

The Language of SUPPLY CHAINS

By Corey Billington

Around the world, manufacturers are looking for ways to characterize and refine the order-fulfillment processes they have developed for their products. To supplement technical and analytical approaches, this article proposes an innovative technique that builds on the analogy of natural language. Here we develop and illustrate a descriptive grammar, if you will, for designing the supply chain that makes the best sense for the product. It can be refreshing as well as useful to think of the optimal supply chain in terms of syntax, semantics, and style.

The language of supply chains has grammatical rules and syntactical features. One of the most important of these is the "push-pull boundary"—the pivotal point at which product goes from being *pushed* in anticipation of customer orders to being *pulled* by actual demand. The linguistic approach outlined here helps managers understand these rules. And it allows them to explore supply chain alternatives that make the best economic sense for their products while delivering value to the customer.

Corey Billington is director of strategic planning and modeling at Hewlett-Packard. His articles have appeared in the Sloan Management Review, Operations Research, and other publications.

In our view, a supply chain grammar would consist of syntactical units for tasks, activities, and processes, along with rules for combining them. The semantics of the supply chain—"the grammatical sequence that makes the best sense for the product"—would be the value created by achieving maximum efficiency. Our fundamental point is that using excellent grammar in designing a supply chain ultimately has an *economic* meaning. This approach allows us

to analyze supply chains, examine their structure, develop new designs, share insights, select the design that creates the greatest value, and improve order fulfillment using a creative methodology.

In formulating a supply chain grammar, the most important syntactical feature is something we call the *push-pull boundary*.¹ This inflection point is where demand information—the actual customer order—exerts its influence on manufacturing. Inventory needs to change dramatically depending on the push-pull boundary, since some version of the product (or part) must be available to be processed and delivered when orders are received. Before this point, the product is *pushed* into a state of readiness in anticipation of customer orders. After orders are received, the product is *pulled* by demand—preferably with the speed and reliability needed to satisfy customer requirements.

In terms of language, the push-pull boundary is the point at which the audience for an utterance—that is, anything spoken—begins to determine its shape and meaning. The place where something pushed begins to be pulled by demand should be positioned to balance the costs of holding inventory against the need to serve customers quickly and reliably. Having a grammar enables us to revise the supply chain until the most economically advantageous position for the push-pull boundary has been found.

To illustrate the application of supply chain grammar, we develop a simple lexicon for restaurant types and use this language to describe different approaches to value creation in that industry. Then we present a more complex example from electronics—the manufacturing, localization, and delivery of printers by Hewlett-Packard. In each case, a standardized grammar enables a systematic exploration of order-

fulfillment characteristics. The linguistic metaphor, moreover, allows us to envision new structures and value delivery systems that may be more effective than those currently employed. Throughout this discussion, we draw upon suggestive conversations we have had with Brian T. Pentland, author of *Grammatical Models of Organizational Processes*; Robert Sutton, professor of industrial engineering at Stanford University; and others.

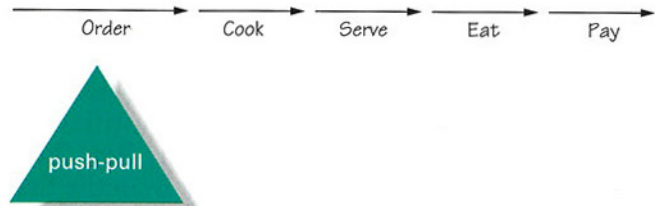
The Lexicon of Fine Dining

Let's begin by considering fine restaurants, where customer preferences for quality and service are of paramount importance. An establishment wishing to compete on the quality of its cuisine would begin with a familiar series of processes, the lexicon of fine dining. Following Pentland, we can think of these lexical units as "moves" involving restaurant staff and customers. Of course, many moves are made in a sophisticated establishment, but at a basic level the lexicon can be broken down into the following verbs:

- **Order**—The customer selects from a menu.
- **Cook**—The chef prepares and cooks the meal requested.
- **Serve**—The restaurant staff serves the meal.

- **Eat**—The customer consumes the meal.
- **Pay**—The customer pays the bill.

