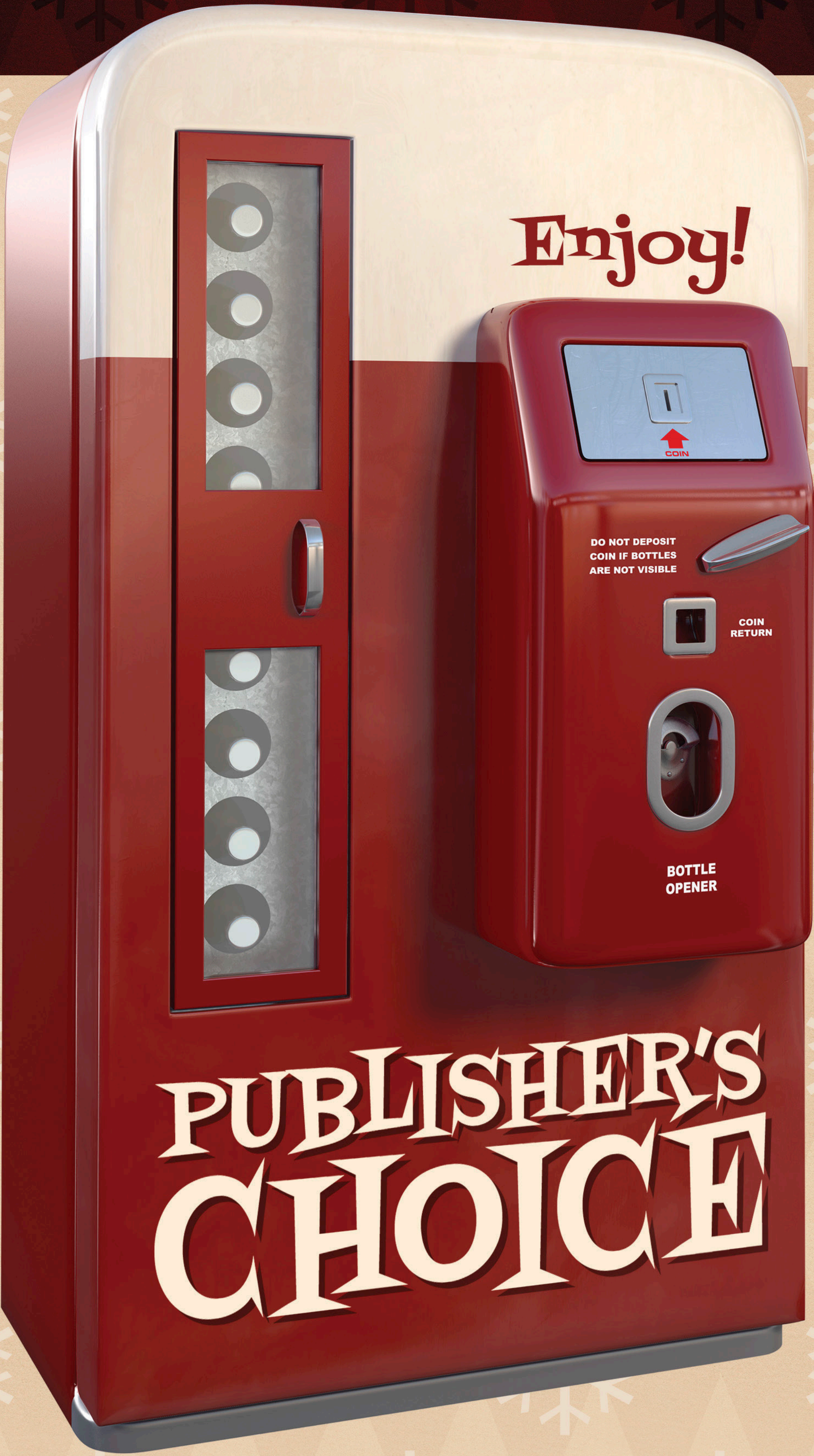
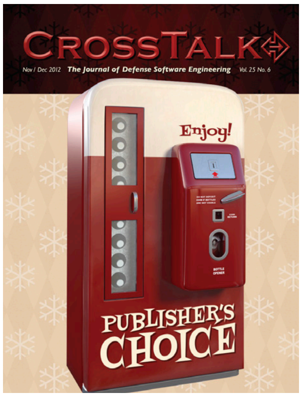
CrossTalk

Nov / Dec 2012 **The Journal of Defense Software Engineering** Vol. 25 No. 6



PUBLISHER'S CHOICE

Departments



CROSSTALK
Nov / Dec 2012 The Journal of Defense Software Engineering Vol. 25 No. 6
Enjoy!
PUBLISHER'S CHOICE

Cover Design by
Kent Bingham

Publisher's Choice

CrossTalk

NAVAIR Jeff Schwalb
DHS Joe Jarzombek
309 SMXG Karl Rogers

Publisher Justin T. Hill
Advisor Kasey Thompson
Article Coordinator Lynne Wade
Managing Director Tracy Stauder
Managing Editor Brandon Ellis
Associate Editor Colin Kelly
Art Director Kevin Kiernan

Phone 801-775-5555
E-mail stsc.customerservice@hill.af.mil
CrossTalk Online www.crosstalkonline.org

CrossTalk, The Journal of Defense Software Engineering is co-sponsored by the U.S. Navy (USN); U.S. Air Force (USAF); and the U.S. Department of Homeland Defense (DHS). USN co-sponsor: Naval Air Systems Command. USAF co-sponsor: Ogden-ALC 309 SMXG. DHS co-sponsor: National Cyber Security Division in the National Protection and Program Directorate.

The USAF Software Technology Support Center (STSC) is the publisher of CrossTalk providing both editorial oversight and technical review of the journal. CrossTalk's mission is to encourage the engineering development of software to improve the reliability, sustainability, and responsiveness of our warfighting capability.

Subscriptions: Visit <www.crosstalkonline.org/subscribe> to receive an e-mail notification when each new issue is published online or to subscribe to an RSS notification feed.

Article Submissions: We welcome articles of interest to the defense software community. Articles must be approved by the CrossTalk editorial board prior to publication. Please follow the Author Guidelines, available at <www.crosstalkonline.org/submission-guidelines>. CrossTalk does not pay for submissions. Published articles remain the property of the authors and may be submitted to other publications. Security agency releases, clearances, and public affairs office approvals are the sole responsibility of the authors and their organizations.

Reprints: Permission to reprint or post articles must be requested from the author or the copyright holder and coordinated with CrossTalk.

Trademarks and Endorsements: CrossTalk is an authorized publication for members of the DoD. Contents of CrossTalk are not necessarily the official views of, or endorsed by, the U.S. government, the DoD, the co-sponsors, or the STSC. All product names referenced in this issue are trademarks of their companies.

CrossTalk Online Services:
For questions or concerns about crosstalkonline.org web content or functionality contact the CrossTalk webmaster at 801-417-3000 or webmaster@luminpublishing.com.

Back Issues Available: Please phone or e-mail us to see if back issues are available free of charge.

CrossTalk is published six times a year by the U.S. Air Force STSC in concert with Lumin Publishing <luminpublishing.com>. ISSN 2160-1577 (print); ISSN 2160-1593 (online)

CrossTalk would like to thank NAVAIR for sponsoring this issue.

The Pros and Cons of Code Re-use

Software represents a significant investment for any organization and the quest to lower that upfront cost to field a capability and make use of that investment as technology evolves, delivery mechanisms change and requirements "morph" is an ongoing effort for all organizations.

Abstraction layers, containers and wrappers, middleware, and application frameworks are some of the approaches currently being used that allow the re-use of software. The interface and boundary layers in these systems and the degree to which the design is modular, not tightly coupled and uses non proprietary components and interfaces leads to the open architecture label and impacts total cost over the lifecycle.

The decision to re-use including COTS, GOTS and "open source" versus developing new software is an important consideration in the estimation and planning process as an efficient design that meets only your requirements can have a lower cost to integrate and maintain over a module or application that does multiple and potentially unknown functions or is "tightly coupled" with other software. With an efficient approach to the use of existing code it is worth considering if the code is modified versus re-used.

A common assumption for re-use is that the software is untouched at some module, app or aggregate level and the cost savings in estimation is based on the maturity level of the product being re-used. Many times modified code is called re-use or subtle distinctions are made that allow modified to be called re-use when less than five percentage change is made to the baseline code. It is difficult to realize the expected savings for re-use when the lineage of the code is untraceable and the artifacts that go with a module or app like requirements, unit test history and defect density either were never tracked at that level or are no longer relevant at least at the test level based on modification. Going forward all software developers should be considering the partitioning of functionality and interfaces in their designs that would allow efficient re-use. Minimizing modification and the overall size of the end product will lower the cost to integrate and maintain.

Defect density is not universally tracked and the necessary reliability of the end product is driven by the application. Safety, security, and reliability are all end product application requirements that should be addressed early in the development process. Reliability and functionality to protect against errors and failures is a driver for software cost based on more stringent requirements, different or potentially modified development processes, and increased test requirements. It is worth considering the inherited properties of the end product based on re-use and the introduction of schemes that can provide isolation and minimize risks. These system level design considerations are difficult to make after the design is complete and they need to be made early in development at the architecture level.

No matter what software development process you follow, waterfall, agile, etc., it is important to understand the requirements and interfaces associated with the modules, applications or aggregate software you are developing and integrating. When decisions of make-or-buy and re-use including are made, those interface and requirements are fixed and cost will be a factor if changes are necessary.

The prospects for better-faster-cheaper products as we evolve to new delivery environments and mechanisms is exciting, but on our journey to develop "open" architectures that will allow us to re-use the investments from many programs we cannot lose sight of the actual product we are reusing as it is this product at the lowest level that will be the source of savings or inherited lifecycle costs.

G. A. Graton

Gary Graton
SW Engineering Manager
NAVAIR SW Engineering Division

Schedule Adherence and Rework

Walt Lipke, PMI Oklahoma City Chapter

Abstract. When project performance is such that the product is delivered with expected functionality at the time and price agreed between the customer and supplier, it is deemed "successful." The rework, encumbering any project, has a measurable impact on whether a project can achieve success. The project manager (PM), who exercises control of the contributors to rework, can greatly enhance the prospect of delivering the product within its constraints. A significant portion of rework is caused by deviating from the project plan and its associated schedule. The measure of schedule adherence is derived from applying Earned Schedule (ES) to Earned Value Management (EVM) data. This paper first reviews the concept of schedule adherence and then develops an approach to understanding the cost impact from not adhering to the schedule. Finally, an index is proposed which provides information to assist project control and to forecast the cost associated with imperfect schedule adherence.

Background

An extension to EVM, ES was introduced in the March 2003 issue of The Measurable News [1]. The purpose of ES was to overcome the anomalous behavior of the EVM schedule performance indicators by providing reliable time-based indicators.[1] After ES was initially verified [2] and, subsequently, extended to forecasting project duration [3], it was shown to have further application.

One unique quality of the ES measure is that it facilitates identifying the specific Planned Value (PV) that should have been accomplished for the reported Earned Value (EV). This characteristic was first explained and examined in the article, "Connecting Earned Value to the Schedule," published in the Winter 2004 issue of The Measurable News [4]. Subsequently, this extended capability of ES was more fully elaborated in the April, 2008 CrossTalk article, "Schedule Adherence: a useful measure for project management" [5].

Because the task specific PV is identifiable, comparisons can be made to the task EV reported. The differences in PV and EV for each task are utilized to isolate problems occurring in the execution of the project. When the difference, EV – PV, is negative, there is a possibility of a constraint or impediment preventing task progress. This information is extremely useful. Having these tasks identified, allows the PM to focus on investigating and relieving problems that are causing workarounds. Minimizing the impact of constraints and impediments, in turn, minimizes the extent of workarounds, thus maximizing execution in agreement with the schedule. The more execution agreement there is between actual accomplishment and the schedule, the greater the performance efficiency becomes—for both cost and schedule.

Along with the negative differences previously discussed, there are positive differences identified for specific tasks. The positive differences expose areas where rework may occur. There are many causes of rework:

- **Poor planning stemming from requirements misinterpretation, incorrect task sequencing, and poor estimation**
- **Defective work**
- **Poor requirements management**
- **Schedule compression during execution**
- **Over zealous quality assurance**

However, the rework identified when EV – PV is positive is none of the ones cited above. The rework for which we are concerned is solely caused by project execution not in the activity sequence prescribed by the schedule. Although out of sequence performance is only one of the six contributors to rework mentioned, it has a major impact. Out of sequence performance is pervasive in that it is not aligned with a single aspect or project event. Rather, it occurs dynamically and can involve any, and possibly all of the project team throughout the entire period of performance.

For readers who have some background in quality and process improvement activity, the discussion thus far may bring to mind the idea of process discipline. The lack of process discipline leads to the creation of defects and inefficient performance. As has been described thus far, ES provides a way to identify and measure process performance discipline.

Schedule Adherence

Figure 1 provides a visual for discussing further the ideas from the previous section. The darkened tasks to the right of the vertical ES line indicate performance resulting from impediments and constraints or poor process discipline. Frequently, they are executed without complete information. The performers of these tasks must necessarily anticipate the inputs expected from the incomplete preceding tasks; this consumes time and effort and has no associated earned value. Because the anticipated inputs are very likely misrepresentations of the future reality, the work accomplished (EV accrued) for these tasks usually contains significant amounts of rework. Complicating the problem, the rework created for a specific task will not be recognized for a period of time. The eventual rework will not be apparent until all of the inputs to the task are known or its output is recognized to be incompatible with the requirements of a subsequent task.

This conceptual discussion leads to the measurement of schedule adherence. By determining the earned value (EV) for the actual tasks performed congruent with the project schedule, a measure can be created. The adherence to schedule characteristic, P, is described mathematically as a ratio:

$$P = \sum EV_k / \sum PV_k$$

PV_k represents the planned value for a task associated with ES. The subscript "k" denotes the identity of the tasks from the schedule that comprise the planned accomplishment. The sum of all PV_k is equal to the EV accrued during time duration at which an EV measurement is reported (AT). EV_k is the earned value for the "k" tasks, limited by the value attributed to the planned tasks, PV_k. Consequently, the value of P, or P-Factor, represents the proportion of the EV accrued which exactly matches the planned schedule.

A characteristic of the P-Factor is that its value must be between zero and one; by definition, it cannot exceed one. A second characteristic is that P will exactly equal 1.0 at project completion. P equal to zero indicates that the project accomplishment thus far is not, at all, in accordance with the planned schedule. In opposition, P equal to one indicates perfect conformance.

When the value for P is much less than 1.0, indicating poor schedule adherence, the PM has a strong indication the project will have rework at some point in the future. Conversely, when the value of P is very close to 1.0, the PM can feel confident the schedule is being followed and that milestones and interim products are being accomplished in the proper sequence. The PM thus has an indicator derived from ES that further enhances the description of project performance portrayed by EVM alone.

Derivation of Rework

The diagram shown in Figure 2 is provided to aid the derivation for computing rework. To understand how P can be used beyond its qualitative application, let us refresh the fundamental relationships to this point:

1. EV accrued = $\sum EV_i$ @ AT = $\sum PV_k$ @ ES
   subscript "i" identifies tasks that have earned value

2. EV earned in accordance with the schedule:
   EV(p)= $\sum EV_k$ @ AT = P ∑ EV (see note 2)

3. EV earned not according to the schedule:
   EV(r) = EV − EV(p) = (1 − P) • EV

These relationships provide a basis for examining the impact of rework and are extremely important to the remainder of this section of the paper.

To begin, we know from the earlier discussion of the P-Factor that a portion of EV(r) is unusable and requires rework. If the unusable portion can be determined, then the quantity of rework is calculable. Progressing on, the rework and usable fractions of EV(r) are defined as follows:

Rework fraction: f(r) = EV(-r) / EV(r)
Usable fraction: f(p) = EV(+r) / EV(r)

where EV(r) = EV(-r) + EV(+r)
and f(r) + f(p) = 1

Using the definitions, rework (R) can be computed from EV, P, and f(r):

R = EV(-r) = f(r) • EV(r) = f(r) • (1 − P) • EV

The quantities, EV and P, are obtainable from the reported status data. A method for determining f(r) is all that remains to have a calculation method for rework.

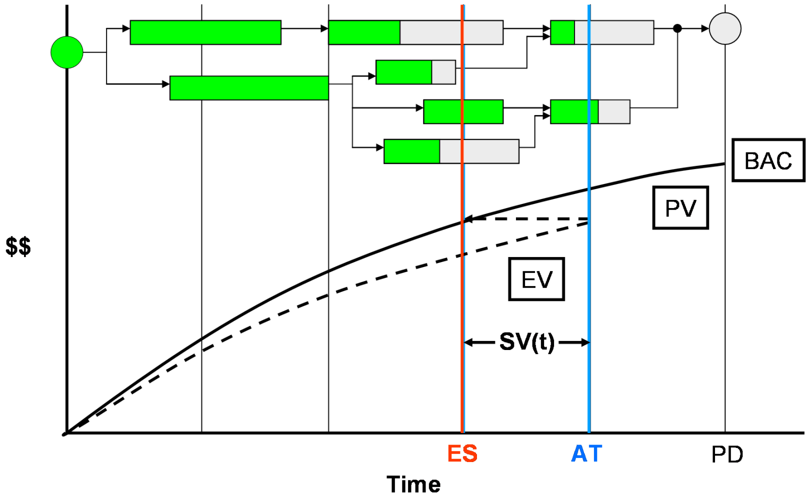


Figure 1. Actual Versus Planned Performance

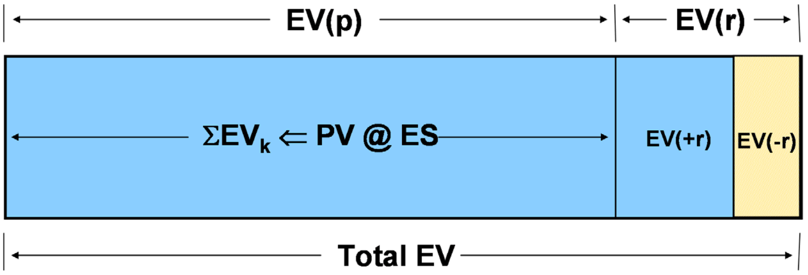


Figure 2. Rework Diagram

Logically, the project team's ability to correctly interpret the requirements for the work remaining increases as the project progresses toward completion. The end point conditions for this relationship are: f(r) = 1 when C = EV/BAC = 0 and f(r) = 0 when C= 1. Carrying this idea forward, the fraction of EV(r) forecast to require rework must then decrease as EV/BAC increases. It is further hypothesized that the rate of rework decrease for f(r) becomes larger and larger as the project nears completion.

The formula proposed which meets the conditions outlined is:

f(r) = 1 − C ^ n • e ^ (-m • (1 − C))

where C = fraction complete of project (EV/BAC)
e = natural number (base "e")
^ = signifies an exponent follows

The exponents, m and n, are used to adjust the shape of the f(r) curve. Presently, calculations of f(r) are recommended to be made using n = 1 and m = 0.5. These values for the exponents yield a nearly linear decreasing value for f(r) as fraction complete increases. It has been speculated that the behavior of f(r) should be more exaggerated; for example, a graph of f(r) versus EV/BAC having the general appearance of the perimeter of a circle in the first quadrant. The mathematical equation for f(r) is capable of generating this behavior as well as others. Further research is needed regarding the behavior of f(r) to substantiate use of the equation above and the recommended values for m and n.

Inserting m = 0.5 and n = 1 into the general equation for f(r), the equation for rework can be stated:

R = (1 − C • e ^ (-0.5 • (1 − C))) • (1 − P) • EV

Thus, in its final form, rework is a function of the EV accrued, the degree of schedule adherence (P), and the fraction complete (C or EV/BAC).

Computation Methods

The equation for R computes the amount of rework forecast to occur from the present status point to project completion due to the current measure of schedule adherence. It is an intriguing computation, but it is not a useful indicator for PMs. Recall that P increases as the project progresses and concludes at the value of 1.0 at completion, regardless of efforts by managers or workers to cause improvement. Thus, the computed value of R from one status point to the next cannot provide trend information concerning improvement and neither can it lead to a forecast of the total amount of rework caused by lack of schedule adherence.

At this point R appears to be a useless calculation. However, by recognizing that the rework value computed is distributed over the remainder of the project, it can be transformed easily to a useful indicator. It makes sense to normalize R to the work remaining; i.e., the project budget, less reserve, minus the planned value of work accomplished.[3]

The value of R divided by work remaining is the definition for the Schedule Adherence Index (SAI):

$$SAI = R / (BAC - EV)$$

The indicator is useful for detecting trends and is, therefore, an indicator by which a manager can gauge his or her actions taken. The interpretation of the indicator is straightforward. When SAI values increase with each successive status evaluation, Schedule Adherence (SA) is worsening. Conversely, when SAI decreases with time, SA is improving.

Having SAI provides the ability for calculating the rework created within a performance period along with the cumulative effects from imperfect SA. Additionally, it provides computational capability for forecasting the total rework from the lack of schedule adherence. Rework within a performance period is computed through a trapezoidal approximation technique, illustrated in Figure 3.

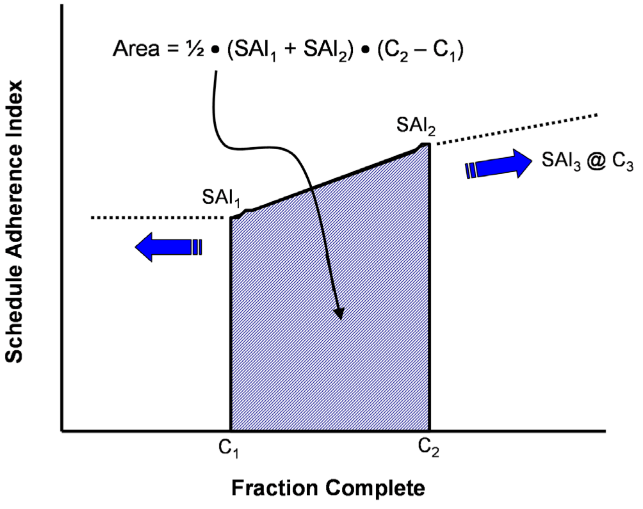


Figure 3. Area Calculation Method

For the graphical depiction, the area computed for each period is in terms of cost of rework per unit of budget. Thus, to obtain the rework cost for any period, the computed area is multiplied by Budget at Completion (BAC):

$$Rp(n) = BAC \cdot [½ \cdot (SAI_n + SAI_{n-1}) \cdot (C_n - C_{n-1})]$$

where n = the performance period of interest

The first and last index values, SAI_0 and SAI_N, are equal to 0.0.

With the methodology established for computing the cost of rework for any period, it becomes a trivial matter to calculate the cumulative cost. The cumulative accrual of rework (R_{cum}) generated from imperfect SA is the summation of the periodic values: $R_{cum} = \sum R_p(n)$.

The method for forecasting the total rework caused by performance deviations from the schedule is very similar to the formula used for forecasting final cost from EVM.[4] The formula for the Total Rework Forecast (R_{tot}) is

$$R_{tot} = R_{cum} + SAI \cdot (BAC - EV)$$

This formula makes possible, for each project status point, the computation of total rework forecast from imperfect schedule execution.

To clarify what R_{tot} represents, it is the forecast of actual cost for rework from imperfect execution of the schedule. From experience, rework cost is closely aligned with planned cost. It, generally, does not experience the execution inefficiencies incurred in the initial performance of the tasks.

Notional Data Example

The data provided in Table 1 is utilized to demonstrate the theory and calculation methods described in the previous sections of this paper. For our example, the schedule adherence shown by the values of P are very poor. P does not exceed 0.8 until status point 9, where the project is nearly 85% complete. Normally, P-Factor values are expected to be greater than 0.8 before 20% complete. Because the adherence to schedule is poor, we should expect rework to be large with respect to BAC.

The computed values for SAI and forecast rework are tabulated in Table 2. As observed, the value of SAI increases until the project is approximately 60% complete and then improves as the project moves toward completion. As discussed previously, the value of SAI for the final status period (11) is shown equal to 0.0.

The values for the rework forecast are observed to rapidly increase until the project achieves 30% complete. From that point, the values increase at a slower rate until the peak value of $60 is reached at 61% complete. Afterward the SAI values improve and the rework forecast decreases and concludes at $46. To a large degree the rework forecast is reasonably stable from 30% complete until completion.

Possibly a clearer understanding of the computed results can be obtained from viewing Figure 4. SAI is observed to be rapidly increasing from the beginning, indicating schedule adherence is worsening. Then, once the project has progressed past 60% complete, SAI dramatically improves. The forecast cost of rework, due to imperfect schedule adherence, likewise rapidly increases from a value of $13 at the first status point to the maximum value of $60. Although SAI greatly improves after its peak value, it is seen that the rework forecast improves only marginally. As the project moves toward completion, there is less and less of the project remaining upon which the SA improvements can have impact. Thus, the rework forecast is affected, but not to the extent of the change in SAI.

Real Data Example

The data in Table 3 is actual performance data from an in-work project, beginning at 22% through 84% complete. The BAC for the project is $2,488,202. As shown, the P-Factor is a high value initially, 0.930, and increases to 0.995 by 75% complete, and remains fairly constant for the status points that follow. The schedule adherence for this project is incredibly good. Not only is SA good, Cost Performance Index (CPI) and Schedule Performance Index-time (SPI(t)) are very good as well, 1.05 and 0.98, respectively.

Although only a single set of correlated data, the fact that all of the indexes have relatively high values demonstrates the conjecture that when SA is good, cost and schedule performance are maximized. If the conjecture is true, then the SA index is an important management indicator. The implication is the appropriate use of SAI as an additional management tool will increase the probability of having a successful project.

