

Management Brief

The New Value of the Business Analyst

*Why your organization should
be investing in this valuable resource.*

January 2011

Author: Dan Drislane



frontier-strategies.com
info@frontier-strategies.com

Copyright © 2011 Frontier Strategies, Inc. All Rights Reserved

No part of this work covered by copyright herein may be reproduced in any form or by any means—graphic, electronic, or mechanical—including photocopying, recording, scanning, taping, or storage in an information retrieval or computer system, without prior written permission of the copyright owner.

This paper originally published in a slightly different form for Requirements Networking Group. (April 2008). www.requirementsnetwork.com

Frontier Strategies, Inc.

1106 West Park Street, Suite 444, Livingston, Montana 59047-2955 USA

Phone: (+01) 248-207-9020 Fax: (+01) 406-794-0506

E-mail: info@frontier-strategies.com

frontier-strategies.com

The BA as Phoenix

If you were to drop into any ten Fortune 500 companies, requested to meet one of their IT managers, and then asked that person how many business analysts they have on staff and what they do, chances are that half of the ten will pause thoughtfully, then answer, “We don’t have any BAs.” The other five managers will nod, and then tell you five different stories about what their BA folks do. While these numbers are anecdotal, I never usually get an argument when I state to colleagues and clients that there are as many types of BAs as there are people you ask. You may think this is worrisome but I think it’s just a healthy sign of a job position in transition.

As any of you who’ve been in the IT business for a decade or more know—especially if you now *are* a genuine business analyst (whatever that may be)—your role has evolved from numerous sources and roles in the past. You come from humble beginnings. After having done a brief stint as an engineer, I started out in IT in 1983 when I joined Digital Equipment Corporation as a software specialist. We as specialists working in the field with DEC’s sizable customer base were expected to do everything the programmers¹ and testers didn’t. This included engaging our customers, documenting requirements (the terms we bandy about today, “eliciting” and “capturing,” were unheard of), and then writing thick functional specifications that, to be honest, few bothered to read, especially the programmers. But as software specialists, we were also expected to prepare proposals, sketch business processes, read code, and maybe even write a FORTRAN routine in a pinch.

Our extended roles weren’t unique, nor were our titles. The software specialists at DEC were the system analysts at GE; were the programmer analysts at Alcoa; were the application support engineers at GM; and were the financial systems analysts at Smith Barney. The job function that would eventually evolve into the business analyst came from myriad roles, pedigrees, college majors, wage classes and industries—oh, and also that stalwart of positive change, re-orgs. But one evolutionary step that helped define the heretofore unknown business analyst in the mid ’80s was, oddly enough, the emergence of object-oriented languages, notably C++.²

Though there are certainly additional factors, and probably conflicting opinions, object-oriented paradigms that promoted the idea that a “class” could encapsulate both data and business logic (“the code, man”) led to legitimate roles that centered on software design, software architecture, and software development as a lifecycle. Relational database technology and its nascent market, just gaining traction at that time, also helped, and these developments in turn inspired thought leadership on software modeling (i.e. use cases³, Unified Modeling Language), component architectures, and eventually things like extreme programming and agile development.

If I haven’t lost you, then good! What all this meant was that, for the first time, there emerged a role that didn’t require you to be a propeller head—and I certainly was one—to get the job done. In the early 1990s, business analysis began to earn its keep as a separate discipline where the person filling the role

¹ “Developer” as a recognized role (and also “software engineer”) was relegated to serious product development teams back in Boston; the field guys doing work in PL/1, Macro and Pascal were just plain ol’ programmers.

² C++ was introduced in 1979, but it wasn’t used by field bees like my programmer colleagues until the late ’80s. ANSI didn’t publish a standard until 1998.

³ Although Ivar Jacobson conceived the idea of the use case while working for Ericsson in the 1960s, the idea didn’t take off until the OO community embraced it almost 25 years later.

really had little to do with building the software and more to do with understanding and then documenting what needed to be built. The role then was filled by generally two types of people: disenfranchised programmers, who found the new OO-language revolution daunting—COBOL was user friendly by comparison—and business workers intrigued by being coined an *analyst*. For the latter, mostly subject matter experts (SMEs) with titles like product specialist, marketing support manager—you name it—business analysis was really what they did each day anyway. Business cards I collected in the mid-90s attest to the role being labeled as such. While the IT machine was immersed in new tools and development paradigms, and emerging architectural frameworks, business analysts were worrying about what business people thought and worried about. At least that’s what they should have been worrying about.

