

# Standardisation Strategies for the Android Platform: The Politics of the Stack

Catherine E.A. Mulligan, Richard Mortier, Robert Houghton, Hanif Rahemtulla, Martin Flintham

Horizon Digital Economy Research,

University of Nottingham,

Nottingham, UK

email: [catherine.mulligan@nottingham.ac.uk](mailto:catherine.mulligan@nottingham.ac.uk)

Christian Nord

CTO Office,

Sony Ericsson,

Lund, Sweden

email: [christian.nord@sonyericsson.com](mailto:christian.nord@sonyericsson.com)

**Abstract—** The mobile device landscape has changed forever with the advent of a truly successful open source mobile platform: Android. This paper illustrates how Android has redefined bargaining relationships within the mobile platform industry. Despite the stated open source nature of Android, many companies have difficulty influencing the direction of the platform. Android, however, can be viewed as supply chains of software, formed from the many open source projects Google integrate to produce Android. This paper identifies leverage points within this supply chain and concludes with an examination of the standardisation strategies companies may use to influence Android.

## I. INTRODUCTION

A key shift has recently occurred in the mobile device industry: previously the dominant operating systems (OS) for mobile handsets was Symbian - an OS that was mainly driven by a mobile company, Nokia within the confines of the innovation regime that had driven the European mobile industry since the 1980s (Mulligan, 2011). The dominance of Symbian was first challenged by the introduction of the iPhone by Apple and then by the release of Google Android in 2008. Within a period of a few years, the majority of OS for mobile devices in developed economies are now created and owned by 3 large computing companies: Apple, Google and Microsoft. More importantly, perhaps, is that the innovation regime of the mobile handset industry is now significantly more US-driven and software based. The Mobile ecosystem has changed fundamentally as a result.

Despite the locus of innovation now being focussed in the USA, the majority of handset manufacturers are not. As a result, these manufacturers are now faced with understanding a new set of innovation rules that have been unleashed on their industry that affect traditional methods for defining architectural rules and distribution of wealth across the industry. All handset manufacturers therefore have an interest in understanding this new form of value chain and how to ensure that their interests are protected within the development and definition of the Android platform. Due to space limitations, this paper focuses specifically on the role that standardisation strategies can play in the new era of mobile handset platforms.

## II. EXISTING VIEWS ON OPEN SOURCE INNOVATION

Traditionally within literature, Open Source OS have been treated as though there is no real platform owner, but rather that firms contribute in order to expand demand by the users of these software products (Economides and Katsamakas, 2006, Lessing, 2001). Developers, meanwhile contribute to maximise user surplus and their development reputation.

### A. Linux vs Windows

Perhaps the most well-known competition between a proprietary platform and an open source one is Linux and Windows and many studies have been performed into the competition between the Linux and Windows ecosystems, e.g. (Bonaccorsi and Rossi, 2003; Bonaccorsi et al. 2004, Economides and Katsamakas, 2006). The majority of the impact of the Linux ecosystem on Windows sales, however, was felt on the server-side, rather than the client side. Large multinational corporations, such as IBM invested a large amount of money and also employee time into promoting and developing Linux for the Enterprise market in order to battle Microsoft's dominance in the market (Economides and Katsamakas, 2006). IBM had teams of employees investing time and effort into developing the Linux platform, which they contributed to the existing governance structure of the Linux ecosystem. IBM used the open source nature to create a battle ground between themselves and Microsoft - it did not attempt to control the governance structure of Linux itself.

The existing literature on open source projects is therefore in direct contrast to the Android platform in two ways, therefore. Firstly, Android represents the first real challenge by Linux into the client-side market. Even today within the PC industry, Linux represents a very small percentage of client-side OS in comparison to Windows which still had 92 percent of the market in 2009. Android, however, now enjoys a majority market share of the Mobile Internet Device (MID) market. In addition, the governance structure of the Android platform is such that it is strictly controlled by the platform owner - Google. To some extent, therefore, Android may actually

be viewed as a proprietary platform with an open source contribution mechanism that companies may contribute to if Google allows them to.

