



combined **strength**

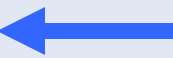
Wireless Innovation Forum Implementing a Generic Front Panel

November 30, 2010

Presentation by: Serge Harnois

Location: Washington

Agenda

- Introduction 
- Relevant SCA APIs for HMI development
- Proposed design patterns and best practices
- Virtual Front Panel project
- Conclusion

SCA well defined areas

- Cross platform communication infra-structure with CORBA
- Variety of components with their characteristics
 - Devices
 - Resources
 - Services
- Set of standard application programming interfaces (APIs) to exchange information between components
 - PropertySet
 - TestableObject
 - LifeCycle
 - PortSupplier
- Application deployment mechanism in order to launch and teardown applications

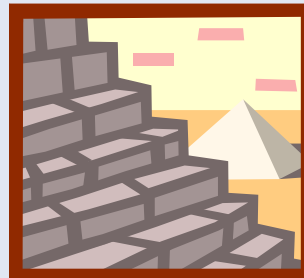
How to use it ?

- In the context of control and monitor applications

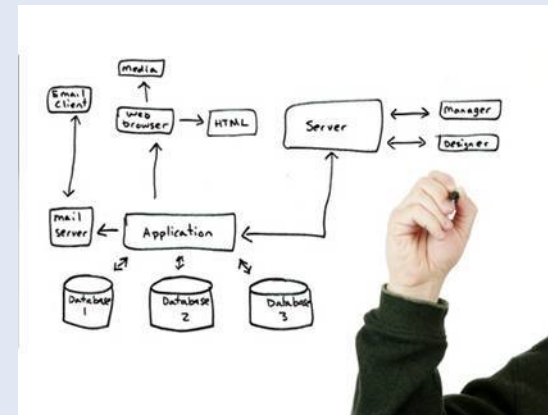
— Which kind of interface to use for HMI?



— How application has to be structure?



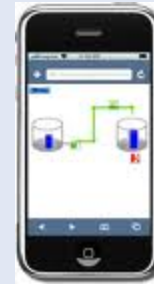
— Is there design pattern or best practice that can be use?



Understand control and monitor application requirements

- Example of control and monitor application

- Web pages
- PC application
- Keypad and Display
- Intelligent Handset
- Etc...

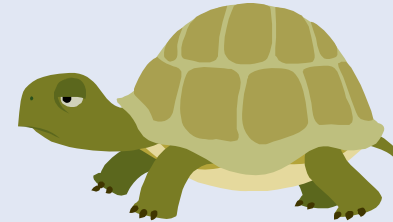


- What do they have in common

- They have dynamic content
- They are initiating actions
- They need to be refresh

Understand control and monitor application requirements

- Radio control and monitoring does not impose tight requirements for speed because human reaction time is relatively slow



- The interface should easily provides a suitable way to expose an understandable interface for each possible kind of platform and waveform application



- HMIs are not very useful if the data displayed is out of date or if the user needs to refresh it manually, so there is also a need for data to be refresh.



Agenda

- Introduction
- Relevant SCA APIs for HMI development ←
- Proposed design patterns and best practices
- Virtual Front Panel project
- Conclusion

SCA Interfaces adapted for HMI

- Two Apis can fulfill all those requirements:
 - PropertySet
 - **An API like *PropertySet* can be used as an interface between any generic or specific tool and SCA applications**
 - **It is not the most efficient way to exchange information in term of speed but it is in term of visibility of your applications features and characteristics**
 - Event Channels
 - **Another SCA standard API can be used to fulfill this aspect of the tool design; the Event Channel is perfectly adapted for this kind of requirement**

