2010

Openness of Information Resources – A Framework-based Comparison of Mobile Platforms

Daniel Schlagwein
University of Cologne, schlagwein@unsw.edu.au

Detlef Schoder
University of Cologne, schoder@wim.uni-koeln.de

Kai Fischbach
University of Cologne, fischbach@wim.uni-koeln.de

Openness of Information Resources – A Framework-based Comparison of Mobile Platforms

Journal:	*18th European Conference on Information Systems*
Manuscript ID:	ECIS2010-0467.R1
Submission Type:	Research-in-Progress Paper
Keyword:	Mobile commerce, Open innovation, Open standards, Management science

OPENNESS OF INFORMATION RESOURCES – A FRAMEWORK-BASED COMPARISON OF MOBILE PLATFORMS

Schlagwein, Daniel, University of Cologne, Department of Information Management, Pohligstr. 1, 50969 Cologne, Germany, schlagwein@wim.uni-koeln.de

Schoder, Detlef, University of Cologne, Department of Information Management, Pohligstr. 1, 50969 Cologne, Germany, schoder@wim.uni-koeln.de

Fischbach, Kai, University of Cologne, Department of Information Management, Pohligstr. 1, 50969 Cologne, Germany, fischbach@wim.uni-koeln.de

Abstract

In the smartphone sector we are discerning a competition between Symbian, Apple, and Google for the dominating mobile platform. In the design of their mobile platforms and operating system these organizations use varying degrees of "openness" on different platform elements. We follow a design science approach to construct and apply a framework to better understand how these organizations vary the degrees of openness of different information resources in order to create successful mobile platforms.

Keywords: Symbian, Apple, Google, smartphones, mobile platforms, mobile operating systems, openness, open innovation, open value creation, information management, information resources.

1 INTRODUCTION

We are currently discerning a transition from standard cell phones to smartphones in the mobile industry. Smartphones provide advanced abilities including – among other characteristics, which are mostly derived from PDAs – advanced internet functionality (Ballagas et al. 2006). These advanced abilities are a key factor for the increased mobile internet use instead of PC-based internet use (Economist 2008, Symbian Foundation 2009d). Nokia is still the global leader in the fast-growing market for smartphones, but its devices are losing ground to Apple’s iPhone and Google’s Nexus One (Economist 2010). In this paper we examine the strategy of three major platform providers, which are competing for dominance in the mobile platform sector: Symbian (Nokia), Apple, and Google. These firms are especially interesting to contrast, as their core business is rooted in three different industries. The analysis of the competition between these platform providers offers revealing insights, as the design of the different platforms and operating system is based on varying degrees of “openness”.

The most commonly used system software for mobile devices is the operating system of *Symbian*. Symbian OS is licensed to mobile device manufacturers worldwide (Nokia 2008a). Symbian Limited was founded in 1998 as a joint venture between Psion, an early innovator in mobile computing, and the hardware manufacturers Nokia, Ericsson, and Motorola. In 2008 Nokia became the single owner of Symbian (Nokia 2008a) and until April 2009 transferred all its leases and trademarks to The Symbian Foundation. In 2007, following the great success of Apples iPod music players, *Apple* put its newly developed iPhone on sale (Dowling and Barney 2007). The iPhone is equipped with the proprietary iPhone OS, which is also known as OS X iPhone as a reference to Apples home computer operating system Mac OS X. Later in the same year, *Google*, the founder and main contributor to the Open Handset Alliance (OHA), announced Android, a free and open operating system for smartphones (Open Handset Alliance 2007).

Considering the different approaches how these rivals strive for the dominance in the sector, we see different strategies of “openness” in terms of access to and control of the elements of the mobile platforms. The mobile operating system constitutes the core element of a mobile platform. Such a mobile operating system can be proprietary or non-proprietary software. *Proprietary software* is owned software, where property rights and the control of the software belong to the holder. Third-party use of proprietary software is controlled with strict licenses, if licences are given at all (Pantoni and Brandão 2009). *Non-proprietary software* is usually open source software (OSS), and as such its source code can widely be enhanced, copied, redistributed, and integrated into other software (Open Source Initiative 2009). The degree of openness varies between different open source licenses, like the GNU General Public License (Free Software Foundation 2009), the Apache Software License (Apache Software Foundation 2009) or the Eclipse Public License (Eclipse Foundation 2009). Additionally, operating systems can be categorized as open or closed systems, which is not necessarily identical to the proprietary/non-proprietary distinction. A *closed system* is an operating system that is bound to a specific hardware, while *open systems* can work with hardware from different manufacturers (Chau and Tam 1997). For example, Microsoft’s Windows operating system is proprietary software, but can be used with hardware of different manufactures and is therefore an open operating system. Apple’s Mac OS X is partly non-proprietary software (part of it is open source), but it can only run on Apple’s hardware, i.e. a closed system. Furthermore, mobile operating systems differ with respect to the possibilities for third parties to create and distribute different types of application software for the system.

We could find only one comparative case study of Symbian, Apple, and Google (Suarez et al. 2009), but it does not focus on the importance of openness. We aim to contribute with this paper to the understanding of the determinants of success in the mobile platform competition. First, we will develop a framework to describe the different degrees of openness. Secondly, we will apply this framework to mobile platforms, in order to highlight tangible differences between the different platforms. To do so, we break the rather complex construct of "mobile platform" down into information resources that are then examined separately.

We follow a research procedure derived from design science approach (Hevner et al. 2004) for the analysis, and align our research steps to this approach. Design science aims to solve practical and theoretical problems by creating artefacts – constructs, models, or methods (March and Smith 1995) – that can contribute to theoretical knowledge of, or practical solutions to, problems that are identified in an organisational context. We strictly apply the steps of the design science research method as recommended by Peffers et al. (2007). The resulting research design is depicted in Figure 1.

