IBM Credit Revisited

Timothy S. Vaughan
University of Wisconsin – Eau Claire

This paper provides a critical examination of the reading "IBM Credit" from Hammer & Champy's Reengineering the Corporation. A critical reading provides an opportunity to dig deeper into the scenario presented, and thoroughly examine the source of the performance improvement realized. While the aspect of the scenario highlighted by Hammer and Champy is an important one, generalizing the solution to other contexts requires recognition of additional factors likely to have come into play.

Keywords: process reengineering, product focused system, process focused system

INTRODUCTION

In their book *Reengineering the Corporation: A Manifesto for Business Revolution*, Hammer & Champy (1993) presented a landmark exposition of what has come to be known as "Process Reengineering". The process reengineering concept has two primary components, the first being a "clean slate" approach to the problem at hand. As explained by Hammer & Champy, this involves "fundamental rethinking and radical redesign" . . . "to achieve dramatic improvements in critical, contemporary measures of performance . . .".

While the "clean slate" component is generally recognized (and often mistakenly assumed to be the single distinguishing feature of "reengineering"), the other and more critical component of the reengineering philosophy is the attention to *process design*. This would then distinguish "process reengineering" from the "kaizen" approach (Masaaki 1986), where kaizen might be paraphrased within this discussion as "a continuous stream of small, incremental improvements of the *existing process*", while process reengineering would be "fundamentally rethinking the underlying design of the *process itself*".

Certainly, the importance of process design was recognized prior to publication of *Reengineering the Corporation*. Hayes and Wheelwright (1979a, b) introduced the familiar "product/process matrix", which in a general sense delivers a similar message of matching the design of the process to product/market demand characteristics such as volume and variety. Nonetheless, the Hammer and Champy book remains a landmark publication in raising the visibility of and attention to process design, not only in manufacturing but across the spectrum of industries and environments within which the concept might be applied.

IBM CREDIT

One of the vignettes presented early in the book, and referenced throughout, is an example titled "IBM Credit". This reading describes a case in which IBM Credit, a wholly owned subsidiary of IBM, is dealing with performance issues in their credit approval process. The primary issue was the time required to process loan approvals, which required on average six days but could at times take as long as two weeks.

The initial design would be characterized within one classification scheme as "process focused". Once received, loan applications were routed across different departments e.g. credit department, business practices department, pricer, and quote letter preparation. Within each department an individual would complete one step of the process, corresponding to the specific function of the department in question. In short, the resources (people) involved in credit application processing were functionally organized according to the type of processing they do.

While investigating the cause of the long lead times, two senior managers walked an application through these steps, and discovered that an application typically required about 90 minutes of actual work. The remainder of the lead time is caused by the design of the system, essentially the wait/move (and possibly batching) of the applications as they are routed through the various departments.

The reengineered process as described by Hammer and Champy involved replacing the "specialists" with "generalists". Rather than having four individual specialists, each doing one step of the process, they realized that the typical credit application is (in most cases) a fairly simple operation that could done by a single, suitably trained individual, particularly when provided with appropriate technical support.

The result of this redesign is impressive. The average turnaround time for a credit application was reduced to four hours. More impressive, (and central to the discussion below) Hammer and Champy noted that this was achieved while the volume of credit application had increased to 100 times the original volume.

The IBM Credit case does an excellent job of explaining the difference between incremental improvement vs. fundamental process redesign. Hammer and Champy note that even if the processing time for each of the individual steps could be cut in half, this would only reduce the 90 minutes of work on each application to 45 minutes. Thus, solving the problem required fundamental rethinking of the process design itself, as opposed to simply tinkering with incremental improvements in how the individual steps are done.

CRITICAL ANALYSIS

The IBM Credit reading is an excellent addition to the Principles of Operations Management course. It introduces students to the process reengineering approach, and in particular the importance of critically evaluating the design of the process. Moreover, the reading makes this point using a non-manufacturing example, which helps to solidify the broader relevance of this course content.

Nonetheless, a critical reading of the case provides an opportunity to question the nature of the improvement realized (vis-à-vis the aspect emphasized by Hammer and Champy), and perhaps extend or refine students' understanding of why the dramatic lead time reduction was realized. In particular, we need to address the question of whether the improvement was due primarily to:

(a) Changing the process from one in which each application is processed by four individuals, to one in which each application is processed from start to finish by a single individual, (e.g. degree of specialization and division of labor), vs.