Rather than an Open Source project as defined in previous literature on Linux, therefore, Android is perhaps better viewed as a de-facto standard that companies need to understand how to best utilise for their benefit. Using Android as an OS on a mobile device is in some sense more similar to companies that selected to use de-facto standard SQL in the early 1970s. Through aligning themselves with the most likely industry standard, companies such as Oracle were able to establish strong markets for themselves. In selecting SQL, they also created a positive feedback loop for IBM - ensuring the success of their technology (Mulligan, 2011).

### *B. Platforms and Industrial Structure*

A platform defines the architectural rules and guidelines for interaction [Cusumano, 2010] and have formed the basis of the computing industry since the 1960s and the mobile industry since the 1980s (Mulligan, 2011). Platforms are the "hubs" around which firms form their business and technical strategies (Economides and Katsmakas, 2006). The computing industry has generally followed de-facto standard creation for software. The telecommunications industry has followed a more strict process of standardisation where operators and other interested parties all commit and contribute to the standards before any products are made [Olsson, 2007 quoted in Mulligan, 2011]. Within the telecommunications industry, therefore, the standardisation process has to a large extent been the place where the architectural guidelines for a particular platform are defined and agreed across all parties. Until recently, the platforms of the mobile and computing industries functioned separately.

Within the computing and mobile communications industries today, however, there are now a multitude of platforms, all interacting with one another in a series of complex market dynamics (Mulligan, 2011). As a result, changes within the innovation regime of one sector of the market can have dramatic effects on the innovation regimes of the adjacent platforms. As an example, the introduction of platforms such as Android have fundamentally reformed the innovation regimes of the service layer of the entire mobile industry, not just the mobile handset industry. As a result, the immense changes in the mobile handset industry caused by the iPhone and Android have profound effects on the strategy of a large number of companies in the mobile industry.

### *C. Implications of Android for Corporate Strategy*

It is difficult for any company to precisely define the scope of the developments that the new innovation regimes will bring, but there is no doubt that the Android platform requires companies to rethink their strategy. Traditional corporate strategy, in particular the focus on ownership of resources in order to make money from them and the ability to prevent others from copying or re-using the product once made, are being fundamentally challenged in high-tech industry today

(Chesborough, 2009). The traditional corporate response to when, where and how platform architectures and guidelines are established therefore needs re-assessing. In particular, Android challenges the traditional standardisation strategies of the mobile handset industry.

Until recently in the mobile handset industry, mobile device manufacturers often relied heavily on operators and network vendors to do the heavy lifting in standards organisations. For example, Sony Ericsson would rely on the companies active in the RAN groups to drive the necessary parts of Radio and Core network standards for them within 3GPP. Sony Ericsson would then take the standard and implement it. The traditional bargaining relationships between operators, network vendors, mobile handset manufacturers were established in the early era of the mobile industry, when it was heavily driven by European firms and standards institutes. As a result, the innovation regime was heavily influenced by ensuring that a firm's patents were included in the correct standard. Subsequently, a network vendor would use their economies of scale to drive a volume business across the globe. Companies in this industry have therefore developed techniques that place ownership (IPR), barriers to entry, switching costs and co-operative competitors at the centre of their corporate strategy (Chesborough, 2009). These aspects, however, become of secondary importance when faced with a phenomena like Android as it fundamentally challenges the business model not of just one company, but an entire industry.

When investigating corporate strategy around the Android platform, therefore, it is important that the business model of the firm in question is not only understood as a revenue generation model (Amit and Zott, 2001), but as fulfilling the dual goal of value creation and value capture (Amit and Zott, 2001, Chesbrough and Rosenblum, 2002, Teece, 2007).

### *D. Link between Traditional Standardisation and Product Development*

Standardisation is essentially a choice about whether to ensure a particular product is compatible with competitors products [Katz and Shapiro, 1995]. Katz and Shapiro, in their investigations of market outcomes where companies have formed compatibility coalitions identified that in industries where network externalities are large, the choice of where to make products compatible will be one of the most important dimensions of market performance [Katz and Shapiro, 1985]. Standards bodies within both the computing and telecommunications industries have functioned to co-ordinate the expectations of firms with regards to the underlying technology base that their services will be built on [Church and Gandal, 1992]. In the mobile communications industry standardisation basically functions to align the expectations of the operators, vendors and other interested parties. Standard forums help to regulate the constant power struggle that is usually experienced between mobile operators, core network vendors and handset vendors [Olsson, 2007 quoted in Mulligan, 2011].

