# AHP-Based Software Quality Risk Assessment Method for Information System

Mao Yujun
Shanghai Academy of Spcaeflight Technoligy Beijing
R&D Center
Beijing, China
maoyujun510@yeah.net

Li Lingli,Dong Shuaijun, Lan Guofeng
Shanghai Academy of Spcaeflight Technoligy Beijing
R&D Center
Beijing, China
wlw@wp.com

***Abstract*—with the development of war towards network, intelligence and systematization, the function of military information system is improving constantly, and the quality of system software development has been paid more and more attention. Based on analytic hierarchy process (AHP), this paper constructs the index system of software quality risk assessment, calculates the weight and order of risk factors, and carries out risk assessment analysis. The results show that in order to improve the quality of information system software and reduce the risk of software development, we should pay more attention to the change of requirements and the development process, and reduce the development risk brought by personnel flow and software function change.**

***Keywords—Systematization; information system, software quality; AHP; risk assessment***

## I. INTRODUCTION

With the development of warfare towards networked, intelligentialize and systematization, computer technology has been widely used in military field. Under the condition of system of systems operation, all kinds of weapon equipment and war fighters will gradually become nodes in the system, and the information data interaction level between nodes will increase exponentially; military information system becomes an indispensable part of the operation, the scale and difficulty of information system project continue to increase, and the development quality of system software gradually attracts attention. In the development process, the problems such as the change of military requirements, the lack of project management risk management and control, and the replacement of technical personnel are the important factors of the software quality problems in the information system.[1][2-4]

Based on the AHP, this paper constructs the information system software quality assessment index system, studies the information system software quality risk assessment method, carries on the risk assessment and the analysis to the risk factor, reduces the information system software development quality risk, enhances the software development level.

## II. ANALYTIC HIERARCHY PROCESS

Analytic hierarchy process (AHP) is a multi-objective decision-making analysis method. It is used to stratify, quantify and model the abstract decision-making process, calculate the weight of software quality risk factors, effectively analyze the non sequential relationship between the levels of the objective criteria system, and provide the basis for the final objective decision-making. Analytic hierarchy process is generally divided into four steps: clear research object, establish analytic hierarchy process model, construct quality risk assessment factor judgment matrix, weight calculation and consistency verification.[5][6]

### A. *Ensure research object*

Firstly, it analyzes the composition, function and process of the research object, decomposes the research object into several layers and elements, and determines the internal logical relationship of the research object.

### B. *Establish analytic hierarchy process model*

Secondly, according to the internal logic of the research object, establish a top-down hierarchical relationship to form a criteria layer; establish an independent element system to form a scheme layer; the final formed analytic hierarchy process model is shown in Figure 1.

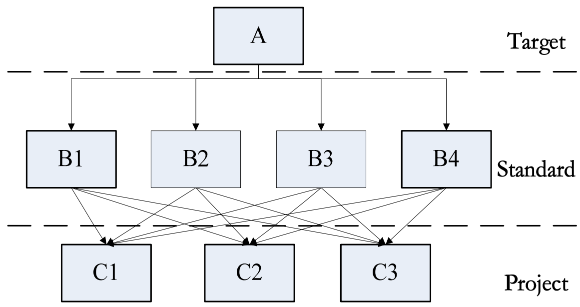



Fig. 1. hierarchical analysis structure

### C. *quality risk assessment factor judgement matrix*

According to the influence degree of each quality risk assessment factor on a certain layer, the weight of each factor in the layer is determined.The comparison results of each factor are determined and the evaluation judgment matrix is constructed by comparing the importance scale value and the two factors.If the importance of the criteria layer in Figure 1 is compared, the evaluation judgment matrix can be obtained as follows:

$$A=\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \tag{1}$$

The judgment matrix has the following characteristics:

(1) $a_{ij}>0, a_{ii}=1, a_{ij}=1/a_{ji}$

(2) the judgment matrix is positive reciprocal matrix.

Among them, $a_{ij}$ It is called comparative scale. The ratio of evaluation factors to the influence or importance degree of the overall target performance. The value range is 1, 2, 9. The meaning of each value is shown in Table 1.

