How Can Companies Minimize Total Cost of Ownership for Enterprise-Wide Software Solutions?

Knowledge Transfer Basics - Part 2

by Chris Kay

Enterprise software solutions, especially those that solve innovation and product development challenges, are difficult to implement and even more challenging to sustain. With most enterprise solutions in production for more than 10 or 20 years, Total Cost of Ownership (TCO) for operational execution is almost always much more expensive than development and implementation of the original solution. The real challenge is balancing the need to minimize costs while doing everything possible to ensure success and reduce risk. And of course, the solution must still meet the requirements and address the business problems that lead the company to implement in the first place.

Minimizing TCO – with business drivers in mind – starts in the earliest parts of planning for innovation transformation: as companies develop an overall innovation strategy and as they assess their organization’s current capabilities during the first phase of an implementation project.

There are certain strategic, architectural, lifecycle, and software engineering decisions that impact the TCO of the solution. These include:

- Defining the business drivers and metrics the solution will address, and establishing a baseline of current performance
- Determining the appropriate level of information or data that should be shared across the business
- Determining the appropriate level of process standardization required across the business
- Aligning with enterprise-wide IT architectural standards, especially in the areas of complexity and change (specifically platform extensibility and upgradability)
- Understanding how the solution aligns with your enterprise architecture standards

Once the strategy has been approved and a specific project is being planned, the first phase of the project is when user requirements are specified. The deliverables in the first phase that impact knowledge transfer include non-functional requirements like testability, maintainability, extensibility and scalability, as well as definition of the operational owner of the solution.

In terms of TCO, costs associated with non-functional requirements can be reduced by:

- Defined use of the “V” model, coding standards, architecture, design and code reviews
- Use of a company-specific software development lifecycle (SDLC) process, templates and tools
- Use and reuse of code libraries to minimize the development of new code and maximize reuse, while standardizing behavior and performance

To ensure the operational support team maintains focus on these decisions, implement an ongoing post-implementation measurement system that measures and rewards the realization of the business drivers used to justify the project.

More Reading

Video - How to Invest Users in an Enterprise System: The Basics of Knowledge Transfer

How can Companies Assure Operational Success of an Enterprise-Wide Software Solution?

How Can Companies Minimize Total Cost of Ownership for Enterprise-Wide Software Solutions?

How can Companies Retain the Knowledge the Software Development Team has Built into a Solution?

Originally published on October 5th, 2015
What's your view? Add your question or comment
About the Author

Chris Kay
chris.kay@kalypso.com

Chris brings 30 years of experience in R&D, product data management (PDM) and product lifecycle management (PLM) to Kalypso's clients.