The Missing Link – Risk Identification

David Hall

Hall Associates

159 Golightly Spring Rd
Toney, AL 35773

dave.hall@hallassociateshsval.com

Laurie Wiggins

CEO, Sysenex, Inc.

47766 Mariner Court
Sterling, VA 20165

wiggins@sysenex.com

Abstract. All risk management standards, guides and process descriptions note that risk identification is a key component of a robust risk management framework. Further, an effective risk identification process should identify all types of risks from all sources across the entire scope of the program/enterprise activities. However, no document or solution provides sufficient guidance for identifying a comprehensive set of risks – a risk management baseline. Further, risk identification as it is practiced today is a subjective, ad hoc, non-comprehensive and non-repeatable process resulting in continuing failures and overruns in all types of product and service development and modification programs.

A radical new approach to risk identification is presented to overcome this serious shortcoming. A large analysis of over 500 programs called the Risk Identification Analysis, its conclusions and the tool developed from those conclusions, Program Risk ID, are discussed in this paper.

The Risk Identification Analysis and Its Conclusions

A fundamental question prompted the performance of the Risk Identification Analysis. "Why is risk management the only Systems Engineering (SE) process that does not require a baseline to be developed?" All other SE processes - configuration management, requirements management, design, architecture, etc. - require that a baseline be developed for each program as one of the first steps after a program is established. This requirement does not exist for risk management. Risk management standards and guidelines [1,2,3,4,5,6,7,8,9,10,11] do indicate that risk identification is a very important step in the RM process, but do not require that a risk baseline be developed. We define risk to be potential problems that can affect program cost, schedule and/or performance. The term program also includes projects, activities and operations.

There is a widespread belief that each program has a unique set of risks. This is false, based on the results of our Risk Identification Analysis (herein known as the Analysis). It has been determined that every program inherently has the same risks as every other one. Of course, the *specifics* of the risks vary. For example, "Technology Risk" will change from program to program depending on what technology one uses or what skills personnel possess or require. These change as do the assessments and the impact(s) depending upon the specific program. But the overall baseline of technical, enterprise, operational, management, organizational and

external risks that should be considered remains the same. *There are no unknown* unknowns or Black Swans[12], only *unconsidered risks.* All risk management standards and guides fail in this regard: they do not require that all risk management processes/programs use the same basic set of risks for identification, thus creating a risk baseline for each program.

The risk management process has been fairly well standardized over the past 50 years – most risk management standards, handbooks and guidebooks (references as noted above) use the same basic process steps albeit with different names. A closer examination of the definitions shows that the steps are essentially the same – Planning, Identifying, Assessing, Prioritizing, Control and Monitoring in a continuous effort. When examining each of the steps however, one finds that Identification (establishing a risk baseline) is essentially ignored except to state that it must be done. Risk Identification is defined as

"…discovering, defining, describing, documenting and communicating risks before they become problems and adversely affect a project. Accurate and complete risk identification is vital for effective risk management. In order to manage risks effectively, they must first be identified. ***The important aspect of risk identification is to capture as many risks as possible.*** (author italics) During the risk identification process, all possible risks should be submitted. Not all risks will be acted upon. Once more details are known about each risk, the decision will be made by the project members as to the handling of each risk. There are various techniques that can be used for risk identification. Useful techniques include brainstorming methods as well as systematic inspections and process analysis. ***Regardless of the technique used, it is essential to include key functional area personnel to ensure no risks go undiscovered****."*[13] (author italics)

Common Historical Risks. The process steps for risk management, with the exception of risk identification, are outlined and defined. This lack of consideration in establishing a risk baseline is because of the mistaken belief that each program is unique and therefore its risks must also be unique. The Risk Identification Analysis has shown that this belief is wrong. As programs were analyzed, it became clear that the same risks kept occurring, over and over again. A set of common risks emerged. Thus there are a set of inherent risks that are applicable to any program and should be considered in developing a program risk baseline. These common risks provide a comprehensive method that can be used to develop a risk management baseline for all programs - based on this analysis of historical data.