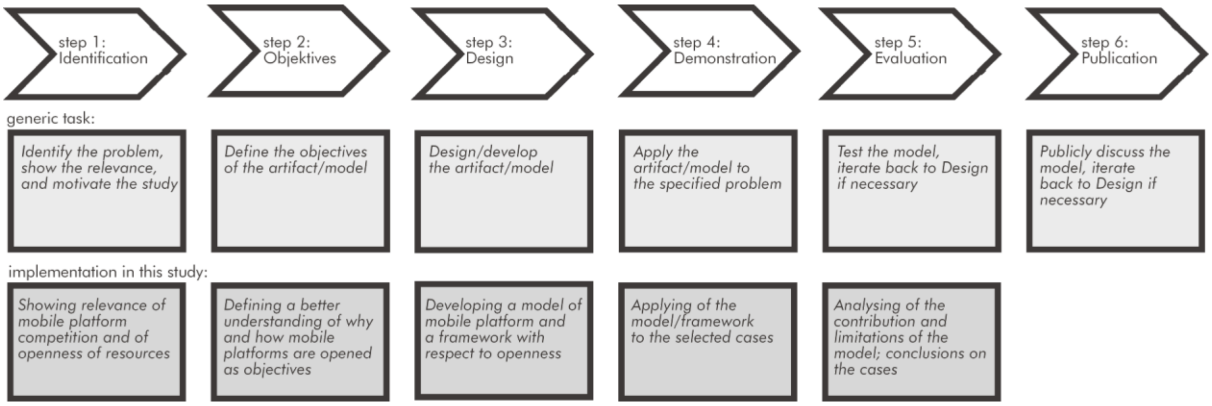

Figure 1. Research design of the study

In this first section (Identification), we introduce the topic, give background information, specify the research problem and question, and explain the manner of proceeding. In the second section (Objectives), we define the aim of a framework for the analysis of openness. In the third section (Design), we develop a matrix framework of access to, and control of, information resources, and a model of mobile platforms that illustrates the objects of our analysis. In the fourth section (Demonstration), we apply the framework to the cases of Symbian, Apple, and Google. We concentrate on the five main elements of the mobile platform model and show differences in the degrees of openness. In the fifth section (Evaluation), we conclude with a discussion. The final step (Publication) of the design science method is defined as the release and discussion of the research, which is realized through public presentation.

2 OBJECTIVES

The objective of this study is to better understand the differences in platform openness in general, and of mobile platforms in particular. To allow for this, we need a framework that indicates the different forms of openness.

An information resource consists of information that is economically relevant and repeatedly applicable (further discussion below). A common definition of innovative *value creation* describes it as a new and valuable combination of resources (including tacit information resources) and know-how (such as ideas and solution knowledge). The potential for finding a valuable combination of information resources (e.g. an operation system) with know-how increases exponentially with the number of external parties having access to these information resources. Opening a platform will support this by allowing external parties to combine their resources with internal information resources to create valuable elements (e.g. applications). The perceived value of the platform is thereby enhanced for the end-user, which might in turn benefit the platform initiator. Since mobile platforms and devices have some of the characteristics of net products (Shapiro and Varian 1999), it is crucial to rapidly achieve a certain stock of adopters (both end-users and application developers) in order to profit from network effects that create positive feedback (Katz and Shapiro 1985, Shapiro and Varian 1999, Arthur 1996). While third-party willingness to contribute increases with the openness of a platform, the ability of the platform initiator to benefit from these contributions will be the lower, the higher the degree of openness, as this constitutes a regime of weak appropriability in the sense of Teece (1986). To solve this contrast, companies facing a platform competition open up information resources in different ways.

Levintan (1982) uses the term *information resource* for information that can be used inter-subjectively. Others consider information resources to be both intangible information and tangible processing information systems (Information Resources Policy Project 1977). For the following, we define information resources as an economically relevant set of interconnected information that constitutes a delimitable intangible element of a system. In this sense, an information resource (e.g. a specific application) can consist of other information resources (e.g. set of content data).

How information resources that comprise a mobile operating system are used and created differs greatly. We identify two dimensions of differences: Firstly, who can *access*, use, and contribute to the information resource? Secondly, who is in *control* of the information resource and the terms of access to it? Different information resources may vary in both access and control. We can use these two dimensions to create matrix framework to describe the openness of information resource. Additionally, we need to identify the elements of mobile platforms that can be open (or closed). Therefore, in order to examine information resources of mobile platforms, we need a model that should be able to (1.) describe the means of access to these information resources, (2.) describe the control of these information resources, and (3.) define the elements that constitute information resources. The first and second objectives are of a general kind and a generic model seems to be appropriate, as the dimension of access to and control of information resources are valid categories for any given set of information resources. The third objective demands for a problem-specific model, as elements that constitute information resources of mobile platforms might not be present or relevant in other contexts.

3 DESIGN

We take the perspective of the platform leader, the initiating firm, in the following section. To differentiate openness of information resources, we create a matrix with the dimensions *access to resource* and *control of resource*. The access dimension describes which parties are having access to the information resources. The values are *open*, *group*, and *exclusive*. The control dimension describes the ability of the firm, in this case the platform leader, to control access to information resources. The values of this dimension are *internal*, *shared*, and *external*. This induces a nine-field matrix framework for structuring types of openness as shown in Figure 2 and described below.

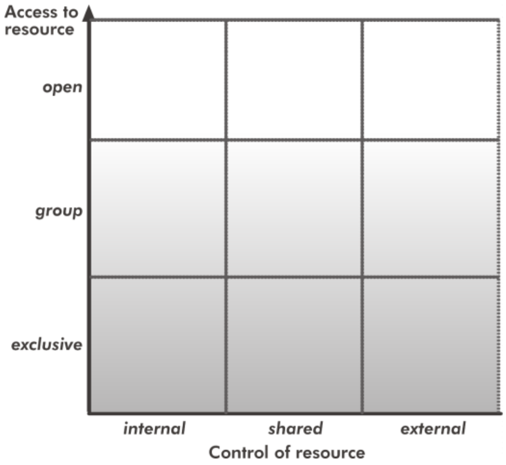

Figure 2. Matrix for structuring types of openness

Access to resource describes the circle of access beneficiaries to an information resource and therefore to whom the information resource is open. An information resource in *exclusive* access of a company cannot be used by any third party. This can be achieved through technical and legal protection mechanisms, such as intellectual property rights (Maskus 2000). Information resources of this kind can be used exclusively by the owner to create added value and to achieve sustainable competitive advantages (Barney 1991). On the other hand, ideas and knowledge of any entity other than that of the firm itself are excluded, which may prevent useful innovation to be created or developed to marketability (Chesbrough 2003). If information resource access is given to a *group* (a limited number of partners), more resource combinations are allowed for. Actors can build a collective resource pool as a basis for cooperative added value processes (Kogut and Zander 1992, Khanna et al. 1998, Inkpen 2000, Specht et al. 2002). *Open* access, which makes resource combinations even with an open sphere possible, establishes the highest added value potential for any given information resource (Arrow 1962). Opening resource allows for open value creation (Schlagwein et al. 2010), but may imply a weaker appropriability regime (Teece, 1986).

