



Top 10 Tips For Legacy Rejuvenation



White Paper
January 2009

Introduction

In these times of financial upheaval legacy IT platforms can easily become a beacon of inefficiency, as they consume more monetary, physical and human resources than their modern counterparts. However, they support important applications, services and data, otherwise you would not be running them, but this often implies a risk of system failure.

With capital expenditure on replacement applications being limited in the current economic climate, many organizations will now be re-looking at their existing legacy systems in order to add functionality rather than replace them. Areas to be investigated typically include providing better access to the data, re-using the tried and tested business logic to deploy services within a Service Oriented Architecture (SOA), or simply moving these applications to more current cost-effective hardware.

For over 20 years Transoft has been helping organizations to migrate and/or modernize their legacy applications, data and infrastructure. Our expertise, automation tools and integration products enable us to work more quickly, less intrusively and at lower cost than you would have imagined.

We can help you to explore:

- reducing your legacy expenditure
- mitigating your risk of system failure
- transforming your legacy platform to become part of your modern infrastructure.

10 Tips to Legacy Rejuvenation

- 1. Assess the business fit**
- 2. Application assessment**
- 3. Define and document the project scope**
- 4. Assess the current platform**
- 5. Assess the target environment**
- 6. Investigate modernization products and services**
- 7. Determine the resources**
- 8. Create the project plan**
- 9. The project implementation**
- 10. Judging the success of the project**

1. Assess the business fit

It is important to understand the value of existing application assets to the business, the need for strategic investment, and the current total cost of ownership of each system before a clear modernization strategy can be determined.

There's no point in considering modernizing an application that no longer meets the basic needs of the business. This application has low value to the business and probably will be considered for retirement or replacement in due course.

But many legacy applications, although rich in functionality and meeting the core requirements of the business, are seen as difficult to modify to meet the changing needs of the business, are less well integrated with other systems and cost disproportionately more to operate. Additionally, if the application operates on a proprietary hardware platform this may be considered as an operational risk.

There are a number of ways such applications can be modernized to become more agile to changing business needs, better integrated with newer technologies, with reduced operating costs and with operational risk mitigated, while continuing to maximise the ROI of the legacy application. These modernization options include:

- Incrementally exposing core business functionality from the legacy applications as Services within a SOA opening up the re-use of this functionality within new agile business processes.
- These SOA Services can be integrated into new client facing applications built in modern technologies such as .NET and Java/J2EE. For example, new Web or mobile applications can be built based on the re-use of the existing application Services. This ensures greater agility to more quickly meet the changing needs of the business.
- Consolidating legacy data sources into relational databases enables easier data access, the implementation of Business Intelligence and better reporting and simpler integration.
- Migration of legacy applications from older proprietary platforms to more cost-effective and better performing UNIX, Linux or Windows servers. This is often the first step in any modernization strategy for such an application.
- Accelerating the development process to create the agility to react to business change. Many development shops still use development tools based on 'green screen' sessions, for example, for COBOL or RPG development. Modern development tools, such as Microsoft Visual Studio provide support for COBOL and other older languages. Moving to a modern development environment can help improve developer productivity.

2. Application assessment

Having established a likely candidate application for legacy rejuvenation, the next step is application assessment. This process is a detailed analysis of the application and usually starts with the completion of an assessment document to guide you through the key elements of the application, including: details of its current platform, the numbers of programs, the database used, the integration points, third party products used, and so on. It will also include sections that cover the business changes you require and the preferred target technology landscape.

Transoft and other modernization vendors can provide these assessment documents and have the specialist skills to help evaluate the best alternatives to meet these new business and technology requirements leading to a modernization road map which may consist of one or more discreet projects.

3. Define and document the project scope

With any modernization project there is always the temptation to attempt to do everything at once such as changing platform, changing user interface, changing database and improving integration with other applications. The most successful projects are those that prioritize these steps and break them up into individual projects and deliver each separately, where possible.

When starting any IT project it's important to document the project scope in detail beforehand. This ensures that the most appropriate solution can be identified and minimises scope creep by enacting a formal change request procedure where the

implications are signed off by all project owners. Without these terms of reference it's not easy to determine progress and judge success or failure.

4. Assess the current platform

For any application modernization project to be successful the long-term issues regarding the current platform should be addressed upfront. This doesn't just mean looking at the current hardware, but also reviewing its future. Is it proprietary? How many years of life will you get out of the current hardware, its operating system and the database? Who currently supports the hardware? Have you got sufficient skills and resources at the moment and are these skills easy to find, now and longer term? Will the platform be discontinued? If you have to upgrade to the next version of a proprietary platform, should you take the opportunity to move to an open systems environment?

5. Assess the target environment

If you conclude you need to move your application off its current platform to a modern environment, then again this isn't just about the hardware. It's about assessing the operating system, languages, user interface, database required, any third party products required and how the application interfaces with other systems. Taking a proprietary language to another environment isn't all that difficult as there are tools on the market to do this, for example proprietary COBOL to .NET COBOL on Windows. However, if you wish to convert the existing application code to another language it's significantly more complex, for example converting COBOL to Java or C#. There are many questions to be posed such as - What will the architectural framework be? How readable and maintainable will the resultant converted code be? Do you have the relevant skills? .. and so on. Code conversion, particularly from a procedural language to an object oriented language is unlikely to be a fully automated process.

