Adventures in Agile Contracting: Evolving from Time and Materials to Fixed Price, Fixed Scope Contracts

Teresa Franklin
Info Tech, Inc.
Teresa.Franklin@infotechfl.com

Abstract

For Info Tech, Inc., the transition to Agile practices three years ago lead to a change in the way we collaborate and contract with our long term, mostly government customers. Our evolution covered Time and Materials (T&M) contracts, a Hybrid of T&M and Fixed Price and most recently a Fixed Price, Fixed Scope, Fixed Schedule contract that supports Agile development. Our goal has been to meet our customer's needs for predictable results while maintaining our commitment to agile practices. Key success factors include developing a responsive contract modification process that allows for quick change management and identifying non-tangible scope deliverables beyond the software components.

1. Introduction

Info Tech, Inc. was founded in the 1970s initially focusing on statistical and economic consulting services for the Florida Attorney General, developing the first computerized methods for detecting collusive behavior in sealed bid markets. Over the decades since, Info Tech developed additional software solutions to meet the needs of the highway transportation industry at the federal, state and local levels. For the most part, the projects that lead to the completion of software followed the waterfall method, even when requirements were not well defined up front.

From the contracting perspective, services such as maintenance and customer support are provided on a T&M basis whereas enhancements and new development have mainly been firm fixed price contracts with payment upon delivery of milestones like System Requirements Specifications (SRS) early in the project and Alpha Test Results towards end of project. We learned over time that delivering on original requirements does not always lead to high quality results based on usability. We joined the Agile revolution in order to improve our engineering and testing practices, increase collaboration with the user community and produce software that meets their needs.

The overall initiative was and still is to rewrite an existing muti-module client/server system in use for 10 to 25 years, depending on the module, by more than 45 different departments of transportation. Yes, generic system yet highly customizable. The biggest challenge to the development team was delivering acceptable and expected results by the set time, despite the number of firsts:

- First time using Agile collaboration practices with customer giving them visibility and opportunity to improve system
- First time using Agile test driven development and estimation with a large developer team (14 now 25 and growing)
- First time using Web development tool at huge scale of system infrastructure and functionality (enterprise wide database on multiple platforms with complex data validation and algorithms, hundreds of customizable reports and screens)
- First time rewriting the modules with the goal of unified data model, architecture, and business rules where there are dependencies with other modules in the product suite to be developed in later phases yet aligned and integrated with the overall system.

The initial 2-year project had several triumphs considering the daunting endeavor but we did not meet customer expectations. To address customer concerns with not meeting cost, schedule and scope requirements, the administrative and contractual aspects along the way have evolved from T&M model to a fixed milestones model.

We have ultimately worked well with the customer to overcome the challenges of Agile development in their world of fixed constraints while reaping the benefits Agile brings to the resulting software quality.

978-0-7695-3321-6/08 $25.00 © 2008 IEEE
DOI 10.1109/Agile.2008.88

2. The Cultural Shift: Transitioning from Waterfall to Agile in Consulting Services

In a company with close to 30 years of software development and consulting services, the evolution from existing waterfall methodology to the more iterative and collaborative methods of agile required a major shift in corporate culture and team dynamics. The transition took place in multiple overlapping stages.

Stage Zero: Waterfall, Fixed Constraints

Serialized analysis, design, engineering then system testing is the method used for more than 25 years and is still used in cases where scope is small, requirements are known and the risks are minimal. With Waterfall, the software met customer needs for predictable scope, cost and end date but the results did not always meet customer requirements for usability and return on investment (ROI). Interaction with the customers was mainly at the start of the project to document the requirements and at the end of the project during beta testing prior to production. By then it was too late to address scope deficits within the fiscal contract year. Contract modifications were typically related to small extensions in schedule we requested in order to complete testing and error resolution. Waterfall practices often lead to years of ongoing maintenance and enhancement costs to address deficits in requirements and design discovered after the software was in the hands of the users. The need for Agile practices was obvious.

Stage One: Agile, Time and Materials Contracting

After exposure to and successful use of Agile practices on small scale projects in 2004 we lead the initiative with our main customer, and software owner, to adopt the standard practices of Agile contracting: ongoing user collaboration and exposure to emerging software in a T&M funding model within overall scope, cost and schedule objectives. All labor was paid for per sprint based on actual time spent by all staff working on the project at set rate per hour by level of their position. All direct material costs were reimbursed as they occurred, for example hardware and software purchased for specific use on the project.

Stage Two: Agile, Time and Materials Contracting, Adding Fixed Price Milestones

With close to two years of Agile development under our belts and the need to negotiate a new fiscal year contract to complete the initial phase of the software rewrite, the customer presented concerns about continuing the T&M model in light of the expended cost and time without sufficiently complete functionality. With the direction to add fixed price milestones during the project we created the concept of a hybrid (mixed T&M and fixed constraints) contract. A percentage of average per sprint labor cost was allocated to a fixed price per sprint deliverable, paid upon customer acceptance of the deliverable.