PropertySet API is adapted for HMI

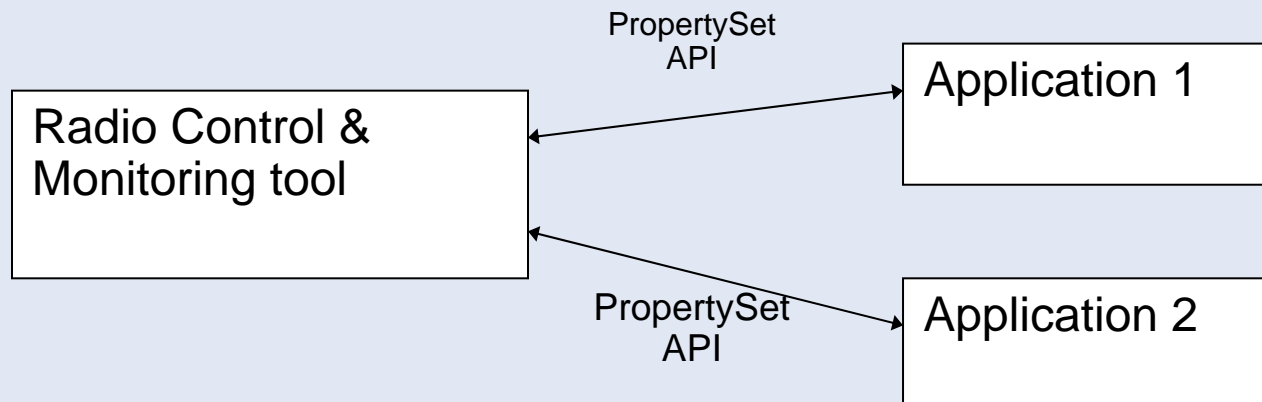
- *The PropertySet interface provides two functions*
 - “query” in order to get property value
 - “configure” to modify property value
- This interface allows to configure and query properties of all types, i.e:
 - simple
 - integers
 - strings
 - floating points
 - Etc...
 - sequences of simples
 - structure of simple
 - sequence of structures

PropertySet API is adapted for HMI (cont'd)

- *Properties are readable and understandable*
 - *The combination of the name, type, and description allow properties to talk for themselves*
- *Properties are described in the SCA Application metadata (XML files)*
- *A Control and Monitoring tool can be developed in no time with properties attached to dynamic elements*

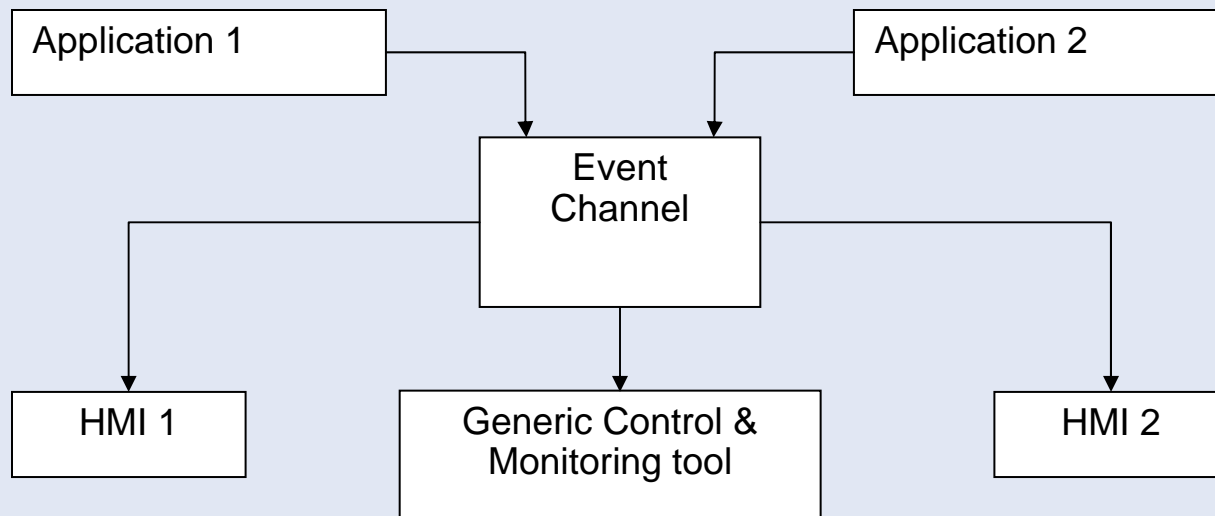
PropertySet API is adapted for HMI (cont'd)

- *The following figure shows how simple a HMI can be integrated to an SCA design with the PropertySet interface*
- *No need to define specific Interface Definition Language (IDL)*
- *The entire mechanic is already present in the SCA standard interface part of the Core Framework*



Event Channel API is adapted for HMI

- The Multiple Input Multiple Output (MIMO) structure of an event channel allows more than one application to broadcast information intended to be refreshed by any listener on the event channel
- It also allows more than one listener, in that case HMIs and generic control and monitoring tools, to receive this information and update their display if needed