On the following timeline, we see that most activities are performed *after* the order has been taken. Customization and service (measured by customer satisfaction) are hallmarks of the system, though customers must be prepared for a significant wait before the meal arrives. For better or worse, the push-pull boundary is positioned very early in the supply chain design:



This grammatical sequence—Order, Cook, Serve, Eat, Pay—supports a system in which value is created through meal variety in response to individual requests; it's the classic *build-to-order* situation. However, there are opportunities for variation—for example, *process reversal* (a standard appetizer served before the customer orders) or *postponement* (pastries and toppings prepared in advance for custom desserts).

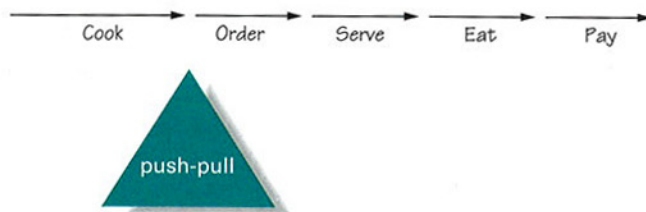


ILLUSTRATION BY KATHERINE MAHONEY

■ LANGUAGE

Furthermore, we may be able to change the push-pull boundary to enhance the order fulfillment and economic behavior of the supply chain. Alternative designs might create greater economic value and make the restaurant more competitive.

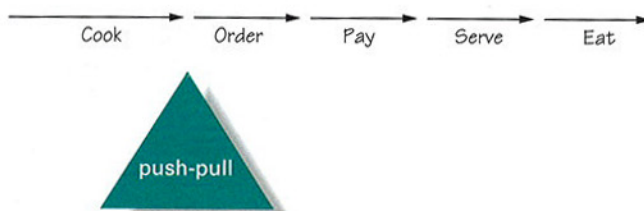
An interesting variation, for example, would be an establishment that prepares meals in anticipation of customers. Some restaurants specialize in cuisine—from prime rib to Peking duck—that takes more time to prepare than customers have time to wait. In such cases, the push-pull boundary for specific meals is different—since cooking is done before ordering—and inventory (in the form of cooked food) is put at risk. But a sufficient number of reliable customers for a specialized cuisine often justifies the sequence of Cook, Order, Serve, Eat, Pay. This reversal moves the push-pull boundary from the first position to the second, as shown below:



This new syntax often involves greater expense (more perishable inventory at higher unit cost) and more inventory held at risk (cooked food with a shorter shelf life than uncooked food). But the supply chain supports a value proposition rarely available. And there may be an advantage in cooking at a lower cost before customers begin ordering—and waiting. Many successful restaurants repeat variations on this theme to keep customers returning for the fine-dining experience.

The Syntax of Fast Food

Against fine-dining establishments offering a high degree of customization and personal service (along with high prices), we can compare the fast-food industry. The most familiar category of quick-service restaurants can be characterized as “eat last”—a theme that applies to the entire menu. The syntax looks like this:

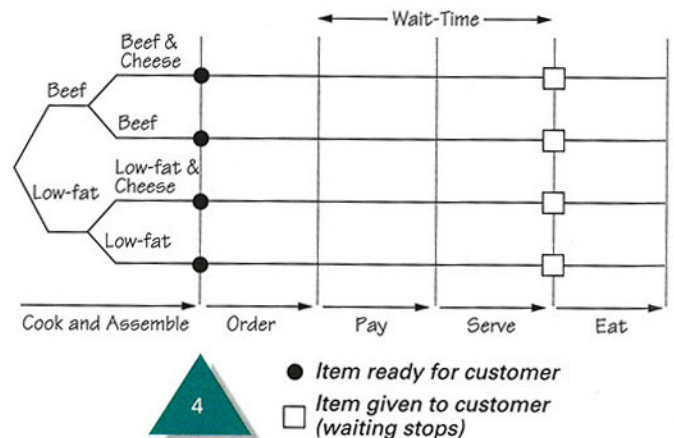


This approach involves not only cooking all meals in advance, but also disengaging from the customer as soon as possible. A clear example would be McDonald's at noon. Later in the day the restaurant might return to the sequence of Order before Cook, but at all times eating comes last. Note that the final transaction between the customer and the restaurant occurs *before* the meal is eaten. After payment, all that remains for the restaurant is to clean the tables and empty the trash—and by then the customer has disengaged from the supply chain completely.

The usefulness of supply chain grammar can be observed by extending this example. Many fast-food restaurants begin with the template just described but introduce variations to attract customers and keep them coming back. In all cases, however, the grammar and syntax of fast food illustrate the importance of the push-pull boundary, especially its impact on efficiency, customer satisfaction, and expected profitability. To demonstrate the point, let's pursue the most efficient design for a restaurant that might satisfy the needs of a fast-moving business crowd.