6. Investigate modernization products and services

If the start of your application modernization road map necessitates a migration, then having reviewed the current platform and the target environment it's time to start to investigate how you get from one to the other and who can assist you. Has this been done before? Is there an automated toolset to assist? Does the vendor provide these as a service or can they be purchased?

All too often modernization projects have failed due to a lack of planning at this stage. Modernization projects typically start off with very simple ideals, such as replacing a flat file system with a relational database. However the choice of toolset to achieve this can vary from replacing all of the existing I/O statements with embedded SQL which is a very time consuming, labour intensive approach and risky as the business logic may be disturbed, through to a range of middleware solutions that seamlessly translate flat file I/O statements to relational database access, without the risk, cost or time associated with the former approach.

However even these middleware tools differ in their architecture and it's vital to investigate them fully to ensure that the chosen solution will provide the performance and scalability required.

7. Determine the resources

In most modernization projects there will be a mixture of internal and vendor external resources used. Internally, you will need to appoint a strong project manager (PM)

who together with his vendor counterpart will have overall responsibility for project success. It will also be the responsibility of the internal PM to ensure that the relevant application assets, the test plans, testers and other resources are all available when required.

The vendor will usually provide specialist resources to either train internal resources in the use of the modernization tools, should there be 'spare' internal resource, or to directly use the tools to effect the solution. Either way, it is recommended that the vendor provides a PM to ensure the project stays on the straight and narrow.

8. Create the project plan

The project plan will be drawn up jointly by you and the vendor and consist of a minimum of two parts. Firstly, a Gantt chart which should show each discreet element of the project together with the required internal and external resources to achieve it. There will usually be a set up phase, and for a large project (over 3 or 4 months), it is good practice to deliver the implementation in defined work packages, each taking typically two months. Ensure that sufficient time is allocated to both vendor and user testing and at the end of each major milestone build in a small contingency.

The second part of the project plan will consist of a detailed commentary of the tasks and responsibilities of both your organization and the vendor. It should include the extent of the inventory being modernized; the agreed target environment; the detailed steps of each project phase; the number of test cases the vendor is to carry out for each work package; the extent of final acceptance testing and the acceptance and success criteria. In addition, the roles and responsibilities will be defined for each participant, how the project will be monitored and the project change procedures should the scope need to change during the project. Lastly, it is important to document what is outside the project scope.

9. The project implementation

It is essential that the project managers keep on top of the tasks, particularly at the beginning of the project, where seconded internal staff often continue to give priority to their 'day-jobs' rather than the project tasks resulting in immediate slippage if you are not careful.

Throughout the project, the project managers will have a formal progress meeting, typically weekly, bringing in technical team members as appropriate. It will be the vendor project manager's responsibility to provide a progress status report for each meeting and will usually take the minutes to document all decisions taken, actions required, new risks established and the plan to mitigate them, and so on.

After each deliverable, it is all important to monitor any issues that arise from acceptance testing and ensure they are resolved in a timely manner.

Finally, after a successful project it's important to carry out a post mortem to document what went well and what didn't go so well in order to learn from the experience, so that the best practices can be applied to the next modernization project.

10. Judging the success of the project

As part of the project plan the success criteria will have been documented. The primary outcome of the project will usually be economic advantage demonstrated by a tangible

return on the investment. Indeed, having modernization expenditure approved in the first place will rely on the advocate being able to demonstrate a solid financial case.

If the legacy platform is to be replaced there will typically be cost savings in terms of hardware costs, licensing fees, maintenance charges, and specialist skills over a number of years. Most migration vendors will provide a fixed price for such projects with organisations looking for a return on the investment over typically one to two years. If the application is critical to the business then the risk of irreparable failure or extended outage due to aging hardware can be the key driver to undertaking the project as the loss of business or even the business not surviving, in the most extreme cases, cannot be tolerated.

But the success criteria of the modernization project are not all financial. They may include improved performance, ease of use through a new graphical interface and improved data access leading to greater Business Intelligence. Improved business processes may be a goal with legacy applications better integrated in real time straight-through processing with new technology-based (.NET or Java) customer facing applications (Web or mobile applications). These improved processes lead to reduced errors, improved quality and timeliness of data for staff and/or customers, and better supported sales and marketing initiatives. Clearly such improvements can affect revenue rather than costs, but they may represent the dominant economic impact.

About Transoft

For over twenty years, Transoft has been a leading provider of innovative and pioneering modernization solutions that enable organizations to increase business value and maintain competitive advantage by maximizing the potential of existing applications. This provides rapid return on investment, reduced costs, improved productivity and efficiency, and the ability to manage operational risk.

Major organizations such as The Gap, L'Oreal, Boeing, Balfour Beatty and Christie's, and thousands of other customers around the world have enjoyed the business benefits of a Transoft application modernization strategy. We work with a large network of VARs, System Integrators, ISVs and technical partners to offer unparalleled solutions.

Transoft is part of IRIS, the largest UK privately-owned specialist software business with an exceptional reputation for delivering award winning, market leading solutions to more than 60,000 organizations worldwide.

www.transoft.com
newsolutions@transoft.com

Europe & Rest of World Tel: +44 (0) 1753 778000
North & South America Tel: +1 (770) 933 1965

Transoft is a trading name of Computer Software Group Limited, which is part of the IRIS Group. Transoft and IRIS are trade marks. © Computer Software Group Limited 2009. All rights reserved

Copyright Transoft 2009