LiMo Foundation: Toward a Common Linux-based Mobile Platform

Mitsuaki Morita, Yuichi Ichikawa, Kazuaki Terunuma and Hyunsuk Seung

*LiMo Foundation was launched in January 2007 with the aim of creating the world's first globally competitive Linux®*1*-based platform for mobile terminals. This article describes efforts undertaken by the LiMo Foundation to develop this platform.*

1. Introduction

Functions equipped on mobile terminals are becoming increasingly advanced every year. To provide users with services that are compelling and unique, innovative new software is needed these days to implement functions such as music and video playback, display of up-to-date information via captions flowing across a standby screen, security functions for misplaced or lost mobile terminals, etc. This trend is not limited to Japan—it is also happening in the global market. But software for satisfying users' need for diverse and high-performance services are becoming more complicated and sophisticated, and the cost of software is rising as a result. Finding ways of reducing this cost burden has become a common challenge among mobile operators and mobile terminal manufacturers.

Against this background, NTT DoCoMo and five other mobile operators and mobile terminal manufacturers launched the "LiMo Foundation" in January 2007. The LiMo Foundation aims to create the world's first Linux-based global platform for mobile terminals to overcome the above challenge.

This article describes an overview of the LiMo Foundation, its software architecture, the platform it provides, and the platform's relationship with Mobile-phone Oriented Application Platform (Linux) (MOAP (L)).

2. LiMo Foundation Overview

2.1 Aims and Objectives of the LiMo Foundation

The Foundation aims to enable the provision of attractive services to users for mobile operators and mobile terminal manufacturers to shorten development time and time-to-market of mobile terminals and reduce the price of mobile terminals by reducing license fees. Furthermore, founder members—NTT DoCoMo, Motorola, Inc., NEC Corporation, Panasonic Mobile Communications Co., Ltd., Samsung Electronics Co., Ltd., and Vodafone Group—aim to create a Linux platform for mobile terminals that is competitive compared to other platforms, construct an ecosystem essential for the expansion of the platform over the medium and long term, and enhance consumer experiences across a wide range of users. The idea here is to expand the adoption of Linux in the mobile phone industry through contribution-based development efforts of member companies.

2.2 Benefits of Participation

Linux OS is distributed under an open-source license that provides the development community with three key features: (a) transparency—anyone may reference the OS source code, (b) innovativeness—OS functions on

*1 **Linux®**: A registered trademark or trademark of Linus Torvalds in Japan and other countries.

servers and personal computers can be improved by programmers throughout the world, and (c) extendibility—the Linux OS can be extended to niche applications much easier than other OS's. The LiMo Foundation will combine these three features with the best practices of the commercial-development model in the mobile industry to provide participating members with a globally competitive platform for mobile terminals [1].

Operator Benefits

- Reduced terminal purchasing cost
- Shortened terminal development time
- Reduced support costs, greater compatibility, etc. for new global services

Mobile Terminal Manufacturer Benefits

- Reduced terminal development cost
- Formation of collaborative relationships among mobile terminal manufacturers
- Allows a company's own engineering team to focus on product differentiation

Chip Vendors

- Ensures a platform supporting a company's own chip sets
- Access to mobile-terminal source code

Middleware Vendors

- Greater opportunity for a company's own products to be adopted
- Reduced support costs through use of a common platform
- Continued support of existing business models through royalty-bearing code

Application Vendors

- Creation of new services and businesses and building of a new customer base through adoption of a company's own software

2.3 Delivered Items

The LiMo Foundation will, in principle, provide members with the source code of the Linux OS-based software platform for mobile terminals on a copyright royalty-free basis. It will also publish the Application Program Interface (API) specifications[*2] defined by the LiMo Foundation for public use and will enable third party developers to create application products using development environments such as Software Development Kit (SDK)[*3] provided by various companies.

2.4 Organization

Figure 1 shows the basic organization structure of the LiMo Foundation.

The main components of the Foundation are the Board of Directors (the "Board"), Executive Council, Architecture Council, and Requirements Council. It also includes the Foundation Program Office that manages the day-to-day operations of the Foundation and the activities of Foundation members.

