Application of Object-Oriented Programming Thought in Test and Verification Work

Jianping Yu[a], Peng Zhang[b], Jiaxiang Hong[c], Chong Yang[d, *]

Kunming Shipborne Equipment Resrarch and Test Center, Kunming Yunnan, China

[a]ylyujianping@163.com, [b]bitzhangpeng@163.com, [c]credandx@163.com, [d]e_yangchong@163.com

[*]Corresponding author

Keywords: Object-Oriented, Verification, Programming Thought

Abstract: This paper uses object-oriented thinking to understand the inspection work, extract and sort out the inspection work requirements, establish work ideas models, and establish the relationship between them. Finally, with the problem of zero fault return in the inspection work, the problem of zero fault return is solved by using the most representative programming language java program of object-oriented thinking. The results show that the application of object-oriented thinking to solve problems will be clearer, the goal will be clearer, and the efficiency will be doubled.

1. Introduction

With the continuous reform and development of market orientation and enterprise demand, efficiency issues have attracted more and more attention, requiring employees to respond and satisfy customers as quickly as possible. Leaders make decisions with the most accurate knowledge. As the important work of most production enterprises, the same problem exists in the inspection work. The traditional experience management is more and more unsuitable for the needs of production inspection. How to improve the test efficiency has become an urgent problem for the inspection workers.

The main purpose of object-oriented thinking lies in simplifying the process, making "operations" in the program, and "checking" the main actions of the operation effect, avoiding the length and complexity of the process. Object-oriented (OO) thinking introduces a new way of cognition and representation of the world, from a process-centric problem-solving strategy to an object-centered solution strategy. The focus of attention is no longer a process, but a problem. The various objects involved in each other are, in essence, object-oriented is to determine the subject of the action is "first", and the process of performing the action is "after" [1]. Object-oriented programming (OOP) has many implications. For managers, it enables faster and cheaper development and maintenance processes, and for analysis and designers, modeling becomes easier, generating clear, easy-to-maintain designs, for programmers, object models Elegant and shallow [2]. In order to improve the efficiency of inspection work, this paper aims to introduce the object-oriented ideas widely used in software development into the actual inspection work.

2. Analysis of inspection work characteristics

In the test and inspection work, taking a certain type of product as an example, the process that the product needs to go through from entering to the exit is: the transfer and re-inspection of the product and product information with the producer, the disintegration of the product by the operator, and the assembly of the components. Segment system adjustment, total assembly adjustment, product actual flight test, product recovery, post-test processing, re-inspection, test result determination, product and product information and production side handover, personnel training management, equipment management, product storage management, etc. jobs. In this process, managers need to face the

decision-making of product testing related issues, personnel arrangement and training, coordination between units and units; operators face the operation and implementation of the entire product test; It is effective to control and ensure the safe operation and implementation of product testing [3].

3. Combination of object-oriented and actual testing

3.1 Object-oriented thinking for product inspection

The inspection work performed by the inspector is related to the safety and effectiveness of the product. Therefore, the inspector needs to establish a systematic idea to deal with these tasks, and form a plurality of ideas modules for the work ideas in the inspection work and the ideas for handling the problems. Linking these modules together forms a working idea for the system. When inspectors conduct product inspections with object-oriented thinking, they can quickly find the interface to deal with the problem in these modules, so that the work can be performed in an efficient and high-quality way. This is the methodological theory that the object-oriented is to determine that the action subject is "first", and the action to be performed is "after", that is, to determine the main body "work ideas" of the action, and to perform the action "test work".

When a product enters the field, the inspector must face the work of product handover, product adjustment, product testing, fault handling, and factors affecting product safety testing. These work objects (except for fault handling) are the processes that each product needs to go through, so these work objects are the commonality of this type of product inspection work. To form a system of test work ideas, we need to find out these common attributes and the methods that will perform these actions (ie, the encapsulation process), through which we abstract out the independent classes and shield their internal information. Certain methods change the attributes, and the package clarifies the respective responsibilities of each package unit, so that it is easy to focus on solving problems (faults) to improve work efficiency. The following is an idea of object-oriented thinking to understand the whole field of test and inspection work of a certain product, and finally deal with the problem of zeroing of faults with object-oriented thinking [4-5].

