Globus® Toolkit C Web Services Core

The Globus® Toolkit C Web Services Core (C WS Core) components provides a small, efficient implementation of web services, including tooling to generate client and service bindings, implementations of core WSRF services and port types for services, a web service hosting environment, and client programs. The C WS Core implementation is based on web services standards: SOAP 1.1, HTTP, XML Schema, WS-Addressing, WS-Security, WS-Notification, WS-ResourceLifetime, WS-Resource Properties, WS-ServiceGroup. C WS Core builds upon technology and experience used in developing the Globus® Toolkit, including the **Grid Security Infrastructure** (GSI) and **Globus eXtensible I/O** (XIO),

The web services tooling program **globus-wsrf-cgen** converts WSDL documents and XML Schema documents which describe services, data types, and messages into C data structures and functions which can be used for web service and client implementations.

The schema parser handles XML enumerations, unions, choices, model groups, sequences, default values, and many styles of type restrictions supported by the **XML Schema** specification. Type bindings simplify allocation, management, serialization, and deserialization of SOAP messages. Deserialization is done via an efficient XML pull parser. XML types may be used with or include XML wildcard elements.

Client stubs support both service endpoint URIs and WS-Addressing Endpoint References when invoking web service operations. Client SOAP calls may be optionally done via **nonblocking API** functions for use in callback-driven code. The client engine automatically detects in-process service invocations. When these occur, the engine avoids XML serialization for higher performance.

Service bindings enable an application writer to focus on writing service code without having to worry about network I/O or message processing. **Operation providers** allow code implementing a WSDL port type to be reused by multiple services. The C WS Core includes operation providers for the WS-ResourceProperties, WS-BaseNotification, WS-ResourceLifetime, and WS-ServiceGroup standards. The C WS Core also includes APIs for creating service-specific resources, resource properties, notification topics, and service group entries.

The program **globus-wsc-container** implements a stand-alone web service hosting application which loads services on-demand. It uses the HTTP/1.1 implementation provided by **Globus XIO** to accept service requests and the globus service engine to dispatch service requests. Per-service handler chains allow customized processing of SOAP message headers and attributes. Current handlers provided by C WS Core include WS-Secure Messaging and WS-Addressing. The **service engine API** may be linked into existing applications to provide a lightweight service interface.

The C WS Core provides **client programs** for interacting with services which implement the core WSRF port types, such as GetResourceProperties or SetTerminationTime. These small

native applications provide a fast way to script or interact with web services via the command-line.