

## **SO.A Related Processes to Software Outsourcing**

### **SO.A.1 Supplier Management (applies to big clients)**

#### **SO.A.1.1 Definition**

The process of optimizing customer-supplier relationships to add value to the business.

#### **SO.A.1.2 Subprocesses**

- Categorize suppliers by value to the organization (e.g., volume, sole source, commodity, strategic alliance). Allocate resources to most important (key) suppliers.
- Develop and maintain a relationship strategy for each category of supplier.
- Establish and communicate performance expectations that are realistic and measurable.
- Monitor, measure, and assess vendor performance.
- Provide vendor feedback on performance metrics.
- Work with suppliers to improve performance continuously. Know when to say when.
- Continuously assess supplier qualifications against requirements (existing and potential suppliers).
- Ensure relationship roles and responsibilities are well defined.
- Participate in industry/technology information sharing with key suppliers.

### **SO.A.2 Supplier Performance Management (Applies to big clients)**

#### **SO.A.2.1 Definition**

The process of assuring continuous improvement in all elements of the IT procurement framework.

#### **SO.A.2.2 Subprocesses**

- Define and track meaningful process metrics on an ongoing basis.
- Conduct periodic quality reviews with suppliers.
  - Provide formal feedback to vendors on their performance.

- Facilitate open and honest communication in the process.
- Collect and prioritize ideas for process improvement.
- Use formal quality improvement efforts involving the appropriate people.
  - Participants may include both internal resources and vendor personnel
- Recognize and reward quality improvement results on ongoing basis.
  - Recognize nonperformance/unsatisfactory results.
- Audit vendors' facilities and capabilities.
- Conduct ongoing performance tests against agreed standards.
  - e.g., acceptance test, stress test, regression test, etc.
- Utilize appropriate industry standards (e.g., ISO 9000, SEI, Capability Maturity).
- Periodically review vendors' statistical process control data.

## **SO.B User Requirements Specifications**

### **SO.B.1 Introduction**

This section should provide an overview of the entire document and a description of the scope of the software.

#### **SO.B.1.1 Purpose of the document**

This section should:

- (1) define the purpose of the particular URD;
- (2) specify the intended readership of the URD.

#### **SO.B.1.2 Scope of the software**

This section should:

- (1) identify the software product(s) to be produced by name;
- (2) explain what the proposed software will do (and will not do, if necessary);
- (3) describe relevant benefits, objectives, and goals as precisely as possible;
- (4) be consistent with similar statements in higher-level specifications, if they exist.

#### **SO.B.1.3 Definitions, acronyms and abbreviations**

This section should provide the definitions of all terms, acronyms, and abbreviations, or refer to other documents where the definitions can be found.

#### **SO.B.1.4 References**

This section should provide a complete list of all the applicable and reference documents, identified by title, author and date. Each document should be marked as applicable or reference. If appropriate, report number, journal name and publishing organization should be included.

#### **SO.B.1.5 Overview of the document**

This section should:

- (1) describe what the rest of the URD contains;
- (2) explain how the URD is organized.

## SO.B.2 General Description

This chapter should describe the general factors that affect the product and its requirements. This chapter does not state specific requirements but makes those requirements easier to understand.

### SO.B.2.1 Product perspective

*Describe related external systems and subsystems.*

This section puts the product into perspective with other related systems. If the product is 'standalone', it should be stated here.

### SO.B.2.2 General capabilities

*Describe the main capabilities required and why they are needed.*

This section should describe the main capabilities and why they are needed. This section should describe the process to be supported by the software, indicating those parts of the process where it is used.

### SO.B.2.3 General constraints

*Describe the main constraints that apply and why they exist.*

This section should describe any items that will limit the developer's options for building the software.

This section should not be used to impose specific requirements or specific design constraints, but should state the reasons why certain requirements or constraints exist.

### SO.B.2.4 User characteristics

*Describe who will use the software and when.*

This section should describe those general characteristics of the users affecting the specific requirements.

Many people may interact with the software during the operations and maintenance phase. Some of these people are users, operators and maintenance personnel. Certain

characteristics of these people, such as educational level, language, experience and technical expertise impose important constraints on the software.

Software may be frequently used, but individuals may use it only occasionally. Frequent users will become experts whereas infrequent users may remain relative novices. It is important to classify the users and estimate the likely numbers in each category. If absolute numbers cannot be stated, relative numbers can still be useful.

