PRISMTECH™
AN ADLINK COMPANY

Software Productivity Tools for SCA 4.1 Developers

Tim McGuire – Senior Software Developer
PrismTech
November 2017

Speaker Introduction

- **Tim McGuire**
  - tim.mcguire@prismtech.com
- **Senior Software Developer**
  - UML
  - Tooling
  - Eclipse
  - Embedded Operating Systems
- **Background**
  - PrismTech
  - Zeligsoft
  - IBM Rational
  - ObjecTime
  - Bell Canada

PrismTech

- **PrismTech is a leading provider of SDR and SCA solutions**
- **Our Spectra product suite for radio system developers and integrators provides Commercial-off-the-Shelf development tools, software infrastructure and development and test platforms including:**
  - **Spectra CX** - model-driven development tool that greatly simplifies, accelerates, and validates the SDR/SCA development process
  - **Spectra CF** - a high performance, ultra low footprint, COTS SCA Core Framework
  - **Spectra CDB** - a fully integrated and optimized SDR middleware stack running across a wide range of FPGA, DSP and GPP elements
  - **Spectra DTP** - a wideband, high performance, baseband and RF SDR Development and Test platform
  - **Spectra Radio Services and Devices** - a series of standard SCA compliant components that implement the Application Programming Interfaces API's published by JTNC

PRISMTECH™ AN ADLINK COMPANY

Agenda

- Challenges of building SCA SDR systems
- Proven benefits of model-driven development to design complex SCA SDR platforms and waveforms
- Key differences between developer tooling for SCA2 and SCA4 environments
- SCA 4 features which are supported in Spectra CX4
- Technical challenges encountered and overcome in evolving/adapting development tools for SCA2 to SCA4 compliance
- Overview of generated XML and optional C++ code that can be produced with SCA4 development tools

Building SCA SDR Systems

- **Advanced technologies**
  - Widely varying skill sets required
  - Widely varying tool environments
  - Widely varying hardware environments
- **High demand on product families**
  - Multiple products in short time span
- **High demand on flexibility**
  - Applications on multiple platforms
  - Components on different devices
  - Waveform and platform variations

Building SCA SDR Systems …

- **Varying industry applications**
  - Powerful telephony base stations
  - Robust and flexible military radios
  - Space
- **Varying requirements**
  - Power
  - Reliability
  - Security
  - Performance
  - Productivity

PRISMTECH™
AN ADLINK COMPANY

Building SCA SDR Systems …

- **Add SCA**
  - Goal is not to simplify SDR development
    - Introduces SDR Framework and new interfaces
    - Implementation technologies mandated by frameworks
    - Specification details – behavior under error conditions
  - Allow development of software utilized across radio families

Agenda

- Challenges of building SCA SDR systems
- Proven benefits of model-driven development to design complex SCA SDR platforms and waveforms
- Key differences between developer tooling for SCA2 and SCA4 environments
- SCA 4 features which are supported in Spectra CX4
- Technical challenges encountered and overcome in evolving/adapting development tools for SCA2 to SCA4 compliance
- Overview of generated XML and optional C++ code that can be produced with SCA4 development tools

Model Driven Development

- **Address SCA skills gap**
  - Allow people to focus on their expertise
- **Document requirements**
- **Facilitate communication of design**
- **Assist implementation**
  - Wizards, context menus, palette
- **Validation for SCA environment**
  - Easily identify and fix issues
- **Generate**
- **Test**

Model Driven Development …

- **Address the following issues**
  - XML complicated and difficult to edit directly
  - Source code can be complex, especially with properties
  - XML and code can become out of sync without common source (Model)
- **Ease iterative development**
  - Add functionality with each iteration, validate, regenerate and test
- **Integration with other tools**
  - SVN, Git, …

Agenda

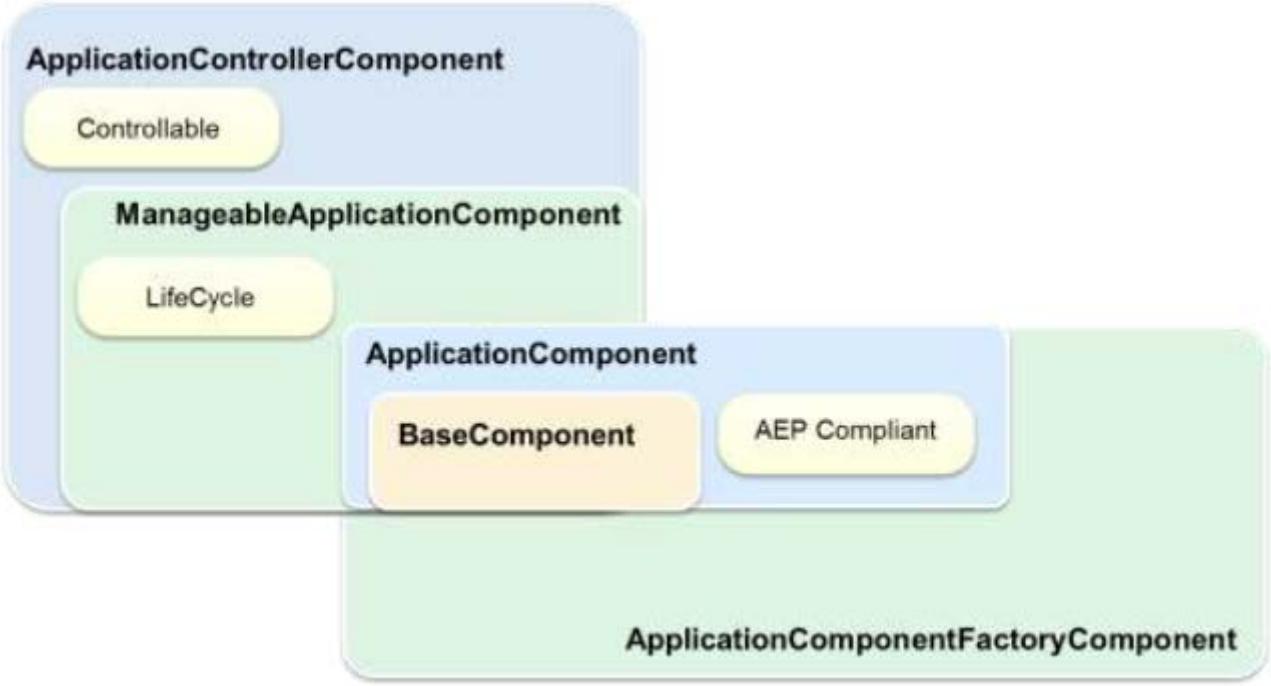
- Challenges of building SCA SDR systems
- Proven benefits of model-driven development to design complex SCA SDR platforms and waveforms
- Key differences between developer tooling for SCA2 and SCA4 environments
- SCA 4 features which are supported in Spectra CX4
- Technical challenges encountered and overcome in evolving/adapting development tools for SCA2 to SCA4 compliance
- Overview of generated XML and optional C++ code that can be produced with SCA4 development tools