The Board consists of 13 seats, 6 of which are held by the Founder Members. NTT DoCoMo serves as chair of the Board. The Executive Council consists of members corresponding to the Board representation and is essentially a decision-making body that coordinates discussions outside the Board agenda and oversees the work of the Architecture Council, the Requirements Council, and the Foundation Program Office. The Architecture Council oversees technical matters and has working groups under its wings to perform development work on a contributory basis. The role of the Requirements Council is to capture service requirements.

2.5 Membership

There are three classes of Foundation membership: Founder Member, Core Member, and Associate Member.

1) Founder Member

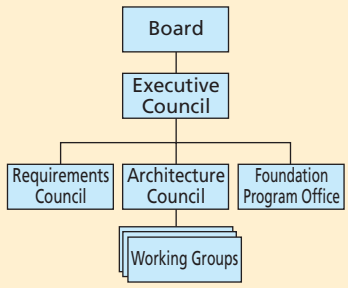

Figure 1 Basic organization structure of the LiMo Foundation

*2 **API specifications**: Interface specifications that enable upper-level software to use OS and middleware functions.

*3 **SDK**: A set of documents, tools, libraries, sample programs, etc. needed to create applications.

Eligible to sit on the Board for the first five years. Basic rights other than the Board seat are the same as that of the Core Member.

2) Core Member

Basic rights include the following: eligible for election to the Board, eligible to vote in accordance with Foundation Bylaws, eligible for election to all Foundation Councils and to chairs of the same, right to access and modify Foundation source code and to ship products. This is the membership class that is expected to be the center of Foundation activities.

3) Associate Member

Enables participation in the Foundation with a low membership fee. Although Associate Members are not eligible for election to the Board, they are eligible for election to the Requirements Council and have the right to participate in working groups. This category was established to invite participation from Independent Software Vendors (ISVs)[*4].

2.6 IPR Policy

The Intellectual Property Rights (IPR) model [2] provided by the Foundation combines open source license policy with IPR policy unique to the Foundation resulting in the following four types of licenses. The code listed below refers to software code, such as source-code and object-code, submitted by members that is a component of, or is proposed to be a component of, the Foundation platform.

1) Open Source License (Common Code)
2) Foundation Public License (Common Capable)
3) Foundation Public License (Non-Common Capable)
4) Proprietary License (royalty bearing)

Of these, only types 1) and 2) are considered for inclusion as part of the Common Code of the Foundation platform. That is, products conforming to the Foundation platform specification must contain all Common Code. Types 3) and 4) correspond to code that may not be common to all members but provides service- or geographic-market-dependent functions for product differentiation. Common Code is licensed to all members on a copyright and royalty-free basis. Non-discriminatory royalty fees may be collected for license types 3) and 4) at the discretion of the rights holder. Also, as a unique scheme of the Foundation, code contributed under license types 3) and 4) may be automatically converted to license type 1) or 2) on the automatic-conversion date designated by the contributor at the time of the contribution. This scheme aims to broaden the portion of code that can be used in common for free, thereby lowering the cost of Foundation products.

3. Software Architecture Overview

The LiMo Foundation provides middleware functions configured as frameworks and plug-ins that can be used in common by mobile terminals. A framework is an abstract layer that provides functions independent of proprietary functions or proprietary hardware. Functions that are dependent on specific services or specific hardware are implemented as plug-ins. The LiMo Foundation platform with a configuration consisting of frameworks and plug-ins achieves a balance between uniformity and flexibility.

Figure 2 shows the configuration of the software architecture provided by the LiMo Foundation [3]. The portion titled "Platform Scope" corresponds to the platform provided by the LiMo Foundation. The functions within Platform Scope are configured as frameworks or plug-ins as needed. The role of each of these functions is summarized below.

1) Application Manager Framework/ Application UI Framework

The Application Manager Framework activates applications and includes a secure package installer for downloaded applications. The Application User Interface (UI) Framework manages user input, provides a user interface, and includes The GNU image manipulation program ToolKit (GTK+), a Graphical User Interface (GUI) tool kit.

