Mitigating the Risks Associated with Complex IT Projects

Every IT project failure occurs as a result of poorly managed and complex change requirements that costs today's Chief Executives time, money and value to the organization.
In order to mitigate these risks, IT Leaders must move beyond traditional practices and employ a new simplification methodology that enables continuous prosperity of IT investments.

Dr. Mykhailo Magal, PhD., Author

Table of Contents

Introduction

What does it mean to successfully deliver an IT project, without failure?

Timely delivery of IT projects is becoming one of the greatest challenges of the 21st Century for both commercial and governmental organizations. Organizations need assurances that key project deliverables are achieved on time, on budget, and within expected service levels required required by the business. Unfortunately, IT projects continue to fail.

According to the 2009 U.S. Budget, a not too optimistic trend shows that the ratio of "at-risk" IT projects had been were steadily climbing between 2007 and 2009 , from 30.6% in 2007 to 43.3% in 2008 to 66.0% in 2009. This is further confirmed by more current events such as the cancellation of a major healthcare project by the Government in the of Ontario Government. The E-Health Project failed to deliver "value for money" [2]. On a macro scale, these numbers are do not improve. The term "IT Project" is a significant search inquiry in Google and when adding the word "Fail", 75% of "IT Project" in search inquiries suggests IT Project Failures are continuing to be a serious global concern.

These numbers point to the need for to identifying why projects fail and the need for a better model of mitigating risks of failures in complex IT projects and quantifying the impact they have on business performance and customer experiences. The findings in this whitepaper are based on the consensus of published data that supports our empirical experience gained through years of developing and managing complex IT systems.

This research promotes a unique approach, based on a mathematically supported methodology, for mitigating complex IT project failures and presents the root-cause reasons of the failures, as well as a change-measuring model to enable systematic monitoring and forecasting risks of IT projects in real time. This document identifies the key imperatives for driving project success, governance requirements and new management approaches that are necessary for successful delivery of complex IT projects.

What is a Complex IT Project?

According to our definition, a Complex IT Project is a unique system of consistent and repeatable processes and/or sub-projects, which are logically connected and aligned to achieve controlled business results. In this document, we will expand on the definition of the Complex IT Project and share observations of an approach on how to eliminate Complex IT Project failures, thereby providing new assurances that key business deliverables can be achieved on time, on budget, and within expected service levels.

For simplicity, an IT project which is considered to be complex, is defined as one with a budget that exceeds CAN$10 Million. However, these approaches and methodologies can be applied to IT projects with budgets of $1M to $10M as well.

It is important to distinguish the relevance and use of the terms complex and complicated system [3]. A Complicated system is one that is very difficult for a non-trained person to understand. In contrast, a complex system has nonlinear interactions within or between the components. These nonlinearities compound across the entire system in such a way that the entire system-level behaviors are more difficult or potentially impossible to predict, even with the knowledge of the components. These interactions impact the entire system-level behaviors, colloquially, to emerge from the underlying component interactions. As a result, non-linear systems are often studied through use of simulations.

Why are IT Project's Becoming More Complex?

Today, both business and government institutions strive to automate increased amounts of their business processes, in the quest to reduce the cost of services and to better enable the line of business.   This enables new revenue and growth, quicker time to market, and the ability to globalize business with minimum investment. But most of all, to ultimately serve buyers, customers or citizens' needs better and faster in order to meet their demanding expectations.

The relevant requirements of the external environment (partners, customers and other stakeholders) are a considerable contributor to the complexity of today's IT projects. As a result, key project and service owners are required to possess a wider multi-disciplinary knowledge base, with stronger project skill sets in order to effectively perform in the fast-paced environment. It requires a broader awareness and substantial intelligence to mitigate the complexity of new or existing IT systems, especially legacy ones.

Why Are Complex IT Projects Continuing to Fail?

Why do complex IT projects appear to be bound to fail all the time? There is a consensus of published data, [4] [5] [6] [7] [8], that supports our empirical experiences, indicating that there are more fundamental reasons that are not only technical in nature - they lie in the following four domains:

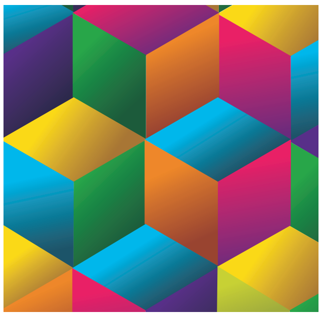
- ever increasing complexity of IT ***system design*** being implemented within a project;
- a flawed ***governance*** structure that prevents:
    - the setting of robust business and resource quality requirements,
    - unrealistic project and budget timelines
    - utilization of the necessary experience that aligns to the project;
- poor **change management** structure that disables the adaptability of the project.
- the lack of a **systematic model** to monitor and forecast risks in real time.

According to research published in the Harvard Business Review [9], 67% of organizations failed to terminate unsuccessful IT projects due to flawed governance and 34% of organizations undertook IT projects that were not aligned with corporate strategy.

This whitepaper promotes an approach to solving the complex IT project problems in these domains and is discussed in subsequent paragraphs that focus on the proposed Six Key Imperatives for Driving Success of Complex IT Projects.

Furthermore, we submit that an Independent Verification and Validation (IV&V) process, is essential as part of the IT quality assurance and testing and when used in conjunction with internal staff to “self test”, falls short of the discipline and rigor that must accompany proper delivery practices due to potential worker’s self-preservation, poor quality assurance skills and most of all, scheduled time.

Why is this white paper important to your role in IT?



Google the term “Complex IT Project Fail” and your search inquiry will generate 5 times more results than the term “Complex IT Project” without “Fail”. This suggests that 80% of complex IT projects are being searched in relationship to failures and that failure of Complex IT Projects is continuing to be a serious global concern.

As you review these results, you will note validation from various authorities and observers, all of which share their perspective of what a Complex IT Project is and each invariably, point to an example of a large Complex IT Project that has failed.