(b) Changing from a process in which the work is done across four departments, as part of the broader range of responsibilities assigned to each of those departments, to one in which the work is done by people who focus solely on processing these credit applications (which, as an aside, also enables those individuals to now process entire applications from start to finish).

Thus, understanding why the solution worked requires an understanding of why performance was poor in the first place. In the exercise described below, we realize the reason the performance was poor is

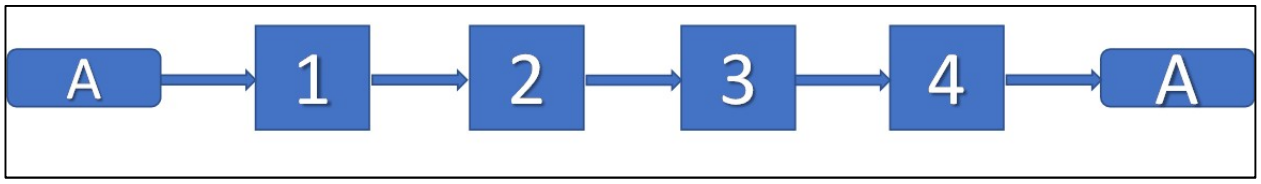
not solely (or even primarily) because different steps were performed by different people. Rather, it appears that the individuals processing the credit applications were not focused solely on that particular workflow.

THE EXERCISE

To begin, students are assigned to read the vignette, and come to class with a written answer to the question, "Would you characterize the original IBM Credit design as product focused or process focused? Explain why." (As the intent is to use the reading on a routine basis, permission to post an electronic copy is acquired from the Copyright Clearance Center www.copyright.com at a modest cost, approximately $1.00 per student.) The concepts of product focused (resources organized with reference to producing a particular product) vs. process focused (resources organized according to the type of processing they do) has been covered in preceding material using a furniture manufacturing example.

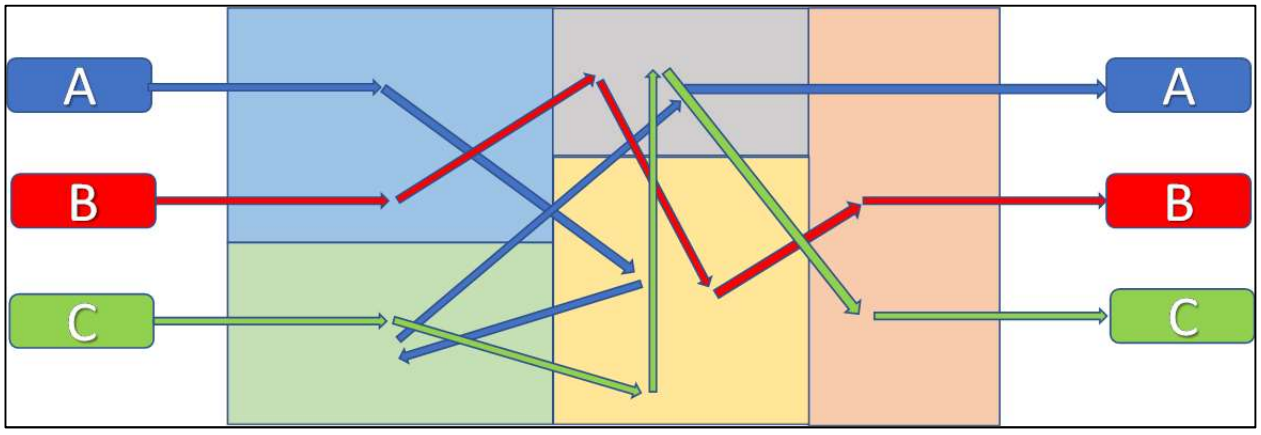
Interestingly, usually about half of the students will suggest the original process was "product focused". When pressed to explain their reasoning, their logic is based on the observation that in the description of the original system, every application went through the same steps in the same order. As such their image of the system reflects work flowing through a system of resources, where those resources are dedicated to processing units through this particular workflow:

FIGURE 1
GENERAL SCHEMATIC OF "PRODUCT FOCUSED" SYSTEM



This provides an excellent opportunity to review and contrast the alternative, process focused system. Here a number of different products follow (possibly variable) routings through a system of shared, functionally organized resources, as shown in Figure 2:

FIGURE 2
GENERAL SCHEMATIC OF "PROCESS FOCUSED" SYSTEM



This review demonstrates the simple fact that having all credit applications (possibly "Item A" in Figure 2 above) go through the same processing steps in the same order does not necessarily imply that

we have a product-focused system. Rather, the question hinges more importantly on whether the resources in question were also responsible for processing other items or workflows.