However, that’s not to say it was the correct role, or an effective one. Throughout the ‘90s, Fortune 500 companies started to legitimize the role and created job classes with BA-like names. Although most role responsibilities were ambiguously polarized along the lines of “analysis of business stuff in order to build software stuff,” the precise skills, expectations and value of the BA were up for grabs. Anyone who job-hopped as a BA type during this time knows what I mean: the new BA job you’d just landed was wholly different from the last one you left. And your marketable asset wasn’t your analytical skill, mind you. It was your knowledge of the *biz*, what we casually call today *domain expertise*. It was more rewarding, financially and otherwise, to know the difference between a mortgage broker and a servicer than a business rule and a business activity. Then in the mad rush of Y2K, the breadth of the role widened and diluted even more as armies of people (including BAs) did whatever it took to meet the coming deadline.

In the years since 2000, the role of the business analysis has refocused due in part to the burgeoning emphasis on requirements excellence, business software modeling, and also business process management, all would-be provinces for the BA. The web site from which you are reading this (requirementsnetwork.com) probably wouldn’t have been possible eight years ago because, though industry publishers like Addison-Wesley were trotting out requirements management books at the time (Leffingwell and Widrig’s *Managing Software Requirements* was an early entry), the notion of requirements management as a core competency didn’t resonate with most companies.⁴

But eight years makes a difference. Though it’s just a fraction of the 60-year pie where a good business analyst would have been a boon to the computer age, in this first decade of the new millennium, the BA role is filling out. Though it’s really just coming of age, it’s also reinventing itself, a phoenix arising from those early days when one did analysis, programming, testing and then some. More and more companies realize that business strategies and tactics, where they involve software initiatives, are getting lost in the translation, that mud of ambiguity that permeates many software development processes. Though software defects attributed to faulty and misunderstood requirements have been known (and suffered) since the 1960s, companies today are focused on doing something about it. The BA is the person who squarely sits in the driver’s seat to consistently and effectively deliver the business requirements from the product division or marketing department.

To help with this challenge, new tools, approaches and benefactors are available and growing each year. For tools, there are numerous vendors, some from the old school⁵ and newbie’s as well, who continue

⁴ That’s not to say that some weren’t focused on managing requirements effectively. Those who were good at product requirements management—notably the aerospace and defense industries, the pharmaceuticals, and to some degree the Japanese automotive producers—adopted some of their best practices in this area to software development, if only for their own in-house initiatives.

⁵ IBM’s Rational division and Borland are two established vendors offering tools for the BA.

to improve the BA's lot with integrated requirements capture and business modeling applications. These same vendors and others continue to enhance their business process automation offerings, and it won't be long before the still disparate business process management and business software modeling domains will be unified.

Standards organizations and professional societies have bolstered their focus on the business analysis practice. The Object Management Group over the last five years has been prolific in the business process and architecture arena. Its Business Modeling & Integration Domain Task Force's mission is to develop specifications of integrated models to support management of an enterprise to include integration and collaboration of people, systems, processes, and information across the enterprise, including business partners and customers. The International Institute of Business Analysis (theiiba.org), though only five years old, has garnered support worldwide from people working as BAs and companies who wish to develop their BA area of competency. Like the Project Management Institute is for PMs, IIBA is really party central for the BA role, and as such has introduced popular PMI concepts like local chapters, the Book of Knowledge and certification for BAs.

Coming to Market: The New Business Analyst

Today's business analyst is expected to do much more than just document business requirements and perform rudimentary analysis tasks. The BA is now equipped with the skills and tools required to thoroughly express the requirements for a new system to be built (or a legacy system to be enhanced) without any loss of business vision, intended business behavior of the system, business policies to be observed and user experience to be realized.

The Business Analyst has the ability to specify every aspect of desired system business behavior through the use of business scenarios for use of the system, user navigation, business rules to be fired, how user tasks (business work) will change because of the system; the authorities and roles that are necessary to control access to system resources; the business content of forms, documents and UI screens themselves, and also how operational and deployment business considerations of the system will be realized. Each of these specification areas are linked with other specification elements, then traced to the original stakeholder requests to form a comprehensive, pure-business elaboration of the new system.