### III. THE ANDROID SUPPLY CHAIN

Open source operating systems for mobile handsets, such as Android, raise a series of new issues for the mobile device manufacturers. Android allows them to reduce the cost of device development and leverage the Android name currently fashionable with end-user consumers and developers. A problem arises, however, in the fact that Google does not act as a traditional supplier. As discussed in previous sections, the bargaining relationships of the mobile industry had remained relatively stable since the 1980s despite the fact that the technologies had not. A manufacturer developed Requests For Information (RFIs) or Requests For Quotes (RFQs) towards component and operating system suppliers. Based on well-established decision-making processes within each company, an operating system or a particular semiconductor was selected. This system was based on the traditional research and development regime within the mobile device industry: release schedules planned at least 12 months in advance. Suppliers therefore had a reasonable ability to inform device manufacturers of their plans. The ability to use many suppliers was a key aspect of how platforms were developed - as much as possible a manufacturer did not want to be dependent on only one supplier of a piece of hardware or software.

With Android, however, this is not possible. The nature of an open source project means that release schedules are reasonably fluid, to some extent driven by the needs and wants of the community involved. It is therefore not possible for a manufacturer to request a release schedule for features on such a platform. The politics and bargaining relationships around Android itself, however, are even more complex. While it is touted as an open source platform, Google retain significant control over the code base, deciding what is included in the next release of the platform. While others may contribute, Google decides the final functionality of the platform. Many companies have found it difficult to contribute code to the project and have it included as has been widely discussed in the industry. This is especially true for the development of the latest version of the Android platform, as this is considered to be a closed project by Google and key partners until it is released to the wider community for use. An overview of the Android supply chain is illustrated in Figure 1. Due to space limitations, a selection of applications that form the basis of a mobile handset platform have been selection for discussion.

The complexity continues even further, however, when the entire Android stack is considered. The Android stack is not built entirely by Google - it pulls together several other open source projects into a bundle that is then released under the Android name. The Android stack may therefore be viewed as a form of supply chain, with a key System Integrator, Google, which organises its supply chain through open source concepts. Despite its reliance on such concepts, however, this supply chain is still beholden to the same nature of bargaining relationships as those chains formed using traditional methods.

