OMA Case Study Outline: GotAPI

OMA GotAPI Overview

- Abbreviation of Generic Open Terminal API Framework
- Standardized by OMA released in February 2015
- A total framework with enhanced security features to enable device Web APIs in smartphones like Android, iPhones
- Applications (Web apps, native apps, and hybrid apps) are able to work with external devices and internal applications, through the APIs using Web technologies
- Web applications, downloaded from Web sites, work in standard browsers with external devices
- Ultimately, the GotAPI framework allows web-based device APIs in smart devices to work seamlessly across multiple platforms and operating systems.

Why it is important for DOCOMO to have GotAPI standardized in a standards body (OMA)?

Standardization in OMA

- Accelerates wide spread adoption of GotAPI technology
- Helped inauguration of the Device WebAPI consortium (http://en.device-webapi.org/index.html) in April 2015 which promotes the use of GotAPI in various applications
- Help worldwide reach of the technology
- Helps support adoption by making the detailed specification open and endorsed by the OMA community

Case-1 : Vuzix (http://www.vuzix.com/consumer/products_m100/)

Circumstances / Difficulties:

- No suitable development framework for Vuzix Monocular Smart Glass, M100.
- Vuzix would use a communication protocol like BLE which is not supported by the device.

Solution / Way to Achieve:

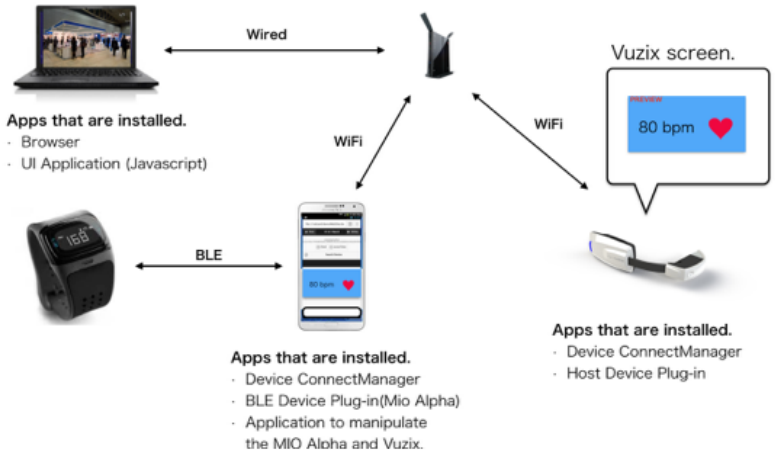
- Improve connectivity to external devices for remote operation support.
- Framework to be able to construct a system for easy live streaming.

Action:

- Constructed the system with WebRTC to do live streaming by GotAPI, no longer dependent on any web browsers.
- It is able to control the device via REST API to switch monitor screens of glass and other external devices, enable to show on glass screens for video chat on smartphones and monitoring the users heart rate from BLE device via GotAPI, which can also be seen on the glass screen.

Result of the Action:

- Vuzix demonstrated the system and its functions at the OMA booth at MWC 2015[1], promoted M100 with the system in exhibitions and has been making solution proposals based on the system.



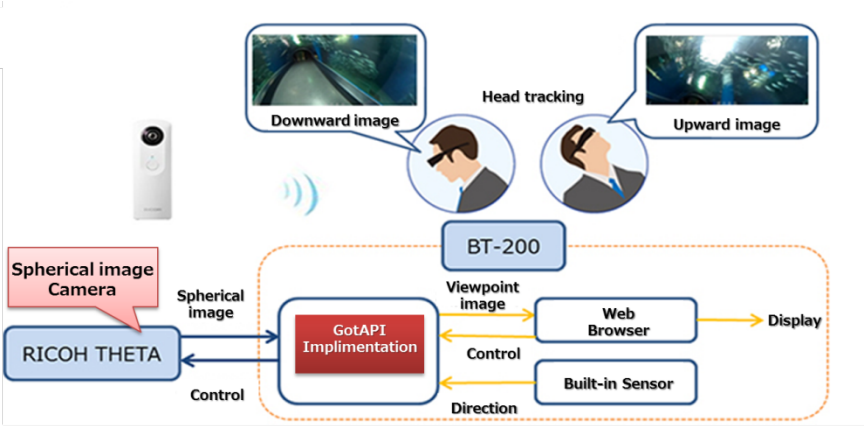
[1]:http://www.prnewswire.com/news-releases/vuzix-to-attend-mobile-world-congress-2015-300043386.html

