

Governance structure for managing globally distributed work

Rajeswari R. Yudhish K.

Program Manager-Quality Program Manager-Quality
Infosys Technologies Limited, Plot No. 44 & 97A, Electronics City, Bangalore 560 100
rajeswari_rathnam@infosys.com . yudhish_kubendran@infosys.com

Abstract

Defining a Governance model that gives benefits to a large program is always a challenge. Mature Governance Model defines the model that has been defined, modeled and piloted with excellent business benefits to Infosys for one of their Tier 1 clients. The model provides the way projects need to be governed in and across the global scenario with various group of activities being done in different locations within and across countries. This ensures that the projects with multiple tracks that are being executed by different groups in different locations are delivered on time to the global clientele, with quality in terms of processes and the delivered product. It is understood that all project parameters, Schedule, cost, quality, functionality, usability are delivered to the client, through continuous monitoring, taking actions and continuous improvement to reach a point of excellence.

Keywords

Governance, Risks, Tracking, Decision, Authority, PMO

1. Introduction

This paper provides a model that has been derived based on the need to track the challenges of a global project; some of the challenges to be mentioned are the Time and work culture differences, Technology and skill set mismatches, differing customer perspectives and understanding of business importance, continuous communication and information flow. Absence of a global methodology that addresses the project data flow – process, product data, and multiple points of decision making and tracking also have led to this model creation.

This model takes inputs from the best frameworks available in the market like CMMI, PMBOK of PMI and has combined the industry wide best practices and models to re-define/tailor some of them as per need. It is generalized so that it can be used by projects where Global delivery is planned.

Projects executed in the industry are mostly of a service delivery nature, where in the requirements are elicited, the product is created to address the requirements following processes, which might be followed by a maintenance and support phase. Also there are cases of projects where this normal behavior is changed and this project is one of its kinds where in we have an existing Banking product that has to be enhanced to suit the requirements specified by the

customer, to be delivered as a completely integrated solution to replace the customer's existing Internet Banking application. At the same time, this difference is not a restraining factor for the model which can generally be applied to any project/solution delivery scenario where process and product quality are given utmost importance apart from the functionality.

Management forms the binding factor for the various delivery tracks of Resources, Technology and processes hence a mature management model helps projects and program across both the delivery models.

2. General Challenges faced by Global delivery

Most of the current projects have a hardware and network vendor different from the software vendor and in some cases wherein we have large size programs the Software companies out source the work of program management for IT projects, hence increasing the complexity further.

Process complexities, low/mismatched team skill levels, differences in the maturity of development teams and project management teams, socio cultural differences, technical complexities, usual Business complexities, management of financials and margins add to the challenges. The organizations are having a strong review of TCO and ROI these days. Adding an additional resource is not an option any more, since knowledgeable workforce is scarce; it is usually very costly to train them, but even if they are trained, it can never replace experience completely.

In most of the cases, the customer is not the end customer and the end customer's IT knowledge is very much different from the teams involved in that IT project. Customer is to be looked at from different levels and perspectives.

2.1 Customer expectation

The customer expectations for solutions that are delivered as a globally distributed work are much higher than a normal small project executed locally. The complex expectations cannot be executed by a local team or a team centered in a single location or within a single country. Hence there is a need to not only meet customer expectation using a global delivery model but also to ensure the execution is controlled and well planned.

In the cited case here, enhancements to the product displaces a complete application development, testing hence needs to take care of Regression to a large extent, which means that there are activities such as integration of enhanced modules to product baseline, management of latent defects, multiple releases and conflicting customer requirements which are additional challenges in the product scenario.

Solution Delivery	% of project
Existing product	75
New enhancements to product	15
Customization	8

Integration, Migration, Performance	2
--	---

Table 1

Once the site is live, the customer has a strong IT team that would take care of support and maintenance in which case product quality metrics is very important, parameters such as Cohesion, coupling have to be completely incorporated, whereas complying to such requirements from a product perspective is a costly venture since there could be a complete set of sources existing that could not get changed to suit the customer requirements. From the above table (Table 1) it is clear that the existing sources are different from new sources, hence this complexity of complying with a specific PQM (Product Quality Metrics) standards is to be done diligently.

