Bindings in the Real World

Realizing Use Cases in AGL

Scott Murray and Matt Porter
scott.murray@konsulko.com, mporter@konsulko.com

About Konsulko Group

- Konsulko Group is a services company founded by embedded Linux veterans
- We do community and commercial embedded, Linux, and Open Source Software development
- Involved in AGL since 2015
- Have been working on build system, application framework bindings, and demo application improvements for AGL since Fall 2016
- See www.konsulko.com for more information

Konsulko Group

AGL Binding Review

Binding Overview

- The AGL application framework provides an API binding mechanism to allow abstracting an application's UI from its back end logic
- This allows re-using application logic with different UI implementations (e.g. Qt and HTML5)
- Allows the application framework to control access to APIs and resources in a fine-grained manner, effectively sandboxing applications based on their permissions.
- End goal is to provide a complete and consistent API for AGL applications
- More information
  - http://docs.automotivelinux.org/docs/apis_services/en/dev/reference/af-binder/afb-overview.html

Konsulko Group

Binding v2 Registration

- Binding implementation is a shared library
- A binding implementation:
  - Registers a unique binding ***api*** name
  - Registers a list of binding verbs to perform actions
  - Contains the binding verb/event backend implementation
  - Optionally registers ***preinit*** and ***init*** routines for the binding
  - Optionally registers a ***specification*** containing an OpenAPI v3 description of the binding
  - Optionally registers a text ***info*** description of the binding
  - Optionally registers an ***onevent*** callback for handling subscribed events
  - Optionally set ***noconcurrency*** flag to avoid concurrent verb calls

Konsulko Group

Application and Binding Initialization

- Application and binding packaging (widget) includes a config.xml file that:
  - Specifies the name, description, author, license
  - Lists any permissions that the package requires
  - Lists any bindings that the package requires
  - Lists any bindings that the package provides
- Application framework spawns an instance of afb-daemon
  - Loads and initializes the specified bindings
  - Executes the application, passing port number and authentication token arguments to it for binding access
  - Important to remember that each instance of the binding is separate
- More details
  - http://docs.automotivelinux.org/docs/apis_services/en/dev/reference/af-main/config.xml.html
  - http://docs.automotivelinux.org/docs/apis_services/en/dev/reference/af-binder/afb-bindings-overview.html

Konsulko Group

Application Binding Usage

- Submit requests in JSON format via HTTP or WebSocket
  - e.g. [2, "9999", "hvac/set", { "LeftTemperature" : 16}]
- Receive request status (success or failure) and any additional requested data
- Responses are also in JSON format
- Can subscribe / unsubscribe for events
- Events arrive asynchronously via WebSocket
- More details
  - http://docs.automotivelinux.org/docs/apis_services/en/dev/reference/af-binder/afb-bindings-writing.html

Konsulko Group

Best Practices

Fundamentals

- Avoid unnecessary build dependencies
  - Use cmake instead of qmake to avoid Qt dependencies
  - agl-service-radio is converted to cmake
- Implement as a standalone services and export APIs
  - Unless only ever used by one application
  - Radio binding previously was part of the radio demo app, now in agl-service-radio
  - Radio binding APIs exported by option in config.xml
- Enable standalone service bindings to start at boot
  - agl-service-bluetooth and agl-service-wifi are configured to start on boot
  - Allows connections to be automatically re-established
- Logically split bindings by functionality
  - Allows fine-grained loading of binding APIs when in use
  - E.g. telephony (ofono) functionality not a part of agl-service-bluetooth, but uses agl-service-bluetooth

Konsulko Group

Asynchronous operation

- Limit blocking synchronous APIs to fast operations
- Use events
- Use threads to manage lengthy operations
- e.g. Radio, Bluetooth, WiFi Scanning

Konsulko Group

JSON object lifecycle

- JSON-C has a reference counting object model which is an integral part of the application framework.
- Several core application framework APIs help by calling ***json_object_put()*** on the json object argument upon completion.
  - ***afb_req_*()*** [e.g. ***afb_req_success()***]
  - ***afb_event_push***
- Error paths should call ***json_object_put()*** on the object so it is freed
- https://gerrit.automotivelinux.org/gerrit/#/c/10687/1/binding/mediaplayer-api.c

Konsulko Group

AGL Binding APIs

New bindings by release

- Daring Dab
  - Radio Binding
  - Telephony Binding
  - MediaScanner Binding
  - Bluetooth / Settings Binding Enhancements
- Electric Eel
  - GPS Binding
  - GeoClue Binding
  - GeoFence Binding
  - MediaPlayer Binding