TABLE I. MEANING OF COMPARATIVE SCALE

| *scale* | *Meaning* |
|---|---|
| 1 | The two factors are the same as the influence (importance) on the upper level (general objective) |
| 3 | $A_i$ Slightly greater than $A_j$ |
| 5 | $A_i$ Greater impact than $A_j$ |
| 7 | $A_i$ The impact of the $A_j$ |
| 9 | $A_i$ The influence of the ratio is absolutely great than $A_j$ |
| 2,4,6,8 | $A_i$ The influence of and is between the above two adjacent levels |

*D. Weight calculation and consistency verification*

Firstly, the maximum eigenvalue λ and the corresponding eigenvector of the judgment matrix *A* are calculated, which represent the importance order of each factor.There are many methods to calculate the eigenvector of judgment matrix. If it is difficult to calculate the eigenroot and eigenvector according to the mathematical definition, it is usually replaced by approximate calculation value. The common algorithms usually include eigenroot method and software programming calculation. In this paper, MATLAB software programming method is used to calculate the eigenroot and eigenvector.

The software quality risk assessment system is divided into three levels of AHP decision-making problems, in which the objective level only contains one factor of software quality risk, the criterion level contains one factor, and the scheme level has one factor. The weight vector of the criterion layer to the target layer is expressed as:

$$W^{(2)} = (W_1^{(2)}, W_2^{(2)}, \ldots, W_n^{(2)})^T \tag{2}$$

The weight vector of the scheme layer to the target layer is expressed as:

$$W_k^{(3)} = (W_{k1}^{(3)}, W_{k2}^{(3)}, \ldots, W_{kn}^{(3)})^T \qquad k = 1, 2, \ldots, n \tag{3}$$

Then the column vector forms the combination matrix:

$$Y^{(3)} = (W_1^{(3)}, W_2^{(3)}, \ldots, W_n^{(3)})^T \tag{4}$$

The combination weight vector of the scheme layer to the target layer can be expressed as:

$$W^{(3)} = Y^{(3)} W^{(3)} \tag{5}$$

The second is to check whether the judgment matrix conforms to the consistency standard, that is, whether the weight coefficient distribution is reasonable. For the pairwise comparison matrix *A*, the ratio of its consistency index *CI* to the random consistency index of the same order (the same to *n*) is called the consistency ratio *CR*.

If $CR = \frac{CI}{RI} < 0.1$ , the degree of inconsistency is considered to be within the acceptable range.

The consistency index calculation formula is defined as:

$$CI = \frac{\lambda_{\max} - n}{n-1} \tag{6}$$

Among them, it is the ideal state, which is the consistency matrix; the larger the value of, the more serious the degree of inconsistency. Generally, the judgment matrix constructed above is not a consistent matrix. In order to make the software quality risk assessment results more accurate, the degree of inconsistency should be limited within the allowable range.

In order to make the degree of inconsistency within the allowable range, it is necessary to have a specific standard to measure the inconsistency index of the matrix. In addition, the random consistency index is introduced into the chromatography method. The random consistency index values are shown in Table 2.[7]

TABLE II. VALUES OF RANDOM CONSISTENCY INDEX *RI*

| ***N*** | 1,2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
|---|---|---|---|---|---|---|---|---|---|---|
| ***RI*** | 0 | 0.58 | 0.90 | 1.12 | 1.24 | 1.32 | 1.41 | 1.45 | 1.49 | 1.51 |
| Note: *RI* = 0 when *n* = 1 and 2 in the table, because the positive reciprocal matrix of order 1 and 2 is always the same matrix. | | | | | | | | | | |

If the test results meet the conditions, the degree of inconsistency is considered to be within the allowable range, and its eigenvector can be used as the weight vector.Otherwise, it is necessary to compare again and adjust the elements of the judgment matrix until the conditions are met.

## III. INFORMATION SYSTEM SOFTWARE QUALITY RISK ASSESSMENT

The research object of this paper is the information system software of weapon equipment, which usually belongs to the military A-level or B-level software (software key level standard). The software quality and safety are particularly important. The characteristics of the software project development process, such as hardware dependence, function uncertainty and development process iteration, increase the software quality risk.According to the data statistics of software quality problems in an enterprise's information system development process, the risk model and influencing factors of information system software development are established. The first level risk factors are demand risk, technology risk, process risk and management risk. The contents and causes of various software development risk factors are as follows.

