



DEVELOPER  
**PROGRAM**

**SDL ECOSYSTEM**  
SDL DEVELOPER CONFERENCE  
MWC AMERICAS 2017



Go Further

# AGENDA

---

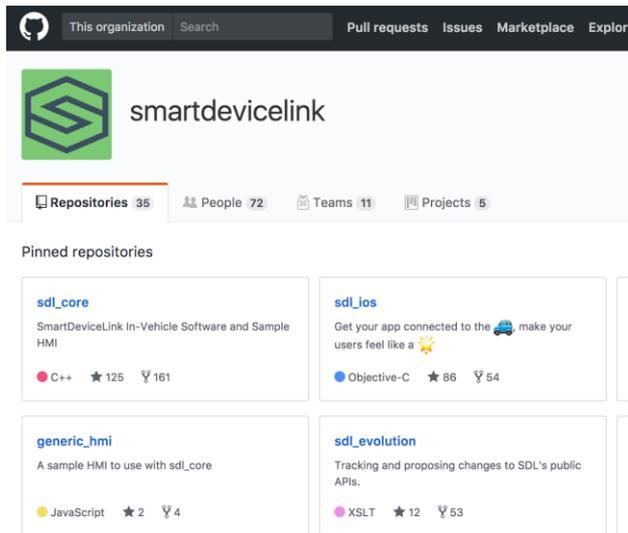


- Introduction
- Developer Tools
  - iOS
  - Android
  - Core
- Documentation
- [SmartDeviceLink.com](https://smartdevicelink.com)
- AppIDs
- SDL Evolution
- Contribution

# INTRODUCTION/GIT, GITHUB & GIT-FLOW

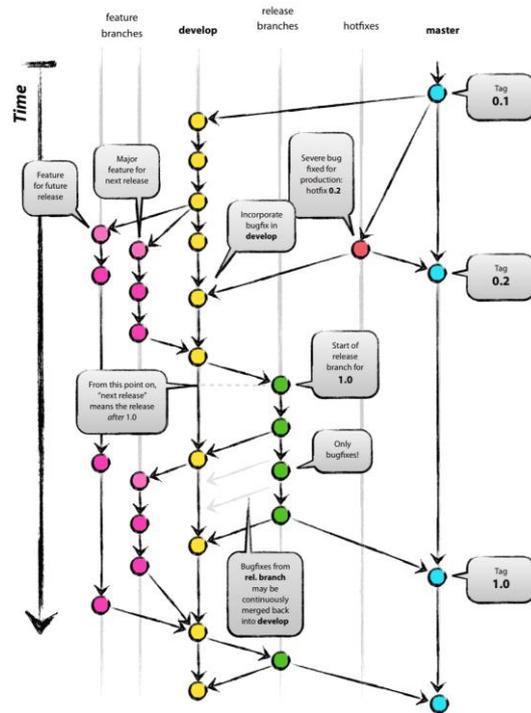
## The SDLC uses:

- git for version control
- github for hosting the project
- git-flow as the branching strategy



The screenshot shows the GitHub organization page for 'smartdevicelink'. At the top, there are navigation links for 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. Below the organization name, there are statistics: 'Repositories 35', 'People 72', 'Teams 11', and 'Projects 5'. Under 'Pinned repositories', four repositories are listed:

- sdl\_core**: SmartDeviceLink In-Vehicle Software and Sample HMI. C++ language, 125 stars, 161 forks.
- sdl\_ios**: Get your app connected to the car, make your users feel like a 🌟. Objective-C language, 86 stars, 54 forks.
- generic\_hmi**: A sample HMI to use with sdl\_core. JavaScript language, 2 stars, 4 forks.
- sdl\_evolution**: Tracking and proposing changes to SDL's public APIs. XSLT language, 12 stars, 53 forks.



More information about git-flow:

<http://nvie.com/posts/a-successful-git-branching-model/> (image source)

<https://www.atlassian.com/git/tutorials/comparing-workflows#gitflow-workflow>

# INTRODUCTION/SEMANTIC VERSIONING

---

**The SDLC uses semantic versioning for all projects.**

Version Scheme: X.Y.Z

- X: MAJOR version for incompatible API changes
- Y: MINOR version for adding functionality in a backwards-compatible manner
- Z: PATCH version for backwards-compatible bug fixes

<http://semver.org/>

# DEVELOPER TOOLS/IOS-VERSION

## Current Version

4.7.0

## Next planned release:

5.0.0

## Highlights:

Removal of developer access to the SDLProxy object.

The new SDLManager will replace the direct access and allow the SDLC team to overhaul large parts of the project, without constant braking changes.

[https://github.com/smartdevicelink/sdl\\_evolution/blob/master/proposals/0019-ios-proxy-layer-nonpublic.md](https://github.com/smartdevicelink/sdl_evolution/blob/master/proposals/0019-ios-proxy-layer-nonpublic.md)

The screenshot displays a Jira backlog for version v5.0.0. It is organized into three columns: Features Backlog (1 item), Blocked (2 items), and Bugs Backlog (11 items). A search bar labeled 'Filter cards' is located in the top right corner.

- Features Backlog (1):**
  - Item: Add Swift example app and cleanup existing sample app. #622 opened by brettywhite. Labels: enhancement, example app.
- Blocked (2):**
  - Item: [SDL 0081] Automatic Haptic Interface Detection for UIKit based VPM / Nav applications. #698 opened by theresalech. Label: proposal.
  - Item: [SDL\_0054] ChangeRegistration-Manager. #617 opened by theresalech. Label: proposal.
- Bugs Backlog (11):**
  - Item: `EAAccessoryDic ation` often not f background reco beta. #718 opened by Tnr. Label: bug.
  - Item: Fix some RPC prc incorrectly retrie. #713 opened by joe. Label: bug.

# DEVELOPER TOOLS/IOS-AVAILABILITY

---

The iOS Library may be integrated via different services.



CocoaPods



Carthage



Source Code

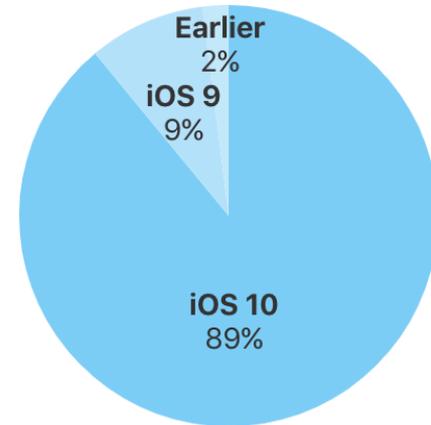
# DEVELOPER TOOLS/IOS-COMPATIBILITY

---

Compatibility is a serious problem. It balances the wish and need to use new operating system features and programming language options with the desire of the App Partners to be compatible with as many users as possible.