Konsulko Group

Radio Binding

- Features
  - Radio binding based on rtl-sdr SDR FM demodulation code previously used to build the QtMultimedia plugin from the Chinook release
  - Additional hooks added to FM demodulation code to add scanning support
  - Radio QML application reworked to use binding in place of QtMultimedia QRadio class
    - Only minimal changes were required, the QML interface for the binding emulates QRadio's interface to a large degree
    - Application enhanced to add scanning support
- Future Development
  - Additional tuner hardware support
  - Metadata support (e.g. RDS)
  - HD Radio support

Konsulko Group

Radio Binding

- Verbs
  - **radio/frequency** - get/set frequency
    - Input: { "value" : INTEGER }
      - Value is frequency in Hertz, e.g. 107100000
    - Input: None
    - Output: { "frequency" : INTEGER }
  - **radio/band** - get/set band
    - Input: { “band” : STRING “<band>” }
      - Band is “AM” or “FM”
    - Input: None
    - Output: { “band” : STRING ”<band>” }
  - **radio/band_supported** - check band support
    - Input: { “band” : STRING “<band>” }
    - Output: { “supported” : BOOLEAN }
      - Value is 0 or 1

Konsulko Group

Radio Binding

- Verbs (continued)
  - **radio/frequency_range** - get band frequency range
    - Input: { “band” : STRING “<band>” }
    - Output: { "min" : INTEGER, "max" : INTEGER }
      - Values are frequencies in Hertz
      - e.g. { "min" : 87900000, "max" : 107900000 }
  - **radio/frequency_step** - get band frequency step
    - Input: { “band” : STRING “<band>” }
    - Output: { "step" : INTEGER }
      - Value is band frequency step in Hertz, e.g. 200000
  - **radio/start**
    - Input: None
    - Output: None
  - **radio/stop**
    - Input: None
    - Output: None

Konsulko Group

Radio Binding

- Verbs (continued)
  - **radio/scan_start**
    - Input: { "direction" : "forward" }
    - Input: { "direction" : "backward" }
    - Output: None
  - **radio/scan_stop**
    - Input: None
    - Output: None
  - **radio/stereo_mode** - get/set stereo mode
    - Input: { “value” : ”mono”}
    - Input: { “value” : ”stereo”}
    - Input: None
    - Output: { “mode” : STRING “<mode>” }

Konsulko Group

Radio Binding

- Verbs (continued)
  - **radio/subscribe**
    - Input: { "value" : "frequency" }
    - Input: { "value" : "station_found" }
    - Output: None
  - **radio/unsubscribe**
    - Input: { "value" : "frequency" }
    - Input: { "value" : "station_found" }
    - Output: None
- Events
  - **radio/frequency** - frequency has changed
    - { "value" : INTEGER }
  - **radio/station_found** - scanning has found a station
    - { "value" : INTEGER }
      - Value is frequency of station

Konsulko Group

Code Walkthrough

Demo HMI radio application
use of Radio Binding

Telephony Binding

- Features
  - Bluetooth Hands-Free Profile (HFP) device support
  - Originate a voice call
  - Answer an incoming voice call
  - Provide status and information on voice call connections
- Future Development
  - Merge incomingCall, dialingCall, and terminatedCall events into the callStateChanged event
  - In-call sending of dial tones (for conference bridges, etc.)
  - Call waiting/hold/forwarding

Konsulko Group

Telephony Binding

- Verbs
  - **telephony/dial -** dial a phone number
    - Input: { “number” : STRING “<phone number>” }
    - Output: None
  - **telephony/hangup -** hangup an active phone call
    - Input: None
    - Output: None
  - **telephony/answer** - answer an incoming phone call
    - Input: None
    - Output: None
  - **telephony/subscribe** - subscribe to a telephony binding event
  - **telephony/unsubscribe** - unsubscribe to a telephony binding event
    - Input: { "value" : "callStateChanged"}
    - Input: { “value” : ”incomingCall” }
    - Input: { “value” : “dialingCall” }
    - Input: { “value” : “terminatedCall” }
    - Output: None

Konsulko Group

Telephony Binding

- Events
  - **telephony/callStateChanged** - state of a phone call has changed
    - { “state” : “active” } - call has been answered
    - { “state” : “held” } - call placed on hold
    - { “state” : “dialing” } - call is being dialed
    - { “state” : “alerting” } - call is alerting remote party (ringing remotely)
    - { “state” : “incoming” } - incoming call is ringing
    - { “state” : “waiting” } - incoming call is waiting due to active call
    - { "state" : “disconnected” } - call has been terminated
  - **telephony/incomingCall** - incoming call is ringing
    - { "clip" : ”<incoming CLIP information>” } - numeric phone number information
  - **telephony/dialingCall** - outgoing call is being dialed
    - { "colp" : ”<outgoing COLP information>” } - numeric phone number information
  - **telephony/terminatedCall** - call has been terminated
    - None