Customer maturity is also a challenging factor since many of the European and American customers are tech-savvy and would expect the software delivery to comply with their standards and specifications.

2.2 Political

The political demands for a global solution delivery model are not explicitly stated or documented, though this is a basic underlying factor to be considered for the delivery. Some of the requirements such as localization, culture have to be taken as basic.

The project is to replace the existing Internet application of the Bank with a new product but with the same features available along with enhancements. Since it is the replacement of an existing application with a large end user base of 4M customers in an advanced market, the UI and functionality has to match with the existing product.

There are 2 challenges from here –

1. To get the same look and feel so that the end user does not see any difference
2. To get the same performance to support 4 M users by specifying the required scalability.

2.3 Demographics

Demographics include regularity, language, end user perspective, knowledge levels of the targeted end users. The profile of a customer in different locations is different and so the global delivery needs to satisfy all the requirements.

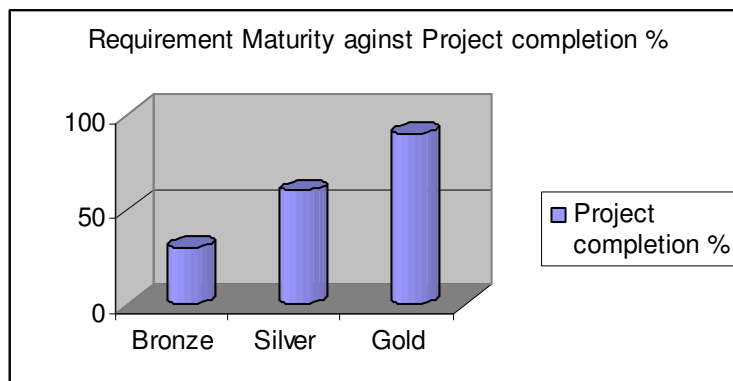
The customer has client base in 2 countries, Australia and New Zealand, with regular requirements being different and usage profiles varying – for example the usage of browser in both the countries is different.

The project is inherently complex, being a financial application open to the Internet that needs to connect to an existing backend banking product to extract and send data, requiring customization.

Security on the net becomes a crucial factor and needs to use the cutting edge technology tools for authentication and authorization apart from the in-built security features.

The requirements for the enhancements have been defined by the Bank's IT team, which has written the used cases for the same. This needs to be reviewed and approved by the Product Management Group, to basically filter out the existing functionality and to approve the model of development for the enhancements as either new enhancement or customization. The continuity of the product cannot be lost since even after this project is completed the product will exist and will need to get enhanced more in future.

The customer, apart from providing the used cases, has also qualified them as per the maturity model of requirements which means that the project team needs to plan for changes to the project scope and functionality based on the maturity of requirements and also on the inputs provided by the usability features which is being done by another vendor.



2.4 Social / Environmental

Project/Program management is complicated due to the fact that the staffing is done globally but the socio cultural differences have made the operations difficult. There is no clean division of these aspects, as normally the countries involved would be kept apart except for the data/info flow between them but due to inadequacy of skills and resources some of the resources work across country which cannot be a limiting factor in a flat world. But at the same time governance of such environments and their impact on the project needs to be done at a cut above the rest, so as to mitigate the foreseeable risks and minimize the acceptable ones.

2.5 Technical

Based on the customer requirements a technical solution, architecture, framework, delivery model are to be decided. A plain vanilla waterfall model cannot satisfy the needs for a globally delivered work. Instead the technical feasibility and execution has to be managed effectively after being setup in accordance with the needs for the global delivery.

