A verification study of the Patterns for Reducing Exceptions and Improving Business Process Flexibility

Yang LIU, Sanetake NAGAYOSHI, and Junichi IIJIMA
Tokyo Institute of Technology, Graduate School of Decision Science and Technology
Department of Industrial Engineering and Management

Abstract: Exceptions are the events or situations that prevent business processes from completing normally. Authors proposed a conceptual level pattern for reducing exceptions in our previous studies. In this paper, authors try to validate our previously proposed patterns for reducing exceptions with a case study in Small and Medium Enterprise (SME) to show how DEMO could be used in analyzing exceptions. Furthermore, exception handling level is also clarified and compared. Through this research, authors discovered that exceptions generated by transformation, like business model change and strategy change, could not be solved only by improving implementation level of “how current work is done”, but also has to consider about “perform current work differently” and “perform different work”.

1 Introduction
An exception is an event that disrupts the normal flow execution. Like in computer science, when a program encounters a situation that it cannot effectively resolve, an exception will be raised. An exception must be handled immediately or the computer program will be terminated.

In the business world, exceptions also exist. A customer might reject delivered products and/or services because the customer is not satisfied with the quality of the products and/or services. Events such as cancellation of order and termination of contract are also exceptions. Such exceptions prevent business processes from moving forward (e.g. entering into negotiation mode or finish mode) until they are handled.

Previous studies [1] suggest that handling exceptions incur extra time, managerial effort, and cost. For example, when a customer rejects delivered products and/or services, the salesperson in charge and his or her manager may need to take time to negotiate with the customer until an agreement is reached. They are sometimes regarded as normal parts of business process such as declining customer’s order. However, it is still worth being specially focused, because efforts on reducing exceptions might provide opportunities for improving the business process, hence enable enterprise innovation.

In general, there are three broad ways to reduce exceptions, all of which involve consideration of the work of the enterprise [2]:
1. Improve how work is currently performed, e.g., reduce variability by importing information system. This is a basic business process improvement to reduce exception.
2. Perform current work differently, e.g., web-enable customer service. This is related to operational change for reducing exceptions, which will also lead to business process change.
3. Perform different work, e.g., outsource manufacturing and focus on service. This related with strategy change which will leads to both operational change and process change.

Despite the importance of understanding exceptions, there is a lack of research on this topic. Nagayoshi,Liu and Iijima [3] proposed patterns for reducing exceptions, which provide an alternative methodology for analyzing and reducing exceptions based on DEMO [4]. In this study authors verifies our previous study by applying another case study to prove the relevance of it.

The remainder of the paper is organized as follows: Firstly, related researches are reviewed in chapter 2; secondly, our proposed patterns for reducing exceptions are introduced in chapter 3. After that, the patterns are tested by another case study based on DEMO in chapter 4. Finally, some discussion and conclusion are given in chapter 5.

2 Literature Review
Despite the importance of understanding exceptions, there is a lack of research on the topic of exception handling.

Some researches on the topic related with business process flexibility and adopt the workflow perspective [4], [5], [6], [7]. However, authors argue that it is difficult to analyze exceptions from the workflow perspective because workflow is an implementation level method, which is difficult to provide a complete view for analyzing the causes of exceptions, hence also difficult to solve them.

Other solutions take position of Total Quality Management [8], to make continual change a routine undertaking. Typical one is described in David [1]. This study provided an idea of control organization by including the generation and operation of organizational artifact to solve dysfunctions. This study makes it possible to handle exceptions by improving “how work is currently performed” by automatically detect and address dysfunction. However, there are exceptions that may not be caused by routine but by high level problems. These exceptions can only be solved in upper level transformation such as: by “perform current work differently”, to provide additional support and management; or by “perform different work”, which leads to a more functional enterprise transformation.
DEMO [1] is an ontological that is model abstracted from all realization and implementation issues to show only the essence of the operation in an enterprise. It analyzes an enterprise as social entity that constructed of cooperation and authorized actor roles, as well as communications and productive action sequences between the actor roles.

DEMO could become a good methodology because of its complexity reducing ability and human central specification, to analyze exceptions from comprehensive level of an enterprise. However, it is still worth researching that “How can the comprehensive level solutions be linked with business process to handle the exceptions well?”.