Control of resource describes the way the access is controlled, that is the property right of the information resource. Opening an information resource does not necessarily result in the surrender of control. For example, Google Maps data can be used by anyone (open access), but control stays fully with Google. Google could make the service unavailable or change usage terms at any time. *Internal* control of information resources allows the company to fully define the characteristics of access. *External* control implies a situation where another company is able to define the characteristics of access. In the case of *shared* control, a group, e.g. a consortium, defines the characteristics of access to the information resource. Proprietary control might discourage third-parties from investments because of their fear of dependencies and power asymmetries. Therefore, it may be beneficial for firms to shift control of certain information resource to a consortium. Shared control can be realised, e.g. through licenses, as in case of open source software (Osterloh and Rota 2007).

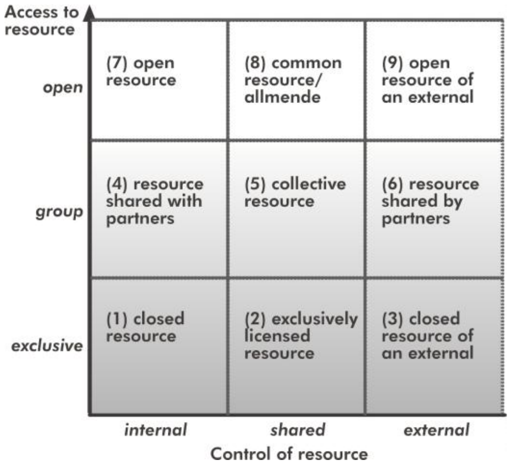

Figure 3. Types of openness

There are nine types of openness, depicted in Figure 3, which result in the combination of access and the control dimension. Internal controlled information resources are proprietary, protected, *closed resources*, which allow the company to gain all of the value creation that can be attributed to these resources. This type of resource, and its key role for value creation and competitive advantages, is discussed in the literature on the Resource-based View (Wernerfelt 1984, Barney 1991). Another revenue source can stem from external technology exploitation (Ford and Ryan 1981, Lichtenthaler 2007), that is selling own resources are an external. *Closed resources of an external* might vice versa be bought to fill resource deficits (Brockhoff 1995, Pisano 1990). An *exclusively licensed resource* might be solely used by the firm, but control remains shared (e.g. agreed veto rights).

In collaborations between companies – technology alliances or joint ventures – the participating companies make certain information resources, e.g. intellectual property, available for group access. If the control of the resources is shared, this creates a pool of *collective resources* (Specht et al. 2002). A *resource shared with partners* is made available without giving up control, i.e. property rights. This happens if a company wants to protect itself from one-sided saturation (Hamel et al. 1989), and therefore protects special resources (Kale and Singh 2000). Conversely, for as *resource shared by partners*, a cooperating firm can make certain information resources available to access for its partners but keep the control over it.

The framework also covers information resources with open access for a possibly unlimited number of users and contributors. Opening an information resource for public access does not necessarily lead to a *commons-based resource* (Benkler 2006), in which case control is waived. Open access to information might be granted, but control might still rest with a certain company, creating the final two fields of the matrix: *open resource* (controlled by the firm) and *open resources of an external*. Finally, we are equipped with a generic framework to describe the different ways in which information resources can be open.

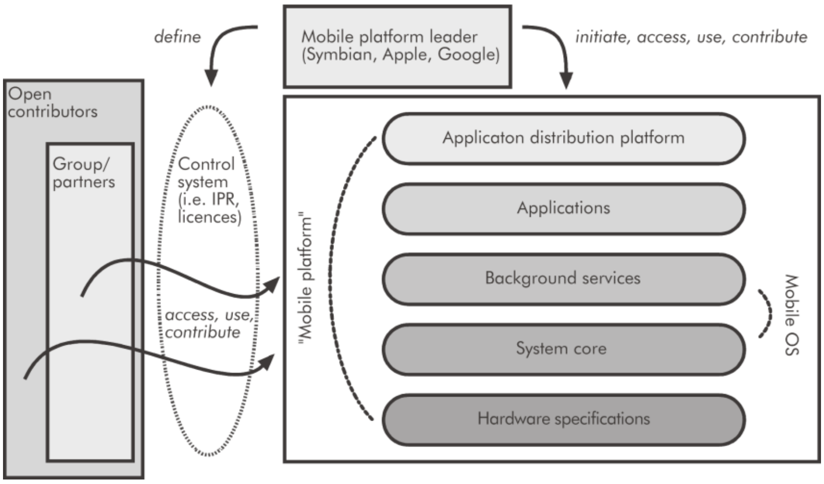

Figure 4. Model of a mobile platform

Figure 4 shows a model with five different layers of information resources, each of which constitutes the mobile platform. The information resources that we use for our analysis are the application distribution platform, the application development platform, the background services, the (operating) system core, and the hardware specifications of the mobile platform.

The *hardware specifications* describe the hardware and its characteristics. Depending on the openness of these specifications, the possibility of building compatible hardware can be limited. The *system core* and the *background systems* combined define the mobile operating system. The system core, or kernel, is the deepest level of an operating system. The kernel of an operating system can directly interact with the hardware through hardware interfaces and IRQs (interrupt request). Background services are applications that directly work on a system level. From a user perspective, such applications do no need to be actively started, but rather appear as integrated parts of the system functionality. An example is an alarm clock function, which can wake up the system from stand-by mode. Depending on the legal regime, contributors can access or even develop the operating system's core and its background services. The fourth resource is the *application development platform*. Applications are usually developed using an SDK (software development kit), another part of a mobile platform. Applications that are not pre-installed on the handset may later be obtained from an *application distribution platform*, the final resource we consider.

