**Articles**

|  |
| --- |
| **What is Scope?**  By: Art Moore, President, Clear Systems LLC Monday May 24, 2010  [Share/Bookmark](http://www.addtoany.com/share_save#url=http%3A%2F%2Fwww.bpminstitute.org%2Farticles%2Farticle%2Farticle%2Fwhat-is-scope%2Fnews-browse%2F2.html&title=Business%20Process%20Management%20%28BPM%29%20-%20What%20is%20Scope%3F&description=BPMInstitute.org%20%20is%20a%20peer-to-peer%20exchange%20for%20business%20process%20management%20professionals%20and%20is%20hosted%20by%20BrainStorm%20a%20trusted%20source%20of%20unbiased%20information%20and%20education%20since%201997.)    “Scope” is an “everybody knows.” And that may be the biggest problem. Consider this example: After reviewing a presentation on a major initiative (tens to hundreds of millions of dollars), one senior executive proclaimed that with scope now established the need was to work specific areas. Another executive, reviewing the very *same* material, pointed out that it was, of course, clear that scope was still not understood!  If you were to ask groups of business and systems analysts (we have), “What is scope?” you would probably find the responses most notable for their variety and for the amount silence imposed on the group by asking for a specific answer to the question. While most feel they have a “good sense” of what scope is, few can say exactly what it comprises, or give a precise definition.  This despite the fact that there is broad agreement that failure to understand and control scope is one of the biggest sources for system development failures.  The most common answer to the “What is Scope?” question is that it represents, at a high level, what the system is expected to do, the capabilities of the system, or some similar wording. This is correct in spirit and alright as far as it goes, but it is at best an incomplete picture. If this were the whole story, then identifying the scope of gifts and activities for your child’s next birthday would simply be a matter of asking the child what he wants to do and have and then fulfilling his stated desires. Most parents would say that is not quite how it works.  A statement of “Scope,” to be worth anything to us, must be grounded in some context of feasibility, constraints, stakeholders and other considerations. What is needed is to identify and clearly state these key characteristics of a confirmed scope so that, looking at a given initiative, we can determine with some precision and confidence whether or not scope had been established. Without this basic understanding, other laudable efforts to improve requirements specification methods and project success become akin to raking leaves in a hurricane.  This article represents what, out of hard won experience, we have identified these key considerations and scope characteristics to be.  **The elements of Scope**  The following are what we have identified, through trial and error and lessons learned, as the key elements of scope. Know these and you know your scope. Miss one or more of them and you can expect unhappiness to ensue. What you will probably find in this list is that none of them are much of a surprise to you; they’ve just never been explicitly enumerated so that one can objectively manage against them.  *The context - Business drivers and strategy*  Clarifying scope includes balancing need and feasibility within a framework of clarified risks, issues and constraints. That being the case, some sense of the project’s place in the organization’s overall strategy and response to drivers is needed to make fully informed decisions in this give and take of functionality. At least for smaller projects, this connection with overall business strategy may be ethereal but at minimum, some context of at least departmental goals should exist.  *Objectives* Even more to the point, no informed decisions about scope can be made in the absence of a clear understanding of the project’s actual objectives – *documented in writing and confirmed*. Keep these last words in mind for every one of these characteristics. If you haven’t done that, you don’t actually have the characteristic defined for your effort. Obvious, but it needs stating as it is so often exactly what does not happen and why the characteristic is never actually established.  *Weaknesses, Opportunities and Need* These are the particular aspects of the current environment that the project is intended to remedy, or at least improve, in order to achieve the stated objectives in support of an overall business strategy.  If everything were fine, there would be no project, or so one would think. (And *who* says these are weaknesses or opportunities?)  *Future Operating Vision* How are things supposed to look and operate once the system is in place? This is the goal operating state. The clearer it is, the better chance you have of arriving, knowing when you are off track, what is important and what is not.  *Capabilities* These are the beginning and end of what many casually conceive of as being “scope,” and are clearly essential. What exactly is the system supposed to do for us? The other elements of scope provide the context of why, in what way, involving what and whom, at what risk and with what feasibility.  *System boundary* Where exactly does the system begin? With what inputs or initiating events, from what stakeholders?  Similarly, where does the system end? With what outputs to what customers and collaborators? Create a system context diagram. Just that that exercise can produce surprising results.  *Impacted areas*   What are the affected processes, organizational units and roles, existing applications, data stores and technology, at what locations? You obviously need to know integration and transition costs. These impacts are the key to uncovering a comprehensive view of risks, barriers, issues, and clearly affect solution feasibility.  *Measures* Some measures are needed to quantify expected outcomes so that the degree of success can be evaluated in some objective framework. These are sadly lacking in many projects because it leads to actual accountability and limits the ability to spin a success story at the end of the project no matter what.  *Principles, Constraints and Assumptions* Principles provide a framework for guiding execution and the selection of options. Should we be using COTS wherever possible? Are existing employees to be retained and retrained? Is adherence to an enterprise view and long term value horizon important?  What constrains must the project operate under, what limitations are there on solution alternatives?  