Table 4 contains the computed results for SAI and forecast of rework cost from imperfect schedule adherence. As expected for such high values of P, SAI is extremely low. The highest value is 0.028, while the lowest is 0.005. To have a sense of the distinction between poor SAI values and good ones, compare the values provided in Tables 2 and 4. The poor values of Table 2 are as much as 89 times greater than those shown in Table 4.

The average of the forecast rework cost for the real data example is slightly less than $42,000 or only 1.7% of BAC, a remarkably low number. The estimate of the standard deviation from the forecast values is $8,300. Utilizing the standard deviation, we can say it is extremely unlikely that the actual final rework cost will be greater than $67,000; i.e., $42,000 plus 3 standard deviations (3 x $8,300 = $24,900).

The graphs of SAI and the rework cost forecast are shown in Figure 5. The two plots are shaped similarly, both having negative trends. The graphs clearly show schedule adherence improving after the project is 40% complete. Assuming the improving trend continues, the rework cost at completion will be less than $40,000 or only 1.6% of BAC.

Summary

From the time of the introduction of the schedule adherence measure, P, there has been a desire to have the capability for understanding its implications; i.e., the cost of the induced rework. It was long thought that the complexity and difficulty of performing the necessary calculations would far outweigh the benefit from having the resultant information. However, as has been shown in this paper, the calculations are not that encumbering. Having the values for the P-Factor, the cost of rework can be forecast with relative ease. And thus, the importance of executing schedule, as intended, can be quantified by cost; i.e., the amount of waste caused by imperfect schedule performance.

Status Point	1	2	3	4	5	6
EV	$14	$37	$58	$82	$97	$113
P	0.082	0.208	0.247	0.337	0.371	0.431
Status Point	7	8	9	10	11	
EV	$125	$137	$157	$177	$185	
P	0.520	0.650	0.822	0.955	1.000	

Table 1. Notional Data

Status Point	1	2	3	4	5	6
Percent Complete	7.6%	20.0%	31.4%	44.3%	52.4%	61.1%
SA Index	0.072	0.171	0.267	0.351	0.407	0.444
Rework Forecast	$13	$29	$42	$51	$57	$60
Status Point	7	8	9	10	11	
Percent Complete	67.6%	74.1%	84.9%	95.7%	100.0%	
SA Index	0.425	0.350	0.213	0.064	0.000	
Rework Forecast	$59	$54	$49	$47	$46	

Table 2. Computed Values (Notional Data)

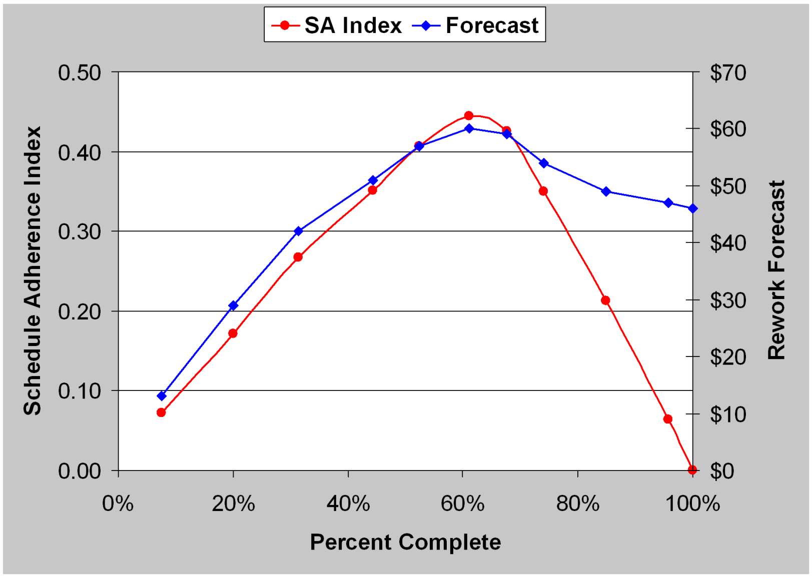
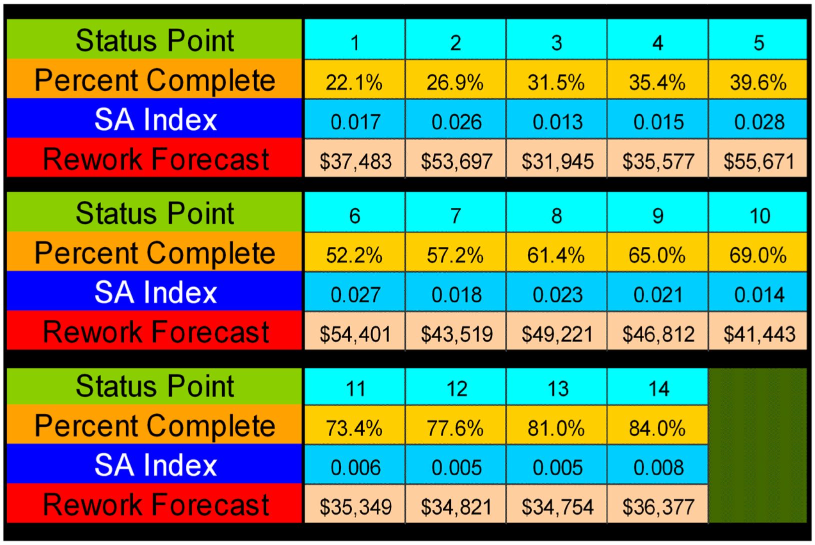


Figure 4. Rework Forecast (Notional Data)

Status Point	1	2	3	4	5
EV	$549,707	$668,776	$784,508	$881,288	$986,529
P	0.930	0.915	0.963	0.962	0.939
Status Point	6	7	8	9	10
EV	$1,299,880	$1,422,033	$1,526,842	$1,617,976	$1,716,130
P	0.957	0.975	0.970	0.975	0.984
Status Point	11	12	13	14	
EV	$1,826,991	$1,930,651	$2,015,852	$2,088,967	
P	0.994	0.995	0.996	0.993	

Table 3. Real Data



Status Point	1	2	3	4	5
Percent Complete	22.1%	26.9%	31.5%	35.4%	39.6%
SA Index	0.017	0.026	0.013	0.015	0.028
Rework Forecast	$37,483	$53,697	$31,945	$35,577	$55,671
Status Point	6	7	8	9	10
Percent Complete	52.2%	57.2%	61.4%	65.0%	69.0%
SA Index	0.027	0.018	0.023	0.021	0.014
Rework Forecast	$54,401	$43,519	$49,221	$46,812	$41,443
Status Point	11	12	13	14	
Percent Complete	73.4%	77.6%	81.0%	84.0%	
SA Index	0.006	0.005	0.005	0.008	
Rework Forecast	$35,349	$34,821	$34,754	$36,377	

Table 4. Computed Results (Real Data)

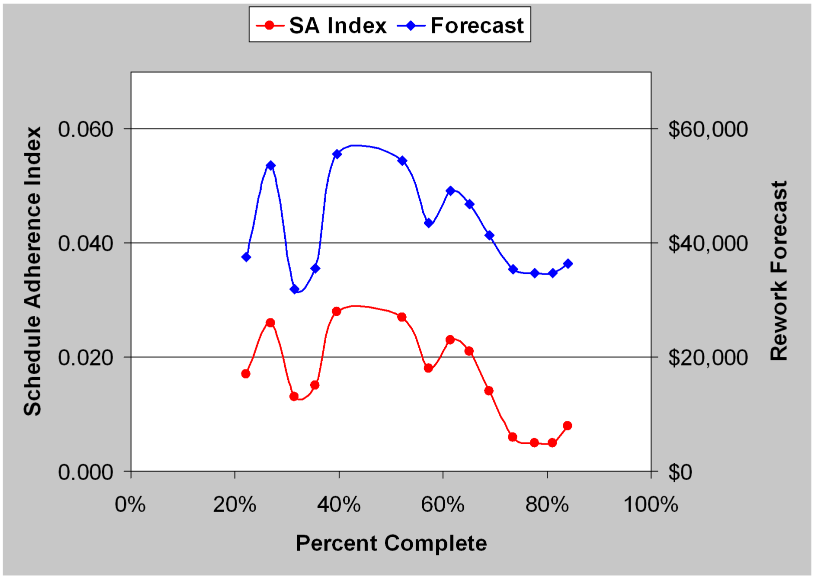


Figure 5. Rework Forecast (Real Data)

In this paper, the introduction of the SAI is shown to be integral to the forecast of rework cost. The approximation method for making the forecast calculation is diagrammed and discussed. The calculation methods are applied to both notional and real data to illustrate their application and simplicity.

The additional capability afforded by ES, to identify the impact of rework from poor schedule adherence, provides PMs an additional and valuable tool for guiding their project to successful completion.

Final Comment

To encourage application and uptake of the capability discussed in this paper, a calculator is made available for download from the calculators page of the earned schedule website, <http://www.earnedschedule.com/Calculator.shtml>. The calculator is titled, "SA Index and Rework Calculator." The calculator includes instructions and example data for trial use. ✧

ABOUT THE AUTHOR

Walt Lipke retired in 2005 as deputy chief of the Software Division at Tinker Air Force Base. He has more than 35 years of experience in the development, maintenance, and management of software for automated testing of avionics. During his tenure, the division achieved several software process improvement milestones, including the coveted SEI/IEEE award for Software Process Achievement.

Mr. Lipke has published several articles and presented at conferences, internationally, on the benefits of software process improvement and the application of earned value management and statistical methods to software projects. He is the creator of the technique Earned Schedule, which extracts schedule information from earned value data. Mr. Lipke is a graduate of the DoD course for Program Managers. He is a professional engineer with a master's degree in physics, and is a member of the physics honor society, Sigma Pi Sigma.

Lipke achieved distinguished academic honors with the selection to Phi Kappa Phi. During 2007 Mr. Lipke received the PMI Metrics Specific Interest Group Scholar Award. Also in 2007, he received the PMI Eric Jenett Award for Project Management Excellence for his leadership role and contribution to project management resulting from his creation of the Earned Schedule method. Mr. Lipke was selected for the 2010 Who's Who in the World.

E-mail: waltlipke@cox.net

REFERENCES

1. Lipke, Walt, "Schedule Is Different," The Measurable News, March & Summer 2003.
2. Henderson, Kym. "Earned Schedule: A Breakthrough Extension to Earned Value Theory?, A Retrospective Analysis of Real Project Data," The Measurable News, Summer 2003.
3. Henderson, Kym. "Further Developments In Earned Schedule," The Measurable News, Spring 2004.
4. Lipke, Walt, "Connecting Earned Value to the Schedule," The Measurable News, Winter 2004.
5. Lipke, Walt, "Schedule Adherence: a useful measure for project management," CrossTalk, April 2008
6. Project Management Institute, Practice Standard for Earned Value Management, Newtown Square, PA: PMI, 2005.

NOTES

1. The schedule performance indicators derived from Earned Schedule are Schedule Variance-time ($SV(t) = ES - AT$) and Schedule Performance Index-time ($SPI(t) = ES / AT$), where AT is the time duration at which an EV measurement is reported.
2. Recall that EV_k is limited by the value of PV_k.
3. In the terminology of EVM, the work remaining = BAC – EV, where BAC is Budget at Completion [6]
4. Final cost (IEAC) = AC + (BAC – EV) / CPI, where IEAC = Independent Estimate at Completion, AC = Actual Cost, and CPI = Cost Performance Index.

Structural Estimation Methodology for Estimating Enterprise Application Integration Projects

Manjula Natarajan, Infosys Ltd.

Abstract. Enterprise Application Integration (EAI) projects aim to integrate a host of application systems built on disparate technologies. The integrating solution should offer a platform to achieve interoperability seamlessly, thereby improving business process efficiency. This article summarizes the structural software sizing approach and its use for sizing and estimating EAI projects. The structural estimation methodology considers functional requirements as well as technical implementation styles of application integration projects.

Introduction

The worldwide Application Infrastructure and Middleware (AIM) space is ever-growing and according to Gartner, the AIM software revenue market totalled $19.3 billion in 2011[1].

In our attempts to measure the size of integration projects to precisely study productivity aspects, traditional software sizing models fell short of addressing the key requirements involved in such projects [2]. Extending traditional models to size integration projects [2] involves approximating the units/weights for every additional processing and systemic requirement that cannot be addressed directly by the base reference model. This also triggers constant validation of the units/weights additionally assigned, in comparison with the recommended size unit. For example, one extended Cosmic Function Point (CFP) should be validated with one standard CFP.

Integration projects are characterized by a host of factors such as the participating systems, the underlying technologies, data interfacing complexities together with the ability to prepare and present data, either synchronously or asynchronously for the participating applications, application of additional business processing logic, and so on, and essentially, making all of these possible at runtime.

While it is to an extent possible to size part of functional requirements in an EAI scenario in terms of data exchange requirements, the implementation aspects of data exchange and processing requirements ranging from i) direct product configurations to ii) extended custom logic, with different shades of these two implementation types, should not be overlooked.

Hence the associated challenges in sizing EAI applications are twofold, namely:

- To assess application complexity related to synchronous or asynchronous data exchange requirements and data processing requirements.
- To assess implementation characteristics, while being able to size the application upfront during requirements gathering stage.

Traditional estimation models fail to integrate the factors of software architectural requirements along with the functional (integration) requirements.

It therefore necessitated the development of a sizing model suited for estimating integration projects from all aspects of functional, technical and systemic requirements. The structural estimation methodology thus had its genesis to provide realistic estimation across the lifecycle stages of enterprise application integration projects.

The essence and the uniqueness of the model lies in its ability to capture the integration project complexities associated with the run-time data exchange and data processing requirements

The structural estimation approach, covering the functional, technical and systemic requirements is equally applicable and extensible to the following application areas demanding various implementation complexities:

- Business process integration and BPM-SOA
- System integration solutions
- Network-based integration services
- Product-based configure and build solutions
- Package implementations

The structural estimation model for EAI projects has been applied and validated on 10 application integration projects. The accuracy of estimation made in planning stages, was assured during design stages, and confirmed during closure. The model also enabled performance comparisons of these integration projects.

This article presents the model, examining the systematic approach for estimating EAI projects and covers the following:

- The integration application complexity comprising of run-time data exchange complexities, data processing complexities, and additional systemic complexities.
- The EAI sizing procedure following the software structural elements and the associated complexity factors.
- The approaches taken to validate the model and the derived business benefits.

The Model

The purpose of software sizing is to determine the cost of development and implementation. It addresses both the business functionality being implemented (what) and the technical implementation of the business functionality (how).

Backfiring methods are used by some organizations to bridge the gap between the functional requirements and technical implementation to derive the size. But these methods lack clarity in combining these aspects to derive the size units, and in the majority of cases, the methods do not consider the implementation approaches.

The structural estimation methodology addresses this gap between what functionality is to be built and how it is to be built, to derive the cost estimation. The model considers the software architectural layers as the focal point. In an EAI scenario, the software architectural layers include the host of integrating systems, and the EAI layer itself comprising of interfaces and data structures needed to integrate the external systems.

Complexities, Representation and Sizing

The structural estimation methodology covers the functional factors attributable to each such layer, and the associated implementation complexities. Size is derived from combin-

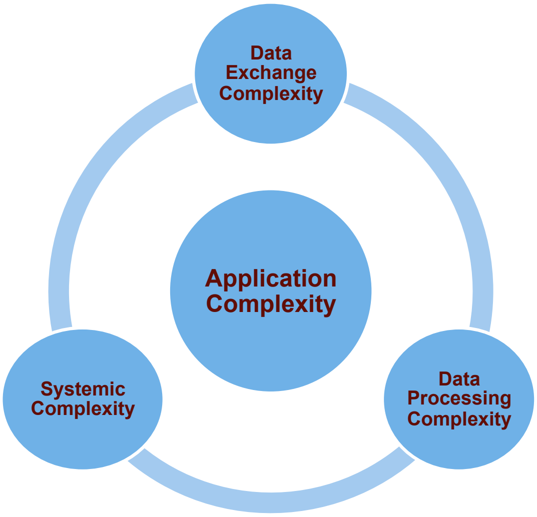


Figure 1. EAI Application Complexity Representation

ing the following requirement complexities: a) data exchange requirements, b) data processing requirements and c) additional systemic complexities. The application complexity categorized with these requirements is represented in figure 1.

The implementation complexities are then studied for each of these requirements and addressed with appropriate weights.

The steps involved in deriving the total EAI application size are represented in figure 2 and described below:

Step 1: List the Software Architectural Points

This includes the host of integrating systems whilst the EAI Layer that integrates the external sources has its own building blocks comprising interfaces, data structures, etc.

For an example, in an EAI project aiming to integrate four different systems using two different interfaces, the software architectural points include the four external systems as well as the two interfaces of the EAI layer.

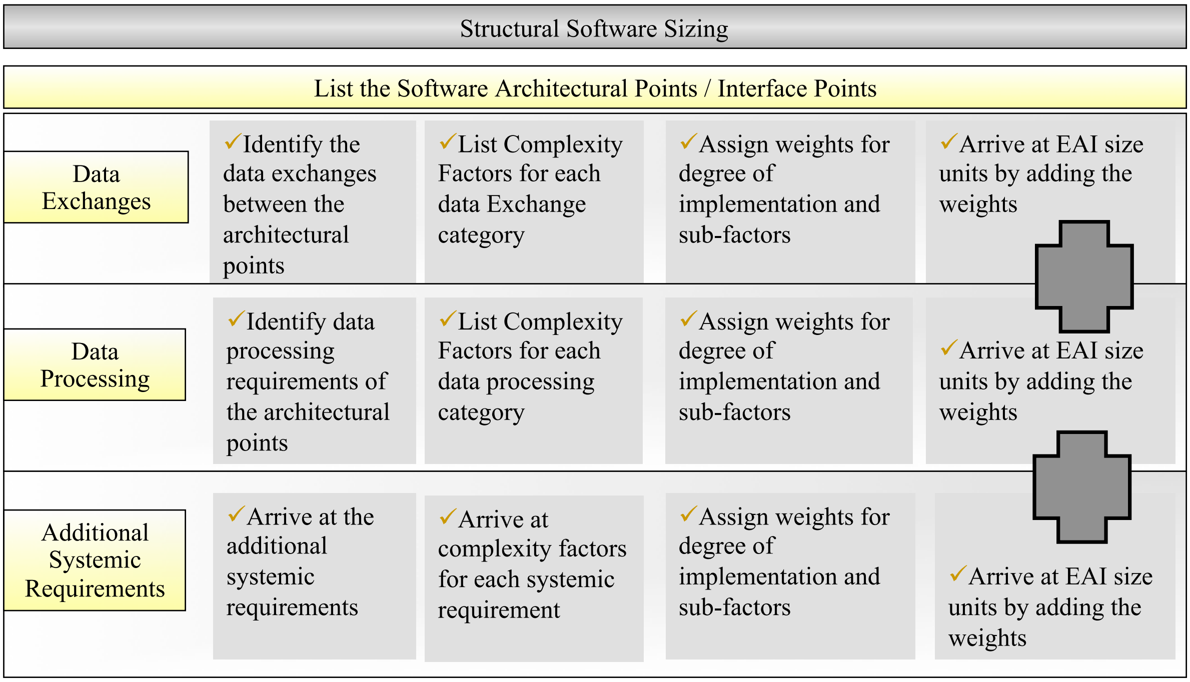
Step 2: Itemize the Data Exchange Requirements and Processing Requirements of Each Building Block

Each building block or the software architectural point will have an associated data exchange requirement—to feed-in data to or to subscribe to data from, other interfacing points. These form the data exchange requirements and typically include:

- Data received from external sources (internal storage)
- Data to be published to external sources (internal storage, logging requirements)

For logical collection and synchronous or asynchronous exchange of processed data, the following data processing requirements might apply: Data mapping, enrichment, transformation, extraction, encryption, decryption, synchronization, data validation, business processing, etc.

Consider a source application sending messages (data exchange) that are to be processed/enhanced (data processing) and transmitted (data exchange) to a subscribing application. Here the messages are received by the EAI interface, processed/



Structural Software Sizing

List the Software Architectural Points / Interface Points

Data Exchanges	✓Identify the data exchanges between the architectural points	✓List Complexity Factors for each data Exchange category	✓Assign weights for degree of implementation and sub-factors	✓Arrive at EAI size units by adding the weights
				✚
Data Processing	✓Identify data processing requirements of the architectural points	✓List Complexity Factors for each data processing category	✓Assign weights for degree of implementation and sub-factors	✓Arrive at EAI size units by adding the weights
				✚
Additional Systemic Requirements	✓Arrive at the additional systemic requirements	✓Arrive at complexity factors for each systemic requirement	✓Assign weights for degree of implementation and sub-factors	✓Arrive at EAI size units by adding the weights

Figure 2: Steps in sizing EAI Applications using Structural Estimation Methodology

Note: The weights are assigned based on a three-point scale, in alignment with the degree of technical implementation—fully configurable, partially configurable and manually constructed. Also the weights vary from one EAI package to another.

enhanced and sent by the interface to the subscribing application. This is a simple example of data exchange and data processing from the three software architecture points, namely the source application, the EAI interface and the destination point.

Step 3: Determine Complexity Factors Associated With the Two Category Heads (Data Exchanges and Data Processing)

For every data exchange requirement, the implementation complexity factors dealing with metadata preparation needed for the data exchange, data exchange/transport mechanism, data routing mechanism and conditions, will be determined.

For example, an EAI interface receives purchase order details from a legacy purchase order processing system. Here the complexity factors to be considered during the data reception shall include: preparation of schema or metadata to receive the order data, size of the metadata as number of data elements, configuration of data reception through appropriate adapter or end point channel, any extraction logic and data parsing conditions.

For data processing requirement, the complexity factors deal with ascertaining if the data processing will be done by directly configuring the integration product or by usage of COTS utilities or by custom logic or a combination of the above three techniques. This will ensure proper size assignment for simple to complex data processing conditions.

Step 4: Determine Additional Systemic Requirements and Associated Complexities Affecting the Overall Application Integration

Usually, the data-subscribing applications in a B2B scenario might require the EAI interface to pass data in an encrypted form. Additional requirements might include passing the data in chunks which will have to be received in a logical sequence and reassembled during execution.

The EAI layer should address such additional requirements which are neither direct data exchange requirements nor data processing requirements. The systemic requirements are those associated with the additional technical requirements applicable for the seamless integration.

Step 5: Assign Weights for Degree of Implementation for Each Factor and the Associated Sub-factors

Additional factors to be considered here include the degree of customization needed, which can be ascertained by the integration product in selection.

For each of the complexity factors considered from Step 3 and Step 4 above, the associated sub-factors need to be analyzed based on how the complexity factor is to be implemented using the integration product. For most of the processing and exchange requirements, the implementation may be facilitated using in-built product features, or by using COTS utilities, or through custom development. The degree of implementation will be studied for each complexity factor and weights assigned based on the nature of work involved.

Based on the above technique the size output is determined by adding up the individual size units, denoted as:

Total Build Size in package-specific "EAI points" =
(Data Exchange Size Units from the assigned weights in "EAI points"

+

Data Processing Size Units from the assigned weights in "EAI points"

+

Additional Systemic Complexities Size Units from the assigned weights in "EAI points").

Validation Approach

The assignment of weights as a unit was carefully made from the multiple iterations of the following steps:

- Determining degree of implementation of each complexities assigned across the three categories.
- Rank ordering the complexity assigned across the three categories.
- Assigning the unit and the weights in scale factors for each complexity.

The sizing model was then validated by applying 10 EAI projects with the integration scope covering the majority of the factors considered, and by using the following approaches.

- Rank ordering of projects based on computed size units and comparing the order with the projects' scope based on expert inputs. The direction of magnitude was confirmed.
- Plotting the size units against the effort consumed for build & unit testing. The observed R squared value was 0.97.
- Checking the size vs. effort relationship at the granular component/interface/complexity level. This was done to validate the approximation of a size unit evenly across the various complexity levels. This step also helped to under stand the consistency of an EAI size unit across EAI projects with varying complexity scope: data exchange rich integrations, data processing rich integrations and complex integrations covering data exchanges, data processing and other systemic requirements.

The methodology usability was verified by conducting a reproducibility exercise for one project with seven expert estimators to determine the size units. The experts were subjected to an initial orientation on the sizing exercise and the project scope for sizing. The insignificant variation in the size units confirmed the usability of the model.

Further, for improved usability and reproducibility, package-specific sizing tools with user interfaces have been created. Users need only to enter the integration requirements, and the tool automatically provides the computed EAI project size in package-specific EAI points. The following graph depicts the linear relationship between EAI points of EAI projects and the associated project build effort.

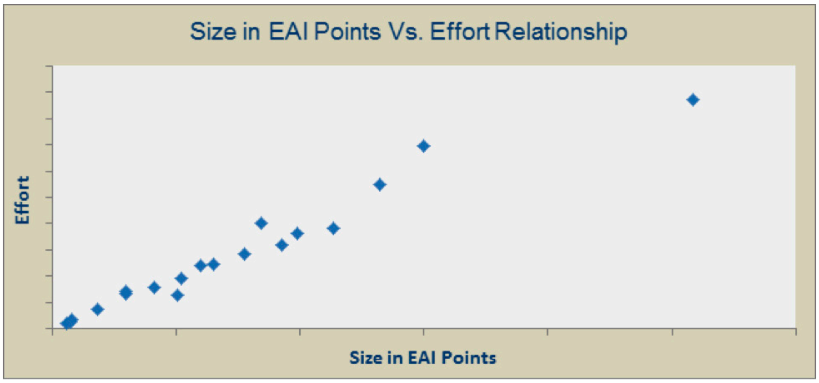


Figure 3. Size in EAI Points Vs. Effort Relationship

Effort Equation From Historical Data

For validating the size units, a linear relationship was established between the size units of each project and the respective build efforts from historical data. The total size units for each past project were listed against their respective build effort and the two variables were studied for linear relationships by plotting the size units against the build efforts. The observed R squared value was 0.97, showing greater linear relationship between the size units and the build efforts. The effort equation thus arrived is used for estimating the build effort for a given size.