The delivery model to be followed is incremental, iterative, since the requirements were evolving. In addition the Usability features were outsourced to another vendor, the inputs from which will come in increments. The requirements maturity model means that change requests will be the

order of the day even though their impact on financials is minimal. This is not only a management complexity but also a technical complexity since it demands that the application be built in a robust, reusable architecture so that when the requirement changes occur, it has to be managed by minimal changes to design.

- **Non component development methodology :**
 - **Water fall model**
 - **Functionality based development**
 - **Followed typical development cycle – RA, HLD, LLD, coding, unit testing limited to functionality testing, Integration testing and delivery for ST/SIT**
 - **Newly developed functionality was integrated to existing product post unit testing and before IT**

Customer involvement in the design and coding phases is also more, where the technical architect will review and approve critical use case designs and then do a code review of some of the source code. Product quality metrics to be followed has been specified up to the number of lines per method, penalty applicable to even the smallest non-compliance.

Being an existing application go live should be planned to the finest details, so that the end user does not feel any impact. Before go live the database migration should be done perfectly since the financial details have to be 100 % correct.

Being an Internet application which is supporting a large customer base - security and scalability are of utmost importance. For the functionality part also, the application was to be thoroughly unit tested for all units before integration, which is an issue if the project's scope was to develop incrementally but deploy complete solution along with the existing product.

On the usage of technical and management tools also a large variation was found, with the customer insisting on Clear case, ART (for risk tracking), Rational test suite for managing testing/defects.

2.6 Management

Project management of such complex programs is not very straight forward, since the prerequisites before the start of the program are already complex. Unless effort is spent to plan for a thorough management model, execution and monitoring of the same, providing the opportunity for plugging in the improvements identified at various stages – it is very difficult to do changes to the management model at a later point in time. Governance model also needs to be setup clearly to manage the work before the program planning and execution starts.

There were many Management challenges:

- Estimation – do we estimate the incremental development or size the complete product? Something that was not done earlier.
 - How do we track the different countries, groups, units and their work?
-

- What data model we should use? Is it only process metrics effort schedule cost defect, if so is it using EV/PV, SPI, CPI, and DPI?
- Training – not only in technology but even product training for new comers to be provided.
- Tools procurement, training and usage
- For work that is outsourced, do we consider them as suppliers and follow Supplier Mgmt or consider them as partners and manage using a part responsibility model?
- How do we track dependencies – if defects are higher in terms of injection – should we reduce it or should we strengthen testing?
- How to consolidate – on what levels and how much of data? Meaning, does the customer steering committee need to know that there is a slippage in one of the projects or should they know only the overall?

The delivery model from an overview perspective is given in Figure 1.

