

Program Management Design: A Viable System Perspective

Program Management (PM) is gaining importance as a competency. This is amidst businesses trying to find a better way to manage portfolios of projects. Projects are getting larger in size and heterogeneous in characteristics. The good news is PM is being recognized as a domain that needs design clarity with an added degree of standardization. Project Management Institute (PMI) and Office of Government Commerce-UK (OGC), independent fraternities in PM, have come up with standards and methodologies for PM.

Understanding the design of PM would require dealing with aspects of structure, governance, processes, and competencies. The science is complex when we see inter-relationships within these aspects and their manifestations in various project environments. Therefore, understanding PM design can be taken from various perspectives, with each throwing distinct light on its design elements.

Here, we will attempt to understand PM design from the perspective of the cybernetic theory of organization structure, namely the Viable System Model. This was developed by operations research theorist Stafford Beer, as an organization structure model responsive to the changing environment. This adds an interesting perspective in understanding the abstractions and inter-dynamics in PM function, and therefore, provides us with a conceptualization of how the PM functions can be defined and made adaptive.

About the Authors

Veerendra K. Rai

Veerendra K. Rai holds a Ph.D. in Industrial Engineering and Management from the Indian Institute of Technology, Kharagpur, and an M.S. in Computer Science from the New Jersey Institute of Technology (NJIT), U.S.A. He has been working with Tata Consultancy Services (TCS) since 1998. At TCS, he has worked in the Systems Engineering and Cybernetics Center, Artificial Intelligence group, and Software R&D group. His research interests include Systems Thinking, Project & Program Management, Software Engineering, and Information Systems. Mr. Rai has a number of publications in these areas and is a life member of Systems Dynamics Society.

K. Subramanian

K. Subramanian has about 19 years of experience with TCS. During this time, he has played key roles in IT Systems Development, Business Consulting, Project Management, Program Management, and Operations Management. He was responsible for TCS's business operations in Australia and New Zealand for five years. His consulting and program management experience includes a number of Fortune 500 companies, including IBM, American Express, GE Capital, Ford Motor Company, Eli Lilly, Novartis, Kellogg, ING, and Johnson & Johnson. He has been trained in the application of Systems Engineering and Complexity Science to business problems. He conducts training courses in Program Management, Risk and Contingency Management, and Systems Thinking. Mr. Subramanian holds a Bachelor's Degree in Mathematics and is an Associate of the Institute of Chartered Accountants. He currently leads the Program Management Practice in TCS.

Table of Contents

1. Background	3
2. Modeling the Program Structure and Control	5
3. Superimposition of Viable Systems on Program Management	6
4. Conclusion and Scope of Future Work	10
5. Case study : Setting a SAP Centre of Excellence (CoE) for a Large Pharma Company applying Viable System Model	11
6. References	13

Background

Changes in the global economy and the impact of IT and ITES reinforcements on it have led to enhanced individual and organizational efficiency and productivity. PM has served as a great enabler in this quest, however, it could not remain unaffected by the changes and has grown exponentially in size and appeal. It has also registered a handsome growth (Mikheev and Pells, 2005). The PM profession now includes PMI, with 15,000 members, and IPMA, with 38 members from the national project management associations from countries in Europe, Asia, and Africa, among them the Russian association of management of projects (SOVNET), the Ukrainian PM association, the Australian Institute for PM, and the Japan PM forum.

The change in PM is not quantitative and linear. Mikheev and Pells (2005) have divided the era of PM into three waves. According to them, the first wave spanned from 1950 to 1980 and was termed the technical paradigm. The second wave spanned from 1980 to 1990 and was referred to as the manager's paradigm. The third wave, started in the year 2000, still continues and has been termed as phenotype paradigm.

The first wave was dominated by management science and its tools, such as Program Evaluation and Review Technique (PERT), Critical Path Method (CPM), and Earned Value Management Systems (EVMS).

The essence of the projects also did not remain static. During the first wave, the essence of the project consisted of technical, social-technical, and biomechanical dimensions. The second wave added the social element into its fold, while the third wave took two more additional elements; biological and biosocial (Mikheev and Pells, 2005).