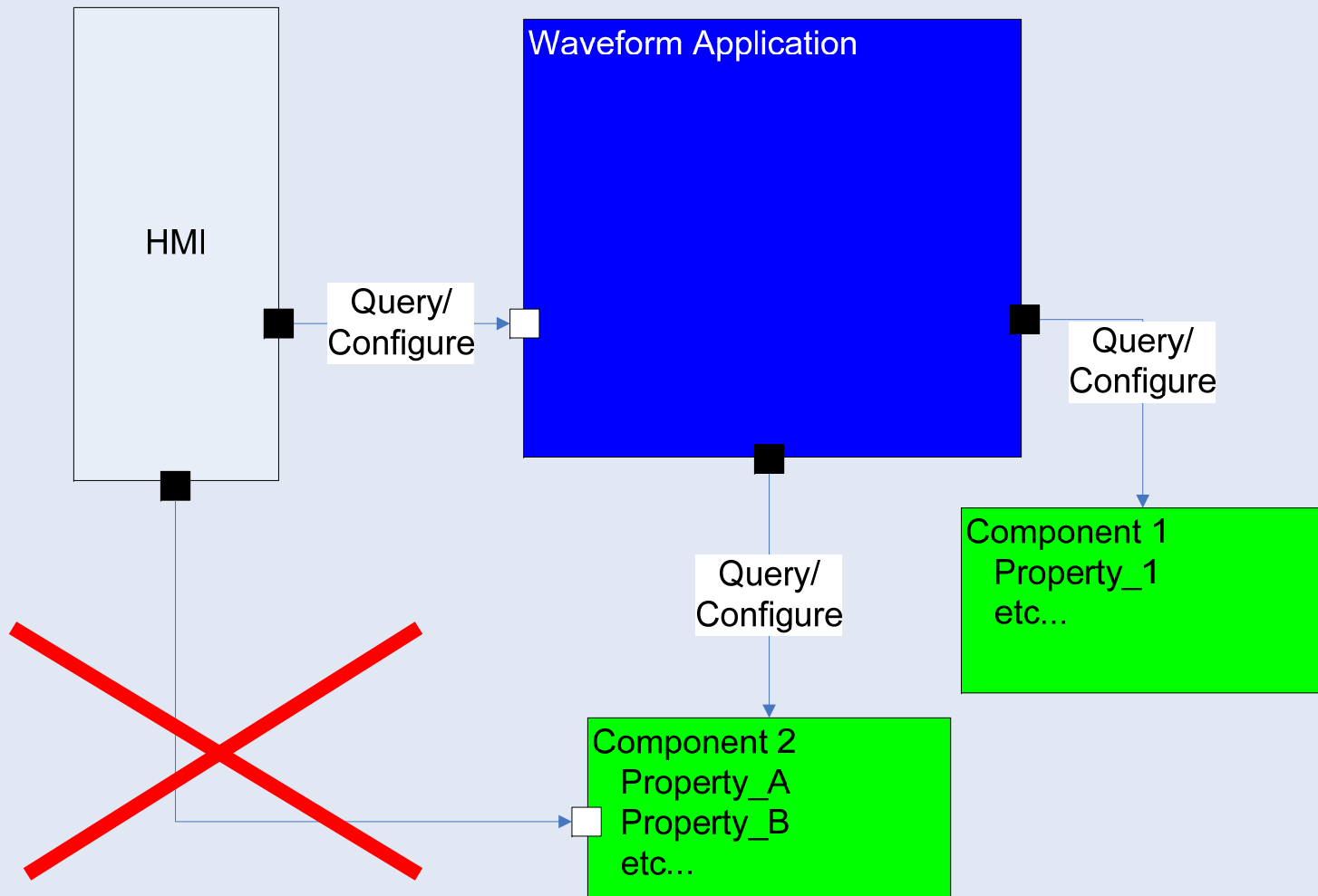


Agenda

- Introduction
- Relevant SCA APIs for HMI development
- Proposed design patterns and best practices ←
- Virtual Front Panel project
- Conclusion

