Prpl High-level API Group

Decision paper: Fundamental Principles

Foreword

The prpl Foundation, an open-source, community-driven consortium with a focus on enabling the security and interoperability of embedded devices for the IoT and smart society of the future, announced last year the formation of a new working group that aimed to increase interoperability and unification for development of Wi-Fi routers and home gateway devices. The goal of this High Level API working group was to put effort on the much needed common Application Programming Interface (API) recommendations for the home gateway industry with the goal to simplify development, reduce overheads and shorten time to market for home gateway devices.

Prpl brought together vendors, silicon manufacturers and the open source community for the first time to relieve interoperability issues surrounding the industry-wide problem of a complex home gateway development and maintenance ecosystem, which is often comprised of proprietary components which makes interoperability efforts very high, increases development costs and creates a lengthy time to market.

Background

Customer demand for ever more features, has led carriers, manufacturers and retail brands to apply more and more software customization to gateways, access-points and routers.

Unfortunately, keeping all this software customization in sync across different devices and models quickly turns into a technical nightmare. The main reason is the large fragmentation of embedded operating systems and middleware stacks across the different manufacturers and suppliers.

To overcome this, the prpl membership, a diverse group of industry players ranging from carriers, to middleware vendors, from manufacturers to test houses, has decided to come together and jointly address this issue. Consequently, the members decided to create the high-level API working group, a group of experts tasked to facilitate the development and standardization of a common intra-device protocol.

The ultimate goal of prpl's high-level API working group is to dramatically reduce the customization time and effort for any router, access-point and gateway. It aims to do so by harmonizing the software development processes and standardizing the on-device APIs for features such as the device's web interface or smart home hubs. All of this will be done using open source protocols. From an end user perspective, this also means more pervasive adoption of technology, such as smart home; the ability to support more services, innovative technology at lower prices points and faster availability of new features and devices.

Over the past twelve months, this group has evaluated different technology approaches as the baseline for such a protocol. The group evaluated carrier-grade approaches such as netconf/YANG, TR-181/TR-069 and community approaches such as OpenWrt's UCI and NetJSON.

To conclude on the first phase of the working groups' task, the group moves to ratify the fundamental principles necessary to move into the detailed specification phase.

Principles

Purpose - Prpl's high-level API is an abstraction API.

It aims to make higher-level processes portable between different middleware stacks. It does so by requiring vendor-specific implementations southbound while offering vendor-agnostic APIs notherbound (e.g. a web UI could be written independent of the underlying middleware by integrating prpl's high-level API).

Scope - Prpl's high-level API is an on-device API.

It shall focus on intra-device and inter-process communication. Access from outside clients or systems needs to be handled by protocol agents (e.g.: USP, MQTT, HTTPS).

Platform support - Prpl's high-level API is a cross-platform API.

It shall not be limited to one implementation or be linked closely to one platform. This is to ensure and encourage the portability of components consuming the high-level API across different software stacks, enabling a cross-platform ecosystem.

Schema language - Prpl's high-level API is defined in JSON object models.

It shall be made up of device and service objects, their addressable paths and respective data-models. It will take strong inspiration from community approaches and carrier-grade APIs alike.

IPC - Prpl's high-level API is designed in an IPC independent manner.

It shall not mandate certain IPC mechanism. Any vendor shall be free to implement prpl's high-level API on their own IPC, as long as the IPC's interface are API compatible with prpl's definition.

Architecture - Prpl's high-level API does not replace any existing API.

Architecturally, it shall sit adjacent to any middleware-specific APIs and use these via its southbound interfaces.

Implementation

API foundation - Vodafone's contributed proposal shall be the foundation.

It contains an object model definition for a set of carrier services, such as, web interface and smartphone application. Each object is defined under its own chapter where properties, methods and events of the respective object is given in form of tables with details of name, type, list of values, access and description. Initial return codes, which are common for all objects are also present.
(https://basecamp.com/2935588/projects/12393738/messages/75812820?enlarge=321622266#attachment_321622266)

Platform scope - OpenWrt & RDK-B shall be the reference target platforms.

These two platforms form the foundation of most stacks across DSL and cable and are therefore the natural reference platforms for any software independent API.

Reference IPC - uBus and DBus shall be the IPC reference APIs.

Based on there two reference platforms, uBus and DBus are the logical choices for the first supported IPCs.