So what, you might ask. Well, what *gives* here is that the BA can now do the heavy lifting like never before. Business specifications can be produced with nary a technical peep in the artifacts. This has the twin benefits of keeping the business requirements and their elaboration (the business specification) comfortably in the lap of the business requesting the capabilities, while assuaging the IT team that their domain is purely one of design, architecture, and raw solution. Object-oriented perspectives correctly inform us that separation of concerns is a good thing for many reasons. It's now possible to extend that to the business domain in far more powerful ways.

There are good analogies to this argument. In much the same way an architect designs a new building with numerous models, models that are so semantically rich that the craftsmen and laborers constructing the building are left with little doubt about where to run wire or frame out a door, the BA can similarly produce artifacts that are unambiguous, approach the business problems from different perspectives and views (i.e. business work, user experience), and render a unified, cohesive business

specification that, like a building’s architectural drawings that “live” with the building, so too can this specification “live” with the software application for its lifetime.

The business analyst today is empowered to preserve the “voice of the customer” through these business specifications, the key benefit of which is that the business retains ownership of the system. Though IT professionals deliver the implementation, the blueprint of this system—the one that teams will go back to when a change is required much the way a future architect will do when that same building undergoes renovation—is firmly in the care of the business analyst for use by the business. Little if any translation is lost due to transformation of the business vision—the voice of the customer—into lower level implementation artifacts.

Preserving this fidelity of the business vision significantly enhances the ability for a business organization to become agile and implement its product and service tactics at a lower cost.

Assets to Bear

Today’s business analysts have three major assets that help make them most valuable players on each project team:

- **Knowledge about the Business** – Though this was once the significant value-add of a BA, it’s just one of many assets in the BA’s hip pocket. Even so, the BA works to understand the business domain, key business concepts and relationships, business acumen, the “language of the business,” and major driving business policies. This knowledge is used to enrich the system business specification of the system to be built (or enhanced).
- **Business Skills** – Armed with numerous soft skills and mastery of up to two dozen analysis techniques (hard skills) enables the BA to draw from many problem-solving competencies in order to understand and precisely document the business problem the solution to be built endeavors to solve.
- **Analysis Tools** – Each hard skill is supported by a technique that is well defined in its description, reasons for use, semantics, and how it not only traces to the original business requirements of the stakeholder, but also how it integrates with other analysis artifacts. These techniques are best in the industry and are supported by key standards bodies and professional societies, including the International Institute of Business Analysts (IIBA).

Have Skills; Will Analyze

With new skills, the business analyst can bring new perspectives to understanding the problems confronted by a business organization. Like object-oriented modeling techniques,⁶ separation of concerns can be preserved using different perspectives of the business problem. (By the way, this holds true with our building architecture analogy: plumbing contractors reference pipe networks while electricians consult wiring diagrams.)

⁶ One popular approach advanced by Grady Booch, Ivan Jacobson and James Rumbaugh in their introduction to UML was the concept of different views in modeling a system. There were five basic views: the use case view, then the design, process, implementation and deployment views.

The BA, through skills and proficiency in a variety of business modeling and analysis techniques can help assure that nine benefits are realized:

- Maintain structure and consistency
- Provide effective liaison between business and IT
- Enriched ideation
- Improved requirements management
- More accurate estimates
- Better project planning
- Richer elaboration of the business vision
- Testing hears the voice of the customer
- Transition

Let's look at each of these and discuss ways the BA can add value to a project.

Maintaining Structure and Consistency

Architectural renderings for a building would quickly lose their cachet if every architect, willy-nilly, decided to fashion his or her own drawing standards, nomenclature, symbol set and semantics. It would be akin to asking kindergarteners to draw the view outside their classroom, complete with a legend. You'd get wildly interesting results. BAs can help preserve structure and consistency by maintaining a discipline of good documentation through robust definition of nomenclature and semantics. Examples abound here.

Probably first on the list is business vocabulary, the language of the business. This can and should be maintained by the BA, and its existence is no less important than the legend on an architectural drawing, or to be fair, a 5-year-old's masterpiece. It amazes me that, even today, few organizations pay attention to keeping a valid and accurate vocabulary. It's this lexicon that will pervade other areas of the business specification, notably business entities and attributes, and also business rules.

Speaking of rules, this is one area the BA should be practicing in. Business rules exist as distinct concepts and should not be second class hitchhikers in other areas of the problem domain. Further, the BA can assure that each rule is expressed using the business vocabulary.