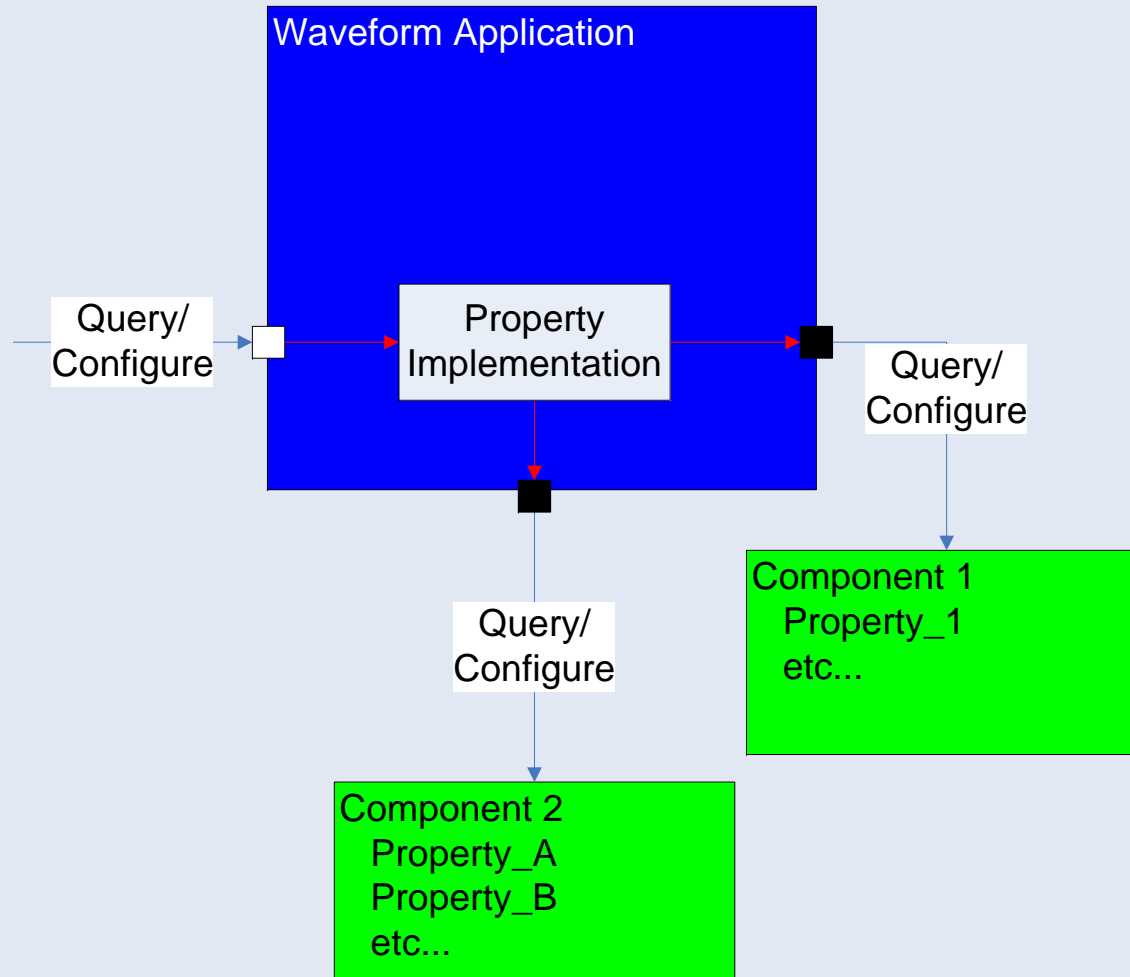
Design practical approaches

- Assembly Controller of an SCA application is always the only entry point of the application



