
Amplifying Agile Practices with Business Analysis

by Ellen Gottesdiener and Mary Gorman

© EBG Consulting, 2013

(initially published in the IIBA Journal March 2013)

EBG Consulting, Inc.



www.ebgconsulting.com

www.DiscoverToDeliver.com

When software development teams adopt lean/agile principles, they focus on delivering a product that yields the highest value in the shortest possible time. In this effort, they soon discover that efficient and effective business analysis practices are essential throughout the process.

This article explains why, recapping key concepts you'll find in our book *Discover to Deliver: Agile Product Planning and Analysis*. Then you'll get a detailed example (based on a composite of our real-world projects) of a team using business analysis to explore and plan delivery of one increment of product requirements.

We call this process a “structured conversation.” It’s a framework you can adopt and adapt for using essential business analysis practices to make value-based product decisions, building a high-quality product that integrates requirements verification and validation and engages stakeholders in mutual learning.

The structured conversation addresses one of the biggest challenges we’ve observed in working with agile teams for more than ten years: how to efficiently slice requirements into valuable chunks for short delivery cycles.

We’ve observed this struggle whether agile teams use timebox delivery (e.g., Scrum) or flow delivery (e.g., Kanban). Slicing requirements—also referred to as refining, grooming, partitioning, or decomposing requirements—compels the team to explore, evaluate, and then confirm the high-value requirements options. This business analysis work is crucial, no matter what specific role or title a business analyst is assigned within the agile team. After all, it’s the goal, not the role (Gottesdiener and Gorman, 2011).

The structured conversation is vital for achieving a key benefit of agile: delivering the high-value portions of a product, fast. In this article, we assume you’re familiar with fundamental agile principles and practices. The practices we describe are neutral of agile flavor. They work whether you’re using any combination of Scrum, Kanban, DSDM, eXtreme Programming, and so on.

You will learn methods agile teams can use to determine business value and view requirements holistically. You'll realize how crucial it is to engage stakeholders as partners to make smart decisions and act collaboratively. And you'll see how the work of business analysis is essential to discover and deliver a high-value product.

Essentials of Agile Business Analysis

Any delivery you plan must be desirable to customers, must be feasible to build and support, and must fit within the organization's strategy and product portfolio. This means that the people doing analysis on agile projects must understand the bigger context while driving toward specificity.

To do this important work, ongoing stakeholder engagement is imperative. Stakeholders act as partners, collaborating in healthy and transparent ways to discover product needs in the most streamlined manner that yields valued results.

In a lean/agile project, product partners analyze requirements in short, highly focused time increments. By delivering only what is necessary and sufficient to ensure a valuable, high-quality product, you reduce the requirements risks that plague all software development projects. These risks fall into two basic categories: building the product wrong, and building the wrong product.

To ensure that you’re building the product right, you define unambiguous, testable acceptance criteria for the increment being planned. This process, known as verification, is baked in to the

analysis process and not tacked on at the end. Verification falls squarely into the bailiwick of business analysis.

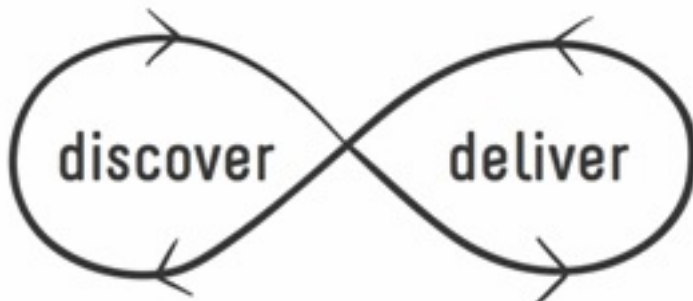
Business analysis also accelerates validation, a process you use to confirm that you're building the right product. Agile development amounts to a series of small investments, each one a delivery cycle in which you discover and deliver a product increment. After each delivery, you validate whether the evolving product is returning its expected value.

Let's explore some important concepts for agile product planning and analysis, and then dive into how the structured conversation works.

Concepts in Product Planning and Analysis

Lean/agile is a method of developing software-intensive products through the ongoing, interwoven activities of discovery and delivery. By "agile," we mean the family of practices for software-intensive products using iterative and incremental development. "Lean" practices focus on maximizing customer value while minimizing waste. We use "agile" as an umbrella term that incorporates both agile and lean practices.