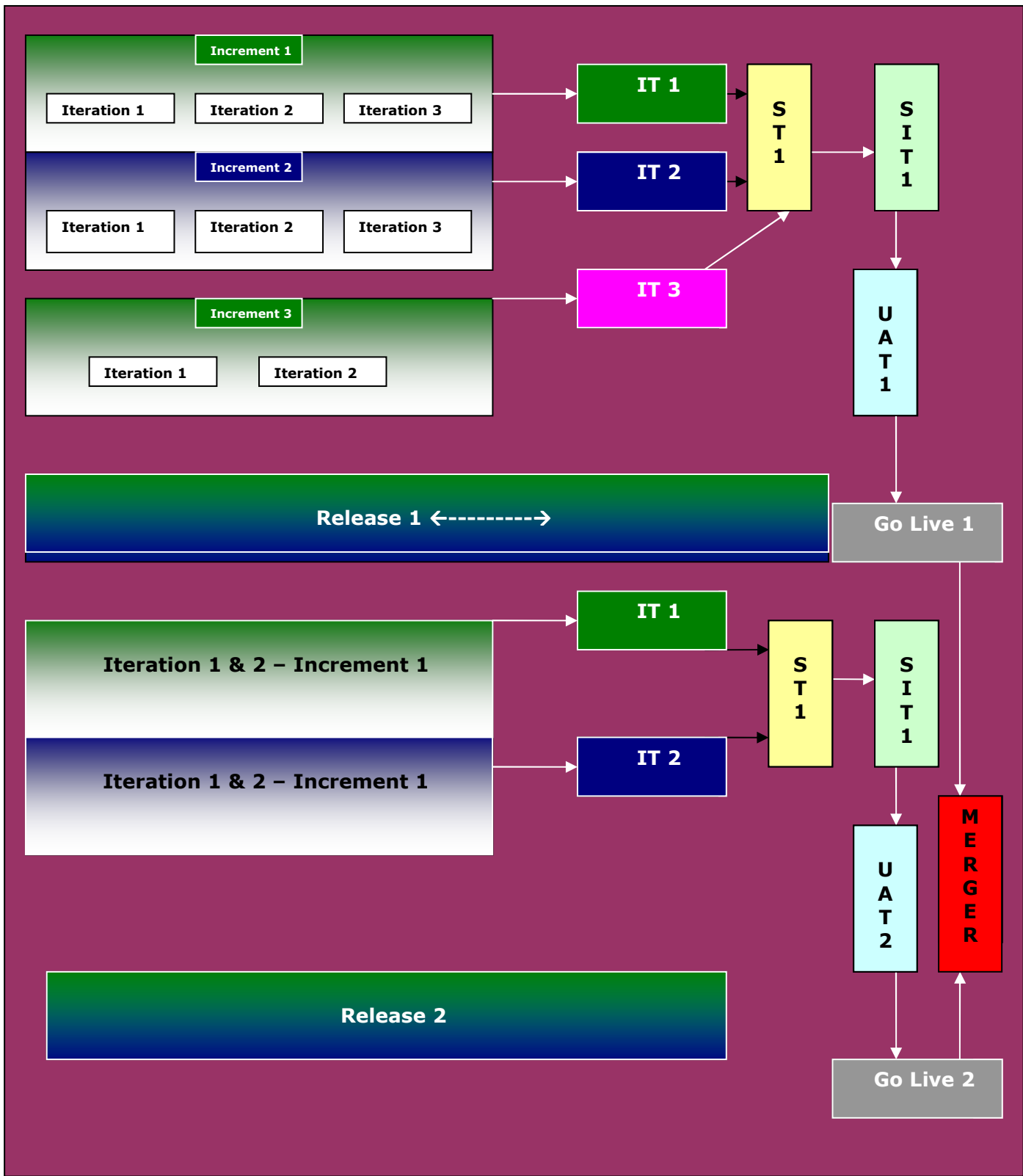


Figure 1 Committed delivery model

The delivery team structure is as follows:

Countries: India and Australia Development centers.

India: Bangalore and Pune locations.

Requirement Analysis by Product group and development by Product group + 1 development team from another unit in the company with J2EE knowledge.

Testing is done by Testing unit of the company. Performance is the joint responsibility of Australia and Bangalore teams, with Australia taking the responsibility of solution Integration (SI).

3. Pilot Model – Addressing challenges

The model of delivery and Governance was integrated with the customer completely in a collaboration model since the status from the usability and hardware vendor is needed for the completion of the delivery. In order to fully understand the challenges a POC was undertaken and the details were shared with customer, this helped the project to have an idea of the complexities involved better and plan for it.

3.1 Customer expectation

A clear distinction of the product changes and the customer specific components like customization was made. This was possible by including the product management team, who understood the used cases separately and decided which needs to be taken as a product change and which as a customized component. The management of both the components was separate from Engineering as well as managing them in source control system. The planning was to take care of the integration of both of them for the customer, but this also enabled the product to be maintained separately from the customized components.

Solution Delivery	% of Delivery program	PQM	Sources/Artifacts	Management
Existing product	75	No PQM, executables delivery	In VSS	Regression testing, impact analysis
New enhancements to product	15	No PQM, executables delivery	Product VSS	Separate module level projects
Customization	8	Fully PQM compliant	Customization VSS	Separate project
Integration, Migration, Performance	2	Planned from design	Strong build process	Separate projects for each area

Table 2

From the earlier stages, it was identified that the integration and build need to form the base of the project and is to be planned well ahead and executed.

