

Contracting for Agile Software Development

Hilton Worldwide, Inc.

Paul Lanigan

Technology Sourcing Executive

Mayer Brown

Derek Schaffner

Counsel



Contracting for Agile Software Development

Derek Schaffner
Counsel
Mayer Brown
+1 202 263 3732
dschaffner@mayerbrown.com

Paul Lanigan
Technology Sourcing Executive
Hilton Worldwide, Inc.
+1 703 883 1087
paul.lanigan@hilton.com

Presenters



Derek Schaffner

Counsel

Mayer Brown

+1 202 263 3732

dschaffner@mayerbrown.com

Derek Schaffner is counsel in Mayer Brown's Washington DC office and a member of the Technology Transactions practice. He represents clients in a wide variety of information technology and business process outsourcing transactions and other information technology licensing and development transactions. *Chambers* noted that the support Derek provides clients on the most critical deals “earns him the respect of clients, who give him ten out of ten.”



Paul Lanigan

Technology Sourcing Executive

Hilton Worldwide

Paul Lanigan is a Technology Sourcing Executive and avid practitioner of positive persistence, who has had the pleasure of honing his expertise for over 27 years with organizations such as Hilton, Marriott, Time Warner, Federal Express and Dun & Bradstreet. Paul has exceptional expertise in enterprise negotiations, strategic risk management, resource optimization, off shoring, process engineering and optimization and building high performing teams. Paul successfully balances the delivery of leading-edge cost control methodologies while optimizing global compliance objectives to deliver maximum cost reductions while accepting minimum risk exposure. In his current role at Hilton Corporate, his team manages the negotiations and supplier performance of thousands of contract transactions a year.

Mayer Brown's Technology Transactions Practice

- More than 50 lawyers around the world focused on helping clients develop and manage relationships with providers of critical services and technology
- Advised on more than 400 key technology transactions with a total contract value exceeding \$200 billion

Recognized Market Leader



“Band 1” ranking
in IT/Outsourcing for
13 consecutive years (*Chambers* 2004-2016)



Named “MTT Outsourcing Team of the Year”
in 2014 and ranked in the top tier from 2010 through 2016



Ranked as one of the top law firms in 2009
through 2016 on The World's Best Outsourcing Advisors list for The
Global Outsourcing 100™



Named 2016 “Technology Practice Group of the Year”

“They are very good at being able to communicate and synthesize information in a useful and easily understandable way.”
~ *Chambers USA* 2016

“They're very practical in terms of trying to identify solutions and giving very good advice on areas where it's reasonable for us to compromise or, alternatively, where to hold our ground.”
~ *Chambers USA* 2015

“An excellent team of people for outsourcing agreements globally - pragmatic in their approach, with a wealth of experts they can call on.”
~ *Chambers Global* 2014

“Their knowledge in this area is tremendous. They know us so well they blend into our deal teams and become a natural extension to our in-house team.”
~ *Chambers USA* 2014

Hilton Worldwide, Inc.

- Hilton is a leading global hospitality company, with a portfolio of 14 world-class brands comprising more than 4,900 properties with over 800,000 rooms in 104 countries and territories.
- In the nearly 100 years since its founding, Hilton has defined the hospitality industry and established a portfolio of 14 world-class brands, including its flagship Hilton Hotels & Resorts brand, which is the most recognized hotel brand in the world.
- Its premier brand portfolio also includes Waldorf Astoria Hotels & Resorts, Conrad Hotels & Resorts, Canopy by Hilton, Curio - A Collection by Hilton, Tapestry Collection by Hilton, DoubleTree by Hilton, Embassy Suites by Hilton, Hilton Garden Inn, Hampton by Hilton, Tru by Hilton, Homewood Suites by Hilton, Home2 Suites by Hilton and Hilton Grand Vacations.
- Hilton has more than 60 million members in its award-winning customer loyalty program, Hilton Honors.

Agenda

- Overview of Agile Software Development
 - What is it?
 - Waterfall vs. Agile Software Development
- Top 7 Challenges of contracting for Agile Software Development
 - Pricing
 - Termination Rights
 - Assurances the “thing” will be built
 - IP Rights
 - Warranties
 - Client obligations
 - Documentation
- The Sourcing Perspective
- Final Thoughts

Overview: What is Agile Software Development?