Over the years, numerous examples were published by industry leaders who are quick to share with the reader a sad story of an IT project that failed. Many will cite that by virtue of the project being “complex”, it was destined to fail. Others will recount the project’s poor management and *speculate that the reasons were primarily beyond control.*

The casualties are not specific to any industry sector or government entity. Retail, Banking, Insurance, Universities and even Tier One “Vendors” themselves were all on the receiving end of a Complex IT project that failed.

HBR global analysis and published casualties

According to Harvard Business Review, [9] the analysis of 1,471 IT projects from across the globe, revealed little difference between IT projects at government agencies and commercial companies.

The dollar amounts from these failings, run into hundreds of millions of dollars, as the impact for many organizations was that they suffered not only financial losses, but lost business/customers/loyalty and revenue.

Only a few examples of such casualties are:

Ontario E-Health Record Project that failed to deliver “value for money for a_$1 billion investment” according to the Ontario Auditor General, Jim McCarter's report, [2],

US Expeditionary Combat Support System of USD$1 billion, designed to save money by combining hundreds of existing computer systems, has been deemed a failure. Sen. John McCain described it as ‘one of the most egregious examples of mismanagement in recent memory’, [10],

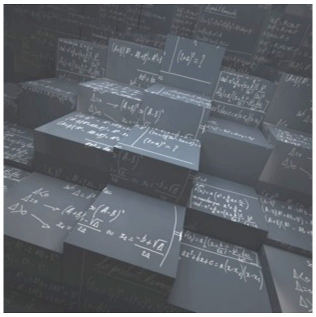
California MyCalPAYS Project of USD$371 million, the "largest payroll modernization effort" in the nation, has now been placed on hold while State officials conduct an assessment to determine whether "any of SAP's work" can be used as part of an ultimate solution, according to a statement from Controller John Chiang, [11]

California’s DMV(Department of Motor Vehicles) IT Modernization Project of USD $208 million, failed to upgrade legacy hardware systems for the DMV's driver's license system and vehicle registration system, [12]

Sainsbury's Warehouse Automation Project of GBR£150 million to install an automated fulfillment system for the British supermarket giant, the system was scrapped after four years after its installation, [8]

These examples point to the need for a more viable strategy and methodology of mitigating risks of failure in complex IT projects.

Introducing (SIP Simplified Iterative Partitioning) Methodology - Measuring IT System Complexity



The simplified formula for calculating the IT system's complexity is given by Robert Sessions [7] as a mathematical equation, which was derived from Glass's Law [13], that states that when the functionality of an IT system increases by 25% percent, the complexity of that system doubles (100% increase). In other words, if a system increases in size from 40 business functions to 50 business functions, the complexity of the system actually doubles. The most important observation is that it doesn't matter what the business functions are or how they are to be implemented.

Briefly stated, SIP-methodology functions as follows:

Firstly, a system is deconstructed into a collection of low-level **Business Functions**. These business functions are then analyzed and placed in subsets based on synergistic relationships. Two atomic business functions F1 and F2 are considered to be synergistic if, from the business's perspective, F2 is not useful without F1 and vice versa. These subsets are then treated as **sub-systems** and/or **sub-projects**. In a service-oriented architecture, these become the definitions for services.

SIP is a pre-design methodology for partitioning set of subsets, in this case, for partitioning business functions into sub-systems (services) such that the overall system has the least possible complexity required to solve the business problem and minimize system failures.

Complexity of any IT system is being measured in SCUs (Standard Complexity Unit), which stands for the amount of complexity that would be found in a system implementing one single business function with no outside dependencies. SCU tells us little about its absolute complexity; however, it helps us to calculate the relative complexity of the system with different numbers of business functions and connections.

(SIP Simplified Iterative Partitioning)

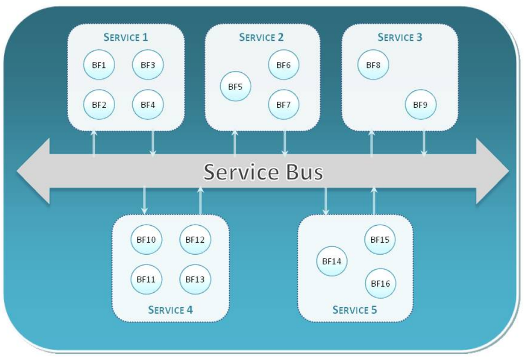


Figure 1. Simplified Service Oriented Architecture consisting of synergistic atomic business functions (BF) partitioned into services that interact over Service Bus.

If we assume that a model system is an SOA (Service Oriented Architecture) consisting of business services, according the SIP-methodology, there are three contributors to the SOA complexity: 1st - the number of business functions in each service, 2nd - the number of connections between business functions, and 3rd - the number of services in the SOA (see Figure 1).

For the purposes of brevity in this document, we do not detail the specific mathematical calculations, nor attempt to demonstrate the significant formulas from them.[7]. We are focused on the fact that: the complexity of the IT system can be calculated and optimized to the least possible complexity, i.e. to the least possible failure rate.

SIP is a methodology of simplification and risk mitigation that is grounded in mathematical models and that uses mathematical rules to create verifiably optimal architectures that can be validated in terms of minimizing complexity. This is based on a probability theory, set theory, and equivalence relations; and brings rigor and a formal discipline to pre-designing IT systems to reduce project complexity before designing them.

Note: You may prefer to use one the four identified methodologies or other frameworks, provided you apply a component of SIP to reduce IT system complexity.
This will empower your existing methodology and enable delivery even greater benefits as intended.

The SIP methodology focuses on the issue of controlling complexity in a structured manner that seamlessly complements the entire project architecture, and is not intended to replace any of the four major methodologies in enterprise architecture which are:

- Zachman Framework for Enterprise Architectures,
- Open Group's Architecture Framework (TOGAF),
- Federal Enterprise Architecture (FEA),
- Gartner Methodology.