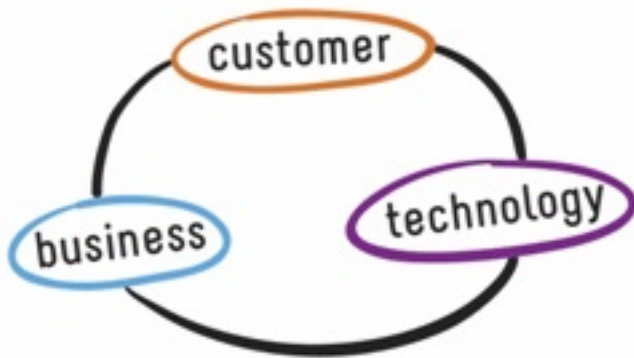
Figure 1: Discover to Deliver Value



The planning for agile discovery and delivery is driven by the anticipated value of the product to the organization, whether the product is intended for internal use, an external party, or the commercial market. Because agile planning is driven by value, the focus is on the product rather than the project.

The product's stakeholders form a product partnership that consists of representatives of three groups: customer, business, and technology. The customer group includes users, buyers, and advisors. Business partners include the sponsor, product champion, product providers and servicers, and advisors who are subject matter experts from various business function areas. The technology group includes the delivery team and technical advisers. As mentioned earlier, partners from all of these groups perform business analysis to varying degrees.

Figure 2: The Product Partners



Working collaboratively under the leadership of a product champion, the product partners offer the perspectives of their disciplines as the product is elaborated through successive releases. They identify the expected value of the product they've planned, and they continually revise their value assessment based on the results of previous deliveries. By confirming the product's results after delivery, the partners learn and increase their efficiency over time, a process known as validated learning.

The product partners first agree on a long-term vision for the product that aligns with the organization's strategy. The partners also identify goals (such as increasing revenue or reducing costs) and objectives (increase revenue 10 percent in six months; reduce costs 15 percent in one year) for the product. Because the product evolves during discovery and delivery, agile is designed to respond to any changes in the vision, goals, and objectives over time.

A product is made up of product options (or requirements), as expressed in stories, text descriptions, and so on. Each product option presents benefits and risks, and the partners assess these as part of determining the option's value. The partners' perspectives reflect their value considerations. For example, customers may value convenience and cost savings; business partners may value alignment with the vision, market positioning, and resource needs; and technology partners may value feasibility, service costs, and quality attributes.

We've found that teams' discovery work is more efficient when they consider product options according to the planning horizon and level of detail they represent: long-term (the general idea of the product), medium-term (the next release), and near-term (the next iteration). We refer to these time horizons as views: the Big-View, the Pre-View, and the Now-View, respectively. As the team progressively elaborates high-value options, the options pass through the three views more than once.

The team analyzes each option using the 7 Product Dimensions. For each option, the team explores each applicable dimension: what users have goals related to this product? What interfaces are needed? What actions are needed? What data are acted on? What controls must be enforced? What environments will the product operate in (and be developed in)? What quality attributes constrain and control the product?

Figure 3: The 7 Product Dimensions

User	Interface	Action	Data	Control	Environment	Quality Attribute

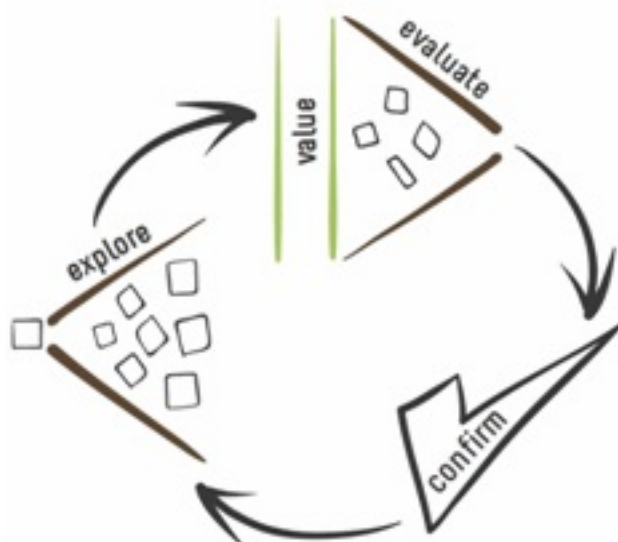
By answering these questions, the partners gain a holistic, comprehensive understanding of the product. Note that the dimensions cover functional (users, actions, data, and controls) as well as nonfunctional (interfaces, environments, and quality attributes) dimensions. The 7 Product Dimensions construct also gives the partners a visual language to use in communicating about the product.