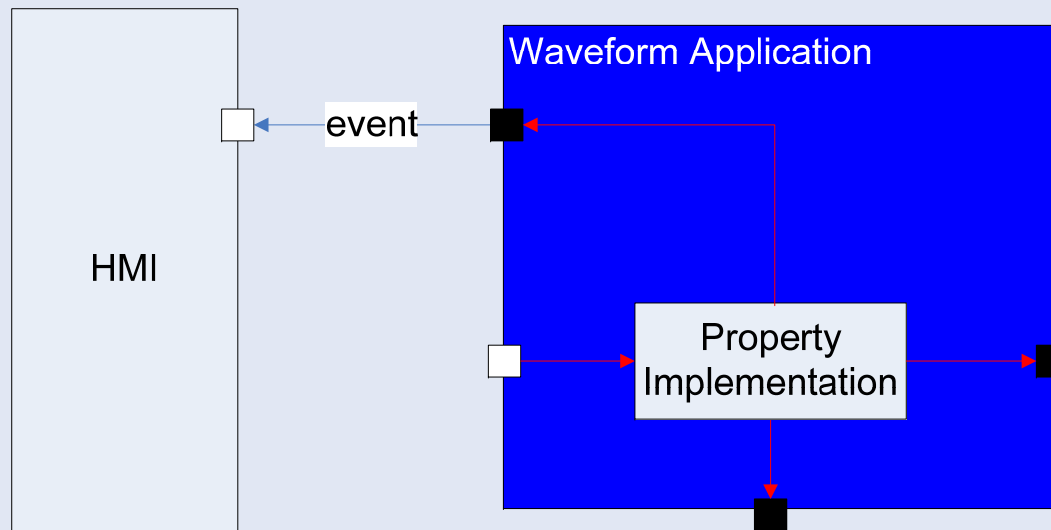
Design practical approaches

- The design pattern called "PROXY" is used at the property level to publish some platform or application attributes to external applications

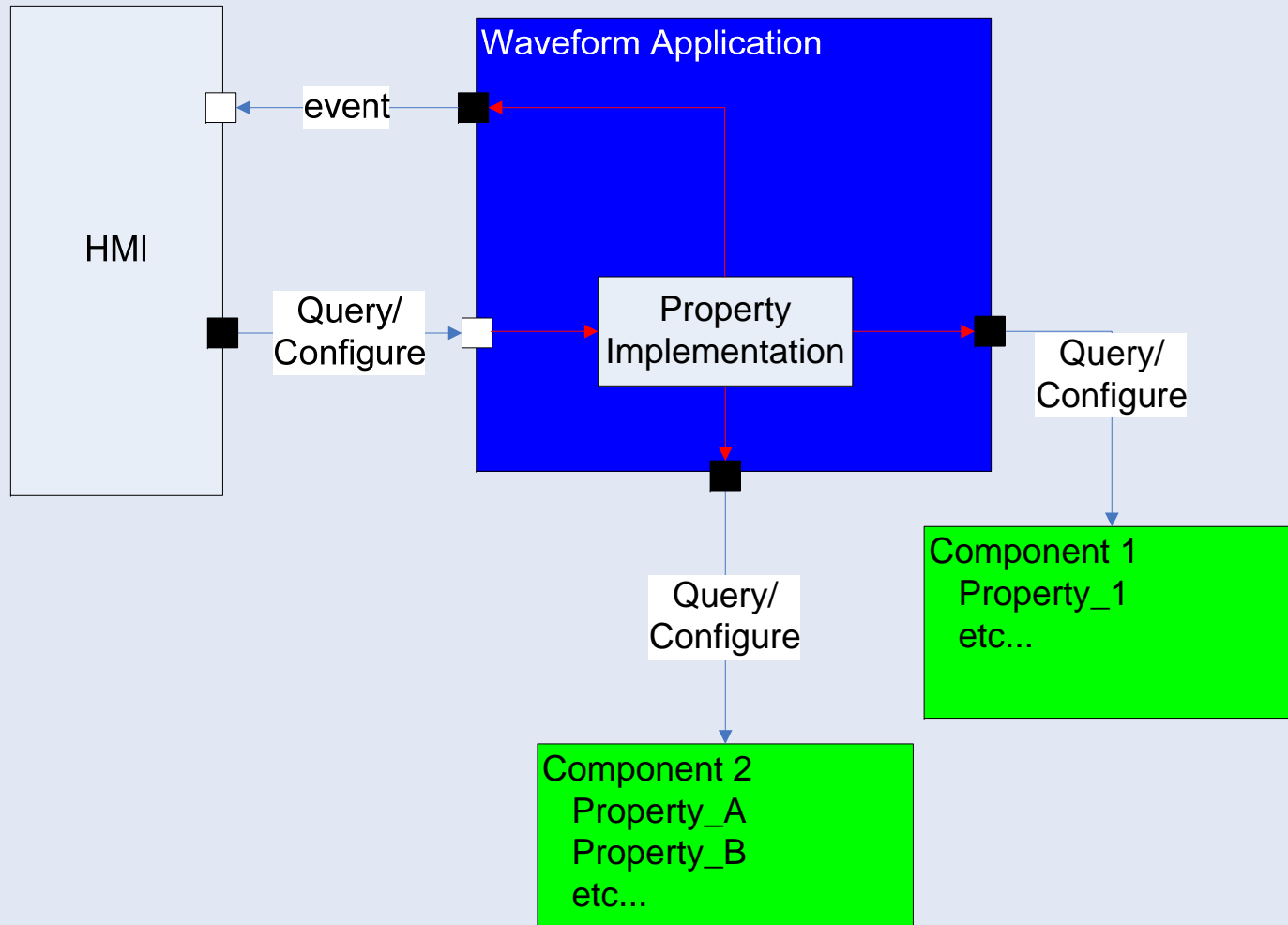


Design practical approaches

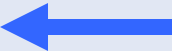
- Each public property of the application should generate an event with its name and value each time the property is modified.



