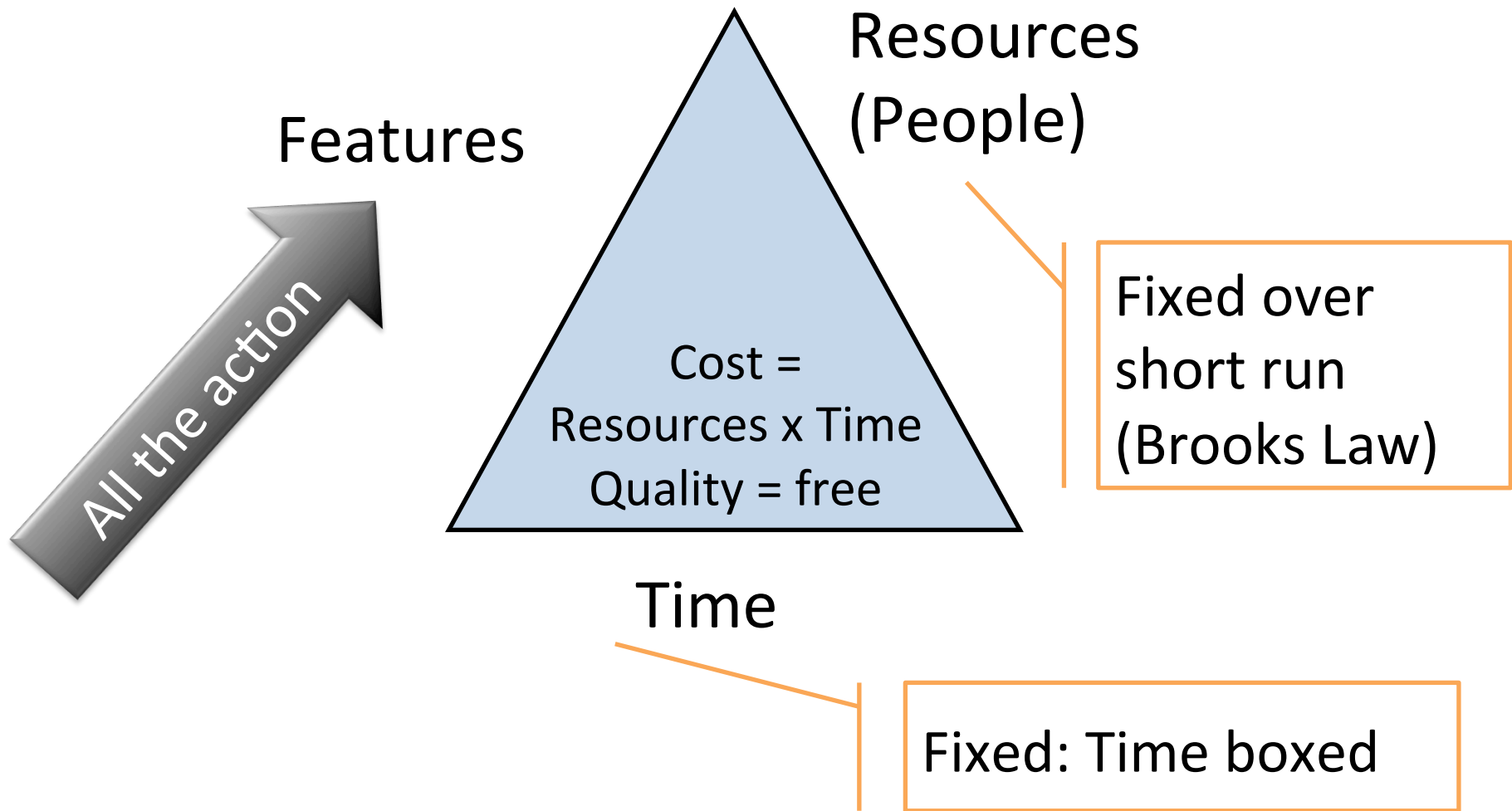


Forward by Linda Rising

- Clearly defines the route to business strategy and operations
- Includes over 38 strategy patterns
- Explains how to implement specific patterns
- Includes sequence diagrams linking patterns

Fixed Price
Fixed Time
Fixed Features

Really?



If you summarize the difference between Agile and traditional working in one thing....

- In traditional work you decide the features before day-1 and try not to change until you have to (usually near the end)
 - & stop when you run out of time
- In Agile you decide features on day-1
 - And on day-15
 - And on day-29
 - And every two weeks
 - & when STOP when the client doesn't want more

You always trade features in the end
(Especially on traditional work)

But clients want to know what they
will get...

Agile Contracts?