- A way to create software based on closer collaboration between developers and business stakeholders.
- Emphasizes frequent delivery of business value by small, self-organizing, cross-functional teams.
- Does not rely on detailed specifications created at the outset.
- Instead, the teams work from a concept of the “thing” to be built and refine as the work progresses.
- Agile is a umbrella term for a set of methods and practices based on The Agile Manifesto.
 - But there are many frameworks to develop software using an agile approach, such as Scrum, Lean, and Extreme Programming (XP).
- Does it produce better results?

Overview: Key Terms

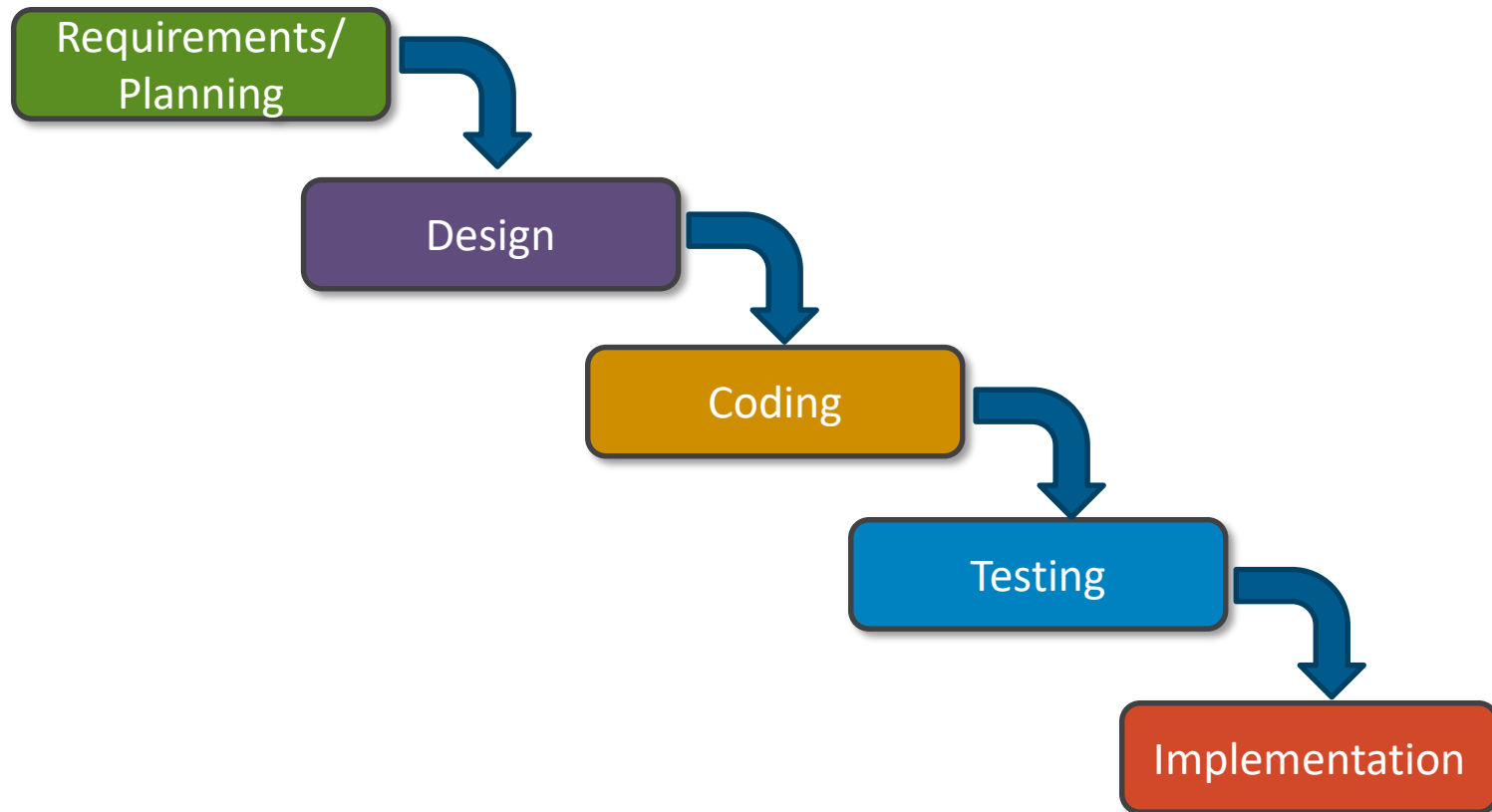
Term	Meaning
Scrum	The most widely-used Agile framework.
Product Owner	The single person on a Scrum team who is responsible for project ROI, has the high-level vision of the product to be built, and has final call on requirements.
Scrum Master	The facilitator on a Scrum development team who removes roadblocks, but has no management authority and does not perform project management functions.
Scrum Development Team	A self-organizing, self-managed cross-functional team responsible for delivering commitments from the Product Backlog.
User Stories	Describe what the end product and its components should accomplish at the end of development. A product will usually have multiple user stories.
Product Backlog	A list of features or technical tasks which the team maintains and which, at a given moment, are known to be necessary and sufficient to complete a project or a release. The Product Backlog can be updated as features and functionality are added or removed.
Sprint (aka Iteration)	A time box of usually 2-4 weeks when the Scrum Dev Team develops items from the Product Backlog.

