



Integration Expert (IX) v1.0 for Windows CE

The Essential Companion to the Microsoft Windows CE ETK

Product Overview

Windows® CE is an essential tool that works in conjunction with Microsoft's Windows CE Embedded Toolkit (ETK) to simplify and accelerate many critical aspects of Windows CE-based embedded system development.

The ETK is the tool used to customize and build the operating system, download to the target, and debug on the target. Where the ETK is command-line driven, IX provides a graphical, intuitive interface to the ETK as well as a variety of analysis, optimization and integration tools not available in the ETK. IX also facilitates the development of application software in parallel with the development of the hardware platform.

The result of using IX is a dramatic improvement in developer productivity and reduction in time to market. The average programmer will spend up to one month or even more learning the ETK to the point where they are proficient enough to do productive work. Improved developer productivity with IX can be translated into tens or even hundreds of thousands of dollars in time-to-market opportunities.

Features and Benefits

First and only GUI for the ETK - Modeled after the industry standard Microsoft® Developer Studio, IX provides a familiar and comfortable graphical environment. It does for the ETK what Windows did for DOS; it makes the ETK easy to use.

Fully compatible with the ETK - IX for CE loads component and module information directly from the ETK. This means that developers can use IX for CE to modify projects they created previously with the ETK or create new projects using new releases of the ETK.

Visual decomposition and modeling - IX provides a visual decomposition of the target, including applications and operating system software. The visual display shows the relationships and dependencies between

various system objects, providing developers with a comprehensive view of the software to be included on their embedded system.

Configuration and optimization - IX provides visual tools to set-up ETK projects, configure Windows CE, manage dependencies between objects, configure the contents of ROM, and configure the registry, database and file system initialization files. It facilitates the minimization of ROM and RAM usage through elimination of duplicate and unnecessary components using a suite of analysis tools.

Parallel platform and application development - The ETK assumes a linear model where a custom OS needs to be defined before application development can begin. IX allows development of the application immediately by delaying the definition of the custom OS until later in the development cycle when the precise requirements of the OS are better understood.

Component integration - The IX Component Gallery allows users to visually select and integrate Microsoft and third-party operating system and application components (such as the Rainbow web server or DeviceCOM) from a local and network-based gallery of components. The IX Gallery also provides a uniform mechanism for OEMs and developers to showcase and to facilitate the distribution of their unique components.

Input, analysis and manipulation tools - IX provides tools to import several data types, and to set up and edit data. These tools simplify the process of configuring the contents of the ROM, registry, database and file system initialization files. IX automatically analyzes and identifies inter-module, component or application dependencies.

Integrated build environment - Based on the configuration selected by the developer, IX for CE generates the files required by the ETK's build tool and then runs the build tool, generating the target system in a visually-integrated fashion.

Development Process Overview

The IX development path divides the task between the application and systems engineers.

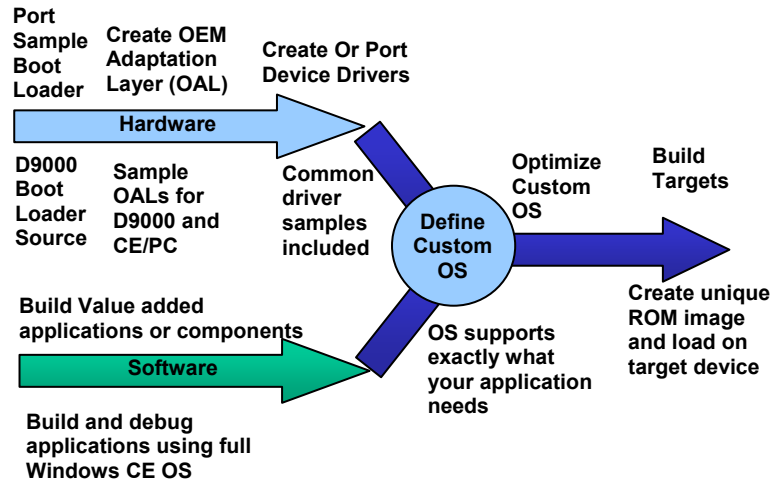
Applications can be developed using the full OS while the platform and drivers are being developed. When it comes time to build the whole system, IX can optimize the whole project so that only the needed components and modules will be added to the final build.

When IX first starts up, the ETK is imported into the Component Gallery by interrogating component and module definition files, binary image builder (BIB) files, registry files, projects, and platforms. After the import, OEMs can traverse the Component Gallery and view the functions and dependencies that make up individual components and/or modules.

The next step is to create a Workspace that will contain the Project (applications, components and modules) and the Platform (OAL and drivers). The developer will have the choice of either creating a new project or selecting a project that already exists. The best feature of IX is the ability to create a new project in the workspace based on the analysis of an application. Using this feature to create a new project, the developer drops their application into the workspace and runs a static analysis to pull in all of the OS components needed to run the application. The static analysis takes the guesswork out of optimizing the OS (vs. many trial and error iterations). Once the project has been created, IX asks for the platform and CPU type. IX will then pull in all of the components that make up the project.

The OEM can edit the project workspace, using familiar editing concepts such as drag n' drop, cut and paste. Edit boxes are used to edit BIB and registry entries. A key feature of IX is to take

components (such as OS modules/components, applications, and drivers) and drag and drop them from the Component Gallery to the project workspace. For example, if you wanted a web server (e.g. Rainbow) or DCOM support (e.g. DeviceCOM), you can grab either component from



the Component Gallery and drag it into the project workspace with only minor or no edits. After an OEM has completed the application and platform development, they can re-run the static analysis on the whole workspace to check and verify that all of the OS components are in place. The static analysis only provides a check for dependencies. Compilation errors on source code will require correcting. Finally, the OEM can build, test, and deploy the OS.

Contact Information

Intrinsyc Software, Inc.
 Suite 1050, 1075 W. Georgia St.
 Vancouver, BC, V6E 3C9
 Tel: (604) 801-6461
 Fax: (619) 673-1432
 E-mail: sales@intrinsyc.com
 Web: www.intrinsyc.com

Specifications

Development Environment	<p>System Requirements:</p> <ul style="list-style-type: none"> • Microsoft Windows NT 4.0 , SP3 • Microsoft Windows CE Embedded Toolkit (ETK) v2.0 or v2.1 with Enhancement Pack • 32Mb RAM (128Mb recommended) • 50Mb free disk space <p>Views and Tools included in IX:</p> <ul style="list-style-type: none"> • Standard Views: Workspace, Registry, Database, File System, Output, and Local and Remote Gallery 	<ul style="list-style-type: none"> • Import Tools: File, Directory, Registry Key/Registry Value, .db File, .dat File, .str File, .bib File, Build Tree Import (dirs and source files) • Analysis tools: Project Dependency Checking, DLL Import • Build Tools: Project Cleaning, Project Building, Platform Cleaning, Platform Building
--------------------------------	--	--