Note: The effort equation is dependent on the historical data which largely reflects the standard organizational process capability and hence the baseline performance.

Once the performance baselines are thus established, the projects can effectively estimate the build effort and plan for improved performance leveraging the organizational process, project and risk management capabilities.

Business Benefits

This model is best suited for estimation at early lifecycle stages of EAI projects and provides the following business benefits:

- Improved accuracy in estimation leading to enhanced cost and schedule planning.
- Structural estimation leads to effective management of multi-vendor outsourcing/contractual projects at a possible logical level.
- Facilitates effective project staffing and execution models based on sizing at specific requirement levels.

Conclusion

The structural estimation approach:

- Adopts a scientific approach towards software estimation, covering the integration requirements and the multi-dimensional complexities of the actual building blocks.
- Provides improved accuracy in sizing, thus leading to proper effort and cost estimations.
- Allows effective management of costing and scheduling the work pieces by slicing and dicing and rolling-up the size units at any required level.
- This feature, allows an organization to effectively outsource different pieces of work, and aids in selecting appropriate project execution models and accurate staffing.

Future developments will include studying its fitment for all types of configure and build solutions. Extensions shall be made to derive appropriate adjustment factors for sizing maintenance work in all the applicable areas.

Acknowledgements:

The author acknowledges the contributions of Prakash R, Raghavendra Rao, Rekha Kodali, Vivek Shaurya, Santhosh Maruthaiyan, Senthil Kumar P.R, and Vinayakaraman S, towards applying the methodology across EAI package services. The author thanks the CrossTalk editorial board, its technical reviewers, Haresh Amre, Moses Raj and Anshuman Tiwari for their support and valuable feedback. ❖

ABOUT THE AUTHOR

Manjula Natarajan has completed her master's degree in computer applications and postgraduate program in management. Manjula has more than 16 years of experience in top Indian IT service organizations, spanning multiple roles including development, delivery management, program management, SEPG/QAG and IT process consulting.

She holds the title, Principal – Quality Programs, at Infosys Limited where she contributes towards establishing engineering processes and measurement frameworks, engineering productivity improvement solutions, business outcome and value-based prediction models for project services covering Oracle packages and EAI.

E-mail: Manjula_Natarajan@Infosys.com

REFERENCES

1. Gartner Press Release, "Gartner Says Worldwide Application Infrastructure and Middleware Market Revenue Grew 10 Percent in 2011", Gartner.com
2. Naveen Krishna, "EAI Estimation Challenges – Cosmic FFP Efficacy", Infosys.com

Maturing an Agency's Private Cloud Toward an Intelligence Community Online On-Demand Information Sharing Framework

Phong Ngo, SAIC
David Fado, SAIC
John Hoecker, SAIC

Abstract. Data sharing has become a given today, especially in cyberspace and social media. It is not entirely the case in the Intelligence Community (IC) due to security concerns and other architectural considerations, despite their quest for connecting the proverbial "dots."

This article will revisit several common data-sharing models and explore how the IC can take advantage of them, while taking security concerns and architectural differences into account. In other words, the discussion will focus on how IC members can mature individual stovepipe clouds into a community cloud where data will have a chance to become more widely sharable.

Section 1: Background/Introduction

In establishing the IC Common Operating Environment (COE), the Office of the Director of National Intelligence stated two major IC aims: (a) achieve IC savings through information technology efficiencies and (b) establish common IT architecture, but allow unique mission or specific capabilities. The ultimate objective is to share mission-relevant information efficiently and securely. Many initiatives have been taken to support the above aims and objectives across the community, such as IC Desktop Environment (DTE) and IC-Cloud. The Excel-Cylinder (EC) project supports these efforts with a data fusion platform that complies with Director of National Intelligence (DNI) standards and links virtual mission spaces into the wider COE.

Traditionally, IC information sharing has been achieved in many ways, such as through formal arrangements (e.g., liaison offices) or analysts "socializing" in mission-partnering situations. However, in modern warfare, including counter-terrorism, cyber operations and asymmetrical threats, "theaters" are dynamic and fluid. The elements of surprise and ingenuity, coupled with lethal force, are the main weapons of the bad guys. Therefore, the need for timely, online, on-demand information sharing beyond formal protocol or informal socializing is becoming more pressing than ever before. In this paper, we will use the EC project of a Special Access Required Agency (SARA) as an example, but this also may apply to other IC agencies' cloud initiatives. As a member of the much-anticipated IC-Cloud, EC is building a framework for more dynamic and timely information sharing with mission-relevant information.

In this paper, we will (a) revisit some major information-sharing models and their architectural implications; (b) review the current EC architecture against the objective of efficient and secure information sharing among IC partners; and (c) explore the next logical steps for maturing EC architecture toward achieving an information-sharing framework—a framework designed to provide optimal usability to users of partnering agencies.

Section 2: Information-sharing Models

Information sharing ranges from (a) a fully integrated environment to (b) a common operating environment (hardware, software, toolsets, and at times, shared domains) to (c) a loosely coupled (federated) environment where information is shared, in most cases, through web services.

A fully integrated environment is an ideal setting for information sharing. However, due to special security or operational considerations, the reality is that such an environment rarely exists even within a single agency. In most cases, it is unfeasible because of differences in partners' legacies, operating environments, cultures, and legal and budgetary concerns.

COE information sharing, on the other hand, is bound by specific interface protocols and aimed at supporting a number of missions. The EC program follows a Defense Intelligence Information Enterprise (DI2E) template for COE information sharing among mission partners and allies. IC-COE DTE is another example for this type of sharing. Though this model is effective when dealing with more stable conventional warfare, dependencies on prescribed hardware/software/applications suites can prevent the community from adopting more advanced technologies in a timely manner. This reduces overall mission effectiveness, especially when dealing with the dynamic nature of irregular warfare.

Lastly, a federated information-sharing model, such as Joint Worldwide Intelligence Communication System Open Search, is more flexible and dynamic. However, it lacks the richness of some tools. For example, due to certain technical and security concerns, analytics and exploitation tools usually available to each agency's domain may not be included.

Section 3: EC Current Architecture in the Context of Information Sharing via IC-Cloud

IC-Cloud is another initiative aimed at providing a richer information-sharing environment to the community by partnering members. It is to allow partners to see more proverbial "dots" from sources from partnering members. At the same time, it allows for unique mission- or agency-specific capabilities. In other words, "share all you can share, and keep what you must keep." EC Shared Cloud Machine (SCM) along with its Community Cloud Interface (CCI) facilitates sharing EC data with the rest of the IC (Figure 1) on a "need-to-share" basis. This is possible because EC shareable data is physically separated from native EC data. The same "need-to-share" principle applies to each

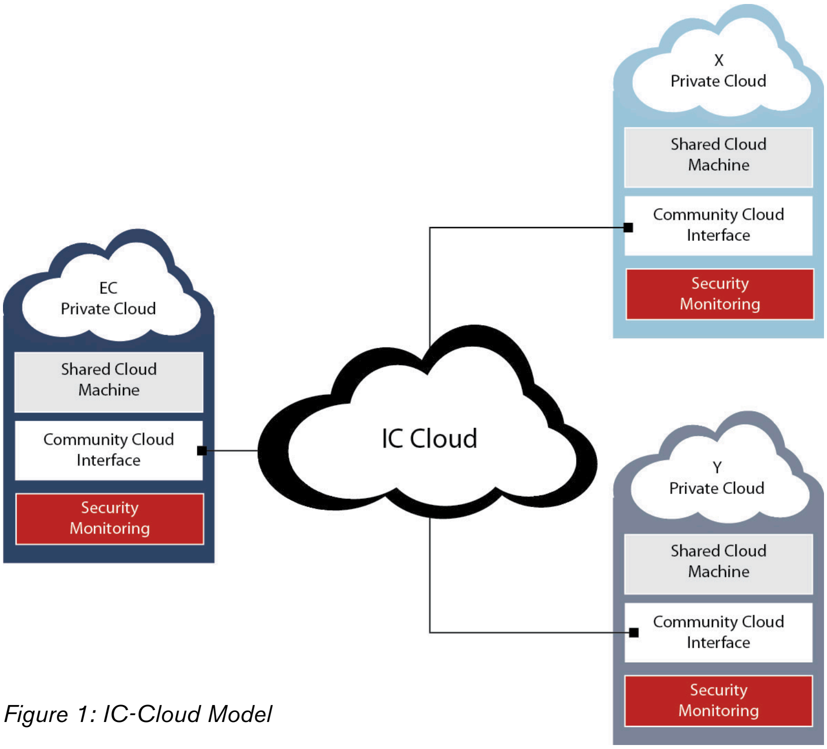


Figure 1: IC-Cloud Model

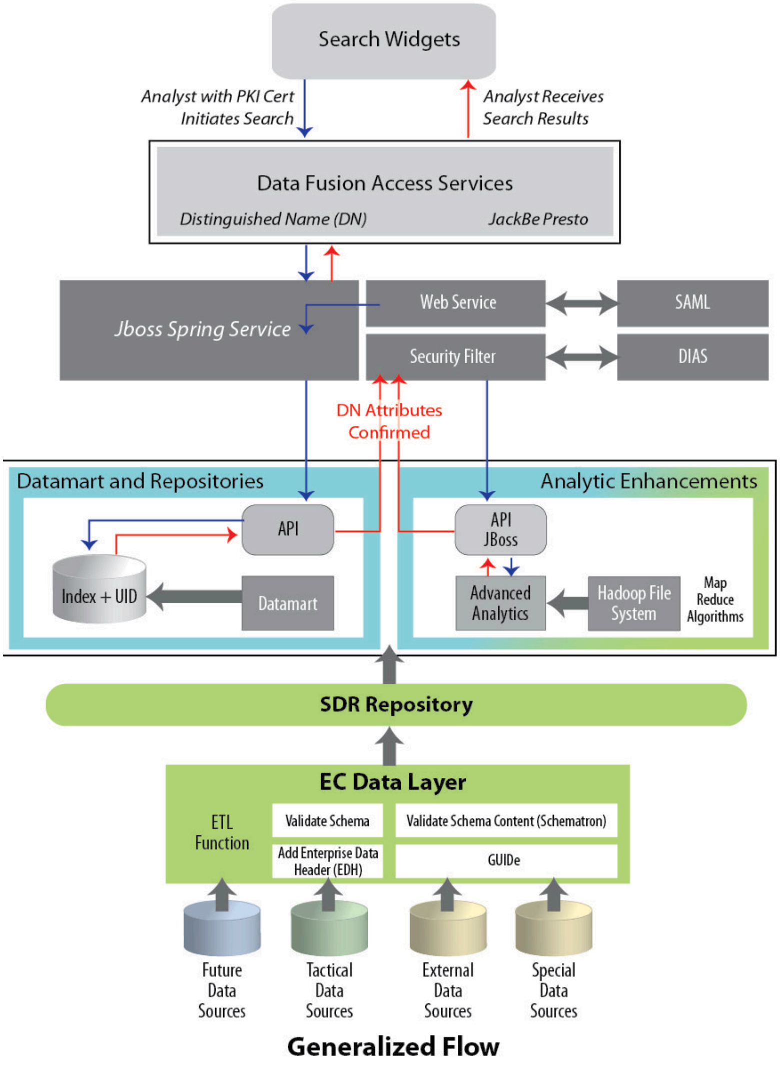


Figure 2: EC Private Cloud Model

and every agency partnering in the IC-Cloud. In other words, like other agencies' private clouds, EC serves its Department of Defense Intelligence Information System (DoDIIS) enterprise, but it also makes its shareable data available to the rest of the IC by participating in the "public" IC-Cloud through its SCM.

This hybrid information-sharing model contains some elements of a COE model, because the IC-Cloud requires partnering clouds to adopt the SCM configuration-prescribed stack as a condition for participation. On the other hand, the same model fosters federated, inter-agency data sharing via web services. Ideally, users from any agency can go to any SCM on the IC-Cloud, security permitting, to obtain requisite shareable information to connect the "dots." Unfortunately, it does not do this seamlessly.

For example, within the EC Private Cloud, EC users can obtain fused information from different sources managed by EC. This can be done in one single search via available ozone widgets or other means using discoverable data services specified/presented by DNI specifications. Using this "one-stop shopping" approach, the EC users then seamlessly compile fused results into their intelligence products (Figure 2). At this time, this is not the case with SCMs and the IC-Cloud.

Section 4: Challenges and Opportunities for Information Sharing on IC-Cloud

4.1 Seamless Sharing Barriers

EC SCM will allow IC partners to retrieve EC shareable information through IC-Cloud. EC users can also obtain information from other partners' SCMs. At this time, however, there is no practical way to issue the same search across all participating SCMs to obtain fused results in one-stop shopping fashion. Notwithstanding legitimate security and other cultural concerns, this limitation damps the usability of the well-intended information-sharing ideals of the IC-Cloud. Therefore, the IC COE Operational Model (Figure 3) with its two-way domain trust framework offers high hope to the IC, because it will allow users, on a need-to-know basis, to "surf" the IC-Cloud for a rich experience in IC one-stop shopping.

EC architecture (Figure 2), with compatible architecture on its SCM, will be ready for such IC information sharing with relatively minimum changes to the architecture. All shareable data in EC SCM is discoverable through RESTful services, retrievable though Ozone widgets or other means. Its data conforms to the DoDIIS Framework and supports DI2E. Its security will be Protection Level 3 (PL3)-accredited (at Initial Operation Capability) and supports need-to-know. Under IC-COE's two-way domain trust paradigm (Figure 3), EC shareable data will become seamlessly discoverable and retrievable across the IC-Cloud. Consequentially, one-stop shopping search and result fused, shared widgets and other advanced analytics can be expanded to cover all the SCM nodes on the IC-Cloud, bringing an enriched experience to the IC end users. Best of all, this much-anticipated intelligence-sharing scenario will enable IC analysts to connect the proverbial intelligence "dots." In more ways than one, the IC-Cloud and its participating SCMs will become increasingly more useful.

4.2 Cross-cultural Knowledge Fine-tuning and Enrichment

Historically, each IC agency developed and fine-tuned its intelligence tools with its own knowledge base. This allowed each IC agency to be in tune with its own culture and modus operandi, thus improving its workforce's efficiency and effectiveness. Case in point: in semantic searches, analysts and technologists develop and augment ontotologies that capture their knowledge on subject matter of interest. Integrating these ontologies into search engines then allows analysts to expand/fine-tune their intended "hits," regardless of what data are involved.

Here is a simple example: In an ontology that an analyst uses to perform a semantic search, the concept or term "table" is associated with "desk," "chair," "furniture," etc. The proximity/hierarchy of each of these concepts/terms in relation to other concepts/terms depends on the culture and modus operandi in which the analysts operate. In this case, a semantic search will use the ontology to retrieve more than just "table." Results with "desk" or "chair" or "furniture" or all of the above may be returned depending upon the search specifications. These kinds of enriched searches have become common must-have tools in the analyst circle. The question then is: How can such ontologies travel with an analyst from one SCM to another SCM to help retrieve information pertinent to the analyst's knowledge base?

In the current IC-Cloud model (Figure 1), there is no provision to allow such a knowledge base to be made automatically available to analysts when they venture out of their own agency's data territory into another agency's realm. If such ontologies cannot dynamically "follow" the analysts as they surf the IC-Cloud for information from sources that pan the cloud, then their ability to perform rich semantic searches can be severely limited. Technically, at this time, sensible ontology portability or harmonization tools are not available. Presently, tools that attempt such portability are neither very useful nor easy nor practical.

Fortunately, EC architecture with its data fusion services layer may provide probable hooks for a dynamic extension from a simple list of registered tags to an expanded list of tags based upon the associations of ontology concepts to registered tags. The expanded list of tags then can be used as search criteria to semantically reach more data, yielding more enriched results than otherwise possible with only registered tags. Since the expanded lists of tags are based upon the analyst's preferred ontologies, they may preserve the effectiveness of semantic searches that analysts have come to find effective.

In the same manner, other agencies may use their own ontologies to expand search terms to achieve similar results on data on EC SCM or any other agencies' SCMs. All these scenarios, however, are predicated upon the assumptions that the IC-COE will become a reality and IC-Cloud surfing will be possible through the two-way domain trust scheme.

As agencies start sharing data, it would be reasonable to predict that they will start sharing knowledge encapsulated in their own ontologies. This extended knowledge-sharing scenario, a much more desirable scenario beyond information sharing, may not be far-fetched. It becomes credible when the level of trust between agencies increases through mutually positive experience with the IC-Cloud and its associated benefits. However, shared ontologies are hardly useful or practical on a machine-to-machine basis, unless they all subscribe to the same frameworks and standards. The contents of ontologies may be different, but by using the same framework and syntax, the chance for one organization to navigate another organization's ontology is entirely possible. Therefore, although the World Wide Web Consortium's Resource Description Framework, Web Ontology Language and Simple Protocol and RFD Query Language may not constitute the most sophisticated ontology framework, they are widely used and will be improved as more people use them. It is common sense to adopt something that is already a standard.

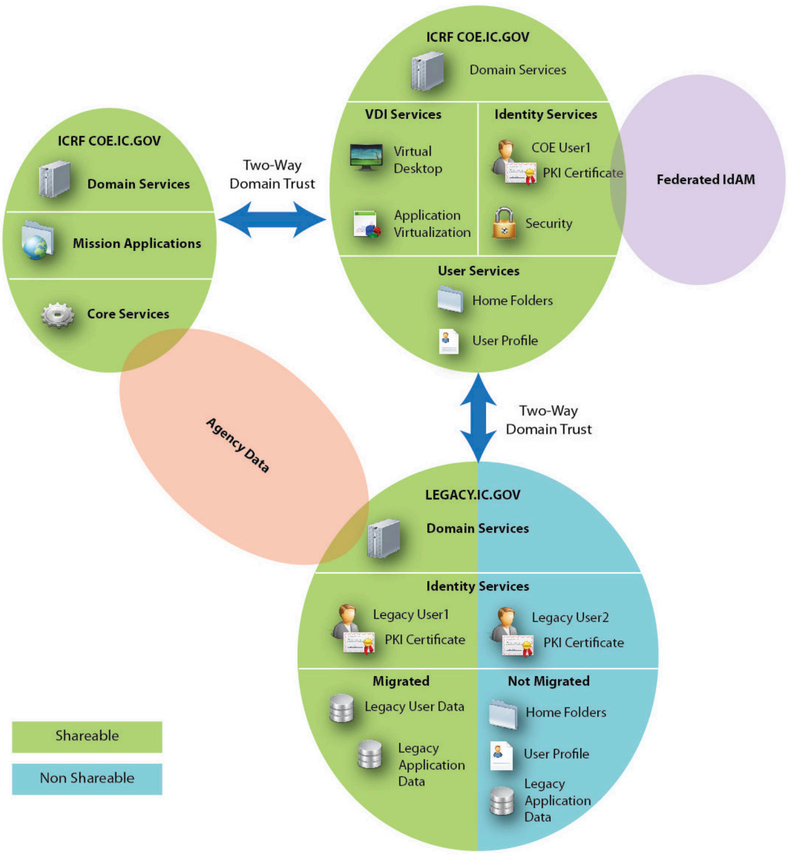


Figure 3: IC COE Model

4.3 Minimizing Data Duplication

In the current the IC-Cloud model, certain subsets of EC native data and systems are somewhat duplicated on the EC SCM. This allows safe sharing with the rest of the IC, eliminating the risk of unauthorized network jumping into a provider's non-shareable repository. The same is likely to be true for other partners' SCMs. The amount of redundant data, however, can be staggeringly large. Over time, especially with the influx of a massive amount of non-structured data, the duplicated data volume can easily be in petabytes if not exabytes. This can be the case even within a single agency. As the demand of sharing data will likely increase as the IC sees the usability of the IC-Cloud, the amount of data can become an increasingly heavy burden on facility, bandwidth and computing resources. In addition, the synchronization between native data and shareable data on SCMs can be problematic due to the ever-growing volume of data.

The burden of duplicating data across agencies can be even more acute. It is not unusual that many agencies are ingesting shareable data from other agencies for their own uses and then, in turn, making these available to other agencies to use. Case in point: EC is ingesting its own data and U.S. Army Intelligence and Security Command-Intelligence Community Data Layer data plus data from other outside sources, such as COMTEX®. In a stovepipe environment, this situation may not present itself as a problem for analysts, since largely they are limited to their own agency's data. In a shared environment such as the IC-Cloud, however, duplication of data sources can become an annoyance or even a major distraction to analysts who surf beyond their agency's enclave. Similar to the old Google search, receiving too many hits of duplicate data simply wastes analysts' time, reduces their analytical efficiency and effectiveness, and increases their frustration. Thus, receiving too many hits would reduce the usefulness of the IC-Cloud.

There are, however, a few great opportunities, especially in the case of EC, for reducing such redundancy in the IC-Cloud within the IC-COE Operational Model:

- Under the current security constraints, physical separation of data to share and data not to share is a sensible approach. However, there is no reason, security permitting, why members of the same agency cannot access all data – shared and non-shared – through a virtual layer, as if the two sets of data were not separated. For external users, shareable data, discoverable and retrievable from the SCM, is nothing more than a virtual layer of the shareable data physically stored in the EC domain. The IC-COE two-way domain trust route will allow IC users to discover and access EC virtualized shareable data seamlessly. Consequentially, data virtualization (Figure 4), security permitting, eliminates the need to duplicate shareable data from a legacy repository to SCM.
- Another fringe benefit of data sharing is the eventual discovery of duplicate data by users who can search shareable data across agencies. Politics and other concerns aside, this kind of discovery can reduce resources (storage, bandwidth, ingest/administration efforts, etc.) while increasing the efficiency and effectiveness of analysts. Of course, this can also help agencies manage their already reduced budgets without reducing the cloud's usefulness to the end-users. Over time, hopefully, there will be mutually beneficial agreements to divide source data provisioning tasking equitably and logically to each agency, thus minimizing the need for data provisioning redundancy.

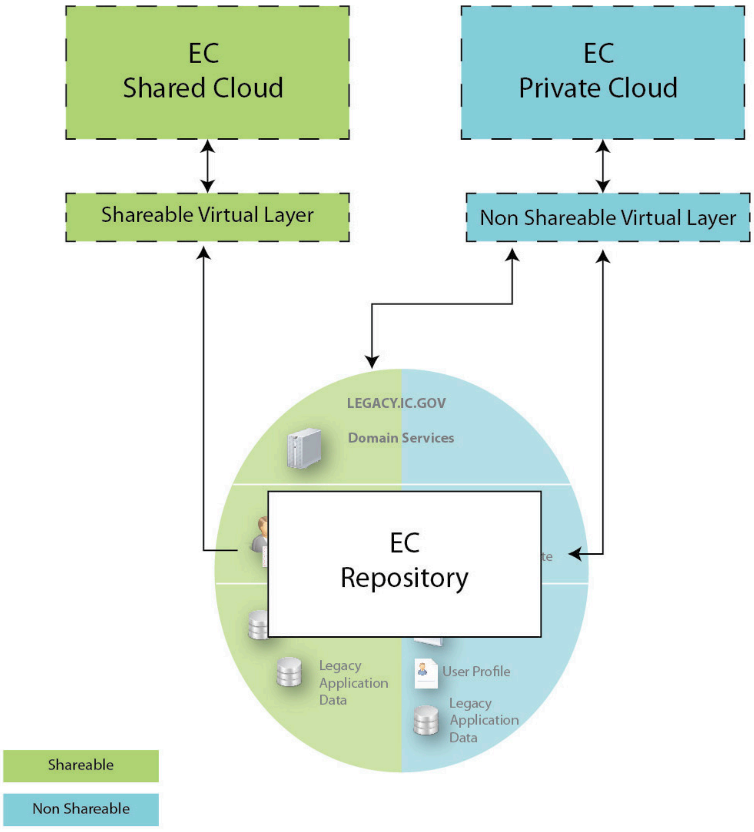


Figure 4: Virtual Data Layers

Section 5: Summary and Recommendations

Through EC, SARA is building major stepping-stones toward better information sharing within SARA and with other sister agencies. Undertaking such an endeavor is a monumental task for SARA and for the entire IC. Taking down one barrier at a time, incrementally overcoming technical difficulties and operational concerns, SARA is building an architecture that satisfies the current IC-Cloud Framework, yet is adaptable to a more mature IC-Cloud that supports the IC-COE Operational Model. Yet the road to seamless data sharing will not be free of obstacles anytime soon.

There are three major challenges facing the IC-Cloud and, consequentially, EC architecture. They are (1) seamless data sharing, (2) supporting cross-cultural knowledge fine-tuning and enrichment, and (3) minimizing data duplication.

1. The EC Team is building an infrastructure of tools for seamless data sharing within the SARA/EC space. Similar experience on the IC-Cloud will depend on the implementation of the IC-COE and its two-way domain trust framework. In the near future, however, it is recommended that SARA experiment a bilateral two-way trust framework with another agency in the same fashion as the IC-COE DTE Memorandum of Understanding with National Geospatial-Intelligence Agency.

2. Data knowledge preservation and enhancement are the cutting edge in intelligence. Rather easily, SARA can implement the first increment of this initiative by building several experimental ontologies for a couple of intelligence domains. Then they can use these ontologies to expand the search terms on EC data to simulate dynamic semantic searches. This experiment will not only benefit analysts at SARA, but can also serve as a reference implementation for other IC partners to adopt.

