

**CLIENT** Brickwalk

**PROJECT** White paper

## OBJECTIVE

Create white paper on business requirements planning, analysis, and management.

## BACKGROUND

Brickwalk is a systems integration firm specializing in planning, deploying, and testing applications for business process management.

In some cases the company's first assignment for an organization is consulting on developing business requirements — an essential, yet complex, first step for any process change.

## PROJECT OVERVIEW

Many senior executives understand the importance of business requirements, but don't have an appreciation for the many nuances and specialized techniques for managing the process for developing and analyzing them..

This white paper, "Purpose. Process. People" presents Brickwalk's best practices for business requirements. This framework balances a disciplined approach to mitigating high deployment risk with change management processes to maintain flexibility during the complete project lifecycle.



.....  
*See below to read this white paper.*

## **Purpose. Process. People.**

A new framework for BPM  
business requirements planning

by Matthew Williams  
President, CEO  
Brickwalk  
October 2006

## **Purpose. Process. People.**

A new framework for BPM business requirements planning

### **INSIDE:**

- 3 **EXECUTIVE SUMMARY**
- 4 **INTRODUCTION**  
Business requirements in the real world
- 6 **THE THREE TYPES OF REQUIREMENTS**  
Business, user, and functional requirements
- 7 **BEST PRACTICES FOR REQUIREMENTS GATHERING**
  - 1. DEFINING OBJECTIVES  
Requirements first, capabilities later
  - 2. IMPROVING PROCESS  
From discipline comes flexibility
  - 3. ASSEMBLING THE RIGHT TEAM  
Navigating unfamiliar territory
- 13 **THE REQUIREMENTS LIFECYCLE**  
Five fundamental steps
- 16 **A SMOOTH DEPLOYMENT**  
Planning against failure
- 17 **A TYPICAL REQUIREMENTS PROJECT**
- 18 **ABOUT BRICKWALK**

## EXECUTIVE SUMMARY

Nearly every IT executive can recall hearing of a major software development project that was rushed through deployment and later scrapped because it didn't meet expectations. Why does this seem to be a recurring theme?

Indeed, implementing business solutions without clearly defined requirements can turn an orderly process into a chaotic and unpredictable chain of events. And in most cases, the difference between order and chaos depends on a set of precise and verifiable business requirements. With them, projects seem to run smoothly. Without them, the risk of major disruption soars.

Yet executives are often tempted to skip this key process step, predicting that it will save time in the end. Unfortunately, the end usually brings soaring costs and frustrating delays.

In this white paper, you'll learn how Brickwalk, a leading systems integration firm, mitigates high risk through a real-world approach to requirements that emphasizes:

- **A clear understanding** of the characteristics of effective requirements
- **A proven process** for gathering, analyzing, and managing requirements
- **A project management philosophy** based on leadership, listening, and an orderly response to inevitable change

You'll see how projects based on best practices in business requirements yield significant dividends. Risk becomes more manageable. Uncertainty vanishes. Technology aligns with objectives with more precision.

And yes, you can still speed deployment of even the most complex development initiatives.

» In most cases, the difference between order and chaos depends on a set of precise and verifiable business requirements.

## INTRODUCTION:

### *Business requirements in the real world*

#### Getting answers to critical questions.

If the thought of beginning a new initiative to optimize or adapt a business process makes you feel a little uneasy, there's good reason. The often-quoted statistic from a Standish Group study suggests that only 16.2% of software projects are completed on time and on budget – with only 42% deployed with all specified features.

What's even worse is that 94% of all projects without business requirements end up being restarted because conflicting dependencies and complex intricacies creates chaos that makes further progress impossible.

Yet many of these projects could have been saved if several critical questions had been addressed in advance. Whether redesigning a process or implementing a system, it makes little sense to begin a project unless you have answers to questions like these:

1. *Does this project require executive-level support?*
2. *Can I clearly articulate the real business drivers behind my idea – and present those ideas to my peers, sponsors, and subordinates?*
3. *Do I understand my potential return on investment and can I estimate the total cost of ownership based on my business drivers?*
4. *Have I identified all stakeholders who can influence my project?*
5. *Do I have a process for updating requirements, priorities, and rules as needs change – without adding cost and delay?*
6. *Is there a clear list of deliverables that a project manager could use to verify performance against requirements?*

These can be tough questions to answer, but those who take the time to consider them see tremendous benefits. On-time and on-budget deployments are actually the norm for organizations that maintain high discipline and focus.