The structured conversation is a lightweight framework the partners use to collaboratively explore, evaluate, and confirm high-value product options across the 7 Product Dimensions. The structured conversation employs essential business analysis practices, including elicitation, analysis, and specification.

Here's what the structure looks like. For each of the 7 Product Dimensions, the partners

- Explore the options.
- Evaluate (slice) the options, selecting a subset of options based on value and dependencies. They then assemble the selected options into a cohesive set and document it in the form of a story, a feature, or a minimum marketable feature, depending on the planning horizon. The closer they are to delivery, the more detailed the description will be.
- Confirm their understanding and expectations—and prepare for delivery—by specifying the criteria they'll use to verify and validate the requirement.

Figure 4: The Structured Conversation



The Structured Conversation: An Example

The partners' goal in the structured conversation is to discover the next most valuable product options. As they converse, the partners use a wall poster (the Product Options board) to organize and display the 7 Dimensions. They use lightweight analysis models and solidify their understanding with examples. In our experience, teams hold structured conversations easily and quickly once they get the hang of it.

Let's dive into an example using the fictitious company Squeaky Kleen (SK), which provides window-cleaning services. In our book *Discover to Deliver: Agile Product Planning and Analysis*, the first section provides a rich dialog of the SK team as it explores, evaluates, and confirms the needs of a product it's building to streamline SK's support functions.

The team's planning decisions are based on SK's vision ("To be a trusted partner for reliable, safe, and competitively priced cleaning"), goals (including "Obtain efficiencies in scheduling"), and objectives (including "Reduce the time it takes to schedule a cleaning service by 20% in the next 2 months"). They also focus on the partners' value considerations.

Customer Value Considerations

- **Save time and frustration:** Scheduling window-cleaning services is quick and easy
- **Convenience:** Job is scheduled according to my personal needs

Business Value Considerations

- **Maintain SK's reputation for personal service** while improving efficient workflow for staff
- **Reduce customer service reps' training time** to save overhead

Technology Considerations

- **Support a variety of software tools** for SK staffers to improve efficiency
- **Have a scalable technology platform** to support growth

Spoiler alert! At the end of the conversation, the team summarizes its results with the following story: "As a Junior Customer Service Rep (CSR), I need to schedule a residential window-cleaning job for an active, individual customer so the customer is confident their request is completed accurately."

Exploring and Evaluating Across the 7 Product Dimensions

Here's a sampling of a structured conversation.

The team started with the feature "Optimize Scheduling," quickly exploring and evaluating numerous options for each Product Dimension.

User Options

The team analyzes the users who interact with the product by asking focus questions:

- Who initiates actions, and what are their goals?
- Who receives output from the product?
- Who participates in fulfilling actions?

The team explores the possibilities and identifies five users:

User Options

- Cleaner
- Customer
- Customer Service Rep (CSR)
- Inventory Manager
- Sales Manager

Evaluating high-value users for the next planning cycle, the team reviews the elements that comprise value for the product (vision, goals, objectives, benefits, risks, and value considerations). They will use these same elements throughout the structured conversation. For this iteration, the team selects two user options: Customer and CSR.

For each selected user, they ask a focus question:

Are there specific life cycle states?

The team discovers three user states for the CSR.

User State Options (of CSR)

- Senior
- Junior
- Trainee

Because junior CSRs do most of the routine scheduling, the team determines that the high value for the next delivery cycle is to focus on them.

Note that the conversation results in eliminating options. Agile thinking entails deciding what not to do—and that might be as important to achieving value as deciding what to do.

Interface Options

The team analyzes the interfaces that connect the product to users, systems, or devices:

What interfaces are needed to send data or messages to or from the product?

They focus on the interface options relevant to the high-value user (junior CSRs) and then consider interface types (user interface, system-to-system, hardware device). They identify the high-value interface options (indicated with an asterisk below), and mark them on the Options board.

Interface Options

- Customer provides service request details
 - Phone (manual)*
 - Website (UI)
- SK sends the customer a job confirmation
 - Email (system to system)*
 - Text message (system to system)

Action Options

To identify the product's action options, the team explores the activities that achieve needed results:

- When does this action occur?
- How is it triggered?
- How does it respond?
- What are the action's preconditions?

