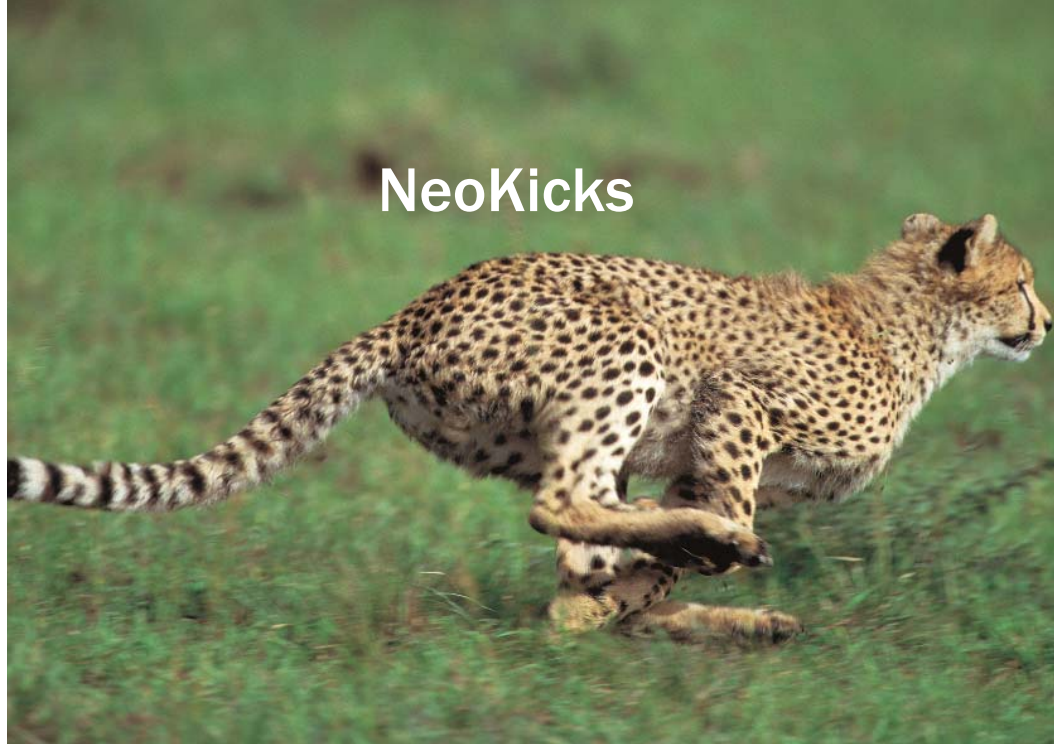


NeoKicks



Executive Summary

NeoKicks™ is designed for migrating CICS applications to the .NET environment, providing a similar system architecture while enabling the applications to take maximum advantage of the modern interfaces and technologies provided by the Microsoft® .NET Framework. You can convert BMS maps into ASP.NET Web page or WinForms interfaces, yet preserve mainframe style keyboard input. This often allows companies to transform one of the most inflexible parts of their application architecture to a modern, agile architecture.

NeoKicks Invigorates CICS

Microsoft's .NET Framework provides the opportunity to give new agility to CICS applications. Its scalable architecture, flexible communication models, and support of modern interface design and execution, make it an ideal platform for CICS applications – not only does it have the necessary power and compatible architectures to match mainframe CICS but it also offers technologies that enable your applications to move into new realms of functionality. NeoKicks™ makes the vital connections that map your CICS COBOL programs to .NET's riches.

Highlights

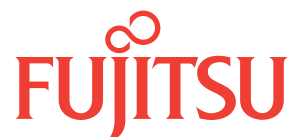
Here are some of the features that make NeoKicks the best way forward for CICS applications:

- Wizard imports your CICS COBOL code into Visual Studio .NET projects, preprocessing EXEC CICS statements into the NeoKicks equivalents. The preprocessed code and generated

projects work with Fujitsu Software's leading NetCOBOL® for .NET product.