Polling Question #1

- Which statement best describes your organization's experience with agile software development?
 - a) None, but we are considering it
 - b) Some, but no great impetus to push
 - c) Some, and looking to do more
 - d) The majority of our projects use agile



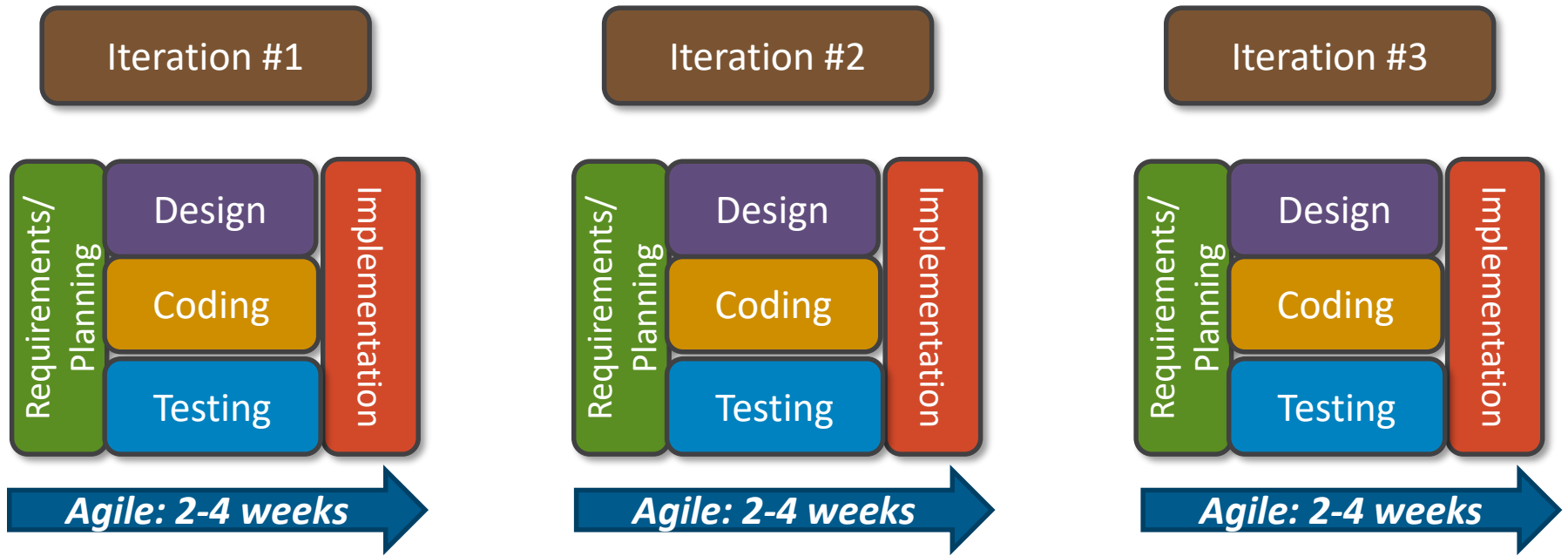
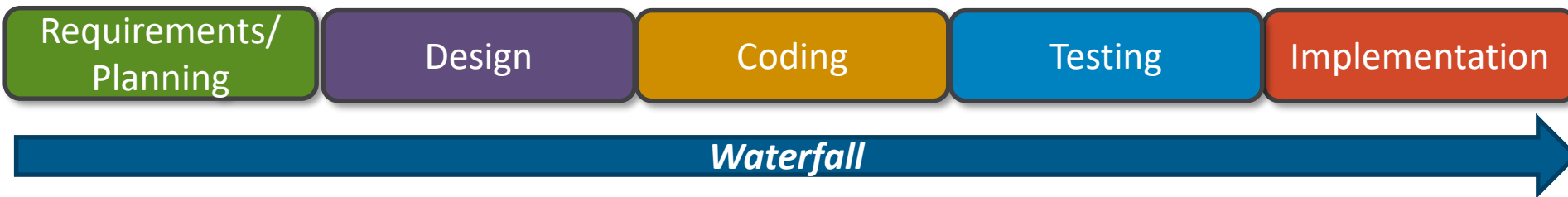
The Waterfall Approach