Action Options

- Schedule a job
- Reschedule a job
- Cancel a job

The team pinpoints the objective "Reduce the time it takes to schedule a cleaning service by 20% in the next 2 months." Because of the complexity of rescheduling or canceling a job, the team defers those options. They then explore scheduling actions in detail. They identify the essential options for the next delivery cycle, and the product champion marks them.

Action Options for Schedule a job

- Verify requested location is within SK service area*
- Verify requested service is provided by SK*
- Calculate price for service*
- Confirm price with customer
- Find available cleaner to perform the service*
- Reserve any special equipment
- Send job confirmation to customer*
- Provide cancellation policy to customer

Data Options

The team discovers the data needed by the product, asking:

- What data do users need from the product?
-

- What data does the product receive, and what data must it store?
- What are the relationships among the data options?

They've determined the high-value users and actions, so they focus on the data to support them: Customer, Job, and Property Type. For each data option, they ask:

Are there unique types or variations? For example, a customer may be an individual or an organization. Property types include residential and commercial.

As with the user options, they ask:

Are there specific life cycle states? Customers may be "active" or "inactive."

The product champion marks the high-value data.

Data	Type	State
Customer	individual* organization	active* inactive
Job		
Property Type	commercial residential*	

Control Options

The team explores the control options the product enforces (policies and business rules).

- Are there control options unique to high-value users?
- Are there control options that must be enforced by high-value actions?
- Are there specific control options for the high-value data?
- Are the controls part of a set?

The team weighs the value of these options and marks the high-value ones.

Control Options

- A customer who has not paid for prior services may be prevented from scheduling new services
- A requested service must match a service currently offered by SK*
- A cleaner must not be double-booked*

Environment Options

To learn options for the product's physical properties, the team asks:

- What are the locations where users will operate the product?

- What are the physical conditions surrounding users when they interact with the product?

The team discusses and evaluates the options.

Physical Property Environment Options

- CSR working off-site
- CSR working in SK office*

The team briefly discusses another aspect of the environment dimension: the product's technology platform options (software, hardware, and standards used to develop, deploy, and operate the product). They agree to discuss these options in their next session.

Quality Attribute Options

The team considers the properties that qualify the product's operation and development (availability, efficiency, performance, safety, security, usability, etc.):

- Which quality attributes are needed to support the user, interface, action, data, and control options?
- Does a quality attribute apply across all dimensions and options, or is it specific?

The team explores and evaluates the options.

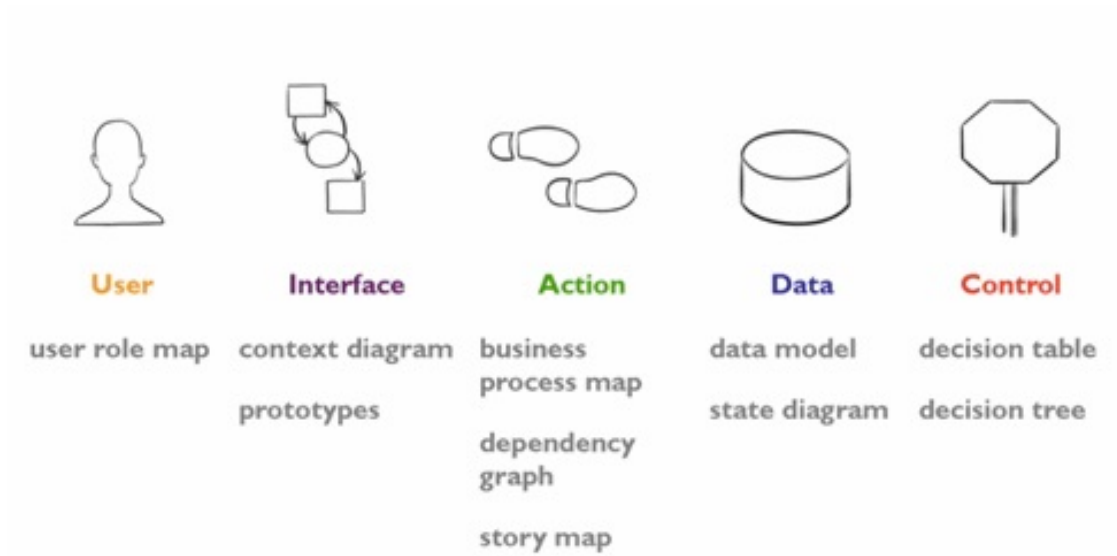
Quality Attribute Options

- **Availability:** The SK scheduling application will be available 95% of the time Monday through Friday from 8:00 a.m. to 5:00 p.m. Eastern time.*
- **Performance:** The calculated price for requested services shall be displayed within two seconds of completing entry of the service type and count.*

