# Dynamic Fulfillment:
# The Three Keys to Unlock Ship-From-Store Success

Discover how cost-based fulfillment harnesses real-time optimization and helps retailers like Zumiez and Torrid best leverage available inventory for maximum efficiency, profitability, and customer satisfaction.

toolsgroup® | Navigate Uncertainty™

If you're a retailer, you understand that omnichannel initiatives like adaptable, efficient ship-from-store programs are vital to compete in a marketplace driven by customer whims and buffeted by supply chain volatility.

Many retailers are also (quickly) realizing that the tools and strategies that once governed their fulfillment processes are severely ill-equipped to support ship-from-store programs. With the addition of stores as fulfillment centers, omnichannel fulfillment just got exponentially more complex.

How can retailers know what item to ship from where without increasing shipping costs, damaging margins, or endangering customer satisfaction?

**The answer: ToolsGroup Dynamic Fulfillment,** an application built to handle ship-from-store complexity. By analyzing every fulfillment option in real-time and calculating total cost-to-serve for each option, Dynamic Fulfillment determines the fulfillment decision that will best achieve business goals and satisfy the customer.

How does it do this in such a complex omnichannel environment?

Read on for the debrief on ship-from-store challenges and how Dynamic Fulfillment not only *can* overcome them but already *has* for some of the world's leading retailers.

## The Challenges of Ship-from-Store

It's important to know what you're up against when you begin fulfilling from stores. The more you understand the challenges, the better you'll be able to nullify them.

### **Challenge #1 -** Broader Fulfillment Networks

When you begin shipping from stores, you're increasing the size of your fulfillment networks. Perhaps you only had a handful of centralized fulfillment centers before, but now, you've multiplied that exponentially, depending on the number of stores. There's a lot of value in that – but also a lot of complexity.

For instance, say you only fulfill from two warehouses. You apply rules-based logic to figure out which warehouse is the best option for fulfilling a specific order at a specific time, and – since you only have two options – it's not a very complicated decision.

Then you suddenly activate 500 stores as fulfillment centers. The logic that once easily applied to just two sites now has to manage a far greater number of options but is not suited to such a complex and expansive network.

### **Challenge #2 -** Solving for Multiple Objectives

Network expansion isn't the only challenge resulting from ship-from-store initiatives. There's an underlying issue that catches a lot of organizations off guard – namely, having a fulfillment logic that can solve for multiple objectives.

For example, traditionally, an objective might be something like "reducing shipping costs." That's a good idea and relatively easy to do.

But that's not the only objective you need to account for in an omnichannel environment. You need a solution that finds the optimal balance between shipping and labor costs, avoiding markdowns in a store, and customer expectations. Trying to account for all of these objectives at once is too much for the built-in logic capabilities in most order management systems (OMS).

## Challenge #3 - Store Fulfillment Variability

In a warehouse, you have a near-perfect understanding of inventory. But a store is more of a wild card, and its inventory is inherently more variable. There may be inventory inaccuracies resulting from erroneous item scans, and products have varying sell-through rates depending on location or season.

This is why you need to be able to predict things like the location-specific demand for an item or store-specific fulfillment rate.


## The Downsides of Traditional, Rules-Based Fulfillment

With these challenges in mind, how do you approach the situation and try to optimize your fulfillment process? What happens when you ship from store using traditional methods?

Most companies leverage their existing OMS and try to apply fulfillment logic through a rules-based approach. This heuristic approach has been used in the past because it is very good for solving for a single objective.

For example, say your goal is to reduce shipping costs. So you build a set of rules. First, you'll narrow down your fulfillment options to stores that combine the order items and ship them all together. Of those stores, you'll look at the stores closest to the customer to get it there faster. Finally, you'll choose the location with the highest inventory on hand to make sure you're not depleting a store's inventory.

In theory, this sounds like a solid plan. But what's actually happening?

Inevitably, you run into edge cases and the OMS sourcing logic falls short – with major detriments to your business.

