

# Component Portability Infrastructure

## Middleware for Waveform-Ready™ Processing Platforms

- Improved waveform code portability with standards-based interfaces
- Increased interoperability using container technology
- Quicker time to market

The Component Portability Infrastructure (CPI) from Mercury Computer Systems is an innovative middleware solution that simplifies programming of heterogeneous processing environments consisting of field-programmable gate arrays (FPGA), general-purpose processors (GPP), digital signal processors (DSP), and high-speed switch fabrics. CPI greatly improves code portability, interoperability, and performance in FPGA- and DSP-based environments by providing well-defined waveform component APIs with a set of infrastructure building blocks that act as a hardware abstraction layer (HAL).

Today's myriad communications standards and rapidly evolving new-generation waveforms have created a need to build communications systems that are ready to accept any present or future waveform. Mercury Waveform-Ready™ processing platforms combine the latest processor, transceiver, and interconnect technologies with the CPI to help customers meet this challenge.

Building on the concepts introduced by the U.S. Government's Software Communications Architecture (SCA) standard (Figure 1), CPI extends component-based architectures into FPGAs and DSPs to decrease development costs and time to market through code portability, reuse, and ease of integration (Figure 2).

### Improved Code Portability

CPI increases the portability of waveform applications in heterogeneous processing platforms by providing APIs, software modules, and intellectual property (IP) cores that abstract the complexities of the underlying

hardware platform away from the application developer. This layer, known as containers, provides the control, configuration, and communication abstractions consistent with a system-level component architecture compatible with the SCA (Figure 3). It exploits the underlying hardware capabilities and performance, while dramatically reducing dependencies of application code on platform technologies, topologies, and configurations. Waveform components use CPI's containers through open and non-proprietary interfaces. Because the interfaces are standardized, they can be reused in any other platform that supports the same interfaces.

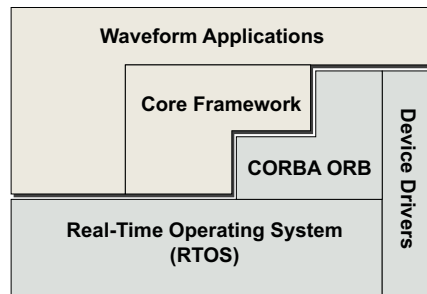


Figure 1. Software Communications Architecture (SCA)

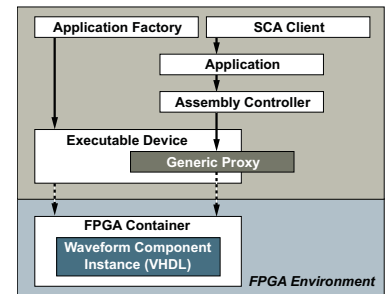


Figure 2. FPGA component implementations

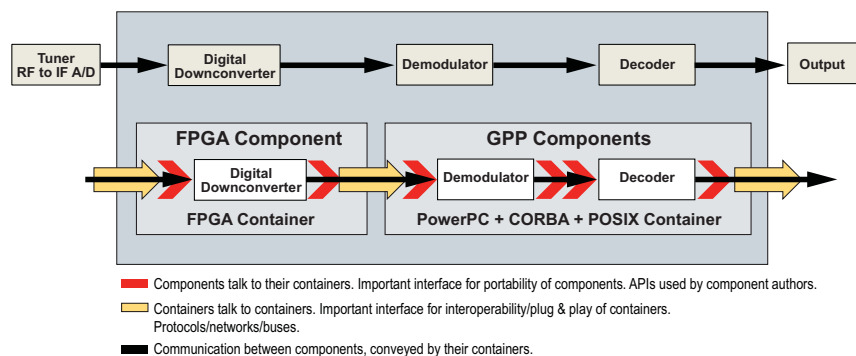


Figure 3. CPI container flow diagram

## Increased Interoperability

CPI also facilitates the interoperability of components executing on different computing device technologies. Typical waveform applications consist of multiple distributed components operating within a heterogeneous embedded environment (Figure 4). CPI provides an environment for FPGA, GPP, and DSP components to seamlessly interoperate. When components communicate with one another, their containers mediate the communication and route it over the appropriate path.