Exploring and Evaluating Across the 7 Dimensions

The team steps back and reviews their Product Options board. Their analysis of the feature "Optimize Scheduling" quickly uncovered numerous options for all 7 Product Dimensions. The few lightweight analysis models they sketched on the board helped visualize relationships, dependencies and flow.

Figure 5: Analysis Models



Their evaluation identified the high-value options within each dimension. Now they needed to evaluate the options' collective value – across the 7 Dimensions. Focusing on the next planning horizon, they made decisions and select a cohesive set of high-value options. (Any options not chosen this time are saved and reconsidered in future planning sessions.)

Product Dimension	"Optimize Scheduling" High-Value Options
Users	Junior CSR*
Interfaces	Customer provides service request details, Phone* SK sends the customer a job confirmation, Email*
Actions	Verify requested location is within SK service area* Verify requested service is provided by SK* Calculate price for service* Find available cleaner to perform the service* Send job confirmation to customer*
Data	Active, individual customer* Residential property*
Controls	A requested service must match a service currently offered by SK* A cleaner must not be double-booked*
Environments	CSR working in SK office*

<p>Quality attributes</p>	<p>Availability: The SK scheduling application will be available 95% of the time Monday through Friday from 8:00 a.m. to 5:00 p.m. Eastern time.*</p> <p>Performance: The calculated price for requested services shall be displayed within two seconds of completing entry of the service type and count.*</p>
----------------------------------	---

But Wait! It Isn't Over Until You Confirm

Before the conversation concludes, the team specifies the acceptance criteria for assessing whether the developed candidate solution accomplishes the stated objectives.

Acceptance criteria list for “Schedule a residential window-cleaning job”

- The date and time for requested schedule date are valid.
- Job date and time match customer's requested date and time.
- When there is a match (requested date and time can be scheduled), assign an available cleaner to the job and verify:
 - Job data is stored.
 - Confirmation message is provided to the customer with the details shown.
- When there is no match, display a list of closest available dates and times.

These acceptance criteria clarify and crystallize the team's understanding and guide development and testing during delivery. (Other confirmation tools include data examples, Given-When-Then, Planguage (Gilb, 2005), and quality attribute scenarios.)

Recapping the Structured Conversation

Explore, evaluate, confirm. The team uses the structured conversation framework to collaboratively analyze “Optimize Scheduling.” They explore and evaluate options within and across the 7 Product Dimensions. They confirm the candidate solution. Their output – stories, analysis models, and acceptance criteria – guide product delivery.

Agile Business Analysis in a Nutshell

Agile principles applied to business analysis can be summarized as follows:

The product partners continually collaborate to discover and deliver the evolving product. The partners explore and evaluate options for the 7 Product Dimensions. They identify and confirm cohesive chunks of high-value options and allocate them to plans at all views.

Good business analysis is essential to—and amplifies the benefits of—agile development practices. Using skilled business analysis helps you improve team communication, planning, learning, and efficiency over time. You furnish evidence that the product is fit for use and generate valuable documentation. Most important, you reduce the insidious risk of building the product wrong or building the wrong product.

Agile—done well—embraces business analysis to accelerate elicitation, verification, and validation. Thus, agile business analysis is essential for discovery and delivery of high-value products.

References

- See: <http://www.discovertodeliver.com/resources/references.php>. (This link includes primary references for Discover to Deliver as well as recommended reading on concepts of planning, products, the structured conversation, and value.)
- Gilb, Thomas. *Competitive Engineering: A Handbook For Systems Engineering, Requirements Engineering, and Software Engineering Using Planguage*, Butterworth-Heinemann, 2005.
- Gottesdiener, Ellen, and Mary Gorman. “[It’s the Goal, Not the Role: The Value of Business Analysis in Scrum](#),” StickyMinds (June 28, 2011).