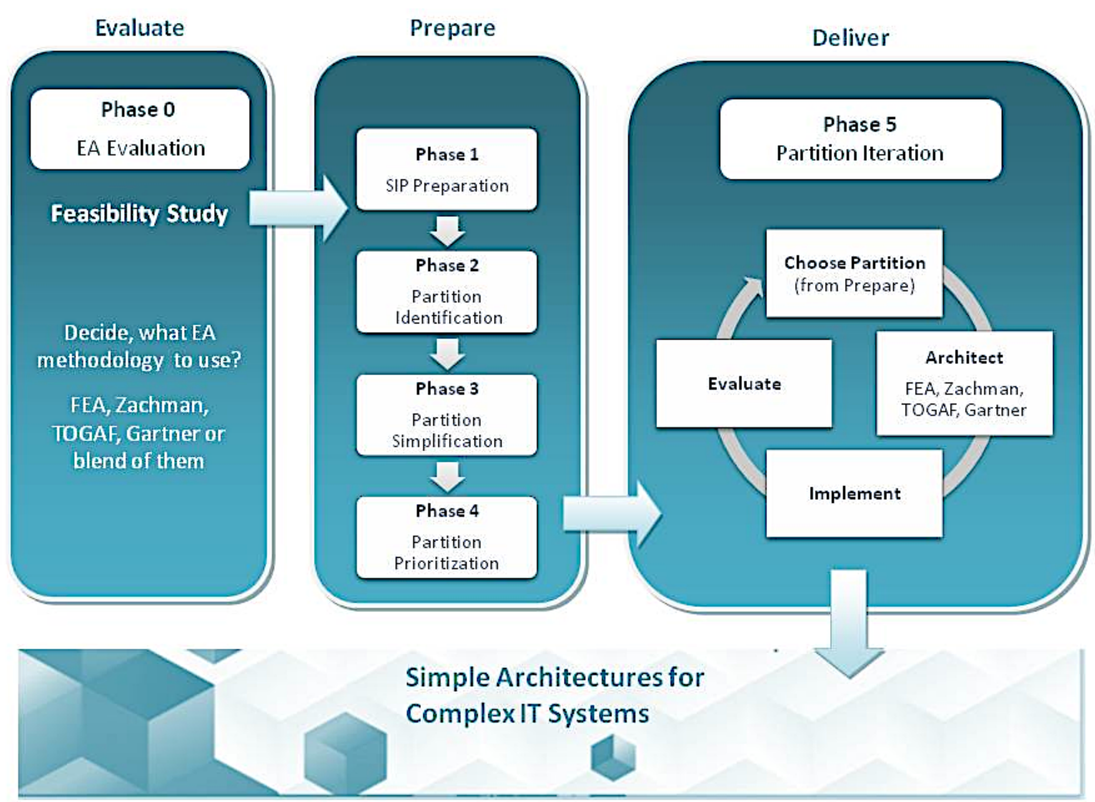
(SIP Simplified Iterative Partitioning)

As it can be seen in Figure 2 below, the SIP process consists of six phases that can be divided into three groups as follows:

Evaluate: an organization evaluates its readiness, and conducts a study whether it is feasible to leverage an enterprise architecture (EA) as well as, determining the SIP-methodology well suited for it's needs.

Prepare: the group includes the identification, simplification and prioritization of the enterprise partitions by leveraging the mathematical models for complexity and synergistic partitioning.

Deliver: the group includes an iterative process of designing, implementing and delivering simple architectures that drive maximum value for the minimum cost.



Mitigating IT Project Failures = Technical Debt relief

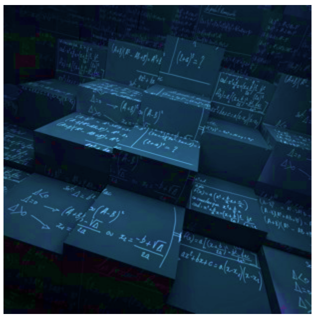
As we expressed earlier in this whitepaper, according our definition, a Complex IT Project is a unique system of consistent and repeatable processes and/or sub-projects, which are logically connected and aligned to achieve business results required.

Considering an IT complex project as a system, we can seamlessly expand the SIP methodology approach on how to partition complex project processes and sub-projects to achieve the least possible complexity of the project. This could apply a rigor mathematically grounded methodology on the phase of initiation of an IT project well before its planning and *dramatically reduce risks of the project failures* on its other phases such as planning, executing, monitoring & controlling and closing.

Such an approach to initiation of IT projects would bring compelling financial rewards to the project, essentially contribute to efficiently solving the business problem and dramatically mitigate typical IT project failures, including budget overruns, late delivery and poor project governance today known as Technical Debt:

- Divide your complex project into simple sub-projects that correspond to the identified sub-sets of business functions. In SOA architecture each sub-set is being considered as service.
- Define requirements to human resources for each sub-project and project on the basis of their complexity rates.
- Evaluate budget of each sub-project. Ideally suited if it does not exceed 10 Million dollars.
- Plan and implement sub-projects and integrate them into deliverables of the complex project. In an agile and iterative fashion.
- Continuously review and validate the IT project alignment with business strategy to follow ongoing business needs in real time.

System Cost, Failure Rates versus IT Complexity



According to independent research [7], there is an almost perfectly linear relationship when one must plot complexity against system failure rates.

As previously stated, if one can measure the degree of system complexity, then you can predict the impact a change has on the system's failure rate with increasing accuracy.

Fortunately, as expressed, ***IT system complexity can be measured***, such that, the change of potential system failure rates can be predicted well before the project's start. As one of the aimed results of our research, we developed a **Change Measuring Model** that facilitates a systematic approach to monitoring and forecasting risks of IT projects in real time. For example in below Figure 3, the graphical dependence of Complexity vs. Number of Business Functions (BF) derived for the sample IT system, shows the impact of up to 30 BFs and up to 30 connections (dependencies) (blue solid line).