Tooling - SCA 2 vs. SCA 4

- **SCA 4.1**
  - Modular components → lighter weight
  - With flexibility comes complexity
    - Units of Functionality (UOFs)
    - Profiles – SCA Application Environment Profile (AEP), Lightweight AEP (LwAEP) and Ultra-Lightweight (ULwAEP)

Tooling – SCA 2 vs. SCA 4 …

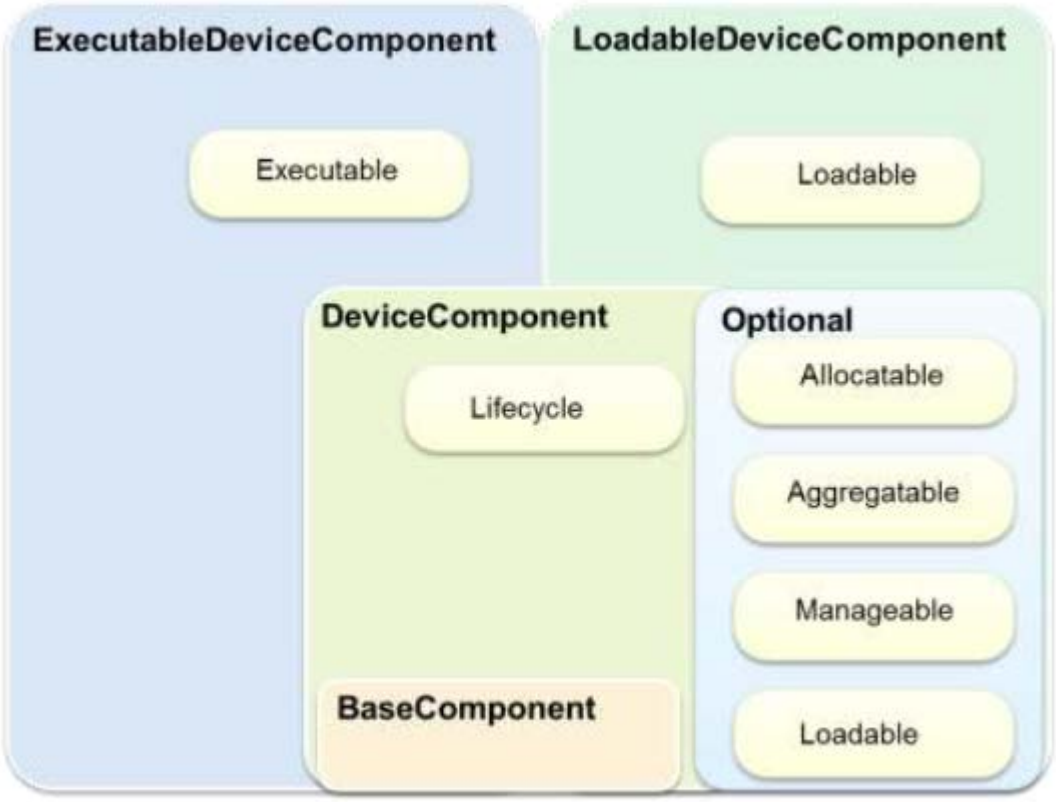
- Support for modeling Application related components and setting their mandatory UOFs



Application Component Units of Functionality

Tooling – SCA 2 vs. SCA 4 …

- Support for modeling Device related component UOFs and setting their mandatory and optional UOFs



Device Components Units of Functionality

Tooling – SCA 2 vs. SCA 4 …

- Support for setting BaseComponent optional UOFs

BaseComponent Units of Functionality

Tooling – SCA 2 vs. SCA 4 …

- **Semantics of finding elements have changed.**
  - NamingService replaced by component registry
  - Updated DomainFinder Types
  - FindBy removed
- **Support for sub-assemblies**
- **Support for multiple assembly controllers**
- **Connections**
  - Uses port fan out connections
  - persistent connections using stringifiedobject
  - DTLTCR and DUBTCR can connect to supports interface

Agenda

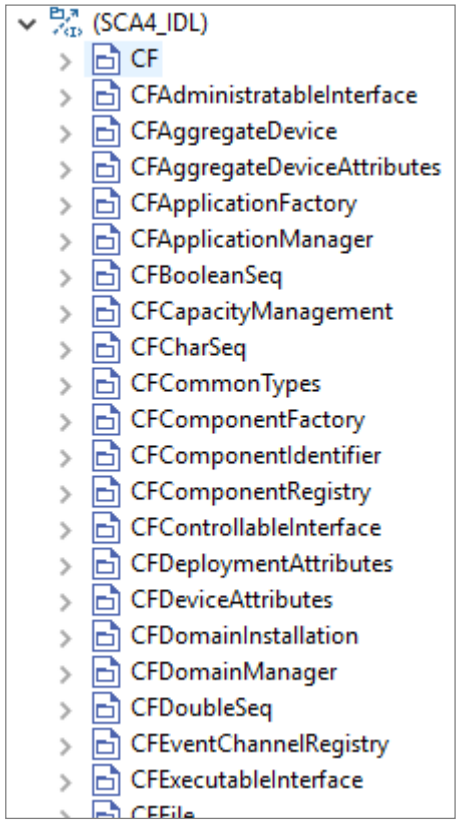
- Challenges of building SCA SDR systems
- Proven benefits of model-driven development to design complex SCA SDR platforms and waveforms
- Key differences between developer tooling for SCA2 and SCA4 environments
- SCA 4 features which are supported in Spectra CX4
- Technical challenges encountered and overcome in evolving/adapting development tools for SCA2 to SCA4 compliance
- Overview of generated XML and optional C++ code that can be produced with SCA4 development tools