Burgers on Demand

As an exercise in supply chain design, we begin working with the syntax of a place called Burgers on Demand. Our value proposition states that customers can enjoy any of four burgers on demand—hamburgers, cheeseburgers, low-fat (vegetarian) burgers, and low-fat burgers with cheese. As these meals are intended to be cooked and preassembled, the supply chain looks like this:

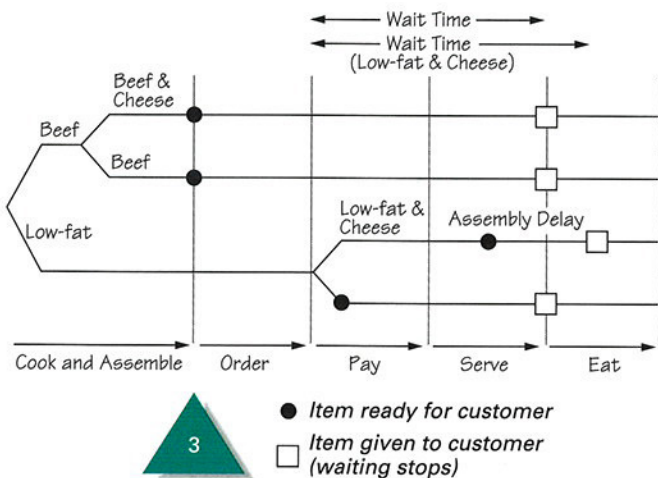


The exercise is like diagramming a sentence (and the weaknesses of the design appear as we examine its syntax). First, the push-pull boundary is placed where no fewer than four perishable items (the number inside the triangle) must be stocked and replen-

ished continuously to satisfy customer orders. There are economic consequences to this approach—the more stock-keeping units needed before the push-pull boundary, the more inventory write-offs to be expected. Second, notice that all products must be assembled and customized before orders are received. Finally, the push-pull boundary initiates a process of ordering and paying that must be aggregated as *customer wait time*. Since people are very sensitive to the time they spend waiting, we can expect some customers to become impatient and avoid repeating the experience. As the diagram makes clear, this design has its limitations.

Looking for ways to improve the design, we notice that all manufacturing takes place before the customer orders. Patties for the four burger types must be cooked in advance, and cheese needs to be added in two instances. Finally, sauces, condiments, and a bun must be added to each of the four products before making them available to customers “on demand.”

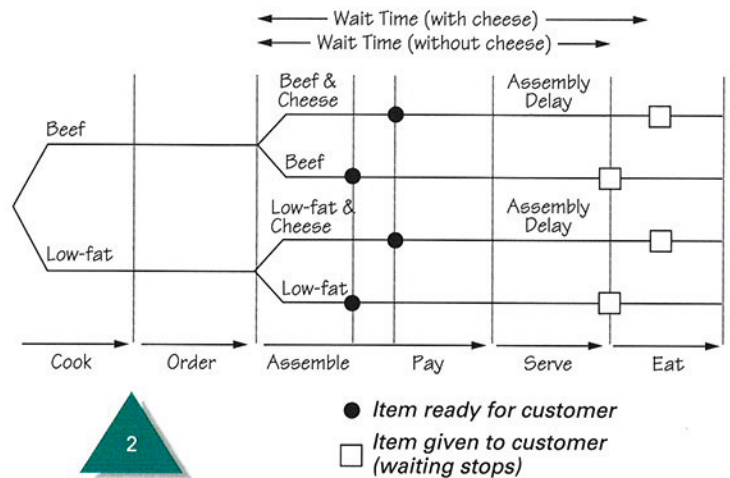
To reduce inventory requirements at the push-pull boundary, we keep ordering at the point where the patties have been cooked, but delay the addition of cheese to the low-fat burger until *after* orders are received. Since cheese is high in fat, we realize that this may be our least popular selection. (Here we begin to appreciate how product design plays a crucial role in determining supply chain efficiency.) The advantage of these changes is that we stock only three items in inventory and postpone customization of one product (low-fat burger with cheese) until it has been ordered.