4 APPLICATION

In this section, we apply the framework to the mobile platforms example in the form of a micro-case-analysis of Symbian, Apple, and Google by following the guiding principles of Eisenhardt (1989) and Yin (1994). The elements to be examined with the matrix framework are the five identified information resources of mobile platforms, as outlined in the previous section.

4.1 Hardware specification

Hardware specifications can restrict a mobile operating system to a specific device or manufacturer, constituting an open or closed system. Figure 5 depicts the handling of the hardware specification resource by the three contrasted mobile platforms (A = Apple, G = Google, S = Symbian).

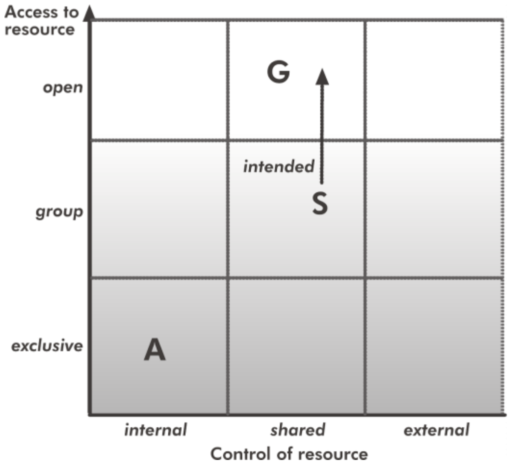

Figure 5. Openness of the hardware specifications

Of the three firms analyzed, Symbian, established as a joint venture, has been present in the mobile industry for the longest time. Symbian gained significant revenue through licensing fees (Suarez et al. 2009). Symbian OS has never been limited to a specific hardware device or a specific manufacturer. Hardware interface specifications are known, so that manufactures are able to produce cell phones compatible with the Symbian OS. The interface specifications are controlled by Symbian.

Apple keeps its mobile operating system closed (Hamilton 2009). The hardware specifications of the iPhone are not available to third parties. The access and the control are both purely internal, which makes Apple the single manufacturer for devices that can run iPhone OS and, vice versa, makes iPhone OS the only operating system for the iPhone.

Google initial approach was to only offer the Android operating system and not to become a hardware manufacturer. Recently however, on January 5, 2010, the first “Google smartphone”, the Nexus One, has been released in cooperation with the Taiwanese smartphone manufacturer HTC (Google 2010). Still, Android is an open system and an open source system with its published under an OSS licence. Compatible phones could only be built after the release of Android’s hardware specifications (Garfinkel 2008). There was no Android-based phone available for end-user at the time the Android operating system was introduced. The access to the interface information is open, which allows manufacturer to use Android by integrating it on their smartphones (Palenchar 2007). Google purposefully shifted the control of Android’s hardware specification to a shared arrangement, in form of the Open Handset Alliance (OHA), to encourage hardware manufactures to invest in compatible devices.

4.2 System Core

The system core is the heart of an operating system. It is the system’s centre, and provides interfaces to hardware (IRQs), applications (application programming interfaces, APIs), and users (user interfaces, UIs). The information resource is defined by the access and control of the system core’s source code. Figure 6 depicts the handling of the system core resource by the three contrasted mobile platforms.

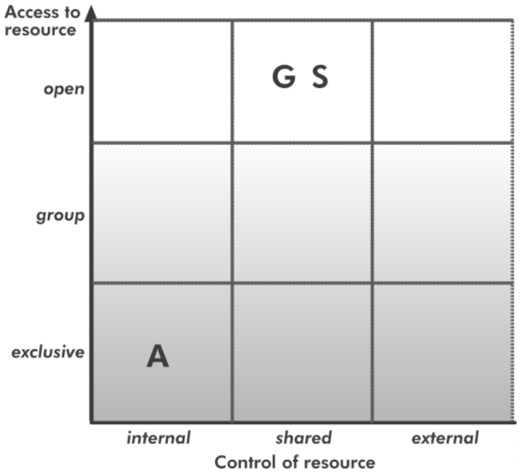

Figure 6. Openness of the system core

Symbian uses a common operating system core called EPOC Kernel Architecture 2 (EKA2) (Morris 2007). There is no official documentation about the system core available from Symbian itself, but a set of books and relevant documentation from third parties teaches interested developers interested how to program the Symbian system core.

In Apple’s case, the iPhone is based upon a variant of the same Darwin operating system core that can be found in Mac OS X. The main difference to the Mac OS X system is that the iPhone OS system core is closed for external access. Darwin is released under an Apple Public Source License (APSL) that requires code created on its base to be published. Still, Apple, as the copyright holder, does not have to abide by the APSL. User extensions to OS X are published under a Berkeley Software Distribution License (BSDL), allowing for open and proprietary re-use. Consequently, Apple was able to create a closed, mobile “OS X”. Apple has exclusive access and internal control of system core.

Android is built upon a Linux-based kernel and all necessary development information, development tools, and the source code can be downloaded for free on Android’s website. Google releases the Android kernel under Apache Software License (ASL), which allows non-proprietary and proprietary re-use (Constantinou 2007). Google can change parts of the system core source code, as can any other members of the OHA. As a result, the control of the resource is shared and the resource is openly accessible.

4.3 Background services

A background service is a function of the mobile operating system that continues to run while another application is actively used in the foreground. Some applications are specifically written to run as background activities (similar to parallel running “tasks” in Windows). The information resource is the availability of the function itself, which can be open (allowed) or closed. In Figure 7 we see an overview.