Agile contracts

- Probably most “Agile” contracts still
 - Time and Materials
 - Fixed price, Fixed scope, Fixed schedule
 - Agile as better delivery



Agile contracts

- Customers need to expect
 - More involvement
 - Regular Show-and-tell
 - Regular Reprioritization
 - Frequent questions



Option 1: Hide IT

- Do Agile but don't tell the customer
- Sign a perfectly normal contract
- Do Agile behind the scenes & reap the reward
- + Sounds like it could work
- + Doesn't upset customers
- Trust, Honesty? Conflicts with Agile ideals

Option 2: T&M + Agile delivery

My guess.... This is what most firms do

+ Easy to understand

- Incentive for supplier to accept everything
 - and let the backlog GROW
- Client sold on “Agile” may also believe everything is possible, it will be fast, cheaper...
 - and they can do what they like
- No particular incentive to change behaviors'

Option 3: No cure, No pay

Best for “Solve my problem” objective

No good for “Do this list of 100 things”

Honesty to the fore

- All risk transferred to provider
- No incentive for client to be involved

Option 4: Rolling

Agree an initial contract, say 2 weeks

Client has option of 2 more weeks if they like

Contract is a framework for work to happen

Option 4: Rolling

- + Team incentive to focus on delivering benefit
- Difficult for supplier to schedule people
 - (But if they are so good they should have plenty of customers)
- No long term contracts
 - Money men won't like, no big headlines

Rolling contracts

Toby Parkins

Sell customer a little bit

Sell a bit more (if they like it)

Fewer big contracts

More repeat business



headforwards

Option 5:



- Money for nothing
- *Change for Free*

- From Jeff Sutherland ~ 2007/8
 - <http://jeffsutherland.com/Agile2008MoneyforNothing.pdf>
 - <http://scrum.jeffsutherland.com/2008/10/agile-contracts-money-for-nothing-and.html>

Money for Nothing

- Supplier delivers monthly
- Client has right to cancel project at any time
- Remaining fee split 80/20

Change for Free

- Start with a feature list/backlog
- Client reprioritizes monthly
- Can add new work at any time
 - Accepts existing work might fall off the end

Money for nothing?

- Sounds good but...
 - They haven't taken over the world
- A fancy version of rolling contracts?

Option 6: Two bids

- Bid twice on the same piece of work
- Offer the client a very traditional contract
- And offer them an “Agile” contract

Think Again



Think Again

Software is a service

Software development is a service

Therefore

Sell the service not a product



Deciding what to build
is part of the work

Think Again

Don't sell

~~We will build
product A with
features B, C, D, ...~~



Do sell

We will provide a
development service to you

We will work with you
to solve problem A

And unlock benefits (which
might be B, C, D, ...)



Contracts do not state what will be built

So sales thinking needs
to change...



Think Price - Typically...

$$\text{Price} = \text{Profit} - \text{Cost}$$

$$\text{Cost} = \text{People} \times \text{Time}$$

$$\text{Price} = \text{Profit} - (\text{People} \times \text{Time})$$

Therefore

We must know cost to start

Add a % for contingency

Add a % for profit

Fudge (what we think the customer will pay)



But...

How much will the customer pay?

How much value is this worth?

When do they want it?

How much can we do something for?

Do sales people leave money on the
table?



Think again

$$\text{Profit} = \text{Price} - \text{Cost}$$

Work to customer deadlines (time is fixed)

Start with a small team

Prove you can deliver

Prove you can deliver benefit

Only then flex team size



With a rolling contract...

Small upfront price tag should make sale easier

Work will keep coming in from happy clients

Trade big sale for ongoing sales

Further Reading – by Allan

- Agile Contract Options
 - InfoQ
<http://www.infoq.com/articles/agile-contracts>
 - Software Strategy (PDF version)
<http://tinyurl.com/mq4qpsn>
- Dear Customer, The Truth about IT projects
 - <http://www.agileconnection.com/article/dear-customer-truth-about-it-projects>
 - <http://tinyurl.com/lw9kqlj>

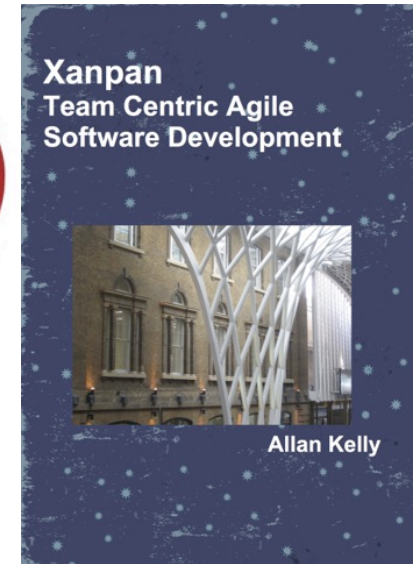
Agile Contracts

allan kelly

Twitter: [@allankellynet](https://twitter.com/allankellynet)

allan@softwarestrategy.co.uk

www.softwarestrategy.co.uk



Xanpan

- Team centric agile software development
- Combining Kanban and XP - inspiration for creating your own hybrid



Allan Kelly

Xanpan

eBook <https://leanpub.com/xanpan>

Print <http://tinyurl.com/nnu7hom>

Half price in September