**Current minimum iOS Version: 7.0**  
**5.0.0 will require iOS 8.0**

89% of devices are using iOS 10.



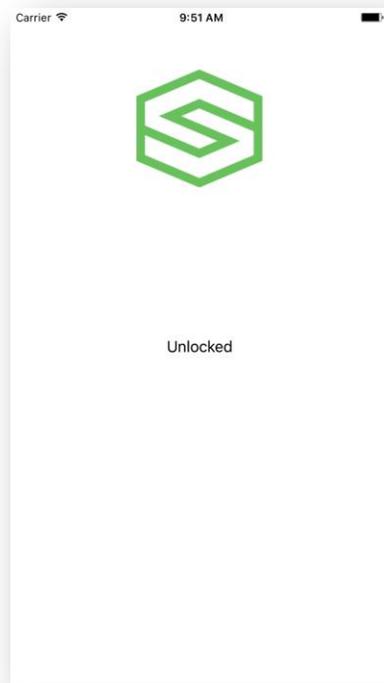
As measured by the App Store on  
September 6, 2017.

\*More information: [https://github.com/smartdevicelink/sdl\\_evolution/blob/master/proposals/0024-ios-8-0-minimum.md](https://github.com/smartdevicelink/sdl_evolution/blob/master/proposals/0024-ios-8-0-minimum.md)  
Image Source: <https://developer.apple.com/support/app-store/>

Basic example of how to use SDL

Will be updated with the latest changes  
for each library release.

[https://github.com/smartdevicelink/sdl\\_ios/tree/master/SmartDeviceLink\\_Example](https://github.com/smartdevicelink/sdl_ios/tree/master/SmartDeviceLink_Example)



# DEVELOPER TOOLS/IOS-RPC\_BUILDER\_APP

---

The SmartDeviceLink RPC Builder is designed to allow free testing of the SDL Interface during development.

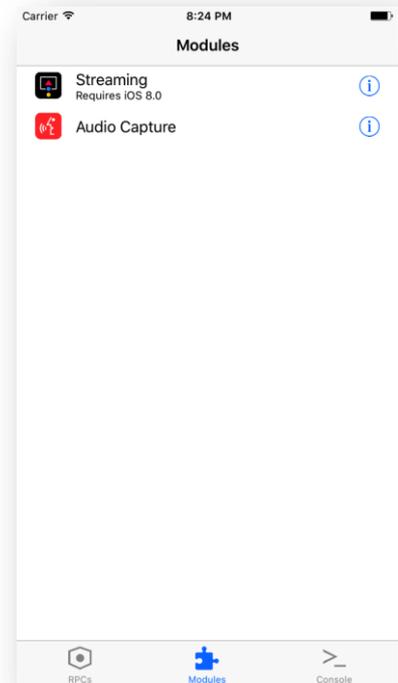
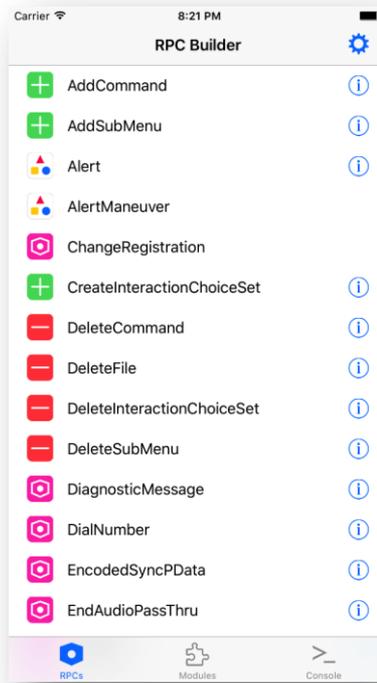
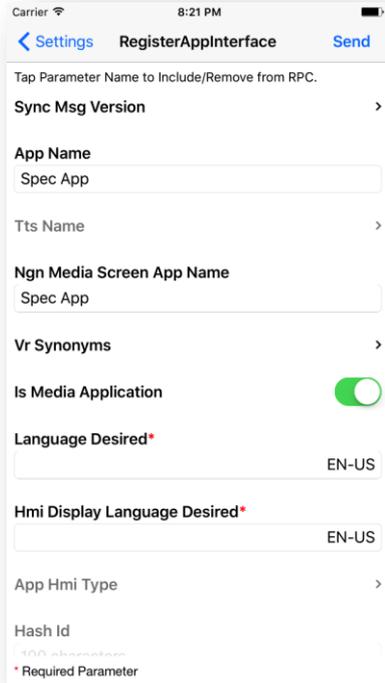
## Features:

- Free usage of any rpc specification
- Dynamic creation of UI to send rpcs
- "Plug in" functionality for advanced test cases
- Open source\*



\*More information: [https://github.com/smartdevicelink/rpc\\_builder\\_app\\_ios](https://github.com/smartdevicelink/rpc_builder_app_ios)

# DEVELOPER TOOLS/IOS-RPC\_BUILDER\_APP

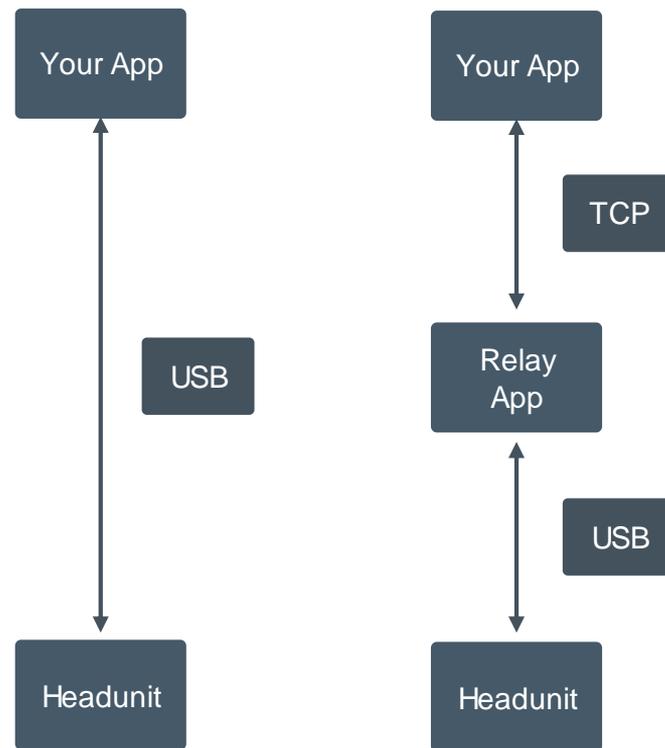


# DEVELOPER TOOLS/IOS-RELAY\_APP

SDL iOS Relay is a tool built for developers building applications that utilizes SmartDeviceLink for connecting their apps to technologies using our [SmartDeviceLink Core](#) connected via USB.

