

# Case Study: Consolidating on Linux, zSeries and COBOL Beat 'Integration' for Banking Services Provider

*Phil Murphy and Jost Hoppermann*

## Giga Position

Organizations today are struggling with how to connect applications that have evolved over time across disparate platforms and represent more organized chaos, than purposeful design. Integration *can be* an effective approach to dealing with disparate technology, but in complex heterogeneous environments, such as synchronizing COBOL code and data across multiple hardware platforms, it can also be decidedly wrong and more costly than other options. Stepping back to consider more fundamental changes to the IT environment will cause other alternatives to surface that ensure greater reliability, flexibility, scalability and enable the organization to take advantage of new business opportunities. This case study highlights one instance where a more fundamental change to the IT environment exposed a much more flexible and scalable business solution, even as it foreshadows the early acceptance of Linux and COBOL for production-level processing by the financial services industry.

## Recommendations

Don't layer good integration tooling on top of an unworkable IT environment. Integration as a means to connect application and data silos is *one* solution, but not the *only* solution. Before rushing to apply message-oriented-middleware (MOM), enterprise application integration (EAI), Web Services or other integration technology as a means to connect disparate platforms, step back and look at the underlying platform, language, database and other architecture pieces. Don't assume that integration is the right answer.

Don't give up on COBOL — like any language, it has its strengths and weaknesses, and it clearly has some growing to do to meet and integrate with newer technology, but it is doing so (see References). COBOL is reasonably portable across PCs, Unix, mainframes and now Linux; it is in widespread use; it is far more similar to the English language, therefore more readily understandable than other languages; and it will remain a popular language for the coming decade and beyond.

Use Linux on zSeries to rehost and consolidate COBOL applications from multiple NT and Unix boxes as processing capacity requirements exceed their capacity, and also to reduce the complexity and administration costs of interacting with larger host machines. Clearly, the combination has matured beyond experimentation but not production in an industry that demands security, reliability and scalability. Production processing of tens of millions of financial transactions per day makes it clear that the combination of Linux, zSeries and COBOL represent a viable recentralization option.

## Proof/Notes

Technological disparity within IT is one of the most costly and troublesome issues facing organizations today. The cost and effort to manage increasingly more complex and diverse IT inventories, myriad vendors, product interoperability and competing/conflicting standards are pushing demand for resources well beyond their breaking points. In the rush to deliver value to the business at an acceptable cost level, "integration" has become a catch-phrase answer — sort of a knee-jerk reaction to technology disparity.

Integration of existing systems is a necessary improvement to applications built as individual silos of business functionality. However, custom-coded point-to-point integration is costly and rigid. EAI tooling requires significant financial investment, and many of the vendors are consolidating with other integration vendors across the broad category of tooling (see Planning Assumption, [The Changing Integration Landscape](#), Mike Gilpin, Randy Heffner, Uttam Narsu and Ken Vollmer).

The latest and perhaps most promising integration solution to address rapidly changing business requirements is the move toward building loosely-coupled services on top of a service-oriented architecture (SOA). During the next decade, SOA is the ultimate strategic goal for organizations with heterogeneous technology bases. Even homogeneous organizations benefit from services orientation by building a single service that accomplishes a business function from any user interface or device. But all these approaches assume that integration is the de facto answer, never pausing to consider the question of *whether* integration is the better approach. Organizations that fail to consider whether to integrate run the risk of layering yet more software on top of an inappropriate infrastructure to the point of widespread system inadequacy/failure. The subject of this case study faced a number of internal integration issues, fueled by significant changes in the industry, driving its decision to change.

### Case Study Subject

The banking software and services provider (BSSP) chronicled within this case study operates under the names of **ABK Systeme GmbH** and **EFiS EDI Finance Service GmbH**. ABK develops and distributes payment processing software to approximately 70 banking customers, and also to EfiS, which was created approximately 10 years ago and uses the software to provide payment processing services to its client companies. Furthermore, EFiS is a SWIFT certified service bureau that handles payments processing for about 40 smaller and midsize banks.