- Wizard also converts your screens into ASP.NET Web Forms or Windows Forms, that you maintain using Visual Studio's designers
- Theme Manager helps you establish styles for your converted BMS maps
- DPL (Distributed Program Link) transactions convert to Web services.
- Integration with Visual Studio .NET gives productive and easy-to-use development environment
- Transaction support for any ODBC data provider, Btrieve or any other external transaction monitor
- User interface and application tiers are separable allowing flexible deployment options
- Mainframe UI behaviors emulated such as:
 - Overtyping behavior
 - Tabbing behavior
 - PF keys, PA keys & Break

Fujitsu Software Corporation
1250 East Arques Avenue
Sunnyvale, CA 94085
Phone: (800) 545-6774 or (408) 428-0300
Fax: (408) 746-6369
Email: cobol@netcobol.com
Web: www.netcobol.com



Benefits

Migrating to NeoKicks provides the following compelling benefits:

- Cost effective, scalable, hardware platform – Windows servers have very attractive price:power ratios
- Escape from a proprietary hardware and software environment in which enhancements come from a single vendor.
- Enter an open environment where enhancements and new technologies are being provided by multiple vendors
- Modernize application interfaces (in a controlled manner – opt for mainframe look and feel until you are ready to move to GUI)

- Expose existing business logic as Web services
- Provide your programmers with the world's best development environment (Visual Studio .NET)

Comparing CICS and NeoKicks

A big factor in the viability of moving to NeoKicks is that NeoKicks provides a very similar architecture to mainframe CICS. This means that applications can be brought into the .NET Framework with no architectural changes by just running them through the NeoKicks wizards. NeoKicks enhances the environment for CICS programs by:

- Providing tools for configuring the styles and environment

- Using a pluggable architecture for mapping mainframe behaviors to Windows concepts (so each installation can fine-tune to their requirements)
- Replacing dated interfaces such as 3270 screens and DPL with modern technologies such as WebForms, WinForms and Web Services.
- Placing applications in Visual Studio .NET projects

Platforms Supported

- Microsoft® Windows® 2000, XP
- Microsoft® Windows® Server™ 2003

NeoKicks is a trademark and NetCOBOL is a registered trademark of Fujitsu Limited. All other trademarks are the property of their respective owners.