During this period, a number of PM models were proposed. Evolution of PM models could be found in the works of Tanaka (2005). Development at NASA and the U.S. Department of Defense (DoD) in the 1950s and 60s led to the development of classic PM models. The classic model was expanded due to growth of the PM profession to include more and more PM functions as recorded in various PMBOKs. Coincidentally, these changes took place during the era of globalization and liberalization of the world economy, where increase in productivity and efficiency was required to survive under intense forces of competition in the market. The role of Information Technology (IT) in all of this was abundantly appreciated and PM was embraced by IT organizations, as well. Thus PM, which started with areas such as aerospace, defense, engineering, and construction industries, moved to other application areas, including manufacturing, during the 1980s and 90s and finally moved to IT. Besides, a neo-classical model emerged in the oil and gas industry in the 1980s as it adapted NASA and DoD models for large complex projects.

As a future extension of these models, Tanaka (2005) predicts a versatile PM model based on user-friendly tools and technologies, and that all of these models would operate in the global market in a parallel fashion. Or, as authors believe, would there be a general theory of PM, which would integrate and synthesize domains and application areas with the help of suitable technology? This appears more plausible given the advances in technology, particularly in IT.

Mikheev and Pells (2005), in the discussion of the third wave, mention phenotype paradigm, where the term phenotype means that projects arise, begin, and develop in environments under the same laws as living organisms with genotypes set originally. This brings us to the realm of systems engineering and cybernetics, which have taken inspiration from living organisms and machines maintaining their identity and purposeful existence in their environments, and have discerned an underlying set of laws that govern these systems. Systems scientists developed theoretical frameworks to model organizations based on these laws. Some of the prominent laws in this set are: the law of requisite variety, recursion, feedback loops, interrelationship of structure and behavior, viability, purposive behavior, and non-linearity.

Programs are complex recursive systems and obey all of the above-mentioned laws. One of the theoretical frameworks based on the principles of variety, recursion, feedback, and viability is Viable System Model.

2.1 An introduction to Viable System Model

Viable System Model (VSM) was proposed by Stafford Beer, an influential thinker in the science of cybernetics in the last century. 'Viability' here refers to the ability of a system to maintain its separate purposeful existence.

VSM consists of five recursive systems:

- System 1 carries out the fundamental operations of the organization that justify the existence of the organization.
- System 2 coordinates the activities of System 1 through a regulatory center, which happens to be local management.
- System 3 performs the function of everyday control of System 1 by senior management besides supervision of System 2. There is a special system designated as System 3*, which deals with audit and costing functions.
- System 4 scans internal and external environments of organization and gathers intelligence.
- System 5 decides on policies and directives of governance (Hilder, 1995).

A detailed study of VSM can be found in Beer (1985) and Espejo and Harnden (1989). A conceptual diagram of the model architecture is given in exhibit 1.