The Analysis was conducted over 15 years by examining over 500 programs. Information from direct experience was utilized on over 70 programs, and the rest were researched using GAO project reports on DoD projects, and anecdotal evidence gained from teaching risk management to approximately 3500 people and train the trainer sessions provided to NASA Goddard personnel. Participants from these training programs shared confidential data during subsequent discussions that was not documented in papers or GAO reports. Programs (a blanket term that covers programs, projects, activities and operations) include those in the commercial and governments sectors, as well as those from numerous domains. Aerospace programs include those from all branches of the DoD and NASA. IT programs covered both hardware and software. Energy and utility programs were covered including facility construction.

Risk Weighting. As the risks were identified, it became apparent that some risks occurred more frequently than others. Also, when the risks occurred, some risks had a more detrimental effect on programs than others. In order to accurately gauge the effect of a given risk on a program, each risk needed to be weighted relative to the others. Once the risks were identified, the analytical hierarchy approach was used to perform the risk weighting.

Risk Levels. Once the inherent risks are in hand, how would one determine the status of the risk? Consider management experience, for example. How does one determine whether this risk has been addressed properly or not? If no standard is provided, the risk status is subjective, completely up to a given individual. In order to reduce subjectivity, risk levels were defined for each risk to define its current status in the solution space for that risk. These risk level statements are based on historical data for numerous programs and incorporate areas like the maturity of the process, the level of the design, the build level of the hardware, etc., for each risk. An example of a set of risk level statements is as follows:

Generic Risk: Requirements Definition

- Level 5 – System and user requirements are not defined, forcing the developer to make assumptions. There is no potential for definition of requirements for the long term.
- Level 4 – System and user requirements are not defined, forcing the developer to make assumptions. Assumptions are informally agreed to by the stakeholders or users. There is no potential for definition of requirements for the long term.
- Level 3 – System and user requirements are not defined, forcing the developer to make assumptions. Assumptions are informally agreed to by the stakeholders or users. Potential for definition of requirements in the short term exists.
- Level 2 – System and user requirements are partially defined: the remainder are to be defined in the short term and formally agreed to by all stakeholders.
- Level 1 – System and user requirements are fully defined and formally agreed to by all stakeholders.
- N/A – This risk is not applicable to the program being analyzed.

The risk levels provide the means of establishing program status for a given risk at a given time. The risk levels also provide a path to risk mitigation, and a guide to assigning likelihood of the occurrence of a risk. The risk levels minimize the subjectivity associated with risk status and allow the assignment of weighting factors to each risk, as well as the risk levels for each risk.

Program Complexity and Its Effect on Program Risk. From long experience performing risk management on programs, we see that programs with larger budgets, more people, and longer schedules are more complex, and thus are higher risk. However, there is no consensus in the literature on how to define program complexity, much less how to incorporate complexity into the risk profile of a program. A survey of sources that discuss definitions of complexity was consulted [14, 15,16,17,18,19,20,21]. We found that certain complexity factors caused the relative weighting of the risks to change, and these five parameters became the way that we describe program complexity: program cost, personnel effort, program duration, number of technologies/disciplines involved, and influencing factors. Influencing factors include conflicting organizational objectives, significant inter-organizational planning, building trust

requirements, and partner drag effects. Levels of program complexity was further defined by 5 levels that include Simple, Average, Moderate, Intermediate and High.

Table 1 illustrates the relationship between a risk, the risk weighting factors, the risk levels associated with a given risk, and the program complexity level.

Table 1. Complexity, Risk Level and Sample Risk Weighting Factors

Risk Level	**Program Complexity Level**				
	Simple	**Average**	**Moderate**	**Intermediate**	**High**
Level 5	16	18	20	22	23
Level 4	13	15	17	18	19
Level 3	8	11	12	14	16
Level 2	6	8	8	9	11
Level 1	4	4	5	5	7
N/A	3	3	3	3	3

Once the weighting factors for each risk and risk level were in place, they were combined to determine the overall risk level of the program (high, medium or low). By using the same risk baseline for each program, program risk levels and risks can be compared. Current risk management programs and methods do not allow a straight comparison.

The common risk set, with risk levels defined for each risk and combined with a set of complexity factors and levels, provides a comprehensive program risk baseline. The sum of these advances becomes a revolutionary approach to risk identification. The final innovation is to use this risk identification system as a diagnostic tool so that program vulnerabilities can be identified and addressed before their consequences are realized. That tool is Program Risk ID.