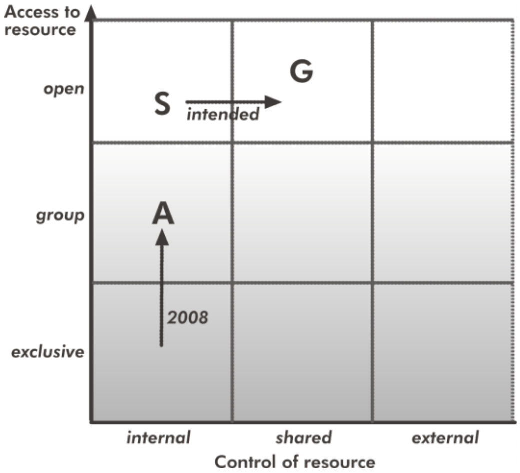

Figure 7. Openness of the background services

Symbian OS allows background services. Therefore, to let multiple applications run at the same time is possible, provided the application itself supports this (Abramovich 2006). Decisions regarding access to background functionality are controlled by Symbian. When the Symbian OS has fully changed to an open source system, the control will move to the shared control. The access to this function is not restricted: In the Symbian platform design, it is possible for any third-party application to run as a background service.

Apple does not allow background services. Instead, a "push service" has been provided in the recent iPhone OS, which allows data to be sent from selected developers' servers to Apple and therewith to the iPhone. Still, this is not comparable with the full functionality of a background service (Marsal 2008). Before the implementation of the push service, Apple completely prohibited the background service functionality for third parties and thereby excluded possible applications for the iPhone (Meier 2008). Apple itself can run real background services on the iPhone (iPod, Mail, and other proprietary functions are running in the background), but this is (officially) not possible for non-affiliated third-party applications. User must modify – "jailbreak" – their iPhones to enable backgrounding for third-party applications.

The Android system allows background services (Meier 2008). The Android operating system provides a full multitasking platform. The access to this service is open. Yet, the control over the background service does not reside exclusive at the OHA. Whether a background service is allowed or not can be changed by the hardware manufacturers that use and adapt Android for their phones. In this case the control would, from Google's perspective, shift from shared to external control (to the right side of the matrix). Such a modification is theoretically possible, but at present is not implemented in an Android-based smartphone by any manufacturer.

4.4 Application development platform

In order to analyze access and control of creating applications for the mobile platform, we need to consider resources that can be used to build applications with, i.e. the software development kits (SDKs). Based on these information resources, developers can create derived information resources, i.e. new applications and games. How the SDKs are accessed and controlled is depicted in Figure 8.

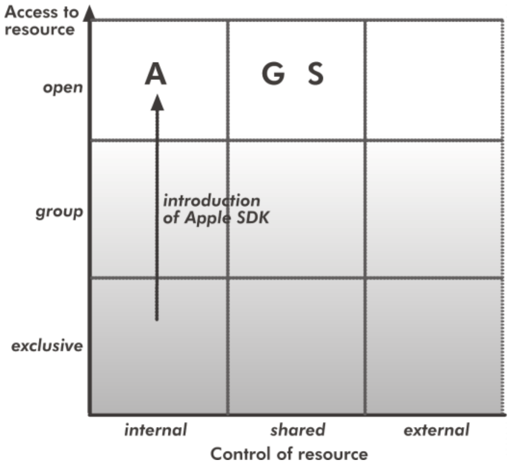

Figure 8 Openness of the application development platform

For Symbian, many different SDKs exist in different programming languages, C++ being the most common. Some SDKs are freely available, while more advanced Integrated Development Environments (IDE), e.g. from CodeWarrior (Freescale 2009), are commercial products. An IDE contains more than an SDK e.g. additional building automation tools or debugging tools (Nourie 2005). Since December 2008, the formerly commercial (Bustarret 2008) Caride.c++ IDE is available for free following registration at the Forum Nokia (Nokia 2009b). The control over competitive SDKs, i.e. the creation of applications, is shared. Access to the SDKs is open.

Since March 2008 Apple provides the Apple SDK for the iPhone (Block 2008; Hendrickson 2008). With the Apple SDK, third-party developers are able to develop applications for the iPhone. The Apple SDK provides a programming environment framework and tools for development. There is no other official SDK than that of Apple. Control stays fully with Apple, but access to the SDK, allowing (front-end) application development, is now open to everyone.

For Android, a SDK with all necessary information and documentation to set up the system and the SDK itself was provided right from the start (Android developers 2009a). Development of the SDK was shifted from Google to the OHA. The 50 OHA members share control over this information resource. The SDK is freely available for interested developers on Android’s website to provide open access.

4.5 Application distribution platform

Application distribution platforms are built to support the mobile platform by distributing applications to the end-user of the system. In Figure 9 the openness of the application distribution platform is depicted for Symbian, Apple, and Google.

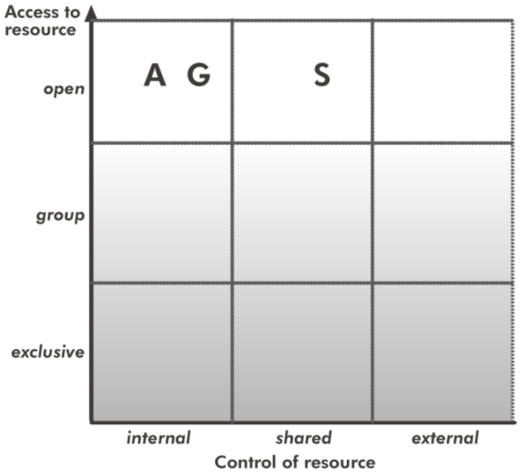

Figure 9. Openness of the application distribution platform

Symbian applications have to be signed and tested. Applications can be “self-signed” or “open-signed” (Symbian 2009b), but none of the dominating application distribution platforms, e.g. Symbian's Horizon (Symbian 2009c) and Nokia's Ovi (Nokia 2009c), will offer it in this case. A Symbian-signed account, in the form of express-signing (20.00 USD per signing) or certified-signing (200.00 USD per year), is necessary for official software releases (Symbian 2009d). Through the signing process access is open, developers have to register and pay a fee. The distribution itself through Symbian’s Horizon is free of charge. Depending on the manufacturer, other signing authorities exist, e.g. Nokia allows Symbian-signed and Java-verified software (Nokia 2008b). As Horizon is leading the market, but is not the only distribution platform, control of this resource can considered to be shared.