In this design, cheese is added to the low-fat burger after the push-pull boundary. Since the probability of orders for these burgers may be low, we can tolerate a delay for orders including this selection. If the addition of cheese takes longer than the

“pay” time, the customer wait time will increase for the fraction of orders that include the low-fat burger with cheese, but remain the same for the majority of orders. (The potential delay in this process is indicated in the chart.)

Noticing the advantages, we consider further postponement opportunities—cooking the patties in anticipation of customer orders, but delaying all other steps until after orders are received. We add “Assemble” to our lexicon to describe these tasks. Only two SKUs are required in this design (cooked patties for the beef burger and low-fat burger). All other components are added after orders are received.



In this configuration, the average customer wait time increases because the number of orders that contain a low-fat burger with cheese or a hamburger with cheese is higher. Note that wait time increases because the probability of receiving these orders has increased. It is important to compute the expected customer wait time through the entire supply chain to understand the costs as well as the benefits of holding fewer SKUs and having fewer inventory write-offs.

In an attempt to eliminate inventory, we could restructure the supply chain by moving “Order” before “Cook” (making our burger joint resemble the fine-dining establishments). As might be expected, this approach entails a much longer average wait time because every customer must wait through the entire manufacturing and order-fulfillment process before being served. Unfortunately, as financial considerations drive us to cut inventories to the barest minimum, we realize that the increasing delays begin to jeopardize our value proposition. Along with the extended wait time, inventory requirements affect performance. Resequencing the “moves” in the supply chain—adding cheese to a low-fat burger

Not included here are estimates of wait time and inventory requirements for the tasks to be accomplished. Because these activities are complex, we must work further on the lexicon. In other words, more refined definitions must be found for the processes included in the supply chain.

Key Process Characteristics

Economic behavior constitutes the “meaning” of the design, so we must define each process carefully and assess its potential impact on value. Actual behavior must be thoroughly understood before we can position the push-pull boundary. Certain characteristics influence the value contributed by each process step. Specifically, we need to consider the impact of:

- **Wait time**—How much does the time required for this process (including quality and rework time) increase wait time for the customer? Intervals of waiting may be long or short; the longer they are, the greater the likelihood that customers will lose interest or turn to a competitor. In general, long process steps in manufacturing or assembly are better positioned *before* and short process steps *after* the push-pull boundary.

- **Options**—The number of options introduced by the process also is critical. Localization, for example, may involve dozens of configurations. And the product will branch into that number of directions when the process is done. If this happens *before* orders are received, forecast error is much more difficult to manage. By contrast, if Add Paper Tray occurs *before* ordering, only two options must be managed in response to demand. Obviously, adding a paper tray is not nearly as value destructive as adding localization *before* the push-pull boundary.

- **Component value**—What is the value of the components consumed? How much “price protection” will be needed to safeguard our partners against devaluation? If parts are costly (like paper trays) or highly perishable (like memory chips), the process should come *after* receiving orders in order to minimize obsolescence costs.

- **Process intensity**—Processes can be heavy manufacturing, light manufacturing, or touch-only. Process and test engineering may be required for heavy or light manufacturing, while touch-only may require neither. In general, high-intensity processes should be positioned *before* the push-pull boundary and low-intensity processes *after*. Before this point, the inventory buffer isolates upstream processes from demand uncertainty, so tasks can be made more efficient, manufac-

turing can be linearized, and total supply chain costs can be minimized. It also may be possible to centralize global manufacturing before the push-pull boundary. Transporting goods across oceans could possibly become less expensive than replicating capacity in different parts of the world.

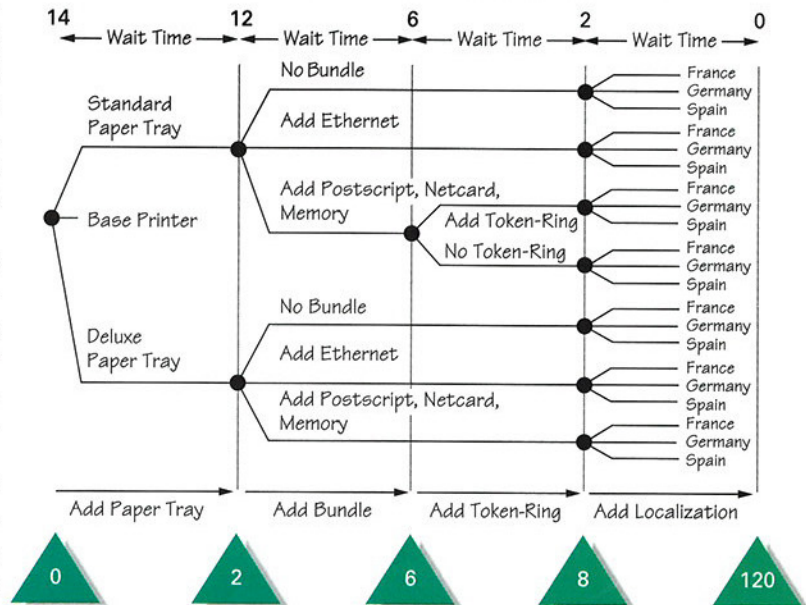
Because of their economic impact, these and other process characteristics must be reflected in the alternatives specified for analysis.