#### SO.B.2.5 Operational environment

*Describe what external systems do and their interfaces with the product.*

This section should describe the real world the software is to operate in. This narrative description may be supported by context diagrams, to summarise external interfaces, and system block diagrams, to show how the activity fits within the larger system. The nature of the exchanges with external systems should be specified.

If a URD defines a product that is a component of a parent system or project then this section should:

- outline the activities that will be supported by external systems;
- reference the Interface Control Documents that define the external interfaces with the other systems; describe the computer infrastructure to be used.

#### SO.B.2.6 Assumptions and dependencies

*Describe the assumptions upon which the requirements depend.*

This section should list the assumptions that the specific requirements are based on. Risk analysis should be used to identify assumptions that may not prove to be valid.

A constraint requirement, for example, might specify an interface with a system that does not exist. If the production of the system does not occur when expected, the URD may have to change.

#### SO.B.3 Specific Requirements

*List the specific requirements with attributes.*

Specific requirements should be described in this section, which is the core of the URD. The acceptability of the software will be assessed with respect to the specific requirements.

Each requirement must be uniquely identified. Forward traceability to subsequent phases in the life cycle depends upon each requirement having a unique identifier.

Essential requirements have to be met for the software to be acceptable. If a requirement is essential, it must be clearly flagged. Non-essential requirements should be marked with a measure of desirability (e.g. scale of 1, 2, 3).

Some user requirements may be 'suspended' pending resources becoming available. Such non-applicable user requirements must be clearly flagged.

The priority of a requirement measures the order, or the timing, of the related software becoming available. If the transfer is to be phased, so that some parts of the software come into operation before others, then each requirement must be marked with a measure of priority.

Unstable requirements should be flagged. These requirements may be dependent on feedback from other phases. The usual method for flagging unstable requirements is to attach the marker 'TBC'.

The source of each user requirement must be stated. The source may be defined using the identifier of a system requirement, a document cross-reference or even the name of a person or group. Backwards traceability depends upon each requirement explicitly referencing its source.

Each user requirement must be verifiable. Clarity increases verifiability. Each statement of user requirement should contain one and only one requirement. A user requirement is verifiable if some method can be devised for objectively demonstrating that the software implements it. For example, statements such as:

- 'the software will work well';
  - 'the product shall be user friendly';
  - 'the output of the program shall usually be given within 10 seconds';
- are not verifiable because the terms 'well', 'user friendly' and 'usually' have no objective interpretation.

A statement such as: 'the output of the program shall be given within 20 s of event X, 60% of the time; and shall be given within 30 s of event X, 99% of the time', is verifiable because it uses concrete terms and

measurable quantities. If a method cannot be devised to verify a requirement, the requirement is invalid.

The user must describe the consequences of losses of availability and breaches of security, so that the developers can fully appreciate the criticality of each function.

#### SO.B.3.1 Capability requirements

The organization of the capability requirements should reflect the problem, and no single structure will be suitable for all cases.

The capability requirements can be structured around a processing sequence, for example:

- (a) RECEPTION OF IMAGE
- (b) PROCESSING OF IMAGE
- (c) DISPLAY OF IMAGE

Perhaps followed by deviations from the baseline operation:

- (d) HANDLING LOW QUALITY IMAGES

Each capability requirement should be checked to see whether the inclusion of capacity, speed and accuracy attributes is appropriate.

#### SO.B.3.2 Constraint requirements

Constraint requirements may cover any topic that does not directly relate to the specific capabilities the users require.

Constraint requirements that relate to interfaces should be grouped around the headings:

- communications interfaces;
- hardware interfaces;
- software interfaces;
- human-computer interactions (user interfaces).

If the software is part of a larger system then any documents that describe the interfaces should be identified.

Requirements that ensure the software will be fit for its purpose should be stated, for example:

- adaptability;
- availability;
- portability;
- security;

- safety;
- standards.

## **SO.C Method for Preparing the Request for Proposal (RFP)**

### **SO.C.1 How to define your requirements**

A major component of the RFP is the statement of work (SOW). This is your vehicle to define your needs. A few words of caution are in order: 1) If it is important to your organization, document it as a requirement; 2) If it is not documented, it will not be performed; 3) Requirements that are omitted, not clearly stated, or are constantly changing will lead to a negative relationship that could result in litigation.