2) Registry

Provides storage space for storing records (system data).

3) Conflict Management

*4 **ISV**: A software company not under the umbrella of certain software providers.

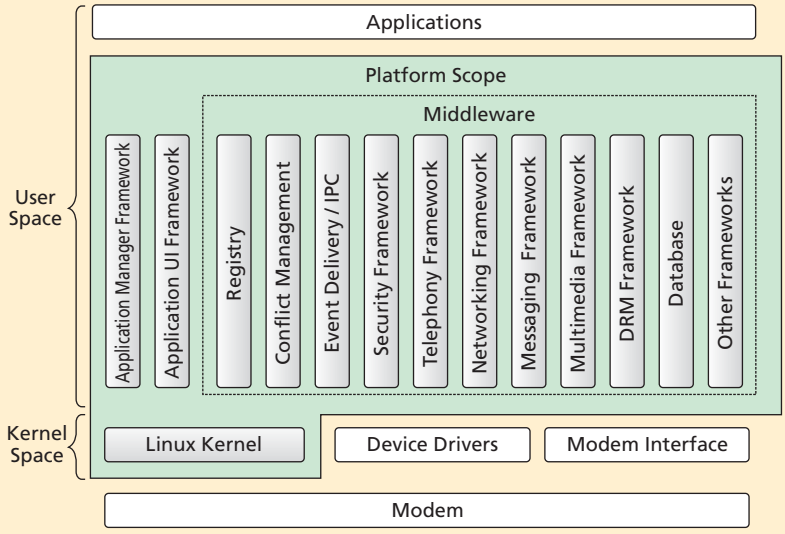

Figure 2 Configuration of software architecture

Resolves and manages conflicts among multiple applications that are simultaneously requesting access to shared resources and conflicts between access requests from applications and system state.

4) Event Delivery/InterProcess Communication (IPC)

Creates, filters, registers, and issues messages for inter-process communication.

5) Security Framework

Scans applications or content before installation and applies security policy according to access rights held by user when attempting to access applications or content after installation.

6) Telephony Framework

A framework concerned mostly with telephone functions. Supports network registration, voice-call control, video-call control, supplementary services, call-charges information, and Subscriber Identity Module and User SIM (SIM/USIM)[*5].

7) Networking Framework

A framework for controlling data communications by providing interfaces for establishing and managing logical connections.

8) Messaging Framework

Manages messaging accounts and profiles and provides interfaces for sending and receiving messages using message functions such as Short Message Service (SMS), Multimedia Messaging Service (MMS), and i-mode mail.

9) Multimedia Framework

Provides capture and playback services for audio, images, and video and manages the operational state of each of these media forms.

10) Digital Rights Management (DRM)[*6] Framework

Provides interfaces for restricting the access and use of content such as music and video according to defined rules and decides whether data can be accessed or decoded for playback.

11) Database

Provides interfaces for performing ACID[*7] (Atomic[*8], Consistent[*9], Isolation[*10], Durable[*11]) data management through data operations such as create, query, update and delete.

4. Relationship between the LiMo Foundation Platform and MOAP (L)

NTT DoCoMo has been collaborating with NEC Corporation and Panasonic Mobile Communications Co., Ltd. to develop the "MOAP (L)" software platform that can be used in common by mobile-terminal manufacturers and software vendors in the development of mobile-terminal software [4][5].

The objectives of MOAP (L) and the LiMo Foundation platform agree in terms of providing common middleware functions on a platform, improving software quality, and reducing development cost.

*5 **SIM/USIM**: SIM is an IC card storing information on the mobile-phone subscriber used mainly in the GSM system. USIM is a functional extension of SIM.

*6 **DRM**: A function for protecting the copyrights of digital content by restricting redistribution, preventing unauthorized copying, etc.