Stage Three: Agile, with Fixed triple constraints of scope, schedule and cost

Nearing three years of development and the need to renew the annual fiscal year contract, we had made significant progress with software infrastructure to support the current and future modules, but several features for the initial module were still pending completion. We were refining requirements, supporting users with hands on use of preproduction software, and collaboration with a dedicated customer review team.

With the direction to go to full fixed scope, cost and schedule for the remainder of the initial project phase, we created two main types of deliverables: 1) per sprint development and testing results deliverable that had a payment associated with it upon customer approval and 2) per release deliverable where one pre-production release was packaged and delivered to customers several sprints into the fiscal year, upon approval of the product owner team. The contract also had associated fixed scope (backlog items) to be completed during the fiscal year unless approved by the product owner team.

3. Stages of Evolution in Agile Contracting

3.1. Agile Contracting in Time and Materials

From the inception of the development initiative, the team committed to and followed standard agile practices.

The T&M model allowed the customer the flexibility to adjust scope priorities each month (sprint) with ongoing discovery, improvements and the risk of rewriting software on a completely new development technology. The customer agreed to the T&M contract that conveyed set time frame using a team that would have a predictable burn rate to accomplish scope objectives. Significant progress was made on functionality and infrastructure however customer expectations on cost and schedule were not met.

3.1.1. Customer Collaboration and Expectations Management. The distributed and remote team of product owners was in an oversight role each sprint as the onsite Business Manager (Info Tech, Inc. representative of the customer product owner) and subject matter experts worked hand in hand to create

user stories, associated backlog items and acceptance tests, refining requirements and design with each sprint towards demonstrable results.

With scope and priorities refined each sprint in parallel with ongoing design optimization (refactoring) came the perception of variable cost and schedule to complete expected scope. Where we understood the fixed cost and schedule within the fiscal year parameters, having fixed scope completed at the level of the entire system was not feasible in the original contract due to the size of scope which increased once requirements were fully established. The customers provided feature prioritization to allow focus on the important functionality per sprint.

3.1.2. Management and Contracting Processes. Payments were based on effort per sprint with annual assessment and changes to contract based on projected remaining scope. The deliverables were fixed based on number of points the team could accomplish each sprint and the resulting software pushed out to an internet preview site that the customers have hands on access to for training, testing, and feedback to resolve issues as soon as possible in the upcoming sprints.

3.1.3. Benefits and Lessons Learned from T&M Contract Model. The full commitment to Agile practices produced optimal code design. Scope was allowed to change in short turn around time with less management overhead. However, with the length of time and cost that the project was taking to complete, we learned that we needed more experience in backlog item estimation, in managing scope creep created by the discovery process and in addressing customer perception of insufficient quality during development when software is exposed before completed.

3.2. Agile Contracting in Time & Materials/ Fixed Constraints Hybrid

The goal, and challenge, with this hybrid model was finding milestones that mapped to tangible scope deliverables without impeding Agile scope discovery and flexibility to change. The customer established a new product owner team, still remote but comprised of knowledgeable users from several different state agencies. The collaboration and momentum improved as did the testing and validation of results each sprint.

3.2.1. Customer Expectations Management. Customer collaboration focused on allowing Agile practices to continue while producing agreed upon deliverables. Weekly webinar sessions were held to prioritize the remaining backlog items and assess newly reported items as errors, enhancements or pending completion. End of sprint review sessions allowed demonstration of sprint results and payment of the fixed price associated with the sprint milestone. The scope for that sprint was agreed upon at the beginning of the sprint during the planning session with the product owner team. Any changes to sprint backlog were documented and agreed upon during weekly meeting except for changes made in the last week of the sprint which were documented by the team and reported on during the sprint review webinar. Changes during the sprint typically included insertion of high priority error fixes or system improvements related to performance.

3.2.2. Management and contracting processes for the hybrid contract model. While similar to T&M approach in that the payments were based on monthly increments, the hybrid model had a total contract cost was now based on fiscal year (12 months) with T&M portion as the largest percentage of the contract and the fixed price milestone payments as the smaller percentage (less than 25%) of the contract. A risk factor was applied to the fixed price payments in the event scope exceeded the agreed upon complexity or the approval process took an extended period of time.

3.2.3. Benefits and Lessons Learned from Hybrid Contract Model. We improved focus on minimizing requirements to what is needed. The downside to the inflexibility of scope, schedule and cost lead however to a reduction in the team being free to do ongoing design optimization per sprint. The customer was faced with prioritizing feature content over code optimization in order to align with fixed schedule and cost.

3.3. Agile Contracting in Fixed Constraints

With the full fixed price, fixed scope and fixed schedule expectations set on the latest fiscal year contract, the focus on constraining scope is at its highest of the three Agile contracting models. As a contractor we receive no payment for work until deliverables are approved by the customer. The fiscal year has set list of backlog items to complete and each sprint a portion of those items are selected, developed, tested and demonstrated to the product owner team for approval and payment. Any issues seen as errors are grounds for disapproval of the deliverable and no payment until all issues are resolved to the satisfaction of the product owner team. The risk of disapproval is high in sprint where portions of functionality are started with the remained of that functionality to be completed in future sprint(s). It is key to document and communicate these situations so that the sprints are approved on their own merits and not on the expectations of the overall feature set.