Risk Identification Today and Program Risk ID

Program failures, overruns (cost, schedule) and performance shortfalls are a recurring problem. This problem applies to both commercial and government programs and to small, medium and large programs. Some examples include the following.

- In March 2014, the US Government Accounting Office reported that the 72 major defense programs they reviewed that had reached the systems development stage were averaging 23 months delay in delivering initial capabilities. [22]
- A KPMG survey conducted in New Zealand in 2010 found that 70% of organizations surveyed had suffered at least one project failure in the prior 12 months. [23]
- A 2008 IBM study of over 1500 project leaders worldwide found that, on average, 41% of projects were considered successful in meeting project objectives within planned time, budget and quality constraints, compared to the remaining 59% of projects which missed at least one objective or failed entirely. [24]
- KPMG research conducted in 2013 showed that only one-third of the IT Project spend for any given organization is delivering the desired outcome. **[25]**
- A study covering 134 companies worldwide shows reports that 56% of firms have had to write off at least one IT project in the last year as a failure, with an average loss as a result of these failures being 12.5 Million Euros ($13.6M U.S.). **[26]**

Risk Identification Today. A 2012 risk management survey conducted by Sysenex, Inc. found that although 75% of companies surveyed had a risk management process in place, 51% of them had experienced a risk-related loss or failure. [27] If one has a risk management process in place and is using it, why is the loss and failure rate so high? In our experience, the primary causes are the current ad-hoc, non-repeatable, non-comprehensive approach to risk identification, the piecemeal approach to risk identification, and the 'Shoot the Messenger' syndrome.

The current ad-hoc, non-repeatable, non-comprehensive approach to risk ID. The better the risk identification process, the better the risk management process. If a risk is not identified, none of the other risk management steps are of any use. We have identified over 60 risk guides and requirements documents. Risk identification is addressed in numerous ways; a representative sample of risk identification approaches are provided in Table 2.

Table 2. A Representative Sample of Risk Identification Approaches

Risk Management Document	Brain-storming	Lessons Learned	Failure Scenarios /FMEA	WBS/ Work Plan	SMEs, Program personnel	Stake-holders	PRA
NASA SE Handbook, SP-2007-6105 [28]	X	X					
Risk Management Guide for DoD Acquisition [10]			X	X			
NASA SE Process and Reqmts. [29]			X	X	X		
NASA Risk Mgmt. Handbook [30]						X	
Engineers Australia Risk Management Guide [31]				X			
Human Rating Rqmts. NPR 8705.2 [32]			X				X
FFIEC Mgmt IT Exam. Handbook [33]			X	X			
NASA Gen'l Safety Program Rqmts [34]			X				X

When participants were asked during the Sysenex survey about how they identified risk, over 83% indicated that they relied on their personal experience, 74% consulted their subject matter experts or colleagues, 67% brainstormed with their colleagues, 55% conducted failure analyses, 50% consulted their stakeholders, and 41% performed Probabilistic Risk Assessments. The problem with these techniques is that they are unique to a given program or project: one starts over from scratch for every new effort.

Further, we have had many conversations with program personnel about risk identification. Despite the best efforts of these dedicated, experienced professionals, they are failing to resolve risks before they suffer the consequences. They know that it is better to find risks earlier rather than later. They also know that they are not uncovering all of their risks, and that they will likely have to deal with these problems later on when they are harder and more costly to fix.

In conjunction with George Mason University, Sysenex conducted a risk management tool survey. Although over 50 risk management tools are commercially available today, none of them provide a risk identification capability.

The piecemeal approach to risk ID. On programs, financial and business risk is often considered separately from technical risk. Having a partial understanding or visibility of program risks can lead to skewed decision-making. A good example of this phenomenon is the 2010 Gulf Oil Spill. The three companies involved, BP, Transocean and Haliburton, all performed risk analyses on their portion of the well system. None of the companies looked at the overall risk of the well system. The results were disastrous.

Shoot the Messenger. This occurs when program personnel that raise risks are blamed for the bad news as if they were responsible for creating the risk. The reasons for this reaction are numerous but are based on fear, denial and embarrassment. The resulting inhospitable and closed environment causes bad news to be suppressed until circumstances conspire to make the problem obvious to all.