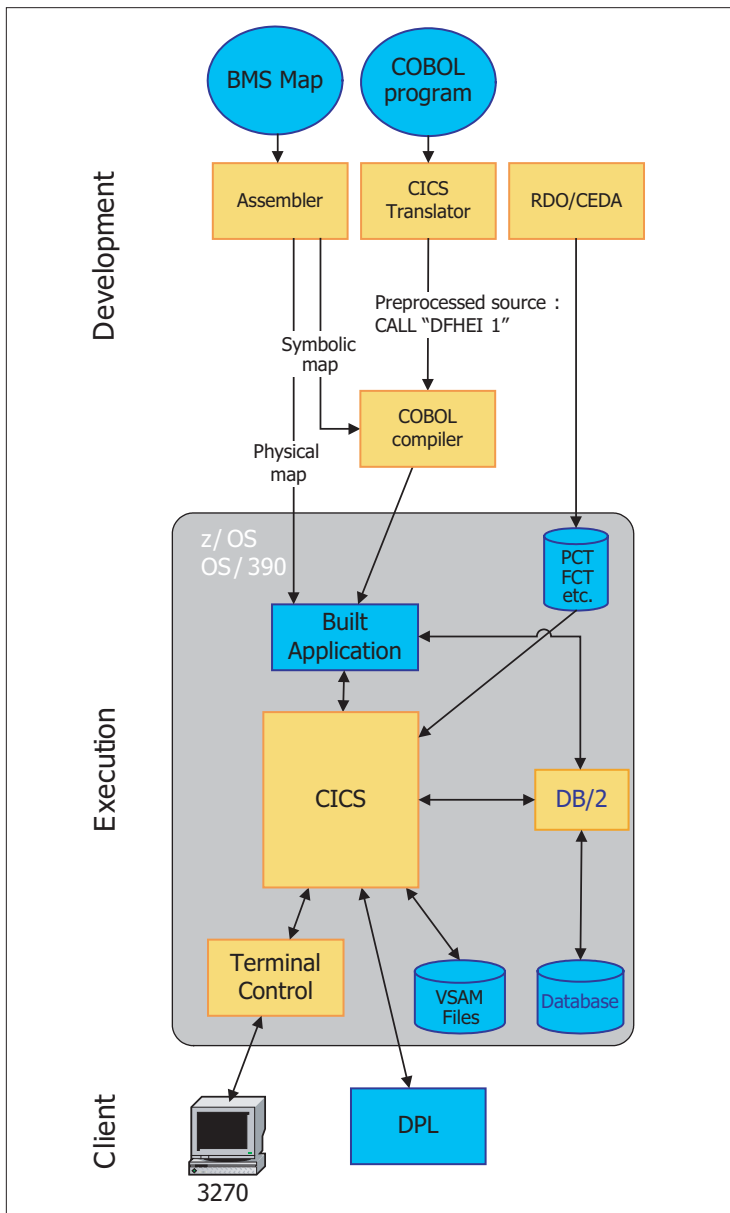


Figure 1. Mainframe CICS Architecture.

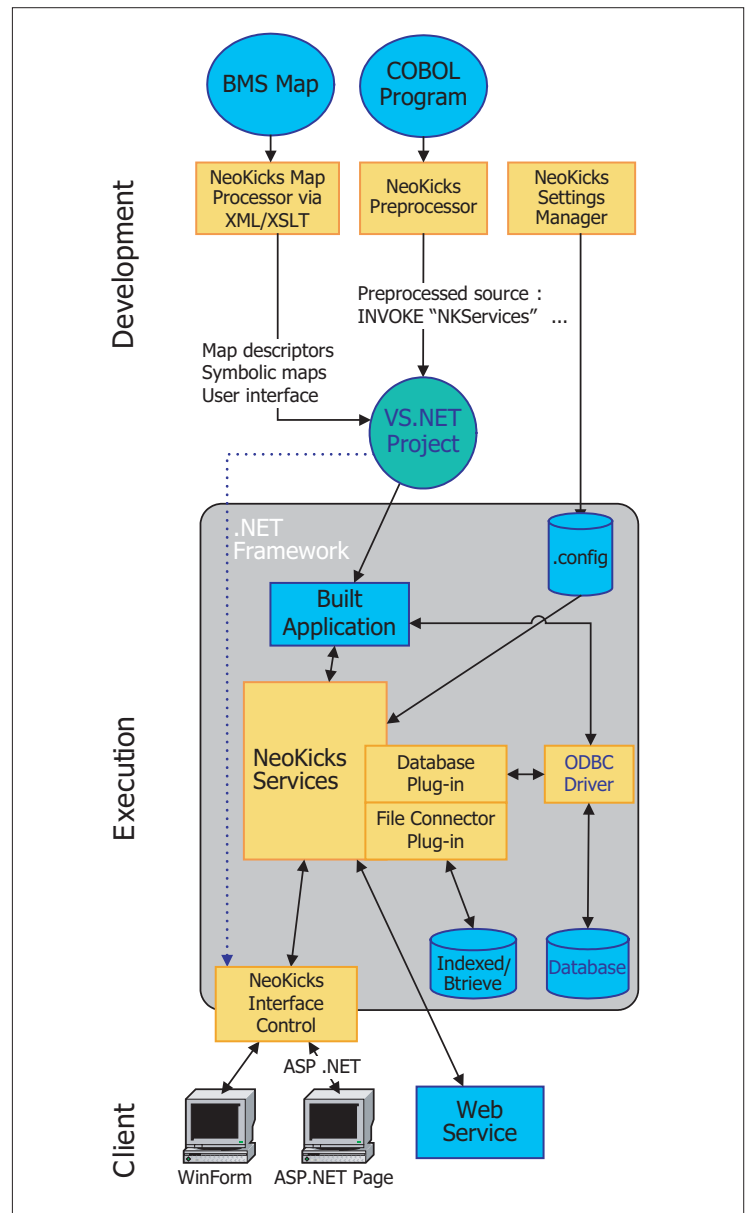


Figure 2. NeoKicks Architecture.