3. Within current security and technical constraints, data duplication reduction via virtualization probably should be one of the priorities SARA must tackle soon. This is necessary because redundancy can be a major drain on already scarce resources. It is advantageous for SARA to achieve these savings for its own benefit and as a reference implementation for other IC partners to adopt.

By doing the above, SARA through EC, will place itself in the forefront of IC information sharing in the quest to connect the proverbial "dots." Similarly, other agencies sharing cloud initiatives that use similar approaches will be able to make the community quest a reality much sooner.

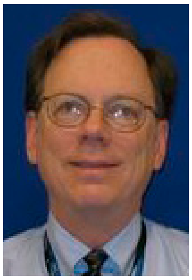
ABOUT THE AUTHORS

Phong Ngo, assistant vice president and technical fellow with SAIC, has more than 30 years of software and systems engineering experience, from COBOL to Cloud. Ngo is a nationally and internationally recognized expert in data management and interchange at the American National Standards Institute and International Organization for Standardization levels; past chair of the ANSI-affiliated subcommittee for data engineering; past chair of the U.S. Technical Advisory Group to its ISO counterpart in data management and interchange; and chief architect of several systems engineering projects.

6909 Metro Park Drive, Suite 200
Alexandria, VA 22310
Phone: 571-319-8479
Fax: 571-319-8382
E-mail: phong.x.ngo@saic.com

David Fado is an information management professional focused on intelligence system processing. He has a background in system modeling to support analytic workflows, specializing in Unified Modeling Language. He was the lead author on one of the early submissions of the UML profile for Department of Defense Architecture Framework and Ministry of Defence Unified Profile for DoDAF and MODAF. He has supported a number of classified and commercial systems.

i-SW
3865 Wilson Blvd, Suite 600
Arlington, VA 22203
Phone: 571-388-0871
E-mail: david.fado@iswcorp.com



John Hoecker, a chief systems engineer with SAIC, has more than 30 years of software and hardware systems engineering and integration experience spanning the systems development life cycle, from concepts and requirements management through operations and maintenance.

Hoecker possesses a B.S. E.E. and an M.S. in systems engineering, both from Virginia Tech and has supported DoD, the IC, and civil agencies. In addition, he was an adjunct instructor of mathematics at Northern Virginia Community College.

SAIC
4001 N. Fairfax Drive, Suite 785
Arlington, Va. 22203
Phone: 703-593-7950
Fax: 703-741-7821
E-mail: john.g.hoecker@saic.com

REFERENCES

- DoDIIS Worldwide Conference Denver 2012, <http://www.ncsi.com/dodiis12/index.html>, April 2012
- Zielecki, Jeff, Intelligence Community – Common Operating Environment (IC-COE), 31 January 2012
- Mitchell, Scott, IC Core Reference Architecture, Overview and Cloud Discussion, 21 October 2011
- Kelly, M.M. and Ngo, P.X., Editors - Chair and Co-Chair, Intelligence Community Metadata Registry Requirements Panel, Final Report, submitted to the IC Metadata Working Group, 20 December 2002
- DoD, DoD Discovery Metadata Specification 4.0.1 -- <http://metadata.dod.mil/mdr/irs/DDMS> , 11 November 2011
- W3C, Resource Definition Framework -- <http://www.w3.org/RDF> 10 February 2004
- W3C, Ontology Web Language (OWL) -- <http://www.w3.org/2009/10/owl2-pr> , 27 October 2009
- W3C, SPARQL Query Language for RDF <http://www.w3.org/TR/rdf-sparql-query> 15 January 2008
- Noy, N.F. & McGoiness, D.L., Ontology Development 101, Stanford University, March 2001
- Ngo, Phong, International Editor, ISO/IEC 11179-6, Information technology – Specification and standardization of data elements. Part 6 – Registration of data elements, 01 April 1997.

Ontology for the Intelligence Analyst

Barry Smith, University at Buffalo and National Center for Ontological Research

Tatiana Malyuta, Data Tactics Corp. and City University of New York

David Salmen, Data Tactics Corp.

William Mandrick, Data Tactics Corp.

Kesny Parent, Intelligence and Information Warfare Directorate

Shouvik Bardhan, High Performance Technologies, Incorporated

Jamie Johnson, EOIR Technologies

Abstract. As available intelligence data and information expand in both quantity and variety, new techniques must be deployed for search and analytics. One technique involves the semantic enhancement of data through the creation of what are called ontologies or controlled vocabularies. When multiple different bodies of heterogeneous data are tagged by means of terms from common ontologies, then these data become linked together in ways that allow more effective retrieval and integration. We describe a simple case study to show how these benefits are being achieved, and we describe our strategy for developing a suite of ontologies to serve the needs of the war-fighter in the ever more complex battlespace environments of the future.

New Demands for Intelligence Analysts

Intelligence analysts are trained to use their knowledge of available sources to enable querying across huge quantities of rapidly changing data. Already the richness and diversity of these sources makes it very difficult for human analysts, even with the most powerful software tools, to leverage their knowledge for analytic purposes. But their problems will only get worse. For while conventional intelligence processes have been focused primarily upon enemy units and on the effects of terrain and weather on military operations, new strategic guidance will require the intelligence community to focus also on disciplines such as cyberwarfare and civil information management [1, 2], and this will imply a massive expansion of the types of information relevant to analysis. The complex operations in which the warfighter of the future will be involved will require not only the mastery of vast quantities of network data but also information pertaining to the entire ecology of daily life in the areas of operation for asymmetric warfare, including information regarding religion, leadership, economics, culture, disease, food, water and other natural resources, and many more. All of this will go hand in hand with a vast expansion of the range of opportunities for the enemy to exploit weaknesses on the side of the warfighter—including weaknesses in our own understanding of this expanded environment of civil/military operations.

This increase in data diversity and volume, and in the velocity of change of data sources will pose an entirely new set of challenges for intelligence analysts, bringing the need for an approach to automated analytics that can solve the problem of rapid integration of heterogeneous and rapidly changing data in a way that can be reapplied in agile fashion to each new domain. This problem is analogous in some respects to the problem faced by warfighters of previous generations, who were attempting to develop the capability for massing timely and accurate artillery fires by dispersed batteries upon single targets. For massed fires to be possible dispersed artillery batteries needed the capacity for communication in real time of a sort that would create and sustain a common operational picture that could be constantly updated in light of new developments in the field. A way needed to be found, in other words, to transform dispersed batteries into a single system of what we might today call interoperable modules. The means to achieve this capability through a new type of governance and training, and through the creation of new doctrine in the field of artillery, were forged only in the early years of the last century at Ft. Sill, Oklahoma [3].

Today, we are facing the problem of *massing intelligence fires*—of bringing all relevant intelligence capabilities to bear on a target of interest in such a way that they, too, can serve as interoperable modules contributing to the development of a single shared evolving operational picture. In what follows we describe a strategy that is designed to address just one part of this problem—a strategy that is already being applied in the field to aid intelligence analysts working with a very large dynamic (cloud-based) data store to support operational decision-making [4]. The approach is of interest not least because it can be applied not merely to enhance existing data sources but also to build new representations in situ to serve analysts in the field.

Military Ontology

An ontology, in brief, is a set of terms and definitions representing the kinds and structures of entities and relations in some given area of reality. An ontology is thus comparable to a computerized dictionary. But it differs from a dictionary in being built around a logically robust classification of the entities in its domain, of a sort that can be used to enhance computer-based retrieval and integration of salient data.

The methods used today in ontology building include getting clear about what the types of entities are in a shared domain of interest, and also getting clear about the sorts of relations between these entities, methods which have been used by commanders and war-planners since the dawn of organized warfare in order to represent the tactical, operational, and strategic-level realities that make up the battlespace (see Figure 1).

The Strategy of Semantic Enhancement (SE)

In the data sources available to the analyst, multiple different terms, formats and data models are used to describe the data. The strategy of SE [6] is a response to the problems created by this diversity resting on the use of simple ontologies whose terms are used to tag (or 'annotate') source data artifacts in a consistent way. Ontologies built for SE purposes provide a restricted vocabulary that will enable analytics tools to see through the inconsistencies and redundancies in the data. This means: providing one term ('preferred label'), and one definition, for each salient type in each domain [7].

As illustrated in Table 1, the terms in an SE ontology are connected together in a simple hierarchy by means of the "is_a" (or subtype) relation. Each term appears only once in this hierarchy, and is associated in a stable way with its parent and child terms in the hierarchy even when new terms or even whole new branches are added to the ontology in the course of time. This stability is important, since the success of the strategy requires ontologies that can be repeatedly reused to annotate many different kinds of data in ways that then serve multiple different analyst communities and thereby contribute to the creation of an ever more comprehensive common operational picture. SE is thus designed to be at the same time more stable and more flexible than the traditional harmonization and integration approaches that, because they are typically based on ad hoc mappings amongst data models, often rapidly degrade in their effectiveness over time.

On the other hand, however, ontology is no panacea. Indeed, the increasing popularity of ontologies in the wake of the Semantic Web [8] has meant that ontologies, too, are now frequently being created in ad hoc fashion to address specific local data integration needs with little or no attention to the issues of consistency and stability. For SE to work, however, it is important that we find a way, through governance, training and doctrine, to counteract this tendency to ad hoc ontology development by bringing it about that a single evolving suite of consistent ontologies is created through the coordinated effort of multiple communities. Already the return on investment from the initial phase of the work described here has shown that such coordinated effort can bring significant benefits by making visible connections between data that had hitherto been walled off in separate siloes.

The Architectural Approach

To this end, the SE ontologies are organized on three levels, with successively greater degrees of flexibility:

- A single, small, domain-neutral Upper-level Ontology (ULO), for which our selected candidate is the Basic Formal Ontology [9].
- Mid-level Ontologies (MLOs), formed by grouping together terms relating to specific domains of warfare, or to specific tasks such as inter-agency information sharing [10].
- Low-level Ontologies (LLOs) focusing on specific domains, for example: EyeColor, HairColor, Name.

The terms used in these ontologies represent what is general or repeatable in reality at successively more specific levels. The level of an ontology is determined by the degree of generality of the types in reality which its nodes represent.

Figure 1: "Rakkasan" Commander Col. Luong issues an opening statement at the start of a sand table briefing. The pieces on the sand table are the result of an ontological process of categorization of the entities in the relevant domain [5].

⌊ = *is_a* (or subtype)

```
vehicle =def: an object used for transporting
  people or goods

  ⌊ tractor =def: a vehicle that is used
     for towing

       ⌊ artillery tractor =def: a tractor that
          is used to tow artillery pieces

           ⌊ wheeled artillery tractor =def: an
              artillery tractor that runs on wheels

           ⌊ tracked artillery tractor =def: an artillery
              tractor that runs on caterpillar track
```

Table 1: Examples of definitions used in SE ontologies.

The ULO is maximally general; it provides a high-level categorization relating to distinctions such as that between an object and a process, or between an object and its qualities (for example temperature), roles (for example, commander), and spatial locations.

The MLOs are general representations formulated using terms (such as database, person, organization) which will be needed by specific communities of SE users and developers.

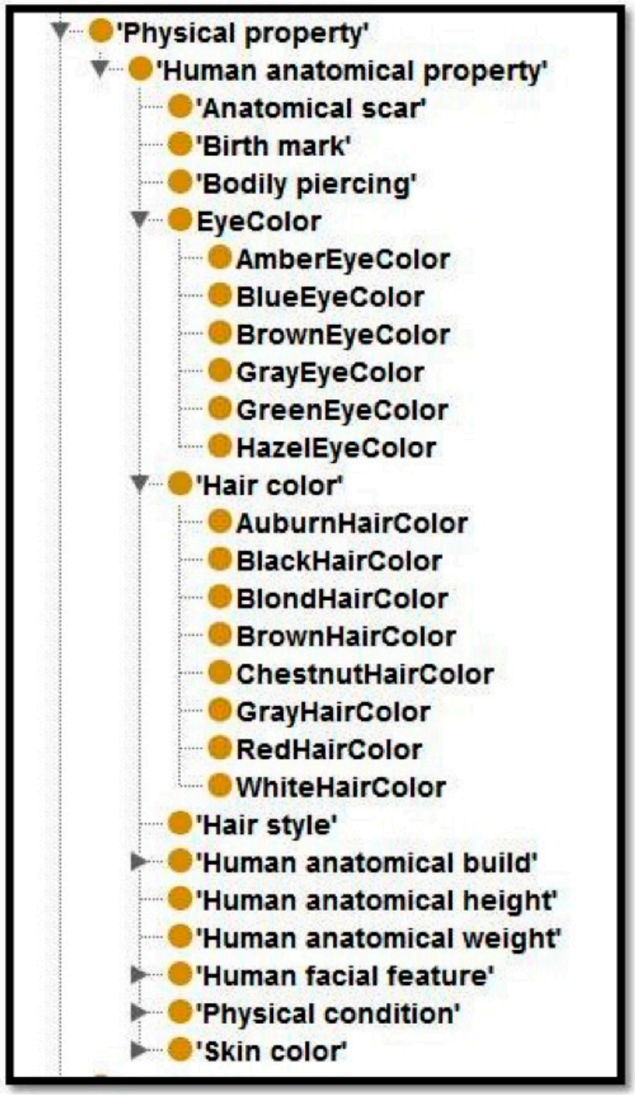
At the bottom of the hierarchy are the LLOs, each representing some narrow homogeneous portion of reality. In the SE approach, the LLOs represent reality in such a way that:

1. For each salient domain, exactly one LLO is constructed that is in conformity with the settled science or military doctrine in that domain.

2. The LLOs are orthogonal (they do not share any terms in common).

3. They are designed to reduce the need for (typically fragile, and costly) mappings between ontologies covering the same or overlapping domains.

4. They are able to be used as reliable starting points for the development of cross-domain ontologies needed for all of intelligence and for specific areas of intelligence analysis.



- 'Physical property'
  - 'Human anatomical property'
    - 'Anatomical scar'
    - 'Birth mark'
    - 'Bodily piercing'
    - EyeColor
      - AmberEyeColor
      - BlueEyeColor
      - BrownEyeColor
      - GrayEyeColor
      - GreenEyeColor
      - HazelEyeColor
    - 'Hair color'
      - AuburnHairColor
      - BlackHairColor
      - BlondHairColor
      - BrownHairColor
      - ChestnutHairColor
      - GrayHairColor
      - RedHairColor
      - WhiteHairColor
    - 'Hair style'
    - 'Human anatomical build'
    - 'Human anatomical height'
    - 'Human anatomical weight'
    - 'Human facial feature'
    - 'Physical condition'
    - 'Skin color'

Figure 2: Human Anatomical Property Ontology

An example SE LLO is illustrated in Figure 2. Other examples are:

- PersonName (with types: FirstName, LastName, Nickname, …)
- PersonIdentification (with types: SocialSecurityNumber, DriverLicenseNumber, …)
- PersonDate (with types: BirthDate, DeathDate, …)
- InformationProvenance (with types: Origin, Credibility, Confidence, …)
- Evidence (with types: ConfirmingEvidence, ContravertingEvidence, ...)

The SE approach is designed to be of maximal utility to intelligence analyst users of data. Ontology content is created only in response to identified situational needs of analysts, and architectural requirements are designed to ensure coherent evolution of the SE resource without sacrificing the flexibility and expressivity needed in actual deployment in the field. As more experience is gained using SE ontologies, intelligence analysts will uncover new ways to exploit the SE resource, and new groups of users will begin to see the benefits to be gained from developing their own complementary ontology resources in a way that is compliant with the SE architecture. Their data will then progressively become integrated with existing SE resources, bringing benefits through increase in the amount, variety and quality of data upon which intelligence analysts can draw [11]. In this way—following a pattern that has been realized already in biology and other domains [12]—the SE strategy will engender collaborative ontology development and re-use over multiple data collection endeavors, both internal and external.

The Discipline of Intelligence Analysis

Joint doctrine [13] defines multiple hierarchically organized disciplines, for example, intelligence, information operations, cyberspace operations; the discipline of Intelligence in its turn has doctrinally defined sub-disciplines such as Human Intelligence (HUMINT), Signals Intelligence (SIGINT), and imagery intelligence [14].

On the typical approach to intelligence analysis, each new set of analytical problems rests on its own collection of data sources, which must be identified and integrated in ad hoc fashion through manual effort by the analyst. A typical analyst may be working with some 100s of data sources, with each source coming from a particular discipline such as HUMINT or Geospatial Intelligence (GEOINT). For an analyst to come to a conclusion or decision, he has to verify each particular piece of information in 3 distinct disciplines. For example, if a GEOINT source says that location X is 'bad', then there has to be something in, say, a HUMINT and a SIGINT source that confirms this statement.

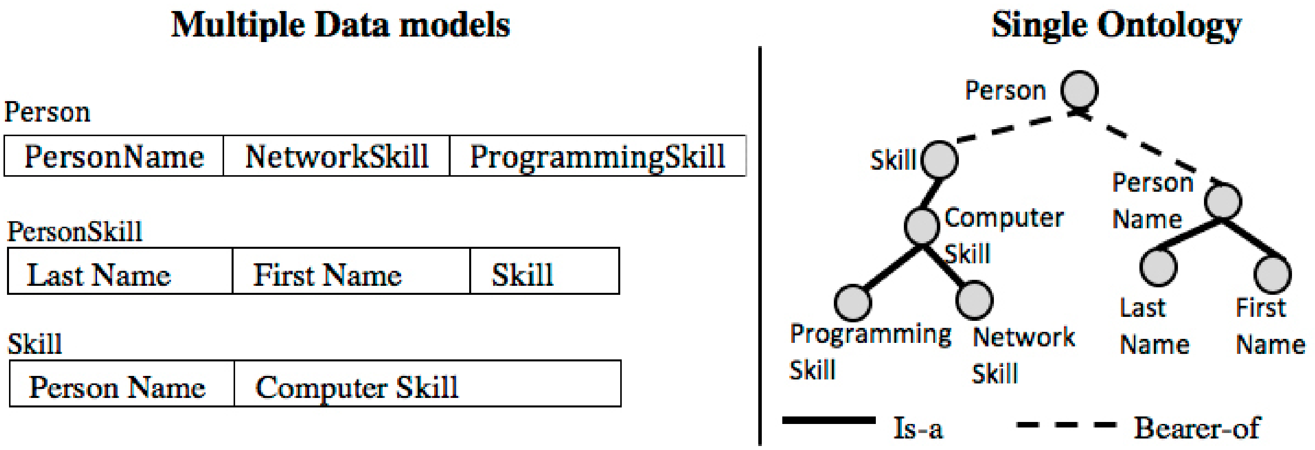
Already here we see the vital need for integration of heterogeneous data for purposes of intelligence analysis. The SE approach has evolved in response to the general recognition that traditional approaches to such integration, both physical and virtual, are increasingly failing in the face of the scale, diversity, and heterogeneity of many data sources and data models. Such traditional approaches fail where they do not address the following requirements:

- Integration must occur without the need for heavy pre-processing of the data artifacts which need to be integrated.
- Integration must occur without loss or distortion of data.
- The integration approach must be able to evolve to accommodate highly heterogeneous and rapidly evolving data.

Already the tagging of intelligence data in consistent fashion by drawing on a simple ontology for describing the different kinds of sources brings benefits to the analyst in a way that meets all of these requirements.

Case Study Illustrating the Benefits Brought by SE to Intelligence Analysis

In what follows we illustrate how these benefits are realized in terms of a simple case study in which the SE approach is applied to a set of cloud-based data sources, including text, images, audio, and signals, as described in [3]. These data sources are stored together with structured descriptions of their associ-



Person

PersonName	NetworkSkill	ProgrammingSkill

PersonSkill

Last Name	First Name	Skill

Skill

Person Name	Computer Skill

Figure 3: Samples of data models, in which arbitrary combinations are allowed (LEFT), vs. SE ontologies, with their constrained hierarchies (RIGHT)

ated data models. The problem that SE is designed to solve arises because different data models can present data about the same entities in arbitrarily many different ways, as illustrated on the left of Figure 3. The SE ontology content illustrated on the right, in contrast, employs simple terms in a stable fashion to ensure that entities of the same types are represented always in the same way.

SE terms are associated with the labels used in the native data model descriptions, as in Tables 2 and 4. To enable benefits from this association in the form of efficient search, the entire aggregated content of our data sources, both structured and unstructured, is indexed, using a Lucene index [15] distributed over Solr [16]. This Index, which is continuously being re-created to ensure synchronization with newly posted data, is a result of pre-materialization; that is, it reflects pre-calculations of the answers to sets of the most common queries posted by analysts.

We consider a simplified example using three native data sources, Db1-3, which we illustrate in each case by column labels and a single row of sample data. To see the sorts of problems we face compare how, in Db1, 'Java' is used elliptically to mean 'Java programming skill', while 'Name' is used to mean 'Name of skill'.

**Source database Db1, with tables Db1.Person and Db1. Skill, containing person data and data pertaining to skills of different kinds, respectively.*

PersonID	**SkillID**
111	222

SkillID	**Name**	**Description**
222	Java	Programing

* *Source database Db2.Person, containing data about IT personnel and their skills:*

ID	**SkillDescr**
333	SQL

* *Source database Db3.ProgrSkill, containing data about programmers' skills:*

EmplID	**SkillName**
444	Java

Second, we use SE ontologies as illustrated in Figure 3 to annotate the data from these databases. Sample results of this annotation are illustrated in Tables 2-4, which are representative of the kinds of tables contained in our aggregated store.

Table 2 contains sample labels used in annotations. The rows of Table 3 represent sample annotations using SE ontology terms. The rows of Table 4 consist of sample statements of the sorts used both in storing native data and in generating the Index.

Label	Source
PersonID	Db1.Person
SkillID	Db1.Skill
Name	Db1.Skill
Description	Db1.Skill
ID	Db2.Person
SkillDescr	Db2.Person
EmplID	Db3.ProgrSkill
SkillName	Db3.ProgrSkill

Table 2. Sample labels from source data models and from the SE ontologies

Source Artifact Label	SE Label
Db1.Name	SE.Skill
Db2.SkillDescr	SE.ComputerSkill
Db3.SkillName	SE.ProgrammingSkill
Db1.PersonID	SE.PersonID
Db2.ID	SE.PersonID
Db3.EmplID	SE.PersonID

Table 3. Sample annotations of labels in source artifacts

To begin to see the benefits of SE for data integration, note how three distinct items in the first column of Table 3—PersonID from Db1, ID from Db2, and EmplID from Db3—are all annotated with the same SE expression, namely PersonID from the PersonIdentification LLO.

Data Value and Associated Label	Relation	Data Value and Associated Label
111, Db1.PersonID	Db1.hasSkillID	222, Db1.SkillID
222, Db1.SkillID	Db1.hasName	Java, Db1.Name
222, Db1.SkillID	Db1.hasDescription	Programming, Db1.Description
333, Db2.ID	Db2.hasSkillDescr	SQL, Db2.SkillDescr
444, Db3.EmplID	Db3.hasSkillName	Java, Db3.SkillName

Table 4. Statements illustrating the sorts of source data used in compiling the Index

The process of annotation proceeds manually as follows. The annotator is required to apply to each label in the target data model the term at the lowest level in the SE hierarchy whose application is still warranted (1) by the meaning of the label and (2) by information the annotator has about the database in question, including (3) information concerning the data values labeled. For example, Db1 contains data about skills in many areas; its label Skill must therefore be annotated with the general term Skill and not with any more specific term. Db2 is known to contain only data about skills in the area of IT; this warrants the use of ComputerSkill in annotating its label SkillDescr.

The Index contains entries of various sorts, as represented in Table 4. Which sorts of entities we index is determined by the ontologies for Person, Place, and so on. The subservient LLOs, which provide the SE labels to be used in annotations for different sorts of data, are used in formulating the field value pairs associated with Index entries.

Currently, the SE Index incorporates the results of inferences over an initial tranche of semantically enhanced content. In Table 5 we see how the Index looks when it is able to incorporate the results of integration over the SE annotations. These inferences rest on the logical structure of the SE ontologies and of their constituent definitions. For example, the term Programmer is defined as Person with programming skill and the Skill LLO incorporates an inferred subclassification of persons, which is represented in the Index using the Subtype field (see the entry for PersonID=444 in Table 5).

When creating the Index, the indexing process crawls statements of the sorts shown in Table 4 and uses SE labels for the Index fields wherever these are available. Thus, as Table 5 illustrates, we obtain fields carrying terms from the LLO Skill and LLO PersonIdentification, as follows:

Index Entry	Associated Field-Value
111, PersonID	Type: Person
	Skill: Java
	Db1.Description:Programming
333, PersonID	Type: Person
	ComputerSkill: SQL
444, PersonID	Type: Person
	SubType: Programmer
	ProgrammingSkill: Java

Table 5. Sample Entries of the Dataspace Index based on the SE

Some native content is not (or not yet) covered by the SE (the Description label from Db1.Skill in our example), reflecting the incremental nature of the SE process. Indexing in such cases is effected using native labels. In this way, incomplete SE coverage of native models does not entail unavailability of the corresponding data to analysts' searches.

A Sample Query Illustrating the Advantages Brought by SE

Suppose the analyst needs to use the Index in order to find, for example, all instances of the type Person referenced in the Dataspace as having some predefined set of skills. When addressed to the sample entries in Table 5, this will yield results as in Table 6.

