Progressive Acquisition and the RUP
Part IV: Choosing a Form and Type of Contract

by **R. Max Wideman**
Project Management Consultant
AEW Services

In *Part I* of this series, we identified the gap between the expectations of traditional procurement specialists and the realistic needs of the software development community and introduced a new "progressive acquisition" approach that can help bridge this gap.

Part II provided a high-level description of how to modify the traditional contracting process to fit a progressive acquisition model that meets the needs of both acquirers and suppliers in a simplified scenario. We walked through the process of obtaining a system, software product, or software service,[1] through legal contract[2] from an independent supplier.

In *Part III* we began looking at what actually goes into a contract, examining basic elements required for an effective contract and hurdles that tend to get in the way of constructing such a contract. We also looked at specific content required for the contracting approach we suggested in Part II, and the reasons most companies use a centralized acquisitions approach. Now, in Part IV, we will describe the variables that govern contract formulation[3] and then discuss how to choose the best form and type of contract to accommodate particular contract conditions.

Variables Involved in Forming a Contract

As we've noted in previous articles, the notion of contracting -- the acquisition of goods and services through contract -- is a very flexible one. A contract can be devised to reflect requirements of any number of variables, and the acquirer's contracting strategy should be designed to optimize overall project results with regard to associated risks.

To determine the best type of contract for your project, you will want to consider the ten variables we discuss below. They are all interdependent to a greater or lesser extent, and all can be depicted on a continuum. In the figures below, the left side of the continuum represents the best interests of the acquirer; the right side represents the supplier's best interests.

Variable One: Product Type

As shown in Figure 1, the product type may range from standard off-the-shelf software to off-the-shelf software with some degree of customization (i.e., "COTS") to fully developed software. Standard "off-the-shelf" software is normally obtained by direct purchase order rather than through contract, since less risk is involved. The more customized the software, the greater the risk; this risk should be managed by adopting strategies recommended in the Rational Unified Process,® or RUP®.
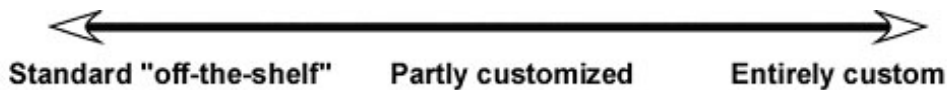
Standard "off-the-shelf"   Partly customized   Entirely custom

Figure 1: Product Type

Variable Two: Product Scope of Work

Figure 2 shows the range of requirements definition, which determines the degree of certainty for the scope of work and the extent and timing of expected changes. The better defined the requirements, the lower the risks to both acquirer and supplier, and the greater the opportunity to establish fixed time and cost parameters.

Well defined requirements   Uncertain requirements

Figure 2: Product Scope of Work

Variable Three: Product Safety and Liability

Figure 3 reflects the extent to which the software product will be used in a safety-regulated environment and the liability in case of failure. For example, an airplane traffic controller system has a much higher safety requirement -- and therefore needs more thorough testing -- than, say, a stock control system. The higher the concern about risk of failure, the higher the level of integrity required. This will be reflected in cost and schedule increases.

Relatively low concern   High concern

Figure 3: Product Safety and Liability

Variable Four: Acquirer's Level of Control

Figure 4 shows the extent to which the acquirer wishes to exercise control over the software development work. Typically, for a fixed-price contract, the acquirer's control is minimal; if they try to exert more control, they risk a contract claim on the supplier's part for "interference." Fixed-price contracts place the risk of software development on the shoulders of the supplier. To mitigate this risk, the supplier must either ensure that the requirements are well defined or insist upon a "time-and-materials" basis for compensation. With progressive acquisition, you can begin with a time-and-materials approach for early phases of the project and then agree on progressively firmer terms for time and costs as the project progresses. This represents an essential advantage over traditional fixed-price contracts.
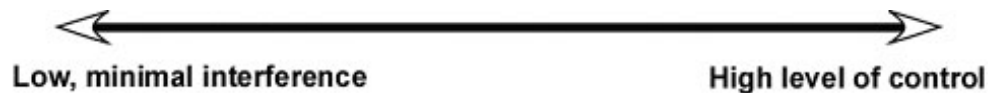
Low, minimal interference ⟷ High level of control