Apple requires a full registration process for anyone who wants to distribute software through the Apple’s App Store, the single application distribution platform for the iPhone. A registration fee of 99.00 USD has to be paid (Apple Inc. 2009a). Developers are bound the specific conditions of the Registered iPhone Developer Agreement (Apple Inc. 2008) set up by Apple. After successful development of any application, an application check by Apple has to be passed. Even if all aspects – e.g. licences, functionality, hardware and software needs – meet the requirements, the publishing decision still depends on Apple (Hamilton 2009). Apple keeps a 30 % share of the revenue of application sold in the App Store. While access is open to any contributor, Apple controls the development process firmly.

Developers on the Android platform can easily create applications through the provision of the Android SDK and the corresponding documentation by Google (Meier 2007). After the application has been developed it has to be signed with a suitable private key (Android developers 2009b), the preparation instruction for publishing has to be followed (Android developers 2009c), and a registration fee of 25.00 USD is necessary to become an official Android Market Developer (Google Inc. 2009c). The Android Market Developer Distribution Agreement (Google Inc. 2009b) gives information about the rules for software distribution. It is intended to ensure that no violation against the rules will occur. In the Android Market, 30 % of the revenues are kept by the carrier. The Android Market is currently open to any third party. Yet, the terms of access are in control of Google.

5 DISCUSSION AND OUTLOOK

By applying the framework to the mobile platforms of Symbian, Apple, and Google, we discern differences in the strategies of the firms with respect to how they handle access and control of information resources. These differences can be described in detail with the aid of the matrix framework and the mobile platform model. The different approaches are related to the business strategies of the firms. While the underlying strategies and intentions of the firms to set up mobile platforms are obviously nor publically revealed, some conclusions may be drawn.

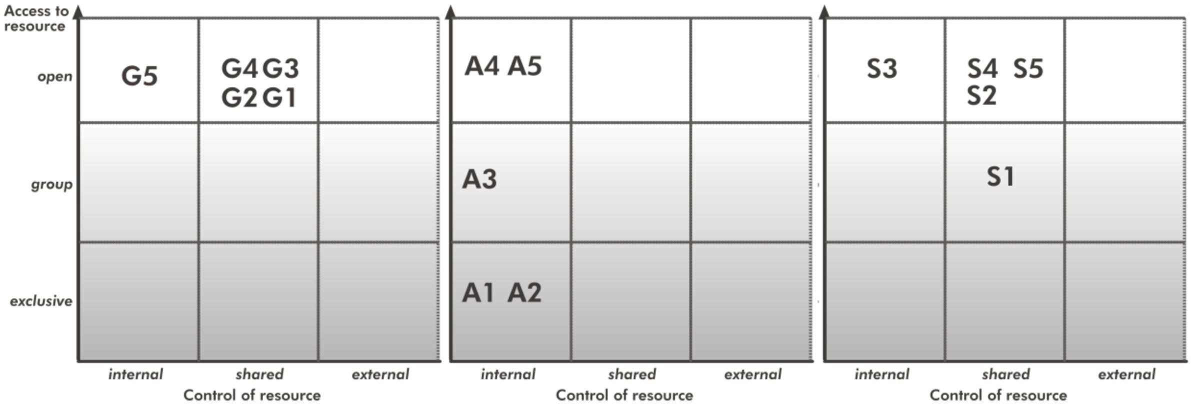

Figure 10: Mobile platforms and their use of openness

Symbian has long been the market leader in mobile operating systems. Since its acquisition by Nokia, the Symbian system has in many respects been opened (see Figure 10, left). Nokia is primarily a hardware manufacturer. Nokia sees its position on the hardware market being attacked by Apple. Even though Nokia still leads the cell phone market, the smartphone sector demands for new systems with enhanced abilities and might turn against Nokia, if it is not able to provide such operating systems. Nokia seems to have acquired Symbian and transformed its proprietary system to an open source system (Boulton 2008) in order to create a mobile platform that can stand up to competition from especially the iPhone (OS). Therefore, contributions on all levels are encouraged by very open rules. Dominance of the Symbian mobile platform secures established shares in the mobile device market. Therefore, Nokia was able to align old competitors to fight the new entrant(s). Yet, the December 19, 2009 release of the N900 smartphone, equipped with the Linux-based Mameo OS – instead of Symbian OS (Nokia 2010) – hints for coming changes in Nokia's strategy.

Apple has strong control over the system and its resources (see Figure 10, middle). Apple sells the iPhone platform as a whole, with the hardware and the operating system being inseparable. Apple is originally a computer manufacturer, where it protected the market share of its proprietary system successfully against rivalrous open systems like Windows (open in terms of hardware compatibility) and Unix/Linux – even those had strong network effects through complementary software. After having successfully established itself as a music player manufacturer (iPod) and a content provider (iTunes Store) recently, the entrance into the smartphone sector seems be a logical consequence of the progressing convergence between different media devices (a smartphone is a substitute to an iPod). With the first iPhone version, Apple tried to earn sales and exclusive contracts with mobile network providers, leveraging its highly valued brand name. This approach was complemented by the opening to third-party contributors with introduction of the second version of the iPhone (Reuters 2008, Ritchie 2008). This move enables Apple to provide a more attractive platform to costumers – and earn a significant share of revenue on every sold application sold for the iPhone (Apple Inc. 2009b). The comparatively closed single device approach of the iPhone benefits Apples, as it ensures stability of hardware and software combinations. Additionally, it is easier for Apple to leverage it image by marketing a single Apple-branded device.

Google comes from a third industry as is primarily a web-software and web-marketing provider. With the rise of the mobile internet, it naturally comes to Google's core business interest to prevent mobile platform providers from shaping the user's mobile web experience at their will (e.g. the "search" functionality). Therefore, Google entered a completely new market field by creating a mobile platform itself. As Google's business does not depend on software or hardware sales, it may pay off to give away significant information resource investments into shared control. If Android is to become the dominant mobile platform, Google's core business interest is not at risk in the mobile world, which avoids opportunity costs (through lost user attention, i.e. lost revenues). On the contrary, larger revenues through more specialised advertisement possibilities might arise. To attract contributors and users, the conditions of access and control are as open and liberal as possible (see Figure 10, right).