### Traditional OMS sourcing logic.

**Uses sequential rules to make decisions.**
Each rule acts as a filter, narrowing options until the final location(s) is selected.

**Common decision sequence:**

1. *Minimize number of shipments,* ***then***
2. *Ship closest to the customer,* ***then***
3. *Pick location with highest inventory OH*

### Where it falls short.

**Only locations that meet the criteria at a single point in the sequence can be selected.**
"Filtered" options can't be considered against future, competing factors.

**Suboptimal outcomes:**

- ⊗ *Hold heavy items together, ship cross-country*
- ⊗ *Fulfill from store with poor pick-rate, higher labor cost*
- ⊗ *Source from full-price inventory vs. stale markdowns*

In narrowing the pool down to stores that can ship the items together, you're actually shipping them from across the country and you've just made the package heavier. So you've increased the weight and transit of the package – which increases your costs.

Then perhaps you choose from the stores closest to the customer, but you're actually choosing a store with higher labor cost, high utilization, or fewer associates available to pick, pack, and ship an order.

Finally, you pick a location with the highest amount of available inventory without considering how well that inventory will sell in that location. It might have been better for you to ship from a store with less inventory but a greater chance of markdowns.

The logic you've built cannot solve for ship cost, labor cost, and markdown avoidance simultaneously. It can only solve for one of those objectives at a time. And the more you try to account for all the variables, the more complex the OMS logic will get – and the more unwieldy and inefficient.

So if the rules-based approach doesn't work, what will?

## The Limitations of Traditional Methods

| | |
|---|---|
| ***Approach*** | ***Rules-based.*** Uses simplistic, "yes / no" heuristics to select a fulfillment location. |
| ***Factors*** | ***Singular.*** Solves for a single objective at a time. |
| ***Scale & Speed*** | ***Limited.*** Built for networks that primarily use a handful of centralized fulfillment centers. |

## The Cost-Based Approach: How Dynamic Fulfillment Achieves True Fulfillment Optimization

Dynamic Fulfillment uses a cost-based approach, meaning that it doesn't just apply a set of rules, one after the other, until you narrow down your choices.

Instead, it looks at all of the possible fulfillment options and calculates the total cost-to-serve of each option. It does this while accounting for multiple objectives and factors like shipping cost, operational behavior at each location, store demand, etc. As a cloud-based application, it's also capable of scaling rapidly, evaluating hundreds of thousands – even millions – of permutations in subseconds.

| | The Status Quo | vs. | Dynamic Fulfillment |
|---|---|---|---|
| ***Approach*** | ***Rules-based.*** Uses simplistic, "yes / no" heuristics to select a fulfillment location. | | ***Comprehensive.*** Predicts the fulfillment outcome that minimizes total cost to serve while meeting both business and customer objectives. |
| ***Factors*** | ***Singular.*** Solves for a single objective at a time. | | ***Extensive.*** Estimates cost to serve and operational behavior to predict the actual, total cost to fulfill an order. |
| ***Scale & Speed*** | ***Limited.*** Built for networks that primarily use a handful of centralized fulfillment centers. | | ***Exponential.*** Evaluates every permutation across thousands of possibilities, providing an optimal recommendation at web-speed. |

## The Three Components of Successful Cost-Based Optimization

Dynamic Fulfillment evaluates and integrates three core components into its cost calculations and optimization decisions.

### 1 - Ship Cost

The first component is ship cost. As Dynamic Fulfillment analyzes each possible fulfillment option, it calculates the cost of shipping from a given location to a customer. It considers the item itself, where inventory is located, and the different carriers you work with.

Dynamic Fulfillment also looks at the fulfillment capacity of a location. The fulfillment capacity represents how many units a location can pick, pack, and ship in a day. This is compared to the number of units allocated to a location to be fulfilled, representing the backlog of items to be picked. By analyzing these inputs, Dynamic Fulfillment determines the utilization of a fulfillment center, how long it will take to get an item out the door, and how many days are left for that item to arrive at the customer's doorstep before missing the SLA target.