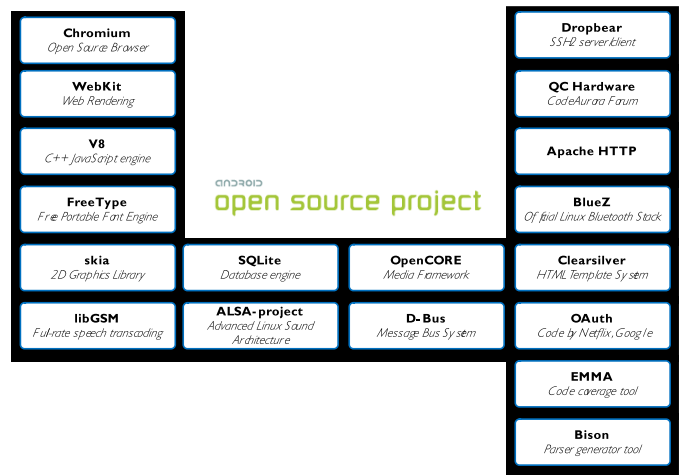


Fig. 1. Android Supply Chain: 2011

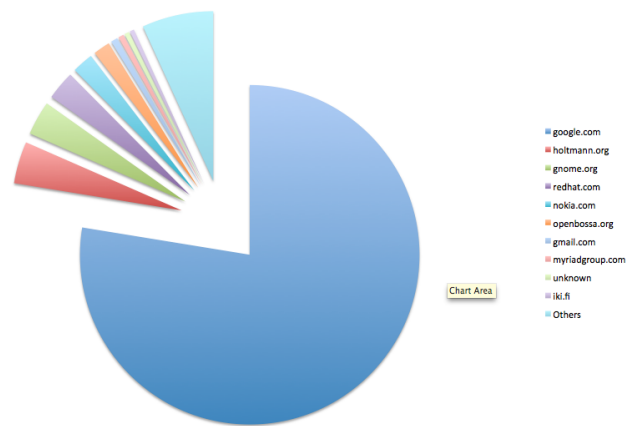


Fig. 2. Top 10 Contributors to Android Browser: 1998 - 2011

#### A. Bargaining Relationships of Existing Stacks

Android, whilst only being released at the end of 2008, it not actually a completely new product as it is built on a standard Linux distribution. This is quite clear from the Android code repositories, some of which date back to 1998, long before Android was developed. As a result, much of the code that Google integrates and relies upon is actually from other companies and developers. Whilst Google in the majority contributor to the Android platform, it is quite interesting to see companies such as Nokia being one of the top contributors to the Browser between 1998 and 2011, as seen in Figure 2. More importantly, however, is the lack of contributions from traditional mobile handset players found in the same era. It is therefore quite reasonable to expect that Mobile Handset manufacturers such as Sony Ericsson would have a low level of bargaining power in the forum for the browser.

A similar situation is found in the MMS application as illustrated in Figure 3

A more interesting picture about the bargaining relationships of the Android stack begins to emerge when investigating the Telephony application on Android, illustrated in Figure 4.