While this analysis contributes to the understanding of how openness is used in the fighting for mobile platform dominance, this may not be the single crucial factor in explaining and predicting platform success. There are more research and practise questions to be asked: How determine other resources (e.g. physical resources such as production capacities) the design of a mobile platforms? How can the competition be modelled in dynamic way, as a set of actions and reactions? How important are brand names and current market positions? These are only some of the questions regarding analysis and prediction of this dynamic and complex sector that demand further research.

References

Abramovich, A. (2006). Symbian OS System Libraries: Start-on-boot Registration API. http://developer.symbian.com/wiki/display/pub/Symbian+OS+system+libraries (2009-09-14)

Alliance, O. H. (2007). Industry Leaders Announce Open Platform for Mobile Devices. http://www.openhandsetalliance.com/press_110507.html (2009-09-10)

Apache Software Foundation (2009). Licenses. http://www.apache.org/licenses/ (2009-09-10)

Android developers (2009a). Download Android 1.0 SDK, Release 2. http://developer.android.com/sdk/1.0_r2/index.html (2009-09-10)

Android developers (2009b). Signing Your Applications. http://developer.android.com/guide/publishing/app-signing.html (2009-09-08)

Android developers (2009c). Preparing to Publish: A Checklist. http://developer.android.com/guide/publishing/preparing.html (2009-09-08)

Apple (2008). Registered iPhone Developer Agreement. http://developer.apple.com/iphone/terms/registered_iphone_developer.pdf (2009-09-08)

Apple (2009a). iPhone Developer Program. http://developer.apple.com/iphone/program/ (2009-09-08)

Apple (2009b). Distribute Your Application. http://developer.apple.com/iPhone/program/distribute.html (2009-09-10)

Arrow, K. J. (1962). Economic Welfare and the Allocation of Resources for Invention. In: Nelson, R. (Ed.) (1962). The Rate and Direction of Inventive Activity: Economic and Social Factors. 1st Ed. UMI, Princeton.

Arthur, W. B. (1996). Increasing Returns and the new World of Business. Harvard Business Review, 74 (4), pp. 100-190

Ballagas, R., Borchers, J., Rohs, M., and Sheridan, G. J. (2006). The Smart Phone: A Ubiquitous Input Device. IEEE Pervasive Computing, 5 (1), pp. 70-77.

Barney, J. B. (1991). Firm Resources and Sustained Competitive Advantage. Journal of Management, 17 (1), pp. 99-120.

Benkler, Y. (2006). The Wealth of Networks. 1st Ed. Yale University Press, New Haven.

Block, R. (2008). Live from Apple's iPhone SDK Press Conference. http://www.engadget.com/2008/03/06/live-from-apples-iphone-press-conference/ (2009-09-06)

Boulton, C. (2008). Nokia Goes on the Mobile Offensive. eWeek, 25 (20), pp. 11.

Brockhoff, K. (1995). Zur Theorie des externen Erwerbs neuen technologischen Wissens. Zeitschrift für Betriebswirtschaft, special issue, pp. 27-42.

Bustarret, E. (2008). Carbide.c++ 2.0 Available and Free. http://www.newlc.com/en/carbidec-goes-free (2009-09-07)

Chau, P. Y. K. and Tam, K.Y. (1997). Factors Affecting the Adoption of Open Systems: An Exploratory Study. MIS Quarterly, 21 (1), pp. 1-24.

Chesbrough, H. W. (2003). Open Innovation. 1st Ed. Harvard Business School Press, Boston.

Constantinou, A. (2007). The Significance of Google’s Android. http://visionmobile.com/blog/2007/11/the-significance-of-googles-android/print/ (2007-12-06)

Dowling, S. and Barney, A. (2007). iPhone Premieres This Friday Night at Apple Retail Stores. http://www.apple.com/pr/library/2007/06/28iphone.html (2009-09-10)
Eclipse Foundation (2009). Eclipse Public License. http://www.eclipse.org/org/documents/epl-v10.php (2009-09-10)
Economist (2008). The Meek Shall Inherit the Web. The Economist, 388 (September 4, 2008), pp. 3-4.
Economist (2010). Nokia Reinvents Itself. The Economist 394 (January 9, 2010), pp. 56-57.
Eisenhardt, K. M. (1989). Building Theories from Case Study Research. Academy of Management Review, 14 (4), pp. 532-550.
Ford, D., Ryan, C. (1981). Taking Technology to Market, Harvard Business Review, 59 (2), pp. 117-126.
Free Software Foundation (2009). GNU General Public License. http://www.gnu.org/copyleft/gpl.html (2009-09-10)
Freescale Semiconductor (2009). CodeWarrior Development Tools. http://www.freescale.com/webapp/sps/site/homepage.jsp?nodeId=012726 (2009-09-07)
Garfinkel, S. (2008). Android Calling. Technology Review, 111 (2), pp. 88-90.
Google (2009a). Register as a Developer. http://market.android.com/publish/signup (2009-09-11)
Google (2009b). Android Market Developer Distribution Agreement. http://www.android.com/us/developer-distribution-agreement.html (2009-09-09)
Google (2010). Nexus One Phone. http://www.google.com/phone/static/en_US-nexusone_tech_specs.html (2009-01-10)
Hamel, G. D., Yves L., and Prahalad, C. K. (1989). Collaborate with Your Competitors – and Win. Harvard Business Review, 67 (1), pp. 136-137.
Hamilton, I. (2009). Thoughts on the App Store. http://thoughtsontheappstore.com/ (2009-09-08)
Hendrickson, M. (2008). iPhone 2.0: Enterprise Ready. Developer Ready. http://www.techcrunch.com/2008/03/06/iphone-20-enterprise-ready-developer-ready/ (2008-03-07)
Hevner, A. R., March S. T., Park, J., and Ram S. (2004). Design Science in Information Systems Research. MIS Quarterly, 28 (1), pp. 75-105.
Inkpen, A. C. (2000). A Note on the Dynamics of Learning Alliances: Competition, Cooperation, and Relative Scope. Strategic Management Journal, 21 (7), pp. 775-779.
Information Resources Policy Projects (1977). Information Resources Policy Projects Annual Report V. 2 1976-1977.
Kale, P., Singh, H., and Perlmutter, H. (2000). Learning and Protection of Proprietary Assets in Strategic Alliances: Building Relational Capital. Strategic Management Journal, 21 (3), pp. 217-237.
Katz, M. L. and Shapiro, C. (1985). Network Externalities, Competition and Compatibility. American Economic Review, 75 (3), pp. 424-440.
Khanna, T., Gulati, R., and Nohria, N. (1998). The Dynamics of Learning Alliances: Competition, Cooperation, and Relative Scope. Strategic Management Journal, 19 (3), pp. 193-210.
Kogut, B. and Zander, U. (1992). Knowledge of the Firm, Combinative Capabilities, and Replication of Technologies, Organization Science, 3 (3), pp. 389.
Levitan, K. B. (1982). Information Resources as 'Goods' in the Life Cycle of Information Production. Journal of the American Society for Information Science, 33 (1), pp. 44.
Lichtenthaler, U. (2007) Open for Innovation: Understanding the Determinants of External Technology Commercialization. Zeitschrift für Betriebswirtschaft, special issue, pp. 21-45.
March, S. T., Smith, G. F. (1995). Design and Natural Science Research on Information Technology. Decision Support Systems, 15 (4), 251-266.
Marsal, K. (2008). Apple Tackles iPhone Background App Limits with Push Service. http://www.appleinsider.com/articles/08/06/09/apple_tackles_iphone_background_app_limits_with_push_service.html (2009-09-08)
Maskus, K. E. (2000). Intellectual Property Rights in the Global Economy. Peterson Institute for International Economics study.
Meier, R. (2007). How to Program Google Android. http://blogoscoped.com/archive/2007-11-19-n27.html (2009-09-08)