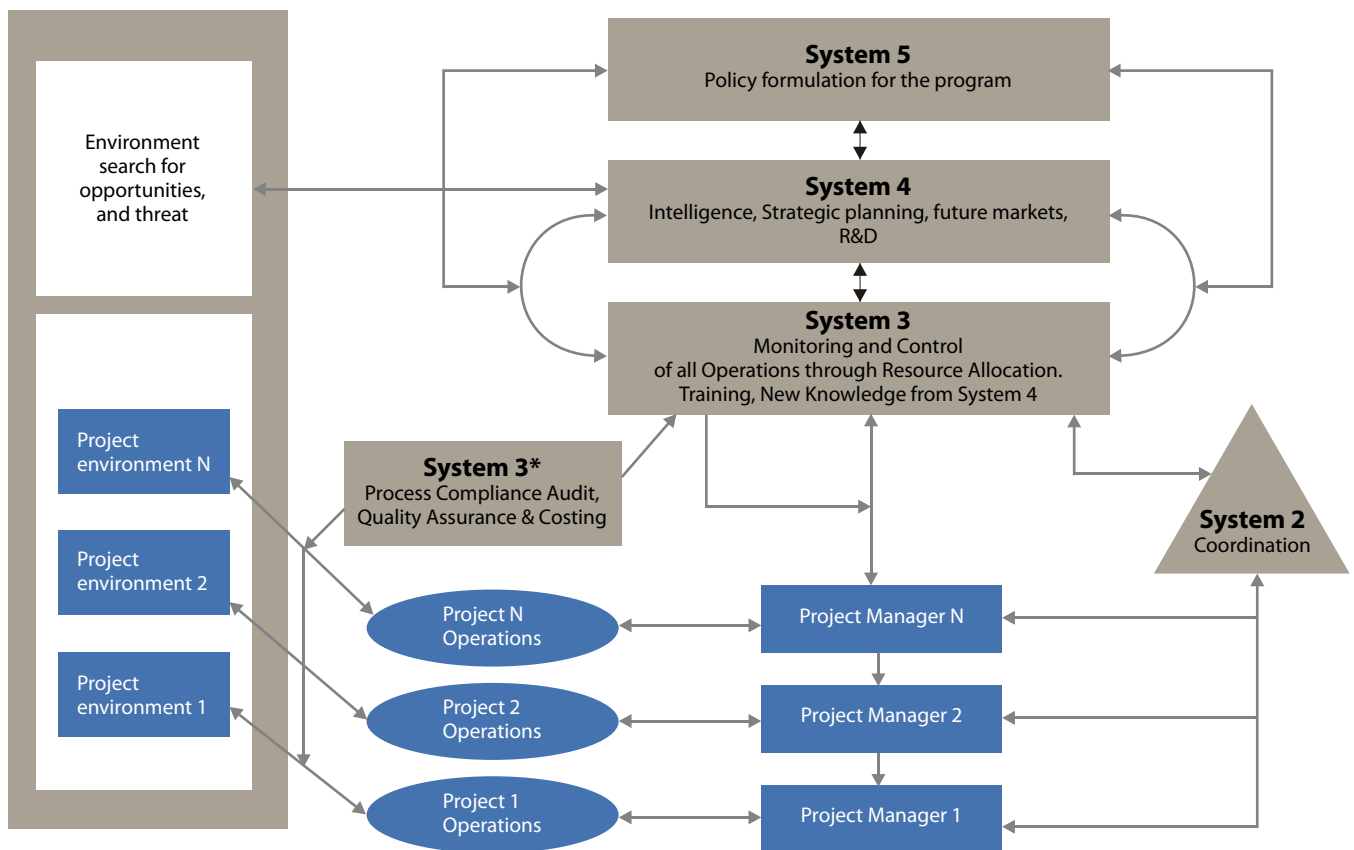


Exhibit 1: Conceptual View of Viable System's Architecture

2.2 Arrangements of Viable Systems

Systems 1 to 5 are arranged in such a way that any complexity left over by the lower system is handled and absorbed by higher ones. All residual variety and complexity is absorbed at System 5, where the accountability eventually stops. The absorption of variety is governed by the law of requisite variety by Rosh Ashby, which states that variety of the controller entity equals the variety of controlled entities for control to be obtained where variety is defined as number of states a system can assume at a given point of time (Hilder, 1995). Variety absorption takes place through two tools, namely, variety attenuator and variety amplifier. Variety attenuators reduce variety and thus complexity by picking up the relevant information and rejecting the irrelevant information. Market research is an example of variety attenuator as only relevant information is focused on. Variety amplifier, on the other hand, helps in increasing the variety of the controller to match the variety of the controlled. An example of variety amplification is usage of tools and technology.

Systems 1 and 4 connect to the environment. Systems 3 and 4 are linked. Systems 1, 2, and 3 are linked. No link exists between 2 and 4, 1 and 4, and 1 and 5. Each of these systems is a viable and recursive system. It is pertinent to note that each of these systems recursively contains all five systems.

Modeling the Program Structure and Control

3.1 Program Complexity: A Glimpse

In the first instance, we outline the dimensions of program complexity and then we superimpose the five viable systems on the entire gamut of PM in the second instance.

Program complexity can be represented across two dimensions: complexity across time and complexity across space. Program phases spread program complexity across time. A typical program cycle consists of the following phases: Program Initiation, Program Definition, Program Mobilization, Program Execution, and Program Closure. It does not take any ingenuity to visualize what would happen if the entire program complexity were to be concentrated in one single point in time and in one single phase. That mind-boggling complexity would simply make any task impossible to execute.

Programs also spread the complexity across space. Projects in the program portfolio spread the program complexity across space. Could we ever manage program complexity being choked in one project at one point in space?