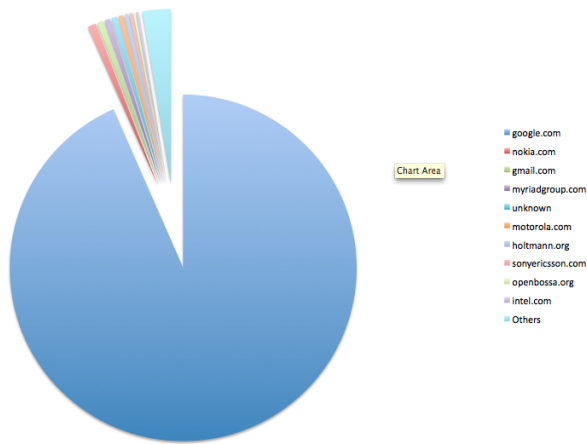


Fig. 3. Top 10 Contributors to Android MMS: 2009 - 2011

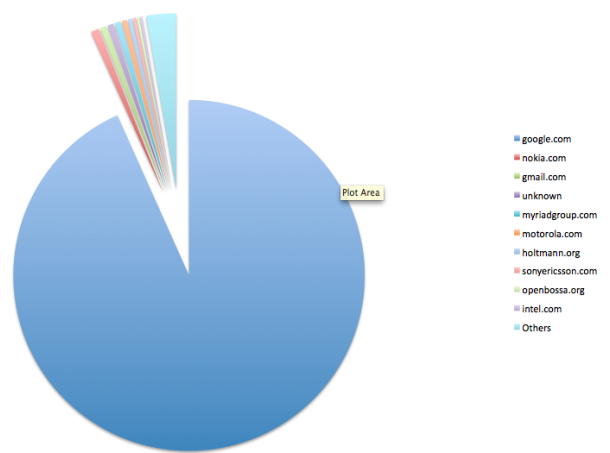


Fig. 5. Top 10 Contributors to Android NFC: 2009 - 2011

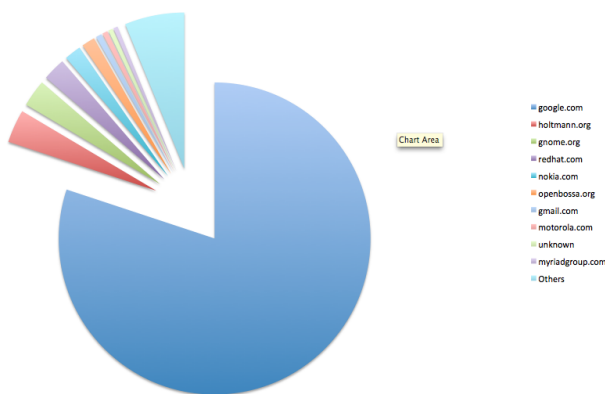


Fig. 4. Top 10 Contributors to Android Telephony: 2009 - 2011

When investigated over the period 2009-2011, Google again is the highest contributor to this part of the platform, with both Nokia and Motorola among the top 10 contributors. Motorola, however, are recorded as contributing a very large amount to the codebase until they are selected as a key partner during 2010 and from April onwards, there are no recorded contributions from Motorola. However, with several products released, it is instead likely that they are included as part of the "Google" contributions. Motorola were effectively brought within the boundaries of Google's area of control for the code base. While Motorola enjoy privileged access to Google's next release, it is unknown how much control they are able to actually execute over the main branch of Android.

### B. New technology introduction

As mentioned, the latest release of Android is considered to be a closed project between Google and a number of different partners. Google, as the platform owner, selects these partners from a set of criteria that matches their release goals for the next version of the platform. Understanding the influence of these partners over the actual platform is extremely difficult. Partners may contribute large amounts of code in order to ensure that a new technology is included in an upcoming release but this is kept internally to the main project - no

firm that is not a selected partner knows which company has contributed what - in the repositories, it is only ever seen as a "Google" contribution. As an example, NXP has been working closely with Google to get NFC into the stack, however Figure 5 illustrates that there have been no contributions by NXP to the Android repository. It is left to externally parties to guess how much NXP have contributed of the large section submitted by 'Google'. From the perspective of the boundaries of the firm, NXP have to some extent become part of Google, tightly bound into their supply chain for delivery of the Android platform.

This example highlights another interesting complexity in the Android supply chain that is less prevalent in other Linux ecosystems: NFC as a technology consists of software and hardware. In practice this means that, in order to have a complete product on the market that supports NFC you must include an NFC chipset. On the one hand, NXP will find itself in a prime-negotiating situation as they have the most knowledge about the platform and can supply both software and hardware to a market eager for Android devices. On the other, Google has essentially formed a digital economy version of a Japanese Keiretsu through blurring the boundaries of the firm suppliers such as NXP, those suppliers also become heavily dependent on Google for both the continued success of the platform and as their main customer. Google, meanwhile, has established near structural control over the relationship between themselves and NXP - Google can call the terms of the relationship or merely exclude NXP from the inner circle of selected partners.

If you are not part of the inner circle of partners that work on the latest release your strategy will undoubtedly involve engagement in projects up-stream of Android. You may opt to not participate and just take the outcome from the public push of Android. This, however, is largely equivalent to not being a member of the right traditional standardisation forums that affect your platform, for example, it is akin to Ericsson not participating in the RAN groups of 3GPP to define the future direction of the radio network platform.

Selecting which upstream projects to engage in is now as critical as any other supply chain decision for a handset manufacturer that has chosen Android. At the same time, those companies that are not part of the select inner circle also have another issue to address - as discussed the Android platform is not a pure open source platform. As a result, even if you establish a key position in one of the upstream open source projects that form the basis of the overall Android platform, it may easily be replaced or re-written between releases, negating the effort that a firm has invested in that particular project. This flexibility again ensures that Google is in a prime position of control over Android - if it appears that another company is gaining too much control over a part of the supply chain, Google may just replace it.

### C. On Hardware

The hardware side of the mobile platforms and the dynamics involved has also changed quite substantially over the last few years. From a model where most manufacturers would do the bulk of the work in-house, to a model where a majority of the device manufacturers uses third parties, or the chipset vendors directly, for the bring-up of a platform. This can quite easily be achieved when the middleware is Android as the expertise is global and applicable to a series of chipsets that are used in Android devices. This change has a major impact on how you develop your product, the core competencies that you bring to the table and the values you provide to the end consumer of your product.

The hardware side is important as a driving mechanism for good PR by the technologically savvy people that will first buy your product. Strategic partnerships between device manufacturers and chipset vendors are becoming a must in order to be first to market with the absolute latest technology.