The software outsourcing committee which is usually a cross functional team, is responsible for preparing the SOW. Listed below are attributes of a good SOW. This is not meant to be all-inclusive, but it represents a starting point for consideration:

- SO.C.1.1 Project goals and objectives and how success will be measured
- SO.C.1.2 Constraint requirements
- SO.C.1.3 Product specification
- SO.C.1.4 Quality requirements
- SO.C.1.5 Control requirements
- SO.C.1.6 Success factors (how performance is measured)
- SO.C.1.7 Roles and responsibilities
- SO.C.1.8 Security requirements
- SO.C.1.9 Interface requirements
- SO.C.1.10 Conversion requirements
- SO.C.1.11 Configuration management requirements
- SO.C.1.12 Communication requirements
- SO.C.1.13 Start dates
- SO.C.1.14 Completion dates, or milestones

- SO.C.1.15 What signifies the end of the project
- SO.C.1.16 Warranty or continuing support needs
- SO.C.1.17 Instructions to prospective bidders
- SO.C.1.18 Description of the selection process

## SO.C.2 Guidelines for Developing a Statement of Work

Answer the following questions as it relates to each process or product:

- SO.C.2.1 “Why” is this process/product critical to the success of the project?
- SO.C.2.2 “What” must we do to satisfy the process/product objective?
- SO.C.2.3 “How” must we perform to ensure process/product compliance?
- SO.C.2.4 “How” will we ensure that the products produced meet our expectations?
- SO.C.2.5 “What” measurement data is needed to give management the assurance that their goals and objective have been achieved?

The following are effective methods and should be incorporated into the statement of work:

- Process standards and procedures: what must be performed and how it is performed.
- Product standards: effectively you pre-define the attributes of each product. This establishes a basis for compliance checking.

## SO.C.3 Formal Change of Management System

Organizational change will inevitably occur. Controlled changes may involve new products and services, redesigning work processes, the use of new technology and response to customer demands. You will be faced with two types of changes during the course of the engagement. Controlled changes are normally aligned to a change in strategic direction, initiated at the executive level, and closely managed throughout the

organization. Conversely, organizations encounter uncontrolled or subtle changes that are more elusive and usually result from informal interactions in the hallway, memos, and daily problem solving. Uncontrolled changes are frequently the source of friction, confrontation, dissatisfaction among all participants, and the prime contributor to runaway projects. This is often referred to as “scope creep”.

It is incumbent upon you to manage change. All suppliers will certainly entertain changes to the original statement of work. Obviously, this will increase the project timeline and costs due to the possibility of rework or the need for additional resources. To maintain a positive working relationship with the vendor, you should:

SO.C.3.1 Have a well-defined process in place to identify and manage change internally:

- a) Who can initiate a request for change?
- b) What is the method for classifying and justifying the urgency of a change?
- c) Have all affected parties been involved?
- d) What are the decision-making criteria?
- e) Who has approval authority?
- f) How will changes be monitored and controlled?
- g) Review this process with the vendor for understanding and clarity.

SO.C.3.2 Most suppliers will certainly have some formal method of change control and invariably will have effective methods to protect their interests. Before ‘linking’ a contract, ensure that you thoroughly understand this process, agree to the process, and integrate your processes into the supplier’s process. Once you have a contract, this will become one of your most important concerns.

Consider the following process attributes closely:

- Impact analysis: How the supplier will assess the effects of change on completed work or planned future work.
- How the supplier will re-forecast cost, schedule, and resource allocations.
- How the supplier will manage and control the change(s).

Change will happen. Success depends on how well you manage change, justify change, and control change.

#### SO.C.4 Supplier Milestones and Quality Reviews

It is imperative that you, the customer, understand the supplier's processes. The project plan is a combination of your requirements and the methods used by the supplier to achieve your goals. It is impossible for you to understand the supplier's work breakdown structure, if you do not understand the entire process.

Given this understanding, you should participate with the supplier during the project planning process. Not all activities are of equal importance. Milestones represent those activities of major importance. Completion of requirements gathering is certainly a major event in the development of software. Major milestones often gain added importance because they serve as interfaces to other project activities. It is at these points that a control method should be deployed to ensure that performance/product expectations have been met. Some of the most common forms of control are reviews, walk-throughs, and inspections. You, as the customer, should participate in the control method to ensure that performance meets your expectations and that corrective actions are taken to remedy any non-conformance issue. Controls should be installed as close to the completion of a product as possible and before interfacing to the next activity. Do not over control, but ensure that your control methods are deployed at major milestone intervals.