Features:

- Allows to connect to a USB based SDL core via TCP/IP
- Open source\*



\*More information: [https://github.com/smartdevicelink/relay\\_app\\_ios](https://github.com/smartdevicelink/relay_app_ios)

# DEVELOPER TOOLS/IOS-RELAY\_APP

**USB Connection**  
Status: **Disconnected**

**EASession**  
Status: **Closed**

**Server**  
  
Status: **Not Started**

Address:  
Port:

**TCP/IP**  
Status: **Disconnected**

**USB Connection**  
Status: **Connected**

**EASession**  
Status: **Connected**

**Server**  
  
Status: **Available**

Address: 1.2.3.4  
Port: 2776

**TCP/IP**  
Status: **Disconnected**

**USB Connection**  
Status: **Connected**

**EASession**  
Status: **Connected**

**Server**  
  
Status: **Available**

Address: 1.2.3.4  
Port: 2776

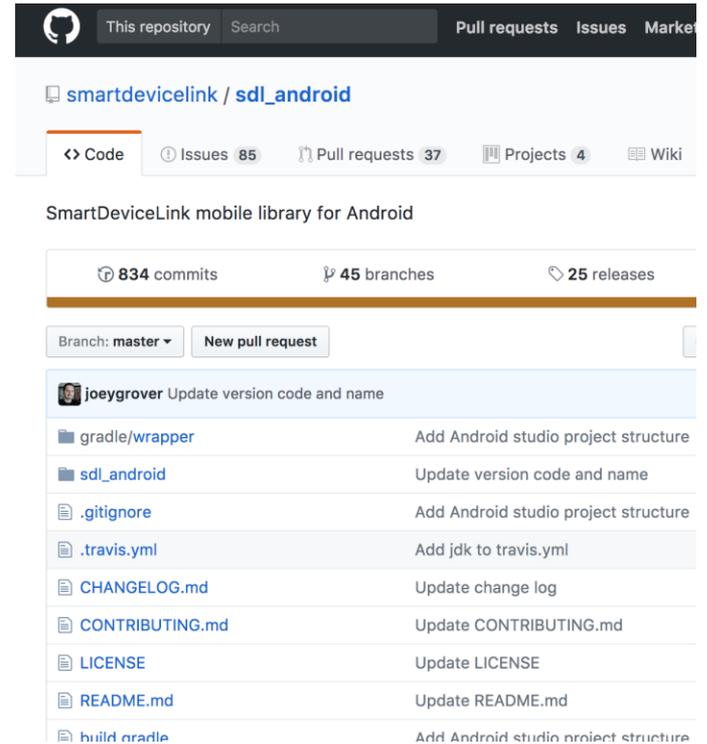
**TCP/IP**  
Status: **Connected**

**Current Version**  
**4.3.1**

**Next planned release:**  
**4.4.0**

## Highlights:

Addition of SmartDeviceLink to Bintray.  
Improvements to Video Streaming API's  
improved unit tests. Android O compatibility  
fixes.



This repository Search Pull requests Issues Market

smartdevicelink / sdl\_android

<> Code Issues 85 Pull requests 37 Projects 4 Wiki

SmartDeviceLink mobile library for Android

834 commits 45 branches 25 releases

Branch: master New pull request

joeygrover Update version code and name

gradle/wrapper	Add Android studio project structure
sdl_android	Update version code and name
.gitignore	Add Android studio project structure
.travis.yml	Add jdk to travis.yml
CHANGELOG.md	Update change log
CONTRIBUTING.md	Update CONTRIBUTING.md
LICENSE	Update LICENSE
README.md	Update README.md
build.gradle	Add Android studio project structure

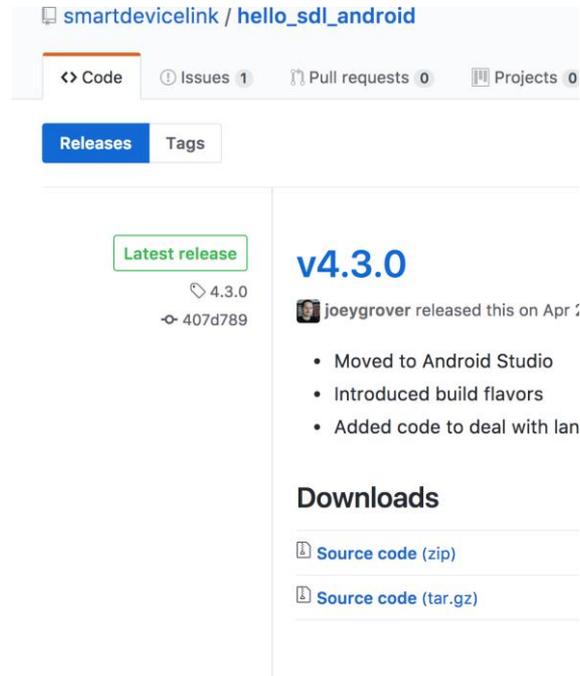
\*More information: [https://github.com/smartdevicelink/sdl\\_android](https://github.com/smartdevicelink/sdl_android)

# DEVELOPER TOOLS/ANDROID-HELLO\_WORLD

Basic example of how to use SDL

Will be updated with the latest changes for each library release.

[https://github.com/smartdevicelink/hello\\_sdl\\_android](https://github.com/smartdevicelink/hello_sdl_android)



The screenshot shows the GitHub interface for the repository smartdevicelink/hello\_sdl\_android. At the top, there are navigation tabs for Code, Issues (1), Pull requests (0), and Projects (0). Below these are tabs for Releases and Tags. The Releases tab is active, displaying the latest release, v4.3.0, which is highlighted with a green 'Latest release' badge. The release was made by user joeygrover on April 1st, 2024, with commit hash 407d789. A list of changes for this release includes: 'Moved to Android Studio', 'Introduced build flavors', and 'Added code to deal with lan'. Below the release information, there is a 'Downloads' section with two links: 'Source code (zip)' and 'Source code (tar.gz)'.

# DEVELOPER TOOLS/ANDROID-COMPATIBILITY

Compatibility is a serious problem. It balances the wish and need to use new operating system features and programming language options with the desire of the App Partners to be compatible with as many users as possible.