### D. The Operators

The major mobile operators have had a situation where they pretty much dictated the requirements for products that would be shipped and consequently subsidised through their sales channels. With Android, the situation is very different in that the major developer of the platform is in itself not a device manufacturer. Now, the Android platform in its open source nature lends itself quite easily to be inspected in detail by the operators. Something that was unheard of just a few years back. With detailed knowledge of what can and cannot be done in the platform, operators can put very detailed requirements towards changes that they would like to see put in by the device manufacturers. For those manufacturers dependent on the Operators sales channels, an interesting situation may emerge - changes made to the platform that are not then included in the main branch for whatever reason, will have to be maintained independently by the device manufacturer that is doing the changes. This may lead to the situation where handset manufacturers, despite saving money from using Android in the first place lose the financial advantage through the amount of research and development they need to invest in maintaining those changes. There is a major risk that

a device manufacturer finds itself in a situation where it needs to back-port its patches and changes to every new version of the platform.

### E. Execution

In order to influence a project such as Android, it is important that you bring dedication and expertise to the table. As mentioned, you have the option to participate in open source projects that are up-stream from Android. All such participation requires a mind-set that lets employees work solely and with full focus on open source activities, in many cases on tasks that clearly will put functionality in place that is for the benefit for the greater community. It is the execution that differentiates those who will succeed from the rest. Whereas you would be used to get technology in closed form before, if you aim at getting it to be part of the Android stack it will have to be developed openly. This is a clear change in the bargaining relationship between technology providers and vendors. Frameworks, stacks and other pieces of technology are now provided with source for everyone to see. There is a fine balance between what you would provide openly and the parts that you would consider to be a competitive advantage in keeping closed. This represents a significant organisational shift for companies with in-house developer communities; creating products to have it contributed to a wider development community can come as quite a shock to some.

## IV. CONCLUSIONS

This paper provided a brief overview of the emerging bargaining relationships in the Android platform for handsets. In contrast to existing literature for Open Source Operating Systems, Android has a clear platform owner that is executing extremely strict control over the entire software supply chain. Android also represents perhaps the first extremely successful creation of a Linux ecosystem on the client-side, rather than the server-side. It is therefore extremely important for new studies to be completed to understand the innovation regimes that are forming around the Android platform in order to properly inform corporate strategy in this matter rather than rely on traditional strategy techniques. The paper also highlighted those aspects of strategy that need to be further investigated by those handset manufacturers that are not within the core internal project run by Google and their partners, which is furthering blurring the boundaries of the firm in the digital economy.

## ACKNOWLEDGMENT

The research on which this paper is based was funded by the RCUK supported Horizon Hub, EP/G065802/1.

## V. REFERENCES

- Amit, R., and Zott, C., Value creation in e-business, *Strategic Management Journal*, Vol. 22, No. 6/7, pp. 493-520, 2001
- Bonaccorsi, A., Rossi, C., 2003, Why open source software can succeed. *Research Policy*, Vol 32, Issue 7, July 2003, Pages 1243-1258

Chesbrough, H., 2003, *Open Innovation: the New Imperative for Creating and Profiting from Technology*, Boston, MA: Harvard Business School Press

Chesbrough, H. and Rosenbloom, R.S. 2002, The role of the business model in capturing value from innovation: Evidence from Xerox Corporations technology spin-off companies, *Industrial and Corporate Change*, vol. 11, no. 3, pp. 529- 555.

Church, J., and Gandal, N., 1992, "Network Effects, Software Provision and Standardization," *Journal of Industrial Economics*, vol. 40, no. 1, pp. 85-104.

Cusumano, M., Technology strategy and management - The evolution of platform thinking. *Communications of the ACM* 53(1): 32-34 (2010)

Economides N. and Katsamakas E., 2006, Two-sided competition of proprietary vs. open source technology platforms and implications for the software industry. *Management Science*, 52(7), 1057-1071

Katz M. and Shapiro C, 1995, Network Externalities, Competition and Compatibility, *American Economic Review*

Lessig, L., 2001. *The Future of Ideas*. Random House, New York, USA.

Mulligan, C., 2011, *The Communications Industry in the Era of Convergence*, Oxford, UK

Teece, D. J., 2007, Explicating dynamic capabilities: The nature and microfoundations of (sustainable) enterprise performance, *Strategic Management Journal*, 28, 1319-1350.