Figure 4: Acquirer's Level of Control

Variable Five: Supplier's Level of Control

Figure 5 displays the converse of Figure 4: the extent to which the supplier has control over the development process, and hence responsibility for the product's performance. As we have noted, the supplier has primary control in a fixed-price scenario; but the supplier's control (and desire for control) is minimal in the case of a "cost-plus" or "time and material" form of contract. These types of contracts place the risk and responsibility on the acquirer's shoulders.
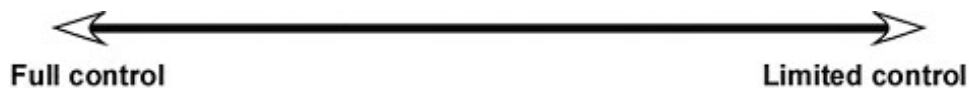
Full control ⟷ Limited control

Figure 5: Supplier's Level of Control

Variable Six: Acquirer's Involvement in Quality Control

Figure 6 shows the range of options for the acquirer's involvement in quality control. In a fixed-price contract, quality control rests almost entirely with the supplier, although inspection and testing may be conditions for interim progress payments.
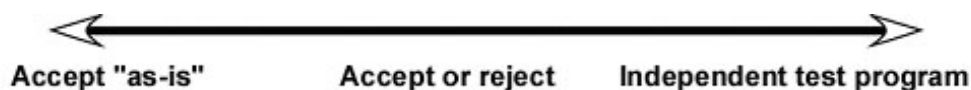
Accept "as-is"   Accept or reject   Independent test program

Figure 6: Acquirer's Involvement in Quality Control

Variable Seven: Timing for Delivery and Pace of Work

Figure 7 shows the practical range of delivery timing. If the target delivery date is earlier than the normal pace of work would allow, then the work must be accelerated by some means that incurs additional costs, such as overtime or extra resources. This may also entail additional risk because it is more difficult to coordinate the work.

As soon as possible   Optimum for cost-effectiveness

Figure 7: Timing for Delivery and Pace of Work

Variable Eight: Form of Compensation

A fixed price, as suggested in Figure 8, appears to be in the best interests of the acquirer. However, if the conditions of such a contract are not met, then the contracted work may well end up being more costly, taking longer, and even ending in failure. The software industry is rife with examples of fixed-price contracts that worked to the detriment of both acquirer and supplier. For this reason, parties often adopt a modified form of fixed-price contract with provision for scope variations and with or without incentives. Time and materials compensation is most appropriate if the extent of the requirements is either not yet known or highly uncertain. As we noted in discussing Variable Four, Acquirer's Level of Control, progressive acquisition allows you to use a time and materials approach up front when there is great uncertainty, and then firm up terms for time and cost later on, when the risk of doing so is lower for both sides.
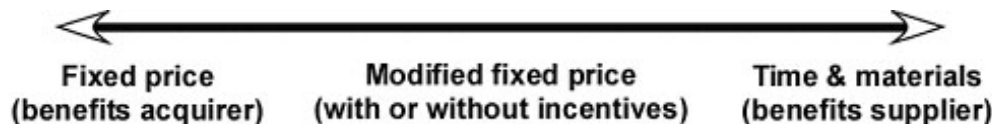
Fixed price   Modified fixed price   Time & materials
(benefits acquirer) (with or without incentives) (benefits supplier)