It's clear that taking on a high-level process improvement initiative means taking on risk. But with a clear set of business requirements linked to strategic objectives, risk can be managed. No, the risk doesn't vanish, but with advance knowledge, there are few last-minute surprises. And that can make the difference between a smooth deployment and one that's filled with cost overruns, delays, and massive disruption.

### “Will this work?”

This is the question that’s on everyone’s mind. You’re hearing it from your colleagues and you’re posing the same question to your IT team, your vendor, and your systems integrator.

But without requirements, words such as “this” and “work” have little meaning. And if the question can’t be answered, there’s no basis for benchmarking and no way to create an effective testing strategy.

Of course the idea that poorly defined objectives leads to poor performance is certainly not news. Yet there’s always pressure to speed up deployment. If you delay too long, you lose your competitive advantage. In many cases, you simply can’t afford to wait until a project becomes a “sure thing,” so how do you balance the need for planning with the need get your project underway?

### A real-world approach to business requirements

At Brickwalk, our approach reflects a real-world view of requirements that recognizes the tradeoffs between speed and certainty. While it may not be possible to nail down every last business requirement in advance, planning for change — indeed, even embracing it — gives you more flexibility and minimizes disruption.

In the real world, change is inevitable, but those who request changes should also be aware of the expense. Changing a requirement during initial planning phases might cost \$500 and add no delay. During development, that cost could rise to \$5,000 and take an extra week. At deployment, the cost could soar to \$50,000 and add months to the schedule.

Is the benefit worth the cost? It depends. Nonessential changes may certainly be worth \$500, but not \$50,000. Naturally, the functionality of a critical change may outweigh the cost and delay, but it’s still frustrating to pay 100 times more to make a change simply because the requirements process broke down.

»» At Brickwalk, our approach reflects a real-world view on requirements that recognizes the tradeoffs between speed and certainty.

## THE THREE TYPES OF REQUIREMENTS

To align planning, development, and deployment, you'll need a set of three different types of requirements. Together, they facilitate communication between your business and technical teams, reducing the risk of errors and delays.

Although assembling this complete set of requirements may seem time-consuming, it's an essential step to establish strategic objectives, specify user tasks, and clarify functionality through requirements organized in these three categories:

- **Business requirements** define the specification of what you want to accomplish, usually expressed in terms of broad outcomes.

*For example: "Produce a report that shows all new customers within the last 30 days"*

- **User requirements** are captured in the form of use cases, describing tasks, rules, and processes from the perspective of the end user.

*For example, a user requirement based on the business requirement above might be:*

Step	Actions (External Events)	System Response	Decision
1	Manager submits request to the report engine to display all new customers within last 30 days	Report generated and displayed on manager's monitor. Display customer name, residence state, account creation date, salesperson	Success
2	Manager chooses the "download to Excel" button and exports data to Excel for further analysis	Export data to CSV file, prompt user for path to save file.	Success

Step	Alternate Actions	Alternate System Response	Alternate Decision
1		No matching records found	Failed

- **Functional requirements** describe what the process must do, so a developer can write verifiable code that fulfills the business requirement and meets the needs of the end user.

*For example, a functional requirement for the business requirement above might be:*

*"Display a named report 'Show all new customers within the last 30 days' in the list of available reports based on today's date. When selected, the report will open in Crystal Reports and can be downloaded into a CSV file and manipulated in Excel if the user chooses the download button."*

## BEST PRACTICES FOR REQUIREMENTS GATHERING

Since 1999, Brickwalk has helped organizations mitigate risk through an objectives-based approach to planning, testing, and deployment that includes:

**Defining objectives** to assure that business requirements are clear, understandable, and verifiable

**Improving process** through a framework for gathering, tracking, and managing requirements that keeps project risk to a minimum

**Assembling the right team** by staffing the project with professionals skilled in listening, translating objectives into requirements, and managing projects

### 1. DEFINING OBJECTIVES

*Requirements first, capabilities later*

Without clear requirements, gaps soon form between business strategies and the underlying code needed to bring your initiative online. You may not notice this gap at first, but odds are your developers will notice them when it comes time for testing and deployment.

