EXECUTIVE SUMMARY

MICRO FOCUS

DePaul University
Transformation of Legacy Systems

It's up to 25 times faster to Web-enable a legacy system using EnterpriseLink than it is to migrate languages or hand code.

Introduction

Results Show Significant Cost Benefit of COBOL Application Reuse Over Java Migration

In the relatively short history of the Internet, many organizations made expenditures for Web applications without regard to cost or return on investment (ROI). However, in today's business environment, senior business managers no longer see the Web as a panacea or a must-have at any price; their understanding of the web has matured. Rather than focusing on the dreams of instant e-commerce, many enterprises are putting the emphasis on increasing the efficiency of supply chains, improving customer relations and reducing operating costs by tightly integrating their e-commerce applications with existing core processes and systems.

Today's economy demands Web and e-commerce projects to utilize existing systems and applications in order to lower the cost of ownership and increase return on investment. A key to achieving this goal is the implementation of technology that provides automatic generation of new presentation and programmatic interfaces that recast existing business processes and rules to new platforms and environments. This study, conducted by DePaul University, presents additional evidence that certain methods of Web-enabling legacy systems can rapidly produce a significant increase in ROI, application usability and flexibility, as well as improving competitive advantage.

Overview

The methods chosen include:

1). Application Process Integration (API/typified by Micro Focus EnterpriseLink® [EL])

2). System migration to Java

In mature mainframe-based organizations, business processes are most likely found within legacy systems; most often written in COBOL utilizing IBM's CICS. Of major importance to organizations that depend on mission-critical legacy systems is the issue of how to effectively and efficiently Web-enable these systems. This study focused on two methodologies for Web-enabling such systems. The methodologies chosen were adapted from the Hurwitz taxonomy (Frye 2000) that presents a list of possible approaches to the Web-enablement and enterprise application integration problems.

Basis for Study

Determine Estimates of Cost and Time Associated with Two Differing Web Enablement Methodologies

Operational and information technology management must be able to answer three important questions regarding their choice of Web-enabling tool and approach:

- How much will they cost?
- How long will it take to start achieving benefits?
- How will it fit in with our organization's tactical and strategic goals?

To obtain quantitative estimates of these parameters, researchers Dr. Howard Kanter (The Institute for Software Metrics) and Dr. Thomas Muscarello (Graduate School of Business Administration) at DePaul University measured the time and costs associated with the tasks required to expose a legacy application to the Web. To facilitate this investigation Micro Focus provided appropriate software tools for legacy-to-web integration – namely, EnterpriseLink.

The Application

A Legacy Mainframe Payroll Application

The researchers believe that the use of a specially contrived and reasonably sized system is appropriate for this type of study. The original system was a payroll processing CICS COBOL system designed by DePaul for teaching purposes. This application was used as the basis for Web access and manipulation. The system incorporated many of the standard features and programming techniques including the ability to enter, alter, and query employee data, employee benefits, departmental data, salary data, timesheets, paycheck generation, pay-stub generation, withholding tax and FICA payments.

The Results

When the Study Was Completed, Developers Using EnterpriseLink Significantly Outperformed Those Using the Java Programming Approach

Empirical data was collected from study participants (all seasoned programming professionals) about the methods they used, including both qualitative and quantitative data. The EnterpriseLink method involved two steps. The first step was to study the problem and use the tool's automated feature to generate a first pass, Web-enabled system. Next, the programmers spent time to use the features of the tool to create an optimized and customized (specially designed screen layouts) version of the Web-enabled system. An analysis of the collected data produced the following results:

- Time required to implement an automated Web interface to an original interactive legacy COBOL system using an application processing integrator (EnterpriseLink) was significantly shorter than time required to implement using Java hand coding – twenty-five times faster (45 minutes using EnterpriseLink vs. 1275 minutes hand coding Java)
- Time to implement optimized/customized Web interface to an original interactive legacy COBOL system using an application processing integrator (EnterpriseLink) was significantly shorter than time required to implement using Java hand coding – more than five times faster (221 minutes using EnterpriseLink vs. 1275 minutes hand coding Java)

The costs for each of the methods were evaluated using the average of the high median salary for each geographical region of the United States given in a 2001 salary survey (Data Masters 2001).

Tools such as EnterpriseLink are not silver bullets, but they clearly provide a substantial reduction of the cost associated with extending legacy applications to the Web.

