International Journal of Business, Human and Social Sciences
ISSN: 2517-9411
Vol:14, No:11, 2020

The Effect on Lead Times When Normalizing a Supply Chain Process

Bassam Istanbouli

***Abstract*—**Organizations are living in a very competitive and dynamic environment which is constantly changing. In order to achieve a high level of service, the products and processes of these organizations need to be flexible and evolvable. If the supply chains are not modular and well designed, changes can bring combinatorial effects to most areas of a company from its management, financial, documentation, logistics and its information structure. Applying the normalized system's concept to segments of the supply chain may help in reducing those ripple effects, but it may also increase lead times. Lead times are important and can become a decisive element in gaining customers. Industries are always under the pressure in providing good quality products, at competitive prices, when and how the customer wants them. Most of the time, the customers want their orders now, if not yesterday. The above concept will be proven by examining lead times in a manufacturing example before and after applying normalized systems concept to that segment of the chain. We will then show that although we can minimize the combinatorial effects when changes occur, the lead times will be increased.

***Keywords*—**Supply chain, lead time, normalization, modular.

I. Introduction

SINCE supply chain processes are the backbone of most organizations, it is important that they are flexible and well designed in order to reduce the ripple effects when changes happen. Normalization is a concept, when applied it minimizes those effects by making those processes evolvable and modular. But also lead times are important. "In particular, lead-time and inventory management play a crucial role in companies' overall supply chain efficiency and responsiveness. Effective lead-time management is considered a source of competitive advantage" [1, p.53]. Therefore, when modularizing a supply chain, we must search for an optimum solution that satisfies both flexibility and customers' needs for their orders lead times.

The purpose of this paper is to point out that although normalization makes the supply chain processes more flexible, it can also have a negative impact on lead times. This paper will try to show the impact of normalization on lead times and point out that although the combinatorial effects may be reduced when SCP are modularized, the lead times may get longer.

Definitions

Following are some essential definitions that will help readers understand the material written in this paper.

Bassam Istanbouli is with the Antwerp University Department of Economics, Belgium (e-mail: bistanbouli@gmail.com).

Combinatorial Effects

It is the impact on a system proportional to the system's size, not to the size of the change: "Functional changes causing impacts that are dependent on the size of the system as well as the nature of the change correspond to instabilities of the information system." Those instabilities are called combinatorial effects [2, p.270].

Evolvability

Evolvability was mostly defined for software, but in our case, it also can be related to supply chain processes. One of the good definitions related to our subject is: "we describe evolution as changes in a system's environment (domain), requirements (experience) and implementation technologies (process). Then we define evolvability as a system's ability to survive changes in its environment, requirements and implementation technologies." [3, p.2]

Lead Time

From the supply chain material flow model, Fig. 1, lead times are the combination of the following time-periods: Procurement time + Suppliers (S1 … SN) preparing and shipping time + Manufacturer receiving into RM warehouse and inspection time + Issuing material to the manufacturing plant + Producing the products + Receiving the finished product into the FG warehouse + Shipping to distributors and/ or customers

In our case study, we will focus on the issuance and receiving material duration times from/to the RM warehouse and leave all other variables constant.

II. Literature Review

Following are some literature reviews relevant to this paper in an effort to build the knowledge needed by the readers to understand this paper.

Normalized System and Modularity

Modularity was the starting block and platform from which many concepts emerged, including the normalized systems theory. The modularity concept was applied to many areas like product designs, software designs, and business processes etc.

The concept of normalized systems theory started at Antwerp University. "Normalized Systems (NS) are new modular structures with unique evolvability characteristics where combinatorial effects are systematically controlled or eliminated" [4].

Normalization of the supply chain processes (SCP) is about finding an optimum modular structure of that chain where the combinatorial effects are controlled or minimized when

International Journal of Business, Human and Social Sciences
ISSN: 2517-9411
Vol:14, No:11, 2020

changes occur.

The concepts of modularity started with software in 1968 with McIlroy and the concept of assembling programs instead of writing them by making them modular. "The idea of subassemblies carries over directly and is well exploited. The idea of interchangeable parts corresponds roughly to our term `modularity,' and is fitfully respected" [5, p.138]. Modularity is about breaking down large complex systems into smaller, loosely coupled blocks. It is a way to manage a complex system by dividing it into modules "—which can then communicate with one another only through standardized interfaces within a standardized architecture—one can eliminate what would otherwise be an unmanageable spaghetti tangle of systemic interconnections" [6, p.19].