Evaluating Two Promising Alternatives

Although our template suggests many alternatives, the two most promising are diagrammed below—Add Paper Tray First and Add Bundle First. In our language, triangles show the number of stock-keeping units that would be required at each potential location of the push-pull boundary. Supply chain inventories increase with the number of SKUs required by the selected alternatives.

Add Paper Tray First

The first alternative conforms to our template. Here we see the SKUs needed at each potential location of the push-pull boundary (the numbers in the triangles). We also have estimates of the wait time customers must endure as a result of each process step. (To simplify, we ignore the time required for ordering, serving, and other activities.)



Scanning this diagram, we recognize trade-offs between wait time and inventory.

- With build-to-order (far left position), only generic parts are stocked so we minimize invento-

■ LANGUAGE

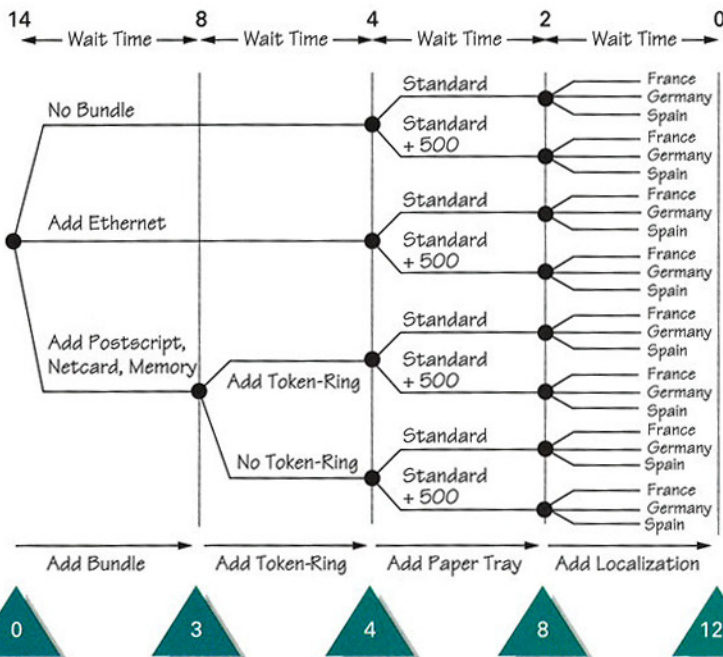
ry (zero) and estimate substantial wait time (14). Again, these are our best estimates of the time and material required at this process step.

■ With build-to-stock (far right position), fully localized products are built to forecast. Although there is no wait time, we need 120 different SKUs in stock and foresee substantial inventory requirements.

More interesting, however, are potential locations in the second, third, and fourth positions. When we model the economic behavior of this alternative, we need to understand the impact of holding two inventory items with 12 units of wait time (second position), six inventory items with six units of wait time (third position), or eight inventory items with two units of wait time (fourth position). Along with the other kinds of information discussed below, these estimates must be available to the evaluation process.

Add Bundle First

Process resequencing gives us our second alternative. In an attempt to improve the supply chain economics, we move Add Bundle and Add Token-Ring ahead of Add Paper Tray. Build-to-order (far left) and build-to-stock (far right) are the same as in the first alternative, but the intermediate positions are different. The analysis needs to consider three inventory items with eight units of wait time (second position), four inventory items with four units of wait time (third position), and eight inventory items with two units of wait time (fourth position).



Preliminary Evaluation

The objective of the evaluation is to find the supply chain design alternative that minimizes the *total cost*. The following estimates help us perform a preliminary evaluation:

■ Estimated inventory-driven costs. For every SKU maintained before the push-pull boundary, we anticipate significant inventory-driven costs. We estimate these costs by multiplying the number of units by \$300,000.