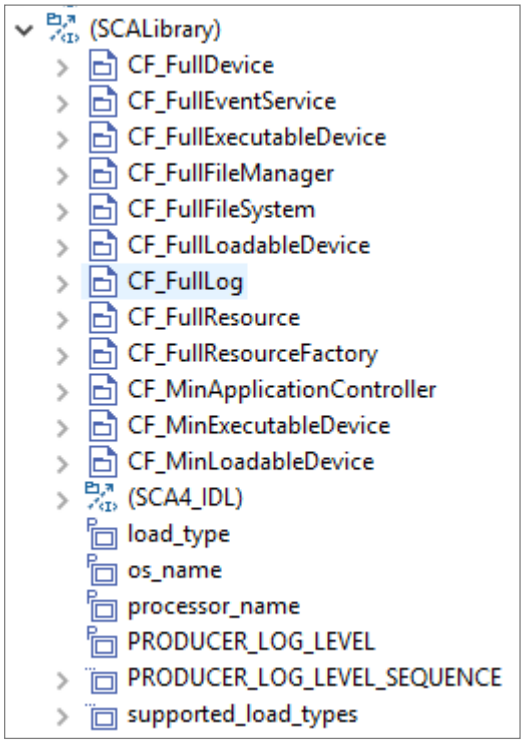
Spectra CX4 Features

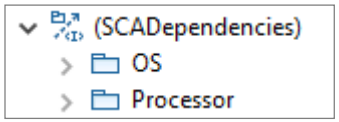
- SCA 4 Model Libraries
- SCA 4 Model Migration
- SCA 4 Component Wizard
- SCA 4 Validation
- SCA 4 XML Generation
- SCA 4 C++ Generation
- RTCORBA
- Non-CORBA Port
- UML Generation

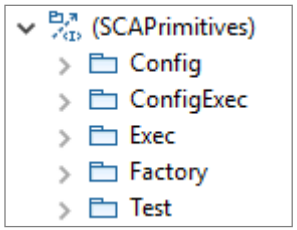
SCA 4 Model Libraries

- SCA4_IDL, SCALibrary, SCADependencies, SCAPrimitives model libraries





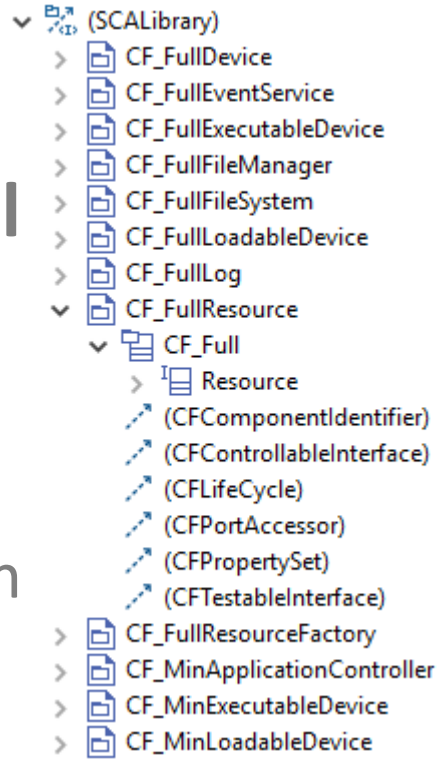




PRISMTECH™
AN ADLINK COMPANY

SCA 4 Model Migration

- In place migration
- Converts a Spectra CX SCA 2.2.x model to a Spectra CX SCA 4.1 model
- Requires model library with SCA 2.2.x similar interfaces
  - SCA 2.2.x Resource maps to ManageableApplicationComponent with CF_Full::Resource UOFs
- Follows Joint Tactical Networking Center (JTNC) migration guide for interfaces



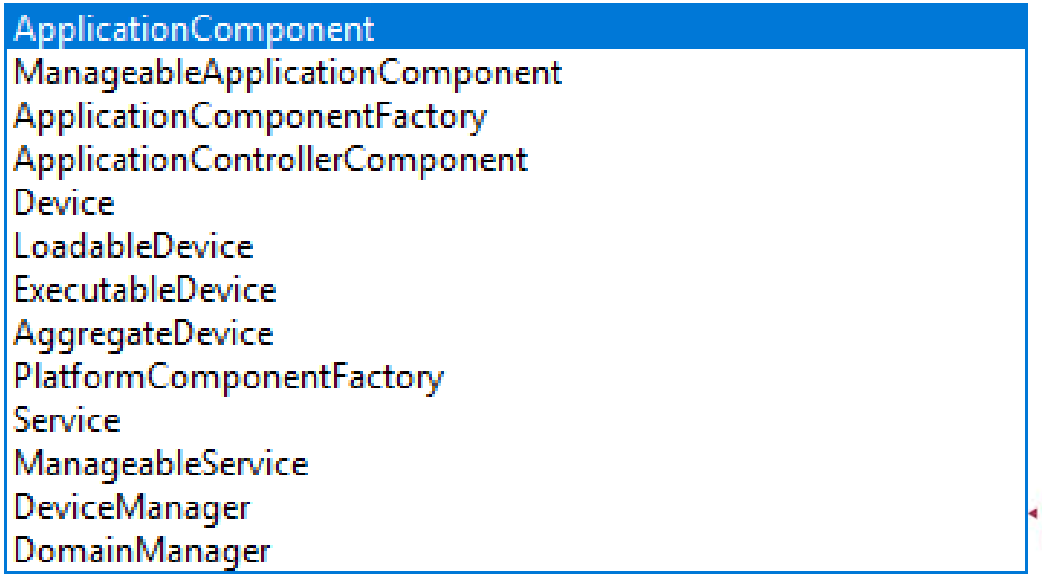
PRISMTECH™ AN ADLINK COMPANY

SCA 4 Model Migration …

1. UML profile changed from SCA 2.2 domain to SCA 4.1 domain
2. Default model SCA 4 model libraries added
3. replace SCA 2.2 concepts with equivalent SCA 4.1 concepts
4. replace references to SCA 2.2 model library elements (e.g. CF IDL files and interfaces) to SCA 4.1 equivalents
5. remove SCA 2.2 model library imports

