

Systematic Acquisition of Products and Services for Information Technology

By

Woldeloul Kassa and Dawit Birhanu

Cisco Systems Inc.
170 West Tasman Drive
San Jose CA. 95134

Abstract

Technology purchases are highly risky, since products and services may not actualize as advertised or expected. There may be interoperability issues, or the products, services and designs may be out of date or may not comply with required standards. This paper discusses issues and concerns that should be addressed before and during the process of acquiring technology products and services from vendors, service providers and integrators. It presents general guidelines that can be used as a starting point to build on and to adapt to any particular need. The paper also provides best practices on the acquisition processes, including analysis and design, identification of possible design consultants, integrators, service providers and/or vendors, generation of request for proposal evaluations, pilot deployment, evaluation, phased deployment and verification.

1 Introduction

Technology has costs. It doesn't do everything the salesman says. The cost and effort required to get it to work and function as expected are frequently higher than anticipated. Regardless of whether you are buying products or services, there are some basic rules that always apply. Preparing a procurement strategy, a statement of requirements and evaluating vendor proposals, is not always an exact science, and this paper can only present some of the major issues and standard tendering strategies. Where an organization lacks either the experience or the skills to undertake a major procurement project, it is recommended that professional advice be sought from an independent consultant.¹

This paper will not go into issues specific to particular IT procurement of products and services, but it will look at some broader issues of acquiring IT services and solutions. Purchasing technology products and services is a challenge. There are few guarantees and lots of things that can go wrong. The impetus behind this paper is to prepare and assist you in laying out a systematic process in order to avoid potential problems.

2 Understand needs

Whether it is a network infrastructure, a computing platform or application software that you want to acquire, you need to clearly understand the goal you want to meet or the problem you want to solve. A thorough analysis of the problem determines the type of solution. Do not generalize your needs and problems. Be specific and realistic. It is important to put significant upfront investment into analysis before seeking vendors and integrators, since that analysis determines the effectiveness of the final solution. Upfront investment in analysis results in cost effective, scaleable and well designed engineered solutions. Moreover, needs assessment prevents over-engineering while binding the project scope and magnitude to the intended goal in a manageable way.

2.1 Project Scope

Once you understand your requirements, you need to generate a project scope and requirements document. Do not mix and match prospective solutions that are mutually exclusive. If your needs are network infrastructure, do not bring up data base software issues at the same time. The

expertise of vendors, designers and integrators is limited to specific areas. Some resellers will tell you they can do everything from bathroom plumbing to fixing an aircraft carrier. They probably can, by employing various contractors from all over the world. But this will be a headache for your project managers. It is tempting for any customer to describe a very broad need and to seek a turnkey solution that does everything from A to Z. The scope and requirements of the project must be simplified, itemized and explicitly outlined so that it can be managed.

2.2 Design Consultant

Once you have identified and scoped the project, you can start to undertake the design. Consider hiring an independent consultant to design the system. Even if the design is done by your own IT professionals, consider hiring an independent consultant to validate your design. Consultants come in different suits and shapes. Hire professionals instead of institutions that hire other professionals. Make sure that the consultant has the empathy and integrity to serve you to the best of his ability. Some consultants tend to produce tons of glossy and colorful documents by doing a cut-and-paste from user manuals and configuration guides. Engineering and IT solution must be clear, precise and to the point. Any IT design solution should address your particular problem, not general IT problems.

3 Solution Proposals

Using a consultant to produce your “Requests for Proposal” (RFP) is a good initiative. For some organizations, particularly government units, an RFP can be a very useful tool for providing the necessary structure for the acquisition of your project. If your design is solid and if you do not want to waste too much time with the vendor’s technical response to the RFP, then as a short cut you may want to issue a Request for Qualification. Alternatively, generate a list of vendors and/or integrators that you prefer to get proposals from invite them to respond to your RFP. Don’t overlook small and startup groups that are more agile, cost effective and eager to do a good job.

3.1 Standards based solutions

Unlike standards-based solutions, proprietary solutions tend to have interoperability problems with other platforms and/or devices. Some proprietary solutions may be supported by the marketplace. However there is a risk of being locked in to one supplier with a unique solution. In a case where the unique solution is considered, this approach has to be measured against the benefits of cheap and readily available solutions. Taking a standards-based approach is useful when the technical domain in question is mature. Vendors tend to sell you what they have and what they know best. Most of the problems of IT procurement are compatibility and interoperability. Always seek standard based solutions whenever possible. Proprietary solutions require specialized training and support.