Customer handover is planned and tracking PQM metrics has been identified as important from initial stages.

The challenges from the above model indicate that the individual components are managed as a separate project end to end – from Requirements to the delivery completion.

Referring to the proposed model of Governance, this is at the lowest level, which is purely project management driven. The challenge is to ensure that the data from each project is available from which the overall program status can be derived. Additional tools and processes are to be set to ensure that this data flow is smooth and can readily provide the consolidated status.

A tool for managing the performance in terms of SPI, CPI, DPI has been formalized that can be utilized for the tracking and to provide the information to the Governance team.

3.2 Political

A strong focus on the functionality and Performance Engineering has been planned right from the beginning.

In order to manage the performance and the UI requirements, which are key to the program, the same approach has been followed, viz, treating the UI and Performance track as separate projects. The key to manage these projects are:

- Arrive at a tailored Engineering process for UI/ performance. The challenge is enormous since UI per se does not have specific requirements, whereas in the case of performance project there is a criteria set in terms of the hit rate, number of logged in users etc – there is no design/development that happens separately.

- Arrive at a tailored project management process – the parameters for monitoring these projects is entirely different from others. An example is that instead of looking at the DPI for the performance project we may need to look at % adherence to throughput/scalability against the goal setting.

The goal setting itself will change based on the profile of the project.

3.3 Demographics

From an overall project management perspective, the project has been planned as a series of deliverables in accordance with the requirements maturity model, meaning to say that the change management is part of the Development activity. The typical project flow of requirements, design, and build test has been altered to include a Change management track at the required intervals, such as the compliance to Silver requirements planned at the completion of the detailed

design of the bronze requirements; this should overall be able to cover the requirements process proposed by customer.

The challenge here is to ensure that the design is robust so that further changes to the requirements do not destabilize the current build, which requires that there are specific change management processes and build and testing processes.

Project management has identified various sub processes that need to be addressed and planning has been done based on the same. Some of the processes defined and deployed were

Change Management process, which was specified for the project with individual responsibilities.

The Configuration management process was tailored for the project with build timings on a daily basis, including source sync up plans.

Defect / Risk tracking processes, which were detailed out for the project specifically. This would address both testing and the risk management of all involved projects.

For the PQM and SPC processes, we have used a tool to track individual projects and to consolidate data. Wherever possible existing/new tools were used to automate most of these processes.

Some of the processes to be mentioned are TSP and PSP, Scrum model of development, for better understanding of the team on the changing requirements.

3.4 Social / Environmental

The best model for managing such a complex socio-environmental situation is to keep the resources at their respective locations and make the Governance strong so that it serves as a bonding factor and also a direction providing mechanism.

The problem was addressed by making a clear distinction between people and their location/roles. We have allocated resources based on their base location so as to avoid lot of travel and social issues that would arise due to change of location. Instead the communication/reporting/escalation mechanism was strengthened so that the required details were available across the board. On the other hand the development team was trained sufficiently so that their dependency with the other location teams was minimal – a tailored was created on how to handle work handover due to the difference in time between countries. To show this perspective in a clearer format, refer below:

1 team combining people – all the socio/economical issues are present.

Alternative is:

1 team for module 1 team for location 1 team for unit (testing/functional) and a Strong management team

We would thus address this as the “Divide and Govern” mechanism.

3.5 Technical

The resource pool was initially large and we had utilized additional resources due to the technical model of development and so training was flowing as a track throughout the execution of the program. At the same time the incremental iterative model compensated for the lack of clarity on the requirements and has given confidence to the team through small increments, where in there was an opportunity to plug in the learning’s. At the end of each LC stage RCA was done to understand all issues and action was taken for the next stage.