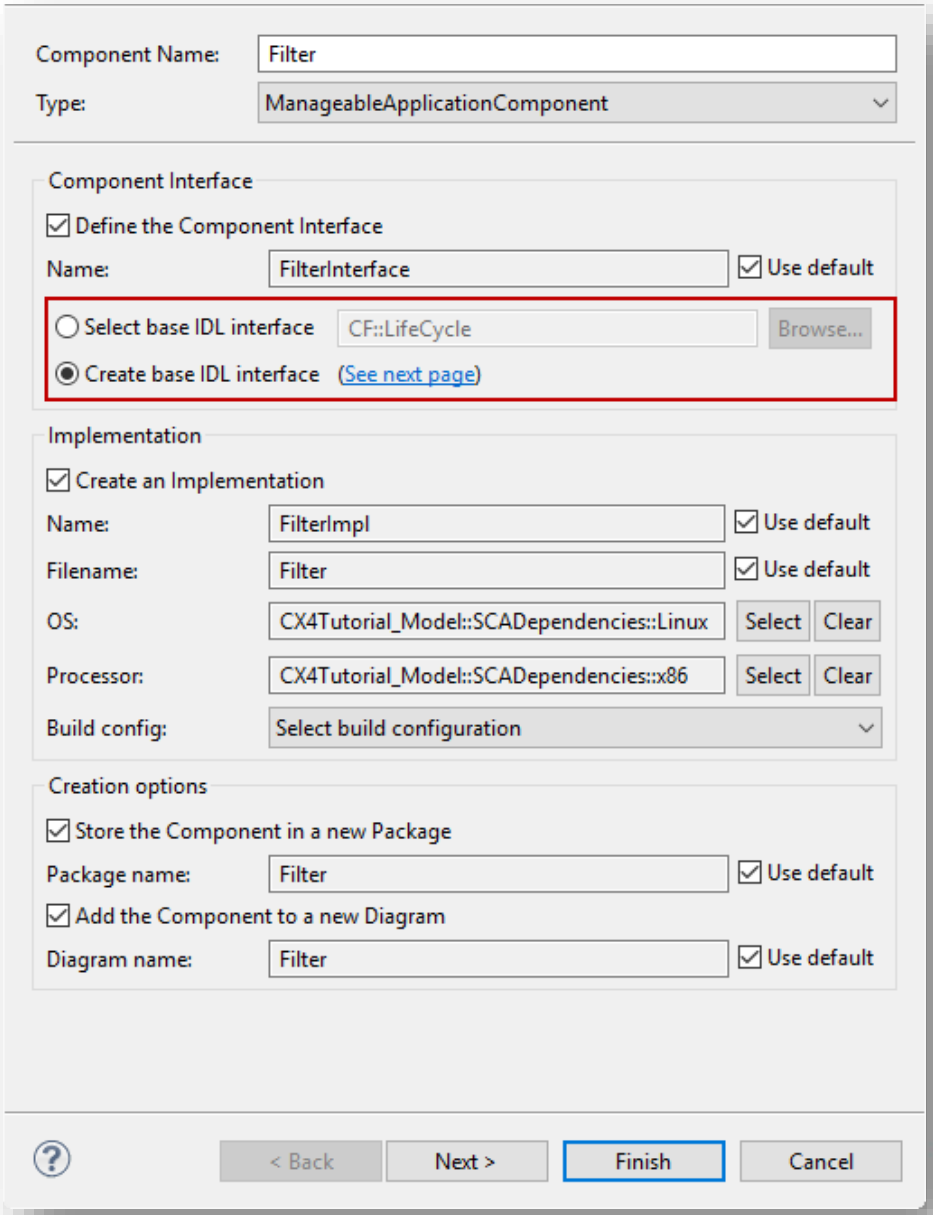
SCA 4 Component Wizard

- Component types require different mandatory UOFs.
- Assists users to create component types with the mandatory UOFs and any optional UOFs
- Supports all SCA 4 component types



