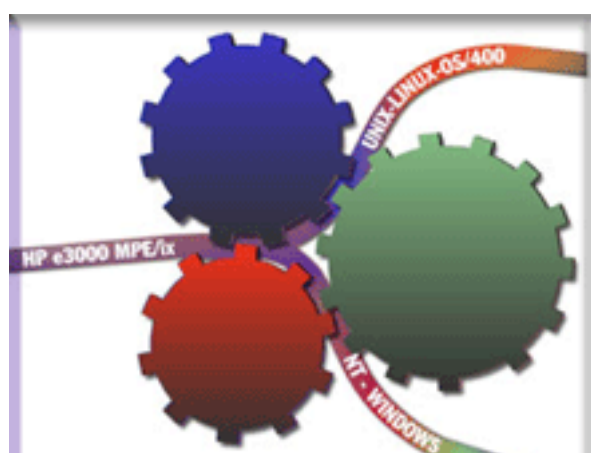




# LINUX MIGRATIONS: ADVICE TO THE FEAR-LORN



*Homework needed in corporate's application migration path to Linux? Absolutely. But your biggest task will be calming business nerves.*



by Jon Power, CEO, Sector7

*Somewhere between heaven and hell is Linux migration. Looking upwards, one can anticipate a Linux price/performance value proposition that cannot be ignored, as corporations increasingly adopt Linux for infrastructure solutions, edge-of-network applications, development platforms, and technical computing. Looking down, however, they face the abyss of fears. Jon Power, Sector7's CEO, has the questions that need to be asked and resolved. First established in the UK in 1985 and now Austin-based, [Sector7](#) has migrated applications from OpenVMS to Windows and UNIX, UNIX-to-UNIX migrations, and now Linux. Sector7's tools and services enable application developers to migrate their applications from proprietary systems to operating systems like Linux and UNIX. In this overview, Power lets you know what issues and variables you need to assess in constructing a migration path toward Linux and the fundamental hurdle ahead.*

The very idea of being able to reduce computing costs by implementing a Linux-based infrastructure—the very lure of a low cost hardware platform and a free operating system—seems to be just what the doctor ordered in these days of corporate budget cuts and spending scrutiny. And the future looks bright for those who can dole out the new prescriptions. As more and more universities and colleges release IT graduates whose only exposure to a non-windows operating system has been Linux, we are reminded of the days when free VMS and SUN systems were installed in centers of learning solely to create future decision makers to be naturally drawn to what they know. It worked for DEC, it worked for SUN, and history indicates it will work for Linux.

But what about today? Where do we stand on moving mission-critical applications from the work engines that allow today's decision makers to feel comfortable and secure?

An analysis of the source operating system and its native capabilities must be mapped against the middleware that's available on Linux. The result of this mapping effort will determine if a straight port is feasible or if some application re-engineering will be required. Should a specific package that you are using on your current system not be available on Linux, then you must choose an alternative that fits your needs, the functionality mapped, and an accurate project plan created to reflect the true effort, both in terms of time per line item and number of items to be addressed.

Fundamentally, application migrations fall into two broad categories: non-UNIX-based operating systems and UNIX-based operating systems.

## Non Unix-Based

Non Unix-based systems such as OpenVMS, OS400, MVS (or zOS as it's known today) DOS mainframe VSE, MPE, Tandem, and Unisys, all fall into the 'difficult' category, at least at first glance. To take a typical application from the 'difficult' group will normally require some parts of the application to be reengineered into a form that supports the availability requirements of the original system

The key to a successful migration from the 'difficult' category is analyzing every aspect of the application; finding an analogue or replacement for the third-party products or the native intrinsic functionality of the operating system and then being able to accurately create a project plan with accurate and realistic time estimates for each task and mapping group. Generally, such a plan will be 3,000 to 5,000 line items with no one line item exceeding 40 hours. What's more, whatever the time line and resource requirements are for migrating and mapping the code, generally the test/ debug/ fix cycle will be 3 times that amount.

## UNIX-Based

What about the second, UNIX-based group? For this group of applications, i.e., UNIX derivatives (AIX, SOLARIS, Tru64, HPUX, NCR, SCO and Berkeley), the task is significantly simpler, but the same principles apply. Here, the languages are typically Java, C, C++ or COBOL. While these are mostly portable, C++, due to a lack of standards in its early years, can cause some late nights. The issues in this 'easier' group boil down to third-party package availability and 'version-for-version' compatibility.

Many core IT applications can be newly developed on Linux, using middleware messaging products such as Oracle, Tuxedo, and MQSeries to overcome perceived instabilities in either the operating system or hardware, by providing a level of failover and redundancy at the software level.

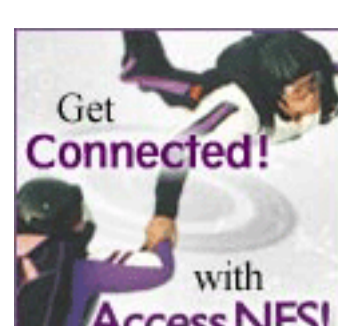
## Linux Migrations

Application migrations to Linux have been done, and there are real experiences to draw from with curious results. At Sector7, we have ported many applications to Linux; we have seen applications run for months on end without any unexpected errors. And yet, even with a proven reliability profile, the applications have not been deployed. Why? The answer is nervousness within the ranks of upper management.

The questions around migration applications to Linux today are similar to what we at Sector7 saw back in 1987. At that time our complex migration offerings were from OpenVMS to a newly available commercial operating system called UNIX. In those days, it came in a few flavors – System V, System III, Version 7 – all delightfully quirky and mostly free of any third-party products. Some versions did not support record locking; none had clustering. Universities were getting their first taste of Unix on ALTOS, PDP11, FORTUNE 3216, APOLLO, and SUN.

In contrast, however, Linux seems to be emerging as the 'unified' UNIX. Also, with major players such as IBM standing behind their Linux offerings and providing full support, it is just a matter of time before Linux becomes an accepted platform for mission-critical applications—not to mention the fact that Microsoft's recent licensing policy announcement is further making a move to Linux appear to be rather attractive.

Today's big question about porting applications to Linux is not "Can it be done?" The answer is "Yes." The key question is, "Can I sell the concept up the management chain to a CIO/CFO with millions of dollars in potential share options on the table that Linux is a failsafe solution?"



[Click](#) to learn about **Shaffer Solutions'** powerful set of integration tools (a finalist at the August 2001 LinuxWorld for best network server) that provide fast and secure connections between Windows and Linux platforms.