Meier, R. (2008). Android v iPhone: SDK Showdown. http://blog.radioactiveyak.com/2008/03/android-v-iphone-sdk-showdown.html (2008-04-24)
Morris, B. (2007). The Symbian OS Architecture Sourcebook: Design and Evolution of a Mobile Phone OS. 1st edition. Jon Wiley & Sons, Chichester.
Nokia (2008a). Nokia to Acquire Symbian Limited to Enable Evolution of the Leading Open Mobile Platform. http://www.nokia.com/press/press-releases/showpressrelease?newsid=1230415 (2009-09-10)
Nokia (2008b). Application Signing. http://wiki.forum.nokia.com/index.php/Application_Signing (2009-09-09)
Nokia (2009a). Your Life. Connected. Ovi by Nokia. http://www.ovi.com/services/ (2009-09-09)
Nokia (2009b). Driving Innovation for Mobile Application Developers. http://www.forum.nokia.com/ (2009-09-07)
Nokia (2010). Nokia N900 Mobile Computer, Technical Specifications. http://maemo.nokia.com/n900/specifications/ (2010-01-10)
Nourie, D. (2005). Getting Started with an Integrated Development Environment (IDE). http://java.sun.com/developer/technicalArticles/tools/intro.html (2009-09-07)
Open Software Initiative (2009). The Open Source Definition. http://opensource.org/docs/osd (2009-09-10)
Osterloh, M. and Rota, S. (2007). Open Source Software Development - Just Another Case of Collective Invention? Research Policy, 36 (2), pp. 157-171.
Palenchar, J. (2007). Android to Set Wireless Markets Free: Supporters. This Week in Consumer Electronics, 22 (24), pp. 78-81.
Pantoni, R. P., Brandão, D. (2009). Developing and Implementing an Open and Non-proprietary Device Description for FOUNDATION Fieldbus Based on Software Standards. Computer Standards & Interfaces, 31 (2), pp. 504-514.
Peffers, K., Tuunanen, T., Rothenberger, M. A., Chatterjee, S. (2007). A Design Science Research Methodology for Information Systems Research. Journal of Management Information Systems, 24 (3), pp. 45-78.
Pisano, G. P. (1990). The R&D Boundaries of the Firm: An Empirical Analysis. Administrative Science Quarterly, 35 (1), pp. 153-176.
Reuters (2008). Apple Changes iPhone Business Model. http://www.smh.com.au/news/biztech/apple-changes-iphone-business-model/2008/06/10/1212863586054.html (2009-0910)
Ritchie, R. (2008). Apple's New iPhone Business Models. http://www.theiphoneblog.com/2008/05/28/apples-new-iphone-business-models/ (2009-09-10)
Schlagwein D., Schoder D., Fischbach K. (2010). Open Value Creation. Proceedings of the MKWI 2010.
Shapiro, C. and Varian, H. R. (1999). Information Rules - A Strategic Guide to the Network Economy. Harvard Business School Press, Boston.
Specht, G., Beckmann, C., and Amelingmeyer, J. (2002). F&E-Management. 2nd edition, Schäffer-Poeschel, Stuttgart.
Suarez, F. F., Edelmann, B., and Srinivasan, A. (2009). Symbian, Google, & Apple in the Mobile Space. Harvard Business School case study.
Symbian (2009a). Symbian Signed Grid. http://developer.symbian.com/main/downloads/files/Symbian_Signed_Grid.pdf (2009-09-09)
Symbian (2009b). Open Signed Online. https://www.symbiansigned.com/app/page/public/openSignedOnline.do (2009-09-09)
Symbian (2009c). Symbian Horizon. http://developer.symbian.org/main/horizon/ (2009-09-09)
Symbian (2009b). The Symbian Foundation. http://www.symbian.org/about/ (2009-09-10)
Teece, D. J. (1986). Profiting from Technological Innovation, Research Policy, 15 (6), pp. 285-305.
Wernerfelt, B. (1984). A Resource-based View of the Firm. Strategic Management Journal, 5 (2), pp. 171-180.
Yin, R. K. (1994). Case Study Research: Design and Method.1st Edition. Sage Publications, Thousand Oaks.