Overall approach



Agenda

- Introduction
- Relevant SCA APIs for HMI development
- Proposed design patterns and best practices
- Virtual Front Panel project 
- Conclusion

Virtual Front Panel Project

- Allows remote control of a radio unit (AN/GRC-245 HCLOS) using the same interface
- Virtual Panel can be used in combination with the physical front panel
- Virtual Panel is hosted on a computer and behaves exactly like the physical front panel



Virtual Front Panel Project (cont'd)

SCA Architecture enables a quick development approach

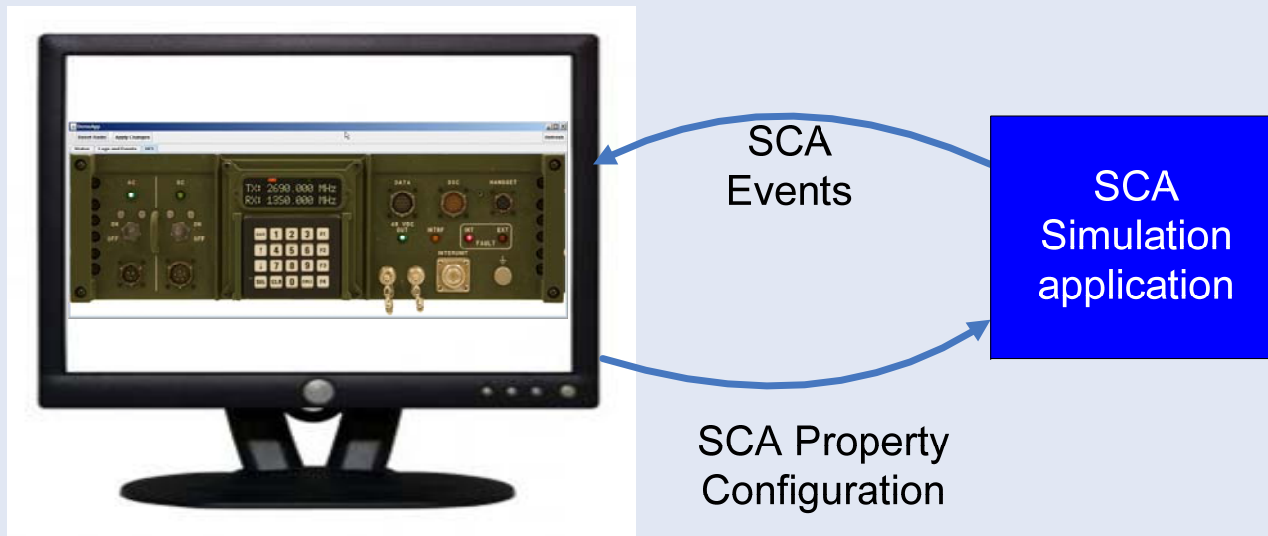
- The project was an R&D exploration project.
 - The time allocated was only one month
 - The main goal was to create a remote MMI
 - There was very little time for integration
 - There was only two developers on the project
 - One developer at CRC: GUI portion
 - One developer at Ultra: Radio portion

The Use of Standard API reduces the time of integration and facilitates information transfer

Virtual Front Panel Project (cont'd)

Development Approach

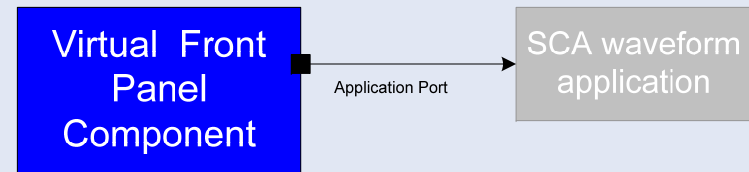
- Needed a simple and efficient way to interface between:
 - The two teams → Specification
 - The two portion of the system → SCA standard APIs
- Needed to be able to quickly build a simulation environment
 - CRC needed to simulate the radio MMI hardware