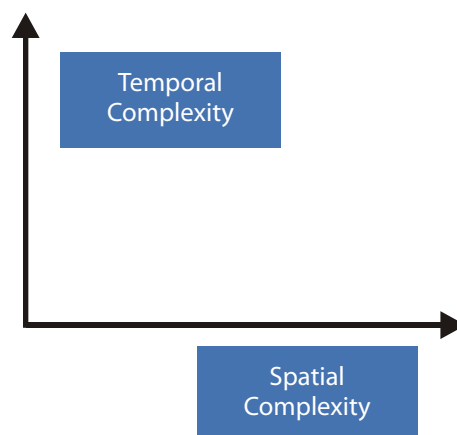


Exhibit 2: Temporal and Spatial Complexity of Programs

Superimposition of Viable Systems on Program Management

4.1 System 1: Implementation

Fundamental operations of a program are carried out by projects that a given program portfolio consists of. Projects and their corresponding operations justify the existence of a program. Each project in the program is managed locally by a project manager. Projects in the program regularly interact with their environments to meet their immediate and short term goals. Projects may have their separate environments, partially overlapped environments, or fully overlapped environments, depending upon their interrelationships. The external environment for the projects consists of the client and their environment. The internal environment consists of local, senior, and top management. Each project in the program is a viable entity in itself by virtue of a set of well defined goals and associated structure and resources to achieve the goals. Exhibit 3 shows the graphical display of System 1, Implementation.

4.1.1 Complexity Handling At Implementation Level (System 1)

Some of the mechanisms for handling complexity at the implementation level are given as follows:

Distribution of Programs into Projects Recursively

The programs are divided into projects and the projects are divided into subprojects and so on, to the level of a well defined modularity.

Selection of Appropriate Project Managers for Projects

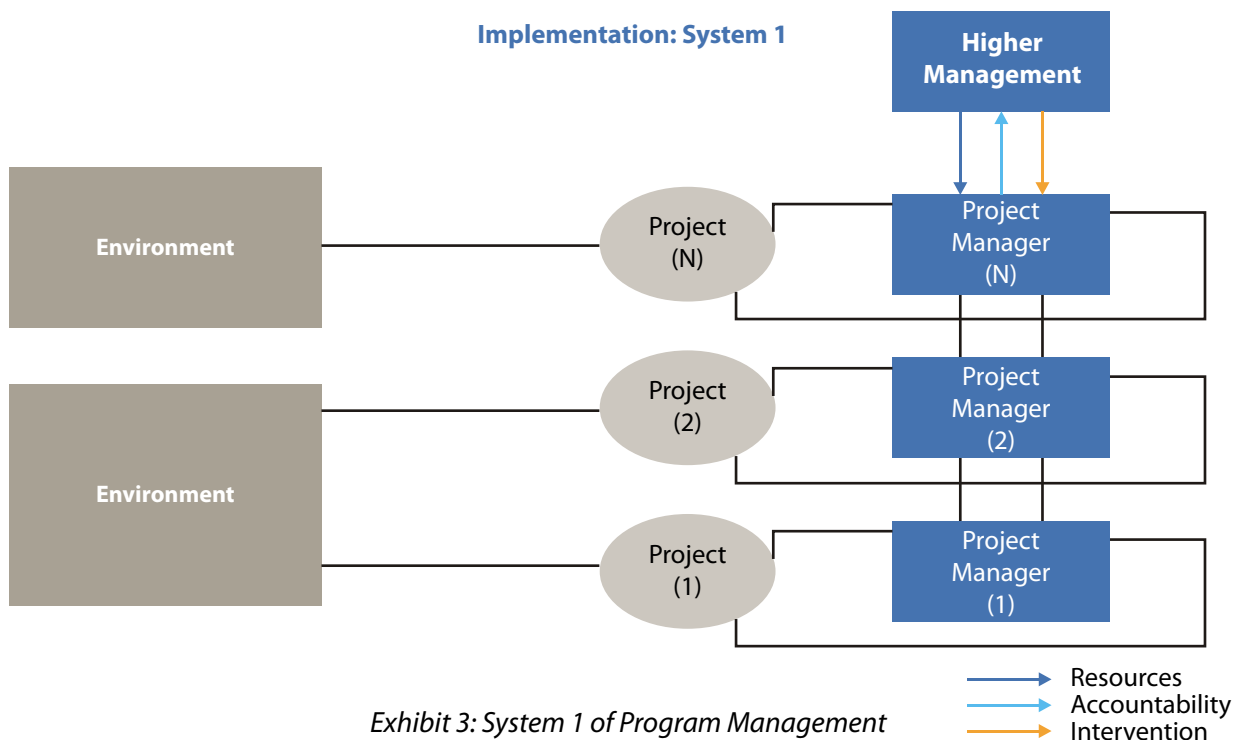
Selection of an appropriate project manager for the projects in the program befitting the project and program type is a key to ensure successful completion of the project and eventually the program.