In summary, risk identification today is an ad hoc, non-comprehensive, non-repeatable, subjective exercise. Risk guides and requirements only partially address these problems. There are no tools to assist personnel in their efforts, and personnel are mostly left to their own devices to do the best they can. Risk identification is often performed piecemeal on programs, leading to a fragmented or incomplete understanding of program risks, which can distort decision-making. Program personnel are sometimes discouraged from reporting risks.

This is why Program Risk ID was developed – www.programriskid.com .

Program Risk ID (PRID) is designed to be used by personnel (Users) who are knowledgeable about their program, allowing program personnel to perform this analysis for their programs. PRID provides risks found on a wide variety of past programs to help inform current development efforts. PRID provides the risk framework so that Users can assess their program for vulnerabilities – risks - that are addressed before they cause cost and schedule overruns and performance shortfalls. Users are typically the most knowledgeable about their

specific product or service development or modification program, and so are best positioned to perform the analysis.

The Risk Identification Analysis revealed 218 risks that fall into six broad areas: Technical, Operational, Organizational, Managerial, Enterprise, and External risks. For each risk area, PRID further subdivides the areas into categories by subject, with individual risks within the categories for ease of analysis and to assist in tool navigation. Examples of risk categories and individual risks are shown in Table 3.

Table 3. Examples of Risk Categories and Individual Risks

Risk Area	Risk Categories	Example Risks
Enterprise	Enterprise Approach, Processes, Security and Risk Approach	Experience, culture, reputation, security processes
External	Customer Focus, Funding, Labor, Regulatory and Legal, Threats, Environment	Customer interaction, country stability, threats
Management	Management Approach/Experience, Personnel Approach, Funding, cost and schedule, Management Processes, Measurement and Reporting	Program scope, management experience, staffing, personnel experience, turnover
Operational	System Maintenance, Security, Processes and Personnel, Failure Detection and Protection, Readiness, Impact on Company, User Considerations	Obsolescence, personnel training/experience, contingencies, human error, profitability, user acceptance
Organizational	Organizational Approach, Processes and Procedures, Security	Organizational experience, culture, personnel motivation, processes, data protection, security
Technical	System Definition/Integration, Common Technical Risks, Design, Software and Hardware Specific Risks, Processes, Production, Test, Reuse	Requirements, dependencies, quality, training, data quality, integration maturity, reliability, root cause analysis, fabrication, testing

PRID is intended to augment and enhance current risk identification efforts. Beginning as early as possible during a development or modification program, PRID analyses are ideally performed at periodic intervals throughout the program. As a program evolves, so does its risks, and PRID helps to identify new risks as they arise. Also, PRID will show the effectiveness of risk mitigation efforts as scores of individual risks fall, remain the same, or rise, from analysis to analysis.

A User sets up an analysis by inputting basic program information including program name, start and end dates, and the like. Users choose the program type: software only, hardware only, or both. Since PRID includes both hardware and software risks, and not all programs

have both, selecting the appropriate program type enables PRID to provide only those risks that are pertinent to a given program. Users answer the complexity questions, given in Table 4. Five ranges are provided for each answer: the range endpoints are given in Table 4.

Table 4. Complexity Questions and Answers

Complexity Question	Range of Answers
Program Duration – months	From 13 to 49 months+
Program Cost - dollars	From $1M to $100M+
Personnel Effort - days	From 2000 to 50,000 +
Technologies/Disciplines	From 1 to 5+
Influencing Factors	From 0 to 4

Based on the answers, PRID determines a program complexity level. The User can choose to agree with or to change the complexity level. While it is recommended that the User agree with the tool, there may be mitigating circumstances not accounted for by the tool that cause the User to adjust the complexity level. Once the setup information is input, PRID provides a screen for checking User input.

Once the analysis is set up, the User is presented with the six risk areas, and chooses one to begin the analysis. Each risk is presented the same way as shown in Figure 1. The Management Experience risk is shown with five risk levels and N/A, as described earlier.