Case-2: Epson

(http://www.epson.com/cgi-bin/Store/jsp/Landing/moverio-bt-200-smart-glasses.do)

Circumstances / Difficulties:

- Epson planned to make a system to look at pictures taken by an omnidirectional camera, RICOH Theta, in Virtual Reality with their binocular eyewear, Moverio BT-200. However the official app of Theta was not provided for the Moverio, furthermore web browsers in the eyewear didn't support the newest HTML5 spec (WebGL etc), therefore it was very difficult to generate a VR image which matches to its features.



Solution / Way to Achieve:

- Generate VR image with browser base on an old HTML spec version.
- Real-time image processing with sensor for attitude of Moverio to be able to look at images on omnidirection.
- Establish communication connection between Moverio and Theta.
- Framework to be able to retrieve pictures from Theta and control Theta to take pictures or movies.

Action:

- Constructed the system to control Theta and generate real-time VR image with omnidirectional picture from Theta. The system is able to be made without the device's own specs and also technical knowledge of 3DCG processing.

Result of the Action:

- Epson has released to the press that the system was developed with GotAPI by collaborating with RICOH, NTT Docomo[2].

[2]: http://www.slideshare.net/device-webapi/ricohthetamoverio (Japanese Only)

Case-3: Sony-CSL (http://kadecot.net/English/)

Circumstances / Difficulties:

- Sony-CSL provides a middleware for controlling home appliances named “Kadecot”. To expand to apply to more appliances, it needs much more time and increased workload due to the many varieties of appliances.

Solution / Way to Achieve:

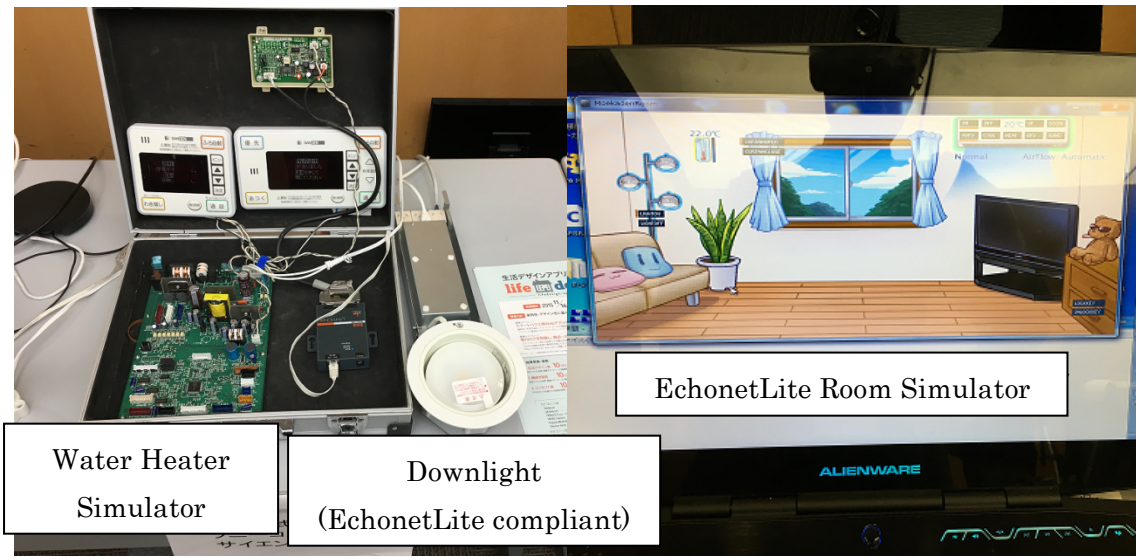
- Simple framework which enables to apply appliances and expand applications more easily.

Action:

- Cooperated with Kadecot and “Device Connect”, the framework complies with GotAPI, produced by NTT Docomo.

Result of the Action:

- Added some functions of GotAPI to Kadecot (published in Google Play). It enables GotAPI from Kadecot, and also GotAPI can control devices which are managed in Kadecot because Kadecot can perform as a plug-in of Got API[3].