In the whole test work, the inspection work is complicated and cumbersome. The following takes a certain type of product as an example to classify these complicated and trivial work objects, and understand the whole inspection work with a systematic idea.

First, analyze the inspectors' work objects throughout the work. After the products enter the market, the inspectors first face the products that the producers send to the on-site packaging. According to the quality manual of our field, the products need to check the influence of the five major factors affecting product quality (human, machine, material, law, and ring) before implementing the operation. The inspectors need to implement effective supervision and control on the five factors affecting product quality, and timely discover And eliminate the quality hazard, so the inspectors must consider the five factors before the product inspection. For example, when performing product assembly operation inspection, the prerequisites for this inspection object are: must have qualified operators (persons), reliable equipment (machines), complete internal (outer) parts (materials), operating procedures and specifications. (French) and reliable environmental factors (rings). After the above factors meet the requirements, the inspector shall perform product re-inspection, unpacking inspection, and check the relevant handover of product related attributes (product number, appearance, data, etc.). Secondly, after the management's decision, the product will participate in the test, and the inspector needs to carry out the assembly, commissioning and inspection work on the pre-test group components and the whole product. After all the test items in the commissioning process are inspected according to the relevant inspection requirements, the product can be tested for actual flight. Finally, the internal performance data and the external measurement data generated in the actual test of the product are tested and tested. If the product is qualified, the inspector and the production unit will hand over the product. If an item in this process is unqualified, the action will be suspended and the follow-up work will be continued after the fault is

processed. As shown in Figure 1, a product entry test is described, and the inspector is required to face the work object of the inspection workflow.

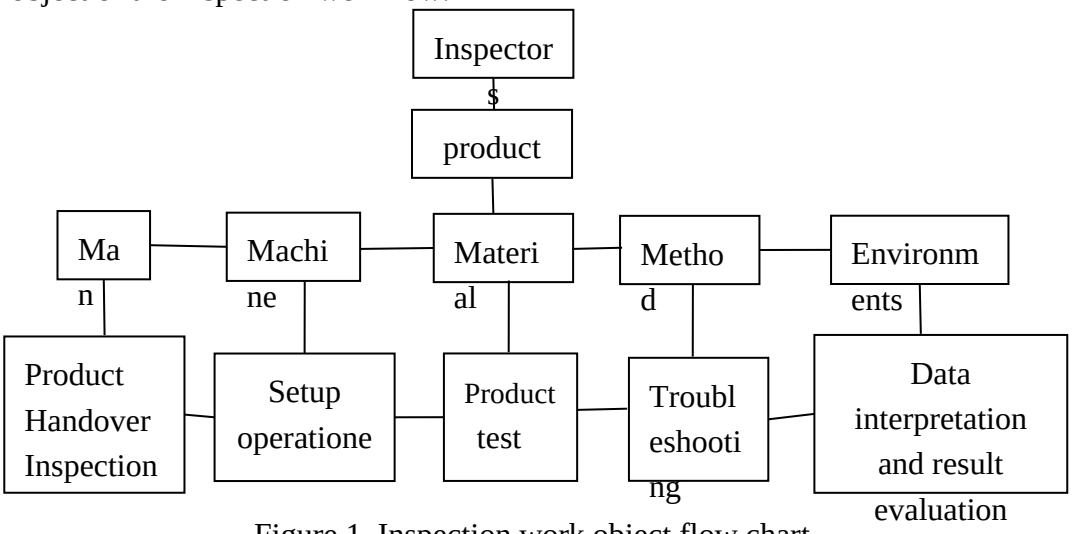

Figure 1. Inspection work object flow chart

The above is the history of each product from entry to exit (except for fault handling), which means that they are the commonality of each test product inspector to face the object of inspection. According to the object-oriented thinking, The commonality of these objects is proposed to form their own unique modules, and then associate them to form a systematic test.

3.2 Inspection work design

The commonality of the above test objects was classified, and eight classes were created, as shown in Table 1.

