THE EFFECT OF INFORMATION MANAGEMENT SYSTEM ON THE PRODUCTION QUALITY

Nechaev K.A.
Scientific advisor: prof. O.V. Stukach
Tomsk Polytechnic University, 30 Lenin Str, Tomsk, 634050, Russia
E-mail: akadelpher@gmail.com

Introduction

MES (Manufacturing Execution System) is a compilation of integrated hardware and software solutions, designed for statistics collection, control and operational planning on manufacturing enterprises [1]. Successful implementation and embedding such a system results in increased production volume, reduced product cost and improved production quality.

At present it is clear that industrial enterprise information system is not optional tool, it is necessary for effective management. Embedding of the information system is very important problem due following: about 80 % of all adverse circumstances have informational basis, and fight against them can only be effective when using comprehensive system of enterprise management [2]. When information system is embedded, you can gain following advantages:

- Increased volume of production;
- Reduced production cost;
- Improved quality of product.

Keep in mind that only embedding MES in itself does not magically improve your production on every level; improvement depends on many factors, and the main factor is information system quality. Architecture defects, bad-quality specification, wrongly organized development cycle – all of it will have its influence on MES quality as well as on entire enterprise. It can even lead to situation when information system embedding just deteriorates production.

Selecting of MES architecture

Usually, the requirements to information system architecture (without binding to any specific area) are the following:

- It should be flexible, loose-coupled and capable of adapt to the different technologies;
- Developers should use modern technologies for increasing system lifetime;
- Architecture must not be bound to any specific security system;
- Modularity is required for reducing number of dependencies between parts of the system.

Wrong choice can lead to situation when scaling and even support of the system become impossible. Then you will either develop new system again or deal with reduced functional. The both imply raised expenses and failed objectives.

The most suitable choice is service-oriented architecture (Fig. 1) [3]. This approach is based on using disturbed, loose-coupled components, equipped with interfaces for interaction [4]. MES designed by this architecture allow simply reduce number of dependencies in production system. Furthermore, such architecture of information system meets modularity requirement, and every module is either service provider or consumer.

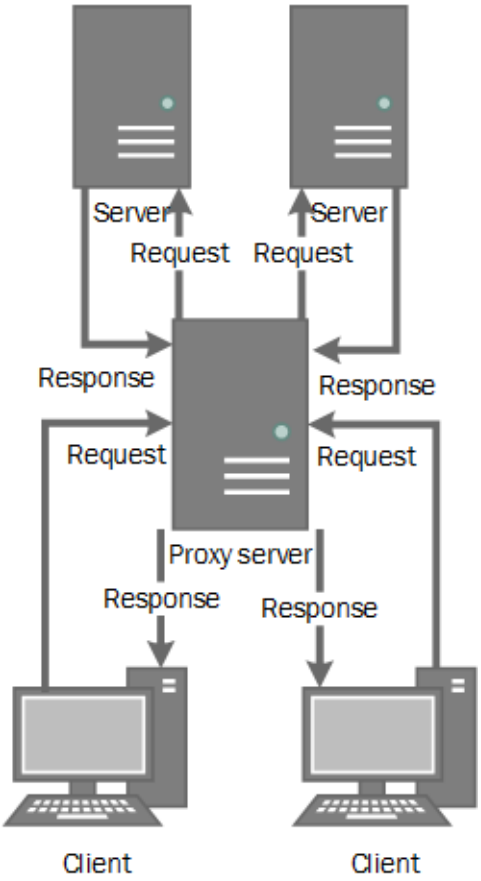

Fig. 1 – Service-oriented architecture

Information system testing

Quality of code is the next important factor defined quality of MES. At first, system must operate correctly under all conditions, provided in specification. Secondly, if user made a mistake, system should report about that without terminating module or the whole system.

Testing process should be automated because of manual testing can become tedious due to large variety of interfaces. Generally, developers use automated unit tests as the main testing method. Unit test is a function, which calls methods of some module or system core. Each test is designed to check if particular function or module behaves as expected. Wherein, developers can guarantee a few amounts of

bugs in tested methods only due to existence of those tests. It is essential to run tests after every development step to see if any new bugs appeared.

Automated unit tests can drastically reduce development time on account of reduced testing time and possibility of early errors detection.

Early errors detection is a key factor in design of the information system. The earlier bug is found the less cost will be of that bug. Thus, code update on the last stage of development cycle will be 10 times more expensive than in the coding phase [5].

When testing, developer has to cover in tests all methods which receive user data. He must sure that such methods are filtered of data, check of types etc. In other words, methods should validate the data. After processing, methods should return adequate response.

In case of force majeure, information system must try to save user data and send crash report to developers. Crash reports are very important when system engineers try to figure out why crash happened. So developers should test logging functionality too, and very carefully. Such testing attitude can help eliminate the causes of crash.

If developer does not pay much attention to testing, system may distort or delete important data, and company will lose money.

Interface of information system

Unfortunately, MES developers don't really think that user-friendly interface is important part of system. So, clients have to spend time and money to teaching personnel how to work with system, and that education process can take a very long period. As personnel usually learns in working time, production volume decreases. If people learn after work, they become exhausted and soon start to make mistakes.

Even after embedding process, bad interface has its influence on quantity and quality of production: if user has to spend a great deal of time just to launch module or understand a system message, he will not be able to concentrate on his business.

Thus, interface of information system plays an important role in both processes: embedding and using.

Information exchange at different levels of the device life cycle

The most important factor which allows increasing production quantity is information exchange implementation between company divisions that produce components of varying degrees of integration: elemental base unit, the module unit, and device installation. Wherein, life cycle can be implemented in a single enterprise (full cycle) and on a number of partner companies (supply chain).

Examples of information, transferred "from bottom to top", are: modes and manufacturing conditions of each specific component, yield rate in a batch, results of various tests, etc. "From top to bottom" can be transferred: storage conditions of obtained components, installation time in the next level component, failure statistics of obtained components, etc.

Effective organization of information exchange between supply chain participants allows implementing comprehensive approach to solving production quality problems and improving quality continuously.

Conclusion

Quality of the information system is determined by many factors, such as system architecture, code quality (of architecture and system modules), code coverage, interface usability, etc. All of this eventually affect the whole enterprise and quality of its products.

If MES is properly designed, embedded and operated, it can reduce production time, reduce human factor influence, increase employee's quality of work and, of course, can be used as a tool to solve production quality problems.

Notice that improvements will not be seen right after embedding [2]. They will appear gradually, after a certain period after starting.

Duration of that period depends on production type, system quality and the willingness of the company and its leaders to such transformations.

Anyway, it will hardly be less than one year, but properly designed manufacturing execution system will pay for investments and bring a positive economic effect.

References

1. Architecture of automated system [electronic resource]. – Access mode: http://www.bookasutp.ru/Chapter1_0.aspx

2. Korablev, A.V. Economic benefits from the implementation of the integrated enterprise management information system [electronic resource] / A.V. Korablev. – Access mode: http://www.r-p-c.ru/sites/default/files/artical_org_kis.pdf

3. Matveev V.V., Nechaev K.A., Zykov D.D. Approach to developing manufacturing execution system // Volga Scientific Bulletin. – 2012. – N 10. – P. 12–17.

4. K.A. Nechaev. Development of Manufacturing Execution System Service Oriented Software Architecture / K.A. Nechaev, V.V. Matveev, D.D. Zykov // 2013 International Siberian Conference on Control and Communications (SIBCON).

5. McConnell S. Code Complete: A Practical Handbook of Software Construction, Second Edition / S. McConnell – Microsoft Press, 2004. – 960 p.