As seen in Fig. 2, it represents a system A, part (a) as one block with various inputs and outputs, and part (b) the same system but broken into smaller modules.

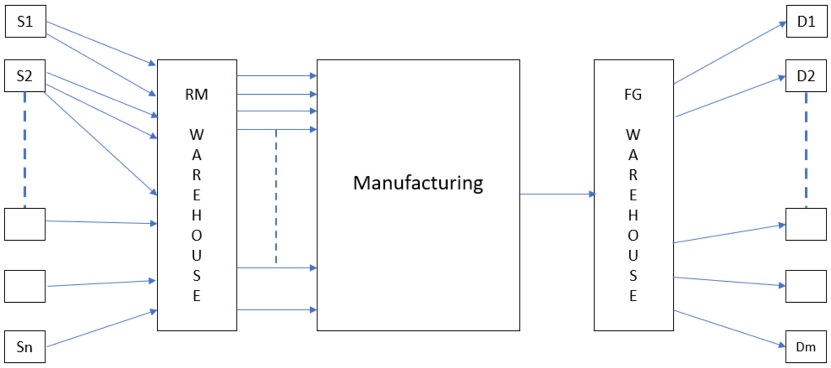

Fig. 1 Material Flow from Suppliers to Distributors/Customers

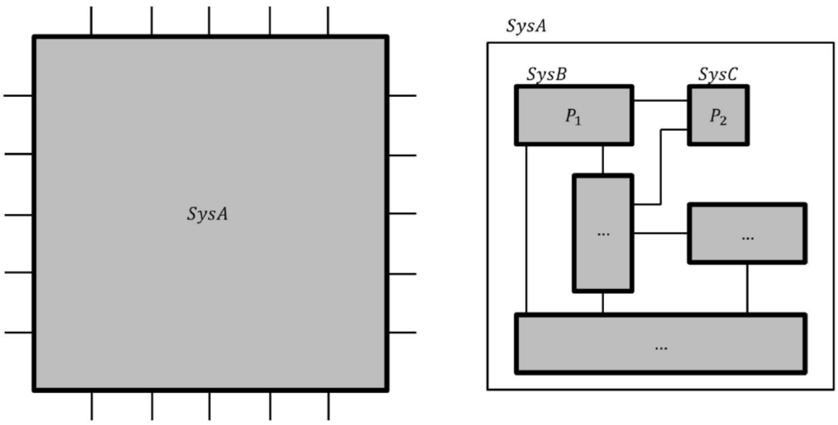

Fig. 2 (a) System A as one black box, (b) System A broken into smaller modules [2, p.101]

Modules are made of modules which in turn are made of modules [2], hence a complex system can be divided into modules which in turn can or may be divided into modules and so forth until we reach an optimum solution for our analysis or get to lowest module which maybe at the component level.

When a system is broken down into smaller sub-systems, those sub-systems/modules should be simple and should be able to be changed without the knowledge of other modules or affecting them. Also, the changes should not affect the interfaces. Hence major changes can be done to individual modules only [7].

"Modularity is a strategy for organizing complex products and processes efficiently. A modular system is composed of units (or modules) that are designed independently but still function as an integrated whole" [8, p.86].

Modularity and Evolvability

In software, programs change continuously and as they change they lose their usefulness and at the end, it will be more cost effective to change the whole program. "As an evolving program is continuously changed, its complexity, reflecting deteriorating structure, increases unless work is done to maintain or reduce it" [9, p.1068].

Modularity increases evolvability: "when we modularize a system and later changes happens, then only limited changes should happen to the system and only few modules should be affected" [2, p.126]. Also a module is considered stable if "…the behavior of the model is not excessively sensitive for small changes in the quantities which steer the behavior of the model" [2, p.51].

There are two rules to follow when modularizing a system: high cohesion and low coupling [2]. As seen in Fig. 3: "High cohesion means that the internal elements/components of modules should have a high degree of cohesion, they should tightly stick together. While Low coupling means that the modules should be connected loosely together" [2, p.22].