**Dynamic Fulfillment Inputs**


Allocated quantity/ capacity


Item weight and dimension


Transit Time


Rate Cards

Then, Dynamic Fulfillment does your rate shopping for you, identifying the estimated ship costs for each possible option from each possible carrier for each order. It finds the lowest option that gets a package to the customer in the given amount of time.

Even as variables are changing, the speed of Dynamic Fulfillment's computations can keep up and find the best possible option for each shipping decision.

### 2- Labor Cost

Another component Dynamic Fulfillment considers is labor cost. It estimates the cost to pick, pack, and ship a unit from a specific location. It examines productivity, regional financial differences, and labor constraints to determine which location can optimally fulfill an order.

So perhaps one location is very productive and can pick 60 units an hour. The cost-per-pick will be lower than a location that can only pick 10 units an hour. Factoring in labor cost allows you to fulfill from a store where productivity is higher.

### 3- Expected Revenue Estimation

Perhaps the most valuable component of cost-based fulfillment is expected revenue.

What do we mean by *expected revenue* in this context?

Think of it in terms of opportunity cost. When you ship an item from a location where it would sell full price instead of shipping from a location with a markdown, that's an opportunity cost for the company. Ideally, you would sell the item from a location with a greater markdown risk.

So what you need to know is the expected revenue of that item – the estimated sell price of the last unit of a specific product in a given location.

How do you come up with this number? It's a two-step process.

**First:** Using your location-specific forecast, estimate the stock-out date from an item in a location.

**Second:** Using your markdown cadence, determine the selling price of the item on the estimated stockout date.

Consider the example below.

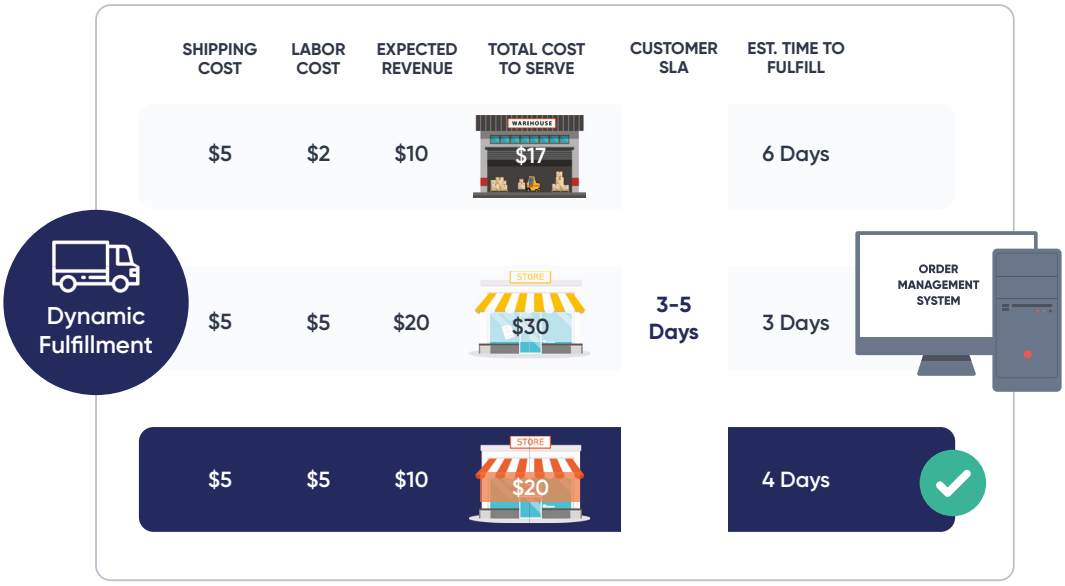

In Store A, the last unit of a product is estimated to sell *before* the markdown price, meaning that, if sold from this location, it would sell for regular price. In Store B, the last unit is estimated to sell *after* the markdown, so it would sell for less.