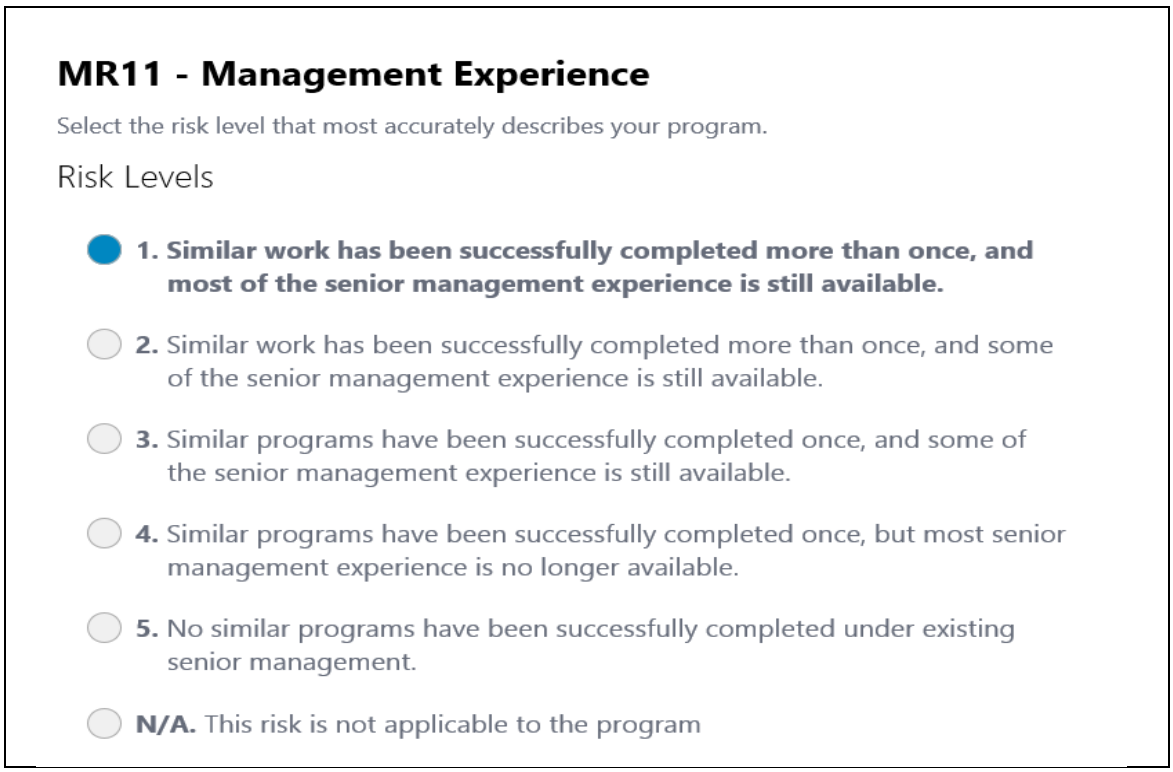

Figure 1. The Program Risk ID Management Experience Risk

If a program has experienced managers that have successfully completed a similar past program, then that program is low risk for the Management Experience item. Levels 3, 4, and

5 are more problematic, and additional effort is required to reduce the risk. It is recommended that all risks designated as Levels 3, 4 or 5 be examined further so that mitigation efforts can be undertaken, in alignment with program priorities and the availability of resources. A User chooses N/A if the judgment is made that the risk is not applicable to the program. We advise caution here as often, upon further examination, the risk is actually applicable, so N/A should be chosen rarely. PRID outputs scores at the program, risk area and individual risk area levels, facilitating progress tracking through time. Reporting capabilities include a list by risk level as well as numerical listing of risks by risk area. Reports are exported in a variety of formats to accommodate input to a wide variety of risk tools: MS Word, Excel, PDF, CSV, XML, MHTML and TIFF.

When two or more analyses for a given program are performed, PRID provides a trending capability so that previous analysis results can be compared with current analytical results so that risk mitigation efforts can be evaluated, and new identified risks can be addressed.