Better to know and explicitly state these up front than experience a later, expensive train wreck. Too often project sponsors and management, finally frustrated and at a loss to distinguish between fogs of negativity and responsibly stated limitations, begin to turn a blind eye to blatant barriers and, in the worst case, begin a death march.  What things do we assume will be there when and where we want them? What other factors impacting the ability to complete the project or the ability of the chosen solution to operate, do we assume will be as we expect and need?  Critical Success Factors should also be added to this list.  *Risks, Issues, and Solution Strategies* If we don’t know these and address them, how can we say our “scope” has any feasibility?  Walking through all the preceding scope characteristics - in particular future operating vision, impacts, principles, constraints and assumptions – exposes these, provided you’re looking.  *Stakeholders (and their concurrence)* Have all the affected stakeholders actually been identified? Is their perspective on the problem known?  Have they been appropriately briefed on the proposed solution and its impact on them, and has their buy-in been established?  If this hasn’t been accomplished, your solution – and probably even the problem statement - is not validated and you can expect the obvious consequences.  **Scope and the Solution Concept**  “Scope” without some matching Solution Concept that has been vetted for acceptability and feasibility is like the child’s list of birthday wishes. It may represent a thorough list of needs, but until it is compared against the framework of an actual solution concept, it does not provide a useful boundary of action. This may seem too obvious to state but experience says otherwise. In some cases, projects, in an extreme effort to keep the business view “pure” and unfettered by technology “arbitraries”  - because of past difficulties in collaborating successfully - can engage in broad (and expensive and time consuming) gathering of client need, before turning serious attention to technical approach and ability to deliver. The consequences generally include unpleasant surprises, dissatisfaction, and another brick in the wall separating business from IT. This problem only gets worse by ignoring it, especially as technology continues to advance and be an ever larger driver in the business itself.  As already noted, Scope and Solution Concept are inextricably bound. A project needs to iterate between these, with all necessary stakeholders at the table, until an explicitly stated balance of expected value, feasibility and risk is achieved, before proceeding further. This means that how much scope definition is necessary at the concept stage is not black and white. It is a matter of how much risk you are willing to assume, including the risk of not knowing what all your risks are. You owe to yourself to at least confront this head on, rather than stumble over it later and take a nasty fall. At a minimum, assess your project against all these factors and decide for yourself.  Do you know the objectives of your project? How well and how much of a risk is that? Do you know its boundaries? And so on. Otherwise you are likely, among other unpleasant outcomes, to be defining architecture for elements you will never build, and gathering and modeling more explicit requirements than you have any hope of satisfying, a famous sore spot with business stakeholders.  **The “When” of Scope**  When do you have to know scope? At the end of a project, scope is very well known, inasmuch as it ***is*** the delivered system. That, of course, is a little late. But you certainly do know more and more as you proceed through the life cycle. So to make the question more specific, how much of the above do you need before proceeding to actually architect your solution?  As already noted, Scope and Solution Concept are inextricably bound. A project needs to iterate between these, with all necessary stakeholders at the table, until an explicitly stated balance of expected value, feasibility and risk is achieved, before proceeding further. This means that how much scope definition is necessary at the concept stage is not black and white. It is a matter of how much risk you are willing to assume, including the risk of not knowing what all your risks are. You owe to yourself to at least confront this head on, rather than stumble over it later and take a nasty fall. At a minimum, assess your project against all these factors and decide for yourself.  Do you know the objectives of your project? How well and how much of a risk is that? Do you know its boundaries? And so on. Otherwise you are likely, among other unpleasant outcomes, to be defining architecture for elements you will never build, and gathering and modeling more explicit requirements than you have any hope of satisfying, a famous sore spot with business stakeholders.  **Summary**  Few if any of these scope elements should be a big surprise. Maybe that’s why they’ve gotten so little attention and so little clarity has apparently been achieved. “Everybody knows.”  What we can say is this:  By using this list we have found gaps in scope definition to be a very common occurrence. We have also found that just having the list of items to work off has been a tremendous aid. In some cases it has served to expose, if not solve, even more fundamental problems, for example, tremendous responsibility and expectation to deliver without the access, means or authority to even investigate and establish these scope elements at all. Even in these cases, knowing what clarity of scope *should* consist of is the first defense.  “Everybody knows” that a clear definition of scope is critical to project success. Now we can all know what we actually mean when we say that.  *Art Moore, president of Clear Systems LLC, Business Methodology consultant and Business Rules Architect for the IRS Business Rules and Requirements Management Office, has over twenty-five years of systems experience, from systems development to IT strategic planning and practice management. Mr. Moore has focused extensively on assisting large projects and organizations to establish practical, integrated business analysis activities and specifications, including incorporation of business rule related artifacts, methods, tools and management practices. Mr. Moore is a contributing author to Business Rules Applied (Wiley, 2002), as well as The Business Rules Revolution (2006).* |