# ORIE 4820: Spreadsheet-Based Modeling and Data Analysis
# Swifty Service Parts
# Spring 2013

---

Swifty Service Parts, Inc. is a regional provider of maintenance support for office equipment and specialized printing systems. Swifty currently has a contractual agreement with an office supply merchandising chain whereby it provides maintenance service (i.e., parts and labor) to customers who have leased or purchased certain products at one of the merchandiser's stores. The agreement covers customers in *ten different service areas*.

In each of the ten service areas, Swifty maintains its own "store", a place for stocking service parts that are used to satisfy customer demands that arise in the area. (These "stores" are not places where customers shop – they are simply inventory storage locations for use by Swifty personnel who make service calls.)

The contractual agreement between Swifty and the merchandiser stipulates that:

(1) The merchandiser will pay Swifty a fixed fee for each service call made to a customer *that results in problem resolution*. The merchandiser will only reimburse Swifty for the cost of parts *used in servicing customer equipment*.

(2) Swifty is responsible for purchasing, holding, and dispatching all service parts needed to support the merchandiser's customers. Any excess, damaged, or obsolete service parts are *solely Swifty's responsibility*.

(3) Swifty must undertake staffing and inventory policies to ensure that *in each of the ten service areas and for each product class, 95% of the merchandiser's customer demands are resolved in the same day*.

Same-day resolution of a customer problem is possible *only* if the service part needed is on hand at the store when the customer demand arises. Therefore, to comply with part (3) of the agreement, ***each of Swifty's ten stores must hold enough inventory so that for each product class, 95% of the customer demands for service parts can be fulfilled immediately***.

***The Plotter***, a high-end CAD/CAM printing device, is included in the contract. The Plotter is the only product in its class. Swifty is concerned about the Plotter because the six service parts used in its maintenance (A, B, C, D, E, and F) are unusually expensive to purchase and hold. Moreover, these parts are unique to the Plotter and cannot be used to service any other product.

The Plotter service parts A, B, C, D, E, and F are *replenished on a one-for-one basis* at Swifty's stores. That is, each time a customer demand arises for one of these parts at a store, the part is dispatched (provided the store has one on hand), and a replenishment order for another unit is placed immediately with the part manufacturer. This type of policy, called a *base stock policy*, is often used when dealing with lower-volume and/or expensive items. The lead time needed to ship a unit from the part manufacturer to a store varies by part type and by store. All parts are dispatched from stores to customers on a first-come-first-served basis. Demands that cannot be fulfilled immediately are backordered.

One of the features of a base stock policy is that the inventory position of a part at a store (on hand + on order – backorders) is *constant*, and is equal to the part's *base stock level*. Because the Plotter service parts are expensive, Swifty would like to be prudent about setting base stock levels for these parts at each store. Specifically, Swifty wants to determine:

***What should the base stock levels $s_{ij}$ be for all service parts $i \in \{A, B, C, D, E, F\}$ at each store $j \in \{1, 2, \ldots, 10\}$ in order to meet the contractual service level requirements for the Plotter (95%) with minimal inventory investment?***

That is, we want to:

$$\textbf{minimize} \left( \sum_{\text{items } i} \sum_{\text{Stores } j} c_i \cdot s_{ij} \right)$$

System Inventory Investment

$$\textbf{subject to:} \sum_{\text{items } i} w_{ij} \cdot f_{ij}(s_{ij}) \geq F_j \quad \forall \textbf{ stores } j$$

Service level guarantee at store *j*

$$s_{ij} \geq 0 \text{ and integer } \forall i,j$$

Relative likelihood that service part *i* will be demanded at store *j*

Immediate fill rate of service part *i* at store *j* (depends upon the base stock level of part *i* at location *j*)

Since this problem is *separable by store*, we can *focus our attention on a single store j*. Note that the problem to be solved for store *j* involves an objective function that is *linear* in the $s_{ij}$'s and a *single* constraint. This constraint, however, has a *nonlinear* function of the $s_{ij}$'s on its left-hand side (LHS). The right-hand side is simply a constant corresponding to the service level guarantee (95%).

This week, you will implement an algorithm to solve this optimization problem for Swifty.