**Current minimum Android Version 2.2**

**Current Android Target Version: 4.4**

More info about Minimum vs. Target:

<https://medium.com/google-developers/picking-your-compile-sdk-version-min-sdk-version-target-sdk-version-a098a0341ebd>

Version	Codename	API	Distribution
2.3.3 - 2.3.7	Gingerbread	10	0.7%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	0.7%
4.1.x	Jelly Bean	16	2.7%
4.2.x		17	3.8%
4.3		18	1.1%
4.4	KitKat	19	16.0%
5.0	Lollipop	21	7.4%
5.1		22	21.8%
6.0	Marshmallow	23	32.3%
7.0	Nougat	24	12.3%
7.1		25	1.2%

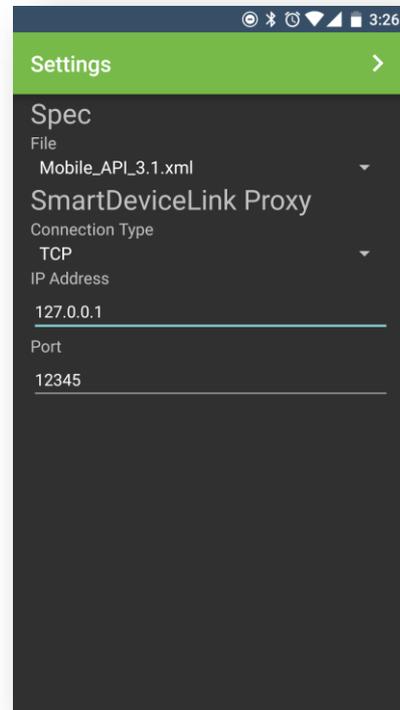
*Data collected during a 7-day period ending on August 8, 2017.  
Any versions with less than 0.1% distribution are not shown.*

# DEVELOPER TOOLS/ANDROID-RPC\_BUILDER\_APP

The SmartDeviceLink RPC Builder is designed to allow free testing of the SDL Interface during development.

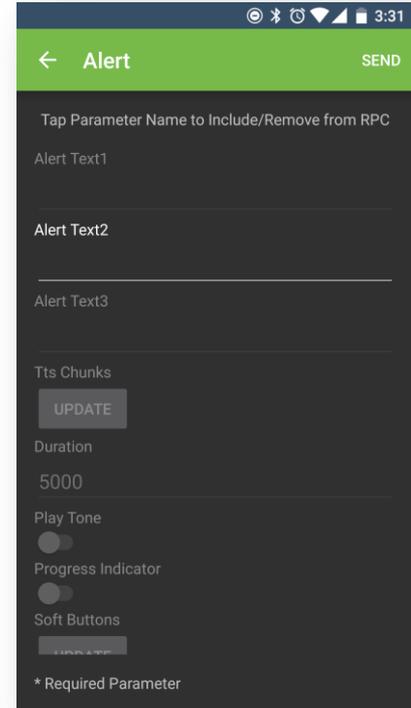
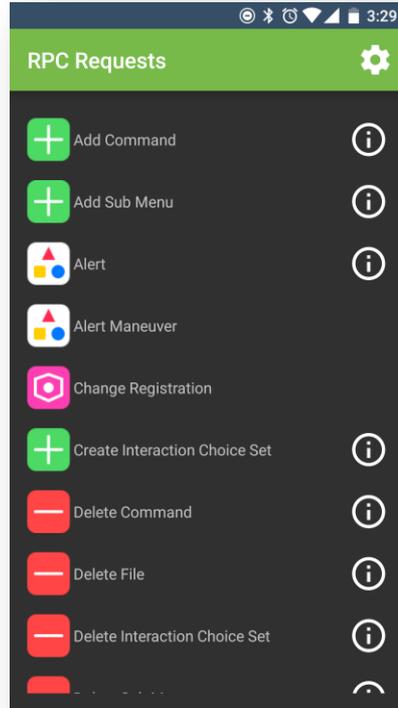
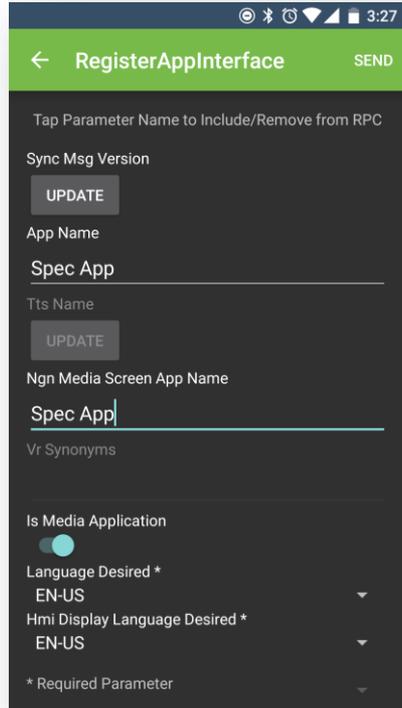
## Features:

- Free usage of any rpc specification
- Dynamic creation of UI to send rpcs
- Open source\*