## A. Demand risk

The requirement risk of information system software is mainly divided into three aspects: military requirement change, development plan change and system function change.Information system software is an important part of realizing weapon system. Its development requirements depend on the military requirements of weapon system, and are closely related to the military requirements of the weapon in the system combat system, such as function positioning, combat process, combat mode, etc.The development plan and functional requirements of information system software directly affect the coding cycle of software.The above-mentioned variety of demand changes form the risk of information system software development.

## B. Technology risk

The technical risks of information system software are mainly divided into system scale and complexity, software code reuse rate, software modularization degree, system key technologies, etc.Under the trend of system of systems operation, the function of weapon system is more and more, the scale and complexity of information system software is increasing rapidly, which can not effectively realize software modularization; at the same time, due to the different design of various weapon systems, the actual reuse rate of software code is low; the research and development process rework caused by the exploration and iteration of system key technology.

## C. Process risk

The process risk of information system software mainly includes system software and hardware coupling research and development, personnel flow and coding specification, project developer configuration, development cycle and team collaboration efficiency.In the process of information system software development, due to the dependence of software on hardware platform, the progress of software development is often constrained by hardware; the development cycle of weapon system is long, and the flow of personnel in the same post is inevitable, and the unification of coding specification can minimize the risk brought by the flow of personnel; the allocation of project developers and the development cycle of software development team are different. The process risk of information system software is an important factor of software quality risk.

## D. Management risk

The process risk of information system software mainly includes the rationality of plan arrangement, plan implementation, coordination efficiency among departments, resource allocation and other risk factors.In the early stage of weapon system development, the demonstration of schedule plan is insufficient, and the schedule is divorced from the actual situation, so it needs to be adjusted in time; the information system software is usually multi person, multi department cooperation, and there is competition and prevarication.

# IV. SOFTWARE DEVELOPMENT RISK ASSESSMENT PROCESS

Taking the development of an information system software as an example, a reasonable software quality risk assessment model is established, and AHP method is used to analyze the weight of quality risk factors and effectively identify risks.

## A. Establish risk assessment model

Based on the previous theory and experience in the development of information system software, a risk assessment model is established, including four primary indicators and multiple secondary indicators, as shown in Figure 2.

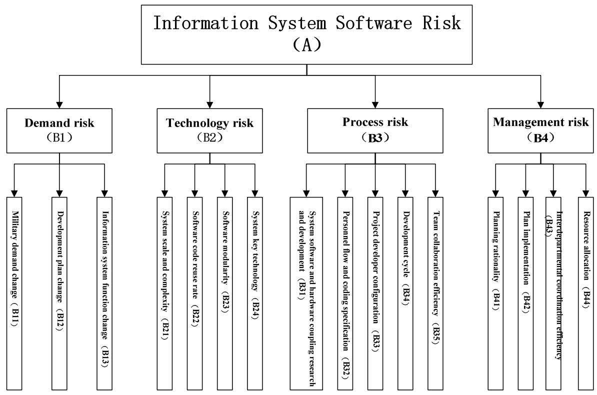



Fig. 2. software quality risk assessment model

## B. Construct the judgment matrix of risk factors

According to the actual situation of the development of information system software and the requirements for risk analysis of equipment development[8], the appraisers establish the risk factor judgment matrix according to the above rules.

The following will take the data of a group of risk factor judgment matrix as an example to evaluate the risk of software quality development. The first level risk factor judgment matrix is:

$$\mathbf{A} = \begin{bmatrix} 1 & 5 & 1 & 2 \\ \frac{1}{5} & 1 & \frac{1}{5} & 1 \\ 1 & 5 & 1 & 7 \\ \frac{1}{2} & 1 & \frac{1}{7} & 1 \end{bmatrix}$$