- entering `Skill = Java` (which will be re-written at run time as: `Skill = Java` OR `ComputerSkill = Java` OR `ProgrammingSkill = Java` OR `NetworkSkill = Java`) will return: persons 111 and 444
- entering `ComputerSkill = Java` OR `ComputerSkill = SQL` will return: persons 333 and 444
- entering `ProgrammingSkill = Java` will return: person 444
- entering `Description = Programming` will return: person 111
- entering `SubType = Programmer` will return: person 444

Table 6: Sample queries over the Dataspace Index and their results with SE

To see the advantages that have been brought to the human analyst by the SE process, contrast now Table 7, which shows Index entries corresponding to those of Table 5 as they would have been generated prior to SE. Table 7 reveals two sorts of obstacles faced by the analyst using pre-SE data. First: because person IDs and names of skills in the native sources are listed under many different headings, querying these sources without SE, even for simple person ID or skill information, requires knowledge on the part of the analyst of the idiosyncrasies of each data source. Second: because data models are flat, in the sense that they do not define hierarchical relations between more general and more specific types, querying across sources that contain data at different levels of detail is virtually impossible.

Indeed, however much manual effort the analyst is able to apply in performing search supported by the Index entries illustrated in Table 7, the information he will gain will still be meager in comparison with what is made available through Table 5. Even if an analyst is familiar with the labels used in Db1, for example, and is thus in a position to enter Name = Java, his query will still return only: person 111. Directly salient Db4 information will thus be missed.

Index Entry	Associated Field-Value
111, PersonID	Type: Person
	Name: Java
	Description: Programming
333, ID	Type: Person
	SkillDescr: SQL
444, EmplID	Type: Person
	SkillName: Java

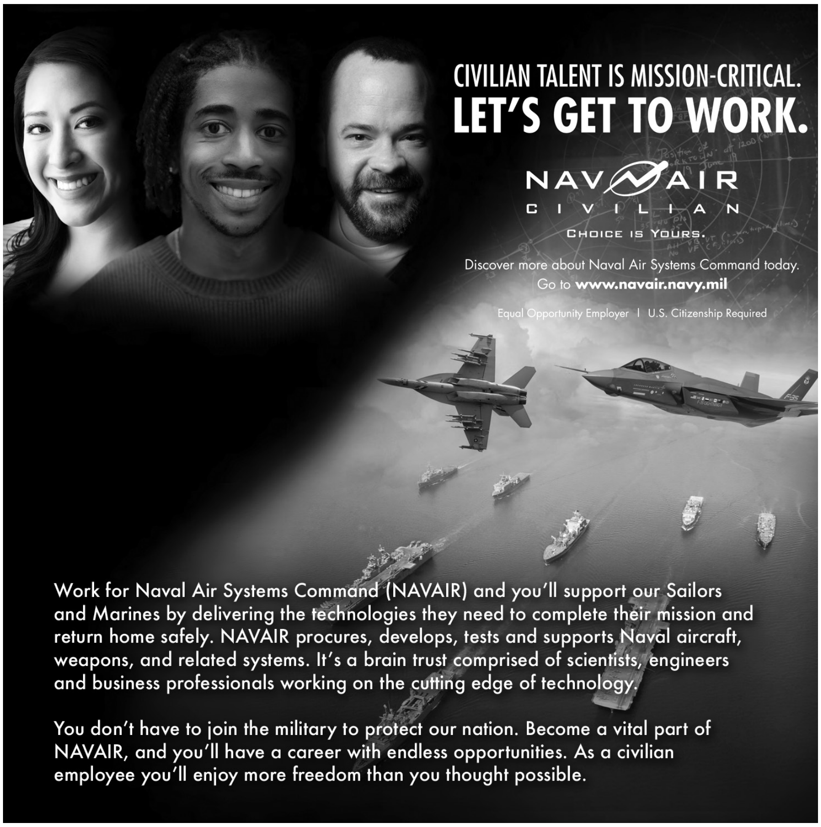
Table 7. Sample Entries of the Dataspace Index prior to SE

Conclusion

Analysts are of course trained to be aware of the types of information that are available in different sources. But in today's dynamic environment, in which ever more domains and ever more associated data sources become salient to intelligence analysis, it is practically impossible for any analyst to know the content of all sources. The likelihood that important data will be missed remains very high, and the need for agile support for retrieval and integration of the sort provided through the strategy of semantic enhancement becomes all the more urgent. This strategy was designed, in effect, to remedy some of the consequences of the inevitable lack of coordination in the development of information resources in the intelligence domain, and thereby to support massed informatics fires against ever-new types of intelligence targets.

Acknowledgements:

This work was funded by U.S. Army CERDEC Intelligence and Information Warfare Directorate (I2WD). The authors wish to thank Peter Morosoff of Electronic Mapping Systems, Inc., H. Shubert of Potomac Fusion, Inc. and C. Rosenwirth of MITRE for their contributions to the content of the SE and for helping us to understand the discipline of intelligence analysis. We also thank Dr. Guha and S. Patel of I2WD for their support and guidance on advancing this body of work.



ABOUT THE AUTHORS

Dr. Barry Smith is a prominent contributor to both theoretical and applied research in ontology. He is the author of some 500 publications on ontology and related topics, with a primary focus on biomedical and defense and security informatics. He is director of the National Center for Ontological Research and University at Buffalo Distinguished Professor.

E-mail: phismith@buffalo.edu

Dr. Tatiana Malyuta is a Principal Data Architect and Researcher of Data Tactics Corporation and an Associate Professor of the New York College of Technology of CUNY. She is a subject matter expert in data design and data integration. Recently she has been working on integrated data stores on the Cloud. She received a Master's Degree in Applied Mathematics and a Ph.D. Degree in Computer Science from the State Polytechnic University in Lviv, Ukraine.

E-mail: tmalyuta@data-tactics.com

Dave Salmen is the Chief Technology Officer of Data Tactics Corporation, armed with over 20 years of extensive experience with full life cycle database system development with an emphasis on initiatives involving intelligence data. His recent work includes DCGS SIPR data cloud (Rainmaker), Information Integration Pilot (I2P), and Zones of Protection (ZoP). He has experience with cloud architecture, cloud data structure design, high volume data ingest, cloud deployment, and cloud security work.

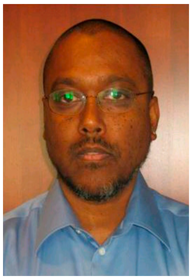
E-mail: dsalmen@data-tactics.com

Dr. Bill Mandrick is a Senior Ontologist at Data Tactics Corporation and an Adjunct Professor at the University at Buffalo. He is also a Lieutenant Colonel in the U.S. Army Reserves with deployments to Iraq and Afghanistan where he has commanded soldiers, planned for major operations, and served as the primary civil-military operations advisor to a Brigade Combat Team. Recently he has been working on intelligence related ontologies for the Intelligence and Information Warfare Directorate (I2WD).

E-mail: william.mandrick@us.army.mil

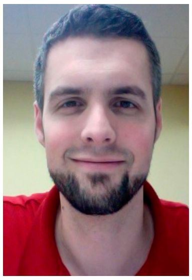
Kesny Parent is a Branch Chief in the Intelligence Information Warfare Directorate (I2WD) at the Communications-Electronics Research, Development and Engineering Center (CERDEC). He has worked in the Intelligence, Surveillance, and Reconnaissance (ISR) domain since 1989. He leads the Development and Integration for the DCGS-A Standard Cloud (DSC) project, a major Army initiative to integrate Cloud Computing Intelligence infrastructure across the entire Intelligence Community. In this capacity, he directed the design, development, and fielding of a highly complex cloud computing architecture with tools that greatly enhance the capabilities available to soldiers.

E-mail: kesny.parent@us.army.mil



Shouvik Bardhan has more than 25 years of experience in the field of complex software design and development and continues to be a hands-on developer on J2EE/PKI/Hadoop based enterprise software. He has managed, architected and delivered systems ranging from FISMA based Certification and Accreditation automation, supply chain management and financial applications to identity federation and document control. Most recently he has worked on U.S. Army's cloud project where as a part of the core development team he design and develops software for an Ultra Large Scale (ULS) Cloud computing environment. He holds a BS and MS in Computer Science (MS from Johns Hopkins University, MD) and is a Ph.D. student in the department of Computer Science in George Mason University, Fairfax, VA.

E-mail: sbardhan@drc.com



Mr. Jamie Johnson, is a Software Developer at EOIR Technologies. He has worked with the Intelligence Community for the past eight years as a Department of Defense Civilian Employee and as a Civilian Contractor. Most recently he has worked on cloud scale search and indexing technologies for the DCGS-A Standard Clouds integrated data store. He received a Masters in Computer Engineering from Stevens Institute of Technology and a Bachelor's Degree in Computer Engineering from Rutgers University.

E-mail: jjohnson@eoir.com

REFERENCES

1. Publication 2-01 Joint and National Intelligence Support to Military Operations, Chairman of the Joint Chiefs of Staff. Washington, DC. 05 January 2012: <http://www.dtic.mil/doctrine/new_pubs/jp2_01.pdf>
2. Strategic Guidance Document, Sustaining U.S. Global Leadership: Priorities for 21st Century Defense, Secretary of Defense. Washington DC. 05 January 2012: <http://www.defense.gov/news/Defense_Strategic_Guidance.pdf>
3. Boyd L. Dastrup, Cedat Fortuna Peritis: A History of the Field Artillery School, Combat Studies Institute Press, US Army Combined Arms Center, Fort Leavenworth, Kansas
4. Distributed Common Ground System - Army (DCGS-A), from 2011 Army Posture Statement, <https://secureweb2.hqda.pentagon.mil/VDAS_ArmyPostureStatement/2011/information_papers/PostedDocument.asp?id=151>
5. For more examples of the role of ontology in the history of military decision-making see <http://militaryontology.com/>.
6. David Salmen, Tatiana Malyuta, Alan Hansen, Shaun Cronen, Barry Smith, “Integration of Intelligence Data through Semantic Enhancement”, Proceedings of the Conference on Semantic Technology in Intelligence, Defense and Security (STIDS), George Mason University, Fairfax, VA, November 16-17, 2011, CEUR, Vol. 808, 6-13.
7. Here ‘type’ is used to refer to what is general in reality (thus: military unit, vehicle, monsoon, headgear, and so on), as contrasted with particular instances (this military unit, that vehicle, last season’s monsoon, Haneef’s keffiyeh, and so on).
8. Tim Berners-Lee, James Hendler and Ora Lassila, “The Semantic Web: A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities”, Scientific American Magazine, May 2001.
9. <http://ifomis.org/bfo.>
10. Barry Smith, Lowell Vizenor and James Schoening, “Universal Core Semantic Layer“, Ontology for the Intelligence Community, Proceedings of the Third OIC Conference, George Mason University, Fairfax, VA, October 2009, CEUR Workshop Proceedings, vol. 555.
11. W. Brian Arthur, Increasing Returns and Path Dependence in the Economy, Ann Arbor, University of Michigan Press, 1994.
12. Barry Smith, et al., “The OBO Foundry: Coordinated Evolution of Ontologies to Support Biomedical Data Integration”, Nature Biotechnology, 25 (11), November 2007, 1251-1255.
13. Joint Publication 1. Doctrine for the Armed Forces of the United States, Chairman of the Joint Chiefs of Staff. Washington, DC. 20 March 2009. <http://www.dtic.mil/doctrine/new_pubs/jp1.pdf>
14. Joint Publication 2-0 Joint Intelligence, Chairman of the Joint Chiefs of Staff. Washington, DC. 22 June 2007 <http://www.dtic.mil/doctrine/new_pubs/jp2_0.pdf>
15. <http://lucene.apache.org/java/docs/index.html>
16. <http://lucene.apache.org/solr/>

Cyber in the Cloud

Lessons Learned from Idaho National Laboratory's Cloud E-mail Acquisition

Troy Hiltbrand, Idaho National Laboratory
Daniel Jones, Idaho National Laboratory

Abstract. As we look at the cyber security ecosystem, are we planning to fight the battle in the same way we did yesterday, with firewalls and Intrusion Detection Systems (IDS), or are we sensing a change in how security is evolving and planning accordingly? With the technology enablement and possible financial benefits of cloud computing, the traditional tools for establishing and maintaining our cyber security ecosystems are being dramatically altered and organizations need a way to effectively manage this transition.

During World War II, the Japanese took possession of U.S. soil only once. For a short period of time, they occupied the tiny islands of Attu and Kiska, off of the Alaskan coast [1, 2]. In response to this occupation, U.S. and Canadian forces fought hard and succeeded in reclaiming Attu, but not without heavy casualties. Coming off of this ordeal, these same forces stormed Kiska armed with the knowledge that they had earned through blood, sweat, and tears, anticipating that conditions would be nearly identical due to the similarity in the islands. Upon securing the island of Kiska, they learned that their efforts been in vain. The Japanese had changed tactics and had slipped through the Navy blockade surrounding the island under the cover of fog and had escaped instead of fighting a losing battle. This did not mean that the fight for Kiska went over flawlessly. In fact, there were casualties due to friendly fire. From this, we can learn a great lesson about the nature of battles and their ever-evolving nature. Just because we understand what has happened in the fight to this point does not mean that we are completely prepared for the fight ahead.

As we look at the cyber security ecosystem, are we planning to fight the battle in the same way we did yesterday, with firewalls and IDS, or are we sensing a change in how security is evolving and planning accordingly? With the technology enablement and possible financial benefits of cloud computing, the traditional tools for establishing and maintaining our cyber security ecosystems are being dramatically altered.

For this purpose, we need to migrate our thinking from an incident-response model, in which we put in place controls and safeguards against threats based on historical activity to a risk management framework, where we assess our greatest risks areas and apply our resources and efforts towards only those risks which demand the most attention.

Additionally, the cyber security domain has been the purview of engineers and technologists. As cloud computing services are deployed, organizational technical personnel will no longer be the sole provider of security controls and incident response. The primary functional domains of cyber security, in the cloud, will be mission/business, legal, and contractual. Technology will remain a critical functional domain, but for many organizations this responsibility will be transferred to the cloud service provider or a joint responsibility.

Recently, Idaho National Laboratory (INL) participated in a push to move e-mail services to the cloud and through this activity has identified some mechanisms that can help facilitate ensuring cyber security in the cloud.

Risk Management Framework

In the past, we have had physical and logical controls over all of the layers in our computing environment and so we were relatively confident that we could defend all of the resources equally well. No longer are we able to put up the fortifications around our network boundaries and treat all of our informational assets equivalently within that boundary. Cyber security exists to protect those information assets of highest value to the organization. As we move towards a cloud model, our control changes and we have to identify both how to best protect those resources with the highest value and identify which resources are and are not candidate to move into a cloud, which is beyond our physical control. To do this, it is first important to understand which organizational resources are candidates to be hosted in a cloud model. The historical "peanut butter spread" approach is not financially sustainable.

Mission/Business Context

The first step in assessing what assets are candidate to move into the cloud is to evaluate the impact of the move in the following areas:

1. Mission/business benefits and impacts
2. Legal analysis
3. Financial analysis
4. Human/cultural impact
5. Technical cyber security review

Within each of these categories, the organization assesses whether the risk profile is affected in a positive or negative manner and to what extent that impact occurs.

Mission/Business Benefits

Technological decisions cannot and should not be made independent of the mission or business. All technology decisions are ultimately business decisions and require that the mission-related benefits be factored into the overall risk assessment. Moving to the cloud can help enhance or hinder mobility, accessibility, flexibility and agility and needs to be assessed to determine if the movement to the cloud assists or precludes the business from achieving its mission.

At INL, one of the major drivers on the horizon is the ability to collaborate and communicate with external partners in the performance of research and development activities, including foreign partners. The use of collaboration in the cloud positions us to meet the business needs for the future.

Legal Analysis

Organizations are legal entities and are bound by Governance, Regulatory and Compliance (GRC) requirements, including:

- Export Control
- eDiscovery
- Information ownership and use rights

Export control entails protection and control of specific information from leaving the boundaries of where it is created. As information moves to the cloud, is it necessary to understand how the risk profile of the information in the cloud change and also the impacts of the organization to control future movement of information. With our acquisition, International Traffic in Arms Regulations (ITAR) information was a significant consideration due to our mission objectives [3].

eDiscovery involves the responsibility of participating in the discovery process and delivering applicable information to a court of law on request. With the tools provided by our cloud provider, we were able to significantly increase our ability as an organization to participate in the discovery process and comply with legal regulations. With the increase of capabilities, the laboratory had to further refine retention policies associated with information. This was to ensure that we were being as protected as possible, while also ensuring that we were maximizing our responsiveness and compliance with GRC requirements.

Information ownership and use rights are also critical. When an organization places information assets in the cloud, ownership and utilization rights to the information have to be addressed, including the rights of the provider to disclose the nature of the relationship to further its own pursuits. The Terms and Conditions and Terms of Service of the contract are the vehicles that establish ownership and utilization along with Federal and State laws.

Financial Analysis

One of the major pushes associated with moving into the cloud is financial. The models associated with the cloud are inherently different from an organization hosting the same solution on premises. The fundamental selling point of cloud computing is that it provides organizations maximum flexibility, especially in terms of incremental investments. With on premises solutions, the financial model requires up-front capital investment to install and configure the solution and then a reduced operational budget over the life of the solution. With the cloud, the up-front acquisition and implementation are reduced, but a greater portion of the total cost of ownership lives as operational costs associated with maintaining the solution.

With cloud solutions, organizations are more agile in their ability to increase or decrease service in small increments based on demand. The extent of this scalability is bound by the nature of the cloud. A cloud with more tenants (e.g. public cloud) is more flexible than one with limited tenants (e.g. private cloud).

At INL, we were moving into the cloud from an organizationally hosted legacy technology that was acquired and implemented during the 1990s. The technology had become outdated and was no longer sustainable and necessitated an upgrade. We opted to adopt the cloud finance model because it allowed demand and supply to be more flexibly matched.

Human Cultural Impact

Although businesses are entities, they are the composite of individuals. It is the cohesion and direction of those individuals under the charge of a defined organizational leadership that makes or breaks an organization. This requires that the impact on the culture for a given solutions needs to be assessed. Understanding whether the move to the cloud will help or hinder individuals from being successful is important. This entails understanding the impact on individual's effectiveness in performing work, attitudes and behaviors towards safety and security, and the perception of their role in security. There is often fear, uncertainty, and doubt among the organization's culture when moving to the cloud because the execution of work changes location and people are uncomfortable with change. This does not automatically exclude the cloud because people are hesitant to change, but the ability to mitigate this risk does need to be assessed. If the organization has the capability and the responsiveness to cultural change, movement to the cloud can be successful. If past efforts have shown that the culture is incapable of making the change, the risk in this area needs to reflect this challenge.

At INL, this has been a significant consideration. We understand that over the next 10 years, a large portion of our workforce will be ready to retire and that the upcoming generation, defined by the Federal CIO as the "Net Generation," [4] will demand working in a much different way than is common in our workplace today. In looking at our current workforce, we have identified that through effective communication and organizational change management, they will be amenable to the change and that it will position us for a more high performance workplace for the future. Balancing the needs of the current workforce and the future workforce has been a significant consideration in the movement of collaboration and communication into the cloud.

Technical Cyber Security Review

When taking any asset into the cloud, it is important to understand the technical impact on other assets. If components of information are moved to the cloud, there is potential for unintended repercussion on other information assets. This is especially critical when information is integrated between systems. If integrated assets are shared between the internal network and the cloud, the overall risk profile of that relationship can potentially increase. The entire scope of the move needs to be understood and the impact to the overall risk profile needs to be assessed.

As INL reviewed the movement of e-mail to the cloud, there were a number of key technical issues that had to be considered. With much of e-mail throughout the laboratory being encrypted in transit, key management was a major consideration in the movement to the cloud. Moving the keys to the cloud did not make sense for the organization, but process had to be established to allow the use of these keys by a service that resides in the cloud. Through the use of OAuth and a security gateway, we were able to preserve complete control of our key management and still be able to administer secure login management to the cloud.

Net Scoring

With each of these areas assessed, we were able to combine to score the direction and relative magnitude of the risk impact to identify the overall risk profile for the organization with respect to moving e-mail into the cloud.

As INL performed the risk assessment of moving e-mail to the cloud, we identified that overall risk profile of our organization improved by moving this particular service into the cloud. Below represents the scoring in this specific assessment:

- Mission benefits (+2)
- Legal impact (0)
- Financial impact (+2)
- Human/cultural impact (+1)
- Technical cyber security review (0)
- Total (+5)

We did not ignore the fact that there would be some technological cyber security challenges as well as some legal challenges relating to export control, but in the end the overall needs of the organization outweighed the challenges.

This does not mean that these areas of challenge need be ignored. In fact, mitigation activities have been put in place to focus on these specific areas as we proceed into the cloud. This allows us to ensure that we are focusing on the right cyber security efforts and not merely the same efforts that we focused on under the on-premises paradigm.

Procurement

Once this risk assessment is complete and the organization understands whether there is a net benefit for the organization to move into the cloud, it becomes crucial to select the right cloud provider who fits conceptually with the positive risk attributes identified above.

Cloud Provider Relationship

In the past, the relationship between an organization and a provider has been characterized in two main ways. The first model is a product sales and support model. This includes engagement through the initial purchase and the establishment of a support contract to deal with product issues. The product provider is most successful when they can provide a solid product that requires limited support. The more effective that a company is in driving down support incidents, the more they can increase their capacity to be profitable. The support contract becomes an insurance policy against risk for the organization and a residual income for the provider. Providers continually engage the organization in selling additional products as a mechanism to further this type of relationship.

The second relationship model is a service provider relationship. This includes a promised service and engagement through the process until the service is fulfilled. Service providers have a financial interest in ensuring continued service excellence because this is where their residual income arises. Organizations look to get the maximum service for the right price point. Providers look to expand the nature and extent of their service offerings to further this relationship.

Many other types of relationships exist, but these two have been most pervasive across the industry in recent years.

With the cloud, a new and slightly different model is emerging. Although, this relationship has many similarities to a service provider relationship, it has some subtle nuances that are more similar to a product provider relationship. Unlike a project, where costs associated with execution are based on a fixed bid, cost plus fee, or actual costs agreement, a cloud provider costs out their service on a licensing model similar to the product provider. This causes some tension between the organization and the provider because the organization is targeting getting the highest service possible and the provider is looking to establish a residual income stream with as little hands-on activity as possible. Cloud providers cannot and do not ignore customer service, but it is fundamental to understand the dynamics inherent to a provider who is trying to find the ideal balance between cost savings and service excellence. A provider is most effective in focusing on those services that are the greatest value-add and eliminating or automating other non-value add services.

This new relationship is very reliant on both the organization and the provider coming together in a partnership and agreeing up front how this relationship will be managed on both sides. This relationship is not formed after the contract has been signed and the service offering begins, but begins prior to the request for proposal leaving the door.

Statement of Work

With an understanding of the nature of the relationship, it is vital that the organization put together a cohesive statement of work that establishes the basis for what services are critical as part of this relationship. This statement of work needs to clearly delineate which aspects of service are must-haves and which aspects are nice-to-haves.

As INL commenced defining the composition of the cloud e-mail service, we pulled together participants from across the laboratory to participate in a road show of the major cloud providers. The purpose of this road show was not to have the end users choose a provider, but to expose the art of the possible and to assess which features were critical for future success. For many in the laboratory, they had been using the same toolset for 15 years and had settled into outdated paradigms. Establishing a new mindset throughout the laboratory was crucial. Primary organizational contributors were:

- Legal council
- Supply chain management (contracting)
- Records management
- Information technology
- Cyber security

From this and other pre-request for proposal activities, INL was able to collect hundreds of individual requirements. We recognized that establishing the statement of work based on a laundry list of hundreds of requirements would not effectively establish the prioritization of services that was critical in the future relationship. As we looked at our risk assessment, there were some key must-have requirements that rose to the top as go/no-go requirements that had to be met by any provider of the service.

Go/No Go Decision Point

With the nature of our environment, information protection was high on the list of go/no-go requirements. This included ensuring that the provider had the right level of controls in place to protect information. This was verified by the provider's ability to obtain a Federal Information Security Management Act of 2002 moderate level certification that they had been through an independent assessment of controls and had met the minimum qualifications set forth by the Office of Management and Budget [5].

In addition, it was necessary that the provider protect the information both in-transit and at-rest based on the Federal Information Processing Standard. This would ensure that the information was being protected as it traveled across the public network and once it was resident in the provider's data centers [6].

With the challenges associated with both export controlled data and ITAR data, it was important to us to have the cloud provider that could support data centers managed only by U.S. citizens. With the potential sensitivity of this information, either physical export of this information to a foreign country or consumption of this information by a citizen of a foreign country could be considered a deemed export. With U.S. citizen managed hosting facilities, we could ensure that outside of the technical protections guarding our information, we would also have an assurance that those technicians coming in contact with the physical hardware associated with our information did not pose risk to exposure of sensitive information.

Finally, in our environment, we needed to ensure that we had secure access to e-mail through mobile devices. This became an important decision point to ensure that the provider could support the current and future mobility needs of our workforce.

Each provider was required to respond as to how they would meet the go/no-go requirements. Since these requirements could be accomplished in multiple ways, it was important to understand the risk profile associated with the manner in which the provider offered each service.

Technical Requirements

The other requirements gathered during the pre-procurement process were very applicable to selecting the right provider, but were included as ancillary technical requirements. Each provider was asked to respond whether they currently had functionality that met the requirement, whether it was planned on their future product roadmap or whether this was not planned as a future feature set.

This allowed us to get a more complete understanding of the nature of both the product being offered and the nature of the service relationship in production.

Summary

With a risk assessment in place to understand which services are candidates to be moved to the cloud and a carefully defined relationship with the cloud provider, organizations have a strong foundation for effectively managing cyber security in the cloud.