Info	Analysis1	Analysis2	Analysis3
Program Name	1330 - Bugle Program	1331 - Bugle Program	1332 - Bugle Program
Complexity *	Moderate	Moderate	Moderate
Last Saved	10/22/2015	10/22/2015	10/22/2015
Program Stage	Concept	Concept	Concept
Program Score	1936	1271	747
EXR	131	85	50
ORG	145	95	55
ER	158	107	64
MR	352	231	135
OR	334	218	128
TR	816	535	315

Risk Name	Analysis1	Analysis2	Analysis3
ER1 - Enterprise Experience	14	9	5
ER2 - Enterprise Lessons Learned Process	8	5	3
ER3 - Business/Mission Benefit	8	5	3
ER4 - Enterprise Culture	7	5	3
ER5 - Enterprise Contingency Planning	7	6	3
ER6 - Enterprise Management Processes	8	5	3
ER7 - Enterprise Financial Process	14	9	5
ER8 - Enterprise Critical Processes	8	5	3
ER9 - Enterprise Business Process Change	8	5	3
ER10 - Enterprise Interest in Personnel Motivation	7	7	7
ER11 - Enterprise Reputation	7	5	3
ER12 - Enterprise Risk Management Process	7	5	3
ER13 - Overall Enterprise Data Protection	14	9	5
ER14 - Overall Enterprise System Protection	14	9	5

Figure 2. An Example of a Program Risk ID Trending Report

Conclusions. We have presented in this paper a radical new approach to risk identification based on an analysis of over 500 programs, their risks and outcomes, called the Risk Identification Analysis. The conclusions of the study include the emergence of 218 common program risks, risk levels for each risk, the identification of pertinent program complexity parameters and their effect upon the program risk profile. These conclusions provide an antidote to the serious problems that plague risk management today: the lack of a baseline to assist programs in identifying risks, thus addressing the short-comings associated with the ad-hoc, non-comprehensive, non-objective and non-standardized approach currently taken towards risk identification today.

A software tool based on this analysis has been shown to be useful anywhere risk identification is performed today for product and service development and modification. This approach has been used on product and service development/modification programs for numerous domains including aerospace, IT, and energy. It has been used on both commercial and government programs, including proposal efforts. It has been used on programs with various development approaches including Waterfall, Agile, Rapid Application Development, and Component-Based Development. This risk identification approach can be used on one program or on a portfolio of programs to compare risks across them directly.

References

1. ISO (International Organization for Standardization). 2009. ISO 31000:2009 – Risk Management -- Principles and Guidelines. Geneva, CH: ISO.
2. ISO (International Organization for Standardization). 2009. ISO/IEC 31010:2009 - Risk Management - Risk Assessment Techniques. Geneva, CH: ISO.

3. ISO (International Organization for Standardization). 2009. ISO Guide 73:2009 - Risk Management – Vocabulary. Geneva, CH: ISO.
4. FERM (Federation of European Risk Management Associations). 2002. A Risk Management Standard. http://www.ferma.eu/risk-management/standards/risk-management-standard/
5. OCEG (Open Compliance & Ethics Group). 2009. "Red Book" 2.0:2009 - GRC Capability Model. http://thegrcbluebook.com/wp-content/uploads/2011/12/uploads_OCEG.RedBook2-BASIC.pdf
6. BSI (British Standards Institute). 2008. BS31100:2008 - Code of Practice for Risk Management. BSI.
7. COSO (Committee of Sponsoring Organizations of the Treadway Commission). 2004. COSO:2004 - Enterprise Risk Management - Integrated Framework. COSO.
8. SOLVENCY. 2012. SOLVENCY II:2012 - Risk Management for the Insurance Industry. Brussels: European Commission.
9. DoDI (Department of Defense Instruction). 2014. DoDI 8510.01:2014 - Risk Management Framework (RMF) for DoD Information Technology (IT), March 12, 2014. Office of the Secretary of Defense.
10. DoD. 2006. Risk Management Guide For DoD Acquisition, Version 6, August 2006. Office of the Secretary of Defense.
11. Hall, D. 2011. "Making Risk Assessments More Comparable and Repeatable" Paper presented at the International Committee for Systems Engineering International Symposium, Denver, Colorado. Vol. 14, No. 2, pp 173-179.
12. Taleb, N. 2007. *The Black Swan: Second Edition: The Impact of the Highly Improbable*. New York, US-NY: Random House.
13. Acquisition Community Connection. Long Description Risk Identification Introduction.
14. Ferreira, P. 2001. "Tracing Complexity Theory". (on-line notes for ESD.83 – Research Seminar in Engineering Systems at MIT). http://web.mit.edu/esd.83/www.notebook/ESD83-Complexity.doc
15. Dann, Z. and I. Barclay. 2006. "Complexity Theory and Knowledge Management Application." The Electronic Journal of Knowledge Management, Vol. 4, Issue 1, pp 11-20.
16. Lehman, M. and L. Belady. 1985. *Program Evolution: Processes of Software Change*. San Diego, US-CA: Academic Press Professional, Inc.
17. Henry, S. and D. Kafura. 1981. "Software Structure Metrics Based on Information Flow." IEEE Transactions on Software Engineering, Volume SE-7, Issue 5, pp 510 – 518.
18. Chidamber, S. and C. Kemerer. 1994. "A Metrics Suite for Object Oriented Design." IEEE Transactions on Software Engineering, Vol. 20, Issue 6, Jun 1994, pp 476 – 493.
19. Kearney, J., R. Sedlmeyer, W. Thompson, M. Gray, and M. Adler. 1986. "Software Complexity Measurement." *Communications of the ACM*, November 1986, Volume 29, Number 11.
20. Author: Magee, Christopher, M. and O. de Weck. 2004. "Complex System Classification." INCOSE (International Council On Systems Engineering), 2004-07-24.
21. Browning, T. 1998. "Sources of Schedule Risk in Complex Systems Development." Proceedings of the 8Th annual Symposium of INCOSE, July 1998.
22. Archstone Consulting. 2014. "Delays, Delays: A roadmap for improving performance across the Aerospace and Defense supply chain."