The second level risk factor judgment matrix is as follows:

$$\mathbf{B_1} = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} \\ 2 & 1 & 1 \\ 3 & 1 & 1 \end{bmatrix} \qquad \mathbf{B_2} = \begin{bmatrix} 1 & 5 & 5 & 2 \\ \frac{1}{5} & 1 & \frac{1}{2} & 1 \\ \frac{1}{5} & 2 & 1 & \frac{1}{2} \\ \frac{1}{2} & 1 & 2 & 1 \end{bmatrix}$$

$$\mathbf{B_3}=\begin{bmatrix}1 & \frac{1}{5} & \frac{1}{3} & \frac{1}{3} & 1\\ 5 & 1 & 4 & 3 & 4\\ 3 & \frac{1}{4} & 1 & 2 & 2\\ 3 & \frac{1}{3} & \frac{1}{2} & 1 & 2\\ 1 & \frac{1}{4} & \frac{1}{2} & \frac{1}{2} & 1\end{bmatrix} \quad \mathbf{B_4}=\begin{bmatrix}1 & 2 & 5 & 2\\ \frac{1}{2} & 1 & 5 & 3\\ \frac{1}{5} & \frac{1}{5} & 1 & 1\\ \frac{1}{2} & \frac{1}{3} & 1 & 1\end{bmatrix}$$

*C. Weight calculation and consistency verification*

The technical risks of information system software are mainly divided into system scale and complexity, software code reuse rate, software modularization degree, system key technologies,etc. Under the trend of system of systems operation, the function of weapon system is more and more, the scale and complexity of information system software is increasing rapidly, which can not effectively realize software modularization; at the same time, due to the different design of various weapon systems, the actual reuse rate of software code is low; the research and development process rework caused by the exploration and iteration of system key technology.

According to the above judgment matrix, calculate the weights of risk factors at all levels, then calculate the main characteristic root and weight vector of the matrix consistency check matrix, and judge the rationality of the risk factor evaluation matrix developed by the software (see Table 3 for the results).If the consistency test conditions are not met, the corresponding judgment matrix will be adjusted.

Table 3 weight vector and verification calculation results

| Judgement matrix | Weight vector | consistency check |
|---|---|---|
| Primary risk factor A | w0= (0.3455,0.0838,0.4684,0.1023) | CR=CI/RI=0.0599<0.1 |
| Secondary risk factor B1 | w1= (0.1692,0.3874,0.4434) | CR=CI/RI=0.0158<0.1 |
| Secondary risk factor B2 | w2= (0.5331,0.1196,0.1399,0.2074) | CR=CI/RI=0.0783<0.1 |
| Secondary risk factor B3 | w3= (0.0731,0.4795,0.2002,0.1579,0.0893) | CR=CI/RI=0.0369<0.1 |
| Secondary risk factor B4 | w4= (0.4389,0.3374,0.0907,0.1330) | CR=CI/RI=0.0657<0.1 |

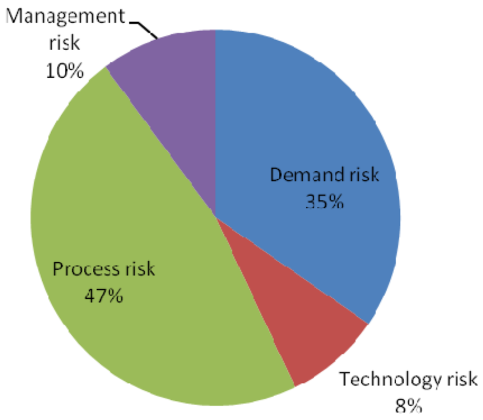
At the same time, according to the above formula, calculate the total order of the weight of the secondary quality risk factors to the total objective, and test the consistency, CR = 0.0387 < 0.1, which shows that the secondary quality risk factors have good consistency to the total objective.See Table 4, figure 3 and Figure 4 for the distribution of risk factor weights.