3.2 Product and solutions roadmap

When you buy hardware, and particularly software, you want to know what new products the vendor is doing to develop. Ask vendors for a hardware and software roadmap of the product you are considering to purchase. The roadmap should show whether new features will be developed for the product, to give it an extended life cycle. Also ask the vendor to provide expected end-of-sale (EOS) and end-of-life (EOL) dates for the product and its components. You want to know what you’ll be able to upgrade to. You want to know if you should hold off on your purchase and wait for the new one. Ask for shipment dates on new versions and models. If something is on the horizon and the sales rep wants you to move now, ask for a guaranteed free upgrade. And if labor will be required to install that new version, ask for that as well. You have to weigh the benefits of the new version vs. the benefits of the product you’re giving up.

4 Examining vendors and consultants

You want to make sure that the company, its skills and its people are the right match for your project. Check if they have a track record of delivering goods and services in a timely fashion. Check if they have deviated from their initial designs or procurement proposals. Also see if they have incurred additional cost to their customers. When you sit down with a vendor, take control of the meeting. Vendors would prefer to be asking the questions so that they understand the points of leverage that can be used to get the deal. Instead, you should be asking questions. Explain the process that you'll be going through to ensure that you have a successful installation, and that this process will be of benefit to them as well. Describe your plans for a demo or pilot. If they're not qualified to do the job, you'll know by the deer-in-the-headlights look in their eyes.

4.1 Demo

A Technology Demo provides real proof of the technical feasibility and capability of what is hyped or marketed. Remember that a Technology Demo is an ideal visualization tool to show you (the client), or the decision-making body, what the final application will be like. It is not really a final product. Tech Demo is what the sales rep wants to show you. It is a well rehearsed show and most of the punch lines and marketing terminologies are carefully crafted. A vendor likes to demo his product because it enables him to showcase and highlight the advances of his product compared to his competitors. The advantage of Tech Demo is that it gives you an idea of how easy or hard it is to install, service, and upgrade compared to a similar product. Demo may also show you the ease of use for regular operations and manageability, again compared to another product. If you attended one vendor's demo, you should ask for and attend others. Do not base your judgment of a product on only one demo. During the demo, be ready to ask questions and what-if scenarios to get as much information as possible. Also request the vendor to allow you to try configurations on your own.

4.2 Lab Testing

Some companies have a "Try and Buy" policy where they give you the product to try for a while and then buy it if you like it. A trial or test unit is normally given free for a limited period of time or at significant concession for lab use only. If you are satisfied with the specs, demos and pricing, it is important to consider lab testing before any purchase commitment. Before you start lab testing, you should develop a plan to test the product under a scenario similar to real deployment scenario. Use of test equipment may be necessary to generate stress load and verify some of the functionalities. The purpose of the testing is to verify that the machine and the automation system perform up to the mutually agreed production objectives and quality standards.² You should also test boundary conditions, stress, stability, performance and scalability. The "Test run" will be done by you for weeks or months, as the product is supposed to perform in your environment. Lab testing is also extremely useful for knowledge transfer and to get the most information about the product from the vendor.

5 Selecting Winners

Once you have seen product demonstration and/or trial runs of the products and services, based on the narrowed-down design, type of service, quality of the goods and associated cost, you can now review the vendors' proposal in collaboration with your consultants. Take your time to review the technical details of the proposal. Some vendors tend to overwhelm you with documents. Ask them to give you an oral presentation of the detailed design and implementation. This is when they send their technical marketing guy or their good engineer. After listening to two or three of them, your IT professionals and engineers along with your consultants will have enough of an idea to judge how the proposed solution, product and/or service fits your requirements and addresses the problems.

5.1 Guarantees of Support and Services

When negotiating the contract, leave yourself a clear “out” if the vendor doesn’t meet your expectations. Make those requirements crystal-clear to the contractor as part of your scope. Demand a guarantee of satisfaction. To get the best results, you have to have the best people, not the best company. Beware of statements along the lines of "You are hiring our company, not the individuals." Let us face it, the work is going to be done by individuals, not the company. Immediately address any contractor’s or consultant’s performance if you don’t feel they have the expertise, skill or attitude that you are paying for.

When reviewing the warranties on products, discuss both the initial period of free support and service and the costs after the warranty expires. Make sure you understand the response times and service levels that are covered by the warranty and what are extra. Any system can fail and according to Murphy’s Law, at some point it will probably fail. Either the hardware, software or the services will, at some point, not act in the manner that you expected and were promised. There are few guarantees in this business. You have to be very careful about your purchase, to make sure you’ve done everything possible to get it right, because you probably won’t be able to get a refund. Unfortunately, your satisfaction is not routinely guaranteed. When dealing with the vendor, ask what the guarantees are. Get any commitments about functionality in writing.³

6 Deployment

Deployment should be broken down into phases.⁴ The purpose of the pilot phase is basically proof-of-concept in a controlled environment. If any project is very big and takes a long time to finish, if any project requires building and installing a huge infrastructure, or if the project involves large software platform migrations, then you should ask for a pilot deployment on a small scale, limited to a single location or environment.