Criticisms of the Waterfall Approach

- Not appropriate if project requirements are uncertain or fluid.
- Does not promote (and perhaps discourages) creativity during the process.
- Business owner has little interaction with development team after initial specifications are created.
- Problems may not be discovered for a long time (e.g., in testing).
- Client does not receive value until the end of the process.
- Difficulty of specifying all requirements upfront, combined with a rigid change management process and frequent change order and price adjustments leads to disappointment.

Overview of Agile Software Development



Why Do Developers Like Agile?

- The software development process is more fluid, requiring greater interaction between business and technical teams as the project moves through the development life cycle.
- Agile enables software to be developed in continuous cycles based on short iterations, which developers find more efficient and creative (i.e., more quick wins, fewer long slogs).
- Focus is placed on producing working code (fun) and not on documentation and testing (dull).
- The need to test the entire system is minimized since testing (and acceptance) occurs at each iteration.
- More client participation throughout the process.



Making the Shift to Agile

- How do you deal with the client concern that the contractual clarity and upfront planning/milestones under waterfall are absent under agile?
 - Not truly a leap of faith
 - Each party's interests are more aligned; at a well-run scrum meeting, you cannot tell which sides the members are from
- Clients need to have some people trained in an agile methodology
 - But don't need to know how to program; agile brings developers and business teams closer together via iterations/sprints
- Does agile scale for large, mission-critical projects?
 - Yes (e.g., SITA)
 - Some enterprises are moving away from an annual IT project funding process to a quarterly process thereby matching the speed at which software is developed and the agile iterative approach.

Polling Question #2

- If your organization has made the shift to agile software development, which statement best describes the results?
 - a) The results are better and cheaper than waterfall
 - b) The results are better than waterfall, but not cheaper
 - c) The results are not better but cheaper than waterfall
 - d) The results are not conclusive



Contractual Issue #1: Pricing

- **CORE ISSUE**: How do you structure the pricing model to motivate the right behavior and the desired result?
- Waterfall projects: Because the deliverables, requirements and scope of work are defined upfront, the parties can agree to a fixed price, with payment tied to the successful completion of defined milestones.
- Agile projects present greater contracting challenges from a pricing perspective.
- Agile projects are lighter on upfront specifications, and rely more on collaboration and trust.
- As a result, it is difficult for a provider to estimate the required level of effort and commit to a fixed price.
- And, clients are reluctant to agree to an uncertain price for a loosely-defined product.