Konsulko Group

MediaScanner Binding

- Features
  - Media binding to report media insertion/removal
  - Media detection and path reporting
  - Receive metadata from Bluetooth binding

Konsulko Group

MediaScanner Binding

- Verbs
  - **mediascanner/media_result** - get all available multimedia
    - Input: None
    - Output: { “Media”: [STRING “<file>”, …] }
      - e.g.: { “Media”: [“/run/media/sda1/Track 1.ogg”, …] }
  - **mediascanner/subscribe** - subscribe to a media binding event
    - Input: { “value” : “media_added” }
    - Input: { “value” : “media_removed” }
    - Output: None
  - **mediascanner/unsubscribe** - unsubscribe from a media binding event
    - Input: { “value” : “media_added” }
    - Input: { “value” : “media_removed” }
    - Output: None

Konsulko Group

MediaScanner Binding

- Events
  - **mediascanner/media_added** - media is attached to the device
    - { “Path”: STRING “<path>”, “Media”: [STRING “<file>”, …] }
      - e.g.: { “Path” : “/run/media/sda1”, “Media”: [“/run/media/sda1/Track 1.ogg”, …] }
  - **mediascanner/media_removed** - media is removed from device
    - { “Path”: STRING “<path>” }

Konsulko Group

MediaPlayer Binding

- Features
  - Media binding based on gstreamer
  - Media audio playback and control
- Future
  - Video playback

Konsulko Group

MediaPlayer Binding

- Verbs
  - **mediaplayer/playlist** - get/set playlist
  - **mediaplayer/controls** - playback controls e.g. play, pause, etc.
  - **mediaplayer/metadata** - get metadata of current track
  - **mediaplayer/subscribe** - subscribe to mediaplayer events
  - **mediaplayer/unsubscribe** - unsubscribe to mediaplayer events
- Events
  - **mediaplayer/metadata** - position/duration of current track
  - **mediaplayer/playlist** - playlist changed

Konsulko Group

GPS Binding

- Features
  - GPS Binding based on gpsd protocol
  - Provides GNSS location data

Konsulko Group

GPS Binding

- Verbs
  - **gps/location** - Get GNSS data
    - Input: None
    - Output:
      - "altitude" : FLOAT (meters)
      - “latitude" : FLOAT (degrees)
      - “longitude” : FLOAT (degrees)
      - “speed” : FLOAT (meters per second)
      - “time” : STRING (timestamp string)
  - **gps/subscribe** - subscribe to a gps binding event
    - Input: { "value" : "location"}
    - Output: None
  - **gps/unsubscribe** - unsubscribe to a gps binding event
    - Input: { "value" : "location"}
    - Output: None

Konsulko Group

GPS Binding

- Events
  - **gps/location** - GNSS data updated
    - "altitude" : FLOAT (meters)
    - “latitude" : FLOAT (degrees)
    - “longitude” : FLOAT (degrees)
    - “speed” : FLOAT (meters per second)
    - “time” : STRING (timestamp string)

Konsulko Group

GeoClue Binding

- Features
  - Based on GeoClue location library
  - Supports gathering location data from multiple sources:
    - WiFi AP databases
    - 3g/4g 3GPP tower information
    - GeoIP database
    - GPS

Konsulko Group

GeoClue Binding

- Verbs
  - **geoclue/location** - Get GeoClue location data
    - Input: None
    - Output:
      - “accuracy” : FLOAT (meters)
      - "altitude" : FLOAT (meters)
      - “latitude" : FLOAT (degrees)
      - “longitude” : FLOAT (degrees)
      - “heading” : FLOAT (degrees)
      - “speed” : FLOAT (meters per second)
  - **geoclue/subscribe** - subscribe to a geoclue binding event
    - Input: { "value" : "location"}
    - Output: None
  - **geoclue/unsubscribe** - unsubscribe to a geoclue binding event
    - Input: { "value" : "location"}
    - Output: None

Konsulko Group

GeoClue Binding

- Events
  - **geoclue/location** - GeoClue location data updated
    - “accuracy” : FLOAT (meters)
    - "altitude" : FLOAT (meters)
    - “latitude" : FLOAT (degrees)
    - “longitude” : FLOAT (degrees)
    - “heading” : FLOAT (degrees)
    - “speed” : FLOAT (meters per second)

Konsulko Group

GeoFence Binding

- Features
  - Add/remove/list geographic bounding boxes
  - Generates enter/leave events when ingress/egress occurs in fenced bounding box
  - Leverages GPS binding for location data
- Future
  - Add support for GeoClue binding location data
  - Add support for per-fence dwell transition timing
  - Make “dwell”, “entered”, and “exited” separately subscribed events