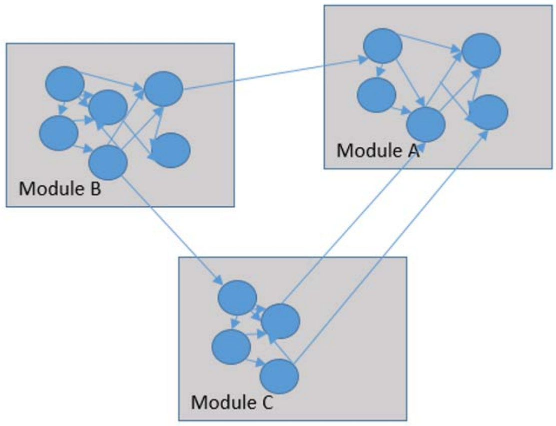

Fig. 3 Example of High Cohesion and Low Coupling

After a system is divided into modules with high cohesion, the integration process starts and integrates those modules together, with low couplings, so that they will act together as the original system. This process is not easy, as given in the following example of "the addition of pipes for water and electricity in a building plan. In case a non-experienced architect did not take this into account…there is a fair chance that diverse walls and constructions will have to be modified" [2, pp.104-105].

III. Case Study

In this case study, manufacturing a bicycle is used as a simple example in order to demonstrate the concept. Processes can be much more complex in a manufacturing environment. The idea here is to show a simple example of impact of normalization on lead times.

Let us take the bicycle as the example (Fig. 4) in this study.

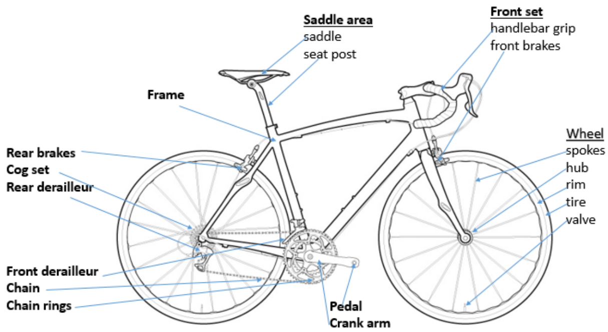

Fig. 4 Bicycle and its components

Fig. 5 is the Bill of Material of the bicycle example. As seen, the bicycle, which is the finished product, consists of many components: the first level are called subassemblies and the second level are just components that can be bought from suppliers and stored in a raw material warehouse.

For the purpose of this paper, two scenarios for manufacturing the bicycle are used. The manufacturing part of the Supply Chain is used, as seen in Fig. 6.

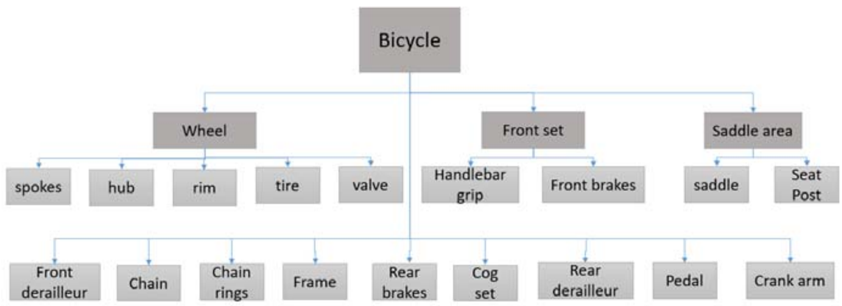

Fig. 5 Bill of Material of the Bicycle

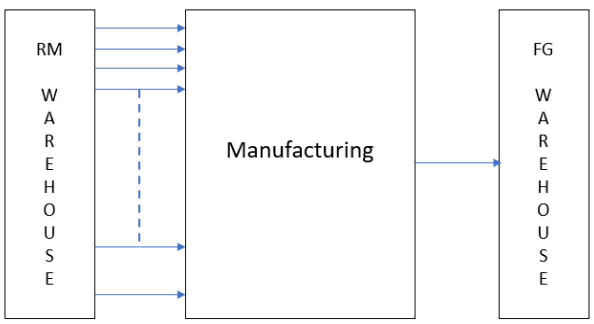

Fig. 6 Manufacturing part of a supply chain

Scenario 1: Before Normalization

In this scenario we are looking at the manufacturing floor as one block, one working area, in which all the parts are assembled together to produce a bicycle. Fig. 7 shows the raw material issued from the RM warehouse that goes into that working area to produce one manufactured bicycle that is sent to the FG warehouse. So, we feed the manufacturing floor with all the parts:

- A = 2x (spokes, hub, rim, tire, valve)
- B = 1x (handlebar grip, front brakes)
- C = 1x (saddle, seat post)
- D= 1x (front derailleur, chain, chain rings, frame, rear

brakes, cog set, rear derailleur, pedals, crank arm)

