MDCF Tutorial
Device Interface and App Development

http://mdcf.santos.cis.ksu.edu/

Acknowledgements:
Funding provided by US National Science Foundation awards 0734204, 0930647
Clinical documentation and hardware provided by CIMIT MD PnP, DocBox, Inc., Cerner, Inc.
Device Interface Development

**Current Workflow**
- Specify device interface
  - Currently limited to input/output event ports
  - Auto-generate .xml descriptors of component interfaces
- Auto-generate device code template
- Complete coding
  - Mock device: fill in code to simulate data streams or device control
  - Real device: use host’s communication ports to communicate with real device
- Install device signature
  - Vision:
    - Signatures installed through Maintenance console with authentication, or
    - Signature communicated dynamically during “plug-n-play” step with authentication
  - Current:
    - Signature is placed in a designated directory that MDCF Network Controller will load at start up

**Vision:** IDE for Driver Development & Validation

- Automatic code generation
- Core “business logic” written by driver developer to either simulate device or access device over host’s communication ports
- Connection protocol & channel set up code auto-generated
- Device Interface (ICE Device Model)
**Current Workflow**

- Break app functionality into components and specify interfaces of each component
- Auto-generate .xml descriptors of component interfaces
- Auto-generate code templates for component implementations
- Implement logic of components
- Wire component interfaces together to create *app configuration*
- Auto-generate .xml description of app configuration
- Create app archive (.jar) and install
App Archive

- Install app archive
  - Vision:
  - App archive installed through Maintenance console with authentication, or
  - App archive communicated dynamically during “plug-n-play” step with authentication
- Current:
  - App archive is placed in a designated directory that MDCF Network Controller will load at start up

Archive (.jar)

XML-based configuration information

- <CONFIGURATION_PASS>
- <HOME>
- <COMPONENT>
- <ID> <…></ID>
- <EVENT_SUPPLIER>
- <…events this component supplies…>
- </EVENT_SUPPLIER>
- </COMPONENT>
- </HOME>
- </CONFIGURATION_PASS>

archive generation produces .jar containing...

- .cfg.xml

- XML for component signatures used in app
- Java implementations of (non-device) components used in app
Tick Tock Example

BUILDING SIMPLE APP
TickTock Example

- TickTockEmitter (Device Component)
- TickTockInverter (Logic Component)
- TickTockDisplay (App Panel Component)
Designing Logic and App Panel component

APP DEVELOPMENT I
App configuration

APP DEVELOPMENT II
Building App Archive

APP DEVELOPMENT III
APPENDIX
Roles of Components

- **Device Component**
  - Represents an interface to a real medical device or which encapsulates a mock (simulated) medical device

- **App Panel Component**
  - Provides the control logic necessary for supporting a UI that will appear on the MDCF Supervisor Console

- **Logic Component**
  - Represents software processing elements inside the app such as device data transformers, control logic, smart alarm decision components, etc