Team Establishment and Skill Selection

Division of labor, defining the team structure, and roles and skill selection absorbs the variety at the operational level of the project.

Holding Project Managers Accountable

Senior management holds the project manager accountable for respective projects in lieu of the resources given to them for project execution.



4.2 Need for Coordination (System 2)

Projects in the program may have interdependence among themselves. Within a project, there are interfaces between value-adding primary functions and between primary and support functions. For instance, project execution activities interface with human resource, training, and quality assurance that entail coordination between the primary and support functions. In the program management context, some of the instances of coordination activities are:

- Coordination among projects of the program portfolio
- Coordination among different phases of the program
- Coordination among respective project managers
- Inter-team and intra-team coordination

As Espejo and Gill (1989) note, the term coordination is used in the sense of mutual adjustment between the functions rather than command and control. A graphical structure of coordination, that is, System 2 is depicted in the following exhibit, where PMO stands for Program Management Office.

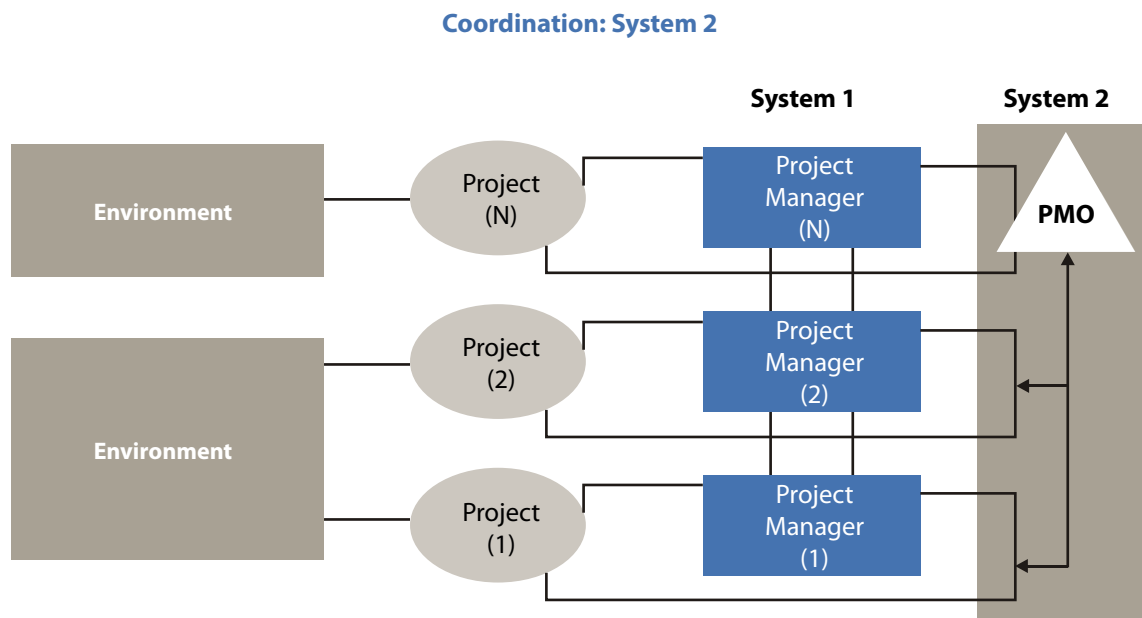


Exhibit 4: System 2 of Program Management

4.2.1 Dealing With Complexity at System 2

Program complexity at this level is handled through the following tools, techniques, artifacts, and actions:

- Project plan
- Project templates
- Project management tools
- Resource leveling
- Workflow definition
- Business process redesign
- Training and learning; knowledge and skill management
- Process definition and adherence
- Implicit and explicit inter-team and intra-team exchanges
- Establishment of program management standards
- Program management methodology

4.2.2 Program Management Office

Program Management Office (PMO), of late, is emerging as best practice for the vast majority of Fortune 500 companies. It is not, however, yet another project management layer in IT organizations. PMO was created as a best practice to drive IT initiatives, while being accountable to the business for benefits, including and not limited to Return on Investment (ROI). With this notion in mind, the PMO could be positioned as Coordinator and constitute the System 2 of PM as a viable system.

4.3 Need for Control (System 3)

The information provided in the status reports of projects are likely to be biased. Senior management, therefore, needs to have some alternate channel to verify that information received reflects the accurate status of the activities carried out in the projects of the programs. To achieve this, senior management might take a walk through the operations of the project and bypass the local management of projects. The intention is not to shoulder-snoop the project manager, but only to show legitimate interest and know first-hand about the state of affairs of the project. This is more of an audit function embedded in the system to ensure projects' compliance.

Control: System 3
<ul style="list-style-type: none"> • Monitoring and control of the activities of projects in the program portfolio • Supervision of coordination activities i.e. of Program Management Office • Internal assessment of Programs (Strengths and weaknesses) • Project performance evaluation against set goals

Exhibit 5: System 3 of Program Management

4.3.1 Complexity handling at System 3

Complexity is handled at System 3 with the help of the following:

- Audits
- Status report
- Spot checks
- Resource negotiation tools
- Accountability reports
- Exception handling system
- Functional autonomy of projects to avoid unnecessary intervention

4.4 Need for Intelligence: System 4

Programs continuously interact with the environment in order to:

- Understand the total environment in which the program is embedded
- Scan the external program environment for opportunities and threats to the program
- Connect to and from the control unit for program control
- Maintain a model of the program and its environment at any given point of time

Exhibit 6 shows the graphical display of System 4.

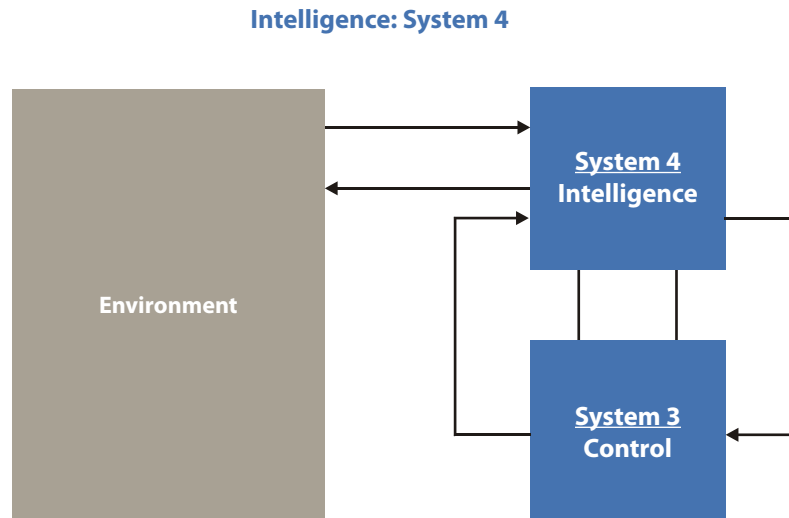


Exhibit 6: System 4 of Program Management

4.4.1 Complexity handling at System 4

Complexity at System 4 is handled as follows:

- Conducting research and development to exploit the opportunities and ward off the threats to the program
- Using information technology and tools extensively
- Creating an intelligence and operations 'Call Center' for the program
- Using program modeling and simulation based on the data sent by System 3 and sending feedback to System 3 to take appropriate action

4.4.2 The Program Manager

The program manager is an essential part of PM design, and selection of a suitable program manager is probably the most crucial decision to be made to ensure successful execution of programs. A full discussion on program manager goes beyond the scope of this paper. Suffice is to say that the program manager serves as a vehicle for transmitting information on policies, trends, and current thinking affecting PM. System 3 (control) and System 4 (intelligence) are handled by program managers in most of the current organizations. However, a different set up could also be possible. For instance, System 3 is handled by the program manager, while System 4 is handled by the client organization. A number of combinations are possible. The point to be noted here is that there are certain functions to be performed. Who, and in what role, is performing these functions is of no consequence as these functions need not be personalized.