Moving to the cloud is not right for every organization, nor is it viable for every application in their environment, but it can provide significant benefits to the organization when it can be accomplished, such as the business benefits. To be successful in moving to the cloud, organizations have to approach it differently than they have in the past by applying risk-based mitigation instead of merely technological solutions. As INL pursued transforming the manner in which we provide e-mail service to our organization, we learned that through the judicious application of a risk management framework to cyber security we could take advantage of this new service delivery model and still ensure effective information protection.

Disclaimer:

This manuscript has been authored by Battelle Energy Alliance, LLC under Contract No. DE-AC07-05ID14517 with the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a nonexclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes.

STI Number: INL/JOU-11-22852

ABOUT THE AUTHORS

Troy Hiltbrand is a lead in Information Management (IM) Strategic Planning and Enterprise Architecture at INL. In this capacity, he is involved with coordinating efforts to align IM activities with laboratory strategy and vision and in ensuring that business process, information, technology, and security are brought together in a way that supports the achievement of mission success.

Phone: 208-526-1092
E-mail: troy.hiltbrand@inl.gov

Dan Jones is currently an Information System Security Manager for the Idaho National Laboratory with Battelle Energy Alliance. In this capacity he is responsible for unclassified cyber security, which includes managing the maintenance and operation activities and DOE Program Cyber Security Plan efforts.

Phone: 208-526-6477
E-mail: daniel.jones@inl.gov

REFERENCES

1. Battle of the Aleutian Islands: Recapturing Attu. (2006, June 12). Retrieved March 5, 2012, from HistoryNet.com: <http://www.historynet.com/battle-of-the-aleutian-islands-recapturing-attu.htm>
2. Beyer, R. (2005). The Greatest War Stories Never Told: 100 Tales from Military History to Astonish, Bewilder, and Stupefy. Harper.
3. Subchapter M - International Traffic in Arms Regulation. (1993, July 22). Retrieved April 5, 2012, from U.S. Depart ment of State: <http://pmddtc.state.gov/regulations_laws/documents/official_itar/ITAR_Part_120.pdf>
4. Naylor, R., & Smith, C. (2010). Net Generation: Preparing for Change in the Federal Information Technology Workforce. Washington, D.C.: Chief Information Officers Council.
5. Federal Information Security Management Act (FISMA) Implementation Project. (n.d.). Retrieved April 5, 2012, from National Institute of Standards and Technology (NIST): <http://csrc.nist.gov/groups/SMA/fisma/index.html>
6. Federal Information Processing Standards Publications (FIPS PUBS). (n.d.). Retrieved April 5, 2012, from National Institute of Standards and Technology (NIST): <http://www.itl.nist.gov/fipspubs/>

Is a Public Health Framework the Cure for Cyber Security?

Brent Rowe, RTI International
Michael Halpern, RTI International
Tony Lentz, RTI International

Abstract. The public health community has developed robust systems for objectively identifying and studying health threats and coordinating interventions, whereas the cyber security community is still relatively immature in its use of an objective, systematic approach. In this paper, we present a detailed public health framework—including descriptions of public health threats encountered and interventions used—and develop parallels between public health and cyber security threats and interventions. We propose that employing a public health framework to understand individual risk preferences for cyber security can identify the types of interventions and related implementation and communication strategies that will more effectively improve cyber security.

Section 1: Introduction

A significant and growing component of U.S. and worldwide cyber security is the relative insecurity of individual Internet users—the threat that some individuals pose to themselves or others through their vulnerability to cyber attack. Cyber threats are difficult to identify and are often poorly understood by users, which may leave them more vulnerable to attacks than they would otherwise perceive. Moreover, the anonymous and dispersed nature of today's cyber threats have proven that these threats are particularly difficult to target for preventative intervention. As the number of worldwide Internet users approaches 2 billion, the scale of affected individuals shows no sign of slowing.

Although a variety of distributed methods have been used to incrementally improve the cyber security of individuals and businesses, a new broad strategic framework may be needed. In the past, organizations and individuals have been marketed to by cyber security companies such as McAfee and Symantec. More recently, a diverse and growing number of software, hardware, and service providers advertise offers to improve cyber security. No centralized approach has been successfully used to coordinate action; the government has played a relatively limited role, developing standards for industry and, more recently, distributing educational materials online and through presentations to schools and civic organizations. At present, regulation is being considered as a way to increase widespread action, with most of the focus on business security.

In light of the complexities of cyber security, the field of public health offers a framework that may help to focus and improve cyber security research and the selection of intervention strategies. Cyber security threats, like public health threats, often pose a risk not only to the targeted or infected individuals but also to others who are at risk of secondary exposures to a contagion. Recently, members of the private sector, public sector, and the research community have begun to discuss the benefits of this new paradigm [1, 2, 3].[1]

Over the years, the public health community has had many successes [4] that may offer models for understanding and addressing cyber security. Much of public health focuses on identifying and monitoring threats, preventing illnesses or injuries before they occur, and diagnosing conditions in early stages when they are most easily treated and cured. Cyber security threats can similarly be addressed by seeking to prevent successful attacks or stopping the spread of threats at various stages of proliferation.

In this paper, we present a public health framework that can be used to identify and describe specific cyber security threats and potential solutions. We then focus on specific ways in which public health research may inform cyber security research by asking the question: how can the established body of public health research be leveraged to assess cyber security risk perceptions, an area of identified need in the cyber security community?[2] A copious amount of research has investigated individuals' risk perceptions regarding the threat and spread of infectious disease and the factors that may influence an individual to engage in activities to prevent disease transmission. We propose that research is needed that seeks to identify types of cyber security interventions—modeled on public health successes—that would be effective in increasing cyber security, based on individual risk preference estimates. Public health successes would be used to select potential cyber security solutions, and models for understanding demand for specific cyber security solutions would be developed based on public health models of risk preference. By improving understanding of cyber security risk preferences, cyber security researchers, and the cyber security industry would be better able to develop and promote products that more effectively and efficiently improve cyber security.

Section 2: Past Research

The cyber security community has yet to identify a suitable framework through which both the private and public sectors can together effectively combat threats to the cyber security of individuals and businesses. Several past research efforts have sought to explore definitions of the threats or to identify potential solutions by using a public health framework [5]. Of particular importance to cyber security coordination is developing an understanding of risk preferences, and the public health community offers many lessons.

Previous papers and research that have looked to the public health domain for lessons on cyber security have focused on identifying the core concepts and practices that could be adopted to promote better "cyber health." In a 2010 white paper published by Microsoft, Scott Carney, Corporate Vice President of Trustworthy Computing, suggested that stakeholders concerned about addressing cyber threats should support practices modeled on efforts to address human illness; moreover, he proposed that cyber security efforts modeled on public health techniques ranging from the simple to the systematic should be adopted. Charney [2] promotes a security approach centered on device health. He lays out two complementary approaches to advancing device health: (1) bolstering efforts to identify infected devices and (2) promoting efforts to better demonstrate device health. Ultimately, this approach would result in devices presenting a "health certificate" that demonstrates the current state of health of the device, which would allow other devices to take a series of actions based on the information contained in the health certificate.

Another recent white paper, issued by IBM [6] argues for cyber security and IT specialists to move away from "military or security metaphors commonly used" and to embrace a new perspective based on the public health and safety model. The paper's authors suggest that the current cyber security paradigm is too rigid and not flexible enough to meet the day-to-day challenges cyber threats present. Instead, the cyber security problem should be addressed in a "flexible, inclusive, and coordinated manner" for which the public health and safety model is well suited to provide and has demonstrated success in doing. The public health and safety model approach to cyber security should focus not only on detection and prevention of threats, but also on "risk-management, coordination, and communication among a broad range of stakeholders." As others have suggested [3] adopting a public health and safety approach could allow for the cyber security problem to be viewed as part of an ecosystem, where problems are constantly evolving.

The most comprehensive view of adopting public health as a model for cyber security has been advanced by Mulligan and Schneider [1]. Mulligan and Schneider argue that cyber security is a public good and any future doctrines of cyber security should recognize the parallels between public health and cyber security as public goods and develop strategies based on this idea.

Section 3: Lessons From Public Health

Definition of Public Health

To consider how public health may serve as a model for cyber security activities, it is necessary to first define the term public health and understand the activities or components that are part of this discipline. In the 1988 Institute of Medicine report The Future of Public Health, public health is defined as "what we, as a society, do collectively to assure the conditions in which people can be healthy" [7]. A somewhat expanded definition of public health is "the science and art of protecting and improving the health of communities through education, promotion of healthy lifestyles, and research for disease and injury prevention."[3] A key element in both of these definitions is that public health refers to the health of communities or populations. Clearly, communities are made up of individuals, and many public health activities involve addressing health issues at the individual level. However, the main distinction of public health as opposed to other types of health care is that public health focuses on the health of groups of people rather than on one person at a time. In addition, although individuals need medical care only at certain times, communities need public health all the time to stay healthy.[4]

A Classification Framework Based on Categories of Public Health Threats

As a starting point for the use of public health activities as a framework for considering cyber security activities, it may be most appropriate to consider the major categories or types of public health "threats," that is, diseases, health impairments, and health risks targeted by public health professionals. We developed the following framework based on a review of various public health classification systems and consideration of the types of threats that are the focus of most public health activities. Further, this framework was conceived with the objective of showing parallels between public health and cyber security; that is, our plan was to present public health threats in a context that would allow for a similar or related classification system for cyber security threats.[5] In our classification framework, public health activities directed at specific categories of threats include the following:

1. Communicable diseases. These threats include illnesses that are directly spread between individuals or can be transmitted between individuals by a nonhuman vector (e.g., spread of malaria by mosquitoes). Examples of public health activities addressing this class of threats include vaccinations, screening and treatment for tuberculosis and sexually transmitted diseases, control of vectors that can spread communicable diseases (e.g., mosquito control), and potential quarantine of individuals who can transmit diseases.

2. Noncommunicable diseases. These include conditions that are not directly spread among people, such as coronary artery disease, cancer, diabetes, arthritis, and chronic obstructive pulmonary diseases. An important characteristics of many noncommunicable diseases is that they may begin as asymptomatic conditions, either undetectable or detectable only by specialized screening tests, and over moderate to long periods of time can develop into lifelong conditions that can severely affect quality of life and survival. Precursors that increase risk for the development of noncommunicable diseases may include communicable diseases; for example, certain strains of human papillomavirus, a communicable agent, can increase the risk of development of cervical cancer. The goals of public health activities related to noncommunicable disease threats are to prevent development of these conditions (through preventing the development of/exposure to risk factors or identifying and treating risk factors prior to disease development), identify conditions early in the course of the disease when they have had limited effects and are more easily treated, and stop further progression of conditions once they have fully developed.

3. Risk behaviors. As a type of public health threat, risk behaviors are not fully separate from communicable or noncommunicable diseases; many risk behaviors can lead to the development of such diseases.[6] However, risk behaviors may be thought of as a separate public health threat because the public health activities addressing them are structured differently.
For the communicable and noncommunicable disease threats described above, public health activities are often focused on the individual; vaccinations and screenings are examples. In contrast, activities addressing risk behaviors often involve edu-

cational intervention targeting broader populations or population subgroups. These activities include programs related to preventing or facilitating the cessation of tobacco use and other types of substance abuse, improving physical activity and nutrition, and encouraging injury prevention through the use of seat belts or bicycle helmets.

4. Environmental exposures. As with risk behaviors, environmental exposures are not fully separate from communicable or noncommunicable diseases; these exposures are threats because they can cause communicable or noncommunicable diseases. For example, environmental exposures include food- and water-borne infectious agents.[7] Nevertheless, environmental exposures are generally considered a separate focus for public health, and often involve public health professionals who specialize in these areas. Further, public health activities addressing environmental exposures generally occur broadly, involving programs that could affect the health or larger population groups rather than focusing on the individual. Public health activities related to environmental exposures include inspection of foods and food processing/preparation facilities and water and air quality testing. Activities in this category of threat also include interventions related to potentially hazardous exposures in the "built environment," such as activities to monitor and minimize exposures to dangerous substances (e.g., asbestos) or other threats (e.g., radiation, excessive noise) in the workplace, homes, or public structures.

We intentionally developed this framework, based on the threats that are the focus of many public health activities and the desire for a parallel structure that can be applied to cyber security, to include the two broad categories of diseases (communicable vs. noncommunicable) and two additional categories of public health threats (risk behaviors and environmental exposures). There is clearly overlap between the two disease categories and the two additional threat categories. For example, participation in health risk behaviors can increase the risk for communicable diseases (e.g., blood-borne infections transmitted via intravenous drug use) and noncommunicable diseases (e.g., smoking and lung disease). Similarly, environmental exposures can include infectious agents (e.g., Salmonella bacteria) as well as pollutants (e.g., mercury or asbestos) that increase the risk of noncommunicable diseases. However, in categorizing different types of public health threats to use as a framework for considering cyber security threats, we felt that including risk behaviors and environmental exposures as separate threat categories was crucial for two reasons:

1. The types of public health responses to risk behaviors and environmental exposures is often different than the responses to communicable or noncommunicable diseases that do not occur as a result of risk behaviors or environmental exposures.

2. There are additional types of health impacts, such as head injuries, burns, and hearing loss, that can result from risk behaviors or environmental exposures and are the focus of public health activities, but are not disease conditions (although they may predispose effected individuals to subsequent diseases).

Although the goal of public health is to protect or improve the health of groups or populations, public health interventions can be broadly classified into two categories based on the unit or level being targeted by an intervention: interventions implemented at the individual level versus those performed at the system (organization, population group, or society) level. Examples of individual-level public health interventions include vaccinations, screening for infectious diseases (e.g., HIV, tuberculosis), cholesterol screening, and smoking cessation counseling. All of these interventions necessitate direct interactions between a health care professional and a potentially at-risk individual.

In contrast, system-level interventions rarely involve professionals whose main activities focus on the delivery of medical care. These interventions seek to reduce the risk of public health threats to large groups of people through a planned action or program rather than focusing on interactions with each individual separately. System-level public health interventions include educational campaigns, implementation of government laws or programs, and policies to reduce or prevent contact with potentially harmful exposures.

In addition, individual-level interventions can be broadly classified into three groups:

- Primary prevention: addressing a potential threat before it can affect an individual
- Secondary prevention: responding to a threat after an individual has been affected but before an adverse impact of the threat has developed
- Tertiary prevention: intervening after an adverse impact of a threat has developed to prevent worsening of the impact

Lessons Learned From Programs and Interventions Addressing Public Health Threats

Based on the framework described above and a review of public health literature, there are a number of important lessons from previously-enacted public health programs and interventions that have relevance for cyber security:

1. For public health interventions to be successful, recipients need to first recognize that a threat exists for which public health interventions would be beneficial. For this to occur, communication is vital. Easily understood information needs to be provided to a diverse audience using a variety of media or communications channels. Overall the goal is to engage and activate the target population. That is, to show that the public health threats are relevant to the target population—that these problems could affect them—and that there are actions they can undertake to address these threats.

2. Once the nature and potential severity of a public health threat is understood, individuals who may receive public health interventions need to be assured of the safety and effectiveness of the proposed interventions from a credible source. The goal here is to introduce potential solutions in a way that establishes a measure of trust.

3. Public health interventions need to be provided in a convenient and attractive (or at least not unattractive) framework. Even if there is belief in the importance of a public health program (e.g., decreasing obesity), individuals will not support or engage in it if participation is difficult, expensive, or incon-

Table 1.
Comparing Public Health Threats With Cyber Security Threats

Public Health Threat Categories	**Definition**	**Cyber Security Threat Categories**	**Definition**
Communicable public health diseases	Threats that are directly spread between individuals or can be transmitted between individuals by a nonhuman vector (e.g., tuberculosis, malaria spread by mosquitoes)	Cyber Security Communicable Threats	Threats that are directly spread between host computers or network hardware/software or, more commonly, are transmitted through ISPs and other backbone Internet providers prior to host- or network-level infection
Noncommunicable public health diseases	In contrast to communicable diseases, these threats that are not spread among people, but people may be at higher risk as a result of communicable disease exposure (e.g., HPV increases cervical cancer risk). Threats often worsen/evolve over long periods of time, and may go from being asymptomatic (detectable only by special screening tests) to having severe effects on quality of life and mortality	Cyber Security Noncommunicable Threats	Some threats are not spread among host computers, but similar to public health, the risk of these threats can be increased as a result of communicable cyber threats (e.g., a cyber virus can be used to launch attacks on others). These threats may affect your computer's performance as well as impacting others security..
Public health risk behaviors	Threats that are based directly on individual actions that may result in communicable or noncommunicable diseases (e.g., intravenous drug use, smoking) or may result in nondisease conditions (e.g., trauma from not wearing a seatbelt in a car)	Cyber Security Risk Behaviors	Very similar to public health, many cyber threats are based directly on individual actions which result in communicable and chronic threats (e.g., going to risky websites, not installing antivirus software, giving out passwords by phone)
Public health environmental exposures	Similar to risk behaviors, these threats may result in communicable diseases, noncommunicable diseases, or injuries, but these threats are based on exposure to pathogens, chemicals, or other hazardous materials (e.g., radiation) at potentially harmful levels in food, water, air, or the surrounding environment (which can be either natural or man-made)	Cyber Security Environmental Threats*	Threats that interfere externally (i.e., external to a computer or a network) with transmission of information can be considered environmental threats. This could include cut computer transmission lines (as occurred a few years ago with some trans-Atlantic lines), problems with satellites, or issues that interfere with wireless networks
N/A	N/A	Coordinated Cyber Security Threats	Threats that require manual, coordinated, or time-specific action as opposed to more automated (i.e., developed, distributed, and then largely ignored)

*Cyber security environmental threats will not be a focus of this paper as the subject of individual cyber risk preferences is not relevant to this type of threat.

Table 2. Characterizing Cyber Security Threats Using a Public Health Scheme

Type of Cyber Security Threat	Definition	Communicable	Noncommunicable	Based on Risky Behavior	Coordinated
Trojan horse programs	Threats hidden in a seemingly legitimate program	X		X	
Back door and remote admin programs	Programs with unknown access "holes"	X	X		X
Denial of service attack	Attacks in which many computers all attempt to access a website or network resources				X
Being an intermediary for another attack	Host or network being used as attack vector/origin	X	X		X
Unprotected Windows shares	Microsoft Windows share folders/drives are created but not adequately secured		X	X	X
Mobile code	Code written for mobile websites that may allow access to information on mobile phones		X	X	X
Cross-site scripting	A malicious script that is transferred to a computer through a URL link, database query, etc		X	X	X
E-mail spoofing	E-mails purporting to be from a trusted source asking for sensitive information or driving traffic to a bad website	X		X	X
E-mail-borne viruses	E-mails with malicious programs attached or links to malicious programs			X	
Hidden file extensions	A file name that appears to be a certain file type but is not.		X		
Chat clients	Chat programs such as AOL IM, Skype, or ICQ being used to send malicious programs attached or links to malicious programs		X	X	
Packet sniffing	A program that captures data from information packets as they travel over the network.				X

venient. To participate, individuals must believe that they will be able to successfully achieve the intended health objective.

4. Information on the nature of public health threats and available interventions needs to be communicated to a wide variety of audiences. Special attention is needed for audiences who are parts of disparate or particularly vulnerable populations, as they may be at increased risk for certain threats but less likely to receive or respond to information on these threats.

5. Multiple organizations (governmental and nongovernmental) need to be involved in responding to a public health threat. There needs to be adequate coordination among these organizations, including rapid communication and sharing of information as well as delineation of roles and responsibilities. Without this coordination, there are substantial barriers to both tracking and responding to potential threats.

6. The unpredictability of individual behavior must be considered. That is, individuals will often engage in activities that may not appear to have a rationale or scientific basis to public health policy makers. Plans need to be made to address reluctance to participate in public health interventions, ranging from increasing communications as to the benefits of a public health program, providing benefits for participating, or instituting negative consequences for not participating.

Section 4: How Does Cyber Security Fit In?

In contrast to the complex, multiparty public health systems and taxonomies described above, the cyber security community is very individualistic and much less rigorous in its analysis of successes and failures. Most of the efforts of the cyber security community are put toward finding new solutions and little attention is given to ensuring adoption or efficacy of these solutions. In fairness, there are not well-accepted metrics for "success" in cyber security—success generally implies a reduction in threats, vulnerabilities, or losses, but each of these is difficult to quantify, and thus widespread disagreement exists over how to determine whether an intervention works. Further, there are significant barriers to collecting information on the effectiveness of cyber security practices (e.g., legal issues regarding the collection, storage, and distribution of personally identifiable information). As such, there is no equivalent in cyber security to public health laws requiring reporting of communicable disease outbreaks or environmental exposures, and no parallel to state and national registries tracking trends in cancer and other noncommunicable diseases.

Given that the cyber security community lacks a suitable framework for both identifying and evaluating solutions, attention has turned to public health as a potential model for cyber security. Many cyber security threats and intervention strategies are well suited to be reviewed through a "public health lens." However, putting all cyber security threats and interventions into the same framework is no easy task. As described above, in public health, threats can be grouped by several primary categories, which are often overlapping. Cyber security threats can be thought of as having similar attributes that can help to differentiate or classify them. Table 1 aims to connect the high-level categories of public health threats with categories of cyber security threats.

As shown in Table 1, the standard public health characteristics all have relevance to cyber security, except for "environmental exposure" which is largely not relevant in describing common cyber security threats.[8] Cyber security threats are attributable to an "attacker," which is not the case in public health. As such, a new threat category was added in Table 1 for cyber threats to help describe the coordinated nature of some cyber threats. However, coordinated responses are part of public health interventions addressing all four types of public health threats presented in the framework discussed above.

Table 2 provides an overview of how various specific types of cyber security threats can be classified or defined using the four cyber security threat categories introduced in Table 1.[9]

Cyber security solutions can also be described and categorized using a public health frame of reference. Table 3 provides a taxonomy of cyber security intervention strategies for individuals based on the public health framework presented above.

Primary prevention strategies in cyber security include avoiding risk behavior (e.g., Internet users visiting untrusted websites or giving out their passwords by phone or e-mail to someone whose identity they do not sufficiently verify)[10] and maintaining good "cyber hygiene," including installing and updating a firewall and antivirus software. Each of these activities can help to prevent an Internet user from unintentionally allowing a virus, worm, or other type of malicious software to be installed on their computer in the first place. Prevention strategies such as these are not 100% effective at preventing malicious software or malware from being installed on a computer, but they do prevent the vast majority of threats.

Secondary prevention techniques would be used to both identify problems that are present (the equivalent of "screening" in public health) and to remove problems once they have been identified. For example, a computer is running slowly and may have various malware running on it. First, the computer would be scanned using antimalware software to look for threats. Thereafter, similar software would be used to remove these threats, if possible without causing damage to legitimate files. If caught early, largely such threats can be mitigating without catastrophic damage to the system.

Finally, tertiary prevention techniques would be used once the threat has already been causing damage, such as mining data on a host computer (e.g., for credit card or other personal information), attacking other computers or systems, or damaging files on the host computer. Interventions like this have a lower rate of success because the threat has already done some damage and long-lasting harm may be unpreventable. However, deep analysis, often more manual versus automated antimalware tools, can often help to salvage some or all of legitimate files and system components and to prevent damage from similar attacks in the future.

Table 4 provides a taxonomy of cyber security system-level interventions for the four classes of cyber security threats. The solutions described are actions which could be taken by a government agency—likely only the federal government would have the technical capabilities—or by certain private party actors such as Internet Service Providers (ISPs) and, in some cases, organizations such as nonprofit information-sharing consortia, which interact with large numbers of computer users

Table 3: Individual-level Interventions for Cyber Security Threats

Type of Intervention		Cyber Security Threat: Viruses and worms (e.g., computer viruses and worms installed on a computer)	Poor behavior (e.g., freely open e-mail attachments and trust all websites)	Distributed attacks (e.g., DDoS attack aimed a shutting down server)
Primary prevention—avoid threat	Avoid "high-risk" behavior	X	X	
	Firewall	X		
	Antivirus software	X		
	Other primary prevention	X	X	
Secondary prevention—address threat soon after onset to minimize damage	One-time or short-term interventions	X		X
	Ongoing interventions	X	X	X
Tertiary prevention—intervene to prevent fully present threat from worsening		X	X	X

Table 4: System-level Interventions for Cyber Security Threats

	Cyber Security Threat			
	Communicable	Noncommunicable	Risky Behaviors	Coordinated
Type of Intervention (at the System Level)				
Quarantine of affected Individuals (by ISPs)	X			
Mandatory individual-level interventions (e.g., Network Access Control)	X		X	
Monitoring of potential threat sources (by ISPs, government, or nonprofit group)	X			X
Secure configuration management				
Regulation of security of software*	X	X	X	X
High priority patching*	X			X
Mandatory reporting of new cases for assessment of breaches/trends*	X	X		X
Educational information describing risk factors	X	X	X	X
Guidelines/recommendations for early detection	X	X	X	
Potential civil/criminal penalties			X	X

* These interventions are not widely used and are largely industry specific or specific to a certain type of data breach/release.

and act as sub-systems. Similar to public health, some system-level interventions target individuals, but focus on broader activities that are likely to benefit larger groups. . Of note, however, many of these actions have not been to date taken or have only occurred in small settings, such as within a business or in a pilot program.

Quarantining of individual computers or computer systems that have been affected or are suspected of having been affected by a certain type of cyber security threat is a way to protect others from being affected by the same threat.[11] For example, quarantining may be appropriate for home Internet users suspected of having been turned into "bots" (i.e., part of a large network, called a botnet, that is being used to attack other individuals or organizations for a variety of malicious purposes). Alternately, a system (in this case an ISP) may reduce home Internet users' Internet speed or only allow them to use certain ports to connect to the Internet, thus restricting the applications they can access and the harm their insecurity may be able to cause to others. More commonly, many companies restrict their employees' access to certain websites to reduce the threat to their computer and reduce the threat to company data that may be purposefully or unintentionally manipulated by an insider. From a public health perspective, this may be thought of as a reverse quarantine (restricting where you can go rather than preventing you from leaving a fixed location) or perhaps the equivalent of travel restrictions (i.e., recommendations not to travel to certain areas due to the increased risk of communicable diseases in those areas).

