Metrics for the Cost of Proprietary Information Exchange Languages in Intelligent Systems

John Horst
NIST
100 Bureau Drive / MS 8230
Gaithersburg, MD 20899
1.301.975.3430
john.horst@nist.gov

Nathan Hartman
Purdue University
401 Grant Street
West Lafayette, IN 47907-2021
1.765.496.6104
nhartman@purdue.edu

George Wong
The Boeing Company
5301 Bolsa Ave. / MS H012-B438
Huntington Beach, CA 92647
1.714.372.1917
george.s.wong@boeing.com

ABSTRACT

The increasing number of intelligent software components is accompanied by an increasing number of proprietary information exchange languages between components. One of the challenges for the smart technology worker is to achieve intelligent system component interoperability, at the lowest cost possible, without sacrificing the freedom to choose from the entire spectrum of current and future software product offerings. This is best achieved when correct, complete, and unambiguous information exchange standards are implemented in vendor products worldwide. If this is the common sense solution to information incompatibility costs and risks, why is standards-based interoperability so rarely seen? One reason is that a required investment in standards must precede the savings gotten from true interoperability. Corporate management is commonly reluctant to commit to this investment, partly because there appears to be no published set of interoperability cost metrics which technology workers can employ to make an evidence-based business case. This research seeks to remedy this situation by defining realistic, comprehensive, and sector-independent cost-risk metrics.

Categories and Subject Descriptors

D.2.8 [**Information Systems**]: Metrics – *process metrics, complexity metrics.*

D.2.12 [**Software**]: Interoperability – *interface definition languages.*

K.1 [**Computing Milieu**]: The Computer Industry –*standards.*

General Terms

Measurement, Performance, Economics, Standardization, Languages.

Keywords

Interoperability, information, information exchange languages, intelligent systems, intelligent system components, return on investment, standards, information exchange standards, standard languages, interoperability costs, metrics, interoperability metrics, interoperability cost metrics, interoperability risk metrics, proprietary exchange languages.

PerMIS'10, September 28-30, 2010, Baltimore, MD, USA.

1. INTRODUCTION

Intelligent system performance is not maximized by smart system design and manufacture alone, but also by a smart technology infrastructure and a smart system lifecycle, and to achieve the latter includes the interchangeability of system components from multiple vendors, *i.e.*, system manufacturers and end users are able to swap in and out any component of the system from any component vendor worldwide with minimal cost or risk.

1.1 Illusory Interoperability

Component interchangeability is related but not identical to the plain meaning of interoperability: "the ability of two or more systems or components to exchange information and to use the information that has been exchanged." [9]

A common path taken to achieve interoperability (by this definition) is to build systems consisting of components made from a single vendor. Such systems are generally interoperable[1], but not typically interchangeable with components from other vendors[2]. Single vendor mandated interoperability limits freedom of component choice, incurs risk of vendor viability and corporate mergers, and commonly pushes information translation costs down to suppliers and vendors, who pass the cost right back to the end user in the form of higher component prices. Little is saved and much is risked. [12] It is an illusory interoperability. [6]

Furthermore, single vendor mandates generally do not deliver the interoperability as planned due a variety of factors, including vendor volatility, corporate mergers, and acquisitions. Enter language translation. With translation, systems consisting of components made from multiple vendors now become "interoperable," as well as interchangeable, albeit at a cost, since translation incurs labor costs, information quality losses, license fees, and training costs. To make matter worse, the number of translators required for interoperability increases multiplicatively with respect to the number of component vendors at each interface in the system[3]. This too is illusory interoperability.

---

[1] The infamous Airbus A380 interoperability failure is a notable exception: different versions of same vendor's software were not interoperable, leading to billions of dollars of losses. 4

[2] Reference to specific commercial vendors and their products does not imply endorsement by the authors or their affiliate organizations.

[3] If there are M interfaces in the system and n_m component vendors on both sides of the m^{th} interface ($m \in \{1,2,\dots,M\}$), then the system requires roughly $\sum_{m=1}^{M} n_m(n_m-1)$ translators.

1.2 Interoperability and Communication

Some have defined interoperability as "the ability of systems, units, or forces 1) to provide services to and accept services from other systems, units, or forces and 2) to use the services so exchanged to enable them to operate effectively together." Part 1 of this definition is called "technical interoperability" and part 2 is called "operational interoperability." [2]

The distinction between technical and operational interoperability is an analog to the distinction between properly encoding and decoding a message, since 1) interoperability is a measure of the degree that a sender correctly encodes a message and the receiver correctly decodes that message and 2) the ways we measure the correct operations of sender and receiver are not the same.

