

Application Rationalization -

The starting point for SOA



Overview

With the economic downturn and strong global competition, there is a clear need to prioritize application investments and measure what value they deliver. Such an activity when conducted on an ongoing basis (typically, an annual exercise), enables arriving at key decisions which include: *maintain, enhance, retire or combine*. Leading Industry research firms have found that an Application rationalization exercise reduces upto 25% of the operational costs, right at the beginning. This fact is hard to dispute, given that in many enterprises, legacy applications are kept in operation beyond their useful life spans, there are years of unstructured updates on systems and the cost of maintaining such applications form close to 80% of the entire IT budget.

Application Rationalization is a fairly standard process comprising of an end-to-end analysis of the application portfolios in each domain, identifying the overlapping functionalities, unused applications and bottlenecks in systems. Such analysis involves ascertaining the relationship of each asset to process, function, capability, data input and output. Some standard aspects to be examined are:

- The business value of these applications
- Amount spent to maintain, support and upgrade these applications
- The processes these applications support
- The way in which these applications are being used
- Users of these applications
- Performance and scalability of these applications

Application rationalization would be the first step to weed out redundant applications and overlapping functionality. This exercise would also produce artefacts and information that will typically help ensure the stack *stays lean* and serves as the gateway to a service-oriented architecture (SOA) implementation. The consolidation of functionality and exposure of commonly used logic through a service-oriented approach would enable finite and limited changes to smaller self-contained parts while the whole remains relatively intact and unaware of the changes.

This exercise of managing applications as assets, eliminating liabilities and laying the groundwork for future-proof IT initiatives such as implementing a service-oriented architecture (SOA) requires strong IT governance. A governance team would need to be formed with members from the vendor and customer IT teams, to ensure continuous improvement. The effort is typically managed as a project to achieve superior results. The effort would need to be conducted from an enterprise perspective with an unbiased view of the application portfolio.

Objectives

Broadly, Application Rationalization and transformation caters to the following objectives.

- ❖ Save operational costs
- ❖ Simplify the application landscape
- ❖ Expedite deployment of new services
- ❖ Minimize cost of deployment for new services
- ❖ Ease operational support
- ❖ Reduce points of failure
- ❖ Clear SLAs
- ❖ Predict performance and scalability more accurately

Challenges

One important factor to be considered while rationalizing the application stack is that these applications do not exist in silos. They are mostly legacy systems or applications with years of unstructured updates. They are used as a part of multiple business processes and have complex interdependencies among themselves. Many such applications have significant financial stakes. Changing a set of such applications that are interdependent and critical to business is like changing parts of an aircraft while flying! It requires sufficient experience and skills to ensure such an initiative is carried-out to achieve the targets set out. Some common challenges are:

- ❖ Typically, there is no single owner for business processes and IT assets. To establish stake-holders and stakes for the applications being rationalized is fairly challenging in itself.
- ❖ Getting a buy-in from the higher management is a necessity for such an initiative. The higher management must be willing to champion the process and articulate its benefits. Coming up with a clear business case and getting a buy-in from the management is several times a challenge.
- ❖ There is another dimension to application rationalization. The new stack must not only be leaner but also *stay lean*, while it continues to provide enhanced functionality in an evolving business environment. How can this be done? The approach would have to focus on reuse, on agility as defined by the ease with which the newly created assets can be adapted into different forms, to cater to as yet unknown requirements.

SOA, Application Rationalization and Agile

Application Rationalization is typically a prerequisite to SOA. It strengthens SOA by delivering a lean Service inventory for the organization. Such an effort not only eliminates redundant applications but also enables indentifying duplication of functionalities. Building the commonly used functionalities as shared external services through SOA enables systems across the enterprise to call these services and thus encourages reuse.

Both Application Rationalization and SOA cannot be implemented using a big-bang, waterfall approach. It makes absolute sense to implement these initiatives using an Agile methodology, considering both these approaches are ongoing and critical to business, combined with the fact that they require a buy-in from different stakeholders. Agile requires that prioritized functionality be implemented in a time-boxed, iterative approach to realize a quicker Return on Investment (ROI). This enables establishing confidence in the project early on, given that the business can see and experience working code in the early stages of the initiative.

Additional skills and cost benefits could be explored when such initiatives are implemented using a multi-shore delivery model. Negative perceptions still surround distributed agile development, many of which stem from the lack of frequent face-to-face communication. However, cooperation between individual teams located at different sites may be the reality today. Best practices have emerged or rather evolved, on working together over a network, either in the same geographic location or across locations.