Figure 8: Form of Compensation

Variable Nine: Supplier's Cost-Risk Position

Figure 9 reflects the supplier's cost-risk position vis a vis customization. Supplying standard, off-the-shelf software provides the lowest-cost, safest, and quickest return to the supplier. However, standard software may or may not meet the acquirer's needs and typically requires some degree of customizing. Even so, COTS is usually less risky to both parties than an all-custom developed solution.
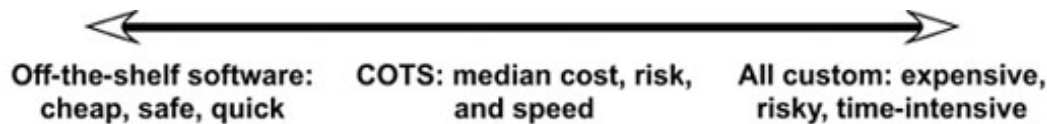
Off-the-shelf software: COTS: median cost, risk,   All custom: expensive,
cheap, safe, quick   and speed   risky, time-intensive

Figure 9: Supplier's Cost-Risk Position

Variable 10: Cost-Risk for Acquirer versus Supplier

Figure 10 broadly summarizes the respective positions of acquirer and supplier for all nine variables we have just discussed: The acquirer's cost-risk is inversely proportional to the supplier's risk. As we have seen, the degree of technological risk is the driving concern for both parties, especially in the case of fully custom-built software. The best contractual relationship should be heavily dependent upon the nature of the technology being acquired, how much is known about the solution, and how the parties want to allocate risk. In principle, each specific risk should be allocated to the party best able to control that risk.

Acquirer's risk relatively low   Acquirer's risk relatively high
(high to supplier)   (low to supplier)

Figure 10: Cost-Risk for Acquirer versus Supplier

Selecting the Optimum Form and Type for Your Contract

Because contracting involves so many possible variables, it is important to select a document form -- and the type of document -- that can reflect both parties' choices regarding these variables. A contract document that does not properly reflect the intent between the parties, or one that is unnecessarily verbose or arcane, is a source of unnecessary risk to both sides.

Contract Form

Many industries and jurisdictions have well-established standard contract documents that suit various typical circumstances. As the language in such documents is usually well understood by those in the industry and has probably been interpreted by the courts, using these documents can help you lessen the likelihood of misunderstandings between acquirer and supplier.

A standard purchase form order will suffice for orders of standard off-the-shelf software or units of software licenses with little or no customization, providing that payment is based on a fixed price or quantity-based unit prices. A more comprehensive, more individualized document is required if any significant amount of customization work is involved; the manner of

payment must also reflect the greater amount of work. Figure 11 illustrates this progression.
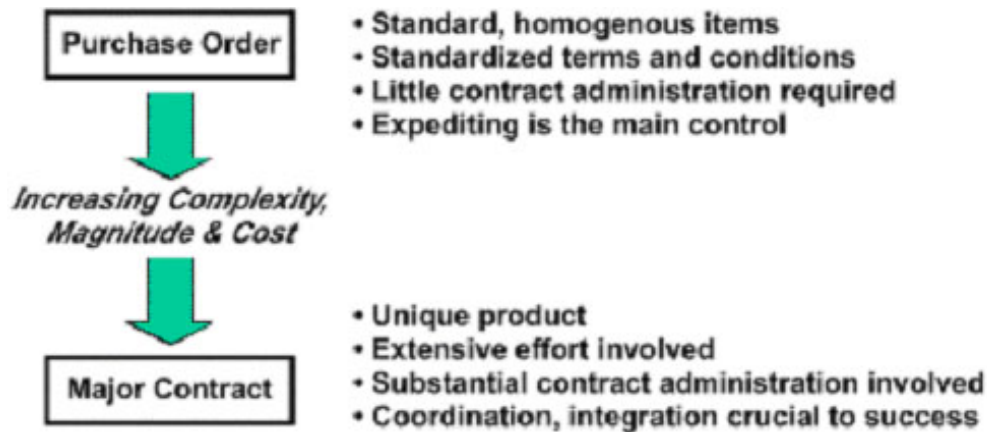

Figure 11: Determining the Best Form for a Contract

Contract Type

Contract type refers to the commercial terms of the contract -- that is, the basis for payment. Given the nature of software products, two considerations govern the selection of contract document type:

1. The degree of responsibility the supplier assumes for the actual costs of performance.

2. The profit incentive the acquirer offers to the supplier for achieving or exceeding the specified goals, standards, or targets.

