C WS Core

- Implementation of the Web Services Resource Framework (WSRF) and WS-Notification (WSN) family of specifications
  - WS-ResourceProperties support: No QueryResourceProperties provider
  - WS-Notification support is only receiver-side (NotificationConsumer)
- Builds on Globus C Asynchronous Events
  - Non-blocking calls and callbacks
  - Flavors and thread pooling
C WS Core

- Service Container and Embeddable Service Engine API
- WSDL Binding Generation
  - Pure C parser/generator
  - Doc/lit support only
- Security
  - Transport
  - SecureMessage (Integrity only)
- Type Registries
- Message Handlers
Clients

- End-to-end solution for WSRF-enabled clients in C
  - Pure C types and interfaces (non-blocking and blocking)
  - EPR encapsulation
  - RP querying
  - Asynchronous NotificationConsumer API
Services

- Full WS support, some WSRF/WSN support
  - Doc/lit operation invocations
  - WS-I Basic Profile conformance
  - WS-Addressing by default

- Service Container
  - globus-wsc-container
  - No RegistryService (accepting contributions now!)

- Engine API
  - globus_service_engine_start
  - globus_service_engine_stop

- Service Modules
Service Implementation

- **Operation Impls and Init**
- **Resource API**
  - Factories and Resource Creation: `globus_resource_create`
  - Creating RPs: `globus_resource_create_property`
- **Operation Providers**
  - Operation Provider Modules
  - How to load WS-ResourceProperties and WS-ResourceLifetime operation providers
Example Client

- **Module Activation:**
  - `globus_module_activate(COUNTERSERVICE_MODULE);`

- **Create client handle:**
  - `handle = CounterService_client_handle_init(..);`

- **Create Resource:**
  - `EPR = CounterPortType_createCounter(handle, endpoint);`
Example Client (cont.)

- Perform operation on resource:
  - CounterPortType_add_epr(handle, EPR, 10);
- Get Resource Property:
  - CounterPortType_GetResourceProperty_epr(handle, EPR, "value");
- Destroy Resource:
  - CounterPortType_destroy_epr(handle, EPR);
- Teardown
  - CounterService_client_handle_destroy(handle);
  - globus_module_deactivate(COUNTERSERVICE_MODULE);
Receiving Notifications

- Write a notification callback
- Init: globus_service_engine_init
- Create Consumer Resource:
  - ConsumerEPR =
    globus_notification_create_consumer(engine, callback);
- Start: globus_service_engine_start
- GetRP(Topic) [optional]
Receiving Notifications (cont.)

- **Subscribe**(ConsumerEPR, <topic>)
- **Block:**
  - while(!done) { globus_cond_wait(..); }
- **Stop:** globus_service_engine_stop
- **Teardown:**
  - globus_notification_destroy_consumer(ConsumerEPR)
- **Destroy:** globus_service_engine_destroy