ABK and EFiS currently offer payment processing software and services for virtually all kinds of payments, in several formats, supporting all major Automated Clearing Houses (ACH) and a variety of message formats, such as SWIFT, EDIFACT, as well as some domestic and Extensible Markup Language (XML)-based formats. Available communications interfaces include MQ series, Merva and SWIFTNet. Total revenues for 2001 exceeded 10 million Euros (approximately \$11 million). As a software provider of COBOL applications to many other banks and financial institutions, the portability of the products from ABK/EFiS (hereafter referred to as “the BSSP”) is of primary concern.

### Industry Drivers

In today’s tough economic environment, financial services organizations face overwhelming pressure to contain costs. To survive, many banks are experimenting with specialized services, differentiating themselves as dedicated sales, product and service banks, and evaluating whether outsourcing some financial services will lower operational costs. For example, international banks are looking for cost-effective alternatives to **IBM**’s payment software **MERVA**, which has been an established part of the financial services industry for nearly 20 years. These drivers manifest themselves in a variety of ways:

- Bank mergers, acquisitions and (in rare instances) failures will reduce the number of existing and potential customers for banking and payment services.
- Customers will settle into one of two categories, small to midsize banks looking to reduce the overall cost of payment processing and the subsidiaries of larger banks looking to lower development costs for payment processing software.
- The software market will become even more competitive than it is today, since the number of potential customers will become smaller.
- The same is true for the services market, where the number of potential vendors is increasing. Vendors competing with the BSSP for services business may themselves be running the BSSP’s software to service customer transactions.

- The software and services companies must focus on cost-competitive offerings to meet the needs of these two customer categories — cost-competitive services for small and midsize banks and attractive software licensing and support for larger banks.
- Cost reduction continues to apply pressure as a general concern and as driven by regulatory mandates. For example, organizations conducting business in the European Union (EU) must address an initiative aimed at equalizing costs for national (intra-country) payment orders and inter-country payment orders.
- Transaction volume will increase significantly, driven by factors such as extension of the EU to Central and Eastern Europe, market consolidation of service banks, the consolidation of domestic and international payment systems to reduce costs. Indicative of the trend, the BSSP had recently received requests for proposals (RFPs) for software that would handle maximum daily transactions in the 100 million range, with hourly burst rates approaching 20 million.

### Issues Faced by the BSSP

The BSSP faced a number of technology limitations with its existing environment. With COBOL running on Windows and Unix platforms internally, and with many larger customers being mainframe-based, the integration and maintenance of data and applications across the Wintel/Unix/mainframe platforms was a continuous problem. Internally, the mix of Wintel and Unix technology and data originating from mainframe-based applications created integration and ASCII/EBCDIC data conversion issues. While these issues are eminently solvable, the larger the data volume, the more likely that the source was EBCDIC, and the more cycles wasted on character set conversion between the incoming EBCDIC and their processing character set (ASCII). Additionally, the BSSP was encountering other issues with its current technology configuration — serial file processing was creating batch window issues, the Unix hardware/operating system was incompatible with many of its customers' architectures, and the BSSP's customers had exceeded the processing volume capacity of the software on the largest available Unix machines.

The industry drivers exposed the need for this BSSP to develop more efficient and reliable software development, deployment, and release management capabilities that would enable new software releases as cost efficiently as possible. In such a competitive market, software must easily scale to meet the performance, load and availability requirements of larger banks (product offering) and to meet a significant number of distinct load profiles of smaller banks (services offerings). In the face of anticipated increases in transaction volume, and in order to take advantage of new business opportunities to service the needs of larger banks, the BSSP decided to effect a more fundamental change within IT that allowed it to detangle an environment that had evolved during the two previous decades.