■ Estimated lost profits. For every unit of wait time, we anticipate losing customer orders. This is a key driver. We might estimate, for example, that each unit of delay will cost \$500,000 in lost profits during the product life cycle.

Quick calculations provide insights that are consistent with our understanding of the processes involved. In both cases, the aggregate cost of inventory and wait time at Start (first position) or Add Localization (fifth position) make these unacceptable for the push-pull boundary. As the following tables show, neither build-to-order (\$7,000,000) nor build-to-forecast (\$36,000,000) would be cost-effective.

Add Paper Tray First

Push-Pull Boundary	SKUs	Cost	Wait Time	Cost	Aggregate
Start	0	\$000	14	\$7,000,000	\$7,000,000
Add Paper Tray	2	\$600,000	12	\$6,000,000	\$6,600,000
Add Bundle	6	\$1,800,000	6	\$3,000,000	\$4,800,000
Add Token-Ring	8	\$2,400,000	2	\$1,000,000	\$3,400,000
Add Localization	120	\$36,000,000	0	\$000	\$36,000,000

Furthermore, the lexical unit (or "move") with the greatest change in significance is Add Paper Tray. This process contributes \$6,600,000 to costs in the first alternative, but its impact drops to \$3,400,000 in the second. The following table illustrates the effect of this process resequencing.

Add Bundle First

Push-Pull Boundary	SKUs	Cost	Wait Time	Cost	Aggregate
Start	0	\$000	14	\$7,000,000	\$7,000,000
Add Bundle	3	\$900,000	8	\$4,000,000	\$4,900,000
Add Token-Ring	4	\$1,200,000	4	\$2,000,000	\$3,200,000
Add Paper Tray	8	\$2,400,000	2	\$1,000,000	\$3,400,000
Add Localization	120	\$36,000,000	0	\$000	\$36,000,000

Although the trade-off between wait time and inventory is clear, a decision based on these results would be premature. Other trade-offs may change the cost structures shown above. To find the best alternative, the analyst must describe economic behavior more completely—taking into account such things as process intensity, component devaluation,

price protection, and other factors to determine the full cost implications of different approaches.

Extended Evaluation

Extending our evaluation, we include the impact of linearization (process intensity) and price protection (component value) relative to the push-pull boundary. Here are descriptions of the trade-offs involved and estimates of their combined impact.

■ **Linearization**—Before the push-pull boundary, processes of different intensities—low, medium, or high—may be linearized for greater efficiency. Extra costs are incurred *after* the push-pull boundary because of capacity restraints and limited optimization possibilities. For this example, we estimate the following intensities: Add Paper Tray (low), Add Bundle (medium), Add Token-Ring (medium), Add Localization (low). To reflect added costs after the push-pull boundary, we estimate \$100,000 over the product life cycle for low-intensity processes, \$1,000,000 for medium-intensity processes, and \$10,000,000 for high-intensity processes.

■ **Price protection**—If market conditions lead to falling prices for consumers, products waiting to be sold are protected and differences may be refunded to our channel partners. The cost of price protection depends on the value of components consumed by a process relative to the push-pull boundary. If an expensive component is added early, there is a greater chance that its value will be reduced before the product is sold. If this happens, price protection may kick in. If an expensive chip is used in the product, for example, ideally we want orders in hand before adding that chip (because its value decreases). To reflect price protection costs *before* the push-pull boundary, we estimate the following risk levels—Add Paper Tray (high), Add Bundle (medium), Add Token-Ring (medium), Add Localization (low). Our financial estimates are \$100,000 (low), \$500,000 (base case), and \$1,000,000 (high) for the value of components at risk before the push-pull boundary.

The following tables present the aggregate costs associated with linearization and price protection for the alternatives under consideration.