Here's how we measure whether a message is **encoded** correctly. Since correct syntax and semantics are both measurable[4], correct encoding can be measured directly, without observing system behavior. Of course, a precisely defined language must exist to define correct encoding.

Here's how we measure whether a message is **decoded** correctly. There is no other way of knowing whether the message was correctly decoded except by observing the "operation" of the message receiver. Of course, the receiver must decode according to the language known and used by the sender, perhaps through a translator, though that adds unnecessary cost.

"Technical interoperability" is then "the ability to correctly and completely encode and deliver a message" and "operational interoperability" is "the ability to perform the correct action inherent in the message." This principle is encapsulated in the statement, "information is as information does." [10]

1.3 Standard Languages

With these ubiquitous costs and risks inherent in so-called "interoperable" systems, it is therefore preferable to redefine interoperability.

> Interoperability is the state of a system wherein the cost of attaining and maintaining correct encoding and decoding of the information exchanged between component pairs is minimized, and minimum cost is attained only when each component pair conforms to a correctly defined standard language, correctly and widely implemented.

Simply put, interoperability is optimized when a single apt language, and none other, is encoded and decoded at each component interface in real systems worldwide. [6] "Correctly defined" standard languages are

- Correct, complete, unambiguous, and timely
- Developed within a standards-generating organization ensuring IP protection, low implementation cost, and open participation

"Correctly implemented" means

- Conformance and interoperability tested
- Certified via conformance tests
- Widely implemented
- Purchased by end users only if certified

[4] However, measuring semantics can be much more difficult than measuring syntax. Shannon identified but ignored semantics in defining his communication theory. [11]

Standard languages for component-to-component information with standards-certified implementations maximize interoperability since interoperability success is directly linked to successful component-to-component communication, and successful communication is dependent on well-defined languages and correct encoding/decoding. Widely implemented and correctly defined standards eliminate all the costs associated with sub-optimal interoperability.

When standards do not exist, we do not have the freedom of interchangeability, or else we must pay extra for it. For example, anyone in a household with multiple cell phones knows well that when he/she misplaces or damages the power charger to his/her phone, the power chargers for other phones in the home typically do not work with his/her phone. Either he/she has to spend time searching for his/her charger, pay for a new one, or go for a time without access to his/her cell phone – all achieving "interoperability" at a cost. If there were a cell phone power charger standard, well defined and implemented worldwide, each charger in the home would work with all other phones – and this is not as elusive a goal as it may at first appear.

1.4 Measuring Interoperability

Using our definition of interoperability, measuring interoperability must be directly related to the ability of one system component[5] to encode and the other system to decode in the context of a language. The language must be well-defined and well-implemented by both the encoding and decoding components. Syntactical rules and a dictionary are all that are required to measure **encoding** success. Measuring component behavior is all that is required to measure **decoding** success.

To the degree that the language definition and its encoding and decoding fail, interoperability fails. So, standard language definition and its encoding and decoding are directly proportional to this quality we are calling interoperability.

It appears that measuring interoperability directly is hard, [3] because it is hard to define (and apply) a metric for syntactical and semantic communication. Happily, measuring **the cost** of less than perfect interoperability (by our definition above) is quite a bit easier than measuring interoperability itself [3] and more importantly, it has greater pragmatic usefulness at helping achieve better interoperability in real organizations.

1.5 Return on Investment

Knowing the cost of interoperability failures and risks in an organization is not the only cost needed. Also needed is the cost of generating, implementing, and supporting standard interface languages. The difference between these two costs has been called the "Interoperability ROI (return on investment)." The cost of developing standards is addressed in earlier work. [5][6]

2. Defining and Measuring Interoperability Costs and Risks

Defining, implementing, and maintaining standard languages between all software systems will generate huge savings of cost and quality. However, measuring that cost savings is difficult for these reasons:

[5] A "system component" can be anything from a temperature sensor to a nation state.

- Businesses do not have a solid understanding of the nature of "interoperability costs"
- Savings due to benefits of mandated standard languages are considered to be a corporate secret
- Costs due to interoperability lapses are an embarrassment, so companies are reluctant to measure these costs

We suggest a new way to address this problem:

- Define interoperability costs metrics
- Describe a process for efficiently and cheaply applying those metrics to get accurate cost numbers to corporate planners

3. The Process for Applying Interoperability Metrics

3.1 Define Organizational Scope

Determine the precise scope of the organizational entity targeted for application of interoperability cost metrics. The entity might be an enterprise, a factory, a division, a work cell, or a manufacturing sector, where a sector might include any or all of the following: enterprise resource planning, product lifecycle management, product design, process planning, machining, assembly, quality control, measurement analysis. End user organizations must include operations of all tier suppliers and product vendors within the scope. Tier suppliers must include operation of all product vendors within the scope. Product vendors need only determine the scope in their own organization.