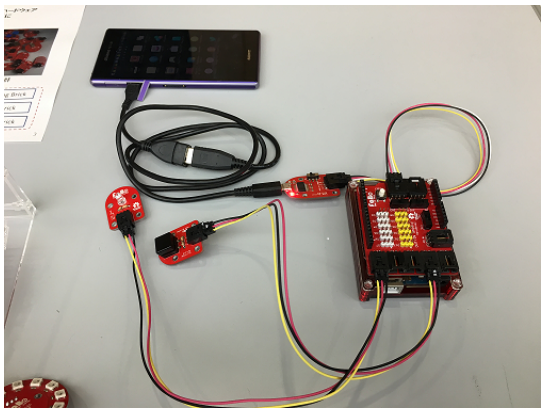


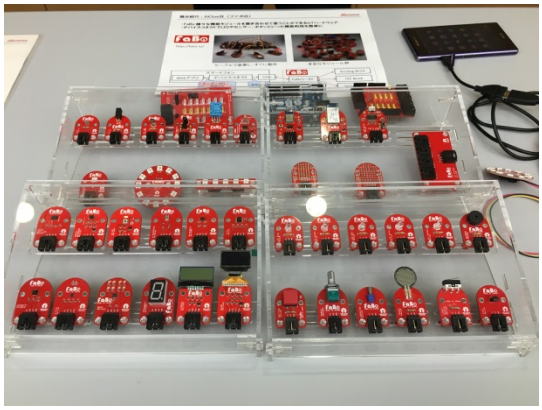
[3]: https://play.google.com/store/apps/details?id=com.sonycsl.Kadecot&hl=ja

Case-4: GClue (http://www.gclue.com/)

Circumstances / Difficulties:

- GClue intends to launch modules "FaBo"[4] which enables to link several sensors and communication modules by only simple wiring harness.





Solution / Way to Achieve:

- Easily accesses modules with USB host function of smartphones.

Action:

- Constructed the system to access those modules from web browser in smartphones via USB host. (Currently only one API is available for GPIO. In future other advanced APIs will be developed for sensors / modules.)

Result of the Action:

- The system solution has been promoted and proposed at many exhibitions.

[4]: http://fabo.io/

Case-5: Utilization in Hackathon, App Contests

Circumstances / Difficulties:

- Many kinds of devices are used in Hackathons and other App events. It would be difficult to understand its spec and make a developmental environment in a limited time if it is the first time to use that device.

Solution / Way to Achieve:

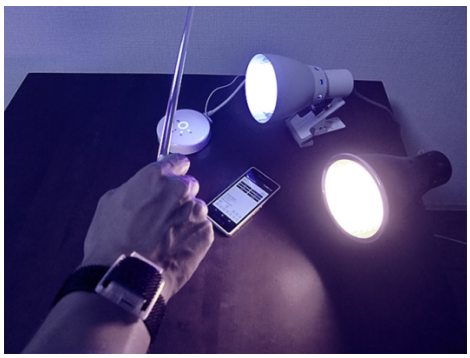
- Provide an environment which enables the use of the device's functions without any concerns of difference of spec on each device.

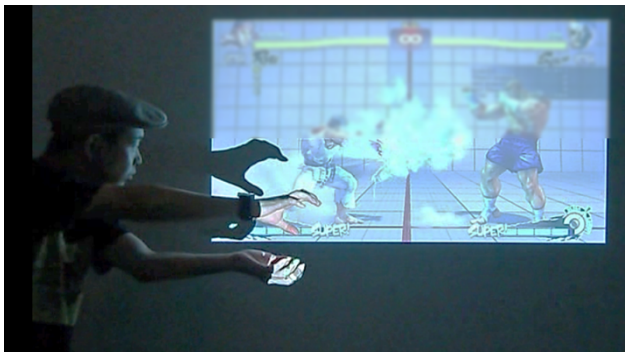
Action:

- NTT Docomo has provided a framework "Device Connect", complying with GotAPI, in the Mashup Awards, the largest App Contest in Japan.

Result of the Action:

- Device Connect was utilized by 2 teams, won in the elimination round in Hackathon, and a product won the 3rd place[5] out of 431 nominees.





[5]: http://hacklog.jp/works/3897