The last milestone for discussion occurs much earlier in the process and is referred to as the contracting process. This is a critical point that should not be taken for granted. You can literally sign away all your rights by affixing your signature to a contract. Review with your legal counsel the goals and objectives, the statement of work, methods to manage and control the outsourcer, and contract verbiage. Do not sign a contract or any other professional services agreement without the approval of legal counsel.

## **SO.D Method for Selecting a Suitable Supplier**

### **SO.D.1 Selecting a Quality Supplier**

Most companies follow a process known as a request for proposal (RFP) as the driver to selecting a supplier. This is normally a “canned” process just waiting to happen. What we fail to realize is that the RFP process must be customized for each opportunity and each opportunity should be founded upon existing business needs, goals and objectives. The selection process is the first opportunity to display a professional awareness and understanding of your business risks and needs. If your selection process is ad hoc or chaotic, chances are you will never have the opportunity to have a positive impact or influence on the direction or quality of a vendor’s work. You are only putting your company at further risk by exposing your vulnerability. The following should be considered when selecting suppliers. 1) the competency of the supplier; 2) whether or not the supplier has well-defined processes in place needed to meet your needs. You certainly do not want to engage the services of a supplier less competent than you are and you do not want to select a supplier who does not already have disciplined processes in place to meet your goals and objectives. Well-defined, disciplined processes are an effective way for you to manage performance risks.

### **SO.D.2 How to select a Quality Supplier**

Do not rush through the supplier selection process. The time spent performing this process will be the most challenging, and at times the most frustrating, but the most rewarding investment of your time.

It is easy to rely on the industry reputation of prospective suppliers; take nothing for granted, leave no stone unturned, do your homework diligently.

Define a selection process should incorporate the use of team dynamics and several prerequisite tasks need to be accomplished:

**SO.D.2.1** Appoint a software contract outsourcing manager: An individual responsible to plan, facilitate, and coordinate all project activities. This individual becomes the prime spokesperson for the company during negotiations with prospective vendors.

- SO.D.2.2      Appoint a software outsourcing committee: This committee should be comprised of business subject matter experts capable of defining the attributes of the process and product; technical subject matter experts capable of communicating technology goals and constraints. The committee will also be responsible to participate in the evaluation of prospective suppliers.
- SO.D.2.3      Understand the goals and objectives: It is imperative that the outsourcing committee understand “why” a particular process/product is being considered for outsourcing and a clear understanding of management’s vision, goals, and objectives.
- SO.D.2.4      Process for selecting suppliers: The committee must define the process they will use to solicit bids, evaluate bids, and the selection success criteria.
- The competency of the supplier and their process discipline are equally important and should not be taken lightly. Both are reflective of the supplier’s ability to perform and meet your immediate and future needs. There are several ways to evaluate prospective suppliers.
- SO.D.2.5      Ascertain if prospective suppliers have achieved certification for an industry-accepted model (SEI CMM, ISO9000, and SPICE). If an assessment has been performed, ask for a copy of the assessment findings and improvement plan. If an assessment has not been performed, you may consider having an assessment performed against candidates to determine their level of process maturity. This will demonstrate to you the level of competency and the level of process discipline.
- SO.D.2.6      It is becoming a common practice to interview employees of prospective suppliers to ascertain if they have the proper skills and experience to meet your needs. The interview should be conducted as if you were considering hiring this individual yourself. At least, you should interview those responsible for managing your project.
- SO.D.2.7      Another method is to interview current and past clients of all prospective suppliers. You should select a time frame of 3-5 years, and ask for all clients served during that time frame. Contact active clients, as well as inactive clients, to

ascertain their levels of satisfaction, as these relate to the performance of their services.

SO.D.2.8 As part of the response to your RFP, request each supplier to submit the documentation for one business process. It might be requirements management, project planning, management and control, or may be configuration management. This may be reviewed and assessed internally or you may wish to engage a consultant to perform the assessment.

Suppliers who cannot demonstrate the necessary competencies or process disciplines should not be considered for the engagement.