Summary of Agile Pricing Models

Pricing Model	Pros	Cons
T&M	<ul style="list-style-type: none">• Supports fluid work flow	<ul style="list-style-type: none">• Total project fee uncertain• Increased monitoring• Incentive to rack up hours
Fixed Fee (Entire Project)	<ul style="list-style-type: none">• Fee certainty	<ul style="list-style-type: none">• Difficult to estimate given lack of specs• Scope changes more difficult
Fixed Fee (Per Iteration)	<ul style="list-style-type: none">• Fee certainty (but only for that Iteration)	<ul style="list-style-type: none">• Total project fee uncertain• Continual negotiations

Contractual Issue #1: Effective Pricing Options

- T&M pricing, with bonuses tied to early or on-time completion
- T&M pricing, with hold back of defined percentage (e.g., 25%) of fees until entire project or defined project segments completed
- Commitment that project can be completed by specific date or within defined number of sprints (with free or discounted sprints if more required)
- Pool of development hours for discounted fixed price
- Fixed price per iteration/sprint
- Fixed price for “must-haves”
- Use of milestones/outcome-based contracting to align interests (tied to known outcomes or set on sprint-by-sprint basis)
- Flexible termination rights (but also a risk)



Contractual Issue #1: Use of Outcomes in T&M

- Outcome-based contracting via milestones can be used under agile T&M projects; for example,
 - Create a milestone tied to payment when your application successfully connects to Google Maps
 - Contract for a certain outcome by the 3rd sprint, but provide an incentive if completed by the 2nd
- Three options to pay T&M upon milestone completion:
 - Pay all T&M charges upon completion
 - Holdback a defined percentage until completed
 - Pay T&M weekly, but contract for a bonus mechanism weighted for early delivery
- In the end, trusted, frequent partners may offer the greatest assurances
 - Highly motivated to perform well and keep price in check

Contractual Issue #2: Termination Rights

- T&M is palatable under Agile Software Development due to more relaxed termination rights.
 - Remember, the goal of each agile iteration is to produce workable code.
- The typical Agile Software Development agreement allows the client to terminate at the end of each iteration with no termination charges.
 - If the client does not see value, it can walk away.
- No termination charges due to lack of future requirements, so bench costs should be minimal.
 - But a client should weigh stranded/bench costs against the need for developer personnel continuity.



Contractual Issue #2: Termination Rights (cont'd)

- To fully take advantage of this termination right, the client should contract for other protections such as:
 - Limiting the developer to only use tools and code that the client can license from third parties or the developer; and
 - Commitments from the developer to conduct knowledge transfer.
 - **Risk:** *Agile Software Development involves minimal software documentation, so restarting a terminated agile project may be more costly since new developers will need to get up to speed.*
- Developers know switching costs are high and will try to lock in clients throughout the project.



Contractual Issue #3: Delivery Commitments

- *Risk: Lack of clearly-defined specs and easier termination rights jeopardize final delivery of the “thing.”*
- An agile project begins with a high-level concept of the product to be developed (the “product vision”).
- The product vision is used as a guide to create “user stories” and eventually the “product backlog” - a list of items to be developed during the project.
- The parties decide on prioritization of items from the product backlog and define what the successful completion of each iteration means.
- The lack of milestones and continual re-assessment allows the parties to adjust on the fly; the final product may be very different than what was originally envisioned.

Contractual Issue #3: Delivery Commitments (cont'd)

- Developers will offer fixed fee for agile projects ...
 - But may add a risk premium to deal with uncertainty and ask for more money upfront to understand the unknowns;
 - And will seek to include a light-weight change control mechanism to modify the price as it learns more.
- Use the agile process to reduce the “cone of uncertainty” to plan and contract for outcomes:
 - Each party needs to understand the broad business outcomes and divide into smaller projects;
 - The agile team should then create high level specs for each project (“epics”) and then define the technical architecture;
 - With this information, the developer can provide “indicative” pricing; the client can then contract for “not to exceed” or fixed-fee pricing for these outcomes to be further refined as code is developed.

Polling Question #3

- Agile projects often require more collaboration between the customer and developer. Which statement best describes your organization's view on deliverable ownership?
 - a) We own all the IP
 - b) The developer owns all the IP but we receive a license
 - c) Ownership is determined on a case-by-case basis



Contractual Issue #4: IP Rights

- More complicated due to client/developer collaboration.
 - Scrum masters should create a journal with notes on all code and ownership rights
 - Need to track daily and close out at the end of each iteration/sprint
- At a minimum, the client should have an unrestricted license to the developed product (including pre-existing materials brought by developer).
- Also, potentially seek restrictions on the developer's use of proprietary ideas contributed by the client.