3. Patterns for Reducing Exceptions
To close the gap mentioned above, Nagayoshi, Liu and Iijima[3] proposed three patterns for reducing exceptions. The patterns can be classified by whether it has ontological change or not. The patterns are shown as follows:

**Pattern 1**: Involve new transactions and corresponding actor roles in ontological level may reduce exception. Ontological model change represents the enterprise reduces exceptions by “perform different work”. Correspondingly, we have three sub-patterns:

- **Pattern 1(a)**: When initiator of transaction is inside the organization boundary, add pre-decision/management type transaction inside boundary might reduce exceptions.
- **Pattern 1(b)**: When executor of transaction is inside the organization boundary, add supportive type transaction to the executor of transaction might reduce exceptions.
- **Pattern 1(c)**: Provide additional service to outsiders as a boundary transaction might reduce exception.

When there is no ontological change, the exceptions can still be handled. In this situation we can see most of the problems that cause exception are routine related. Then we can cope with the exceptions by performing “current work differently” or by improving “how work is currently performed”. Then authors have pattern 2 and 3 as solutions in implementation level.

**Pattern 2**: Verify the actor’s responsibility in implementation level may reduce exceptions.

- **Pattern 2(a)**: If there is an indeed authority expansion of the role with new action rules and increased responsibilities on ontological level, with which the previous actor could not fulfill.
- **Pattern 2(b)**: If there is no indeed authority expansion of the actor role on ontological level, the implementation level adjustment of actors who play the role may also reduce exceptions if the actor takes helpful authority of the other actor role at the same time.

**Pattern 3**: Ensuring complete information in communication loop on implementation level might reduce exceptions.

4. Case Study
4.1 Company Description – Company A
“Company A” is a small retail company founded in 2005. The main business of “Company A” is importing baby goods, such as baby carry, cart, and toys from different countries and selling them locally in Japan. It is running around 50 brands all over the world. At the time of funding, its business used to be totally based on web shopping site without any tangible stores. The headquarters, double with warehouse, is located in the center of Tokyo. It was easy to manage the whole business by just the president and a few part-time job staff. When customer order comes, the staff will check the inventory. If the production is available, they will ask for the payment of the order, after that the goods will be delivered to customer through some deliver servicer. Inventory will be updated weekly. After two years of growing up, three real shops were opened. The first one located in the place just 5 minute-walk from the headquarters. Some small things may be stored in the shop; however most of the goods they sell are stored in the nearby headquarters. For example, when customer wants to buy a baby cart, if it is not in the shop, the ordered cart will be prepared and delivered from the warehouse to the shop in several minutes. If it is not in the warehouse, the staff can also help the customer to book it if necessary. It is not a complex process so they just use a spread sheet to manage the entire inventory.

When the second shop was opened in Yokohama and the third shop was opened in Fukuoka in 2011, “Company A” began to face some challenge in their business process. All the shops have their own inventory because of the huge amount of information for about 1,000 kinds of items. Hence ignoring the huge amount of reports, the staff prefer to communicate through phone to confirm the specific situation. This leads to inefficiency and imprecise inventory control and redundant communication payment.

Because of the communication issues above mentioned, it became difficult for the staff in the headquarters to control variety and quality of purchase. Also because of the inefficiency of their communication method, repeatable issues always generated in different shops. At the result, additional effort and time ware wasted.
4.2 Exceptions Analyzed by DEMO

Figure 1. ATD of “Company A” (T04.a is the deliver through deliver service provider; T04.b is the deliver to walk by customer)

Five main exceptions in “Company A” are classified and analyzed as follow:

Exceptions in Transaction T01:
1. Request of customer is rejected by order completer because there is no available inventory (request of T03 will be rejected).
2. Goods are declined by customer because of the unsatisfied quality.

Exceptions in Transaction T03:
3. Order preparation is rejected because there is no available inventory to fulfill the order.

Exceptions in Transaction T05:
4. Though transaction T05 is a self-initiated transaction, there exist still exceptions. Incomplete requirement information leads to improper purchase and redundant inventory, hence it might generate cash flow issues.

Exceptions in Transaction T06
5. In case that supplier could not fulfill “Company A’s” requirement, the supplier will reject the order.

4.3 Business process improvement in “Company A”

For better handling the exceptions, hence solving communication and management issues, “Company A” began to improve their management and business process. By analyzing the exceptions above mentioned, we can discover that exception 1, 3 and 4 are closely related with purchase and inventory. They are caused by:

- Unclearly responsibility: “Company A” is selling about 1,000 kinds of items from more than 50 brands. In the previous purchasing, there is no specification about who is responsible for which brand’s goods in the headquarters. Salesperson in retail shop just randomly contacted with the staff in the headquarters and order their purchase requirement. Responsibility of the purchase manager and purchaser were mixed.