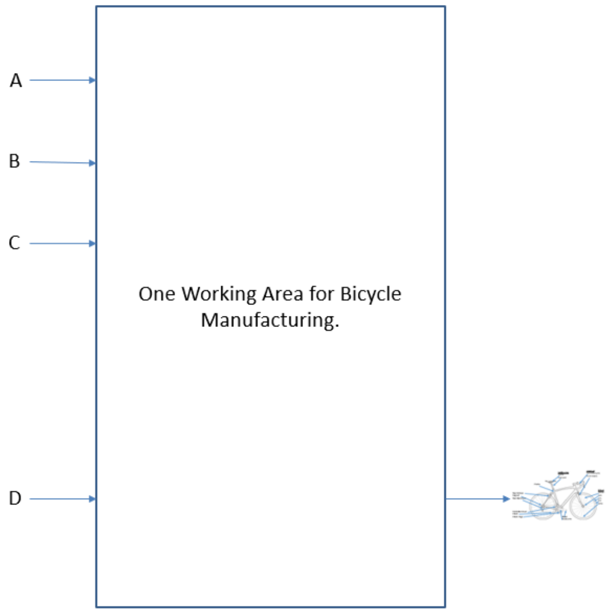

Fig. 7 One work-center for assembling a bicycle

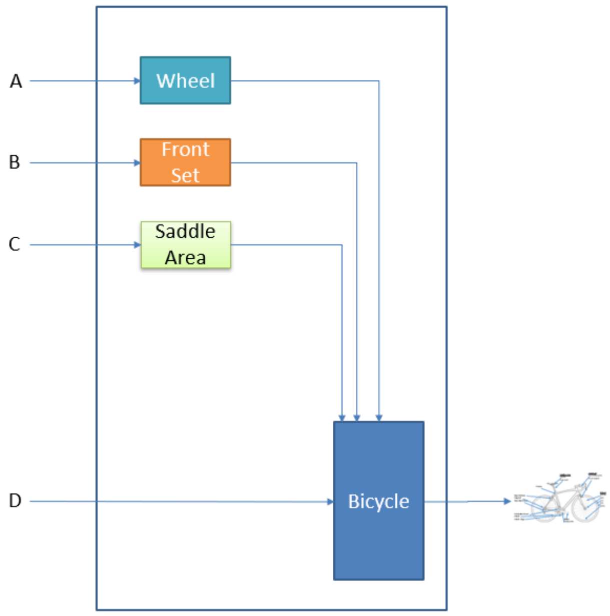

Fig. 8 The assembling area of the bicycle

All those parts will produce one bicycle. If we consider that the lead time for issuing material from stock is the same for all parts Lrm (Lead Time Raw Material), then to build one bicycle we need 18 x Lrm units of time for issuing only.

What happens if something changes in the scenario: Since there is only one work-center and the assembly of the bicycle is done in a series, then any delay in any part of that work area will result in halting or delaying the whole assembly line. Also WIP (Work in Process) is one batch; therefore, any requirement for a change, example a recall or a modification will result in scrapping all WIP, since batches cannot be isolated.

If the processes need to be changed, there will be a tremendous amount of documentation to be changed. Including production orders since everything is connected.

Scenario 2: After Normalization

Starting the normalization process, we can see that the working area for assembling a bicycle can be made of four work-centers instead of one assembling area, as seen in Fig. 8.

If we modularize the manufacturing process by assembling the wheels, front set and saddle area separately and terminate those as subassemblies to the Raw Material Warehouse, and then issue them back with all the rest of the material to another work-center (Bicycle) to be assembled as shown in Figs. 9 and 10, respectively.

Phase 1: Subassemblies

To create the subassemblies, we issue the following:

- A = 2x (spokes, hub, rim, tire, valve)
- B = 1x (handlebar grip, Front brakes)
- C = 1x (saddle, seat post)

which are 9 lead issuance lead times, i.e. 9 Lrm. Then we terminate those subassemblies back to raw material as shown in Fig. 9. Let us assume that the termination time has the same time as picking time that is for terminating the subassemblies we need another 3 Lrm, so the total stock movement lead time will be 12 Lrm.