Contractual Issue #5: Warranties

- Lack of project specs, but the developer could warrant that the working code produced during each iteration meets the specs for that iteration.
- As more working code is built in later iterations, include warranties that (i) the integrated pieces will work together and (ii) the entire product will perform in accordance with the summation of the specifications from each iteration.



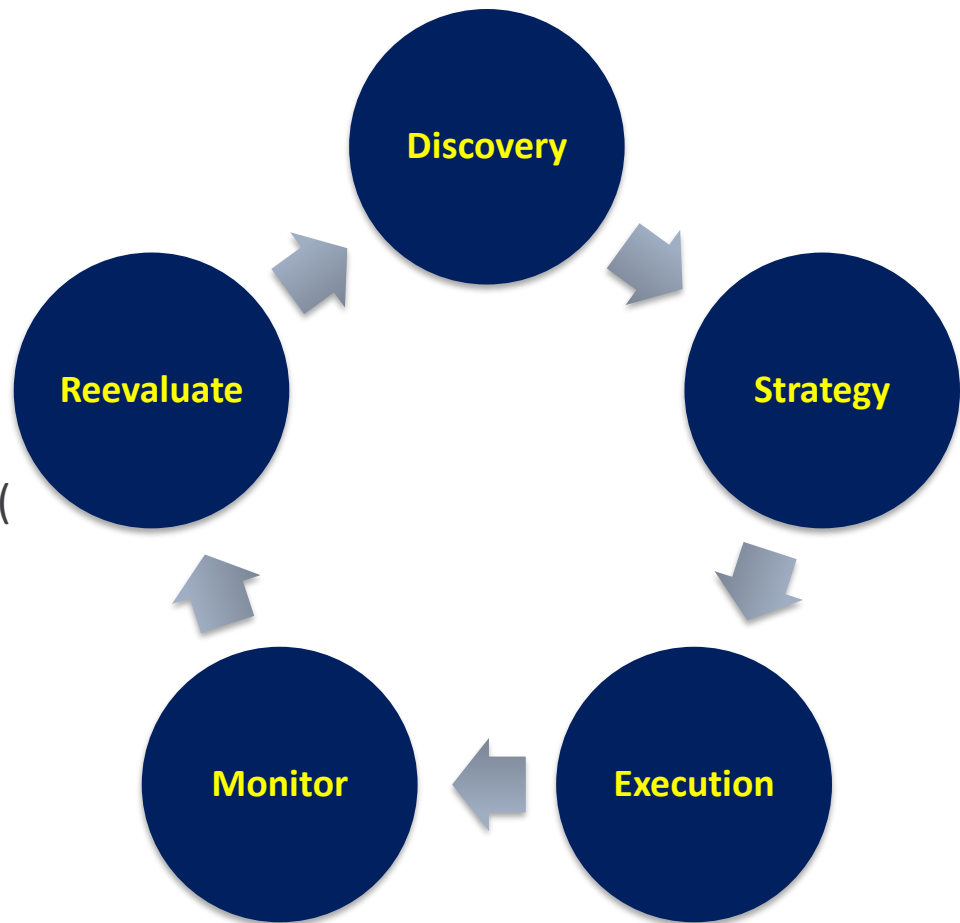
Contractual Issues #6 & 7: Other Concerns

- Client Obligations:
 - Increased collaboration with the client throughout an agile project increases the probability that the developer can blame the client for a failed project.
 - E.g., weak/inexperienced scrum leader or the client tries to manage the development like a waterfall project.
- Sufficiency of Documentation:
 - Agile prioritizes working code over all else, which means deliverables like documentation may be less than what the client is accustomed to under a waterfall approach.
 - Therefore, include a contractual provision that commits to a certain level of documentation detail and quality.

It's Agile's Fault that You have to Change

The Sourcing Process as we know it

1. **Discovery**: Gather the appropriate data to feed the project requirements, market intel, vendor selection, financial and spend.
2. **Strategy**: Leverage the discovery data to align Customer, Technology and Sourcing strategies.
3. **Execution**: Initiate the Sourcing strategies(Negotiation, RFX, SOW, Dispute...)
4. **Monitor**: Track performance to SLA's, Services, Delivery and Compliance.
5. **Re-evaluate**: Customer, Technology and Sourcing review supplier performance and strategy success



So What's the Problem?

