MKS

MKS Toolkit Case Study

UNIX-to-Windows Porting Makes Development a Cinch

Businesses that are successful today base their IT investments on power and flexibility, and heterogeneous environments are the norm rather than the exception. Organizations can no longer deliver only UNIX-based systems to customers. The marketplace requires additional offerings. With an increasing demand for PC-based, multi-platform products, enterprises around the world are looking for solutions that can bridge UNIX and Windows, so they can offer more than a single-platform product and remain competitive. When faced with the decision to duplicate existing UNIX/Linux applications on Windows and take advantage of the new Web development model, the options are clear: either perform a complete rewrite, or find a tool that lets you port the application.

Rewriting old applications can take anywhere from months to years and requires ongoing maintenance of two separate source code bases, which is a major ongoing cost. The second option--porting--should be carefully considered because the price for selecting the wrong tool can be high.

The Situation

Recently, a large international consulting organization faced this UNIX/Linux-to-Windows challenge. The organization's product is a UNIX-based system with the front end running on Windows. Whenever the organization gave a demonstration to a prospect, it would end up either renting a UNIX system and transporting it, or installing their software on the client's machine. In extreme cases, they would have to dial up and connect. Additionally, the sales team had to have the technical skills to set up the environment, instead of concentrating on selling the product.

The company looked at both rewriting and porting their application so all modules could run on a standalone machine, which the sales team could then install only once on their laptops. This way, when they were on sales calls they could focus on their customers' problem and show them their solution.

Rewriting was ruled out because they would need to continue to update both their UNIX and Windows versions with every new release of their product. As a result, they chose to port their application to Windows.

The Solution

The company looked at three porting tools: Microsoft SFU, U/Win, and MKS Toolkit for Enterprise Developers. In the end, they chose MKS Toolkit for Enterprise Developers because it offered the flexibility of using both UNIX APIs and Windows APIs, an important consider-ation because the application uses Pro*C for its database calls. This option allowed this customer to have a single code base, which drastically reduces the maintenance overhead. Another consideration was the comfort of MKS' large user base and professional support.

Although the primary objective was to benefit the sales team, after completing the port with MKS Toolkit for Enterprise Developers, the company realized it had uncovered a host of new business opportunities; it could market its product to smaller customers who had only Windows environments.