Table 4 weight distribution results of risk factors

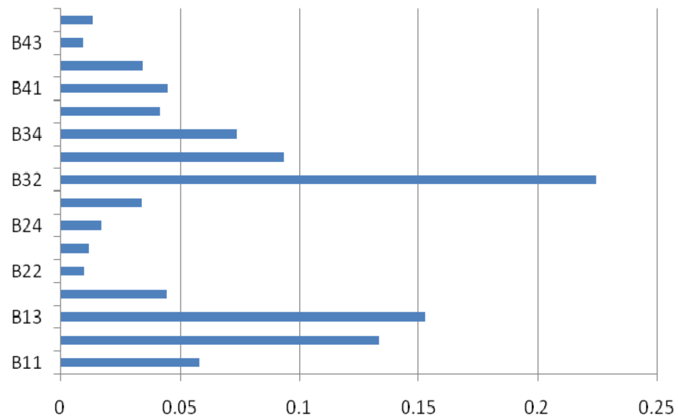
| Target layer | Criterion level | Scheme layer | Primary risk factors weight | Secondary risk factors Total weight |
|---|---|---|---|---|
| Software quality risk assessment | Demand risk | Military demand change B11 | 0.3455 | 0.05845 |
| | | Development plan change B12 | | 0.13384 |
| | | Information system function change B13 | | 0.15319 |
| | Technology risk | System scale and complexity B21 | 0.0838 | 0.04467 |
| | | Software code reuse rate B22 | | 0.01002 |
| | | Software modularity B23 | | 0.01172 |
| | | System key technology B24 | | 0.01738 |
| | Process risk | System software and hardware coupling research and development B31 | 0.4684 | 0.03424 |
| | | Personnel flow and coding specification B32 | | 0.22460 |
| | | Project developer configuration B33 | | 0.09377 |
| | | Development cycle B34 | | 0.07396 |
| | | Team collaboration efficiency B35 | | 0.04183 |
| | Management risk | Planning rationality B41 | 0.1023 | 0.04499 |
| | | Plan implementation B42 | | 0.03452 |
| | | Interdepartmental coordination efficiency B43 | | 0.00928 |
| | | Resource allocation B44 | | 0.01361 |

## V. ANALYSIS OF EVALUATION RESULTS

According to the results of software quality risk assessment, the first level risk factors are mainly from process risk and demand risk, accounting for 47% and 35% of the total risk respectively; the second level risk factors are mainly from personnel flow and coding specification, information system function change, development plan change, project developer configuration, etc.Therefore, in order to improve the quality of information system software, reduce the risk of software development, and reduce the risk of software development, we should focus on reducing the risk of demand and process, reducing the flow of personnel, and reducing the function change of information system software.




A: weight distribution of primary risk factors




B: weight distribution of secondary risk factors

Fig. 3. Weight Distribution

## VI. CONCLUDING REMARKS

Based on the AHP, this paper studies the software quality risk of information system, constructs the index system of software quality risk assessment, and divides the information system software quality risk factors in the development process into demand risk, technology risk, process risk and management risk. The results show that the process risk factor is an important source of software quality risk in information system. In order to improve the quality of information system software and reduce the risk of software quality, we should focus on the change of demand and development process, and reduce the risk of personnel flow and software function change.

## REFERENCES

[1] Cai Mingchun, LV Shoukun. Intelligent war form and its supporting technology system [J]. National defense science and technology, 2017, 2:94-98

[2] Hu Xiaofeng. War Engineering Theory: War methodology towards the information age [M]. Beijing: National Defense University Press, 2012

[3] ANTHONY D. C4ISR, the FINC methodology, and operations in urban terrain[J]. Journal of Battlefield Technology,2005,8(1): 1440-5113.

[4] Zhang Weiming, Yang Guoli, Zhu Cheng, et al. Research on modeling, game and evolution of network information system [J]. Journal of command and control, 2016, 2 (4): 265-271

[5] Ye Yicheng, Ke Lihua, Huang Daoyu. Systematic comprehensive evaluation technology and application [M]. Beijing: Metallurgical Industry Press, 2006

[6] Hu Xiaoran. Quantitative research on software quality based on AHP [J]. Computer application and technology, 2013 (11): 138-141

[7] Yan Wei, Chen Changhuai, Chen Yan. Study on critical value of consistency index of AHP [J]. Mathematical statistics and management, 2011,30 (3): 414-423

[8] China aerospace standardization research institute. Gjb5852-2006 equipment development risk analysis requirements [S]. Beijing: National Defense Science and Technology Industry Commission, 2007