In this case, choosing to ship from Store B means you're pulling from a location where that item would otherwise have sold at markdown price. So you fulfill the order while avoiding a possible markdown at Store B and protecting a full-price purchase at Store A.

## How Dynamic Fulfillment Achieves Multi-Objective Optimization

Dynamic Fulfillment analyzes all of these considerations – shipping, labor, and markdowns – simultaneously to make the optimal fulfillment decision for each order.

Think of Dynamic Fulfillment as the brain that sits on top of your OMS. It's system-agnostic, integrating seamlessly with any order management system and automating fulfillment behind the scenes. For every order, it calculates every permutation of the three cost factors for every possible fulfillment location. All in subseconds. Consider the example below, depicting a simplified version of the decisions Dynamic Fulfillment is making continuously. There are three different fulfillment options, each with its own ship cost, labor cost, and expected revenue.




Based on this information, Dynamic Fulfillment estimates the total cost-to-serve and balances that with customer SLAs. For instance, if the customer expects an item in 3-5 days, fulfilling in 6 days is unacceptable. No amount of cost savings can undo the damage done to brand loyalty if fulfillment is consistently lengthy and unreliable.

However, in the example above, having that range of days allows Dynamic Fulfillment to take advantage of an extra day and fulfill with a lower cost-to-serve while still protecting the customer experience.

## Node Balancing: The Tie-Breaker

When evaluating fulfillment options, Dynamic Fulfillment may find multiple plans that all have the same cost-to-serve. So what acts as the tie-breaker?

In these cases, Dynamic Fulfillment considers backlog and utilization in each location.

Let's look at the example below in which two stores have the same total cost-to-serve.


For reference, if "Backlog Days" equals 1, it means that location is fully utilized for the day. A backlog rate that is less than 1 means that location has capacity for the day, and a backlog greater than 2 means the location is overutilized for the day.

This is important because it indicates how long it takes to get an order out the door to the customer.

In the example above, Store A has .5 backlog days, while Store B has 2 backlog days.

This means, from an operational perspective, that the item in question can be shipped the same day from Store A, but it can't be shipped for at least two days from Store B.

By fulfilling from Store A instead of Store B, the retailer can get goods to the customer faster, elevating customer satisfaction, while reducing strain and labor in overutilized nodes within the network.

## The Benefits of Dynamic Fulfillment

An effective ship-from-store program, powered by the right methodology and the right technology, proves why so many companies want to initiate ship-from-store in the first place.

- **Maximum Profitability**
- **Reduced cost-to-serve**
- **Increased margins**
- **Greater customer satisfaction**
- **Seamless, tech stack-friendly integrations**
- **Faster, more accurate decision making**

The good news is – these benefits aren't hypothetical. Real companies are achieving real results with ToolsGroup Dynamic Fulfillment every day.

## Real-World Examples: Retailers Using Dynamic Fulfillment

Here is how some of the world's leading retailers have achieved ship-from-store success with Dynamic Fulfillment.