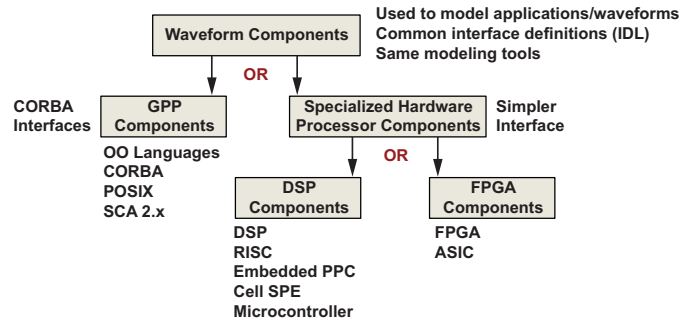


Figure 4. CPI-supported component types

## Faster Time to Market

CPI allows an application to be brought to market faster. CPI's abstraction layer eliminates the need for the application developer to become intimately familiar with the underlying hardware platform. The developer needs to understand and work with only the open and non-proprietary component interfaces that are supported by CPI. In addition, CPI's standard interfaces facilitate the reuse of IP, thereby allowing developers to integrate existing components onto the platform quickly and easily.

## SCA Core Framework Agnostic Design

Many of today's government communications systems are designed to be SCA-compliant to increase interoperability among different platforms. The SCA specification addresses waveforms written for general-purpose software environments, which in turn support a significant subset of the POSIX OS standard, CORBA, and high-level programming languages. CPI operates in these SCA-compliant environments to extend interoperability and waveform code portability into FPGA- and DSP-based platforms. The software modules necessary to easily integrate the infrastructure with any SCA v2.2+ compliant Core Framework are provided as a part of CPI, allowing users to continue working with their preferred Core Frameworks. At your request, Mercury can also provide a JTRS-certified SCA Core Framework via the Mercury Partner Network.

## Code Portability for FPGA Environments

Advancements in silicon technologies continue to fuel generations of FPGAs that are capable of delivering unprecedented levels of performance. Once used exclusively for rapid prototyping, FPGAs now work side-by-side with DSPs and GPPs to deliver some of the world's highest performing programmable computing solutions. Due to their high level of reconfigurability and tremendous I/O capabilities, FPGAs are attractive solutions for wideband communications and SATCOM systems. However, as the complexity of these systems continues to increase, designers are continually faced with new integration challenges that exist both on- and off-chip.

To overcome these challenges, CPI provides a pre-validated set of building blocks to interface the FPGA waveform applications with high-performance switch fabrics, onboard memory, system command and control, and wideband I/O. Building on Mercury's widely used FPGA Developer's Kit (FDK), CPI's non-proprietary interfaces act as an abstraction layer to increase the portability of FPGA applications. A verification suite is also included to facilitate debugging and reduce development time.

## Open Core Protocol (OCP) Profiles

CPI uses the industry-standard Open Core Protocol to define the interfaces for FPGA environments. OCP delivers a non-proprietary, openly licensed, core-centric protocol that comprehensively describes the system-level integration requirements of IP cores. OCP eliminates the task of repeatedly defining, verifying, documenting, and supporting proprietary interface protocols. A clear advantage of using OCP to describe a core's interfaces is that the mechanisms through which one OCP interface can talk to another are clearly defined by the OCP specification. Even if two connected cores have dissimilar interfaces, the fact that they are valid OCP interfaces means that the information needed to resolve those dissimilarities is readily available.

CPI uses well-defined OCP-compliant profiles to define signals and semantics for control, configuration, data, and memory interface patterns. These OCP profiles support waveform component control and configuration, FIFO streaming with flow control, message passing with random addressing and buffer reuse, and memory interfaces for SRAM or DRAM as well as on-chip memories.

Waveform-Ready and Challenges Drive Innovation are trademarks of Mercury Computer Systems, Inc. Other products mentioned may be trademarks or registered trademarks of their respective holders. Mercury Computer Systems, Inc. believes this information is accurate as of its publication date and is not responsible for any inadvertent errors. The information contained herein is subject to change without notice.

Copyright © 2007 Mercury Computer Systems, Inc.

1130.00E-0207-DS-CPI



### Corporate Headquarters

199 Riverneck Road  
Chelmsford, MA 01824-2820 USA  
+1 (978) 967-1401 • +1 (866) 627-6951  
Fax +1 (978) 256-3599  
www.mc.com

### Worldwide Locations

Mercury Computer Systems has R&D, support and sales locations in France, Germany, Japan, the United Kingdom and the United States.

For office locations and contact information, please call the corporate headquarters or visit our Web site at [www.mc.com](http://www.mc.com).