4.5 Need for Policy (System 5)

The last level of the program structure, which besides being primarily responsible for providing direction, values, and purpose for the program, does the following:

- Takes eventual accountability of the program (this accountability could not be delegated)
- Provides basis to System 4
- Constitutes the persona of the organization that executes the program
- Acts as executive sponsor of the program or program management board
- Supplies logical closure to the program
- Monitors the equilibrium between System 3 (control) and System 4 (intelligence)
- Absorbs all the residual variety and handles all the leftover complexities of the program

Exhibit 7 shows the graphical display of System 5.

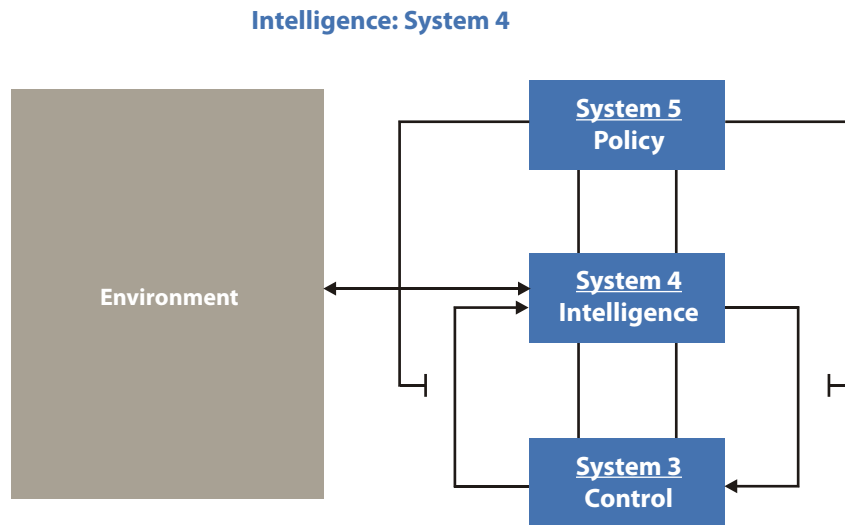


Exhibit 7: System 5 of Program Management

4.5.1 Complexity handling at System 5

Complexity at this level is handled with the help of the following:

- Formulating policies to handle program execution
- Moderating, synthesizing, and providing resolution to any disparities between 'control' and 'intelligence' functions of the program manager as both functions are handled by him/her only
- Setting the priorities
- Using executive fiat to resolve outstanding conflicts
- Holding program manager accountable for the program

Conclusion and Scope of Future Work

This paper identifies recursive and viable structures and controls in the PM domain and attempts to map these onto the VSM framework. In doing so, the paper identifies mechanisms to handle the complexity at each system level and proposes design guidelines. This work establishes the design outline of the PM structure and control. A detailed program design is the intended extension of this work, which would also include the selection of the appropriate program manager, an essential part of PM design.

Case Study : Setting an SAP Centre of Excellence (CoE) for a Large Pharma Company Applying Viable System Model

Overview

The SAP program at a large pharma company aimed to implement common business processes worldwide enabled by an integrated information environment. The benefits of a global design include:

- Reduced complexity and cost
- Faster decision-making
- More flexibility
- Increased ability to manage change

The SAP Centre of Excellence is a unit within the SAP organization whose primary responsibilities include:

- Application support
- Infrastructure
- Validation process
- Testing process
- Change control
- Documentation
- Security

TCS Engagement: TCS was engaged within the SAP Centre of Excellence for providing application support and security services for all the applications within the global business integration program portfolio.

