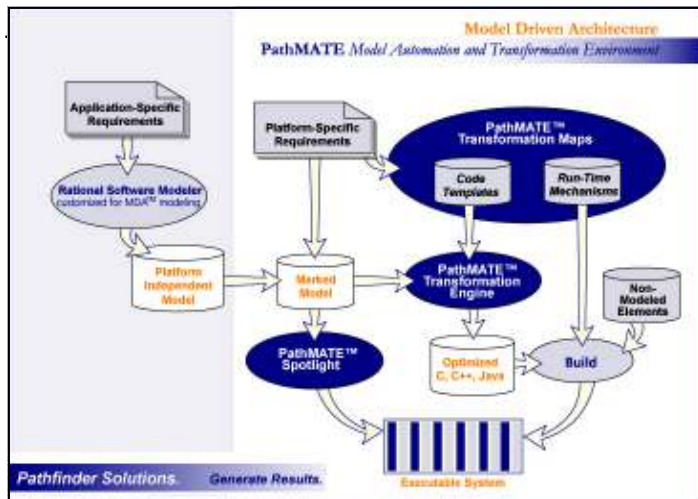


PathMATE™ Transformation Map for C

Generate a complete, efficient implementation for your complex, embedded, real-time distributed systems from platform independent models to a range of embedded and workstation target platforms.

The **C Map** delivers a complete set of transformation rules and templates to generate fully functional, deployable code from platform independent models. High speed automated code generation produces complete, executable systems directly from platform independent models. Deploying to a range of target environments with a host of features from multi-task and multi-processor topologies to model level debugging, the **C Map** provides everything needed to quickly build, test and integrate embedded, real-time, high performance components and systems.



Apply to the **C Map** when you need to:

- Get your complex embedded system running quickly
- Generate efficient and fast C implementation from your platform independent models
- Test your system in the development environment before deploying to your embedded target
- Quickly and fluidly adjust the deployment task and processor topology
- Seamlessly mix and match C++ and/or Java with C.

“The PathMATE C Map generated code that was 3 to 7 times smaller and faster than code from a leading code-in-the-model tool. The self-optimizing capabilities make a remarkable difference.”

*Architect – Software Performance Team
DoD Surveillance System*

Deploy Efficient Code to Your Choice of Target

Designed from the outset to produce highly reliable, efficient, small footprint and fast systems, the **C Map** is your quickest path to building and deploying your complex, embedded, real-time system. With direct support for all Platform Independent Modeling elements, C Map generates ANSI C that runs on the C Map Software Mechanisms run-time execution layer. Designed specifically to run efficiently with as little consumption of RTOS resources as possible, Software Mechanisms run on most common embedded and workstation target environments:

- AIX
- HP-UX
- MercuryOS
- Nucleus Plus
- pSOS
- TI DSP BIOS
- Win32
- Cygwin
- Linux
- MQX
- OSE
- Solaris
- VxWorks
- Win64

The simplicity of the C Map’s portability layer facilitates rapid adaptation to many other environments.



Self Optimization - Only available with PathMATE

The **C Map** provides the unique capability to automatically self-optimize the implementation code as it is being generated. Aggressively leveraging Platform-independent Action Language (PAL), the C Map has complete and easy access to all action details - allowing it to provide a range of **Self Optimizing** capabilities:

- Self trimming of unused model elements
- Only build in run-time layer elements actually used
- Self-selection of optimal memory management, instance data storage and access mechanisms
- Generate tailored infrastructure code only when needed, avoiding unused or inefficient constructs
- Efficient topology resolution, automatically using local accesses where intertask and interprocessor mechanisms are not necessary.
- Transformation-time selection of optimal infrastructure, generating compile time resolution where possible



PathMATE™ Transformation Map for C Features

These **C Map** distinguishing features help customers produce the highest quality systems:

- **Most Configurable** - Easily control optimizations and topology of generated system
- **Most Open** – Full source for all transformation rule/templates and base mechanisms
- **Fastest/Smallest Code** – Generates the most efficient and highest performance code

Integration with Non-Modeled Code

- Hand-written, complex C data types
- Inline C code within Platform Independent Model actions
- Hand-written components, including legacy code and third party components
- Generate custom C-code expansions for tagged domain services

Unique Capabilities for High Performance Systems

- Static initialization of initial class instance populations, reducing start-up time and RAM consumption
- Statically allocated memory pools and a flexible pool manager, with direct control over allocation block sizes and intertask overhead.
- Task-local memory pools, ensuring unneeded intertask protection commonly incurred by general purpose memory managers
- Adaptable to platform specific high performance communication mechanisms from high-volume data transfer

Transformation Performance

- Generates > 50,000 SLOC/minute
- Fastest edit-transform-debug cycle for effective iterative development and testing

Architectural Configurability

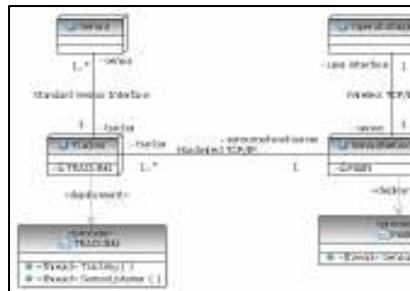
- Reconfigure across task & processor topologies for performance prototyping, varying product line configurations, etc.
- Flexibly redeploy model components in C++ and/or Java in addition to C.

Platform Support

- **C Map** generates systems for immediate deployment on 14 different embedded and workstation target environments

UML2 Modeling Support

- Architectural-level Domain Models/Packages and interfaces
- Classes, attributes, class operations, signals
- Associations and association classes
- Generalization including polymorphism support
- Interface classes, realization
- Harel State Semantics including state entry and exit actions, transition actions and guards, composite states, pseudo states, and history
- UML2 Standard Action Semantics



Product Support

- PathTECH on-line tech resource site
- Effective MDA training & mentoring
- Fully-executable sample systems

The **C Map** generates complete, efficient system implementations from your platform independent models. With *Self Optimization* features that produce smaller and faster code than anything else out there, and open, rules-based transformations to give you the control you need, the **C Map** is your fastest path to building high-quality, high performance systems. *Generate Results: Real Models, Real Code, Real Fast.*

To access white papers, see demos or to request product evaluations, please visit www.pathfindermda.com