Table 1. Inspection work set

Serial number	(a)	(b)	(c)	(d)
method	Product Handover	Five factors affecting product quality	Trouble shooting	Product
	Handover criteria; Coordinate and organize various departments to conduct product re-inspection / unpacking / boxing; Product documentation transfer; Product Handover; ……	Man: operation certificate, knowledge skills, comprehensive ability, quality awareness; Machine: performance, legality, security, integrity; Material: complete internal and external materials; Method: related regulations; Environment: reliable environmental factors ……	Types; Communication and coordination Treatment measures Recheck Fill in ticket / information; Relevant procedures ……	Types; Type (first test / second test); Participation requirements; Different from other products; ……

Serial number	(e)	(f)	(g)	(h)
method	Operation inspection	Test inspection	Data interpretation and result evaluation	Inspector
	Preparation before operation (load and evaluate the five major factors affecting product quality); Process and specifications; One-component adjustment, two-component adjustment ... Total weight	Preparation before operation (load and evaluate the five major factors affecting product quality); Operating specifications; Security check; Performance index check; Normal product test;	skills requirement Communicate with the military, the relevant departments on the field, and the production or research and development parties; Data interpretation	Supervise an examination coordination Evaluation

(1) Product handover category: There is an on-site handover and an appearance handover. Whether it is an entry product or an exit product, the product and document data transfer regulations must be followed. The inspection personnel must organize relevant departments to participate in the handover of the product, and finally hand over the complete, legal products and related materials;

(2) Five major factors affecting product quality: From the inspection work flow chart in Figure 1, it can be seen that the five major factors affecting product quality are the process of executing all objects. This process is to judge whether people, machines, materials, methods, and rings are satisfied. Related requirements, it is a commonality before all operations;

(3) Fault handling class: The product has relevant faults in debugging and assembly. These faults are inevitable in each product during the assembly process. The type may be caused by equipment failure, product quality failure, environmental factors. Faults, human faults, etc. When the product fails, the inspectors need to communicate and coordinate with the relevant leaders, production/scientific parties, and customers, organize relevant personnel to conduct troubleshooting, and handle relevant procedures;

(4) Product category: The product is the attribute object in all inspection work, and all operations are carried out around the product. Before the product is tested, the inspector needs to know the product-related attributes and test requirements in advance, because the requirements for the evaluation items and product types of each product are not necessarily consistent. For example, the product type can be divided into the first test product and the second test product, the types are inconsistent, and the products will be different when performing certain component operation tests. The attributes in the product class are generally attributes of product privatization. To obtain and set product attributes, they can only be obtained and set by specific methods.

(5) Operational inspection class: In this class, the five major factors affecting product quality are specifically proposed, which directly relates to product safety and personnel safety, so this is an important part of the process. All operations in this class must be After this link. There are a variety of components in this class. It is necessary to assign multiple inspectors to participate in this module. This requires close cooperation between inspectors to effectively complete this module. If there is a product failure in this class, you need to associate the product fault class to handle the fault;

(6) Test and inspection categories: This category is similar to the operation inspection category. It is necessary to ensure that there are no abnormalities in the five major factors affecting product quality. The test products must be verified again before the actual aviation test, and the safety and performance indicators of the products are verified. It is important to check the final step before the product is actually voyaged. If a fault occurs, the fault class needs to be loaded for troubleshooting.

(7) Data interpretation and result evaluation: This category is a comprehensive performance evaluation of the product internal test data and external test data after the product is tested. In this category, the inspectors must strictly check according to the technical requirements, and cooperate with the customer and the producer. The opinions of the developers and relevant departments in the field were agreed. If it is judged that the data does not meet the requirements, it is also necessary to load the fault class to handle the fault. Finally, the assessment results are linked to the handover category for product delivery;

(8) Inspector class: This class is the responsibility attribute class of the inspector. The inspector earnestly performs duties such as supervision, inspection, coordination, and evaluation.