When setting objectives, it's important to focus on what must be accomplished in a way that gains consensus from all project stakeholders. This is why effective business requirements must be:

**Precise:**

Ask yourself, would most people interpret this requirement in the same way that you do? If not, then it's probably not precise enough. Words such as "easy," and "quick" can be interpreted in many different ways, making it impossible to create clear functional requirements from such vague objectives.

» Without a detailed set of business requirements, gaps soon form between business strategies and the underlying code needed to bring your initiative online.

**Singular:**

Business requirements formed by run-on sentences with several “and” clauses may work fine to describe business objectives, but may not be related at all in terms of functionality. For example, a requirement might be written to clarify both the format and the schedule of a report. Because format and timing impact different portions of code, splitting such multi-part requirements into two or more separate requirements makes tracking and measuring performance easier as the project moves forward.

**Verifiable:**

Think down the line to testing your application. Can the requirement be verified? Segmenting loan applicants by creditworthiness may be the goal for a mortgage underwriting process, but without some metric (such as a FICO score, for example), it would be impossible to verify that requirement. And if a requirement can't be tested, it's of little use in determining whether the deployment will be successful.

**Consistent:**

If one business requirement details that one specific set of data be used to create a report and then another requirement adds more data to that report, then it's likely to cause development headaches. When such conflicts and inconsistencies are not addressed early on, it's extremely difficult to create effective functional requirements.

It may seem tedious to methodically work through a complex set of business requirements but it's an effort that's sure to pay off. With precisely defined requirements, developers will be more secure in their understanding of what needs to be accomplished. And you could easily save hundreds or thousands of development hours throughout the life of the project.

»» With precisely defined requirements, developers will be more secure in their understanding of what needs to be accomplished. And you could easily save hundreds or thousands of development hours.

## 2. IMPROVING PROCESS

### *From discipline comes flexibility*

Professor Martin H. Fisher observed that while knowledge comes from gathering information, wisdom lies in simplifying information.

Indeed, a complex process with multiple inputs will span across many business groups and infrastructures. Given all these intricacies, gathering requirements in an organized way provides more visibility into the process and gives all constituents the voice they need to evaluate use and functionality.

Multiple business managers usually have multiple goals. In many cases, they don't coincide. End users will have their own opinions and your IT teams are sure to weigh in. Therefore, a cooperative environment provides an improved framework for requirements gathering. Highly complex process issues are almost never clear unless you can see them from all perspectives. Getting buy-in from all key project participants is essential for a successful project.

For example, an airline might want to improve its process for handling baggage. What's wrong with the current process? What improvements should be made? How should the new process be different? Without the perspective of ticket agents, baggage handlers, and even customers — in addition to IT experts and managers — then process change becomes very difficult indeed. You could even end up making the process worse.

On the other hand, too much information tends to cause “analysis paralysis.” Enterprises are not democracies, so not every suggestion will be valid, realistic, or even useful. Without a framework for assimilating all information and putting it into the proper context, business requirements just never seem to get approved.

Change is inevitable, so embracing it makes sense. That means adopting specific procedures for establishing how changes will be evaluated, managed, and communicated to others.

At Brickwalk, we practice what we preach. Over time we have shown many executives the benefits of a framework for business requirements that's disciplined, yet adaptive to change. This framework is based on the lifecycle of five fundamental business requirements steps: CONFINE, CAPTURE, CLARIFY, CONFIRM, and CONTROL — an approach discussed in more detail on page 13.

### 3. ASSEMBLING THE RIGHT TEAM

#### *Navigating unfamiliar territory*

Leadership takes courage, especially for initiatives to develop or improve a major process in a large organization. Without leadership, decisions seem to take forever, major fundamental changes crop up with alarming frequency, and team members never get aligned toward meeting a singular goal.