### The BSSP's Available Solution Choices

The choices the BSSP faced were as follows:

- Continue use of Unix systems, but with large multi-processor **Sun** E/10000 with several Unix partitions. This presents no significant skills issues and development could be done on smaller Sun machines, although the cost to implement and administer such a configuration may not represent any savings.
- Use multiple distributed Unix systems in a "building block" approach. This would not present a skills issue, and the investment is scalable but operational costs can be quite high, and availability issues would likely surface. These first two options also ignored the trend of some of the BSSP's customers toward consolidating multiple Unix instances onto Linux.
- Migrate to a small mainframe with OS/zOS operating system — This would require considerable financial investment. Are mainframe skills available?
- Do nothing and forgo new business opportunities, and economize as much as possible; however, this is not a viable option.

## The BSSP's Choice

In considering a new environment that would satisfy the existing needs and leverage new business opportunities, the BSSP decided that its central design goals must include vendor neutrality, platform independence, high performance and tight security — a cost-effective solution that is scalable and provides independence from the various message formatting requirements. These technology-oriented goals clearly defined a challenge, considering that the BSSP's supports software running on AIX, HP-UX, Solaris, Linux, OSF/1, Sinix and other operating systems. Furthermore, the BSSP's software includes Wintel-based client systems in some parts of the solution. This challenge was reduced somewhat by the fact that most of the applications were written in COBOL and could depend on the reliability and the processing speed for which COBOL was originally designed. Nevertheless, it was a daunting task to become even more competitive, more cost-effective, improve availability and scalability, as well as providing strong security mechanisms.

As a first step, the BSSP tried tuning the COBOL applications to handle I/O more efficiently and evaluated migrating to a faster Unix server, to a server farm configuration and to IBM's zSeries. Through the evaluation process, the BSSP realized several deterministic truths:

### *About its customers:*

- The BSSP's largest existing software customers had hit a performance plateau on Sun, HP/UX and IBM/AIX platforms of 8 million payment transactions per day, with transaction bursts of up to 400,000 per hour. While it could improve performance somewhat by tuning the software, tuning alone would not bridge the gap between current capabilities and emerging customer requirements. Some of the BSSP's customers had recently issued RFPs for software that would approach 100 million transactions per day, with transaction bursts approaching 20 million per hour.
- One customer reported that according to its calculations, even the largest available Unix server would fail to handle its anticipated volume.
- The number of banks and other potential customers using mainframes was significantly larger than the number of banks using high-end, multi-processor Unix solutions.

### *About the solution:*

- The target architecture must leverage existing software investments (COBOL).
- The development environment must be as cost-effective as possible.
- The solution must leverage the skills of existing staff and those of potential local candidates — experienced and entry level (e.g., university graduates) — without any large additional investment.
- The new system must be able to run on low-cost platforms (such as PCs) as well as high-end solutions.

As the evaluation progressed, some of the BSSP's largest customers decided to migrate their Unix platform processing to Linux on the mainframe, lending more credence to, and setting the stage for, a proof of concept (POC) trial of Linux on the zSeries mainframe with the support of IBM and **Micro Focus**. After a few weeks of effort to port the applications to Linux, the BSSP ran benchmarks of its existing configuration against the newly configured Linux mainframe.

The table below shows the increase in transaction processing volume achievable by porting the BSSP's COBOL application to Linux. Some facts to consider about the test are as follows:

- The rows within the column show total daily volume and a burst speed number, which reflects peak-processing transaction rates per hour. Burst speed rates are needed (and sustainable) for only short periods of time and vary based on what else is running on the processor concurrently. Therefore, the total daily volume is not a direct multiple of hourly volume multiplied by 24 hours.

- Linux tests were conducted in laboratory conditions, which do not reflect the reality of production environments. The volumes within the table are restated using a highly conservative approach — actual test results were significantly higher, and it is reasonable to assume that actual production volume may scale as high as 25 percent to 50 percent higher than the restated numbers within the table.
- Overall disparity between internal systems volume and external (customer) systems volume reflects the much smaller Wintel and Unix systems operated by the BSSP, as compared to the BSSP’s largest bank customers.
- The improvement in internal processing volume appears at first glance to be astounding — on average, between 47 and 103 times the initial capacity. However, this improvement is testament in large part to the inadequacy of the BSSP’s original Wintel/Unix configuration. The external customer numbers are more reflective of true improvements.
- Based on test results, external customers could expect throughput increases ranging from 11 to 20 times the current volume, fully enabling the BSSP to approach the 20 million per hour rates noted within the RFPs.