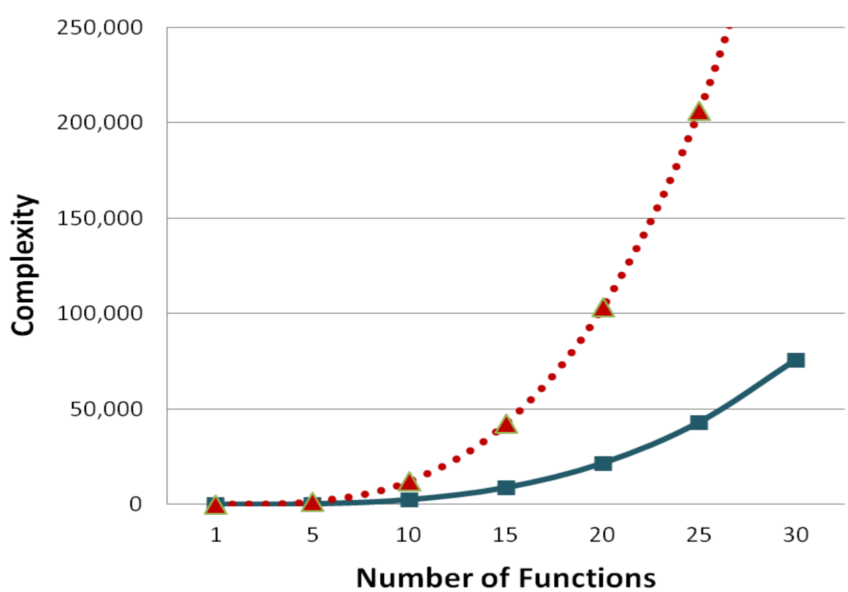
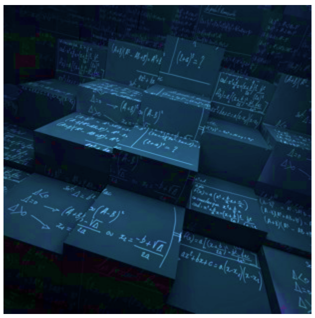


*Figure 3: The business functions and the connections are independent of each other. The business functions and the connections are deemed to be independent of each other. The red dotted line shows how dramatic complexity increases when the number of connections in the system is **doubled.** As a result an IT system consisting of 25 BFs and 25 connections, **when the number of the connections are doubled (i.e. 50 connections), the system complexity increases 5 times.***

What does complexity increase mean in terms of cost?



Since complexity can be measured in SCUs, the cost of the IT system can be represented by a very simple formula:

System Cost = Cost to Produce 1 SCU * Number of SCUs

Cost to Produce one (1) SCU is equal to the cost to produce one (1) *Business Function* with no connections. Because of the linear dependence between the System cost and the Number of SCUs, one can conclude that the increase of complexity is compounded by 5 times, triggering the increase of the system cost by 5 times as well. For this example IT system, the data derived is a graphical dependence between the relative changes of Complexity vs. Number of Business Functions, as is shown in Figure 4 below.

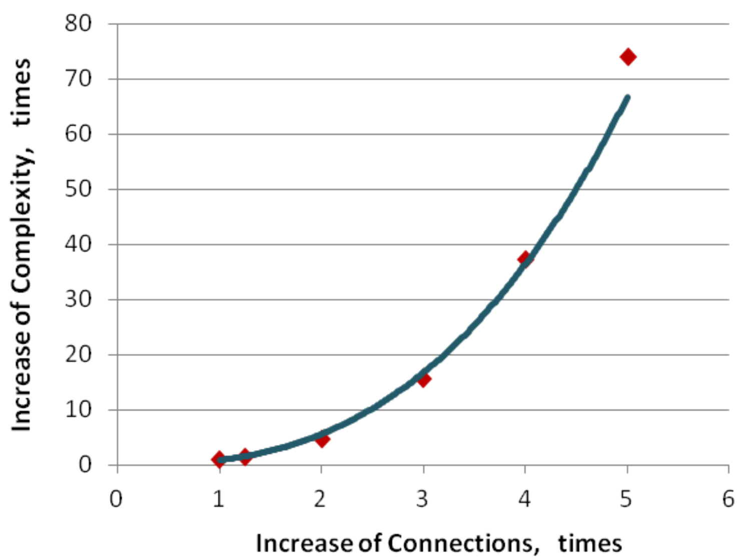
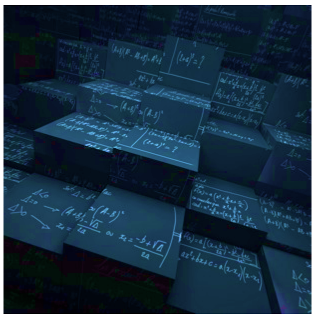


Figure 4. The approximation of the relative complexity change vs. the relative change of the number of connections in our sample IT system. Red square dots – calculation data.

This graph demonstrates the statement made earlier, that complexity increases 5 times when number of connections increases 2 times, i.e. doubles. From the graph, it is also easy to see that if the number of connections in the system increases 5 times, then the complexity increases even more dramatically, by 70 times.

The impact of change in Business Functions or Connections



An important observation that was discovered while deriving this graphical dependence was the fact that this graph can be applied for ANY number of Business Functions and ANY number of Connections; however, this dependence may be completely different in magnitude and behavior for an IT system that employs another architecture and a connections interdependency, e.g., SOA. Based on results of our recent research we also developed new approach to qualitative evaluation of change of an IT system complexity and risks-of-its-failure depending on change of number of system business functions and connections, change of project governance structure and amount of staff involved.

The graph depicts the case of simplified evaluation where as long as the changes in the IT number of business functions and the connections in the IT system are known, one can derive relevant graphs to evaluate the change of complexity and risk-of-failure for a specific IT system.

This whitepaper confirms our research results to demonstrate that system cost changes and system failure rates can be predicted with increasing accuracy, well in advance of an IT project start and along the project management lifecycle, (change process) as long as there is a real time understanding of how the complexity behaves.

How to Eliminate Systemic Failures of IT Projects

System simplification using the SIP-methodology is key to elimination of systemic failures of a complex IT project.