At Brickwalk, we're often called in to help organizations stay focused on their objectives. The secret to doing so is to assemble a team that includes business requirements analysts with these attributes:

#### **Big-picture expertise**

Requirements for business systems can impact your organization far more than non-systems projects. While the goal for your initiative may seem simple enough, the potential impacts could surprise you.

For example, during one project, a Brickwalk analyst was asked to create an external interface to an existing system. But when exploring this request, he discovered this solution could be delivered using existing systems, functionality, and data – resulting in no additional cost.

That discovery was made early enough in the process to save more than \$100,000 and even accelerate deployment. Analysts who are experienced in big-picture thinking can certainly save you time and money, but more importantly, they can help you identify deployment risks early on to assure that the impact, if any, is minor.

#### **Inertial, proactive approach to leadership**

You could apply Newton's First Law of Motion to development projects. Projects at rest tend to stay at rest and projects in motion tends to stay in motion. Projects that get stalled based on a lack of agreement on requirements can be difficult to restart. In many cases, an outsider's

»» You could apply Newton's First Law of Motion to development projects. A project at rest tends to stay at rest and project in motion tends to stay in motion.

perspective can provide the inertia to keep project momentum moving forward. For example, at Brickwalk, we take a proactive role as needed – to arrange discussions, create agendas, make sure meetings stay focused, and coordinate the executive and user involvement it takes to develop effective business requirements.

Assuring that business units, IT teams, and other stakeholders stay “in the loop” is not easy, but consultants who do so play a vital role in keeping a lid on soaring costs and preventing unproductive “scope creep” that can be devastating to your ROI.

### **Thinking three steps forward**

When you work with excellent business analysts, you quickly start to suspect that they’re telepathic. No, analysts certainly aren’t mind readers, but good ones are naturally curious about the implications of what you’re asking for. Sometimes, an outsider can point out negative impacts on seemingly unrelated processes or suggest an alternate approach that can achieve the same objective more efficiently.

In other words, effective business analysts think ahead and anticipate needs. Ironically, while most business processes are expected to work at least 99.99% of the time, as much as 70% of the effort in creating business and functional requirements focuses on that other 0.01%.

It may seem paradoxical that an overwhelming amount of time is centered on exceptions – not the events that *should* happen, but the ones that *shouldn’t*. The question, “Well, what if it doesn’t?” is one that effective analysts are constantly asking themselves. Unanticipated exceptions can cause massive disruption, even more so if the exception doesn’t become apparent until well after deployment.

### **Superior translation skills**

To be effective, business analysts must be fluent in many business languages. Objectives must eventually be translated for engineers who will develop your code, but developers

»» Ironically, while most business processes are expected to work at least 99.99% of the time, as much as 70% of the effort in creating business and functional requirements focuses on that other 0.01%.

can't write code based on business strategies or desired user outcomes. For example, during a mortgage application process there may be a need to "quickly inform the account representative and the applicant if a walk-through, rather than a statistical appraisal is required." But a software engineer can't turn this into the code until they understand all the details. What specific information does the account representative need? What does "quickly" really mean?

Business requirements analysts who are equally conversant in both the language of business and the language of IT will be of great help to you because they can bridge the gap between what you really want to accomplish and the specific functionality it takes to make it happen.

### Questions to ask during business requirements planning

- **Are requirements specific enough** to be tested with certainty, or will the results be inconclusive?
- **Does everyone understand the requirements** in a similar way, or do they seem vague and subject to interpretation?
- **Are the requirements properly classified** so that each has a set of underlying user and functional requirements?
- **Are requirements assigned a priority level** to avoid "scope creep" and to make sure the most urgent needs are addressed first?
- **Is there a change management process** in place to make sure that changes are properly documented and communicated to the entire project team?
- **Are project leaders experienced** in requirements gathering and analysis, or will outside consultants be needed?

## THE REQUIREMENTS LIFECYCLE

### *Five fundamental steps*

Requirements follow a lifecycle that closely tracks the path of your project. At Brickwalk, we view this lifecycle in terms of “5 Cs”— **CONFINE, CAPTURE, CLARIFY, CONFIRM,** and **CONTROL**. Based on our experience in requirements gathering and management, this framework is an excellent approach that balances the discipline needed to manage progress with the flexibility needed to accommodate change.