The range of payment options is illustrated in Figure 12.

	Fixed Price	**Cost Reimbursable**
Form of Payment	- Firm lump sum - Firm unit prices	- Cost plus percentage of costs - Cost plus a fixed fee
Form of Incentive	Fixed price plus an incentive fee	Cost plus incentive fee

Figure 12: Range of Contract Payment Options

From the acquirer's perspective, a fixed price assigns maximum responsibility for a quality result to the supplier, together with an incentive for efficiency. Both of these conditions benefit the acquirer, but acquirers should select a fixed-price type of contract only *if there is a complete*

understanding of the scope of work that can be conveyed effectively to potential suppliers. Acquirers should not price undefined or incompletely understood work this way. If the scope of work is open to interpretation, they should base the specific contract terms on an analysis of the risks involved.

Incentives

The objective of incentives is to align the interests of the supplier with those of the acquirer. Incentives usually take the form of monetary bonuses for improving:

- Negotiated costs
- Delivery schedule
- Technical performance
- Reliability or other desirable outcomes

If you are offering incentives, you can also impose corresponding penalties for failing to meet specified targets. However, note that in practice it is difficult to make precise assessments of performance to calculate incentive payments, and final settlements typically end up being negotiated.

You can define cost performance incentives in terms of:

- Target cost or price
- Ceiling (maximum) or floor (minimum) price
- Point at which acquirer assumes total responsibility for product operation
- Sharing ratio: acquirer's share versus supplier's share

Negotiating a Warranty

In Part III of this series, we listed "software warranty" as an item to include in the head contract for a software project. The warranty deserves special mention, because it comes into play during the vexing period when software development is nearly complete and the dreaded "acceptance testing" process is under way. In "traditional" contracting it is common for the buyer to expect, and the seller to provide, a warranty on the materials and equipment supplied as part of the final product. In practice, warranty requirements are many and varied, but the most common requirement is that any defects in materials or equipment discovered during the warranty period will be fixed or replaced. Warranty periods for "tangible products" are typically one year.

An expectation of warranty is not unreasonable in the case of software development, except that the nature of software is a little different from that of material products. Software doesn't break or wear out in the same sense; it fails because of coding errors, or "bugs," attributable to latent defects that were either always present or introduced during correction of

final functional deficiencies. The problem is that some defects may have been introduced as a result of the acquirer's request for an enhancement -- especially if the request came late in the development cycle.

There is no sure-fire way to resolve the discord that can arise in such situations. But provisions for compromises can be negotiated and written into the head contract. For example: perhaps a warranty period of short duration, such as three months, would be reasonable. During this time, the acquirer would be expected to thoroughly test the software for performance; after that period, any "fixing" would be at the acquirer's expense, at specified rates -- perhaps at-cost rates. Late changes might be specifically excluded from the warranty because there is a much higher risk of errors at a late stage, and on the grounds that the change should either have been identified sooner or held over until the next upgrade.

Another possibility is to include only a very modest warranty in the head contract and then negotiate a separate service and upgrade agreement as the project's final contract work order. As much as anything, a successful warranty period is a reflection of the trust and confidence that the two parties have built up in an honest effort to produce the product they originally agreed upon.

Next month, in Part Five of this series, we will describe how the workflow for acquisition activities fits into the Rational Unified Process.

Notes

1 ISO/IEC 12207 International Standard, Section 3: **Definitions.**

2 Software Acquisition Capability Maturity Model, 1999: **Appendix B: Glossary of Terms.**

3 NOTE: This article is not intended to offer definitive legal recommendations and advice, since these vary from country to country and jurisdiction to jurisdiction. In practice, all contract wording, whether "boiler plate" or specific to a contract, should be reviewed by competent acquisition personnel or legal advisors. For a detailed discussion of contract law refer to appropriate legal texts on the subject that are relevant to the governing jurisdiction.

For more information on the products or services discussed in this article, please click here and follow the instructions provided. Thank you!