As in public health, most system-level cyber security interventions focus on activities that are likely to benefit large groups. For example, organizations such as U.S. CERT in the United States currently seek to collect, aggregate, and disseminate such information. Private companies who sell threat information, such as McAfee and Symantec, also identify "threat signatures" that are used by their software packages to help stop threats. As a result of several regulations, many companies are required to implement "solutions" that identify and seek to mitigate threats (e.g., to personal financial information or personal health information held by private companies). If a significant data breach is discovered (e.g., when more than 500 health records are breached), companies are often required to disclose such to the U.S. federal government and contact affected individuals. A new SEC law may result in additional requirements that certain businesses report breaches that occur more broadly than those that affect certain data types.

Another group of system-level interventions includes environmental strategies aimed at mitigating or preventing threats. For example, a multitude of state and federal laws regulate certain types of security controls and tools that must be used to protect data from unauthorized access, and the procedures that must be followed when certain types of data are breached. Further, educational materials on risky behaviors (e.g., for home Internet users) as well as recommended guidelines for early detection of cyber threats (e.g., by businesses) are available targeting many types of threats. Such information is available through government agencies, nonprofit organizations, industry associations, and professional societies, among other organizations.

When attribution of an attack is possible, criminal or civil consequences may be associated with high-risk behavior and environmental threats. Different from public health, in cyber security the threat almost always originates from an individual or group. As such, when the economic impacts are sufficient to warrant investigating and when the attacker can be identified, criminal penalties and possibly civil consequences can result.

In seeking to use a public health framework to better understand and analyze cyber security, one important area of focus is disparities. In public health terminology, disparities exist when individuals belonging to minority groups, lower socioeconomic status populations, or other underserved individuals are more likely to experience the consequences of communicable diseases or environmental exposures, more likely to engage in certain risk behaviors, less likely to have early detection of and appropriate care for non-communicable diseases, and more likely to have impaired quality of life and decreased life expectancy because of public health threats. This is often considered to be a failure of public health.

It is likely that from a cyber security perspective, certain population groups are similarly more likely to experience adverse cyber events or less likely to have "protections" against these adverse events. Although likely smaller in magnitude, this cyber security divide (if it exists) may be related to economics (i.e., sufficient money to purchase appropriate protections), education (knowledge of the existence of an appropriate use of protection), and risk behaviors (willingness to engage in unsafe cyber practices).

Section 5: Conclusions and Recommendations for Research and Policy

The public health community has been very successful in identifying, monitoring, and reducing the health impacts of many types of threats. Given the many similarities between public health and cyber security, the cyber security community would be wise to leverage relevant public health strategies and analysis techniques. Certainly not all public health strategies will have a comparable approach in the cyber security community. For example, many public health threats are the result of naturally-occurring pathogens or biological events; in contrast, in cyber security, the vast majority of threats are man-made.

Although developing a robust community of cyber security stakeholders organized in any way similar to the complexity and scale of public health is daunting, the use of public health research strategies to better understand cyber security risk preferences is a specific area that should be leveraged in the short term. In the future, we plan to use public health risk perceptions research aimed at understanding preferred characteristics of vaccines to stop specific public health threats (e.g., measles) as a model to assess preferences associated with computer antimalware software to more effectively stop certain cyber security threats (e.g., computer viruses). Such research will constitute a first step at leveraging the public health community's analysis of risk preferences to improve cyber security. ◈

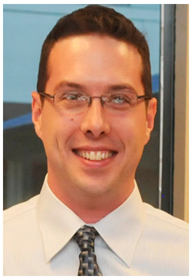
ABOUT THE AUTHORS

Brent Rowe is a senior economist at RTI and director of RTI's San Francisco office. His research focuses on technology policy and security issues. Past work has included assessing the market for Internet service providers to provide more security to home Internet users, estimating home internet users' cyber security risk preferences using a public health framework, conducting a cost-benefit analysis of the National Strategy for Trusted Identities in Cyberspace, and developing economic data to support cyber security strategic planning for NIST. Mr. Rowe is a member of IEEE and has served on numerous government panels and conference committees on cyber security and technology economics. In 2007, he co-authored Cyber Security: Economic Strategies and Public Policy Alternatives.

RTI International
114 Sansome St., Suite 500
San Francisco, CA 94104
Phone: 415-848-1317
Fax: 415-848-1330
E-mail: browe@rti.org

Michael T. Halpern, MD, PhD, MPH, is a Senior Fellow in RTI's Division of Health Services & Social Policy Research. His work focuses on health services, epidemiological, and outcomes research, including evaluations of public health programs; medical technology assessments; health care policy development; and analyses of medical care treatment patterns and quality of care. He is a member of the American College of Preventive Medicine and the American Society of Clinical Oncology.

RTI International
701 13th Street NW, Suite 750
Washington, DC 20005
Phone: 202-974-7813
Fax: 202-974-7855
E-mail: mhalpern@rti.org



Tony Lentz is an economist in the Environmental, Technology, and Energy Economics program at RTI International. Mr. Lentz specializes in the application of economic models to analyze agricultural, environmental, energy, and natural resource regulations, programs, and policies. He has experience conducting projects for the U.S. Environmental Protection Agency, the U.S. Department of Agriculture, DHS, and other government agencies to analyze the economic impacts of climate change, greenhouse gas emissions, renewable energy, energy efficiency, technological innovation, and regulatory compliance.

RTI International
3040 Cornwallis Road
Research Triangle Park, NC 27709
Phone: 919-541-7053
Fax: 919-541-7155
E-mail: alentz@rti.org

REFERENCES

1. Mulligan, D. K., & Schneider, F. B. (2011). "Doctrine for Cybersecurity". Daedalus, 140(4), 70–92.
2. Charney, Scott. Collective Defense . Applying Public Health Models to the Internet. Microsoft, 2010.
3. Department of Homeland Security (DHS). (2011). "Enabling Distributed Security in Cyberspace: Building a Healthy and Resilient Cyber Ecosystem with Automated Collective Action". Available at: <http://www.dhs.gov/xlibrary/assets/nppd-healthy-cyber-ecosystem.pdf>.
4. Centers for Disease Control. Ten Great Public Health Achievements --- United States, 2001–2010. Morbidity and Mortality Weekly Report, May 20, 2011; 60(19);619-623.
4. Charney, Scott. Collective Defense . Applying Public Health Models to the Internet. Microsoft, 2010.
5. Ng, B. Y., Kankanhalli, A., & Xu, Y. C. (2009). "Studying Users' Computer Security Behavior: A Health Belief Perspective". Decision Support Systems, 46(4), 815–825.
6. IBM. Meeting the cybersecurity challenge: empowering stakeholders and ensuring coordination. <https://www-304.ibm.com/easyaccess3/fileserve?contentid=192188; Feb. 2010>.
7. Institute of Medicine. 2008. The Future of Public Health. National Academy Press, Washington DC, page 1.

NOTES

[1] <http://www.dhs.gov/xlibrary/assets/nppd-cyber-ecosystem-white-paper-03-23-2011.pdf>

[2] For example, see Bruce Schneier's 2008 essay on the subject at <http://www.schneier.com/essay-155.htm>.

[3] <http://www.whatispublichealth.org/>

[4] <http://sph.washington.edu/about/whatis.asp>

[5] Our proposed framework is clearly not the only framework that can be used to classify public health threats. However, we believe this framework does capture the main categories of threats in a comprehensive and efficient manner that lends itself to examining parallels with cyber security threats.

[6] It is also important to consider that almost any behavior (e.g., eating a meal, crossing a street) can lead to adverse health consequences. In the context of this paper, we consider risk behaviors to represent conscious actions that increase the likelihood of adverse health consequences beyond that experienced as part of standard activities of everyday living (however that is defined).

[7] In the context of this paper, we consider environmental exposures to be those exposures that increase the risk of adverse health consequences beyond the baseline experienced during standard (or even optimal) periods. Further, environmental factors are generally considered passive; that is, an individual is often subject to an environmental exposure without his or her knowledge or choice, while participation in a risk behavior implies a conscious choice.

[8] Note that there is a type of cyber threat that could be considering as "natural" or similar to an environmental threat to public health. Anything that interferes externally (i.e., external to a computer or a network) with transmission of information would fall into this category. This could include cut computer transmission lines (as occurred a few years ago with some trans-Atlantic lines), problems with satellites, or issues that interfere with wireless networks. Although there is this category of "environmental cyber threats," for our purposes, they are outside the scope of this discussion which focuses on issues more directly related to cyber threats.

[9] This list comes from CERT at Carnegie Mellon. See <http://www.cert.org/tech_tips/home_networks.html>.

[10] Social engineering and phishing are common approaches used by cyber attackers to gain information such as passwords. In phone calls or e-mails, the attackers pretend to be a trusted source–a company IT helpdesk or bank employee–and ask for information which can be used to access their computer, e-mail accounts, bank accounts, etc. Such approaches are very common and often very successful.

[11] <http://www.microsoft.com/en-us/news/exec/charney/2010/03-02rsa2010.aspx>

The Perfect Process Storm

Integration of CMMI, Agile, and Lean Six Sigma

Peter D. Morris, PMP, Process Engineering Consultant

Abstract. Many organizations have struggled over the past few decades with a blizzard of process improvement methodologies such as Total Quality Management (TQM), Kaizen, JIT Production, and Re-Engineering. These operations are understandably leery of adopting new methodologies given their past experience, especially with a focus on return on investments and leveraging existing practices. This article examines the relationship of Agile, CMMI®, Lean Production and the Six Sigma Define, Measure, Analyze, Improve, and Control (DMAIC) roadmap. The intent is to explain how these methodologies might be synergistically combined for a cohesive approach to enhance continuous process improvement.

Introduction

CMMI, Lean Six Sigma (LSS) and Agile development are arguably the most commonly used methods of process improvement in today's technical workplace. Many operations are unique in that they employ all three methods in their project portfolio. This article proposes combining these seemingly disparate methods into a cohesive approach to enhance project process improvement.

- CMMI helps integrate traditionally separate organizational functions, sets process improvement goals and priorities, provides guidance for quality processes, and establishes a point of reference for appraising current methods and procedures.
- Six Sigma's implicit goal is to improve all processes to produce long-term defect levels below 3.4 defects per million opportunities [1]. In recent years, some practitioners have combined Six Sigma ideas with Lean Production manufacturing to yield the LSS methodology that incorporates the elimination of waste; including process waste.
- Agile development is characterized by frequent rapid delivery of useable software by self-organizing teams with regular adaptation to change [2]. Working software is the principal measure of progress; and increased throughput (velocity), by reduction of bottlenecks, is the primary measure of efficiency.

A Brief History of Process Improvement

"I can say, without the slightest hesitation, that the science of handling pig-iron is so great that the man who is … physically able to handle pig-iron, and is sufficiently phlegmatic and stupid to choose this for his occupation, is rarely able to comprehend the science of handling pig-iron."
-Frederick Winslow Taylor, father of Time and Motion Studies

"The old days is just 15 years ago."
- Billy Corgan, The Smashing Pumpkins

Frederick Taylor, regarded as the father of scientific management, was a mechanical engineer in the late 19th century who sought to improve industrial efficiency. Taylor thought that by analyzing work, the "one best way" to do it would be found. He is most remembered for developing scientific management and time and motion studies, wherein a job was broken down into its component parts and measured to the hundredth of a minute.

In my University of South Florida college days, one of our classes delved into Taylor's work. During an exercise where we practiced measuring a worker's activities, I remember the instructor noting, "Make no mistake about it. While you are standing there with your stopwatch scribbling timed activities on your clipboard, that worker hates your guts."

It was at that moment I decided to avoid this profession altogether. Nonetheless, as I went on to be an engineer and project manager most of my life, it seems clear I came to embrace measures, metrics, and process enhancement. I have now spent the last seven years as a full-time process improvement consultant. Go figure.

Modern process improvement began around 1948 with the Japanese Kaizen system, targeting quality, effort, employee involvement, willingness to change, communication, and elimination of waste in business processes. This led in the 1980s to the popular but short-lived TQM concept, meant to improve quality by ensuring conformance to internal requirements (stifling yawn). Then in 1986 the marketing people at Motorola invented Six Sigma, an exciting quality improvement initiative promising to reduce the number of defects and impurities to zero. No one knows quite why they selected six instead of five or four sigma, but it was the new wildfire once Jack Welch at GE went nuts over it and became its leading advocate [3]. Since any project manager can see that a team laser-focused on defects will neglect all their milestones in pursuit of such perfection, this opened the gate in 1990 to Lean Production, based on the Toyota Production System (sometimes called JIT Production), which had fallen in popularity by 1975 in favor of the more generic Lean Production system. In Lean Production, everyone involved in making a product—design and manufacturing engineers, suppliers, laborers, even marketing and salespeople—works together from concept through production. And because the team is focused on one product, there is a cycle of continuous improvement, resulting in cost savings [4].

In the late 1990s AlliedSignal and Maytag decided to combine increased production and reduced defects with the introduction of LSS. Any CEO leery of a process keyed to a single parameter had to love the sound of LSS. In 1996, paired programming and iterative development began when Kent Beck invented Extreme Programming to rescue a Chrysler project that had been scrapped. This first Agile project was subsequently followed by projects using similar iterative methodologies including Scrum, Crystal, and Feature-driven Development leading to the meeting of the Agile Alliance in 2001 where a dozen or so guys (most visibly were Alistair Cockburn, Kent Beck, and Jim Highsmith) generated the Agile Manifesto, promising working software every 30 to 60 days. Software teams worldwide dumped the Waterfall methodology best known for its phased approach where code was not developed until the full set of requirements were identified, documented, and designed (often taking years) for not just rapid development, but rapid delivery. In the software field, LSS concepts have been influential in the formulation of the "Agile methodology."

As shown in Figure 1, the confluence of Agile, LSS and CMMI created a potential perfect process storm that in large part has yet to be realized. Many organizations employ at least one of these process methods, but few if any have deployed all three in tandem despite the benefits of doing so.

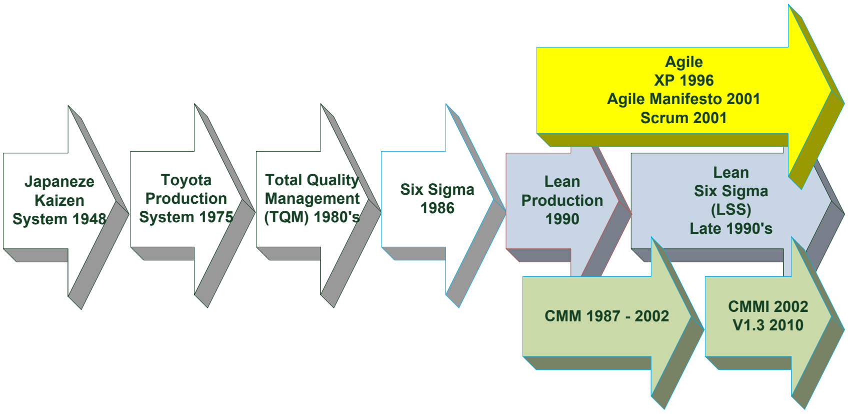

Japaneze Kaizen System 1948
Toyota Production System 1975
Total Quality Management (TQM) 1980's
Six Sigma 1986
Lean Production 1990
Agile XP 1996 Agile Manifesto 2001 Scrum 2001
Lean Six Sigma (LSS) Late 1990's
CMM 1987 - 2002
CMMI 2002 V1.3 2010

Figure 1. Timeline of Modern Process Methodologies

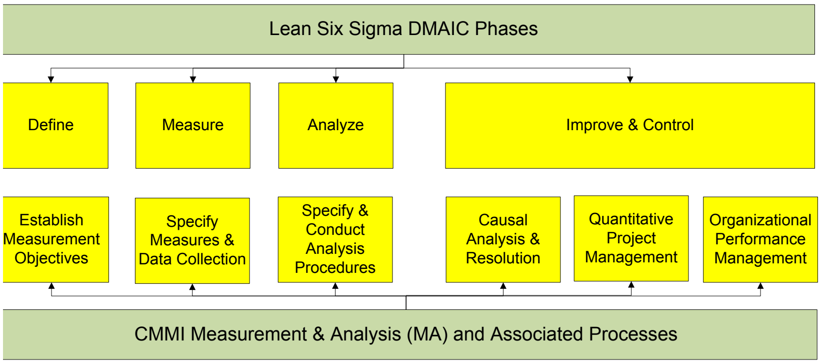

Lean Six Sigma DMAIC Phases
Define
Measure
Analyze
Improve & Control
Establish Measurement Objectives
Specify Measures & Data Collection
Specify & Conduct Analysis Procedures
Causal Analysis & Resolution
Quantitative Project Management
Organizational Performance Management
CMMI Measurement & Analysis (MA) and Associated Processes

Figure 2. Relationships between LSS DMAIC and CMMI processes

While LSS has connections to multiple CMMI Process Areas (PA), this discussion is primarily limited to interrelationships between Measurement and Analysis (MA) and LSS. Hence, the DMAIC aspect of LSS is considered in the process improvement context of CMMI MA, whereas other relationships of LSS that align more closely with the project execution aspects of CMMI are addressed later under "DfLSS and CMMI Compatibilities."

As measurement is critical to both LSS and CMMI, an understanding of how they relate in the context of the MA PA is central to envisioning how they might be used by an organization in combination. The following subsections show how the four general MA areas align with the DMAIC roadmap.

CMMI Measurement and Analysis and LSS

Years ago I began working as a program manager for a software firm that automated various financial functions for about half of the world's 100 largest commercial banks. In my first week I was asked to solve a problem for one of our New York-based banks. Originally estimated as a one-year project, we were already over a year into the implementation, only 50% complete, and we were charging them more than double the original budget. I defined the problem using existing data from which I created measures that allowed me to analyze the inconsistencies. I then improved and controlled the situation through a report to senior management regarding inadequate estimates, double-billing, unacceptable bug rates and other pertinent facts. We then negotiated our billing at 50 cents on the dollar and re-baselined the schedule. I had inadvertently employed an LSS DMAIC solution for an emergency one-time fix. The ROI on this was that we did not get sued and the client remained a loyal customer.

Impressed with my solution, the company asked me to fix the remaining projects that were experiencing similar problems. On average our projects' time-to-market was about 200% of estimate and our defect rate was through the roof. Development blamed Quality Assurance (QA) for testing beyond the requirements and QA blamed development for not coding to requirements. Fortunately we had already collected data on estimates-to-actuals and defect rates by lifecycle phase. I specified measures on this existing data and verified the productivity and quality issues. I then informed the entire company that I would be measuring actuals against estimates by phase along with defect rates, and would be issuing a report after the next billing period. To my surprise, the next period actuals averaged 90% of estimates and defects were virtually non-existent. By simply measuring the problem I had changed it for good. Using causal analysis and resolution techniques I discovered that once project personnel realized they were a team and would be held accountable individually, they began communicating. Business analysts wrote less ambiguous requirements and developers sat down with testers to explain why they coded a function a certain way based on those requirements. Here, I had accidentally used MA techniques suggested by CMMI to solve a problem for the long-term. The relationship between LSS DMAIC and CMMI processes are graphically depicted in Figure 2 and detailed in the following sections.

Define Phase and Establish Measurement Objectives

This first step in the CMMI MA process area aligns very closely with the define phase in a LSS DMAIC project, as indicated in Figure 2. The first important distinction and added value that comes from the conjunction of LSS and CMMI is that LSS places primary emphasis on understanding and managing performance (outcomes) while CMMI (often in practice if not in principle) places more emphasis on maturity level. Whereas maturity level is important for government organizations in particular, it may not be sufficient in and of itself to quantitatively demonstrate improved outcomes in terms of cost, quality, or cycle time.

Using DMAIC, the LSS roadmap provides a very specific approach to establishing the overall objectives and identifying potential measures for an improvement project. Similarly, when properly structured, measurements established under the CMMI MA process should trace directly to business and/or project objectives. In either case, project metrics that fail to support such objectives were likely established as a "good idea" initially, but provide no benefit to the project. They should be discarded as a waste of time. One of the greatest impediments to a successful measurement program is the perception that data collection and analysis are being performed for no apparent reason, or because "we have always done this."

Whether in LSS, CMMI or Agile, individual metrics, as well as the overall metrics program, should be evaluated periodically for usefulness. Notably, in many cases the metrics program itself has been discovered to be the true roadblock to productivity.

Back in my programming days for a defense contractor we were required to count Lines Of Code (LOC) generated each quarter by project. Eventually we developed a macro that differentiated between code, comments, and spaces and spit out LOC. We then stored the data like squirrels hoarding nuts for winter. Winter never came and at project end we just disposed of the data. I guess someone once thought this would be a useful exercise, but all it did was create bottlenecks and reduce velocity. When you collect measures, be sure to follow the Agile concept of avoiding activities that do not contribute to the final product.

Measure Phase and Specify Measures and Data Collection

"Data is like garbage. You had better know what you are going to do with it before you collect it."
-Mark Twain

These second and third general steps in the CMMI MA process area very closely align with the Measure Phase of DMAIC. Again, the LSS roadmap provides detailed guidance for how to conduct these activities. The Measure Phase in LSS is prescriptive while the CMMI MA process area is proscriptive. SEI is unconcerned with the method, as long as the process is defined and repeatable. Use the measure phase of DMAIC to accomplish the goals outlined in CMMI. For example, use the guidance in the plan to measure results and plan to collect data steps from the DMAIC measure phase to accomplish the establishing objectives and specifying measures specific practice in the CMMI MA process area. Similarly, use the guidance in the collect and qualify the data step of the measure phase in DMAIC to collect measures and place them in a measurement repository to satisfy the CMMI MA specific practice of specify data collection and storage procedures.

As a project manager I found that senior managers were always extremely impressed with huge amounts of measures being collected, analyzed and processed. More seemed to be better. Especially in an Agile environment, the development staff will take the opposite approach: keep it simple. Two or three measures, probably dealing with velocity and defect rates, should keep an Agile team busy and informed.

Analyze Phase and Specify and Conduct Analysis Procedures

The analyze phase of DMAIC encompasses the activities envisioned by the MA requirement to specify and conduct analysis procedures. LSS training includes instruction in selection and application of appropriate statistical tools, including criteria for determining which tools and methods are most applicable to a particular situation. While the argument prevails that the DMAIC roadmap provides detailed guidance on how to proceed, and the CMMI MA processes leave such decisions up to the practitioner, organizations usually provide this direction within project measurement plans. The SEI promotes the use of operational definitions for each specified metric. This is typically a table that defines the supported goal, collection/storage criteria, and review techniques spanning simple trend/variation analysis to complex statistical process control.

Report Definition	The Time Tracking Report shows remaining work & accuracy against estimate for both individual tasks and overall backlog.
Goal Supported	Increase Velocity and Increase Quality
Collection Procedures	Automated through JIRA
Collection Criteria	JIRA drop-down list selections required to generate this automated report : Browse Project Tab Select: Time Tracking Report (under Reports on right-hand side) Fix Version: <Version Number> Sorting: Least Completed First Issues: All Sub-Task Inclusion: Only include sub-tasks within the selected version
Derived Measure	List of tasks indicating remaining work & accuracy against estimate for both individual tasks and overall backlog
Storage Procedures	The derived measure will be stored for historic reference according to the Configuration Management Plan

Figure 3. Operational Definition: JIRA Time Tracking Report

Consequently, any specific instructions and criteria demanded by an LSS application can easily be incorporated into the CMMI MA operational definition framework. An example of an operational definition for an automated metric generated through the Agile scheduling and issue-tracking tool JIRA (relax, it is freeware) is given in Figure 3.

Improve and Control Phase and Using the Measures and Analyses

"You will miss 100 percent of the shots you never take."
-Wayne Gretzky

The final steps in DMAIC (Improve and Control) parallel CMMI Level 4 and 5 Support Process Areas. The structure of the CMMI separates MA, where data is collected and analyzed, from the other process areas (causal analysis and resolution, organizational innovation and deployment and quantitative project management) that use the measures and analyses to define and implement improvements. In this respect DMAIC is structurally, although not substantively, different from CMMI. DMAIC envisions a continuous flow of activities from problem definition through solution and implementation performed by the same team, illustrating a distinction between the CMMI "what" (defined by a series of PAs) and LSS "how" (defined by a project roadmap such as DMAIC as shown in Figure 2). Again, the combination of CMMI and use of organizational measurement processes currently provides the "how" and "when" aspects used by LSS practitioners. Additionally, while the requirements of MA are limited to analysis and communication of results, SEI encourages expanding MA efforts to include aspects of higher-maturity to take advantage of benefits associated with causal analysis resolution and quantitative project management. In fact, leveraging these benefits tends to improve MA results and provides the organization with tools to achieve the project's established quality and process performance objectives.

If the primary goal of an improvement initiative is to create organization infrastructure and institutional capability (as SEI intended in government organizations for which CMMI was originally designed), then the separation of MA from various types of improvement activities clearly makes sense. MA focuses on the creation of measurement infrastructure, while DMAIC is typically more narrowly focused on time-limited resolution of a specific problem. Although different in approach, the result over time is essentially the same. Therefore, the integration of LSS and CMMI provides the opportunity to institutionalize a measurement infrastructure that supports quick response to problems that require immediate attention and a process to closure, the very definition of issue resolution in an Agile-based environment.