#### STEP 1: CONFINE

##### *If there's no scope, how do you know if it creeps?*

Every process has a purpose, but it's easy to lose sight of those core objectives as the project moves forward. Those who develop and integrate software will tell you that they sometimes have absolutely no idea how the code they're writing fits into the overall project because of little documentation of purpose and scope.

When requirements are continually added to a project, “scope creep” sets in. That may be acceptable, provided that senior project managers understand all the implications to cost and schedule. But without a scope to begin with, there's no way to determine whether or not it has “creeped.” A properly defined scope focuses on objectives in their proper priority, taking potential assumptions and dependencies into account.

#### Five fundamental business requirements steps

**1. CONFINE:** Establish project scope and purpose, analyze potential assumptions and dependencies, and assure that all constituents understand goals.

**2. CAPTURE:** Create clearly defined process for gathering requirements based on user needs, usually through individual and group interviews.

**3. CLARIFY:** Establish priority level for all requirements to assure that low-level needs don't interfere with critical capabilities, and that requirements can be tested and verified.

**4. CONFIRM:** Review requirements with all process owners, step owners, representative end users, executives, and project sponsors to obtain needed signoffs and approvals.

**5. CONTROL:** Adopt a disciplined, yet flexible approach for managing and documenting all changes to requirements using collaborative change management tools throughout the project lifecycle.



## STEP 2: CAPTURE

### *You don't get requirements unless you ask*

Business analysts often speak of “gathering” or “eliciting” requirements. That’s because requirements rarely appear spontaneously – you have to ask for them, sometimes more than once.

The first step to requirements gathering identifies stakeholders and various classes of users impacted by the process, followed by a series of highly focused group discussions and individual interviews with representatives from various constituencies. Live meetings are more effective than surveys or conference calls. At times, these sessions can go off on an unproductive tangent, but experienced facilitators can help you uncover some revealing insights into your processes.

It’s extremely important to maintain an environment where everyone’s opinion is valued and welcome – and that there’s a clear end in sight. If participants start getting the sense that time is just being wasted, progress can stall. But individuals will be happy to participate if they see project momentum and that their needs are being addressed.



## STEP 3: CLARIFY

### *Turning input into actionable insights*

After a series of input sessions, you’ll end up with many potential requirements. Some will clearly fall within the scope of the project, while others will not. And there will surely be a category of requirements you’re uncertain about.

To clarify which requirements are actionable, they should be organized into a comprehensive document that includes a brief outline of the process, a summary of the major requirements categories, and a list of all requirements organized by those categories and by priority.

Assigning a priority level to each requirement allows you to evaluate the optimum solution, with all “must haves” deployed and as many “nice to haves” as possible. Although users and managers may not see eye to eye on what’s essential and what’s not, this objective approach to priorities helps everyone see the big picture.

At this stage it’s important to examine the intricacies of connections and interrelationships among different requirements. In some cases, deleting or revising one requirement may call for revisiting others. Knowing about these intricacies as early as possible helps mitigate the risk of project delay or failure.



#### STEP 4: CONFIRM

##### *Are these requirements correct?*

The critical question in the end is, “Will this work?” At this stage you may not know for sure, but it’s important that you can answer, “How can we be sure?” If there’s no way to verify a requirement, there’s no way to test it during deployment.

An effective approach to confirming business requirements is to create a series of use cases that examine how users will interact with the process. For example, you might develop a series of use cases for a loan application process that’s based on how loan officers will interact with the system, how underwriters will use it to make decisions, how managers will oversee the process, how customers might submit applications, and so on.

Confirmation is an essential step to make sure your initiative maintains a focus that meets actual needs. If that focus gets lost and you don’t learn about it soon, you can end up with a fragmented and incoherent architecture or expensive functionality that nobody even asked for.



#### STEP 5: CONTROL

##### *Enforcing discipline and focus*

While it would be convenient if major IT initiatives always followed a linear process, that’s an unreasonable expectation. So if change is inevitable, it makes sense to plan ahead. With an effective change control process you can be more confident that any adjustments to requirements are managed with minimal impact.