- Trust is a huge component of Agile. To be successful you have to trust your supplier and your client.
- Agile is an “iterative” process that requires regular engagement. The traditional sourcing process is almost polar opposite. Fewer touches = Faster.
- Classical deliverables are replaced with Success Stories and Demos. The stories may change as the team works through the streams. This adds a new wrinkle to a sourcing delivery team.
- Disputes are significantly harder to resolve. By design:
 - Deliverables are less finite
 - The customer is more likely to be complicit in delivery issues
 - Suppliers are going to offer weaker delivery warranties
 - Acceptance language be your new friend but only if it is drafted specific to Agile impacts.
- In Summary, sourcing teams will need react to Agile by;
 - Adding iterative engagement to their processes
 - Restructuring their agreements
 - Take leadership in selecting trustworthy suppliers and clients

Discovery: Bad Decisions, *do not* make Good Stories any more.

- In Waterfall, success or failure depended on strong, detailed requirements. The client was almost completely responsible for requirement definition.
- In Agile the Story is the driving force. For that story to be successful, there needs to be more early engagement by not only the customer, but technology and sourcing.
- More rigor needs to be focused on defining success and the structure and out put of the Demo's.

Strategy: Best time to repair the roof is before it rains.

- Teams need to be more critical of their Strategies. Can the project cost/risk profile accept Agile to begin with?
- Supplier and Customer Selection is key to success. Sourcing must fully accept its leadership position.
- Design points in the Strategies to review progress with go-no go decisions.
- How will the team know that a Demo will be Acceptable?

Execution: Same war, different weapons.

- Legal artifacts will need to change to match Agile structure. Most notably MSA, SOW's, NDA and RFP's.
- Negotiations need to shift to focus on:
 - Retaining delivery Reps & Warranties
 - Re-enforcing Acceptance language
 - Give more structure to Stories and Demo's
 - Fix pricing what services you can
 - Require that the supplier document the Stories and Demo's as you progress
- Do not be afraid to call the supplier out. In Agile the client is the one accepting greater contractual & commercial risk.

Monitor: No one likes a know it all, especially when they are right.

- Sourcing must engage the project proactively and iteratively. If you wait for failure to engage you will lose the dispute before it starts. Changes to consider:
 - Join the Demo meetings with a Success Check List
 - Identify that sourcing will give the formal Acceptance for Demo's, Documentation and any other deliverables
 - SLA's may be very difficult to create or leverage. However, well defined Step-in rights may be extremely useful.
 - Consider that sourcing may have to be the one who calls the project dead. Keep alternatives available and get used to the fact that no one is ever going to like you!

Re-evaluate: Did we win?

- Use the Strategies from the Strategy Step to evaluate overall cost and delivery success.
- Rate suppliers performance for future use or to avoid use.
- Rate customer performance and determine if they should use Agile in the future.
- Create a repository for the project documentation and Demo's