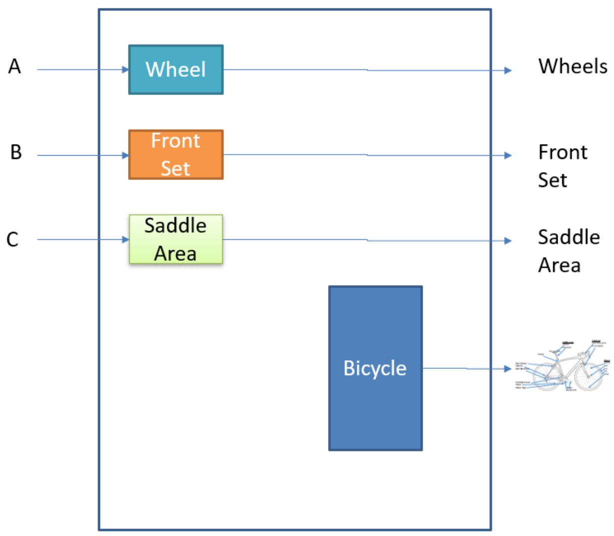

Fig. 9 Creating subassemblies for Wheel, Front Set and Saddle Area

Phase 2: Final Product

To create the final product (the bicycle) as seen in Fig. 10, we need to issue:

- A = 2x wheels + 1 x front set + 1 x saddle area
- B = 1x (front derailleur, chain, chain rings, frame, rear brakes, cog set, rear derailleur, pedal, crank arm)

Hence, a total of 12 x Lrm, since we are not taking quantities into account as it is assumed that the same materials are stored in the same bin, and therefore picking one or two of these items does not add to the lead time.

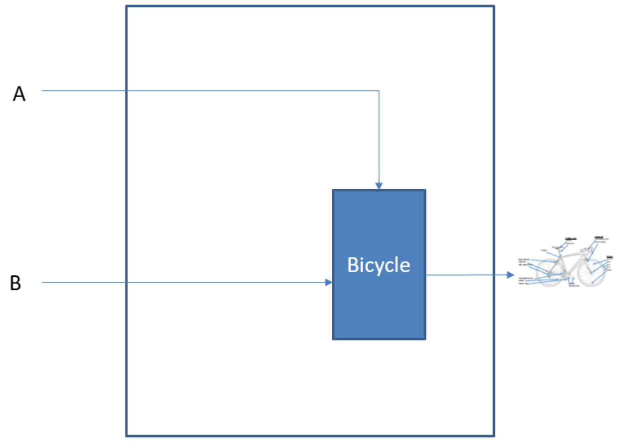

Fig. 10 Final assembly using the subassemblies

In scenario 2, we have: 12 Lrm for building and terminating the subassemblies + 12 Lrm for building the final product = 24 Lrm. Thus, the difference for this simple example will be 6 Lrm.

What happens if something changes in this scenario*:* Since the work-centers function in parallel to each other, any delays in an individual work-center will impact only that part of the assembly line. So, if it is not the main final bicycle assembly work-center that has an issue, then we can still produce bicycles depending how much stock we have from the other subassemblies. Also, WIP is now consisting of multiple batches, any requirement for a batch change will impact a partial portion of WIP only, because in this scenario we will have four batches at a time, not only one batch as in the previous scenario. And, if the manufacturing processes need to be modified or changed, there will be fewer areas and less documentation that need to be changed as well.

IV. Conclusion

There are a lot of variables involved in calculating lead times. This paper assumed many things as fixed and focused on one parameter, just to prove a point. The purpose of this paper was to make the designers of the processes aware that by modularizing a process they may gain a lot of flexibility but that also they may lose some of the lead time.

It may not be always important and sometimes things can be compensated for by access inventory. So, depending on what is more important for an organization, supply chains maybe designed accordingly by finding an optimal solution.

References

[1] Ghaderi, H., 2016, Reducing lead-times and lead-time variance in cooperative distribution networks. *Int. J. Shipping and Transport Logistics, Vol. 8, No. 1, 2016*

[2] Mannaert, H., Verelst, J., De Bruyn, P., 2016. Normalized Systems Theory, From Foundations for Evolvable Software Toward a General Theory for Evolvable Design.

[3] Ciraci, S. and van den Broek, P., 2006. Evolvability as a quality attribute of software architectures. *Journal of Physics Conference Series, January 2006.*

[4] https://normalizedsystems.org/

[5] McIlroy, M.D., 1968. Mass Produced Software Components. NATO Conference on Software Engineering, 1968, pp 138-155

[6] Langlois, R.N., 2002. Modularity in technology and organization. *Journal of Economic Behavior & Organization*, Vol. 49, pp 19-37

[7] Parnas, D.L., Clements, P.C. and Weiss, D.M., 1984. The Modular Structure of Complex Systems. *IEEE*, 1984, pp 408-417

[8] Baldwin, C.Y. and Clark, K.B., 1997. Managing in an Age of Modularity. *Harvard Business Review.*

[9] Lehman, M.M., 1980. Programs, Life Cycles, and Laws of Software Evolution. *Proceedings of the IEEE*, Vol. 68, pp. 1060-1076