The Control phase of a LSS DMAIC project most closely aligns with the following CMMI Generic Practices 2.8 and 3.2:

- GP 2.8 – **Monitor and Control the Process** against the plan for performing the process and take appropriate corrective action.
- GP 3.2 – **Collect Improvement Information**. Collect work products, measures, measurement results, and improvement information derived from planning and performing the process to support the future use and improvement of the organization's processes and process assets.

DfLSS and CMMI Compatibilities

The description given in this article applies only to the LSS DMAIC roadmap and the CMMI MA process area. In order to implement the full connection between LSS and CMMI, organizations need to consider Design for LSS (DfLSS—Define, Measure, Analyze, Design/Build, Verify), generally used to develop new products or processes, as well. While DfLSS is beyond the scope of this discussion, it should be noted that DfLSS can have important implications for all process categories. For instance, CMMI Requirements Management, a project-management process area, entails five specific practices, several of which have direct connections to DfLSS. The most obvious and significant impacts, however, are on the CMMI Engineering category.

- **Requirements Development:** Developing, analyzing, and validating customer/product requirements.
- **Technical Solution:** Goal 1, selecting product-component solutions, aligns most directly with the analyze phase, while Goal 3, implement the product design, aligns with the design/build phase of the DfLSS roadmap.
- **Verification and Validation** directly align with the verify phase of the DfLSS roadmap. Note that certain validation activities are ongoing throughout the lifecycle during define, measure, and analyze [5, 6].

Agile, CMMI, and LSS

"Truth is incontrovertible, malice may attack it and ignorance may deride it, but in the end, there it is."
-Sir Winston Churchill

In "Good to Great" [7] Jim Collins explained it is vitally important for an organization to understand the brutal facts of its environment and its problems, but to never lose faith in the organization's ability to win out in the long term. As he noted, Winston Churchill never failed to confront the most brutal facts. During WWII he created an entirely separate department outside the normal chain of command, the Statistical Office, with the principal function of feeding him—continuously and unfiltered—the most brutal facts of reality. He slept soundly knowing these facts. Recent research defining best organizational practices for project management similarly suggests the optimum way to improve project management is to have the difficult conversations necessary to keep projects healthy. When we maintain a steady culture of discipline, we are able to give our employees more freedom to experiment and find their own best path to results while stimulating change, improvement, innovation, and renewal. Consideration of best practices associated with the integration of Agile, CMMI and LSS concepts within a single project, as opposed to deploying them separately, may well lead to that important culture of discipline.

When viewed holistically, CMMI's ultimate goal (i.e., continuous process improvement) is to cause an organization to become less wasteful, leaner, and more in touch with their actual development progress. Ultimately, both Agile and CMMI, especially in high-trust environments, expect organizations to see gains in productivity by eliminating unnecessary effort. It is true that implementing Agile methods will often eliminate many nonproductive efforts and behaviors at the project level. However, even with Agile retrospectives, what CMMI offers beyond Agile is an infrastructure of organizational learning and improvement that benefits the projects even before they begin [8].

The DMAIC methodology is commonly used to identify problems in a process, measure key data issues of concern, analyze the resulting data, improve the process, and control the future-state process to reduce defects. One of the standard tasks in this methodology is the assessment of process waste, also a core principle of Agile software development. In identifying and eliminating waste in a process, the disciplines of LSS DMAIC and Agile development share many attributes. While Agile practices focus narrowly on improving the software development process, the broad discipline of LSS DMAIC is often used to improve manufacturing and business processes. By highlighting these similarities, the integration of LSS and Agile development, in combination with CMMI continuous process improvement, can lead to that culture of discipline that will allow teams to operate more efficiently while increasing morale, productivity and quality.

Summary

"My greatest strength as a consultant is to be ignorant and ask a few questions."
-Peter Drucker

As organizations truly interested in process improvement mature in CMMI measurement and analysis performance, the relationships between LSS, Agile, and CMMI should be

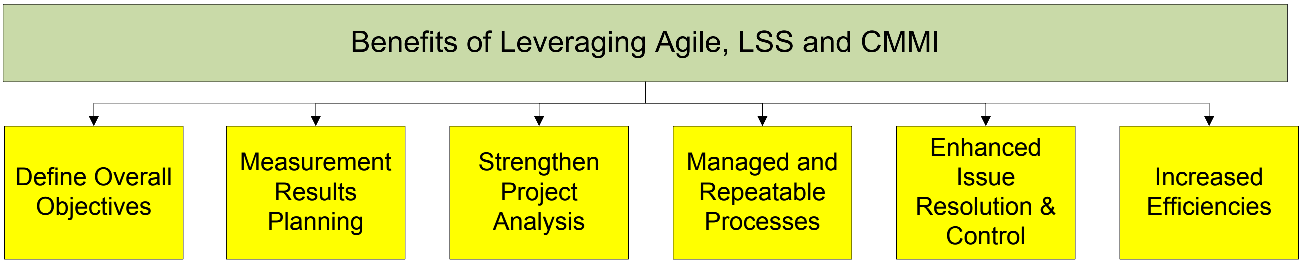

Benefits of Leveraging Agile, LSS and CMMI
Define Overall Objectives
Measurement Results Planning
Strengthen Project Analysis
Managed and Repeatable Processes
Enhanced Issue Resolution & Control
Increased Efficiencies

Figure 4. Process Integration Benefits

understood and leveraged. While a primary focus of LSS is cycle-time reduction and elimination of delays, and Six Sigma targets prevention and remediation of "defects" (in the broadest sense, including cost overruns, schedule delays, etc.), they are in fact highly synergistic and have come to be fully integrated within the LSS framework. Similarly, there are many ways Agile, LSS, and CMMI can be synergistically combined, such as follows.

- **Defining Objectives.** The LSS roadmap approach to establishing overall objectives and identifying potential measures for an improvement project is very similar to the initial CMMI MA practice of tracing business and project objectives to specific measures. The common question of "why" data is collected and analyzed is easily answered in both cases by defining the links to organizational needs.

- **Measure.** The measure phase of DMAIC provides detailed guidance for measurement results planning, data collection, and data integrity. CMMI MA specifies practices for measurement specification, data collection, and storage procedures that include activities designed to ensure data integrity. While LSS designates how these actions should take place, and CMMI leaves the method up to the practitioner (as long as the process is defined and repeatable), the two approaches can be synergistic. Methods such as the SEI Goal-Question-Indicator-Measure (GQIM) process can be used to satisfy both CMMI and LSS approaches to measurement specification, collection, and storage. A version of the GQIM process modified for the Agile-based JIRA tool is given in Figure 5.

The combination of CMMI and use of organizational measurement processes currently provides the "how" and "when" aspects that advocates of LSS infer. Expansion of MA efforts to include the benefits associated with causal analysis resolution and quantitative project management will further this connection.

- **Analyze.** The analyze phase of DMAIC encompasses the activities envisioned by the MA requirement to specify and conduct analysis procedures. While DMAIC provides detailed analysis guidance, and CMMI processes leave such decisions up to the practitioner, relative CMMI MA direction is given within project measurement plans. The CMMI practice of using Operational Definitions for each specified metric helps define the supported goal, collection/storage criteria and simple to complex review techniques. Therefore, any specific instructions and criteria demanded by an LSS application can be easily incorporated into the CMMI MA operational definition framework, and efficiencies inherent to each method will only strengthen project analysis procedures.

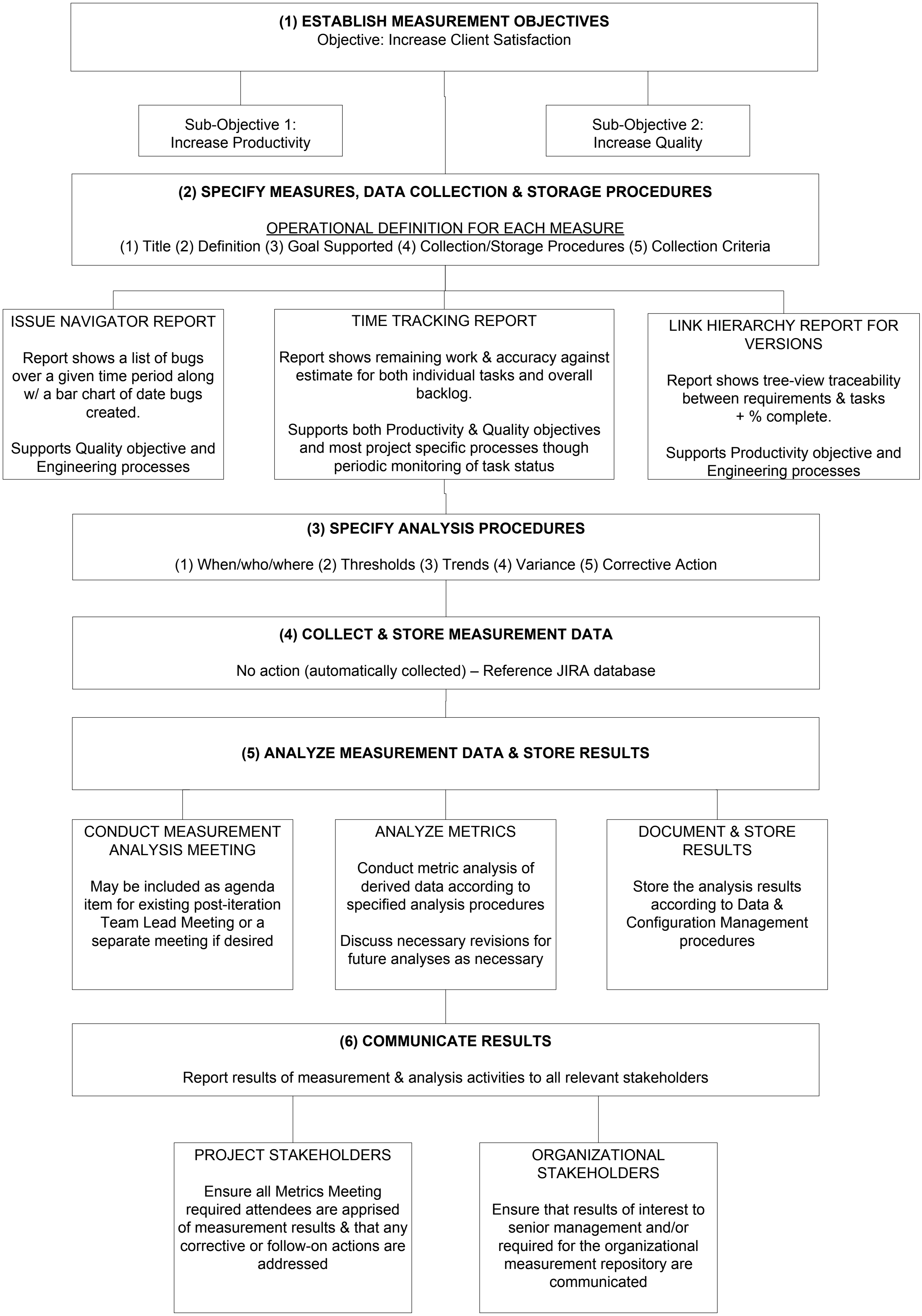
- **Improve.** In general, leveraging both the managed and repeatable benefits associated with MA, and the laser-targeted results of LSS, will provide the organization with tools to achieve the project's established quality and process performance objectives.

- **Control.** Although MA is limited to analysis and communication of results, the higher-maturity CMMI PAs of L4 and L5 can be leveraged to take advantage of benefits associated with causal analysis resolution and quantitative project management. In fact, leveraging these benefits improves MA results—further enhancing organizational tools for achieving established project quality and process performance objectives. The integration of LSS and CMMI provides the opportunity to institutionalize a measurement infrastructure that supports timely response to problems requiring immediate attention and a process to closure—again, the essence of issue resolution in an Agile-based environment.

- **Synergy.** Important connections between Agile and LSS are clear. Both target short lifecycles. What Agile calls velocity, LSS calls throughput, and therefore both attempt to reduce bottlenecks to increase productivity. Both methods are adverse to any activities that do not directly contribute to the final product, such as paperwork (although countless projects that have gone the nuclear option of "no documentation" have lived to regret it).

While not so obvious, there are numerous ways CMMI and LSS can be synergistically combined. Where a CMMI implementation might target the creation of a comprehensive MA infrastructure, an LSS approach would more likely focus on achieving a specific improvement to a particular problem that has a quantifiable (normally currency) near-term benefit—ultimately leading to an infrastructure quite similar to that resulting from a CMMI initiative. While the emphasis is different, with LSS placing greater significance on smaller, shorter (typically 4 months or less) projects with measurable benefits, in the end, aggregate outcomes may be very similar [9].

Agile provides software development methodologies, purposely absent from CMMI. CMMI provides the systems engineering practices (including the process management and support practices) that help deploy and continuously improve Agile methods in a project or an organization, regardless of its size. Unfortunately, project personnel are frequently left out of process design activities and are disinclined or openly skeptical toward the adoption of process improvement activities [8]. This situation is typical of some LSS-style approaches to process improvement as well. Using Agile principles and project personnel input when designing and selecting process activities can create more acceptable and efficient implementations of CMMI, LSS or even Agile itself.



(1) ESTABLISH MEASUREMENT OBJECTIVES
Objective: Increase Client Satisfaction

Sub-Objective 1:
Increase Productivity

Sub-Objective 2:
Increase Quality

(2) SPECIFY MEASURES, DATA COLLECTION & STORAGE PROCEDURES

OPERATIONAL DEFINITION FOR EACH MEASURE
(1) Title (2) Definition (3) Goal Supported (4) Collection/Storage Procedures (5) Collection Criteria

ISSUE NAVIGATOR REPORT

Report shows a list of bugs over a given time period along w/ a bar chart of date bugs created.

Supports Quality objective and Engineering processes

TIME TRACKING REPORT

Report shows remaining work & accuracy against estimate for both individual tasks and overall backlog.

Supports both Productivity & Quality objectives and most project specific processes though periodic monitoring of task status

LINK HIERARCHY REPORT FOR VERSIONS

Report shows tree-view traceability between requirements & tasks + % complete.

Supports Productivity objective and Engineering processes

(3) SPECIFY ANALYSIS PROCEDURES

(1) When/who/where (2) Thresholds (3) Trends (4) Variance (5) Corrective Action

(4) COLLECT & STORE MEASUREMENT DATA

No action (automatically collected) – Reference JIRA database

(5) ANALYZE MEASUREMENT DATA & STORE RESULTS

CONDUCT MEASUREMENT ANALYSIS MEETING

May be included as agenda item for existing post-iteration Team Lead Meeting or a separate meeting if desired

ANALYZE METRICS

Conduct metric analysis of derived data according to specified analysis procedures

Discuss necessary revisions for future analyses as necessary

DOCUMENT & STORE RESULTS

Store the analysis results according to Data & Configuration Management procedures

(6) COMMUNICATE RESULTS

Report results of measurement & analysis activities to all relevant stakeholders

PROJECT STAKEHOLDERS

Ensure all Metrics Meeting required attendees are apprised of measurement results & that any corrective or follow-on actions are addressed

ORGANIZATIONAL STAKEHOLDERS

Ensure that results of interest to senior management and/or required for the organizational measurement repository are communicated

Figure 5. JIRA Measurement & Analysis Process

Conclusion

"Faced with the choice between changing one's mind and proving that there is no need to do so, almost everyone gets busy on the proof."
-John Kenneth Galbraith

To this point I have offered very little in the way of unique thought. Just as I believe that nothing has actually been invented (from the wheel to the iPod), I have simply conducted research, organized and referenced the thoughts of others, and added my opinion derived from my own experience with dozens of projects. But based on my research, I will leave you with one suggestion.

W. Edwards Deming offered 14 key management principles for transforming business effectiveness [10] that were adopted by many American companies hoping to emulate Japanese business success. A number of Japanese manufacturers had applied his techniques widely and experienced theretofore unheard-of levels of quality and productivity. The improved quality combined with the lowered cost created new international demand for Japanese products. Most of these American experiments failed because a framework and corporate culture for integrating the principles did not exist. One prime example is Deming's insistence on all individual performance appraisals being abolished, in order to "drive out fear." This only served to cause fear in U.S. corporate boardrooms.

There are many advocates of LSS who believe that once LSS is in place, projects can simplify CMMI implementation because much of the CMMI work (processes and artifacts) is already done. I would argue the opposite, however, that once a non-prescriptive process improvement framework such as CMMI is deployed, Agile and LSS project methodologies can be easily integrated. Think of CMMI as an empty vessel with bins for continuous process improvement.

Fill the bins with Agile user stories, daily meetings, short lifecycles, and frequent releases. Then apply the LSS roadmap—establishing overall objectives, performance measurement, issue analysis, progress monitoring, and targeted progress goals. The synergy realized, then, enables projects to select the best of Agile, LSS, and CMMI practices, for a cohesive approach to enhance continuous process improvement. ❖

Disclaimer:

CMMI® is registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

ABOUT THE AUTHOR

Peter D. Morris, PMP is a Process Engineering Consultant under contract to SPAWAR Systems Center (SSC) Pacific through Process Engineering Systems. His career, spanning 37 years, has afforded him opportunities to work in both the commercial and DoD/Federal sectors, including several Fortune 500 companies. He has performed as Software Engineer, Electromagnetic Engineer, Program Manager, Director of Operations, and Process Engineer. He has authored numerous technical reports,including publications for the U.S. Army National Training Center (NTC), National Security Agency (NSA), Defense Nuclear Agency (DNA) and the U.S. Air Force Space Command. Mr. Morris has been a major contributor to the CMMI Level 3 for Development certifications at both InnovaSystems International, LLC and SSC PAC. His adaptations of the Goal-Question-Measure (GQM) process have been institutionalized at various corporations, allowing for simplified and automated measurement and analysis programs focused on business goals and continuous Business Process Improvement. Most recently his efforts have targeted BPI for 22 Agile SCRUM projects, deploying Project and Process Management, Engineering and Project Support enhancements resulting in increased velocity, quality and return on investment.

E-mail: pmorrispmp@aol.com

REFERENCES

1. "Six Sigma: SPC and TQM in Manufacturing and Services", Tennant, Geoff, Gower Publishing, Ltd. (2001).
2. "Principles Behind the Agile Manifesto", Agile Alliance, Beck, Kent, et al (2001).
3. "Tearing up the Jack Welch Playbook", Fortune Magazine (July 11, 2006).
4. "Where the Jobs Are: The Right Spots in the Recovery", Time Magazine (January 17, 2011).
5. "Lean Six Sigma and CMMI: Connecting Software Industry Standards and Best Practices", Gack, Gary, Williams, K., Six Sigma Advantage (2006).
6. "Using Six Sigma to Accelerate the Adoption of CMMI for Optimal Results", Siviy, J., Forrester, E., DoD, SEI (2004).
7. "Good to Great: Why Some Companies Make the Leap and Others Don't", Collins, J., HarperCollins (2001).
8. "CMMI or Agile: Why Not Embrace Both!", Glazer, H., et al, SEI Technical Note CMU/SEI-2008-TN-003 (November 2008).
9. "Best Practices Fusion: Lean Six Sigma and CMMI", Gack, Gary, Process-Fusion.net
10. "Out of the Crisis", Deming, W. Edwards, MIT Press (1986).

Upcoming Events

Visit <http://www.crosstalkonline.org/events> for an up-to-date list of events.

12th Annual CMMI Technology Conference
5-8 November 2012
Denver, CO
http://www.ndia.org/meetings/3110/Pages/default.aspx

Software Assurance Working Group Sessions: Winter 2012
27-29 November 2012
McLean, VA
https://buildsecurityin.us-cert.gov/bsi/events/1406-BSI.html

Annual Computer Security Applications Conference
3-7 December 2012
Orlando, FL
http://www.acsac.org

International Conference on Computing and Information Technology
14-15 Jan 2013
Zurich, Switzerland
http://www.waset.org/conferences/2013/zurich/iccit

Technology Tools for Today (T3) Conference
11-13 Feb 2013
Miami, FL
http://www.technologytoolsfortoday.com/conference.html

Strata Conference: Making Data Work
26-28 Feb 2013
Santa Clara, CA
http://strataconf.com/strata2013

Software Assurance Forum - March 2013
12-14 March 2013
McLean, VA
https://buildsecurityin.us-cert.gov/bsi/events/1417-BSI.html

Software Technology Conference
8-11 April 2013
Salt Lake City, UT
http://www.sstc-online.org

IBM Edge 2013
10-14 Jun 2013
Las Vegas, NV
http://www.ibm.com/edge

All I Want for Christmas Is ...

I am writing this column in September, but since you will be reading this during the holiday season; I feel I need write about what I want for Christmas!

First of all, I need a standardized language to program in (notice "need" not "want"—there is a big difference). I have been programming since the late 1960s. Remember the first attempt to unify the myriad of programming languages, PL/I? Back in the 1950s and 1960s, there were two distinct classes of programmers—business and scientific. The scientific programmers had started by using assembly, but most transitioned to Fortran. The business programmers, on the other hand, were almost universally moving from assembly to COBOL. IBM, in bringing out its OS/360 architecture wanted to have a new, unified language that would offer a single programming language for all users. In 1966, the same year that OS/360 was released, the first PL/I compiler was also released. While the language is still used today, it is certainly a niche language for lots of reasons. The language contained elements of both Fortran and COBOL. The Fortran programmers noticed the COBOL features, and considered it a business language. The COBOL programmers noticed the Fortran features, and declared it unsuitable for business programming. The language contained lots of seldom-used features, making the overall language very large. And, in the beginning, it was not known for producing highly optimized object code. All of these issues (and many more) prevented PL/I from ever becoming a unifying language.

Over the years, I have certainly seen other language unification attempts. I was (and still am) an Ada proponent. It was initially offered as a real time and embedded system language, but the current version of the language is object-oriented and general purpose. It has features for both the business and scientific camps. I still teach and use Ada, and still feel that for high-precision or safety-critical systems, it is the best language we have. Alas, for many reasons (some technical, some political) it is now a niche language, also.

We have lots of languages to choose from now—Java, C++, Ruby, Python, Perl, etc. Some are good for large-scale systems, some for scripting; some are more suited for hacking. None have really unified the programming community. I am also quick to point out that a language is just a language—design and requirements doom large software projects much faster than poor language selection. But still, why do I have to go through the same arguments and discussion of what language should be used for every project I consult on?

So, if you can not give me a single programming language, well ...

Secondly, I want a standardized operating system. I "grew up" on UNIX, with occasional journeys on Multics and CTSS, and some GECOS. I also spent some time with CP/M, MS/DOS, VMS, Commodore OS, transitioned to Windows 2.0, 3.0, and beyond. In addition, I have moved through the Mac OS X zoo (Cheetah, Puma, Jaguar, Panther, Tiger, Leopard, Snow Leopard, Lion, and now Mountain Lion). And let us not forget the many, many flavors of UNIX/Linux (Red Hat, SuSe, FreeBSD, etc).

Each of the major operating systems in use today has some really cool features. And there is certainly no serious or significant movement to merge the operating systems, so I will still have to pick and choose which OS to run depending upon what my OS needs are.

But what about the specific needs of the DoD? Oh yeah—we totally forgot about those who need a Real Time Operating System (RTOS). In which case, none of the above are sufficient, and you have to choose from LynxOS, OSE, QNX, RTLinux, VxWorks, Windows CE, etc.

If I can not have a standardized programming language or a standardized operating system, then ...

The third item I would like for Christmas would be a single design methodology. I have been through flowcharts, Program Design Language, Structured System Design, Systems Analysis and Systems Design, Hierarchical Input Process Output charts, Data Flow Diagrams, Control Flow Diagrams—just to name a few. Rather than elaborate with more acronym soup, let us just shorten this paragraph. I have the CMMI®. And, of course, I have various agile methodologies to use, too. And I have UML. One is a methodology or a touch-stone for measuring my maturity, one is a type of methodology, and one is a design language.

While I find UML a wonderful tool for some aspects of design, it is not the notational tool for multiple languages that I had hoped for years ago. And, as for the CMMI and Agile methodologies—let us face it—the much maligned waterfall model is STILL used as the basis for a huge amount of the software development throughout the DoD, the U.S., and the world.

And yet we survive. We somehow manage to get high-quality and mission-critical software delivered to the people who need it—sometimes on time, sometimes within budget, and sometimes with high quality.

It is the end of 2012. I do not have a standard language. I cannot standardize the operating systems. And my design modeling language cannot yet get me all the way from initial design to full code. And the mission-critical code still needs to be delivered on time, within budget.

Should make 2013 an interesting year.

David A. Cook, Ph.D.
Stephen F. Austin State University
cookda@sfasu.edu

CMMI® is registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

NAVAIR Vision Statement:

"Sailors and Marines, Armed and Operating with Confidence"

Because we develop, deliver, and sustain aircraft, weapons, and systems—on time, on cost, with proven capability and reliability—so they cost effectively succeed in every mission and return home safely.

NAVAIR Goals:

Current Readiness: Contribute to delivering Naval Aviation Units Ready for Tasking with the right capability, the right reliability and safety, in the fastest possible time, and at the lowest possible cost.

Future Capability: Deliver new aircraft, weapons, and systems on time and within budget that outpace the threat, provide global reach and persistence, support AIR-SEA Battle, Joint and Coalition Operations, and meet the required adaptability, reliability, safety and total lifecycle costs.

People: To institutionalize a culture of learning, innovation and exemplary leadership that is warfighter focused, motivated and inspired—that leverages diversity, technology, analytics, transparency and accountability for a dynamic, agile and adaptive World Class workforce.

NAVAIR

NAVAIR Process Resource Team (PRT)
(760) 939-6226

CrossTalk thanks the above organizations for providing their support.