To mitigate project failing risk, at a minimum, the following steps are necessary for this model to succeed. The decision criteria must include the following: Decide whether you want to upgrade legacy IT system or establish a new one using both a cost based and time to value Return on Investment (ROI).

Apply SIP-methodology to find the optimal path for your system structure with the least possible complexity. This step includes the following sub-steps:

- Conduct a stress test designed to assess you organization readiness as per the Harvard Business Review recommendations [14] Make decomposition of a system (existing or planned) into atomic business functions - the lowest level of functionality in a system that is still recognizable to the business.
- Partition this set of business functions into sub-sets based on the relationship called synergistic. Two atomic business functions A and B are considered to be synergistic if, from the business's perspective, A is not useful without B and vice versa.
- Confirm that the atomic business functions aren't more like implementation details.
- Confirm that relationships between business function are really synergistic
- Double check, if every one of these sub-sets is autonomous with respect to the other to minimize number of connections between the sub-sets.
- Identify independent autonomous business functions.
- Finding the system structure with the least possible complexity.

Advantages of IT System Simplification Using Proposed Approach

The proposed approach helps to avoid many the broad range of challenges and delivery problems of any complex IT projects, including the following ones:

- Late deliveries
- Overrun budgets
- Inflexibility and missed business objectives
- Missing functionality or limiting functionality
- Poor privacy and security governance

The Cost of Simplification Versus The Cost of IT System Failures

Complexity is rarely considered as cost-adding factor in evaluation IT system proposals. However, it is proven, that complexity is a major problem for IT systems. It drives up cost, delays projects, introduces security problems and risk, and impedes IT alignment to business strategy. In the event there are two alternate proposals, the least complex is almost always the better one for these reasons.

Typically, the cost of failure of a Complex IT Project dramatically exceeds a project's total cost; they are also known as a "black swan". The term "black swan", [9], describes high-impact events that are rare and unpredictable, but in retrospect, seem not so improbable, e.g., a $5 million project that leads to an almost $200 million loss, is a classic "black swan". Accordingly, [9], fully one in six of IT projects studied, was a "black swan", with a cost overrun of 200%, on average and a schedule overrun of almost 70%.

Most people usually assume that complexity is related to system business functions and in fact, the major determinant of complexity is the IT system's structure [4], that is, how the system is structured into subsystems (services) and how those subsystems (services) are related to each other. Any misunderstanding and disregard for complexity in evaluation of complex IT projects can place the success of a project at risk and will increase costs.

The cost of implementation of a SIP-methodology approach is negligible when compared to the cost of the same complex IT project using traditional or no governance. SIP when optimized generates a cost variance of less than *5% while significantly contributing to a much improved better understanding of the entire business process and solution requirements well in advance of implementation. Cost of a complex IT projects using a SIP-optimized versus a non-optimized system structure may vary within 2 to 50 times (i.e. 100% and much more) [7][15]. Plus, these assumptions do not include any indirect costs and technical debt that could appear as a result of failures of IT system.*

Legacy versus Cloud IT System Complexity

With the advent of cloud computing, mobile, web commerce services and their impact on privacy and security, there are technology trends that obligate both private enterprise and government organizations to upgrade their legacy systems and modernize applications that are often outdated.

Projects requiring the upgrade of legacy IT systems (often highly customized) or the transition of the system to a cloud-based model, are considerably more complicated compared to the new IT pure play system projects, because, as a rule, it dictates the marriage (integration) of both new and dated solutions and data structures. Projects related to the upgrade of these systems, according to our definition, are considered to be “Complex IT Projects”. To efficiently integrate this kind of IT system with cloud technology, an approach to mitigate IT system failures, presented earlier in this whitepaper, has to be employed.

Transition to a cloud doesn’t automatically minimize IT system complexity. Planning the installation of a new pure play complex IT system in a cloud, requires applying SIP-methodology to leverage the cloud advantages in achieving the least IT system complexity, maximum efficiency and cost-effectiveness.

New Opportunities Applying the IT System Simplification Model

Simplification brings not only compelling financial rewards, it also brings many intangible rewards as well, such as:

- procurement becomes much easier and improves the likelihood of predictive and/or fixed costs,
- ability to implement new solutions is being increased, usually reserved for large vendors and system integrators,
- small and mid-sized providers can compete successfully on sub-projects of a complex IT project there by reducing sole source risk and change management cost,
- facilitates easier implementation for the application of agile best practices within organization,
- provides a more simplified and effective communication between IT and the business,
- generates greater enhanced outsourcing opportunities, that can dramatically lower expenses.

The end result: IT systems delivered on-time, on-budget and within business expectations.

The simplification approach considering breaking big projects down into ones of limited size, complexity, and duration is also among the recommendations published in Harvard Business Review, [9].

Human Personality Traits Add Risk of Failure to Complex IT Projects

Even when we employ the best available technology, the most effective methodology and a world-class workforce for the implementation of Complex IT Projects, organizations are still at risk because of a number of human factors affecting all IT project phases. Core to our research of a diverse range of information sources related to failures of Complex IT Projects [16] [5], our research revealed the following six human factors that add risks of failure to Complex IT Projects.

Idolization | Idolization is a typical belief among non-IT people that IT is a unique solution for all possible problems in their organization; often IT professionals are supposed to know everything what is necessary to accomplish the achievement of the organization's goals, and external subject matter experts are only being- invited when the project problems become so obvious that high level officials have to be involved to deal with the problems. So, the organization itself, must take the appropriate actions to be prepared for using the best IT solutions available.

Technophilia | Technophilia is typical belief amongst IT professionals that better technology, and more of it, are the remedies for problem solving, [17]. This may lead to the situation when excessive amounts of IT resources are being aligned to the organization's business strategy, and IT project complexity may go up sharply.