PRISMTECH™ AN ADLINK COMPANY

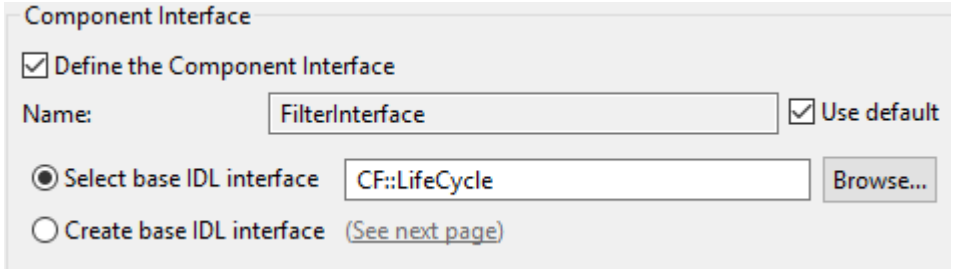
SCA 4 Component Wizard …



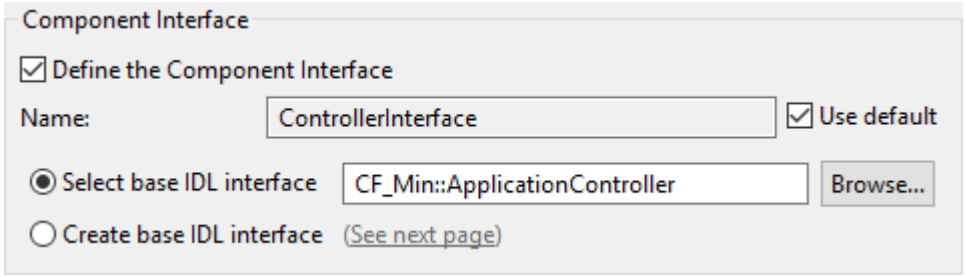
PRISMTECH™ AN ADLINK COMPANY

SCA 4 Component Wizard …

- **Select base IDL interface**
  - Mandatory UOF - Auto set to UOF interface



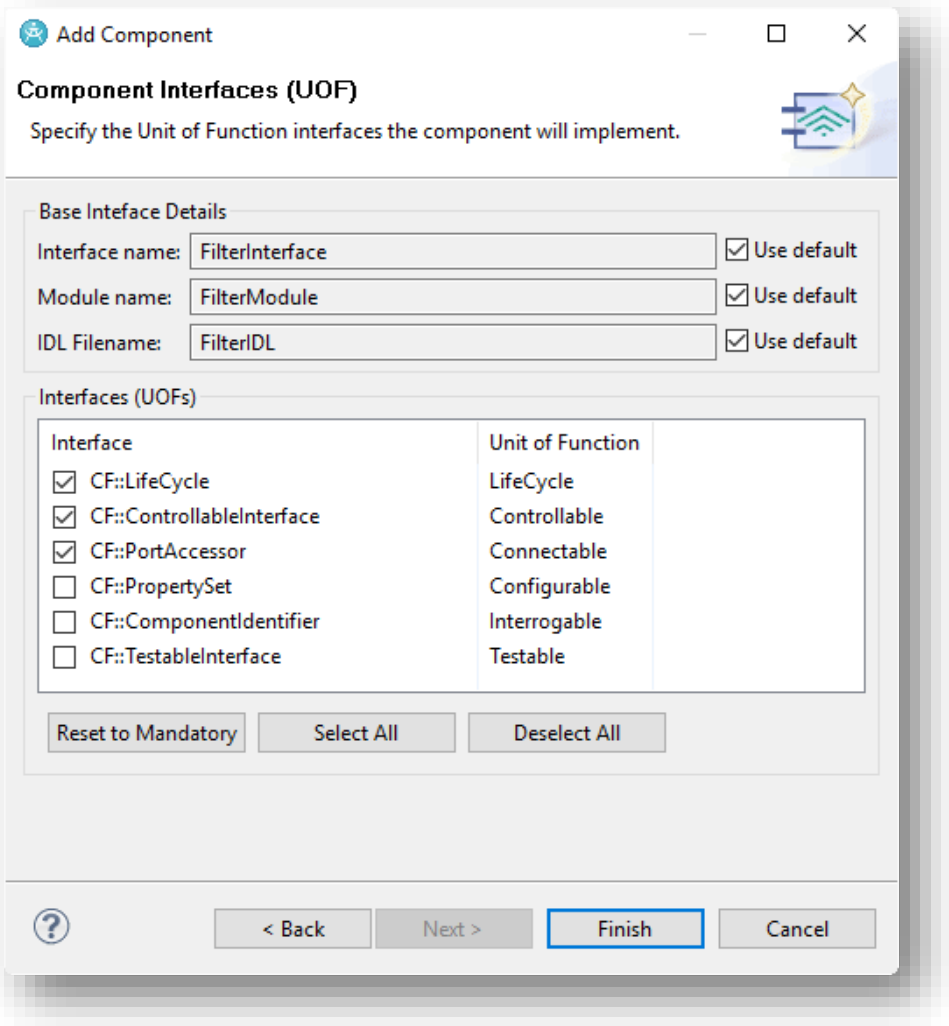
  - Multiple mandatory UOFs - Auto set to SCALibrary interface with mandatory UOFs



PRISMTECH™ AN ADLINK COMPANY

SCA 4 Component Wizard …

- Create base IDL interface



PRISMTECH™ AN ADLINK COMPANY

SCA 4 Validation

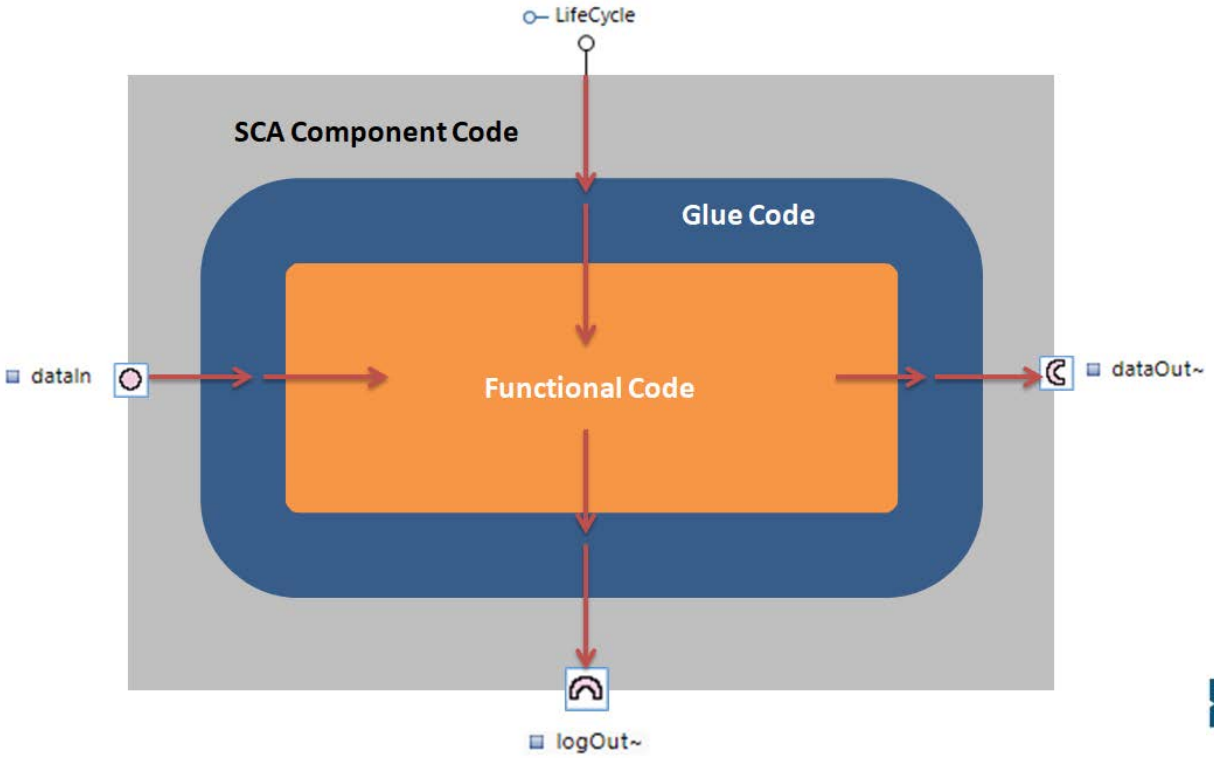
- **Validation can be triggered manually and is done before XML and C++ generation**
- **Spectra modeling CX4 limits and guides users**
  - Invalid drag and drop disabled
  - Dialogs are filtered to show only relevant elements
- **SCA 4 UOF Validations**
  - Mandatory UOFs for component type
  - Component with ports (PortAccessor UOF)
  - Component with properties (PropertySet, CapacityManagement, TestableInterface)
- **Other validations**
  - At least one controller per assemblyComponent
  - Connectors with compatible interfaces
  - Sub-assemblies circular references
  - Values are set for any SCA properties that require C++ code generation
- **Many more validations …**