Milestones for customer reviews and signoff in the WBS helped track them to closure and plan for their rework upfront. End to end planning including the other vendor dependencies was done and the clean division of WBS between the countries was a good management method – meaning, separate WBS for each country but it was exchanged on a weekly basis and updates were discussed, so sync up was implicit. **Development methodology:**

- **Component based development**
- **Architecture based development**
- **Followed matrix development**
- **Integration was at the end in the first increment and parallel in the 2nd increment**

A separate framework/architecture team was formed which was doing the planning and tracking for the technical items such as testing on other browsers, security, unit testing method, product quality metrics and application build.

The project LC stages to manage the complex technical requirements and the team structure changes to address the same are given in Figure 2.

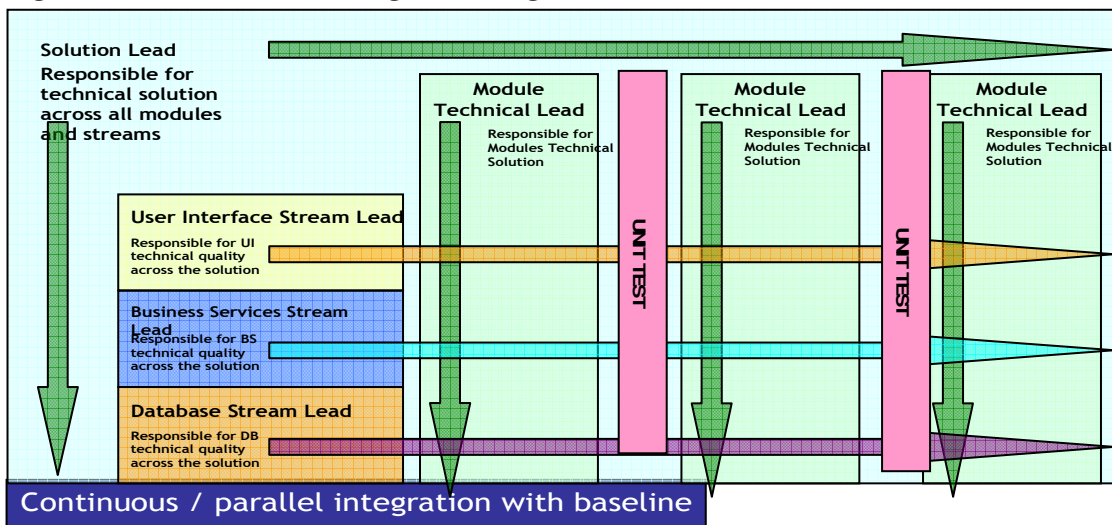


Figure 2 Technical and resource model

3.6 Management

There were many Management challenges:

- Estimation – incremental changes were estimated using FP but existing product impact was used to identify the final effort.
- Countries, units, vendors and groups were treated as separate projects and consolidation was done.
- Data model used was SPI/CPI/DPI with financials being tracked through effort and Budgeting system.
- Training – not only in technology but for even product training for new comers has to be provided – training was treated as a separate project and was going on as a continuous thread.
- Tools procurement – budgeted initially itself.
- Both the models were not used, they were treated as a separate project and had to follow all the processes being part of the program.
- Strong DAR (Decision Analysis and Resolution) principles using appropriate tools were followed.
- Project management consolidated all projects at country level and Program level was done more at the relationship, overall impact level.

The proposed/piloted model of delivery is given in Figure 3.