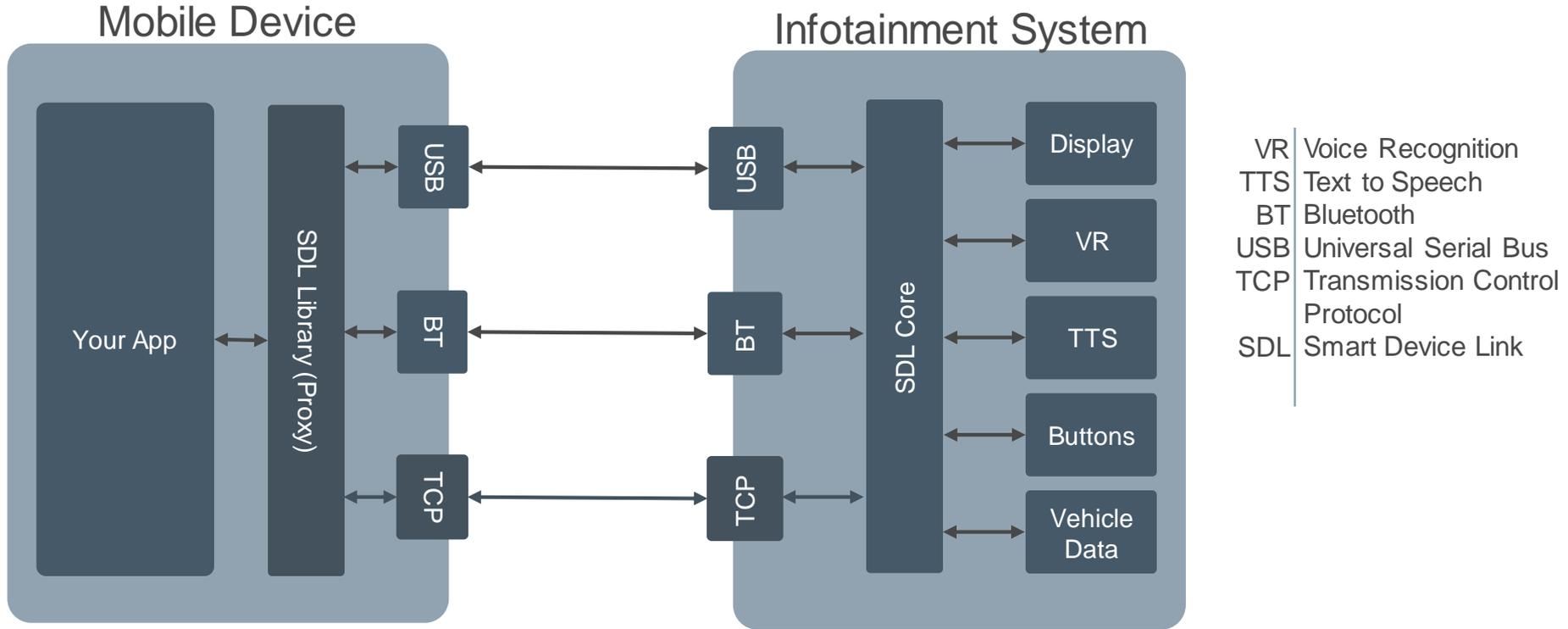


\*More information: [https://github.com/smartdevicelink/rpc\\_builder\\_app\\_android](https://github.com/smartdevicelink/rpc_builder_app_android)

# DEVELOPER TOOLS/ANDROID-RPC\_BUILDER\_APP



# DEVELOPER TOOLS/CORE



# DEVELOPER TOOLS/CORE

- C++ based core library
- POSIX compliant
- Currently supported platforms:
  - QNX
  - Ubuntu
- Currently not officially supported
  - Android
- Connection to embedded systems via web socket
  
- WEB HMI
  - HTML5/JS based
  - Out of date



## Dependencies list

Flag	Description	Dependencies
Web HMI	Use HTML5 HMI	chromium-browser
HMI2	Build with QT HMI	QT5, dbus-*dev
EXTENDED_MEDIA_MODE	Support Video and Audio Streaming	Opengl es2, gstreamer1.0*
Bluetooth	Enable bluetooth transport adapter	libbluetooth3, libbluetooth-dev, bluez-tools
Testing framework	Needed to support running unit tests	libgtest-dev
Cmake	Needed to configure SDL prior to compilation	cmake

# DEVELOPER TOOLS/HARDWARE VALIDATOR APP

---

- Semi-automated tests, prompting the user whenever manual input is necessary
- The ability to read in scripts containing event sequences describing individual tests, to allow for easily expandable test cases
- Must be able to create test suites within these scripts composed of several of these test sequences
- Sequences can include messages to be sent, expected messages from SDL Core, and user prompts
- Logging of all messages sent and received from the app, for simpler debugging
- File logging for all test results, to be presented as validation criteria for a new integration

[https://github.com/smartdevicelink/sdl\\_evolution/blob/master/proposals/0056-hardware-validator-app-android.md](https://github.com/smartdevicelink/sdl_evolution/blob/master/proposals/0056-hardware-validator-app-android.md)

# DEVELOPER TOOLS/DOCUMENTATION-GUIDES

---

Guides are small tutorials that explain the most important SmartDeviceLink functionality.

Guides are available for the most important functionality. For Android and iOS

<http://smartdevicelink.com>

sdl Learn Documentation Resources SDLC

## Guides iOS

← Guides

Filter...

- Getting Started
- Installation
- SDK Configuration
- Integration Basics
- Connecting to a SDL Core
- Displaying Information
- Methods, Filters and Objects

### Installation

In order to build your app on a SmartDeviceLink (SDL) Core, the SDL software development kit (SDK) must be installed in your app. The following steps will guide you through adding the SDL SDK to your workspace and configuring the environment.

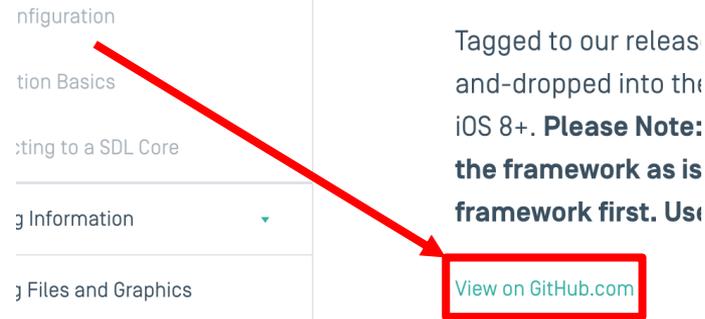
#### Install SDL SDK

# DEVELOPER TOOLS/DOCUMENTATION-GUIDES

Guides are hosted on github  
Dynamic creation of pages for  
smartdevicelink.com

Guide is broken, or inaccurate?  
Open an issue on github and let us  
know! You can find the link to the  
repository on the bottom of each  
guide.

[https://github.com/smartdevicelink/sdl\\_ios\\_guides](https://github.com/smartdevicelink/sdl_ios_guides)  
[https://github.com/smartdevicelink/sdl\\_android\\_guides](https://github.com/smartdevicelink/sdl_android_guides)



Configuration

Installation Basics

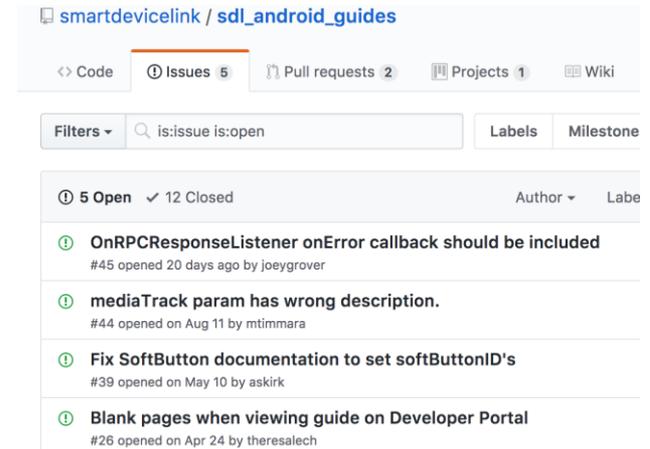
Configuring to a SDL Core

Getting Information

Getting Files and Graphics

Tagged to our releases and-dropped into the iOS 8+. **Please Note: the framework as is framework first. Use**