These eight classes are the methods and attributes proposed to test the commonality of working objects. These objects are the overall process of a product inspection work. Therefore, the eight classes created are abstract classes, and the abstract classes are fuzzy, so the class is only clear. The method, the specific implementation also needs to create a subclass to execute its methods. For example, in the operation class, we know that the participating members have inspectors, operators, and managers. The methods of execution include group component adjustment, test equipment debugging, etc., but the specific components of the model require different members and methods. In these subclasses, these members will be assigned different values, and the methods defined in these abstract classes will be overwritten by different methods. Therefore, when an object is created, the inspector needs to know what members are needed to operate the objects, and what methods are used, but the specific things are implemented according to specific requirements. It can be seen from this that the more fuzzy the class method of the upper layer, the more refined the class method of the lower layer. In the whole process of product testing, it is actually to create objects continuously, and then use the object to call the corresponding class method to perform the work. This class is the test idea, and the object is the inspection work.

There is a relationship between these eight classes. Classes (4) and (8) are the attribute objects of the other six classes, which run through the entire test work object. The created object is executed in the order of (1), (5), (6), (7), (1), and the object is created with the class (2) and the corresponding method in (2) is executed when each method is executed. When one of the products fails, the next object will be suspended, and the fault object will be created by the class (3) to handle the fault. After the fault is processed, the next step is executed.

3.3 Using Object-Oriented Programs to Handle Quality Zeroing Problems

The fault problem is an inevitable problem for each test object, and it is also a key point that must be solved in the product test process. According to the relevant documents and materials in the field, the commonality of the inspectors on fault handling is: check the five factors affecting product quality, suspend the execution of the lower process, understand the basic situation of the fault, communicate and coordinate, perform troubleshooting according to the troubleshooting plan, and reproduce the fault. Fill in the fault/information sheet and check the legality of the fault handling. The Java programming language is the most representative language of object-oriented thinking. The language structure is simple and convenient. The idea of using the Java programming language to solve the problem of zeroing failure is as follows:

```
interface Debugging {//Abstract fault type
void type () ;//Fault type;
void communicate () ;//communication;
void step (); Fault Handling Measures;
void recheck (); Fault recurrence, Learn by analogy;
void page () // Fill in the fault/information sheet and check;
void process () ;// handle relevant procedures;
……
}
class examin {//The class in which the inspector performs the inspection
```

static examiner cheker = new examiner ();
public static void main (String [] args)
throws Exception {
cheker Failure in performing inspection work objects;
create fault object debug;
check invokes methods in DebugMethod in turn by using fault object,
the debug,according to its responsibilities;
cheker is responsible for the failure;
debug Finished processing;
Continue to execute the next work object;
……
}
class DebugMethod implements Debugging {//inherit Debugging
product product = new product () ;//Creating Product Attribute Objects
String modle = product. getModle () ;//Get the current model of the product
int number = product. getNumber () ;//Get the current product number
String task = product. getTask () ;//Getting the current task of the product
String term;// Dealing with prerequisites
String conclution ;//Conclusion
String explain ;//Explain
{
Pause all methods in operation inspection;
}
static {
Inspection of five factors affecting product quality;
}
void type () {
The fault types and phenomena are defined: product quality failure, equipment failure, personnel failure, etc. Major quality faults, serious quality faults, general quality faults?
}
void communicate () {
Reporting failure to department leader;
Notify the client and ask for advice;
Communicate and coordinate with producers and dispatchers to formulate corresponding fault handling schemes.
}
void step () {
Correlate the schemes in the method communicate () method
Scheme used in this method;
Handle the faults according to the scheme of fault handling;
}
void recheck () {
Fault recurrence;
Learn by analogy;
}
void page (Member attributes to be obtained, troubleshooting measures, schemes, comments
Pass in the method) {
Record incoming parameters in paper form;
storage;

```
}
void process ()  {
Check the procedures;
}
……// If the current method cannot meet the fault handling, you can write the method again;
```