Add Paper Tray First

Push-Pull Boundary	Linearization	Cost	Price Protection	Cost	Aggregate
Start	medium	\$2,200,000	none	\$000	\$2,200,000
Add Paper Tray	low	\$2,100,000	high	\$1,000,000	\$3,100,000
Add Bundle	medium	\$1,100,000	medium	\$1,500,000	\$2,600,000
Add Token-Ring	medium	\$100,000	medium	\$2,000,000	\$2,100,000
Add Localization	low	\$000	low	\$2,100,000	\$2,100,000

Add Bundle First

Push-Pull Boundary	Linearization	Cost	Price Protection	Cost	Aggregate
Start	medium	\$2,200,000	none	\$000	\$2,200,000
Add Bundle	medium	\$1,200,000	medium	\$500,000	\$1,700,000
Add Token-Ring	medium	\$200,000	medium	\$1,000,000	\$1,200,000
Add Paper Tray	low	\$100,000	high	\$2,000,000	\$2,100,000
Add Localization	low	\$000	low	\$2,100,000	\$2,100,000

Complete Evaluation

At this point, we aggregate the “preliminary cost” (the trade-off between inventory and wait time) and the “extended cost” (other trade-offs involving linearization and price protection) to complete our evaluation. The following results allow us to compare aggregate costs at each potential position of the push-pull boundary.

Add Paper Tray First	Preliminary Cost	Extended Cost	Aggregate
Start	\$7,000,000	\$2,200,000	\$9,200,000
Add Paper Tray	\$6,600,000	\$3,100,000	\$9,700,000
Add Bundle	\$4,800,000	\$2,600,000	\$7,400,000
Add Token-Ring	\$3,400,000	\$2,100,000	\$5,500,000
Add Localization	\$36,000,000	\$2,100,000	\$38,100,000

Add Bundle First	Preliminary Cost	Extended Cost	Aggregate
Start	\$7,000,000	\$2,200,000	\$9,200,000
Add Bundle	\$4,900,000	\$1,700,000	\$6,600,000
Add Token-Ring	\$3,200,000	\$1,200,000	\$4,400,000
Add Paper Tray	\$3,400,000	\$2,100,000	\$5,500,000
Add Localization	\$36,000,000	\$2,100,000	\$38,100,000

The evaluation clearly indicates that the most cost-effective location for the push-pull boundary is the third position of Add Bundle First (the design created with process resequencing). In our best sequence, Add Paper Tray and Add Localization are postponed until customer orders have been received. The grammatical sequence that makes “the best sense for the product” arises from an analysis of economic behavior made evident by the language of supply chains.

Extending the Language

Many dimensions of economic behavior can be captured by extending the language of supply chains. The approach enables designers to create a unique lexicon for the individual product, quantify the implications of proposed “moves,” perform an economic analysis, and make decisions that reflect customer interests, product designs, manufacturing options,

Many dimensions of economic behavior can be captured by extending the language of supply chains.

distribution networks, and service and support.

In suggesting “extensions” to the language, we want to emphasize the need for creativity on the designers’ part. No less important than product design is the design of the supply chain through which that product will be manufactured and delivered. Our diagrams have included graphics—icons, colors, arrows, and other features—to render the subtleties of design alternatives. With each new dimension comes an opportunity to extend the technique by which economic performance is expressed. For the dimensions discussed below—customer, product, manufacturing, distribution, service, and support—we are developing the means to capture and communicate the nuances of effective supply chains. Here we see how excellence in supply chain design can be expressed through excellence in supply chain rhetoric.²

Customer Dimension

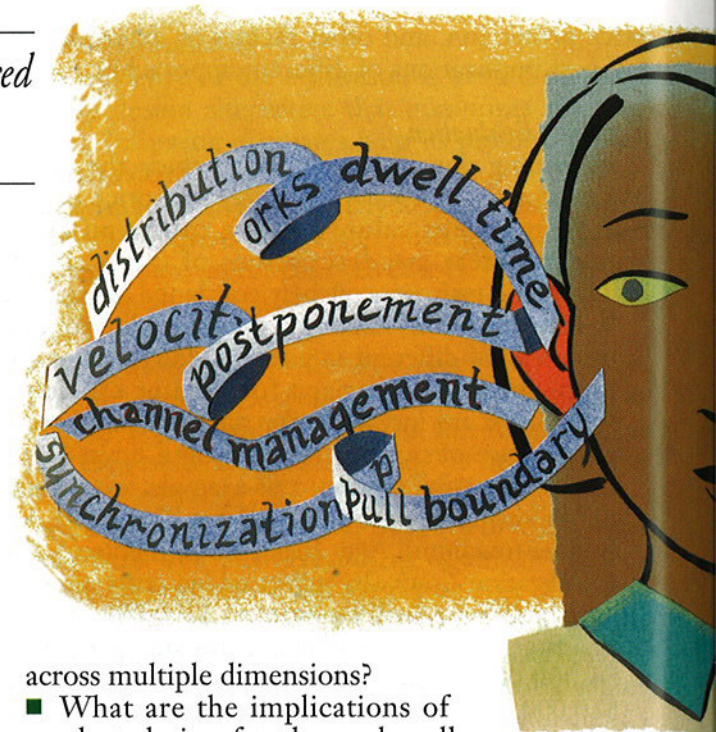
We have already characterized customers by their willingness to wait for what they want. The language captures the wait time for any activity, aggregates wait time across the supply chain, and identifies trade-offs (especially with inventory required to respond to customer urgencies). The language can be extended to address such questions as:

- How do customer interests affect the supply chain’s performance?
- Should the supply chain be optimized to respond to customer concerns?
- What are the trade-offs between customer preferences and supply chain costs?
- Can the push-pull boundary be repositioned to the customers’ advantage?

Product Dimension

Product designs can be characterized by their suitability for different approaches to manufacturing, assembly, customization, and delivery. We have observed the economic advantages of postponement. Other aspects of product design can be reflected in the language, including a surprising range of functionality, modularity, aesthetics, connectivity, and other features.

- How do different product designs affect supply chain performance?
- Can design alternatives result in trade-offs



across multiple dimensions?

- What are the implications of product design for the push-pull boundary?

Manufacturing Dimension

Manufacturing processes can be characterized by their intensity as well as their suitability for linearization, automation, or other engineering approaches. The language allows us to register the impact of such processes and make the right trade-offs. Many considerations can be expressed, including value management of components and parts (with implications for price protection) and appropriate levels of testing, usability, and quality control.

- How do unique product designs influence supply chain performance?
- Can design alternatives result in trade-offs across multiple dimensions?
- What are the main drivers of manufacturing costs? How do they interact?
- How do manufacturing options affect the location of the push-pull boundary?

Distribution Dimension

The language allows us to characterize a broad array of distribution networks, including order-fulfillment processes at global, regional, or local levels using a variety of transportation options. Designers can make trade-offs across multiple dimensions, showing how customer preferences and product design alternatives, for example, interact with distribution options that include mass customization, channel assembly, and localization across the network.

■ LANGUAGE

- How do network capabilities interact with product design to affect performance?
- What trade-offs appear with different distribution network designs?
- How does the push-pull boundary change with different distribution options?

Service and Support Dimension

Less obvious but highly important to the supply chain design are the various approaches to service and support; these too can be reflected in the language. We can characterize alternatives in other dimensions by their impact on service and support, including warranties, trade-ins, and returns. Trade-offs with service and support costs should not be overlooked when working through the design.

- How do customer preferences or product designs affect service and support?
- What trade-offs are possible as a result of service and support alternatives?
- Does the push-pull boundary have service and support implications?

Other dimensions—including advertising, marketing, and sales—can be rendered in the language we have been developing. In each case, the questions to be answered imply opportunities for trade-offs to support the most advantageous supply chain design for the product.

Creatively Exploring the Alternatives

We began by proposing that a language of supply chains would supplement other analytical approaches for designing and evaluating supply chains. We close by drawing attention to the obvious fact that supply chain design is an art (as well as a technical discipline) and deserves consideration for its extraordinary power to create and conserve value. The approach outlined here allows managers to explore how their supply chain contributes to the business,

interacts with customers, and delivers the goods efficiently and effectively. Put into practice, the insights and understandings from these explorations can make the supply chain—and the company—far more competitive than before.

In particular, we think the position of the push-

H*aving a grammar enables us to revise the supply chain until the most economically advantageous position for the push-pull boundary has been found.*

pull boundary—the point at which customer orders enter the system and change the forces at work—is one of the most important features the supply chain manager can control. The linguistic approach helps designers improve supply chain effectiveness through an imaginative exploration of alternative designs. By capturing and rendering influences across multiple dimensions and by revising the grammatical sequence through which these influences are expressed, designers can find the supply chain that makes the best economic sense for the product. Best of all, this innovative approach can be generalized across businesses and industries—and the language of supply chains becomes a methodology widely shared.

Footnotes

¹ A common term across Hewlett-Packard. Also called the “customer order penetration point” by theorists.

² We mean “rhetoric” in the positive sense communicated by Robert Eccles in *Beyond the Hype* (1992), a seminal treatise on the ways rhetorical devices can be used to advance business enterprise.