6.1 Pilot deployment

The pilot deployment is quite different from a demo. It is the initial phase of the real project. It is not a prototype but rather a finished product deployed at your site that goes live and works as it is supposed to. The pilot deployment will let you know whether there is going to be any deviation or engineering change order (ECO) at an early stage. Engineering changes and deviations do happen, but you have to try to avoid them or minimize them. Even if they do occur, the cost and time to finish should be within the limits of a predefined standard deviation for the whole project. The pilot deployment allows you to see how vendors or contractors perform their duties in a small scale real environment and judge if he/she is up to the larger project. Make sure you have a clause in your contract such that the continuance or severance of the project shall depend on the timely and successful completion of pilot deployment.

6.2 Phased Deployment

Once a pilot is complete, then deploying in a larger controlled environment is the next step. The main point is to deploy something large enough to have reasonable confidence that full deployment will work without leaving the network incapacitated. If something goes wrong, it will be easy to make the required changes. Care must be taken before the deployment is done to inform those who might be potentially affected. Please document and try to resolve any nuances that occur during this stage. These may be warning signs of bigger problems to come.

Once a limited deployment is proved successful, the next phase of the full deployment can be considered. This is typically considered to be a site-wide deployment. If the deployment is for an entire enterprise, an entire campus or various regional sites, the phase needs to be subdivided into various smaller phases. Upon completion of the pilot and the first phase, the product or solution

should be ready for full deployment. The advantage of a phased deployment approach is that it ensures that the solution performs as expected. It also helps your internal processes to evolve so as to accept this new process, and that your staff will be ready to follow through the project.

6.3 Managing and gauging deployment

You should have a planned-absence clause in the contract requiring advance notice if a consultant is going to be gone for more than a day or so, for personal reasons, training or to assist at another client. Have the contractors and consultants maintain work journals of their activities, problems and solutions. These journals should be recognized as your property and should always be left on-site. You should also have weekly status meetings so that you can identify problems immediately. Beware of having the lead consultant or manager know more about the project than your own staff does. If he/she leaves, the timetable and execution of the project may be in jeopardy. Having weekly meetings and work journals will mitigate any damage.

7 Deployment evaluation and verification

Analytical measures can be roughly subdivided into two classes: validation and verification. Validation responds very much to the wishes of customers or users and operates with regard to these requirements. In the case of verification, the tester examines whether the system or product works according to the specifications and the documentation. Testing should always be oriented towards customer satisfaction. The customer should demand that the system function correctly according to his requirements, in this way decisively influencing what is meant by quality in this context. The requirements defined by the customer thus determine the testing procedure.

The Testing phase consists of integration testing, as well as the acceptance testing by your stakeholders and users. Integration testing is performed by the project team or your consultant, ensuring that the delivered solution meets your stated requirements. Acceptance testing goes one step further to ensure that the system meets your business needs and maps clearly to business processes. The whole concept of testing at various stages of the deployment is to insure a reliable, correct and acceptable implementation by means of a thorough verification methodology.

7.1 Functionality and Performance Testing

The objective of this test is to ensure that each element of the application meets the functional requirements of the business as outlined in the deployment design and other functional documents produced during the course of the project. These might include custom requests, resolution of issues and change requests that are agreed upon. The other objective is help you identify the correct functionality, performance, operations and compatibility requirements of all parts of the project, so that you are provided with services that ensure optimum performance and long-term reliability. Functional testing ensures that the system operates in the manner expected, and any supporting material such as procedures, forms etc. are accurate and suitable for the purpose intended. It is high level testing, ensuring that there are no gaps in functionality.

7.2 Security Testing

Security testing is in essence vulnerability assessment of your entire Information Technology deployment. Thus it encompasses the newly deployed project as well as your existing setups. It is often stressed that a security policy must be well articulated and that it accurately reflects your organization's needs. We think that you only need some sort of initial/draft security policy to start with. This initial policy will be used as a baseline for comparison with testing results. With a baseline policy, testing results can be used to improve the policy and make it a well polished security policy. Security testing can identify the vulnerability of the system and help in the improvement of the policy as well as the configuration and deployment. Note that this is an

ongoing and never-ending process. In general, systems that should be tested are software and hardware systems that are susceptible to denial of service, various malicious and non-intentional attacks from within and outside your organization. ⁵

7.3 Manageability and Modularity Testing

Modularity lets you keep each design element simple and easy to understand as well as manage. If the system is modular, managing it is made easy because there is clear functionality at each layer. Fault isolation is improved for you because your technicians can easily recognize the transition points in the system to help them isolate possible failure points. Testing for manageability and modularity is a bit subjective. Your consultant should gauge the level of training and understanding of your staff with respect to the complexity of the project in order to justifiably critique whether a certain solution is modular and manageable. Part of the modularity and manageability testing is verifying the “Disaster Recovery” mechanism. It needs to be verified if a procedure exists and if the existing procedure is adequate.