Managerial Faddism | Managerial faddism is the tendency for consultants and managers to employ the newest management methodology or the latest-in "Management Guru" thinking-. It is believed that most problems can be avoided or fixed and deliverables achieved, by improving the organization's management structure along the IT project implementation. In some cases, "managerial faddism" has led to both, total failure of the Complex IT Project as well as management problems within the organization [5].

continued…Human Personality Traits Add Risk of Failure to Complex IT Projects

Lomanism (Arthur Miller's archetypal salesman Willie Loman) **|** Lomanism is the belief, genuine or feigned, that vendor sales representatives and resources develop for their company's products and skills, utopia philosophies where they describe the vendor or system integrator, as a capable developer of the intended products and technologies, whatever the doubts put forward by potential and current customers. Vendors may promise whatever is necessary to close a deal. Organizations are advised to obtain independent verification and validation that involves subject matter experts who can assure that the proposed solution is, in fact, achievable as expected.

Personality and Vendor clashes | Personality clashes may essentially downgrade Complex IT Project performance and deliverables. The clashes may be triggered by the complexity of personal relationships, cultural differences, by unwillingness of superiors to hear about IT project "runaways" for some political reason, or by tensions with the supposed users that wish to do their job in the old manner while making little efforts to learn. This is also often referred to as "finger pointing".

"Unhelpful Doubter" Factor | The "Unhelpful doubter" is the tag for everyone who demonstrates concern, in contrast to the traditionally positive team spirit when the project "runaways" are visible, but still not critical. Usually, no one wants to be the bearer of bad news and to be tagged as the "Unhelpful doubter" for various reasons, including fear for their own future within the project. Because of this, timely and easily corrective measures may not transpire, and the project may steadily move towards unrecoverable problems.

Six Key Imperatives for Driving Success of Complex IT Projects

SIX KEY IMPERATIVES

Research of a large body of literature, COBIT 5 [18] and our empirical experience, show that Complex IT Projects should comply with the following Six Key Imperatives to achieve the highest probability of success.

Imperative 1: Meeting Ongoing Stakeholder Needs.
Every business and public organization exists to create value for their stakeholders delivering benefits for them on time, on budget and at the expected quality. Since every organization has different objectives, the Complex IT Project must align to the organization's context and goals, its specific governance and management processes and practices, as part of the compliance.

Imperative 2: Separating Complex IT Project's Governance from Management.
It is important to make a clear distinction between -Complex IT Project governance and management. These two disciplines include different types of activities, serve different purposes and require different organizational structures. An independent verification and validation (IV&V) must be included to monitor Complex IT Project compliance, against agreed-upon direction and performance parameters and to eliminate any vested interest other than the Complex IT Project success itself.

SIX KEY IMPERATIVES

Six Key Imperatives for Driving Success of Complex IT Projects

Imperative 3: Enabling a Holistic Approach.
Efficient and effective governance and management of Complex IT Projects, requires a holistic approach, taking into account mutually reinforcing interactions of several driving forces, one of the forces being information and related technologies. A set of driving forces has to be defined to support the implementation of a comprehensive governance and management system for Complex IT Projects, so as to achieve the objectives of the project. For example, the COBIT 5 framework defines seven categories of enablers (driving forces) for IT enterprises including:

- Principles, Policies and Frameworks
- Processes
- Organizational Structures
- Culture, Ethics and Behavior
- Information
- Services, Infrastructure and Applications
- People, Skills and Competencies

Imperative 4: Integrating Complex IT Project's Governance into the Organization's Governance.
The information and related technologies employed within the Complex IT Project has to be treated as an asset that needs to be treated like any other asset- in the organization. All IT-related governance and management enablers have to be considered as organization-wide and end-to-end, i.e., inclusive of everything and everyone internal and external that is relevant to governance and management of the organization, information and related IT. A new practice that establishes IT Leadership as a highly effective strategic partner (in contrast to the previous role of "order-taker") is essential to successfully implement this principle.

Six Key Imperatives for Driving Success of Complex IT Projects

SIX KEY IMPERATIVES

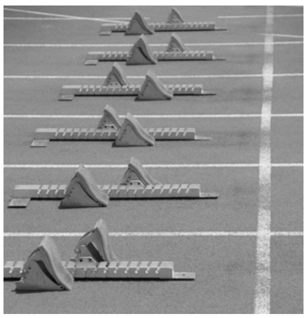
Imperative 5: Applying an Iterative IT Project Management Approach.
From the point of view of business continuity and efficient change management, any Complex IT Project, as it is a long term endeavor, must be approached as an iterative process, where waterfall methodologies no longer apply and have been replaced by agile methodologies.. This doesn't mean the waterfall approach is inherently defective or bad, it is an idea for very short-term and straightforward projects. In the case of Complex IT Projects, –it is trying to "eat the elephant in one bite"! An iterative project management approach dramatically improves the value-to-market performance and, therefore, reduces the difficulty in gaining buy-in from Complex IT Project stakeholders. There are a number of new solutions for project change management currently available on the market, particularly Kanban, where the project is holistically harmonized.

Imperative 6: Applying Automated Traceability of Change.
From a large body of literature on Complex IT Project failures, it is evident that any searches for 'silver bullets' or 'unique techniques', whether it is a new programming language or a cutting-edge system design technology, largely miss the significance when it comes to the fallibility of humans.

Up-to-date change tracing activities, as conducted in industry, are labor-intensive, tedious and prone to human error [19]. In the past 10 years, the traceability research community has worked to address this problem by developing automated techniques aimed at generating traceability matrices between pairs of textual software engineering artifacts. Similar automation approaches have to be applied in Complex IT Projects for building and keeping up-to-date Requirements Traceability Matrices. Unfortunately, practitioners often do not build traceability matrices, or do not keep them up to date.

Success of Complex IT Projects in conditions of the fast paced "agile" market, requires an efficient, accurate and certifiable automated change management process that is aligned with the quality assurance process.

How do you Prepare Your Organization?



The following organizational and methodological actions can contribute to effectively mitigate risks of failures in new or ongoing Complex IT Projects/ systems.