Business process models, arguably another province of the BA (lest there be dedicated analysts for this domain), should have direct representation of the automation solution to be built. This may be a novel concept to many of you, but a vast majority of new green-field software initiatives change the way an organization works. Rather, it should be the need to work differently that calls for new automation capabilities, but you knew that. Task redesign, which is what much of what we do in building software actually addresses—new ways of performing business work—should be married to any process analysis work being done. BAs have a unique role since they should understand how the business functions (its core processes) and, further, they understand how the new software will impact how tasks are performed.

Business entity models are also in the BA's sphere of responsibility. In the most general of views, they are the structural representation of business vocabulary. They represent business concepts, usually in

the form of a noun, such as *Customer*, or *Policy*, and also business relationships. Business entities are the primary modeling element of a business entity diagram. Maintaining a consistent business entity model (one or more diagrams) that is free of technology or implementation artifacts is important for preserving the integrity of how the business represents knowledge. It should come as no surprise to you that most business entity models will transcend a given software project for it really is a static model of a business' vocabulary.

Though less well appreciated than business rules and business entities, BAs can ensure that business events have a direct representation in the automation solution. Business events are important temporal occurrences that a business organization can't afford to ignore. They kick-start processes, workflow and business automation. BAs can help identify and maintain how these events are integral to the business.

As models become part of the fabric of the organization (as they should), the BA should be charged with maintaining these models (like a building's drawings) and ensure that they are reused across projects. After all, the models are not the software per se; they're the business. They represent repositories of knowledge within the organization and should be the mandated leaping off point for subsequent analyses of software projects.

Last, BAs can make certain that the specifications they produce are distinct deliverables for the project. The specification is the milestone that the business problem has been fully elaborated—not by IT, but by the business.

Providing Effective Liaison between Business and IT

The BA is in a unique position to bridge the gap between the business users who demand requirements of an automation solution, and the IT staff who will ultimately build it. Because the BA can express the stakeholder's requirements in an unambiguous, semantically rich business specification, very little is left to interpretation and translation on the part of the implementation team (the architect, designers and developers). This in turn prevents the following issues from occurring:

- **Validation Denial** – Business domain experts cannot validate business correctness of the solution based upon direct examination of the implementation.
- **Testing Deadlock** – Business domain experts cannot directly test their own business logic.
- **Diluted Ownership** – Ownership of business policy driven concepts is either not defined or ambiguous.
- **Line Item Requirements Are the Only Story** – Designers, software architects and developers are left to read, understand, and interpret written business requirements as the sole source for designing and building the solution.

Providing Early Advice of Risk and Complexity during Ideation

Ideation is a term that many companies have recently adopted to expose the inherent opportunity a project team has in discovering the true requirements, dependencies, risks and complexities of a software project. Because this is essentially home turf, the business analyst can perform early analysis of the business impacts that a new system will cause. Largely one of advice and support, the BA role in Ideation should be focused on counseling the project team and business stakeholders that there may be

impacts in the following areas: data configuration and loading; business process change; user task interaction changes; workflow design/redesign; data definition additions/changes; business rule changes; and business content design/redesign.

How does the BA know these? Unless the business unit is brand new and the software project is also new, these items are already encapsulated in the models of the business specification. It would be like the original building architect (as is frequently done) offering consulting services to the current architect on a modernization project. The BA has prized domain expertise, only in this sense the “domain” is the business specification.

Improving on Requirements Management

The BA has the ability to shepherd business requirements from stakeholder request, to precise functional requirements, to business specification, such that all gaps in understanding and interpretation of what the automation solution is to do are identified and resolved. Such a capability shouldn't be taken lightly for it's probably the eminent contribution today's BA can make. This shepherding is possible because there are distinct processes for handling elicited requirements versus requirements mined from existing systems. There are also defined processes for managing and deciding between line item- versus functional flow-style requirements. A defined process for prioritizing requirements based upon contribution to project goals also exists.

Just as important, a framework is in place for identifying how a requirement has been satisfied via each element within the specification. And if traceability is an important strategy, and it usually is, requirements can be tracked to the business strategies and high-level business goals, which they support.