http://www.archstoneconsulting.com/industries/manufacturing/white-papers/delays-roadmap-for-improving-performance.jsp
23. KPMG. 2010. "NZ Project managements survey 2010." http://www.kpmg.com/NZ/en/IssuesAndInsights/ArticlesPublications/Pages/project-management-survey-2010.aspx
24. IBM. 2008, "Making Change Work." http://www.935.ibm.com/services/us/gbs/bus/pdf/gbe03100-usen-03-making-change-work.pdf
25. KPMG. 2013. "Project Management Survey Report 2013." https://www.kpmg.com/NZ/en/IssuesAndInsights/ArticlesPublications/Documents/KPMG-Project-Management-Survey-2013.pdf
26. Parker, D. and M. Craig. 2008. *Managing Projects, Managing People*. UK-London: Palgrave MacMillian. p 139.
27. Sysenex. 2013. "PRID Risk Management Survey Report, Executive Summary." https://programriskid.com/resources/
28. NASA. 2007. NASA/SP-2007-6105, Rev 1 - NASA Systems Engineering Handbook. December, 2007.
29. NASA. 2007. NPR 7123.1A - Systems Engineering Process and Requirements, App. C. March 26, 2007.
30. NASA. 2011. NASA/SP-2011-3422 – NASA Risk Mgmt. Handbook. December 21, 2011.
31. Institution of Engineers Australia. 2005. Engineers Australia Risk Management Strategies Guide. July, 2005
32. NASA. 2008. NPR 8705.2- Human-Rating Requirements for Space Systems. May 6, 2008.
33. FFIEC (Federal Financial Institutions Examination Council). 2004. FFIEC Management IT Examination Handbook. June, 2004.
34. NASA. 2008. NPR 8715.3 - NASA General Safety Program Requirements. March 12, 2008.

Biography

David Hall has over 40 years of systems engineering, risk management and program management experience in DOD, NASA, Department of Commerce, state and local agencies, numerous industrial and commercial companies. He has successfully completed all types of programs and activities, conducting both comprehensive analytical studies and implementing solutions to problems and risks. He is an INCOSE Expert Systems Engineering Professional (ESEP) and a Certified Information Systems Security Professional (CISSP). He has accomplished numerous risk management/systems engineering articles published in both report formats and peer-reviewed journals. The latest article was in the Journal of Systems Engineering.

Laurie Wiggins has 28 years of experience in systems engineering and business. She is a member of the INCOSE Risk Management and Critical Infrastructure Protection & Recovery Working Groups. Formerly with The Boeing Company, Ms. Wiggins gained experience on numerous successful development programs. As a consultant, Ms. Wiggins worked with firms in the aerospace, IT and energy industries. The resulting broad experience base

culminated in the vision for Program Risk ID. Ms. Wiggins led her company in the development of Program Risk ID.