3.3.1. Customer Expectations Management. Customer Collaboration and Expectations Management focuses on more detailed review of product backlog items and requirements up front to develop contract that fits within fixed triple constraints. User stories are reviewed and broken down into children that together comprise the full scope of the parent backlog item.

3.3.2. Management and Contracting Processes. Payments in the fixed constraints model are based on accepted scope deliverables per sprint. Fixed constraints require additional time and management resources to address contract modification processes throughout the project. Done criteria (data model, data validation, documentation, technical infrastructure, work flow dependencies, performance, usability, etc.) are all assessed now as part of the estimation process where before they were addressed but not properly funded or scheduled, hence we went beyond cost and time frames in previous fiscal years.

3.3.3. Benefits and Lessons Learned from Contract Model. We are completing functionality faster but with less flexibility in the Agile practices used to optimize code design and infrastructure. This may lead to more maintenance and enhancement effort in future development work but in the current projects the customers are getting production software that meets their expectations and needs. We learned to improve team and customer awareness of non-tangible deliverables that may not impact the development team and product backlog burn down chart. For example we created a project backlog that supports and compliments the product backlog so that schedule and cost is based on full contract expectation. These non product development expectations include but are not limited to:

- Beta testing site mobilization, training and support
- Project reporting and metrics upon request to executive management
- Detailed meeting minutes and action item tracking from weekly webinars with product owner team
- Maintenance and support of internet software preview site and bulletin board forum accessible by entire user base
- Administrative tasks related to contract modifications to address significant changes in scope (greater than 15% of base) or schedule (more than one sprint beyond fiscal year) or cost (payment for scope beyond the 15% contingency above base)

Non product development items that are not addressed by the team but are a part of the administrative tasks for example, are captured in a separate backlog from the product backlog and addressed by extended team members. For now this "Chicken" log works to ensure all project steps are addressed but we are hopeful to integrate all backlogs related to the product and the project into the same tool to allow an overall view. We are also adhering better to minimal requirements as an ongoing goal.

3.4. What Lies Ahead

3.4.1. Continuing Evolution. In concert with the product owner team we perform ongoing assessments of results and integration of process improvements to address any impediments to success. We also have additional phases of work ahead to integrate into the system. The plan is to write as many user stories as possible for the must have features while listing nice to have features as additional scope if time and funding permits. We do not anticipate a change in the payment model back to T&M in the near future so the key is to mitigate the risk of schedule and scope expansion which will impact the profit margin built into the fixed price. Where user stories are not possible prior to estimation and contract negotiation, the main process is to create feature buckets with set number of points that map to team capacity and we develop the features that we can within that set number of points, stopping to ensure we have points left for final testing, packaging and delivery.

3.4.2. What We Need and No More. We are striving to keep processes streamlined and supportive of the fundamentals of Agile. The key to customer ROI is acceptance of software they can use, not of software that meets original documented requirements. To address cost and schedule constraints, keep the features streamlined to what is needed. This also keeps the complexity of the system to a minimum thereby reducing training costs.

4. Conclusion

Agile development practices can be successful under contracts with fixed constraints of scope, schedule and cost. The ideal approach is to develop software under a T&M model that allows scope flexibility with emergent design as you deliver software increments. That is not a feasible for many customers, especially those in the government sector with well established rules of waterfall development and fixed scope agreed upon up front. The reality of fixed scope by a desired completion date within available funding has to be addressed to meet customer needs.

The key to success is first hand collaboration with members of the user community during the proposal and contract development process to ensure enough user story awareness to estimate the magnitude of scope themes and define tangible milestones that allow payment on delivery of theme components at each sprint cycle as well as larger milestones as releasable software deliverables.

Expectations must be clearly documented and managed along the way with open and thorough communication with the product owner(s) so that the discovery process is seen as a positive and welcomed part of the project towards highest ROI and usability. The number of scope points per backlog item can go above or below the contract estimate as long as the overall scope points for all requirements stays within an agreed upon total number of points. The magnitude of feature complexity must remain within "what we need and no more" in order to keep scope creep and contract modifications at a minimum. Build in contingency buckets for both schedule and scope changes to address the unknowns. At the time of contract negotiation and throughout the project, you don't know what you don't know, respect that and don't be afraid to admit it.

5. Acknowledgements

Appreciation and recognition to Info Tech, Inc. pioneers and supporters of agile development, Bob DeHoff, Tom Rothrock and countless others. Special thanks to Tamara Sulaiman of Applied Scrum, LLC and Uncle Bob Martin of Object Mentor for their creative input and mentoring as Info Tech, Inc. evolved into a service provider of Agile contracts. Most of all, thanks and acknowledgement to our long term customer for stepping out of the box and allowing Agile development to replace Waterfall and meet their needs. The evolution continues.