Application Rationalization Process

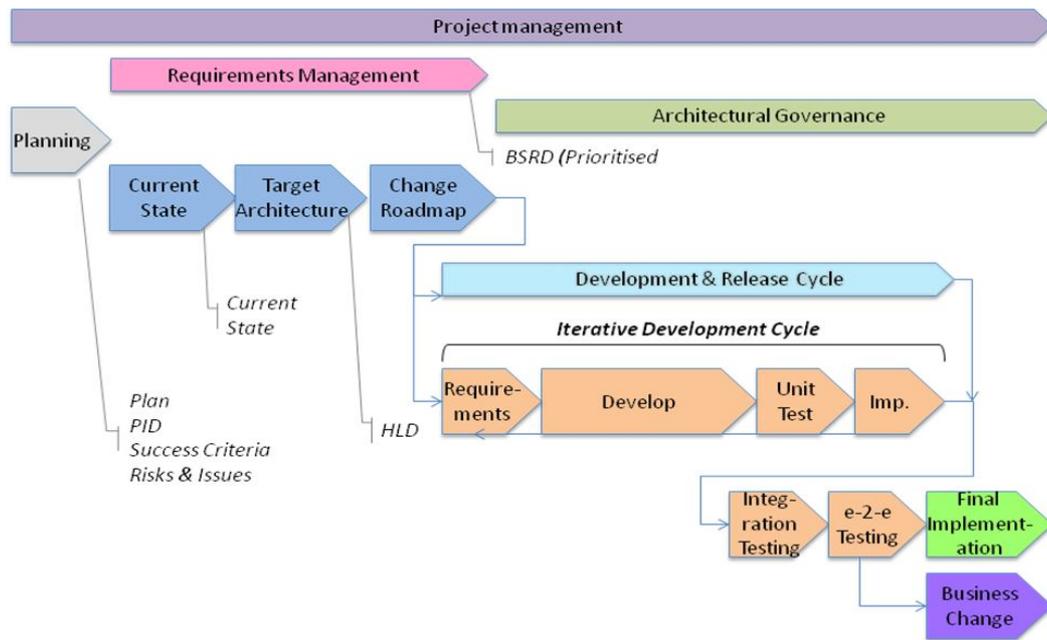
An application rationalization process typically involves the following phases:

- 1) Asset Identification: Each application in the portfolio is to be evaluated to determine its value and its costs, with the help of workshops and analysis activities. This helps in identifying the KPIs and metrics to measure the ROI on these applications.
- 2) Defining the Target Application Portfolio: With the help of KPIs and metrics identified earlier, the key service capabilities are analyzed and the final set of applications and business functionalities are defined. This is the first strategic level where business strategy stating the target application portfolio is defined.

In order to demonstrate early benefits, we define “biggest bang for the buck” and identify “low hanging fruits”. So decommissioning of the unused/deprecated applications will be the first activity.

During this process, governance practices and guidelines are defined to ensure that all ongoing and future initiatives are aligned to the target application portfolio.