SCA 4 XML Generation

- Generate from Model, Package, AssemblyComponent, Node, Component or Platform
- Validates before generation
- Generates XML for all dependent elements
- Complete set of SCA 4 XML and DTD files

SCA 4 C++ Generation

- Spectra CX4 generates
  - SCA component code
  - Functional code
  - Glue code that ties them together



PRISMTECH™
AN ADLINK COMPANY

SCA 4 C++ Generation …

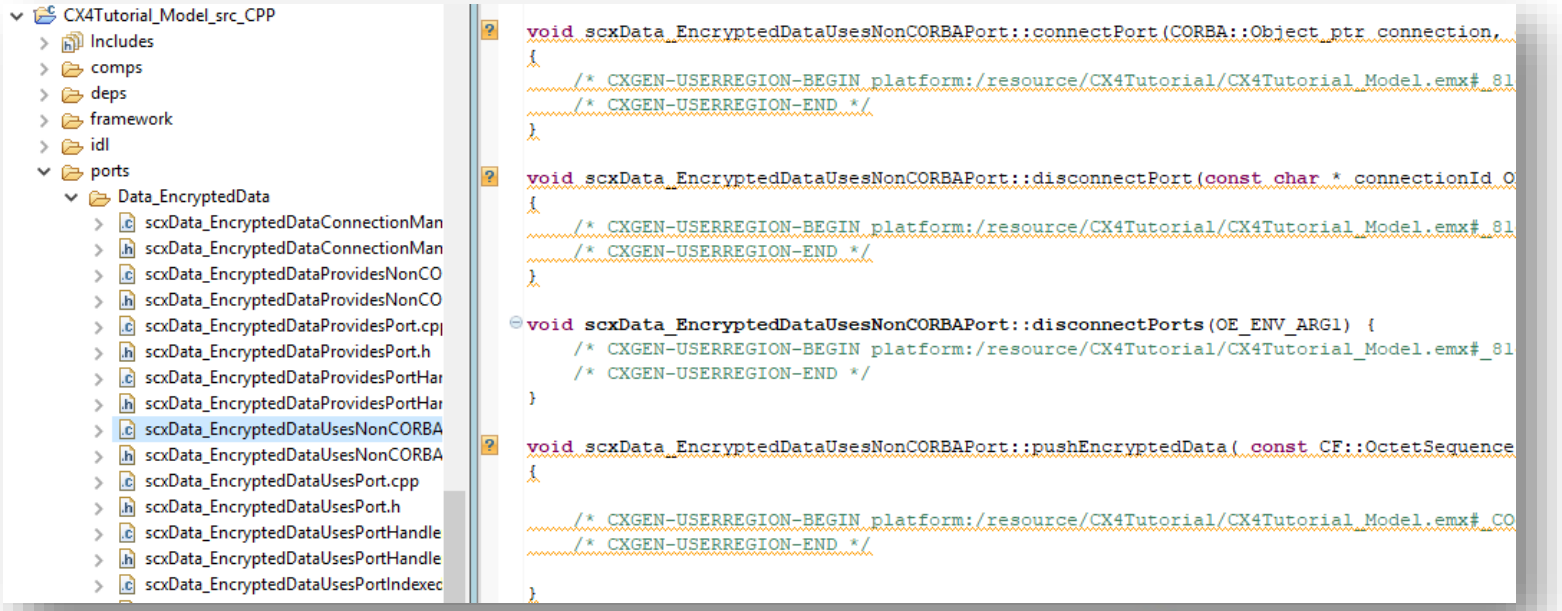
- User code stored in model
- Generates C++ code and build environment
- Multiple target environments (OS, Processor, …)
- Optional package layout generation
  - C++ code folders follow model package layout
- Managed builder
  - Changes made to C++ files external to Spectra CX4 are written back into Spectra CX4 model
- Option to delegate to functional code

RTCORBA

- **Modeling, validation and C++ code generation**
  - ThreadPool, ThreadPool with Lanes
    - Can be modeled and reused in model and C++ code
  - PriorityModelPolicy
    - Can be modeled and reused
  - PriorityBandedConnectionPolicy
    - Can be modeled and reused
  - Protocols (TCP, UDP, …)
    - Can be modeled and reused
    - More than one protocol can be set on element
  - PrivateConnectionPolicy

Non-CORBA Port

- Executable Arguments are passed to the uses and provides port constructor
- APIs similar to the CORBA ports but the user implements the APIs