[View on GitHub.com](#)



smartdevicelink / sdl\_android\_guides

<> Code ① Issues 5 Pull requests 2 Projects 1 Wiki

Filters is:issue is:open Labels Milestone

① 5 Open ✓ 12 Closed Author Labe

- ① OnRPCResponseListener onError callback should be included #45 opened 20 days ago by joeygrover
- ① mediaTrack param has wrong description. #44 opened on Aug 11 by mtimmar
- ① Fix SoftButton documentation to set softButtonID's #39 opened on May 10 by askirk
- ① Blank pages when viewing guide on Developer Portal #26 opened on Apr 24 by theresalech

[Learn](#)[Documentation](#)[Resources](#)[SDLC](#)

Personal Info

Companies & App IDs

● Timurs Little Company

Theme Preferences

## Timurs Little Company

Company Info

App IDs

Users

### App IDs

Create New App ID

An Application ID uniquely identifies an application in the SDL ecosystem. Use this ID in your application with the SmartDeviceLink SDK. Already have an existing ID? [Register it.](#)

Timurs Little App

155a5052-092c-4665-8af9-0c4f4daad8ec

Edit



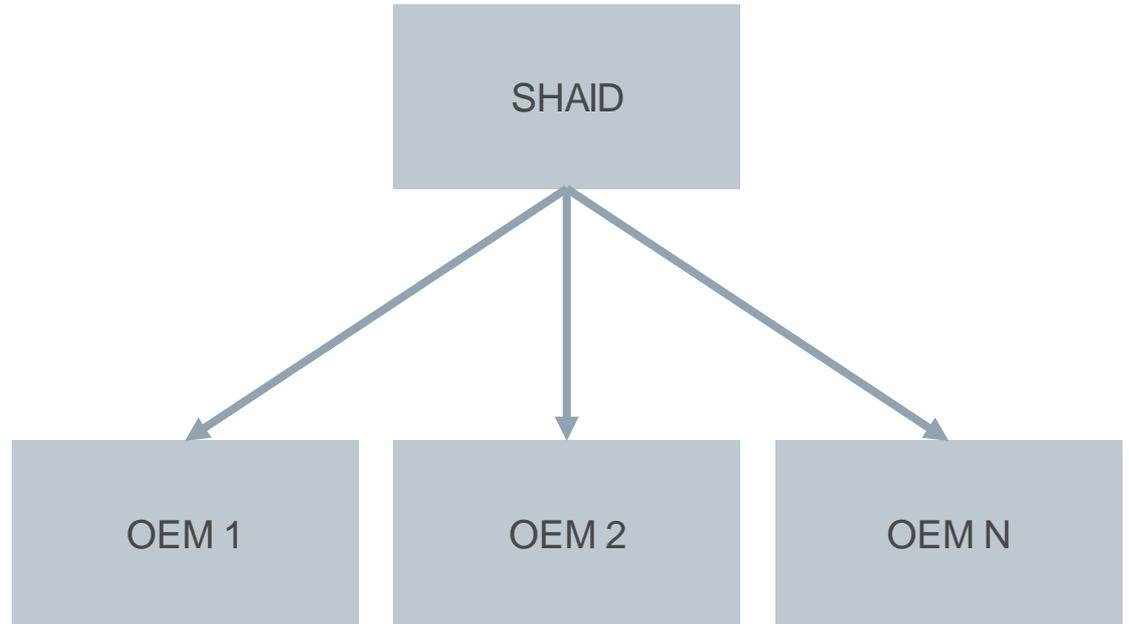
Go Further

# DEVELOPER TOOLS/SHAID

---

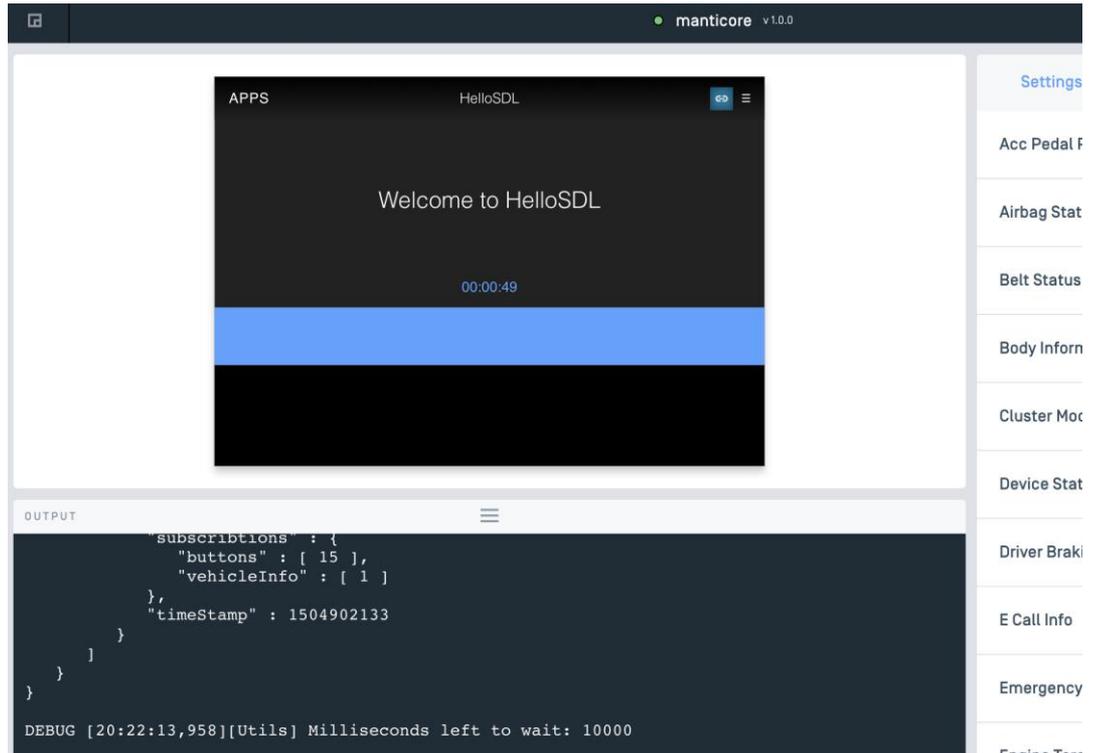
Super  
Helpful  
Application  
Information  
Database

- SHAID allows for centralized management of AppIDs



# DEVELOPER TOOLS/MANTICORE

- Online instance of sdl\_core
- No installation needed
- Quick Testing



The screenshot displays the Manticore developer tools interface. At the top, a dark header bar contains the Manticore logo and version number 'v1.0.0'. The main content area is divided into two sections. The upper section shows a window titled 'HelloSDL' with a dark background and a blue horizontal bar at the bottom. The text 'Welcome to HelloSDL' is centered, and a timer '00:00:49' is visible. The lower section is a terminal window titled 'OUTPUT' showing a JSON object: 

```
subscriptions : {  
  "buttons" : [ 15 ],  
  "vehicleInfo" : [ 1 ]  
},  
"timeStamp" : 1504902133  
}  
]  
}
```

 Below the JSON is a debug message: 

```
DEBUG [20:22:13,958][Utils] Milliseconds left to wait: 10000
```

 On the right side, a vertical sidebar contains a 'Settings' link at the top, followed by a list of vehicle-related status indicators: 'Acc Pedal F', 'Airbag Stat', 'Belt Status', 'Body Inform', 'Cluster Moc', 'Device Stat', 'Driver Braki', 'E Call Info', 'Emergency', and 'Engine Torq'.