Organizational actions:

- Take a stress test designed to assess your organization's readiness. Ask two key questions, [9]:
    - First, is the organization strong enough to absorb the impact if its Complex IT Project goes over budget by 400% or more, or if only 25% to 50% of the projected benefits are realized?
    - Second, can the organization endure the impact if 15% of its medium-sized IT projects exceed cost estimates by 200%? Note: this excludes projects that get all the executive attention, with the secondary ones being often overlooked.

  These numbers may seem improbable, but, as research, [9], of 1,471 IT projects show, they apply with significant frequency.
- Avoid accepting excessive amounts of IT system change requests during the project implementation. As expressed previously, 25% of IT project changes will double the complexity of the project and may become the next Nobel Prize winner for "Lessons Learnt from Failure to Deliver IT Projects in Time and within Budget"
- Conform to today's business development trends by –transitioning IT Leadership's role from "order-taker to a strategic business partner, who leads discussions on market-facing technology ideas", [6], This would be a substantial asset.
- Embrace and be prepared to adopt social media, mobile and emerging technology consumer trends, i.e. BYOD (Bring-Your-Own-Device) where you can no longer predict new advancement, but be open and aware that new "things" will emerge that could cause upstream issues with current in-place services.
- Create a library of accepted organization's enterprise applications as an internal service, where your stakeholders can obtain the application that is relevant to his/her function, managed from your organization's server.

Methodological actions:

- Well before investing in a Complex IT Project, pre-design the intended IT system using the methodology above for partitioning business functions into sub-systems (services) such that the overall system has the least possible complexity required to solve the business problem and minimize system failures.

- For ongoing Complex IT Projects, have an approved termination plan that includes an off ramp, bridge or closure when the project crosses a pre defined cost threshold, time schedule, or non-delivery that is not acceptable and limit or dissolve any further investments or project continuation.

- Implement a compliance plan that protects the business, regardless of claims that it 'is almost there' or 'it will be alright on the night". Apply practices that mitigate the potential for significant technical debt as a result of unnecessary direct and indirect expenses that could have been saved [5]. In this case, the money spent before the project's termination could be considered as a learning tax paid to contribute to your organization's LFF library of knowledge.

- For legacy IT systems, which cannot be abandoned, consideration should be given to implement a soft-decommissioning approach, by applying the system deconstruction into services containing synergistic business functions, with each subsequent integration of these functions into the new IT system functions.

- Use agile, i.e. incremental, evolutionary and modular approach to development and implementation of IT systems.

About the Author – Dr. Mykhailo Magal, PhD

Dr. Mykhailo Magal has over 20 years of experience in the planning, management and integration of complex multi-business solutions with more than 10 years of experience dedicated to the delivery of complex IT systems. Currently he is providing consulting services in solution strategy formulation, complex IT systems planning, development and implementation to global clients.

Servicing both on-site and remote global working environments, Dr. Magal has successfully applied the principles of this whitepaper on Linux, Java and MS platforms. He is well versed in the understanding and successful delivery of complex IT projects, execution and coordination of multiple IT based initiatives in the areas of international trade, investment cooperation, technology transfer, and public-private partnerships mechanisms in both the commercial and government sectors.

Dr Magal's proficiency in application lifecycle development in Linux, Java and MS Windows environments applies Agile lifecycle models and SDLS methodology. He brings to his clients a wealth of experience in executive management of software development projects (PHP-MySQL, Java-Oracle and C#-Oracle integrated development environments). More recent projects include (WMware) based Virtualization and cloud e-commerce + CRM/ERP systems.

Dr Magal holds Ph.D. in physics and mathematics, and a B.Sc. in Computer Science.

He is a member of Project Management Institute (PMI) and International Institute of Business Analysis (IIBA) and certified in IT Service Management (ITIL) Certification

Glossary

ABF (Atomic Business Function) | ABF is the lowest level of functionality in a system that is still recognizable to the business.

Autonomous business functions means, among other things, that its data is not shared between other functions and/or sub-sets of functions and that any activity coordination must occur through work requests.

Autonomous sub-set of Business Functions means, among other things, that data is not shared between sub-sets and that any activity coordination must occur through work requests

Complexity | Complexity is the attribute of a system that makes that system difficult to use, understand, manage, and/or implement. Complexity is measured in units defined by a

Simplification framework, e.g. in SIP methodology, complexity is measured in Standard Complexity Units (SCUs), [20].

EA (Enterprise Architecture) | Enterprise Architecture is a discipline for proactively and holistically leading enterprise responses to disruptive forces by identifying and analyzing the execution of change toward desired business vision and outcomes. EA delivers value by presenting business and IT leaders with signature-ready recommendations for adjusting policies and projects in order to achieve target business outcomes that capitalize on relevant business disruptions. EA is used to steer decision-making towards the evolution of future state architecture [21].

FEA | Federal Enterprise Architecture.

Gartner | Gartner enterprise architecture methodologies and practices.

IV&V (Independent Verification and Validation) | IV&V means that a completely independent entity evaluates project deliverables generated by the team that is designing and /or executing the project.

LFF (Learning From Failures) Library| LFF Library is a collection of certain format documents about the IT project/system failures that have happened within the organization, that is available to staff with relevant functions.

Continued …   Glossary

Nonlinear system | A Nonlinear system is where the output is not directly proportional to its input, e.g. 1 input triggers 2 outputs, 2 inputs trigger 12 outputs, 3 inputs trigger 36 outputs, etc. Most physical systems are inherently nonlinear in nature. Mathematical modeling of non-linear systems is often very difficult or impossible.

Partition | A partition is a set of sub-sets such that each of the elements of the original set is now in exactly one of the sub-sets. It is incorrect to use the word *partition* to refer to one of the sub-sets. The word partition refers to the set of subsets, [22].

SBF (Synergistic Business Functions) | Two atomic business functions F1 and F2 are considered to be synergistic if, from the business's perspective, F2 is not useful without F1 and vice versa [7].

SCU (Standard Complexity Unit) | SCU is defined as the amount of complexity in a single business function implemented in isolation and lacking connections to any other business functions or systems, [20].