Difference between these two platforms are as follows. MOAP (L) is a platform originally developed exclusively for FOMA terminals and is presently not being applied to any mobile terminals other than FOMA. The LiMo Foundation platform, on the other hand, is a platform that is greatly aware of global mobile terminals adding the Vodafone Group, Motorola, Inc., and Samsung Electronics Co., Ltd., as collaborating companies.

Work is now progressing on merging the best functions of MOAP (L) with the best functions of other companies to construct the LiMo Foundation platform. At NTT DoCoMo, the migration plan shown in **Figure 3** is being studied with the aim of making the development of FOMA terminals even more efficient in the future. The platform functions provided by MOAP (L) are divided into unique functions for achieving attractive FOMA functions and common functions for achieving functions independent of mobile operators. From here on, we expect to make the development process even more efficient by making the common functions provided by MOAP (L) common with LiMo Foundation. New functions needed by FOMA terminals will be developed under MOAP (L) to ensure innovative features.

Furthermore, as part of our contributions to the LiMo Foundation in the years to come, we aim to strike a balance between innovative features and reduction of development costs through reduced maintenance-related cost associated with troubleshooting and functional extensions, through license contibutions, etc.

5. Conclusion

The work of the LiMo Foundation will begin with the creation of a Linux-based platform for mobile terminals by the six Founder Members followed by the development of many new common functions in conjunction with new members joining from April 2007. At NTT DoCoMo, we plan to make the most of this platform to reduce the development cost of future mobile terminals and further improve their functions.

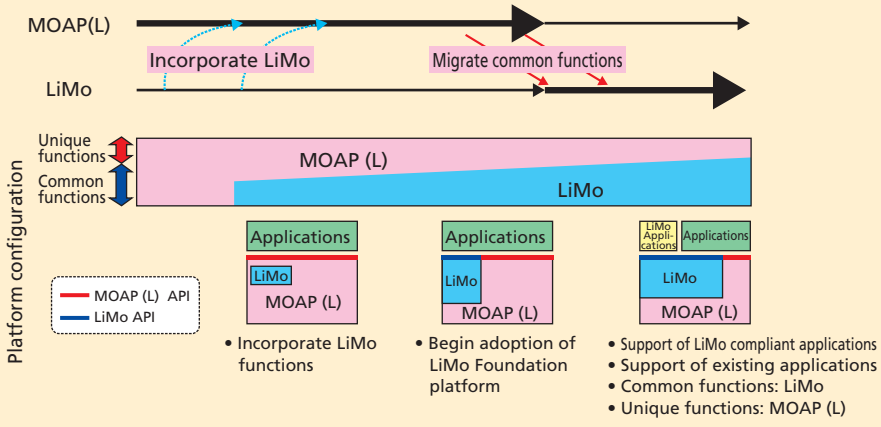

Figure 3 Migration from MOAP (L) to LiMo

REFERENCES

[1] LiMo Foundation, "Overview of LiMo Foundation," Ver. 1.2.2, Apr. 2007.

[2] LiMo Foundation, "Foundation Intellectual Property Policy," Annex A, Bylaws of LiMo Foundation, 2 April 2007.

[3] LiMo Foundation, "http://www.limofoundation.org".

[4] H. Tsuji et al.: ""MOAP," Software Platform for FOMA Terminals," NTT DoCoMo Technical Journal, Vol. 7, No. 1, pp. 40–43, Jun. 2005.

[5] M. Yoshizawa et al.: "Expansion of "MOAP" Software Platform for Mobile Terminals," NTT DoCoMo Technical Journal, Vol. 8, No. 1, pp.15-18, Jun. 2006.

*7 **ACID**: An acronym formed from the words Atomic (see *8), Consistent (see *9), Isolation (see *10), and Durable (see *11), the four conditions that must be satisfied to perform transaction processing in a database management system.

*8 **Atomic**: Guarantees that all or none of the individual operations making up a transaction are executed.

*9 **Consistent**: Consistency is maintained in the database at all times.

*10 **Isolation**: Completely separates transaction processing from other transactions.

*11 **Durable**: Guarantees that a transaction cannot be cancelled once completed.