Application Areas

The application set supports a number of global business process areas, such as:

- Finance
- Human resources
- Supply-chain management, including manufacturing, quality management, inventory management, warehouse management, and planning and optimization
- Project and program management
- Information delivery
- Security

Services Provided

The services provided were:

- Service level based resolution of trouble-tickets and change requests
- Solution delivery activities in certain process areas

Service Levels

There were a total of 14 service levels defined to cover the following aspects of service:

- Response time
- Resolution time
- Commitment to agreed dates
- Aging of trouble-tickets
- Bad fixes
- Minimum work commitment

Volumes

There were a total of 14 service levels defined to cover the following aspects of service:

- Response time
- Resolution time
- Commitment to agreed dates
- Aging of trouble-tickets
- Bad fixes
- Minimum work commitment

Management Structure

The management structure consists of the following components:

- Program manager
- Program management office
- One service manager for each process area
- One module leader for each SAP module

Processes

A number of key processes were implemented to effectively manage this engagement, including:

- Problem management process
- Change management process
- Work supervision
- Quality management
- Service level management
- Operational and management reporting
- Capacity management
- Competency management
- Resource logistics
- Project management office processes

Required Skill Sets

The engagement required a range of skill sets to match the application landscape at the client location. These included:

- Finance and controlling
- Human resources
- Sales and distribution
- Materials management
- Warehouse management
- Production execution
- Quality management
- Advanced planning and optimization
- Catalyst

The team size was approximately 200.

Review Processes

The engagement follows TCS policies through the review process. These include:

- Weekly work progress review
- Weekly SLA review
- Monthly management review
- Quarterly senior management review

Past Issues

The client had the following concerns in the past:

- Growth in backlog
- SLA failures
- High attrition of SAP consultants
- Repetitive quality failures

TCS Interventions

TCS implemented the following to resolve the client's concerns:

- Conducting systems analysis of the root-cause of the issues
- Evolving a transition and support model for a holistic approach
- Strengthening of management structure using the VSM as a reference framework

Adoption of the Viable System Model for effective design

TCS used the VSM to evolve the following management and governance structure for the program:

System 5: Program steering committee involving the client and TCS executive sponsors

System 4: Intelligence embedded in the program steering committee and the program manager

System 3: Role of program manager provided this critical element of the overall program

System 2: The program management office helped in coordination among the different process areas, as well as conflict resolution

System 1: Service managers and the service teams who provide the SLA support

References

Beer, S. (1985). *Diagnosing the System for Organizations*, Wiley, Chichester

Espejo R. and Gill A. (1989). *The Viable System Model as a Framework for Understanding Organizations*. Wiley, Chichester

Espejo R. and Harnden, R. (1989). *The Viable Systems Model - Interpretations and Applications of Stafford Beer's VSM*, Wiley, Chichester

Hilder, T. (1995). *Cavendish Software Ltd., Portions © Stafford Beer 1985.*

Mikheev, V. and Pells, D. L. (2005). *The Third Wave – A New Management Paradigm For Project and Program Management; How Modern Project Management Can Transform Your Business, Organization & People*. Paper for www.pmforum.org, June 2005

Tanaka, H. (2005). *The Changing Landscape of Project Management, Project Management World Today*, March 2005 (available at www.pmforum.org/featindex.htm)

About Tata Consultancy Services (TCS)

Tata Consultancy Services Limited is an IT services, business solutions and outsourcing organization that delivers real results to global businesses, ensuring a level of certainty no other firm can match. TCS offers a consulting-led, integrated portfolio of IT and IT-enabled services delivered through its unique Global Network™ Delivery Model, recognized as the benchmark of excellence in software development.

A part of the Tata Group, India's largest industrial conglomerate, TCS has over 89,000 of the world's best trained IT consultants in 47 countries. The company generated consolidated revenues of US \$4.3 billion for fiscal year ended 31 March 2007 and is listed on the National Stock Exchange and Bombay Stock Exchange in India. For more information, visit us at www.tcs.com

Contact us at

global.consulting@tcs.com

All content / information present here is the exclusive property of Tata Consultancy Services Limited (TCS). The content / information contained here is correct at the time of publishing. No material from here may be copied, modified, reproduced, republished, uploaded, transmitted, posted or distributed in any form without prior written permission from TCS. Unauthorized use of the content / information appearing here may violate copyright, trademark and other applicable laws, and could result in criminal or civil penalties.

Copyright © 2007 Tata Consultancy Services Limited

TATA CONSULTANCY SERVICES

www.tcs.com