Virtual Front Panel Project (cont'd)

Architectural Approach

- What was already available in the existent architecture:
 - Information required by the Front Panel was already available via the SCA waveform Application
 - This information was available via a list of SCA properties and can be accessed via SCA standard API PropertySet

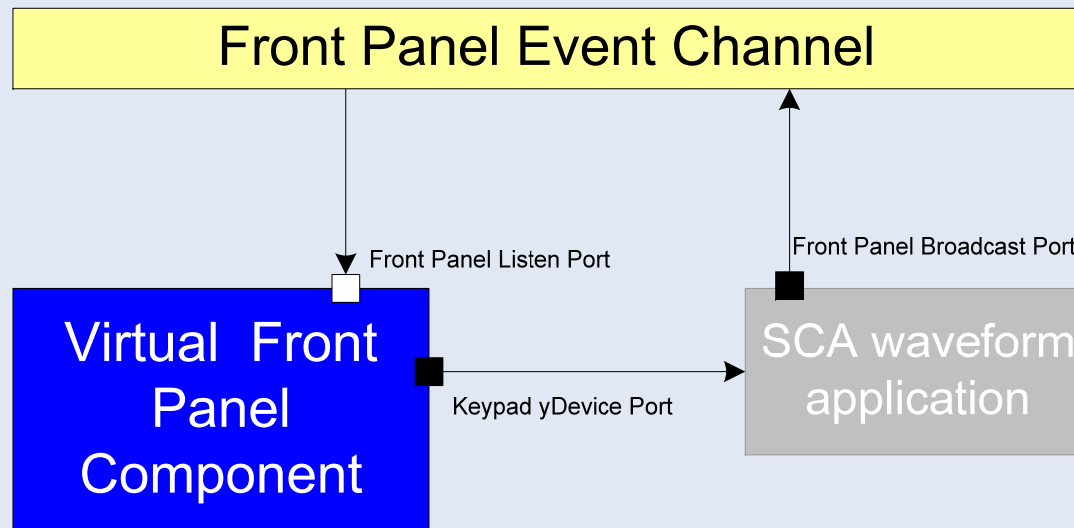


The information is already available in the actual design

Virtual Front Panel Project (cont'd)

Architectural Approach

- What was missing ?
 - An event channel, in order to refresh the Virtual Panel
 - On each relevant property modification, the generation of an event has been added.



Using a Standard Event Channel to notify the Virtual Panel reduces the use of processing resources that can be made available for more useful features

Virtual Front Panel Project (cont'd)

Interface Specification

- Ultra only had to prepare a very short specification document to describe what needed to be done by CRC.
- The outline of the specification was as follows:
 - SCOPE
 - Briefly summarized the project scope and context
 - PHYSICAL FRONT PANEL
 - Described all front panel zones which are animated
 - Describe the MMI capabilities and sub-components
 - APPLICATION PROGRAMMING INTERFACES (APIs)
 - Front Panel Event Channel section
 - SCA Application port section

Virtual Front Panel Project (cont'd)

Application Programming Interfaces (APIs)

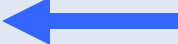
- Information that needs to be displayed will be sent on an Event Channel
 - Describes the format of the different events generated
- Information regarding SCA properties available on the SCA Waveform Application
 - Provides the SCA standard XML description of the properties. This accelerated the development of simulation components by CRC
 - Describes the possible values for each property
 - Describes the behavior of the virtual panel for each property value

Virtual Front Panel Project (cont'd)

From the Management Point of View

- Development of the Virtual Panel done in about 3 weeks.
- Development was done separately.
- No source code was exchanged during development.
- Time of integration was less than a day

Agenda

- Introduction
- Relevant SCA APIs for HMI development
- Proposed design patterns and best practices
- Virtual Front Panel project
- Conclusion 

Conclusion

- PropertySet and Event Channel are two SCA Standard API adapted to interface with HMI
- Three design patterns and best practices can speed-up HMI development when applied to Radio SCA Design
 - Property proxy on sub-component information
 - Assembly Controller used as a gateway for application information
 - Event broadcast on Assembly Controller property modification
- SCA standard APIs speed-up development time and reduce integration time

Virtual Front Panel Demonstration at Exhibit (CRC boot)



Questions