At this point I prompt those students who answered that the original system was "process focused" to explain their logic. Generally, their rationale hinges on the observation of the long turnaround time, a large portion of which is clearly due to queuing (and possibly batching) of work as it moves from one department to the next. Here I need to clarify that those characteristics are often the *result of* using a process-focused system but are not the definition of what a process-focused system *is*. So, we to return to the question of, "Was there other work (other than the credit applications of interest here) also being done within the original system?"

One key phrase from the reading leads some initial insight into this question. When the IBM Credit managers walked the credit application through the system (in the process of discovering each application required only 90 minutes of processing time), they asked each individual to "put aside whatever they were doing and process this request as they normally would, only without the delay of having it sit in a pile on someone's desk".

Our question is then reduced to, "What was this thing they had to put aside?" For the purpose of our investigation, the question has only two important answers, either (a) another one of these same credit applications, or (b) something else. Under a product focused system, the "thing put aside" would have been another credit application following the same workflow. If the "thing put aside" is something else, we have evidence that the resources in the original system were shared resources responsible for processing items through a variety of different workflows.

More critical reading, coupled with a certain amount of inference is required to shed light on this question of what the "other work" was. As noted above, the revised system was able to process 100 times the volume that the original system contended with, without an increase in headcount. If the original system was only processing credit applications (as opposed to "among a number of competing workflows") there are only two explanations as to how we could now be doing 100 times the volume with the same number of people:

(a) Under the original system the resources in question were at less than 1% utilization. As a result, they are now able to handle 100 times the workload they had before and are now operating at closer to 100% utilization. The problem with this suggestion is that one would not expect to see the types of backlogs they were seeing, if the resources were at less than 1% utilization. (In the class where this exercise is used, we have previously discussed the utilization vs. queue time relationship when first discussing the process focused system.)

(b) The other possible explanation is that we are now processing the applications in about one-hundredth the time that was required before. This would suggest that by having one person rather than four different individuals process each application, total processing time per application has been reduced from 90 minutes to about 0.90 minutes = 54 seconds. This also seems unlikely.

We are thus unable to find a reasonable scenario that would explain the ability to experience a hundredfold increase in volume with the same headcount, under an assumption that there was no other source of workload prior to the change. This leaves us with the remaining option that part of the original queuing and "wait/move time" delay was due to other work, e.g. other than the loan applications the revised system is dealing with. When the employee under the original process had to "put aside what they were doing", the work they put aside was some other task (not a credit application) that was delaying credit application processing. This also raises (in a more speculative nature) the possibility that part of the delays experienced under the original system was due to "batching" the applications as they moved through the system.

We thus realize that the dramatic improvement in performance is not solely (or even primarily) due to having one individual rather than four individuals process each credit application. Rather, the more important explanation is that the work was taken out of a system where the resources (people) in question were responsible for a number of competing workflows. Within this system, credit applications had to

queue behind other work, possibly at low priority. After reengineering, the appropriate resources were instead formed into a system dedicated to only processing these credit applications.

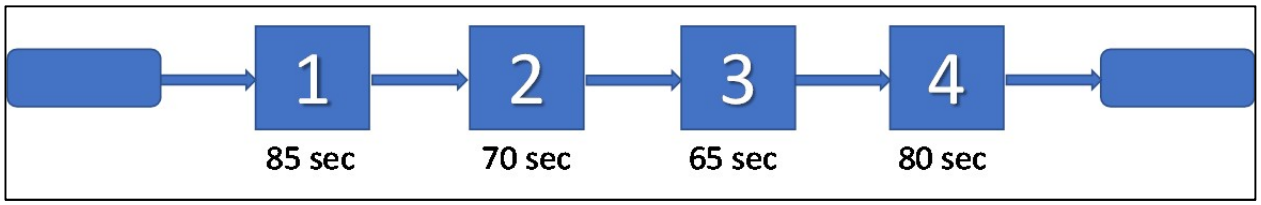
It is easy to conflate the two issues of (a) product vs. process focused and (b) degree of specialization and division of labor. The old system (routing the work through separate functionally specialized departments) implicitly required different people to do different steps. While a process focused system inherently tends to imply division of labor and specialization, the converse is not true. High division of labor does not necessarily imply a process focused system. Rather, a product focused system might be developed using a high or low degree of specialization and division of labor, as appropriate to the case at hand.