When a fault occurs, the object facing the inspector is the fault object, and the fault commonality found above is encapsulated into the fault class. The class encapsulates the fault handling method. These methods are to check the fault handling method, which forms a fault. The thought module, when the fault occurs again, the fault object is used to execute the method in the module. According to the rules of the program in the java programming language, if there is no fault object in the process, only the class is loaded into the memory, and no ordinary method in the class is executed. When the fault occurs, a space is opened in the memory. The fault is also the same in the actual test. When the fault does not occur, the inspector does not have to think about how to deal with the fault at all times. Once the fault occurs, the fault is processed according to the fault handling method. In Figure 3, the program throws an exception from the main method entry after the main method. When the problem/fault inspector cannot solve it, it will rise to the upper management layer and create the executor in the entire inspection class before the main method— The inspection personnel object, the inspection personnel perform the inspection work object according to the normal procedure, and a fault occurs in a certain link in the process, so the fault object (debug) is created, and the debug first quickly finds the fault class's parent class (Debuging) and finds Debuging only clarifies the fault handling method and there is no detailed method. It is just an interface, so I go to the subclass (DebugMethod) to find the detailed method. When I load it into the DebugMethod class, I find that there is static in this class. Load, first execute the static code, perform the check that affects the five factors of product quality. When the fault object is created, the code block in this class is found. The code block is executed when the object is created, so all the methods being executed are suspended. The fault will be processed. Then, the inspector (cheker) uses the debug object to call the method in DebugMethod for processing, and needs related information of the product in the fault processing, so the product product object is created to load the product attribute, and term, conclution, and explain are obtained by this method. Out, the attribute values of these members will be passed to the page() method. The methods in the DebugMethod class override the methods in the interface to provide more specific methods. The following methods are described: type() The method is a basic situation that an inspector must first understand in the event of a failure; the communicate() method is one of the duties performed by the inspector. When a failure occurs, the situation learned in tepy() is related to the relevant personnel. Effective communication, this method determines the troubleshooting plan; the step () method is carried out under the above troubleshooting scheme, and the troubleshooting is performed one by one according to the troubleshooting scheme; the recheck () method is to reproduce the fault after the scheme is completely checked, and Avoid similar failures in future work; the page () method is to completely store the information stored in the member variables. Together to form a paper document (ticket / information sheet) archives; process () method to check whether troubleshooting after the approval of relevant departments of the leadership that is its legitimacy. If the inspector uses the method called by the debug object, the functionality can be expanded. The inspector performs the debugging according to his own duties, and finally restores the working objects after the execution.

The Debugging class and the DebugMethod class are created by the inspector according to the relevant inspection document standards and work experience. When a new admission inspector comes out, he can only create related methods based on the field file, and his experience can be convenient, fast and effective. To solve the problem, you can further expand according to your own experience [6-7].

4. Summary

The above is the use of object-oriented thinking in the entire understanding of the inspection work and the application in the inspection work. It should be noted that applying object-oriented programming ideas to some small problems (requirements) can simplify the problem solving (requirement). For the work problem in the general direction, the application of object-oriented problem solving will be clearer, the goal will be clearer, and the efficiency will be doubled. Applying object-oriented thinking to solve problems in work, the discussion here only provides a way of thinking. In order to deeply integrate object-oriented programming ideas into the work, there are still many aspects worth exploring.

Acknowledgments

We are grateful to the other colleagues in the first laboratory of the Kunming Shipborne Equipment Research and Test Center for their support and assistance in this research. We are grateful to our reviewers for their valuable suggestions and comments, and also to the Organizing Committee. Finally, I would like to thank Director Wang for the language support.

References

[1] Sun Yan. Application Research of Object-Oriented Thought in Management [A]. Logistics Science and Technology, 2006, 29 (133): 129-130.

[2] (US) Bruce Eckel. (Translation) Chen Xiaopeng. Thinking In Java. 4th edition. Mechanical Industry Press.

[3] Chen Henry. Application of object-oriented thinking in the design of avionics system simulation software[J]. Electronic Technology and Software Engineering, 2018(17): 69.

[4] Wu Zhongxing, A Liaosha·Ye, Liu Xingqi, Zhang Yupeng. Design of object-oriented conformance test for communication information collection communication protocol[J]. Electric Measurement & Instrumentation, 2018, 55(15): 65-70.

[5] Zhang Beibei, Zhang Xiaoqiang, Yang Xianggui, Cheng Gongming. Object-oriented quality inspection management system model[J].China Metrology,2017(10):52-54.

[6] Ren Xinbo, Zhang Junyi. Research on Knowledge Ontology and Its Management Based on Object-Oriented Thinking[J]. Value Engineering, 2017, 36(28): 216-218.

[7] Fang Wenyuan. Programming logic modeling and application of object-oriented class source code [D]. Strategic Support Force Information Engineering University, 2018.