# DEVELOPER TOOLS/POLICIES

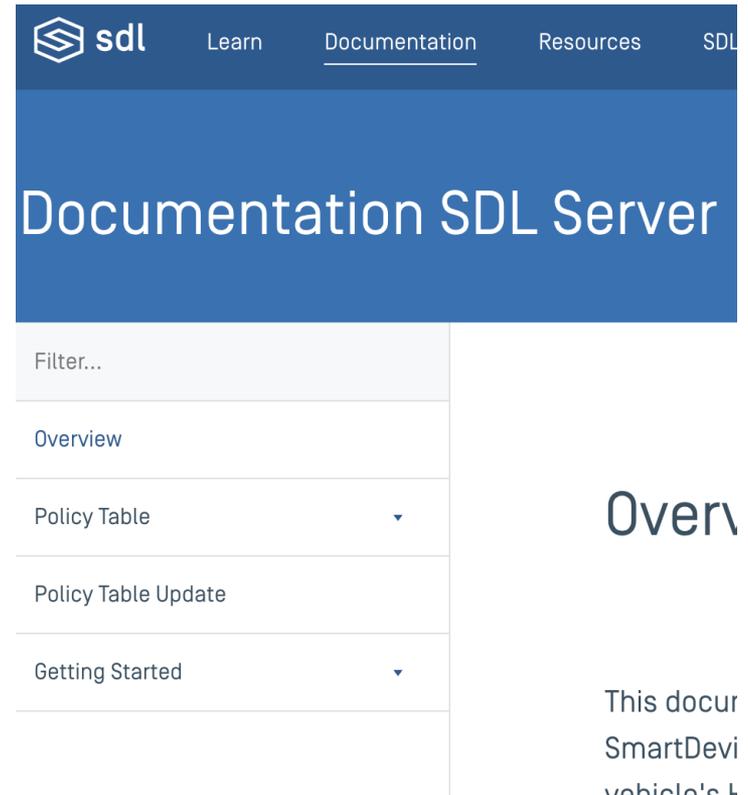
---

- Policies govern access to special functions of SmartDeviceLink
- Restricted RPC examples
  - getVehicleData
  - subscribeVehicleData
  - dialNumber
  - sendLocation
- App sends out policy request
- Backend server creates policy table response, based on its database
- Headunit evaluates policy table response and applies permissions

```
37      },
38      "functional_groupings": {
39        "Base-4": {
40          "rpcs": {
41            "AddCommand": {
42              "hmi_levels": ["BACKGROUND",
43                "FULL",
44                "LIMITED"]
45            },
46            "AddSubMenu": {
47              "hmi_levels": ["BACKGROUND",
48                "FULL",
49                "LIMITED"]
50            },
51            "Alert": {
52              "hmi_levels": ["FULL",
53                "LIMITED"]
54            },
55            "ChangeRegistration": {
56              "hmi_levels": ["BACKGROUND",
57                "FULL",
58                "LIMITED",
59                "NONE"]
60            },
61            "CreateInteractionChoiceSet": {
62              "hmi_levels": ["BACKGROUND",
63                "FULL",
64                "LIMITED"]
65            }
66          }
67        }
68      }
69    }
70  }
71}
```

# DEVELOPER TOOLS/SDL\_SERVER

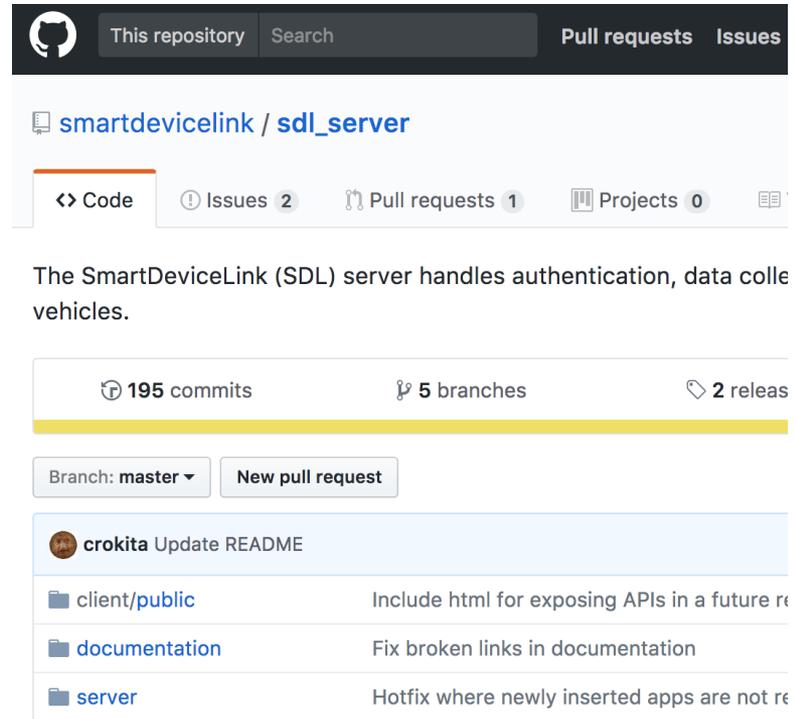
- SDL Server is an example server that allows managing of a backend system
- SDL Server will respond to policy table requests and compile corresponding responses.
- Continuous development to be a turn-key solution for SDL Members



The screenshot shows the top navigation bar of the SDL Server documentation website. The navigation bar includes the SDL logo, the text 'sdl', and links for 'Learn', 'Documentation' (which is underlined), 'Resources', and 'SDL'. Below the navigation bar is a large blue header with the text 'Documentation SDL Server'. To the left of the main content area is a vertical sidebar menu with a search box labeled 'Filter...' and several menu items: 'Overview', 'Policy Table' (with a dropdown arrow), 'Policy Table Update', and 'Getting Started' (with a dropdown arrow). To the right of the sidebar, the word 'Overview' is partially visible, and below it, the text 'This docum SmartDevi vehicle's L' is partially visible.