DIVISION OF LABOR IN PRODUCT FOCUSED SYSTEMS

To underscore this point, I next have the students work through a new example involving two alternative process designs. In both systems we have resources (four individuals) dedicated to producing a single product, but the two product focused systems use alternative degrees of specialization.

In "Option 1" we have the four individuals assigned to stations in "assembly line" fashion as shown in Figure 3:

FIGURE 3
FOUR PROCESSING STATIONS IN SERIES (DEDICATED TO A SINGLE PROCESS FLOW)



Drawing on previous course content, we confirm that this line would have a nominal hourly processing rate of 3600/85 = 42.35 units per hour, where Station 1 is the bottleneck and the line has a cycle time of 85 seconds. The efficiency of the line is 300/(4x85) = 88.23%. Flow time through the system would be approximately 300 seconds, plus queue time created by arrival and/or processing time variation, particularly at the bottleneck. The queue time would be partly a function of resource utilization within in the system.

Taking the same four individuals and having each person process entire units from start to finish (with each individual doing all four steps rather than "specializing" on just one step of the process), each individual would have nominal hourly processing capacity of 3600/(85+70+65+80) = 3600/300 = 12 units per hour. All four individuals working in parallel would then have combined nominal capacity of 4x12 = 48 units per hour. We also note that the capacity of the sequential system (42.35 units per hour) is 88.23% of the 48 units per hour, where 88.23% is the "line balance efficiency" of the sequential system. (48 units per hour is also the maximum potential capacity of the four-stage sequential system, e.g. if the tasks could be perfectly balanced at 75 seconds at each station.)

Thus, (under an assumption of identical processing rates for each task in both systems, which will be questioned below) we see that having each person do one step of the process will generally result in lower capacity as compared to a system in which each person processes entire units from start to finish. This loss of processing capacity is due to "balance delay", e.g. the capacity loss from having non-bottleneck resources keep pace with the bottleneck. The extent of the capacity loss depends on how well the system can be "balanced". Potential line balance/imbalance is therefore one of the issues that need to be addressed when determining the appropriate division of labor in product-focused systems.

Nonetheless, we should note that "balance delay" (loss of capacity due to keeping pace with a bottleneck) is not the dynamic that generated the long queue times in the original IBM Credit system.

Again, division of labor and specialization *per se* was not the root cause of the processing delays in the IBM Credit scenario.

DIVISION OF LABOR AND SPECIALIZATION

To complete the unit, we proceed into a more general discussion of pros and cons of division of labor and specialization, independent of the IBM Credit case. This is generally familiar content (some dating back as far as Adam Smith's *Wealth of Nations*) and raises issues such as:

- The learning curve effect, which calls into question our assumption that processing rate for each task would be the same in both cases (sequential vs. parallel) above. One could assume that a person who only does "step 1" will develop greater efficiency at step 1, vs. someone who is not specialized. Another potential impact on the processing rate between the two systems is simply the need to continually switch from one task to another if the four individuals worked in parallel.
- Training requirements: The specialized approach generally requires less training (and faster learning curve), vs. training each individual to do all four steps. The importance of this issue of course depends on the complexity of the tasks at hand. As noted by Hammer and Champy, the original IBM Credit system was (at least partially) motivated by an assumption that the individual tasks were complex, and required individuals with a great deal of training and experience in the functional area related to each task. *If this assumption had been valid* (the IBM Credit redesign demonstrated it was not), high division of labor and specialization could have been maintained under a product-focused system, e.g. using the sequential processing system demonstrated above.
- Wasted talent: Extending the issue of training requirements, suppose one of the steps requires extensive training and hence incredibly high wages or salary, while some other step could be done by much less qualified and lower-paid individuals. In such a case, having one person process each job from start to finish implies we have our highly trained, very expensive personnel doing very simple (possibly nonetheless very important) work. (As an extreme example I suggest a hypothetical scenario in which "Step 3" is to do the brain surgery and "Step 4" is to mop the floor afterward. Note both steps are incredibly important!)
- Repetition: In some applications, the specialization concept results in highly repetitive work. Repetitive motion injuries may result when the work in question is physical, and boredom (leading to high absenteeism and turnover) may result in any case.
- Capital investment: A system that permits each person to process entire units from start to finish would require each individual to have their own complete set of necessary tools and machines. To the extent that this represents a significant capital investment, a sequential rather than parallel system minimizes investment.
- Process continuity and integration: Some systems may produce "better" outcomes when processing is done by a single individual who can retain customer information, address special needs, etc. across steps. (This is the "case worker" approach described by Hammer and Champy). Additionally, process improvement is generally more effective when supported by individuals who can perform and understand the entire process rather than just a single step.

Many of the above considerations are mitigated by the possibility of cross training and job rotation, which provides something of an intermediate solution.

In general, then, the issue of division of labor and specialization is a multi-faceted one that should be understood by students and future managers. With reference to the IBM Credit case, students should realize that division of labor and specialization *per se* were not the root cause of the processing delays realized under the original system. The more important aspect is recognition that the original system appears to have been process focused.

CASE EPILOGUE

The final topic addressed in the IBM Credit unit goes back to the question of *why* the original process was designed the way it was. Hammer and Champy offer one possible explanation (or a portion of the explanation), which is the apparent assumption that each step was complicated and required specialized knowledge to perform. This provides motivation for division of labor and specialization but, as discussed above, that specialization did not necessarily *require* the original functionally-specialized resource organization.

Another "piece of the puzzle" takes us back to the hundred-fold increase in volume, and in a sense back to Hayes and Wheelwright's (1979) attention to product life cycles. If we are now doing one hundred times the earlier volume, then previously we were doing one-hundredth (.01 times) the current volume. At the time, then, the IBM Credit designers were most likely looking at (a) a low-volume demand which did not justify allocation of resources to produce only this "product", and (b) existing resources that could absorb the credit applications as part of their total workload.

The point here is that the original IBM Credit design was not necessarily "wrong". Rather, it may have been a reasonable process design at the time, while the "right answer" can change over time as volume and variety characteristics change. Essentially, it appears the IBM Credit process "outgrew" being one component of workload within a process focused system. Once volume became large enough, they faced a situation where (a) the delays have now become a noticeable problem and (b) the volume is now sufficient to justify dedicating resources to this particular workflow. And, (as a secondary consideration) once dedicated to this workflow, it also made sense for a variety of reasons to train these employees, so they are able to process the applications from start to finish. In essence, the larger volume *enabled* the "generalist" approach, by first enabling designation of individuals whose "specialty" was *credit application processing*.

In closing, perhaps the most important lesson students learn in this unit is recognizing the importance of *understanding why* a particular solution worked as well as it did. Absent this understanding, students (and eventually managers) often demonstrate a tendency to simply mimic a solution that has worked elsewhere. Lacking proper understanding, they may try to generalize a solution to some other scenario where it doesn't apply. After reading the IBM Credit case as presented, for example, a manager may expect to achieve similar results by "replacing specialists with generalists", while leaving these "generalists" still responsible for a variety of different workflows.

REFERENCES

Hammer, M., & Champy, J. (1993). *Reengineering the Corporation: A Manifesto for Business Revolution.* HarperCollins Publishers, New York.

Hayes, R. H., & Wheelwright, S. C. (1979a, January). Link Manufacturing Process and Product Life Cycles. *Harvard Business Review*. Retrieved January 24, 2020, from https://hbr.org/1979/01/link-manufacturing-process-and-product-life-cycles

Hayes, R. H., & Wheelwright, S. C. (1979b, March). The Dynamics of Process-Product Life Cycles. *Harvard Business Review*. Retrieved January 24, 2020, from https://hbr.org/1979/03/the-dynamics-of-process-product-life-cycles

Masaali, I. (1986). *Kaizen: The Key to Japan's Competitive Success*. McGraw-Hill, New York.