SIP (Simplified Iterative Partitioning) | SIP is a pre-design methodology, proposed in [2], for partitioning business functions into sub-systems (services) such that the overall system has the least possible complexity required to solve the business problem and minimize system failures.

TOGAF | Open Group's Architecture Framework.

Zachman | Zachman Framework for Enterprise Architectures.

References

[1] Office of Management and Budget, Analytical Prospective, Budget of the United States Government, Fiscal Year 2009, Washington: U.S. Government Printing Office, 2008.

[2] CBC News, "EHealth scandal a $1B waste: auditor," CBC, 7 October 2009. [Online]. Available: http://www.cbc.ca/news/canada/toronto/story/2009/10/07/ehealth-auditor.html.

[3] "Large Scale Complex IT Systems," The UK Research & Training Initiative, 2011. [Online]. Available: http://lscits.cs.bris.ac.uk/FAQ.html.

[4] K. Flinders, "Why do big IT projects fail? Part one: The professionals," 24 October 2011. [Online]. Available: http://www.computerweekly.com/news/2240106569/Why-do-big-IT-projects-fail-Part-one-The-professionals.

[5] R. Gauld, S. Goldfinch and T. Dale, Dangerous Enthusiasms: E-Government, Computer Failure and Information System Development, Otago University Press, 2006.

[6] J. Kornik, "The Future of IT," Consulting Magazine, February 2013. [Online]. Available: http://www.consultingmag-digital.com/consultingmag/201302#pg10.

[7] R. Sessions, "The IT Complexity Crisis: Danger and Opportunity," 8 November 2009. [Online]. Available: http://www.objectwatch.com/whitepapers/ITComplexityWhitePaper.pdf.

[8] J. Widman, "COMPUTERWORLD: IT's biggest project failures -- and what we can learn from them," 9 October 2008. [Online]. Available: http://www.computerworld.com/s/article/9116470/IT_s_biggest_project_failures_and_what_we_can_learn_from_them?taxonomyId=73&pageNumber=1.

[9] B. Flyvbjerg and A. Budzier, "Why Your IT Project May Be Riskier Than You Think," September 2011. [Online]. Available: http://hbr.org/2011/09/why-your-it-project-may-be-riskier-than-you-think/ar/1.

[10] L. Myers, "$1 billion wasted on Air Force computer system," NBC News, 9 February 2013. [Online]. Available: http://www.nbcnews.com/video/nightly-news/50749586#50748711.

[11] C. Kanaracus, "California ends contract with SAP over troubled IT project," Computerworld, 8 February 2013. [Online]. Available: http://www.computerworld.com/s/article/9236662/California_ends_contract_with_SAP_over_troubled_IT_project.

Continued …. References

[12] C. Wood, “California DMV Cancels IT Modernization Project Contract,” Government Technology, 14 February 2013. [Online]. Available: http://www.govtech.com/e-government/DMV-IT-Modernization-Project.html?utm_source=related&utm_medium=direct&utm_campaign=DMV-IT-Modernization-Project.

[13] R. L. Glass, Facts and Fallacies of Software Engineering, Boston: Addison Wesley, 2002.

[14] B. Flyvbjerg and A. Budzier, “Why Your IT Project May Be Riskier Than You Think, Harvard Business Review,” September 2011. [Online]. Available: http://hbr.org/2011/09/why-your-it-project-may-be-riskier-than-you-think/ar/1.

[15] R. Sessions, “A Fundamental Metric of Predicting IT Success,” 23 April 2012. [Online]. Available: http://www.objectwatch.com/whitepapers/ComplexityMetricSessions-100.pdf.

[16] A. Tesser and S. Rosen, “Fear of negative evaluation and the reluctance to transmit bad news.,” Journal of Communication, vol. 22, no. 2, pp. 124-141, 1972.

[17] K. Lyytinen and D. Robey, “Learning failure in information systems development.,” Information Systems Journal, vol. 9, no. 2, pp. 85-101, 1999.

[18] ISACA, COBIT 5: A Business Framework for the Governance and Management of Enterprise IT, Rolling Meadows: ISACA, 2012.

[19] D. Cuddeback, A. Dekhtyar, J. Huffman Hayes, J. Holden and W. Kong, “Towards Overcoming Human Analyst Fallibility in the Requirements Tracing Process (NIER Track),” 28 May 2011. [Online]. Available: http://selab.netlab.uky.edu/homepage/publications/ICSE10NIER-camera.pdf.

[20] Consortium for Untangling Enterprise Complexity (CUEC), “CUEC Standard: Definitions of Common Terms,” 11 June 2010. [Online]. Available: http://www.cuec.info/files/5613/3555/1743/CUEC-Std-CommonTerminology-Latest.pdf.

[21] Gartner., “IT Glossary,” Gartner, Inc., 2012. [Online]. Available: http://www.gartner.com/it-glossary/enterprise-architecture-ea/.

[22] R. Sessions, “The Mathematics of IT Simplification,” 1 April 2011. [Online]. Available: http://www.objectwatch.com/whitepapers/MathOfITSimplification-103.pdf.

Is your organization ready to start this model?

The research presented in this whitepaper confirms that industry trends and the climbing failure rates of complex IT projects require a new attitude and more meticulous approach to planning new or upgrading legacy IT systems in order to achieve the expected return of investment from complex IT projects. By applying these approaches, as briefly described in this whitepaper, organizations will no longer suffer from the business challenges due to delays and budget overruns that are triggered by failures of complex IT projects. The time to start is immediately such that your organization gains SIP competencies necessary to reduce the risk associated with Complex IT Projects.

"Removing the complexity of any IT project and making it simple, is a refreshing and timely opportunity to return the value again. Dr. Magal explains to us, the notion of employing a simplified and iterative approach using the principles of math. The result is, a marked improvement in IT project outcomes, reduced risk of project failure and more importantly, the elimination of greater "Technical Debt".

Wayne W. Wood
Managing Partner – QAGov.com


complexit.ca