### ZUMIEZ

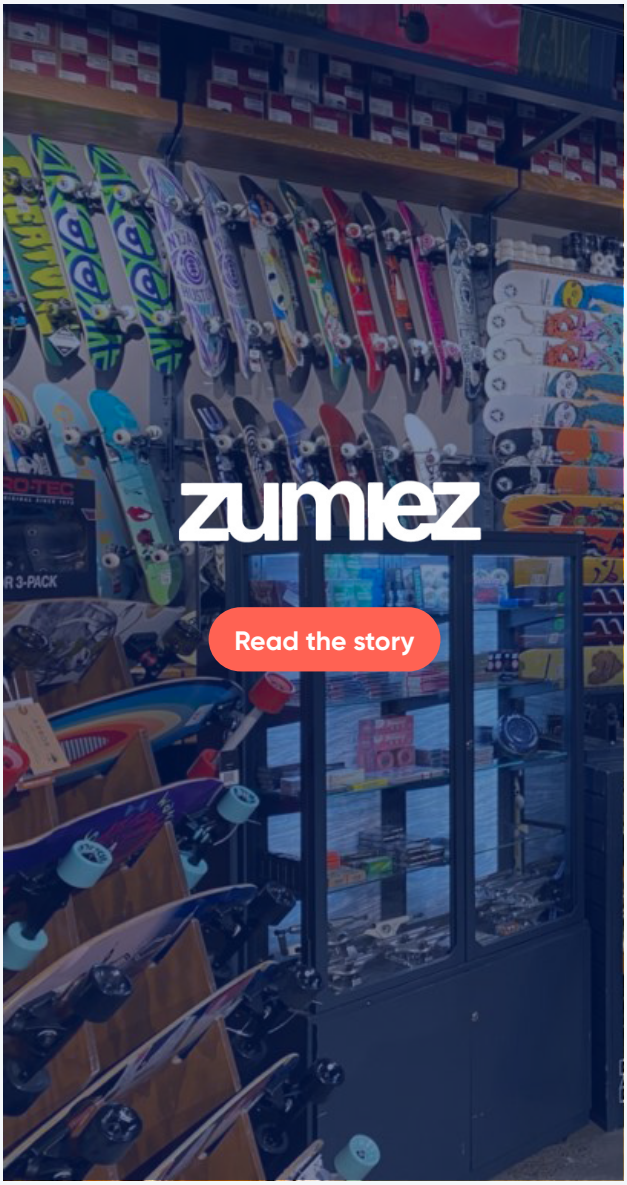
Zumiez is a specialty retailer of apparel, footwear, and accessories for young men and women, with over 700 stores. Unlike traditional fulfillment approaches, Zumiez fulfills almost entirely from stores, pioneering a truly "channel-less" experience.

Dynamic Fulfillment helped them achieve:

- $1.2 million in fulfillment savings in the first year of launch
- 3-10% decrease in operational split rate (est. $1 million in annual savings)

**"The Dynamic Fulfillment solution from ToolsGroup helps us use data to make smarter, faster decisions to create an even better experience for our customers. The fulfillment optimization logic helps us generate operational efficiencies like reducing split rates and shipping costs while better serving the needs of our customers."**

– Paul Kisicki, VP of IT at Zumiez



## Belk

Belk is a US department store chain with about 290 stores, selling men's, women's, and children's apparel, as well as shoes, fine jewelry, accessories, home goods, electronics and more. As its omnichannel operations grew, the company embraced a ship-from-store program. Thanks to Dynamic Fulfillment,

Belk was able to:

- Reduce markdowns, resulting in $29.8M GM savings
- Save $7.5M in shipping costs due to maintaining lower split-rate (19pp lower than pre-Dynamic Fulfillment)
- Achieve higher ship rates


**Watch Belk and ToolsGroup's** joint webinar on the benefits of **Dynamic Fulfillment.**

### How Dynamic Fulfillment Helps Retailers Reduce Shipping Costs and Improve Margins

Watch Now

## Torrid

Torrid is a leader in fast fashion, specializing in plus-size, direct-to-consumer women's apparel and accessories. During the Covid-19 pandemic, Torrid needed to improve their ship-from-store capability to maximize margins and profitability. Dynamic Fulfillment allowed them to achieve multi-objective optimization and enhanced business outcomes:

- Markdown avoidance of $0.91 per unit shipped
- 10 percentage point reduction in split shipments
- $0.21 per unit ship cost
- Increased product availability to customers across all stores

**"In order to enable ship from store capabilities we needed intelligence beyond what was available in our OMS. Dynamic Fulfillment gave us the ability to optimize our fulfillment decisions based on the total cost to serve a customer and the expected revenue we would achieve. The benefit was immediate, and the markdown avoidance calculations are driving significant improvements in our margins."**

– Chris Church, VP of Innovation, Torrid


Learn more about how **Dynamic Fulfillment** helps companies optimize fulfillment for **greater efficiency, enhanced customer satisfaction,** and **better margins.**

Learn more