# DEVELOPER TOOLS/SDL\_ORGANIZATION

- SDL Server is an example server that allows managing of a backend system
- SDL Server will respond to policy table requests and compile corresponding responses.
- Continuous development to be a turn-key solution for SDL Members



The screenshot shows the GitHub interface for the repository `smartdevicelink / sdl_server`. The repository has 195 commits, 5 branches, and 2 releases. The current branch is `master`. A new pull request can be created. The repository contains three folders: `client/public`, `documentation`, and `server`. The `documentation` folder has a description: "Fix broken links in documentation". The `server` folder has a description: "Hotfix where newly inserted apps are not re".

This repository Search Pull requests Issues

smartdevicelink / sdl\_server

<> Code ⓘ Issues 2 🔗 Pull requests 1 📁 Projects 0

The SmartDeviceLink (SDL) server handles authentication, data collection, and management of vehicles.

📄 195 commits 🌿 5 branches 📦 2 releases

Branch: master ▼ New pull request

 crokita Update README

 client/public	Include html for exposing APIs in a future release
 documentation	Fix broken links in documentation
 server	Hotfix where newly inserted apps are not re

# DEVELOPER TOOLS/SDL\_EVOLUTION

---



# DEVELOPER TOOLS/SDL\_EVOLUTION - TRACEABILITY

#197 by theresalech was closed on Jun 21

 **[Accepted] SDL 0067 - Optimize Connection Wait Range** accepted ios

#196 by theresalech was closed on Jun 21

 **[Accepted] SDL 0063 - Display name parameter** accepted android core ios rpc

#184 by theresalech was closed on May 24

 **[Accepted] SDL 0062 - Template images** accepted android core ios rpc

#183 by theresalech was closed on May 24

 **[Accepted] SDL 0060 - Support Indian English and Thai** accepted android core ios rpc

#178 by theresalech was closed on May 17

 **[Accepted] SDL 0059 - Android RPC Refactor** accepted android

#177 by theresalech was closed on May 17

 **[Accepted] SDL 0056 - Hardware Validator App (Android)** accepted meta

#167 by theresalech was closed on May 3

 **[Accepted] SDL 0055 - System Capabilities Query** accepted android core ios rpc

#166 by theresalech was closed on May 3



Go Further

# DEVELOPER TOOLS/SDL\_EVOLUTION - TRACEABILITY

## [Accepted] SDL 0059 - Android RPC Refactor #177

 Closed theresalech opened this issue on May 10 · 6 comments



theresalech commented on May 10

Owner



Hello SDL community,

The review of "Android RPC Refactor" begins now and runs through May 16, 2017. The proposal is available here:

[https://github.com/smartdevicelink/sdl\\_evolution/blob/master/proposals/0059-](https://github.com/smartdevicelink/sdl_evolution/blob/master/proposals/0059-)



theresalech referenced this issue in smartdevicelink/sdl\_android on May 17

**[SDL 0059] Android RPC Refactor #513**

 Closed



theresalech commented on May 17

Owner



Issue entered: [\[SDL 0059\] Android RPC Refactor](#)

- Is the problem being addressed significant enough to warrant a change to SDL?
- Does this proposal fit well with the feel and direction of SDL?
- If you have used competitors with a similar feature. how do you feel that this proposal compares

# DEVELOPER TOOLS/SDL\_EVOLUTION - TRACEABILITY

smartdevicelink / sdl\_android Unwatch 44 ★

[Code](#) [Issues 87](#) [Pull requests 37](#) [Projects 4](#) [Wiki](#) [Insights](#)

## [SDL 0059] Android RPC Refactor #513

Closed theresalech opened this issue on May 17 · 0 comments



theresalech commented on May 17

Owner +

Proposal: [Android RPC Refactor](#)

This proposal is to modify the 'RPCMessage' class to include generalized getter and setter methods. This would allow for the refactoring of 'RPCRequests' and other RPC classes to use these methods rather than continuing to duplicate code that achieves the same result.

Review: [smartdevicelink/sdl\\_evolution#177](#)

theresalech added the **proposal** label on May 17

askirk referenced this issue on Jun 7

**Feature/issue 513 - RPC Refactor #536**

Merged

askirk closed this on Jun 21

# DEVELOPER TOOLS/SDL\_EVOLUTION - TRACEABILITY

smartdevicelink / sdl\_android Unwatch 44

[Code](#) [Issues 87](#) [Pull requests 37](#) [Projects 4](#) [Wiki](#) [Insights](#)

## Feature/issue 513 - RPC Refactor #536

**Merged** joeygrover merged 8 commits into develop from feature/issue\_513 on Jun 8

[Conversation 1](#) [Commits 8](#) [Files changed 146](#)

 askirk commented on Jun 6 • edited Owner +😊

Refactored RPC Class methods using new generalized methods in RPCMessage.java and RPCStruct.java  
RPC unit tests passing.

[#513](#)

  askirk requested review from **brettywhite** and **joeygrover** on Jun 6

 askirk added some commits on May 31

-   Add Generalized Methods and Refactor Setters in RPCs 48d41d6
-   Refactoring with new Getters and Setters in RPCMessage and RPCStruct ... cea9db5

## SDL Evolution Proposal Status

The [SDL evolution process](#) describes the process by which SDL evolves. This page tracks th

### Active reviews

- SDL-0091** **ios** SDLCarWindow Video Projection Developer Interface
- SDL-0092** **core** Mark public deprecated methods
- SDL-0093** **android** Haptic Projection
- SDL-0094** **policy** SDL Policy Server Enhancements - UI Framework

### Accepted (awaiting implementation)

This is the list of proposals which have been accepted for inclusion into SDL, but they are not implemented in time for the next version of SDL, they will roll into a sub-

- SDL-0014** **rpc** Adding Audio File Playback to TTSCChunk
- SDL-0016** **ios** iOS Library Make Transport Classes Private and Cleaner
- SDL-0017** **ios** iOS Library Protocol Layer Should Not Be Public
- SDL-0019** **ios** iOS Library Proxy Layer Should Not Be Public
- SDL-0037** **rpc** Expand Mobile putfile RPC
- SDL-0040** **protocol** DTLS encryption
- SDL-0041** **rpc** Provide Applcon resumption across app registration requests

[https://smartdevicelink.github.io/sdl\\_evolution/](https://smartdevicelink.github.io/sdl_evolution/)