Benchmark Statistics

Task Details and Associated Times for Each Method

All Methods - Timings in Minutes	API (EL)	API (EL)	Java
TASK	INITIAL	OPTIMIZED*	REWRITE
Understanding the enterprise and problem domain	3.5	3.5	240
Understanding the data	7.5	7.5	90
Data design	3.5	3.5	60
Modeling	15	15	45
Understanding the process	4.5	4.5	60
Applying technology	6.5	179	600
Testing	3.5	7	90
Assessing performance/value issues	1	1	30
Conversion of packed decimal data for Java			60
Total Time To Convert	45	221	1275
Total Time to Convert as Percent of Java Rewrite	**3.53%**	**17.33%**	**100%**

The Java rewrite took the longest time to accomplish for a variety of reasons. The Java programmer, although very competent in Java, was unfamiliar with COBOL making the communication of requirements more difficult. As would be true with writing new code in any programming language, the time required for coding, testing and debugging was significant. It is true that a mainframe COBOL system rewritten in Java may provide ancillary benefits, such as ease of future modification and reuse. However, the rewrite in hand-coded Java requires significant incremental cost, as illustrated below.

In addition, the Java rewrite approach requires the conversion of data files that have been maintained by COBOL programs. These files may use data-types that have no equivalent in Java – such as packed decimal. This file conversion requires a substantial amount of time to complete and also increases the overall risk of the project.

Estimated salary expenses are as follows:

Average Salary Expense	API (EL)	API (EL)	Java
TASK	INITIAL	OPTIMIZED*	REWRITE
Understanding the enterprise and problem domain	$2.59	$2.59	$195.76
Understanding the data	$5.56	$5.56	$73.41
Data design	$2.59	$2.59	$48.94
Modeling	$11.12	$11.12	$36.71
Understanding the processes	$3.33	$3.33	$48.94
Applying technology	$4.82	$132.64	$489.40
Testing	$2.59	$5.19	$73.41
Assessing performance/value issues	$0.74	$0.74	$24.47
Conversion of packed decimal data for Java	$0.00	$0.00	$48.94
Total $$$ to convert	$33.35	$163.76	$1,039.98
% of $ to convert using Java rewrite method	**3.21%**	**15.75%**	**100%**

Conclusions

Based on an analysis of the collected data, results indicate that it is between five and twenty-five times faster to Web-enable a legacy system using an application processing integrator (in this case, EnterpriseLink) when compared to language migration and hand coding. Tools such as EnterpriseLink are not silver bullets, but they clearly provide a substantial reduction of the cost associated with extending legacy applications to the Web. This study indicates that application process integration technology is appropriate for organizations when:

- An organization needs to get to the Web in a short period of time
- It is not possible to modify an IBM mainframe production system, perhaps because it is proprietary software
- The skill set necessary to get to the Web is not available
- The production system source code for making necessary changes is not available
- The enterprise has AS/400 applications that need to be Web-enabled
- It is desirable to install Web-based systems in phases, perhaps keeping the same screens, which reduces retraining and minimizes trauma of change; this is especially true when dealing with an intranet-based system whose users would be comfortable with the older, non-GUI screens

History has shown that the cost of language migration from one language to another is not quick, painless or inexpensive. Additionally, the rewards rarely include increased flexibility, reduced long-term cost of ownership or improved performance. Therefore, extreme caution is required when undertaking a language re-write approach for transforming and extending legacy enterprise applications. Seldom does a business issue warrant a programming language change. This is especially true when existing technologies, such as EnterpriseLink, can be implemented to solve these legacy issues and eliminate the risk and cost associated with language change.

About Micro Focus

Micro Focus is the industry leader in COBOL development solutions ranging from traditional maintenance and program understanding to business rule mining, Web-enablement and user-interface transformation. Over 70,000 licensed users at more than 7,000 sites around the world use Micro Focus' unsurpassed breadth of platform support, performance and scalability. Micro Focus offers the most comprehensive suite of development and integration environments to help customers succeed in taking full advantage of the power of their legacy systems. Founded in 1976, Micro Focus is a global company that employs more than 450 people worldwide with principal offices in the United Kingdom and North America. For more information, visit **www.microfocus.com**.

877 772 4450
microfocus.com

Micro Focus Worldwide

Australia . . . . . +61 2 9904 6111
Belgium . . . . . +32 15 30 81 11
Canada
Ontario 905 824 7397
Quebec 800 280 9487
France 0800 835 135
Germany 0800 1825 443
Dortmund . . . . . +49 23 17 58 50
Munich . . . . . +49 89 42 09 40
Ireland . . . . . +353 1 469 3121
Italy
Milan . . . . . +39 02 6943401
Rome . . . . . +39 06 5153931
Japan . . . . . +81 3 5793 8550
Netherlands . . . . . +31 23 5689 138
Norway . . . . . +47 22 91 07 20
Sweden . . . . . +46 8 545 13 390
United Kingdom 0800 328 4967
United States . . . . . (+1) 877 772 4450
Other Countries . . . . . +44 1635 32646

DEPTLS0102-01