Because requirements for complex processes form an intricate web of interrelationships, a collaborative online tool is an excellent method to track and analyze them. High-level concepts can be matched to scope items, then mapped to individual requirements, and then mapped to corresponding functions.

At Brickwalk, we use an Internet-based requirements repository that employees, clients, and partners can access throughout the project lifecycle. With a full record of all changes, when they were made, who requested them, and their impact to the schedule and budget, projects can be managed with both discipline and flexibility.

And at the end of a project, you’ll have complete documentation of each function that was built into the system, its source, how it was tested, who approved it, and the reason that any specific requirements were not fully addressed.

## A SMOOTH DEPLOYMENT: *Planning against failure*

Race car driver Mario Andretti once observed, “If things seem under control, you are just not going fast enough.” Indeed, large enterprises are so used to high failure rates, it seems rather odd when a project is fully deployed with little disruption.

Ironically, a smooth deployment can leave you feeling that you could have done more — explored more options, added more features, deployed more rapidly, etc. But we view it differently. A smooth deployment demonstrates that you managed the risk of process change in a way that assured a successful outcome.

With the right approach to business requirements planning and a management style that’s based on purpose, process, and people, you’ll learn what we already know at Brickwalk — that on-budget, on-time deployments really **can** be the norm. Requirements planning may seem cumbersome at first, but it always pays off in dramatically improved processes and a higher return on investment.

» Large enterprises are so used to high failure rates, it seems rather odd when a project reaches full deployment with little disruption.

## **A TYPICAL REQUIREMENTS PROJECT:**

At Brickwalk, a normal business requirements gathering project takes approximately 4 to 6 weeks for medium to large projects, or as little as 2 to 4 weeks for smaller initiatives.

The process begins with a series of meetings with our business analysts and engineers to identify overall objectives, understand the impact on existing processes and resources, and create a plan for making the most efficient use of your time.

After a series of interviews with relevant groups and individuals, we develop an initial set of requirements and relevant diagrams to circulate among key managers. Then, during follow-up meetings, we prioritize requirements, clarify outstanding issues, and deliver a comprehensive set of business requirements so project sponsors can sign off on them. Once approved, return on investment and total cost of ownership calculations are evaluated to make sure costs and schedules meet expectations.

Upon approval, we develop a comprehensive set of user requirements and use cases (including business rules). Once user requirements are approved, functional requirements and proofs of concept are developed using the actual technology you've selected.

This approach assures that planning and specifications are detailed and complete, that critical functionality is not overlooked, and that testing can be quickly completed. Through a disciplined approach to requirements, you can save hundreds of hours of development and testing time – and focus more effectively on your core strategic goals.

## **ABOUT BRICKWALK:** *Artisans of systems integration*

Brickwalk specializes in helping organizations improve processes and reduce the risk of implementing complex IT initiatives.

As artisans of systems integration, we offer insight, creativity, domain expertise, and project discipline to make certain that your solution is properly aligned to your business objectives.

Our practice areas include:

- Business Requirements
- Project Management
- Development
- Infrastructure Engineering
- Long-Term Support

With a thorough background and expertise in both the art and science of systems integration, Brickwalk can help you understand your processes and why they don't work properly, show you new opportunities you may not have considered, and steer you clear of potential roadblocks.

To learn more about Brickwalk, contact Bill Masurat by email at [bill@brickwalk.com](mailto:bill@brickwalk.com) or by phone at **(314) 822-7494**.

### **Brickwalk Offices**

#### **San Francisco**

425 Market Street,  
22nd Floor  
San Francisco, CA 94105  
(415) 294 4613 Local  
(800) 495 5779 Toll Free

#### **Boston**

60 State Street,  
7th Floor  
Boston, MA 02109  
(617) 648 4475 Local  
(800) 495 5779 Toll Free

#### **St. Louis**

Two City Place Drive,  
Suite 200  
St. Louis, MO 63141  
(314) 414 0235 Local  
(800) 495 5779 Toll Free

#### **800-495-5779**

[sales@brickwalk.com](mailto:sales@brickwalk.com)  
[www.brickwalk.com](http://www.brickwalk.com)