- Inefficiency communication between purchaser and salesperson: Before their improvement, whenever store’s inventory was under the safety inventory, the shop asked for purchasing. This generated a mass communication and inefficiency purchasing process. Also, though sale person reported their inventory and sales situation every day to the headquarters, because of too much information, these reports could not use efficiently to support purchasing decision.

“Company A” had the following solutions to handle those exceptions

Solution 1: New transaction added to perform current work differently. In fact, purchase management is separated into two sub-transactions. “Store purchase management” and “headquarters’ purchase management”. Transaction “store purchase management” is added to retail shops’ business process. Thus purchase can be separately managed in retail shops, instead of intensively managed in the headquarters. Salesperson and inventory manager in each shop will calculate and report their requirement to purchase manager in the headquarters twice a week about their purchase plan. The day after getting the reports, staff in the headquarters will order necessary number of items according to entire gathered requirement. Therefore it became effective business because purchasing process is subject to the actual demand.

At the same time, actors who play purchase manager expanded from only staff in the headquarters to including the retail shop staff. In real case, salesperson in Yokohama shop plays all these three actor roles as “order completer”, “inventory manager” and “store purchase manager”. These make the communication and management process more efficiency.

Solution 2: Verify actor’s authority to improve how work is performed. Two staff in the headquarters’ office were assigned to take the responsibility of purchase management and purchaser. One is responsible for overseas brand goods and the other is responsible domestic brands. Either of them plays both purchase manager and purchaser.
Solution 3: Ensure complete and efficient information transformation in communication loop to improve how work is performed. Before the improvement, whenever the retail shops have to ask for items, they sent an email by free formatted form to the headquarters. After the improvement, all the requirements will be managed by shop purchase manager and send to the headquarters' purchase manager twice a week with standard form document. Thus communication fee is significantly reduced. Furthermore, “Company A” plans to implement a small ERP package to make their communication flow more efficiency.

By applying solution 1, 2 and 3, purchase becomes well controlled. Redundant inventory and frozen cash flow issue was somehow solved.

For handling exception 2, solution 4 was performed.

Solution 4: To reduce customer’s decline, a supportive transaction, T09 construction support, was added. Before carts are delivered to customer, experienced manager checks the construction of cart to confirm quality.

For handling exception 5, solution 5 is performed.

Solution 5: It was main reason that suppliers could not fulfill the urgent order. To handle this exception, “Company A” improved their information shearing process and make purchase order twice a week instead of random sequence.

<table>
<thead>
<tr>
<th>Pattern1(a)</th>
<th>Pattern1(b)</th>
<th>Pattern1(c)</th>
<th>Pattern2(a)</th>
<th>Pattern2(b)</th>
<th>Pattern3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solution1</td>
<td>Solution2</td>
<td>Solution3</td>
<td>Solution4</td>
<td>Solution5</td>
<td></td>
</tr>
</tbody>
</table>

5. Discussion and Conclusion

This research validates four of six our pre-proposed patterns for exception handling by another case study. Through analyzing this case study, authors find that small and medium enterprise may be different from large enterprises, such as the certain service company [2], in their exception handling level. In that case [2], more exception handling belong to ontological level. It means that they have more mutual implementation level method about “how work is currently performed”. And they paid attention to “perform current work differently” and “perform different work” to find innovation chances to handle hence reduce exceptions. Like transformation from production central to service central. However, though they also would like to have ontological level change like innovation, but most of exception handling methods belongs to implementation level for SMEs because of the resource limitation. It means that the first thing they should pay attention to is “improving how work is currently performed.” In this process, one method could be: encourage actor to play multiple actor roles and using different actors who play the same role can make the whole process more flexible and efficiency; the other one is completing and enhancing information to share among related actors. Also, as the case shows that the process of SME expands always compared with enterprise transformation, like perform current work differently and perform different work. These changes can be reflected to DEMO as ontological level change. Authors argue that only implementation level change may not be able to handle exceptions generated from SME expansion. There should have some correspondingly ontological level transformation as well.

For further research, authors will focus on solving the questions like: “How will the ontological model and implementation model change when business model change?”

References