Konsulko Group

GeoFence Binding

- Verbs
  - **geofence/add_fence** - add a geofence bounding box
    - Input:
      - “name” : STRING
      - “bbox”
        - “min_latitude” : FLOAT (degrees)
        - “max_latitude” : FLOAT (degrees)
        - “min_longitude” : FLOAT (degrees)
        - “max_longitude” : FLOAT (degrees)
    - Output: None
  - **geofence/remove_fence** - remove a geofence bounding box
    - Input:
      - “name” : STRING
    - Output: None

Konsulko Group

GeoFence Binding

- Verbs
  - **geofence/list_fences** - list all geofence bounding boxes
    - Input: None
    - Output: ARRAY
      - “name” : STRING
      - “bbox”
        - “min_latitude” : FLOAT (degrees)
        - “max_latitude” : FLOAT (degrees)
        - “min_longitude” : FLOAT (degrees)
        - “max_longitude” : FLOAT (degrees)
      - “within” : BOOLEAN
      - “dwell” : BOOLEAN
    - Output: None
  - **geofence/dwell_transition** - get/set dwell transition time
    - Input:
      - “value” : INTEGER[OPTIONAL] (seconds)
    - Output:
      - “seconds” : INTEGER (seconds)

Konsulko Group

GeoFence Binding

- Verbs (continued)
  - **geofence/subscribe** - subscribe to a geofence binding event
    - Input: { "value" : "fence"}
    - Output: None
  - **geofence/unsubscribe** - unsubscribe to a geofence binding event
    - Input: { "value" : "fence"}
    - Output: None
- Events
  - **geofence/fence** - geofence event occurred
    - “name” : STRING (name of fence generating the event)
    - "state" : STRING (“dwell” OR “entered” OR “exited”)

Konsulko Group

Settings / Bluetooth Binding Enhancements

- New Features
  - AVRCP Bluetooth binding controls
  - Media metadata, and position tracking
  - Device priority list

Konsulko Group

Settings / Bluetooth Binding

- Verbs
  - **Bluetooth-Manager/device_priorities** - list bluetooth devices in priority order
    - Input: None
    - Output: ARRAY { STRING } (bluetooth device addresses)
- Events
  - **Bluetooth-Manager/device_updated** - additional dictionary of metadata added
    - { “Metadata” : { “Title” : STRING “<title>”, “Artist” : STRING “<artist>”, “Duration” : INTEGER, “Position” : INTEGER }

Konsulko Group

Roadmap

- Bluetooth PBAP support
  - Integration with telephony API
- Completion of MediaPlayer binding
- Video support for MediaPlayer binding
- Speech recognition / TTS binding
- WWAN modem support

Konsulko Group

Feedback

- Please suggest changes to enable your production use cases!
- Feedback channels:
  - IRC: #automotive on Freenode.net
  - Mailing list: https://lists.linuxfoundation.org/mailman/listinfo/automotive-discussions
  - Weekly developer call: https://wiki.automotivelinux.org/dev-call-info
  - JIRA: https://jira.automotivelinux.org

Konsulko Group

Resources

- Application Framework training session
  - 13:00-17:00 Friday - Ronan Le Martret, IoT.bzh
- Source git repositories
  - bluetooth : git clone http://gerrit.automotivelinux.org/gerrit/apps/agl-service-bluetooth
  - geoclue : git clone http://gerrit.automotivelinux.org/gerrit/apps/agl-service-geoclue
  - geofence : git clone http://gerrit.automotivelinux.org/gerrit/apps/agl-service-geofence
  - gps : git clone http://gerrit.automotivelinux.org/gerrit/apps/agl-service-gps
  - mediaplayer : git clone http://gerrit.automotivelinux.org/gerrit/apps/agl-service-mediaplayer
  - mediascanner : git clone http://gerrit.automotivelinux.org/gerrit/apps/agl-service-mediascanner
  - radio : git clone http://gerrit.automotivelinux.org/gerrit/apps/agl-service-radio
  - telephony : git clone http://gerrit.automotivelinux.org/gerrit/apps/phone
- Binding Documentation
  - http://docs.automotivelinux.org/docs/apis_services/en/dev/
- Wiki
  - https://wiki.automotivelinux.org/start

Konsulko Group

Questions?

Glossary

- API Application Programming Interface
- AVRCP Audio/Video Remote Control Profile
- CLIP Calling Line Identification Presentation
- CLIR Calling Line Identification Restriction
- COLP Connected Line Identification Presentation
- GNSS Global Navigation Satellite System
- GPS Global Positioning System
- HFP Hands-Free Profile
- JSON JavaScript Object Notation
- RDS Radio Descriptive Service
- SDR Software Defined Radio
- SIM Subscriber Identification Module

Konsulko Group