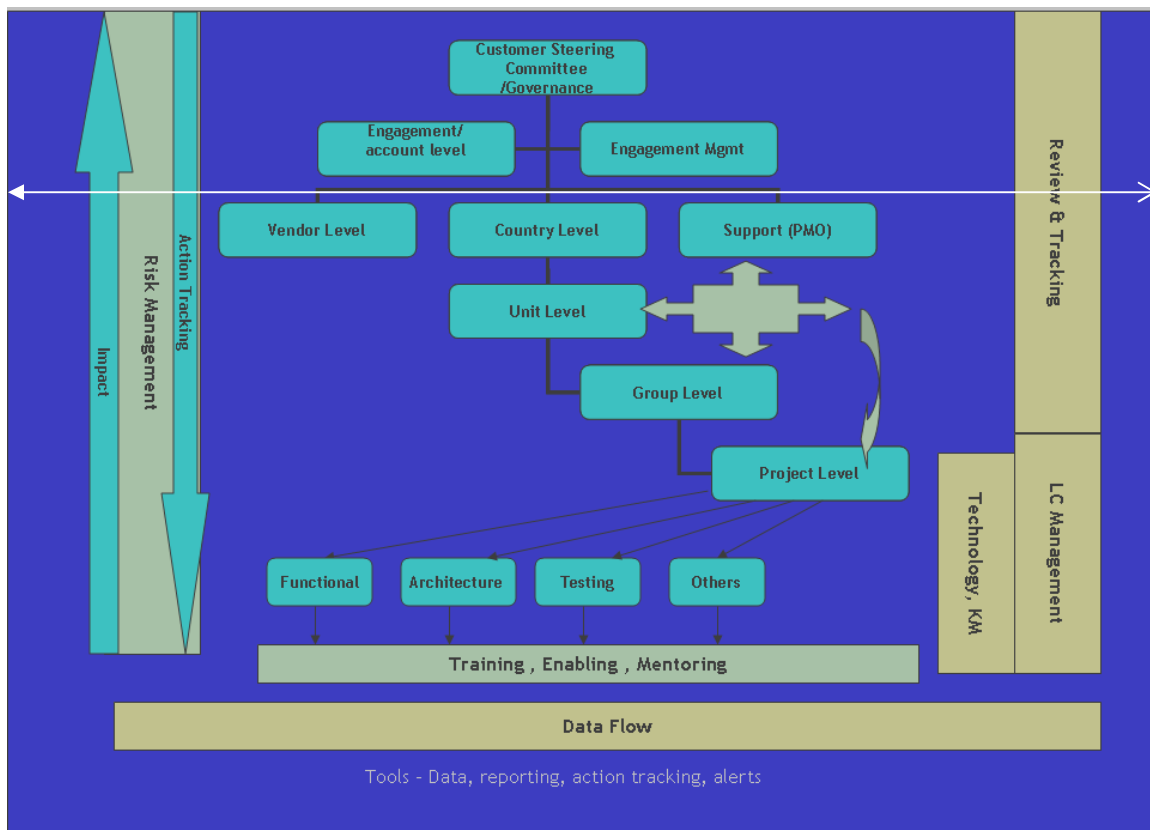


Figure 3 Mature Governance Model

4. Mature Governance Model

As given in the figure above, the model clearly distinguishes between projects and Management. There are tracks such as data and information flow that cut across both. LC management is done at project level depending on the type of delivery from the project. Above the vendor level, country level and support level is pure management activities and Governance that gives direction and manages high level risks, planning and tracking against overall goals. Below this is project management and tracking. This is shown as the dividing line in Figure 3. The idea is to separate project management and tracking from Governance of the complete delivery.

An example can be provided to explain this clearly:

A project level risk such as 2 days delay in completion of HLD as against plan would not be risks tracked at Governance level unless this 2 day delay would result in a delay in one of the customer deliverables say the “HLD document delivery for review to customer”. If such an impact is there then the risk / issue would be tracked at governance level otherwise it would be tracked as a risk at project level.

The Governance team also has the authority to make decisions as per the current status of the project. In case of providing for additional resources, such as VPN links, human resources, trainings, the Governance team has the authority to provide for the same. The Governance team is different from the PMO, which does not have the authority to take final decisions and may not influence the provisioning for the program in terms of facilities and resources.

Also note that the Governance team is responsible for the relationship management with the customer, by providing the regular updates and raising concerns. The Governance team has a higher responsibility since they are seen not only as representatives of the senior management but also as the representatives of the customer. The Governance team needs to balance between the well being of the business interests of the company and the satisfaction of the customer – both are equally important and cannot be negotiated.