Annex

Architecture Overview

Figure 1. Layers

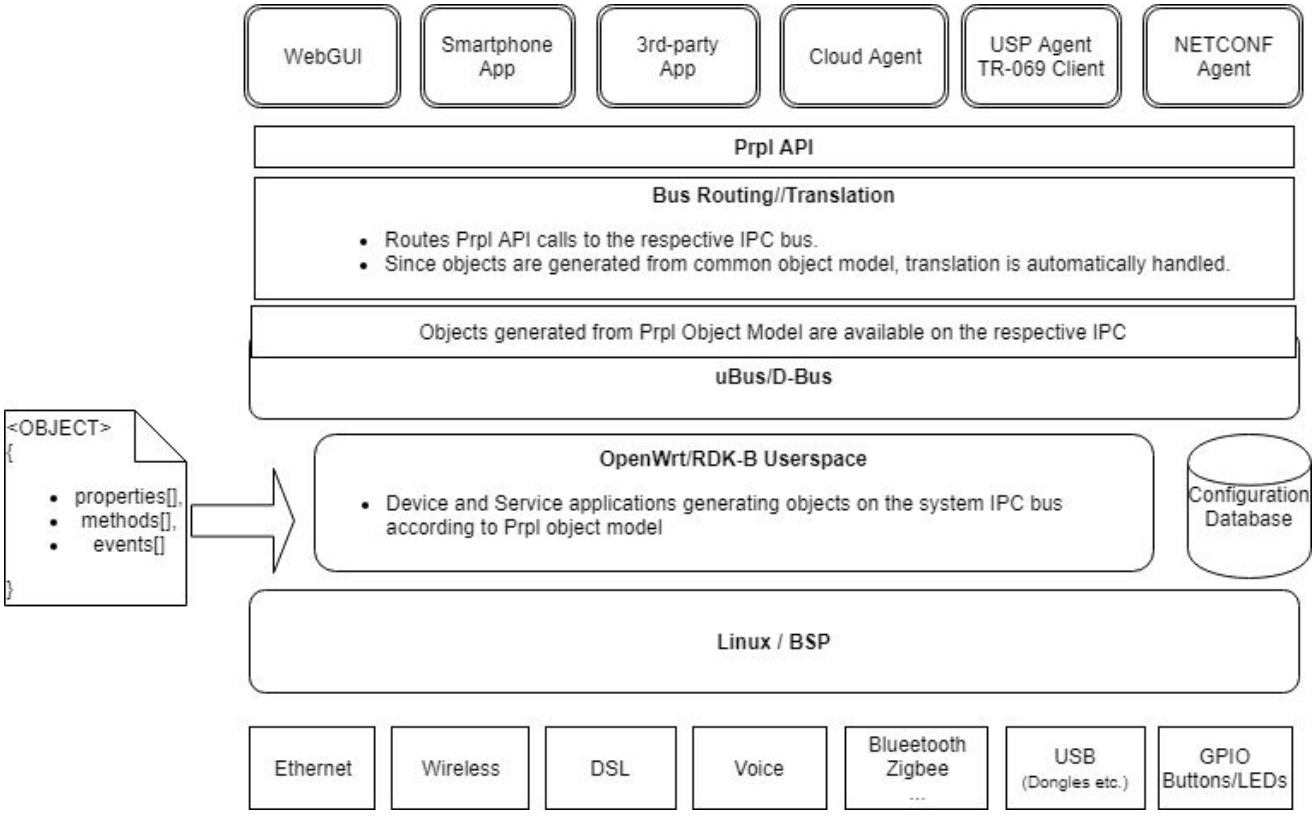
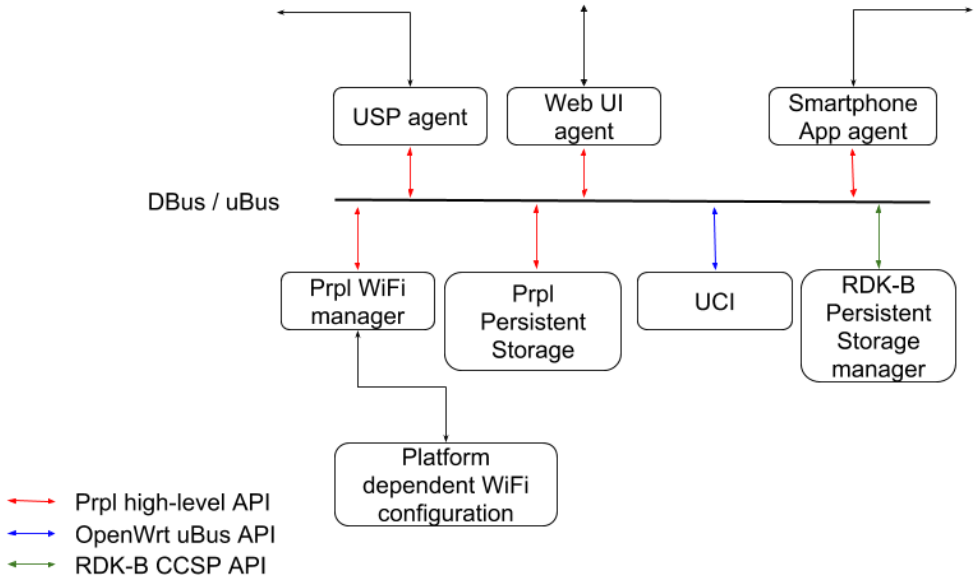


Figure 2. High-level Multi-OS Component/bus Overview



Terms

Term	Definition
Protocol Agent	A protocol implementation, and the whole machinery to translate a protocol into interacting with the bus, translating API's and data models.