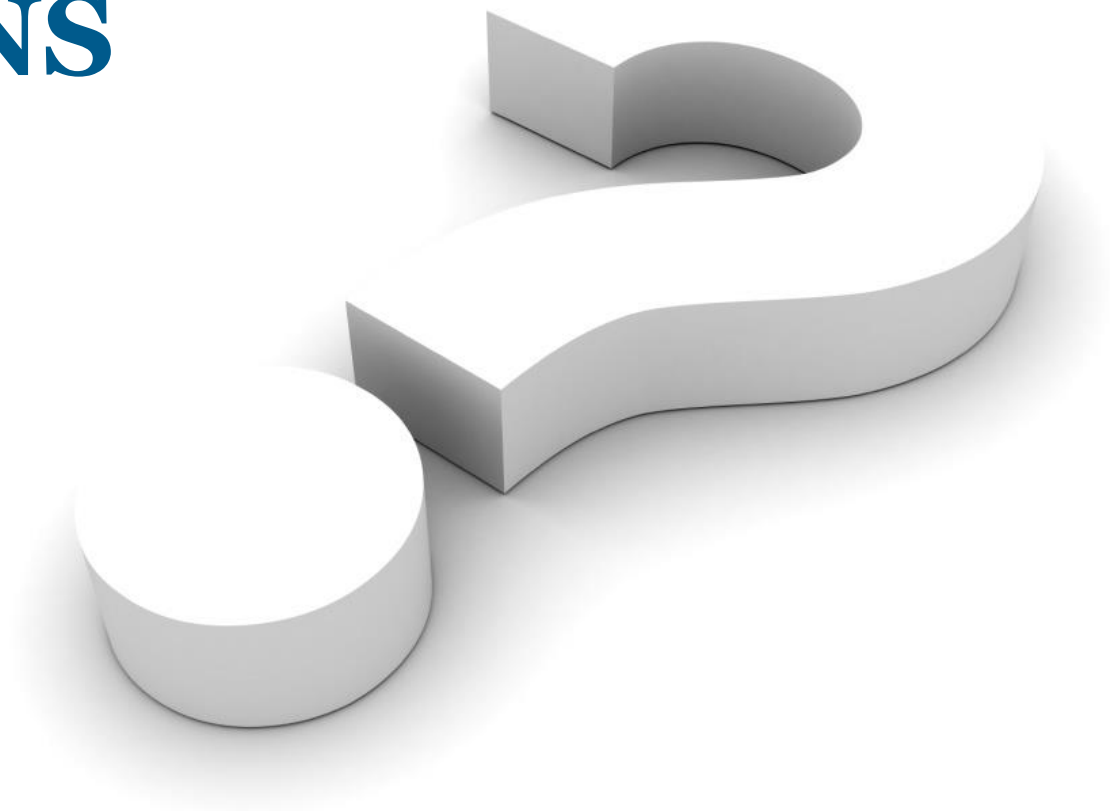
Summary:

- Trust is earned
- Be a leader
- Be ready to engage in each work stream of a project
- Know success and you can limit your exposure to dispute
- Get your legal teams engaged and educated. Your documents will need to be adjusted to consider Agile

Final Thoughts

- Agile Software Development is the future.
- Contracting for software development projects using agile is not as simple as a waterfall approach.
- However, contract levers exist to motivate the right behavior under agile projects.

QUESTIONS



Resources

- Agile manifesto *available at* <http://www.agilemanifesto.org>
- CIO.com “How to contract for outsourcing agile development” *available at* <http://www.cio.com/article/3090569/outsourcing/how-to-contract-for-outsourcing-agile-development.html>
- Law360: “Agile Software Development Brings New Contracting Issues” *available at* <http://www.law360.com/articles/809900/agile-software-development-brings-new-contracting-issues>
- Agile Contracts: Creating and Managing Successful Projects with Scrum, Opelt *et al* (ISBN: 978-1118630945)

Thank You!

Derek Schaffner

Counsel

Mayer Brown

+1 202 263 3732

dschaffner@mayerbrown.com

Paul Lanigan

Technology Sourcing Executive

Hilton Worldwide, Inc.

+1 703 883 1087

paul.lanigan@hilton.com

Evaluation How-to:

Why?

- Your feedback **drives SIG Event content**
- By signing and submitting your evaluation, you are **automatically entered into a prize drawing**

How?

Option 1: App

1. Select **Schedule**
2. Select Schedule **by Day**
3. Select **Day**
4. Select **Session**
5. Scroll to **Description**
6. Click on the **Evaluation link**

Option 2: Browser

1. Go to www.sig.org/eval
2. Select **Session (#13)**

COMPLETE &
SUBMIT EVAL

Session #13

www.sig.org/eval

Contracting for Agile Software Development

Derek Schaffner
Counsel
Mayer Brown
+1 202 263 3732
dschaffner@mayerbrown.com

Paul Lanigan
Technology Sourcing Executive
Hilton Worldwide, Inc.
+1 703 883 1087
paul.lanigan@hilton.com



SIG

Global
Executive
Summit

retweet

retweet

tweet

tweet

retweet



Are You a Winner?

Engage with SIG on **Twitter**
during the Summit!

Follow **@SIGinsights**

Mention **#SIGspring17**
in your tweets!

All **Tweeters** will be
entered to win a
\$50 Gift Card.

*The winner will be announced
Thursday, March 16, 2017*