

# TECHNICAL NOTES

VASIT06



**Why MandateIP®?**



As we approach the corporate or information technology representatives of our future clients, a near universal set of statements and questions arise surrounding their need for our software.

Typical questions and statements are:

- Our software works just fine, that is not our company's problem
- Our software already does that
- Why introduce a completely new architecture into our current, well constructed, conventional, proven architecture – an architecture that is serving us perfectly well?
- We have a wonderful IT staff and the software they provide our organization includes exactly the functionality we need, why do you think that we need something different?
- We have tried that before and it didn't help

First of all, the need or benefit of MandateIP® in an organization is not predicated on anything currently “being broken” with the current information system. That is like saying a car with a manual transmission (stick) is broken because it does not shift automatically. The primary benefits of MandateIP® are equally applicable to organizations regardless of the state of their current software system!. Just what are those “primary benefits” of MandateIP®?

### **Better operational productivity and higher capacity**

There are numerous VAS white papers on specific means of achieving these benefits, however, for the purpose of this paper we will address only one – handling exceptions. Why would we choose an insignificant opportunity for improvement as an example to show the benefit of a claim that MandateIP® can dramatically improve both capacity and productivity? Hummm...

As we take potential clients on tours of facilities that use our software, one particular facility tour seems to almost always lets them see what is so different about the way we think and how our software works. Early in the tour we pass some bulk pallet rack filled with license plates labeled cases of “reserve” product. We then pass a receiving conveyance system. At this point we describe how we handle and track the reserve product. We ask the question as to what they would expect if we were to query the information system about one of the pallet locations, one of the cases on the pallet or the pallet itself. We then ask them what they would expect to happen if we were to remove one of the cases on a pallet in the rack and then throw that case onto the adjacent receiving conveyor? In “normal” systems the case or carton would be “unexpected” for the conveyance system and would probably be routed to a specific area for processing. *We are not normal* – no one to our knowledge has ever accused us of being normal. Upon the conveyance system scanning the conveyor, our software determines that the carton or case cannot possibly be two places at one time (i.e. on the pallet in the rack and on the conveyor). We immediately update the pallet location (and pallet) removing the case. We then update the location of the carton indicating that the carton is on the conveyor. We then determine what is the “best thing” to do with the unexpected carton. The decision process is “opportunistic” in looking for the best way to handle the current situation. The rules for determining the “best thing” to do with the carton are unique for each particular client, and in this particular system we first see if the carton could fill an outstanding outbound order. If so, the carton is routed to shipping for labeling and shipment. Any pending pull for the shipping carton is cancelled. If we cannot ship the unexpected carton, we examine the current active stock level to determine if the carton could be used for replenishment. If so the carton

is routed to the replenishment area delaying the future replenishment effort. Finally, and at last resort, we would route the carton back to reserve area for putaway.

At this point in our tour we usually get a “hummm – that is certainly different”. Hopefully, you too have had a “hummm” moment. If not, you are probably thinking, “people should never do that – if someone does they need to be fired immediately”. We agree, people should not do that! We are in no way advocating “chaotic” management of an operation. For those that are not yet “hummm-ing” don’t worry about the specific example we just presented, think of how such opportunistic handling of exceptions - change would work in other situations. Think of how so many things just get messed up by operations in the execution of the perfect plans your current software has prepared. You might also consider how changing production objectives are handled, how un-expected priority work is incorporated into current workflow. Then think “what if” and reconsider the example.

If you are not “hummm-ing” at this point, the balance of this paper will most likely not be of any value.

What are the implications of an “opportunistic” approach of dealing with unexpected events? The dramatic benefits that our software provides is the result of continuously taking advantage of every single event that occurs within the operation. We continuously take advantage not only of the exceptions, but we also take advantage of every successful execution of a task. The basis of the entire concept is in “how change is handled”. When we talk of labor balancing, continuous workflow, waveless processing, dynamic optimization, idle time reduction, all of these specific techniques are based on a single concept.

Another implication is the role of “planning” in a system. A “normal” system creates a plan “at once” and then expects the plan to be executed. To take advantage of “opportunities” that arise would require “undoing” the plan and then re-planning. We do not think that way (our moms were right). Planning is done in order to meet a set of objectives. The plan is not the objective. Planning is a means to an end. We look at an “evolving” plan where the objective is always kept in focus and every activity undertaken is to meet those objectives. The thought may arise as to how much less “thinking time” a computer takes to make an “at once” plan. That is really not the true, the same decisions (other than creation of rules for handling exceptions) to meet objectives need to be made for both a batch (at once) plan and an evolving plan. In the evolving plan those decisions are just spread over time.

Another implication of the “opportunistic” approach is the relationship between the computing system view of “what is” and the physical view of “what is”. In a “normal” system, an underling paradigm is striving to have the physical system meet the information system view. Our systems have a differing paradigm where the “data is a model” of the physical and should represent the physical view as best possible.

Another implication of this approach is the nature of the computing platform. MandateIP® systems operate on “real-time” computing platforms. “Real-time” in this context indicates that decisions are made on-the-fly and that responsiveness to these decisions must not impact operations.

There are other implications of MandateIP® based systems, and you will discover some of them as you contemplate the approach. We would encourage you to go back and think “what if” as you consider the benefits of our

approach. The inclusion of MandateIP® does not imply an “all or nothing” approach. Our systems can integrate into your existing systems in specific operational areas where you provide us with batch data and we then “opportunistically execute” your and report back to the progress.

We will make a very bold statement that these techniques for dynamically planning, optimizing, and controlling operations will become the foundation of the next generation of systems for supporting distribution, fulfillment and production systems. Hummm...