Analytical Skills that Result in Better Estimates

Like Ideation, the BA is in a unique position to utilize his or her skills to bring a more complete and informed analysis estimate of the project. The BA role in estimation should be as a contributor to the combined estimation effort, and though the items below may be addressed by other team members, there is always an analysis consideration:

- Entity Instance Definition Impact
- Business Process Impact
- Task Design Impact
- Workflow Design Impact
- Entity Design (Yes/No)
- Event Design Impact
- Rule Design (Yes/No)
- Content Design (Yes/No)
- IT Interfaces to "external" systems
- Key Business Scenario Identification

- Volumetrics
- New Technologies
- Mission Criticality
- Number of Related Projects

Better Project Planning Through—You Gussed It—Analysis

Through the BA's knowledge of the business domain, knowledge of previous projects that had similar characteristics, known as a *project archetype*, will help the PM and the rest of the planning team with resource identification; risk identification; mitigation tactics; and milestone identification. My experience in this area is that PMs will push back on the BA's role here for many reasons, most not justified. Admittedly, these are not unique to the BA, but PMs should recognize that the BA should have a voice in all four from a business analysis perspective.

Robust Analysis of the Business Vision

Obviously, the BA plays a key role and should drive all business analysis activities. The following are just some of the techniques and skills the BA brings to analysis (in alphabetical order):

- Authority and Role Capture
- Business Architecture Modeling
- Business Entity Modeling
- Business Event Capture and Modeling
- Business Process Modeling
- Business Rule Capture and Modeling
- Business Scenario Capture and Refactoring
- Content Modeling
- Metric Identification and Linkage
- Navigation Diagram Modeling
- Requirements Capture and Refactoring
- Usage Scenario Modeling
- Use Case Modeling and Development
- Workflow Modeling

Assuring Testing Also Hears the Voice of the Customer

Sometimes not given its due, testing (or quality assurance/control) is a valuable component in the software development lifecycle of a project. The BA's role in the testing process should be one of oversight (albeit limited) by focusing on validation. Key among validation activities is that core functionality is tested. One way to assure this is that the BA should encourage the testing analyst to

participate intimately in the analysis of the riskiest and most complex functions of the new application (or enhancement). One mechanism that aids this discovery is the identification of key business scenarios, which help drive elaboration into use cases, usage scenarios and workflows. If the testing analyst can key into these scenarios as part of his/her test strategy, then the BA has accomplished the mission.

Usability, or look and feel, is a discipline apart from the functions of the system, and should be an equally important concern. Because the BA refactored the original stakeholder requirements and should therefore have captured what usability requirements there are, advice can be offered the test analyst on how to structure the test cases for each usability requirement.

Most projects have some sort of performance requirements which give rise to volumetrics that must be attained. Volumetrics take different forms but can be measured in various ways, including transaction volume, simultaneous users, and database hits, to name a few. The BA can advise the testing team on the approach and validate that volumetrics are addressed.

Though the process of baselining is not strictly a testing concern, the BA can help validate that test strategy addresses the baseline at hand.

Enhancing the Transition Effort

Because the BA has intimacy with all the requirements, the risks, and the business specification (the analysis artifacts), he/she is in a unique position to identify if there are changes to the way the user will perform work and if there are operational or deployment considerations that have to be documented. The role at this stage is one of advice and includes the identifying user task impacts and operational exceptions. It's important for transition and rollout teams to be aware of these considerations which may impact procedural documentation, training, and job aid factors.

At the End of the Day

So, with all these new skills and capabilities and knowledge, the new BA is armed to the teeth. A new warrior is in town. Business analysis will never be the same. But is all this possible? Does it exist today? Are there walking-talking BAs out there that march to this drum well and with conviction? The answer is, as many are when confronting complex topics: Yes, and no.

Our experience is that there are aspects of most every facet of the new BA out there, living and breathing in real day-to-day jobs. Their skills may not be named the same; the artifacts they produce might be of a slightly different stripe. Some may be enlightened business rules practitioners; others specialize in user experience modeling; and still others are masters of good requirements analysis. But all are focused on rounding out their delivery of comprehensive business specifications.

And make no mistake: companies are paying attention. Several large organizations we're working with today are committed to improving their BAs' competencies; their mastery of skills; their impact on the organization. More important, they're willing to knock down the barriers to give the BA a bigger seat at the table. In our estimation, today's BA, with all these new tools, techniques, skills and value in the offering, isn't around the corner; he and she are here now. The onus is on all of us who work on software projects to do a better job leveraging this new and remarkable role.

About the Author

Dan Drislane is the founder of Frontier Strategies, Inc. in Livingston, Montana, an IT consulting firm begun in 1991 in Michigan. He has 29 years of experience in business analysis, business process analysis and project management. Dan's work with clients focuses on two goals: transforming the IT organization's culture so it will be more agile and accountable to business; and integrating an organization's vision and supporting business processes with its enterprise business and system architecture.