PRISMTECH™ AN ADLINK COMPANY

UML Generation

- Generation of UML classes representing generated C++ code
  - Worker, ports, …
- Worker UML class can be modeled with user defined C++ classes and generated
- Document generated code

Agenda

- Challenges of building SCA SDR systems
- Proven benefits of model-driven development to design complex SCA SDR platforms and waveforms
- Key differences between developer tooling for SCA2 and SCA4 environments
- SCA 4 features which are supported in Spectra CX4
- Technical challenges encountered and overcome in evolving/adapting development tools for SCA2 to SCA4 compliance
- Overview of generated XML and optional C++ code that can be produced with SCA4 development tools

SCA 4 Model Migration

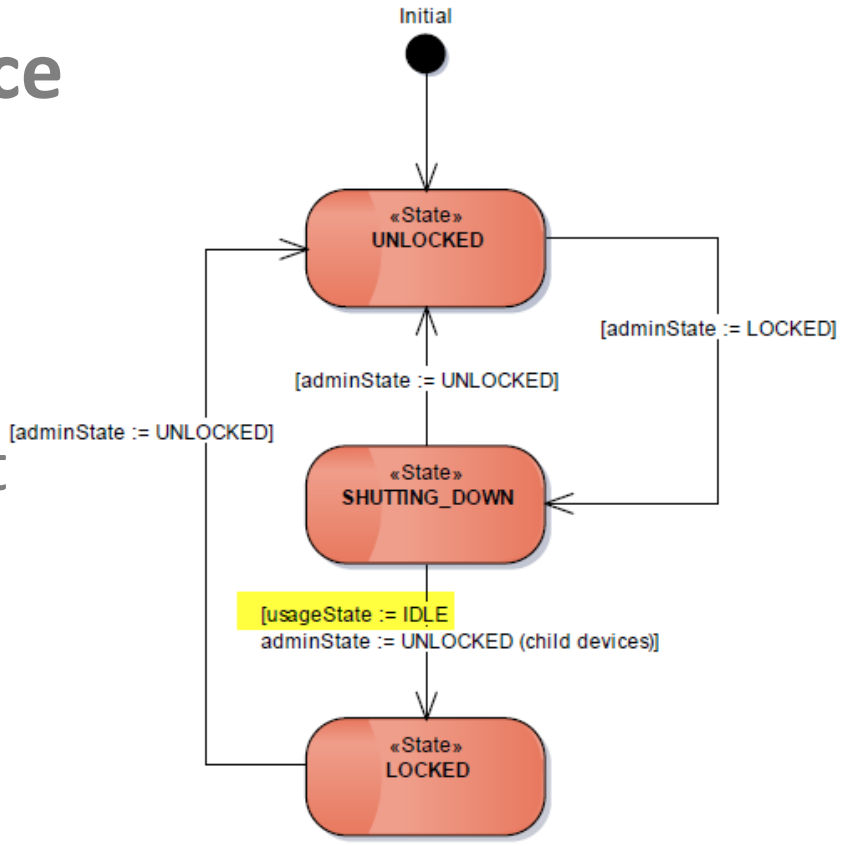
- **Replace SCA 2.2.x concepts with equivalent SCA 4.1 concepts**
  - concepts that migrate to the same concept in SCA4
  - concepts that migrate to a 'trivial' SCA4 equivalent (i.e. concept properties augmented/reduced)
  - concepts that migrate to in a non-trivial way to an SCA4 equivalent (i.e. components)
  - concepts not supported in SCA4

SCA 4 Code Generation

- **SCA 2.2.x code static except for user defined interfaces and properties**
- **SCA 4.1 code dependent on UOFs**
  - Final code is lighter weight
  - Determining what code to write is more difficult
    - Which UOFs to add to component?
    - Some UOFs depend on other UOFs
    - Some UOFs depend on Properties
  - Once the validation and generation rules are written then generation is always correct

SCA 4 Code Generation …

- **AdministratableInterface /CapacityManagement**
  - Setting the adminState sets usageState (if CapacityManagement UOF)