5. Project Level Management

Multiple projects are executed – One for Requirements tracking, another for Change management, another for Development activities from High Level Design to Coding and unit testing, another for test planning from requirements till execution of System Testing, System Integration Testing, User Acceptance Testing, another project for architecture enhancements and tools, Performance management and UI.

In this case, it is critical to provide the consolidated view to the Governance team to facilitate them to make the right decisions and communicate to the senior management and customer. A manual process is prone to errors and may not be uniformly done. Hence there is a critical need for automation of data collection, collation as well as consolidation.

Hence there is a need for a tool that can be used by projects to get the details, analyze and take decisions at project level. This info has to be consolidated by the tool at a program level which will be used by the Governance team after vetted by the PMO.

A tool for the same has been built, configured for use for such a requirement. The tool provides the vital information at the project level and then gets consolidated at a program level.

A consolidated view of the projects is provided which will be shared with the Governance team as follows: (ILLUSTRATIVE DATA ONLY)

Project	Stage	Metric	Date 1	Date2
Project 1	Dev	Total Estimated Effort	1145	1809
		Planned Value	1116	1423
		Earned Value	998	1416
		Actual Effort	1547	1694
		Estimated Completion %	97	79
		Actual Completion %	87	78
		Pending Effort	147	393
		Estimated Defects	267	358
		Actual Defects	112	0
		Projected Defects	157	126
		SPI	0.89	1
		CPI	0.65	0.84
		DPI	0.59	0.35
Project 2	Testing	Total Estimated Effort	1926	2019
		Planned Value	1458	1579
		Earned Value	1207	1393
		Actual Effort	1124	1355
		Estimated Completion %	76	78
		Actual Completion %	63	69
		Pending Effort	719	626
		Estimated Defects	402	542
		Actual Defects	92	0
		Projected Defects	293	542
		SPI	0.83	0.88
		CPI	1.07	1.03
		DPI	0.73	1

6. Benefits at a glance

Finally the proof of the pudding is in eating it as the saying goes – the Mature Governance Model has been piloted and has provided the below benefits.

The benefits cannot be from the Governance model alone but owe a lot to the way the individual pieces have been setup and managed.

However, here is an update in quantitative terms on the benefits achieved (**ILLUSTRATIVE DATA ONLY**):

• Module Integration Testing – Estimates Vs Actual		
Estimated	Actual	% Improvement
1001	712	29%
Estimated(Sev # 1&2)	Actual (Sev #1&2)	% Improvement (Sev# 1&2)
266	187	30%
• System Testing – Estimates Vs Actual		
Estimated	Actual	% Improvement
1092	561	46%
Estimated (Sev# 1&2)	Actual (Sev #1&2)	% Improvement (Sev# 1&2)
328	51	84%

References

1. Global delivery model and business benefits; How do they compare with outsourcing? Difference between a global IT delivery model and an offshore outsourcing model
<http://www.itbusinessedge.com/>
 2. Project Management
http://www.managementhelp.org/plan_dec/project/project.htm
 3. Global delivery model, concepts, need, advantages and associated risks; Project stages and customer expectations
<http://www.offshore-softwaredevelopment.com>
 4. PMI guidelines and best practices
<http://www.pmi.org/info/default.asp>
 5. PSP and TSP
<http://www.sei.cmu.edu/tsp/introducing.html>
 6. What is good Governance?
<http://www.unescap.org/huset/gg/governance.htm>
 7. CMMi Model
<http://www.sei.cmu.edu>
 8. Project Management Process
<http://dijest.editthispage.com/tools/pm>
 9. What is governance as applicable to IT and Definitions?
http://en.wikipedia.org/wiki/Information_technology_governance
 10. Management Review and Responsibility
<http://www.infosys.com/investor/corporategovernance.asp>
-