### Transaction Performance — Before and After Test Results

Transactions	Internal Systems (Unix and Wintel)	External, Customer Systems (Sun, HP/UX, AIX)	IBM Linux on zSeries
Maximum hourly volume (burst speed)	0.4 million	1.6 million	19++ million
Total daily volume	1.5 million	8.0 million	155++ million
Burst speed to total daily volume ratio	1 to 3.75	1 to 5.0	1 to 8.1++

Source: Giga Research, a wholly owned subsidiary of Forrester Research, Inc.

Note: The test payments represent real, depersonalized orders from an existing bank, run on an IBM zSeries 900, three CPUs, three parallel I/O channels, SuSE Linux Enterprise Server

In addition to the significant performance improvements and closer alignment to the platforms of its larger customers, the BSSP gained an important efficiency edge with this configuration. The same set of source code can be used for existing applications on smaller Linux platforms as well as for the mainframe. This is an important consideration for the BSSP’s software distribution line of business (LOB), since compatibility with other organizations within financial services remains a key requirement. Furthermore, the source code portability enables the BSSP to use cost-effective PC Linux platforms and transfer the resulting software packages to more powerful servers or mainframe systems.

The results of this case study demonstrate that COBOL applications running on Linux and zSeries are ready for production workloads, that the BSSP’s expectations for volume processing were, in fact, exceeded and that it is now safe for moderately aggressive organizations to consider it as a viable option to reconsolidate from Unix and NT and perhaps larger workloads.

### Alternative View

The next most likely choice for the BSSP was to evolve toward a cleaner, less complex configuration that would take better advantage of some of the integration tools available. That choice may have used a series of Unix servers configured together to handle the anticipated volume increases, then focused efforts on some of the other problems symptomatic of a highly heterogeneous environment in the hopes of mitigating some of the negative effects. It is important to note that such fundamental infrastructure change was possible in part because of the BSSP’s relatively small size. Larger organizations will be forced to evolve more slowly due to the size, expense, complexity and other circumstances that accompany big-bang types of change.

## References

### Related Giga Research

#### Planning Assumptions

[Maximizing IT Value: Portfolio Management Options for 2003](#), Phil Murphy

[The Changing Integration Landscape](#), Mike Gilpin, et al

[Application Strategy: Planning for Flexibility](#), Jost Hoppermann

[Multi-Channel Can Be a Success With Only a Mainframe](#), Jost Hoppermann

#### IdeaBytes

[Evaluating the Fate of Applications? Think Business, Technology, Core Competency, Opportunity Cost](#), Phil Murphy

[Reducing the Cost of Legacy Application Systems](#), Phil Murphy

[No Monster Shortage of 'Legacy Skills' in 2003](#), Phil Murphy

[Don't Change Operating Platform to Replace Obsolete User Interface Technology](#), Phil Murphy

[Directly Accessing Legacy Data Provides Limited Value](#), Phil Murphy

[Don't Tackle Legacy DBMS Conversion Programming Alone](#), Phil Murphy

[Migrating Legacy Applications to Unix? Not Likely](#), Phil Murphy

[IT Issues and Opportunities for Banks: New Financial Services Scenarios Update](#), Jost Hoppermann

[IT Issues and Opportunities for Banks: Linux Migration](#), Stacey Quandt

### Relevant Links and Other Sources

Micro Focus, [www.microfocus.com/products/netexpress/index.asp?bhcp=1](http://www.microfocus.com/products/netexpress/index.asp?bhcp=1)

IBM, [www-1.ibm.com/servers/eserver/zseries/solutions/s390da/linuxproduct.html](http://www-1.ibm.com/servers/eserver/zseries/solutions/s390da/linuxproduct.html)

ABK/EfS, [www.abk.de](http://www.abk.de)