State Transition Diagram for adminState

PRISMTECH™ AN ADLINK COMPANY

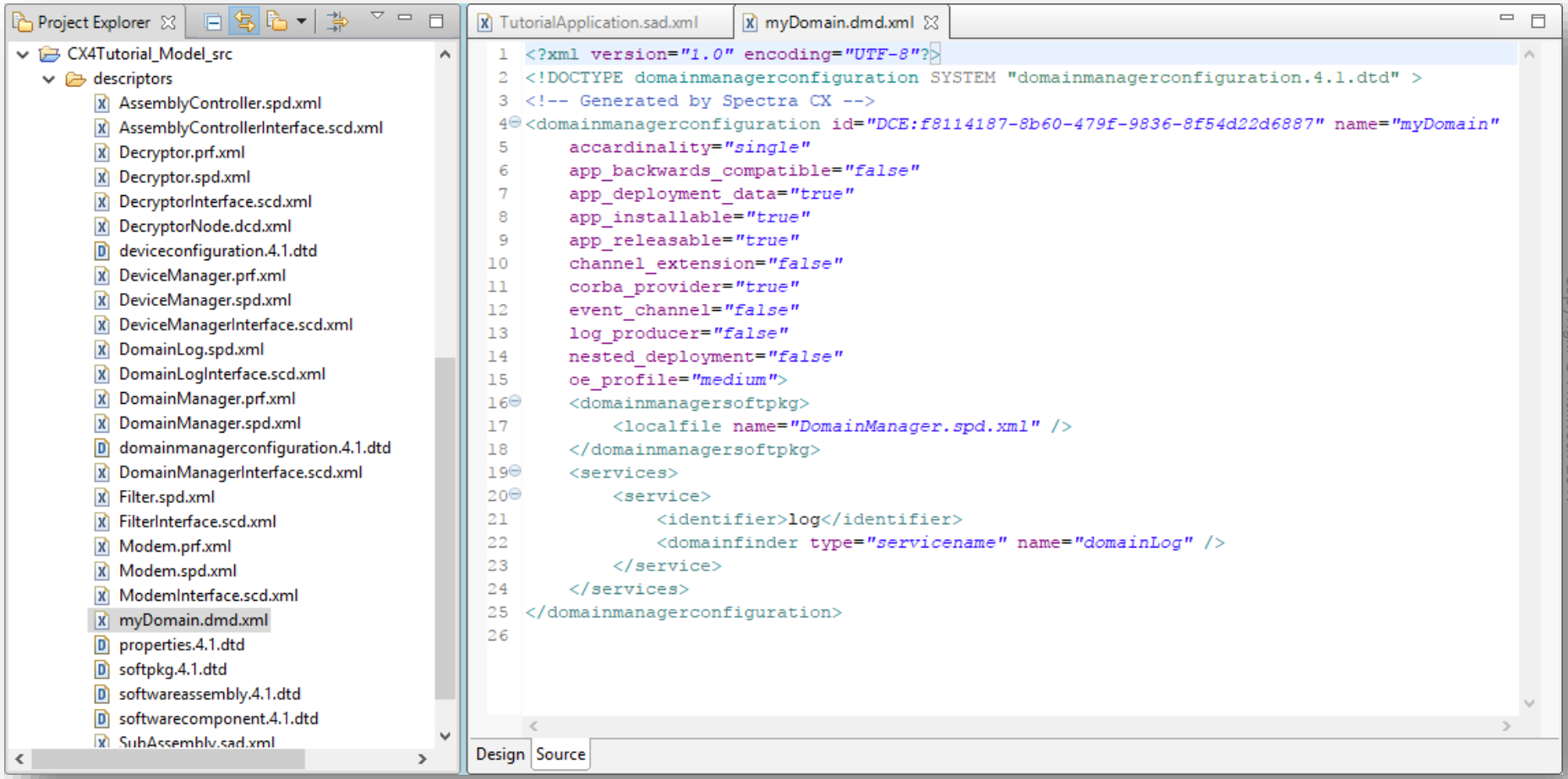
SCA 4 Code Generation …

- **CapacityManagement/AdministratableInterface**
  - SCA262 The deallocateCapacity operation shall raise the CF::InvalidState exception, when the DeviceComponent's **adminState** is LOCKED
- **CapacityManagement/DeviceAttributes**
  - SCA516 The *deallocateCapacity operation shall raise the CF::InvalidState exception, when the* DeviceComponent's **operationalState** is DISABLED.
- **CapacityManagement/PropertySet**
  - External allocation properties need CapacityManagement and PropertySet UOFs

Agenda

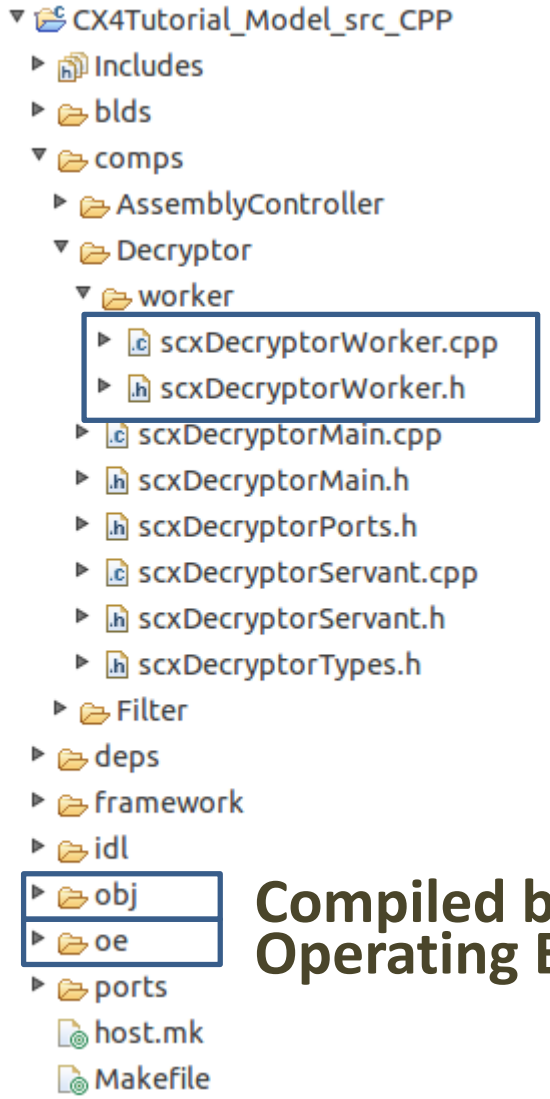
- Challenges of building SCA SDR systems
- Proven benefits of model-driven development to design complex SCA SDR platforms and waveforms
- Key differences between developer tooling for SCA2 and SCA4 environments
- SCA 4 features which are supported in Spectra CX4
- Technical challenges encountered and overcome in evolving/adapting development tools for SCA2 to SCA4 compliance
- Overview of generated XML and optional C++ code that can be produced with SCA4 development tools

XML Generation Overview



PRISMTECH™
AN ADLINK COMPANY

C++ Code Generation Overview



Functional Code

Compiled binaries
Operating Environment Dependent Code

PRISMTECH™ AN ADLINK COMPANY

C++ Code Generation Overview …

- Functional code entered between CXGEN user region tags
- Saved back into the model

```
void scxDecryptorWorker::_CF_Full_Resource_start(OE_ENV_ARG1) {
    /* CXGEN-USERREGION-BEGIN platform:/resource/CX4Tutorial/CX4Tutorial_Model.emx#_q
    // enter user code here
    /* CXGEN-USERREGION-END */
}
```

Integrating Tooling with other SCA 4.1 Components

- Tooling is just one element of a complete SCA 4.1 solution
- PrismTech is pleased to announce its new Spectra CF SCA 4.1 runtime Q1 2018

SPECTRA

PRISMTECH™
AN ADLINK COMPANY

PRISMTECH™
AN ADLINK COMPANY

Thank you

www.prismtech.com

PRISMTECH™
AN ADLINK COMPANY