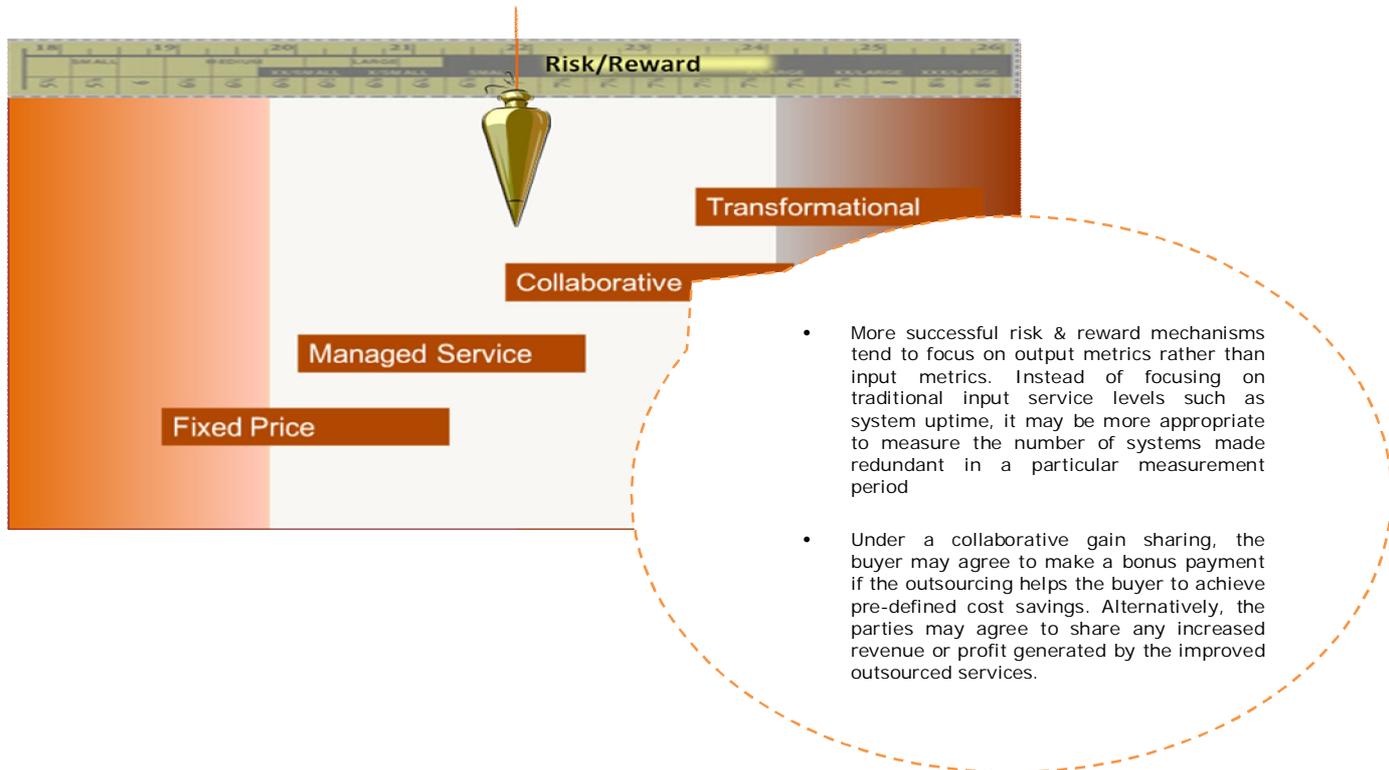
- 3) Defining Rationalization Roadmap: This is the second level of strategic planning, where strategy and iterations are defined to transform the select set of applications for each iteration. This helps to avert / de-risk a full-blown, waterfall approach to transforming the systems. Each iteration in this phase drives the organization closer to its target application portfolio.



- 4) Alignment of Initiatives: All ongoing IT initiatives and business processes would need to be completely aligned, before finally decommissioning the listed applications and reaching the target application portfolio.
- 5) Application Transformation: In this phase, various identified applications undergo transformations to match the target application portfolio. Business processes also need to be re-engineered in this phase in order to adapt to application changes. This phase needs to conform to the organization's change management process.
- 6) Metrics and Feedback: This is the most important phase to ensure consistent success. Metrics and KPIs established earlier on need to be constantly evaluated and revisited to ensure alignment with the target application portfolio. This is done through effective governance practices. The Governance team also ensures future business initiatives conform to the Application Rationalization framework.

Exploring a Gain-share model

A model that enables "owning the outcome" and helps unite the goals of the implementing team (a vendor in many cases) and the organization would be beneficial in such initiatives.



Some characteristics of the 'transformational stage' below:

- ✓ Transformational risk & reward mechanisms take the collaborative approach to the next level. They measure the success of major transformational projects and align incentives with enterprise-level outcomes such as market share or return on capital. The transformation might include the development of a new IT platform to support a new line of business or product. If a transformational project includes a significant new software development, as well as sharing in any increased revenue or cost savings associated with such transformational project, the parties may agree a mechanism to exploit jointly the intellectual property created as part of the project.
- ✓ Gain sharing in this context can only be effective if both parties understand their responsibilities for realizing the benefits of an implementation. It also needs to be made possible to quantify the benefits that might be derived from a particular implementation. This can be done either by agreeing to the extent of the cost reduction up-front at the preliminary design stage; identifying the cost that the project will affect; or using agreed mechanisms to quantify the benefits. Thereafter, it is necessary to agree upon the basis on which gains will be shared, likely using some form of banding arrangement.

Conclusion

Initiatives like Application rationalization and SOA reveal many opportunities for savings and improved efficiencies. When the initiatives have a good governance framework, it ensures follow-through on investment decisions. Tedious as the early stages of such initiatives may be, they begin to pay dividends in-terms of business agility, clearing the way to implementing more strategic business decisions.

Torry Harris Business Solutions (THBS) is a US based IT service provider with development facilities in India and China. The company, started in 1998, has for several years delivered a large variety of middleware services to enterprise clients around the world. Now, with a large pool of highly skilled technologists and rapidly growing, the company remains focused on the middleware and integration space, implementing large projects across the US, Europe, the Middle East and the Far East. The company is committed to Service-oriented Architecture(SOA), which it sees as the logical movement to follow the phenomenon of distributed computing in the late nineties, where THBS was clearly the market leader in implementing the offshore/onsite delivery model.

For more information, write to us at soa@thbs.com.