7.4 High Availability Testing

High availability solutions are usually in the form of a clustering or load balancing environment. The idea is if one unit fails then another unit takes over without downtime and without any disruption of your operation. High availability testing typically requires a failing over mechanism in either a clustered or load balanced environment. The test carefully looks into the whole design and sees if there is any single point of failure (SPF) that may make the whole design susceptible to unreliability. Units whether hardware or software, can and will fail at some time or another. The issue is whether the design has any failover mechanism in case one unit fails. Redundancy in IT is not a luxury but a necessity for reliable IT infrastructure and operation.

7.5 Scalability Testing

Scalability testing exercises an application or a product at an increasing number of concurrent users to determine when the application fails. If a system is reliable for n amount of time or m amount of units, you can not use the induction method to assume its functionality. The reliability of any particular unit for a particular amount of time and usage is good, but you want the system to be live and safe at a larger scale and for a long period of time. Many customers wonder why the number of concurrent users can't just keep growing, without having any effect on the application. This is because there are “bottlenecks” found in every product (hardware or software), that mitigate this problem. Bottlenecks might lurk in one or more computers, servers, database and the network infrastructure or in many other areas.

Scalability testing is designed to reveal the limits or boundaries of a device or a system. You should know that it is essential that vendors and architects fully disclose these limits prior to deploying any equipment. This requirement is even more critical in a multi-vendor environment. It is equally imperative that vendors fully comprehend the capabilities and the limits associated with their products, and you should be clearly informed.

7.6 Documentation verification

Technology is always changing and developing to meet the demands of the consumer. The more complex the product or service offering, the more important it is to provide clear, user orientated documentation. This documentation should be an integral feature of the product and service you acquired or purchased. It is not simply a necessary addition but rather a very important piece of the whole project. Verifying documentation takes aim not only at whether the documents address the product or service but also whether they meet your needs and requirements. Diagrams (logical

as well as physical) should be very clearly labeled. Configuration guidelines must be custom written for you, as opposed to a generic marketing literature.

Electronic and hardcopies of all documents have to be submitted by the vendor prior to the verification time. These documents are baselines for all the verifications steps that have so far been stated. After verification and testing is done these documents need to be updated and/or modified. Make sure the vendor delivers the final version of all documents after the verification, before you write him the final check.

8 Training

Adoption of new tools requires experts who can teach others how to use the technology. You can maintain almost every aspect of the training. The vendor should provide detailed training sessions that are customized for the application and use of the deployed solution. Some vendors may hide a few software commands, backdoor entries to configurations. Some vendors legitimately hide some tools from you for the valid reason of protecting intellectual property. Quite a few vendors do that to hide the bugs and anomalies of their products or to charge you more on service contracts. The training should not be a mere high level understanding of the system or the project. It should be in-depth training which enables your staff to maintain and manage the system during normal and abnormal situations.

9 Conclusion

This paper is a result of an initiative to offer guidance and to be of assistance in the acquisition of goods and services for Information Technology. Its main purpose is to help you understand the key issues involved in any contract for the procurement of technological goods and services.

IT procurement is more than managing contracts with IT suppliers. It is also much more than managing risk in acquiring expensive technology or services. It is about a key component of the IT architecture and strategy.⁷ It has a profound effect on the survival of your business and operation as well as future potential growth. The importance of such a guideline, even though it addresses only basic issues, is very significant. It can save you a lot of money, and moreover can save you from a lot of frustration when and if a project implementation goes wrong.

Reference

¹ Information Technology Association of New Zealand (ITANZ): Information Technology Procurement Guidelines, <http://www.itanz.org.nz/docs/Guidelines/Procurea.doc>

² Olli Nurmi; VTT Information Technology, December 1994
http://www.vtt.fi/tte/colourmanagement/project_folder/Tstprocnews.pdf

³ The Gadwall Group; Information Technology Procurement: The Successful Acquisition of Products and Services, 1999, 2000 The Gadwall Group, Ltd. All Rights Reserved

⁴ Paul Lewis; “General Practices Technology Deployment on the Network” December 2, 2002

⁵ John Wack, Miles Tracy, Murugiah Souppaya; Guideline for security testing Recommendations of the National Institute of Standards and Technology, NIST Special Publication 800-42

⁶ Richard Sitruk; (ETIS) “Information Technology Procurement Best Practice – Addressing Cost and Quality of Services in the Telecoms Industry” <http://www.bbriefings.com/pdf/976/30.pdf>