Past-event records don't make our demonstration easier; they avoid the challenge altogether.

applies. But why do you look troubled? Ah, you want to know how on earth we came up with two more boxes despite having no idea what we're going to do with them. Well, I'm reluctant to admit it, but I haven't the foggiest notion either. I threw them in because that's what the future-event pattern told us to do and, as I mentioned, it is an old and trusted friend. Now that we have them, what say we press on and figure out their use?

Consider the resupply report. "Based on its stock quantity being too low," the goal said. Evidently, we need a field, qty-that's-too-low, in part-item. The daily resupply report will list all items whose on-hand-stock-qty is less than its qty-that's-too-low.

But a moment's thought reveals that once listed, an item will continue to be listed every day until resupply arrives and on-hand-stock-qty gets back up to where it should be. Surely, that can't be what was wanted. After it's printed, the report will undoubtedly go to someone in purchasing who buys parts to resupply the stockroom. Assume she immediately orders everything on the first day's report. How will she feel about being harassed every day about those same items until they arrive? When that happens, we know it will be futile to plead that "We met the goals," for the time-hallowed reply is, "What we really wanted was . . . "—and that path leads to madness.

Should we then suppress an item's appearance on the resupply report if it is already on order? Well, not exactly. If on-hand-stock-qty is 13, and qty-that's-too-low is 350, and she ordered 12, she deserves harassing. She should have ordered at least 337. The solution is to report only those items where on-hand-stock-qty plus the sum of all the already-ordered-qtys is less than qty-that's-too-low.

This leads us to seek a place to put the quantity and expected arrival date of every outstanding purchase order. In other words, we need a record for each expected part-in event. The future-event pattern happens to have provided just that.

As a second example, consider our supply of framis bearings. This vital component's on-hand-stock-qty is 20. Its qty-that's-too-low is 15. And its sum of already-ordered-qty is 10. Since 20 plus 10 is not less than 15, we don't want the thing to show up on report. If it does appear we'll soon have more framis bearings (whatever they are) than we need.

But investigation reveals that Maintenance plans to tear down the paraxylene dehydration tower's regeneration loop and replace 50 framis bearings as routine preventive maintenance. Consequently, they'll need 50 framis bearings next month, and if we fail to include the item in the report, they might not be ordered in time.

CASE IS NOT AN EXCEPTION

Don't think this is an unusual case to be handled as an exception. Major preventive maintenance is costly in labor, materials, and (especially) downtime. If we fail to use this information in selecting items for resupply, our system will consistently fail to keep in stock the very parts that are most needed. Listen! Did you hear those voices? They sounded like, "We met the goals," followed by, "What we really wanted was . . . ."

Taking preventive maintenance into account, an item should go on report if on-hand-stock-qty plus already-ordered-qty minus planned-maint-qty is less than qty-that's-too-low. In other words, we need a record for expected part-out events. We happen to have one handy.

Call the two scenarios "resupply suppression based on ordered quantity" and "resupply triggered by planned use." There's something thought provoking about them. Notice three things: they represent significant requirements, they would not have been noticed, and intuitive solutions would have led to blind alleys. They represent significant requirements because failure to handle either would have made MESPIS unacceptable, generating emergency change requests within days after installation. They would not have been noticed because the craft of design centers around asking what output the user wants. Yet neither case affects output format or content in the slightest. Intuitive solutions would have led to cul-de-sacs because, in both cases, the solutions would have buckled under postinstallation pressures. Suppressing every item once it's been listed leads to insufficient reorder, and treating routine preventive maintenance as an exception ignores the heaviest spare parts use of all.

Yet the future-event pattern provided the solution to both, even before the problems became visible. ◉

Frank Sweet is corporate manager of data administration for the Charter Co., Jacksonville, Fla.