3.2 Define Activity/Component Diagram

End user organizations should develop an activity diagram for each organizational entity. Each activity in the activity diagram must be performed by one or more standalone system components corresponding to real products in the market (otherwise there is necessarily no interoperability problem at that interface). Depending on the organizational entity, there may be duplicate activities in the entity, for example, transfer of information is known to be common between design activities in one part of an entity and design activities in another part of the entity, e.g., CAD-to-CAD, requiring CAD-to-CAD translation/validation.

If the organizational entity is a tier supplier, or if tier suppliers are part of the end-user organizational entity, develop detailed activity diagrams for all tier supplier operations supporting the scope of the organizational entity.

Vendor software/systems entities should identify all activities for which there are software/system products in the (vendor) organizational entity under active support

3.3 Identify All Interface Languages

3.3.1 End users and tier suppliers

Identify all languages, either proprietary or standard, actually in use in the organizational entity at each of the interfaces between activities where information is known to be transferred. In manufacturing, common activities include enterprise resource planning, product lifecycle management, product design, process planning, product machining, product assembly, quality control, and results analysis

3.3.2 Vendor software/system organizations

For each supported product, identify all languages supported, either proprietary or standard, via the maintenance of translators, either internally or via 3rd party

3.4 Prioritize interoperability cost measurements

Assign high priority to interoperability costs suspected to be higher than other costs and whose cost information can be more easily/cheaply gathered. Generally CAD-to-CAD translations and validations are more costly than translating CAD for downstream applications. These is no need to gather cost data for all cost categories, since the requirement is to persuade management of the magnitude of the problem, without spending a lot of time and money gathering the cost data.

3.5 Measure actual interoperability costs

If there are multiple languages at any of the interfaces identified in the activity diagram, translation is required. To keep the expense of measuring interoperability costs low, some measurements may need to be estimated. For those values estimated, if will be necessary to seek and find realistic estimation error values. All costs related to translation and validation can be considered direct interoperability costs.

Costs such as translation and validation must occur when there are multiple languages at a single interface, since an organization must move its information. We consider translation and validation unnecessary costs in an organization, because neither translation nor validation is needed where there exists a single widely-used information standard that is correct and complete. Therefore, translation and validation are considered "interoperability costs" and not considered merely "the cost of doing business." Validation is done after translation and attempts to ensure that the translation was done correctly. If validation is not performed, then clearly only translation costs apply. Here is the list of (direct) costs:

- Information translation/validation of both format and meaning
  - Translation/validation task labor: the cost of each translated/validated file (or cost per average data rate) and the time it takes may be computed from file sizes (or average data rate)
  - Translation/validation software license fees
  - Reduced engineering software usage skill due to multiple platforms
  - Non-optimal software usage with downtimes due to multiple vendor support
  - Translation/validation software product development [vendor cost]
  - Translation/validation software execution labor, which might be gotten from bytes of data translated/validated
  - Unnecessary system maintenance and software training
  - Unnecessary software development
  - If translation/validation is necessary but not performed because it is too costly,
    - loss of expert knowledge in those files
    - reprogramming costs
    - new software purchase costs
  - Proprietary license fees for "direct CAD interface" to downstream activities, such as manufacturing process planning or supplier bidding
  - Information access fees, e.g., Product Manufacturing Information (PMI) with CAD
  - Manual or automatic editing of information due to lost information quality (erroneous syntax or meaning)

- o Quality check and repair healing software licenses
  - o Increase support (staff and equipment)
  - o Manpower (time) for translation and support of the additional systems
  - o The cost of storage of actual amount of translated information
- Storing and maintaining information in more than one proprietary version (keeping the information up-to-date)

3.6 Indirect costs

Certain costs of interoperability, unlike translation for example, are not directly related to interoperability, but can be quite substantial. [1][12] These costs are real, but are the indirect consequence of translation/validation errors:

- Information quality degradation due to translation/validation errors, either from omission or commission, leading to decreased product quality; must check cost to all downstream activities affected by the information quality degradation
  - o Diminished customer product/service confidence
  - o Lost contracts
  - o Contract penalties
  - o Reduced perception of quality
  - o In-warranty service/replacement
  - o Functional failures in product traceable to translation/validation errors (versus traceable to design or manufacturing errors) leading to the following costs
    - ▪ Property damage, injury, and legal fees
    - ▪ Reduced product/service image causing revenue losses
    - ▪ Product recall
    - ▪ If there is vendor-software incompatibility due to a merger or acquisition, end user must pay either to
      - choose and implement new single vendor enterprise-wide or
      - pay for all interoperability costs, such as translation, validation, fees, training, etc. (listed above)
- Increased product development times
- Reduced perception of quality
- Uselessness of long term stored proprietary information
- Profit/market loss due to lack of freedom to choose best-in-class
- Restraints on corporate or technical agility
- Reduced product and process innovation

3.7 Identify and estimate cost risks

The following risks are potential costs, each of which must be traceable to interoperability failures:

- Vendor corporate failure (scandal, mismanagement, poor economy)
- Lost contracts
- Contract penalties
- Increased product price due to interoperability costs
- Competitive disadvantage leading to
  - o Reduced profits
  - o Loss of market share due to
    - ▪ Customer dissatisfaction with high cost
    - ▪ Competitors' lower product cost
  - o Indecision due to interoperability issues and complexities
  - o Lost opportunities
  - o Inability to capitalize on revolutionary technological breakthroughs by other vendors

3.8 Single language mandate costs

If there is a single vendor language mandate anywhere in the organizational scope under test, the same cost areas mentioned in the last two sections will also be suffered by companies down the supply chain, which may not be in the organizational scope. One cost commonly unique to a single language mandate environment is higher fees due to reduced competition, which should also be measured.

4. Conclusion

We have made the argument that to define and implement interoperability cost metrics against the ideal, *i.e.*, complete, correct, and unambiguous interface language, widely and correctly implemented by components worldwide, should help greatly to mitigate information exchange incompatibility (interoperability) problems, since interoperability is directly and inextricably related to correct and complete communication between a sender and receiver, so the proliferation of multiple, redundant (usually proprietary) languages (consisting of both format and meaning) is the heart of the problem.

Managers and intelligent systems workers commonly have the problem of convincing corporate executives and funding sources, of the magnitude and seriousness of the interoperability problem, and ultimately that support of language standards is of fundamental importance, since such standards are the only optimal way to eliminate the many costs that have been identified in this research.

The following work needs to be done which, along with the cost metrics already defined, will help workers easily and accurately determine the cost of interoperability in their enterprise.

- Describe where and how to do cost data collection (i.e., "sensor" placement) to (preferably) automatically collect cost data
- Give guidance on cost and risk uncertainty estimation
- Conduct pilots on the application of the metrics, the process, and cost collection
- Revise and augment cost and risk details based on results from pilots
- Publicize these interoperability cost measurement tools in appropriate venues

5. REFERENCES

[1] Bartholomew, D., PLM: Boeing's Dream, Airbus' Nightmare. In *Baseline* (February 5, 2007)

[2] Kasunic, M., Andrew, T., Ensure Interoperability. In *DoD Software Tech News* (June 2010) Vol 13, No. 2

[3] Kasunic, M., Measuring Systems Interoperability: Challenges and Opportunities. Software Engineering Institute, Carnegie Mellon University, (2001)

[4] DoD Directive Number 4630.05. (May 5, 2004)

[5] Hartman, N. Using metrics to justify interoperability projects and measure effectiveness. In *Proceedings of the*

Collaboration and Interoperability Congress. (May 2009). http://www.3dcic.com.

[6] Horst, J., Hartman, N., and Wong, G. Defining quantitative and simple metrics for developing a return-on- investment (ROI) for interoperability efforts. In *Proceedings of the Collaboration and Interoperability Congress.* (May 2010). http://www.3dcic.com.

[7] Horst, J. Update on Quality and Metrology. In *Proceedings of the Collaboration and Interoperability Congress.* (May 2010). http://www.3dcic.com.

[8] Horst, J. Reduce Costs and Increase Quality with Information Exchange Standards for Manufacturing Quality. In *CMM Quarterly Special DMSC Edition*, (2009) http://www.cmmquarterly.com/ezine/current-issure/special-edition

[9] Institute of Electrical and Electronics Engineers. *Standard computer dictionary: a compilation of IEEE standard computer glossaries*, (New York, 1990)

[10] McKay, D. M., Information, Mechanism, and Meaning. *The MIT Press*. (1969)

[11] Shannon, C.E. A Mathematical Theory of Communication. In *Bell System Technical Journal*. 27 (July, October, 1948) 379-423, 623-656

[12] Tyrka, K. Interoperability: Tying Industry in Knots. In *Global Design News*. (February, 2000), 58 – 62.