EVOLVING AN OPEN ECOSYSTEM: THE RISE AND FALL OF THE SYMBIAN PLATFORM

Joel West and David Wood

ABSTRACT

Two key factors in the success of general-purpose computing platforms are the creation of a technical standards architecture and managing an ecosystem of third-party suppliers of complementary products. Here, we examine Symbian Ltd., a startup firm that developed a strong technical architecture and broad range of third-party complements with its Symbian OS for smartphones. Symbian was shipped in nearly 450 million mobile phones from 2000 to 2010, making it the most popular smartphone platform during that period. However, its technical and market control of the platform were limited by its customers, particularly Nokia. From 2007 onward, Symbian lost market share and developer loyalty to the new iPhone and Android platforms, leading to the extinction of the company and eventually its platform. Together, this suggests lessons for the evolution of a complex ecosystem, and the impact of asymmetric dependencies and divided leadership upon ecosystem success.

Keywords: Platform control; ecosystems; mobile phones; ecosystems; complementary assets

Collaboration and Competition in Business Ecosystems
Advances in Strategic Management, Volume 30, 27–67
Copyright © 2013 by Emerald Group Publishing Limited
All rights of reproduction in any form reserved
ISSN: 0742-3322/doi:10.1108/S0742-3322(2013)0000030005

MDT Exhibit 2037
IPR2025-00544
Page 1

INTRODUCTION

Dynamics of Platform Competition

For nearly 30 years, researchers have been interested in the sort of *de facto* standards battles that are common in consumer electronics, computing, and communications. The early research by Katz and Shapiro (1985) and others established a positive-feedback network effects model mediated by the supply of specialized complementary assets (see Gallagher & West, 2009, for a recent summary). Meanwhile, investments in such assets created switching costs that together with network effects often made insurmountable an early lead gained in a standards contest (Arthur, 1996; Farrell & Klemperer, 2007).

From this, researchers have identified the dynamics of complex architectures of standardized components termed *platforms* (Bresnahan & Greenstein, 1999; Eisenmann, 2007; Gawer & Cusumano, 2002; Morris & Ferguson, 1993). Here, we focus on computing platforms as defined by Bresnahan and Greenstein (1999, p. 4): "as a bundle of standard components around which buyers and sellers coordinate efforts," rather than definitions that might include as a platform the world wide web, or specific applications such as a web browser or Microsoft Office.

One key to platform success is a technical architecture of standards that both facilitates complementary assets and allows re-use between vendors and product generations (Bresnahan & Greenstein, 1999; Gabel, 1987; West & Dedrick, 2000). A successful technical architecture allows modular innovation by both the platform sponsor and by third party complementors (Baldwin & Woodard, 2010). Firms that control the interfaces of such architecture – usually through application programming interfaces (APIs) – can control of the supply of complements and thus the allocation of profits that accrue to the platform (West & Dedrick, 2000).

Another key antecedent of platform success is courting and maintaining a vibrant supply of third party complements ("software") that makes a product ("hardware") more valuable (Katz & Shapiro, 1985). While early research argued that a popular standard with a large installed base would automatically attract such a supply of complements, moderators of the positive-feedback process mean that standards sponsors make technical, product, and economic choices that make a standard more or less attractive to complementors (Gallagher & West, 2009). The platform sponsor must share returns of platform success with complementors to assure an ongoing supply of complements (Gawer & Cusumano, 2002). The interdependence

of the sponsor with its complementors creates an ecosystem (Adner & Kapoor, 2010). Although a sponsor should be able to capture "outsized returns" once an ecosystem is established (Adner, 2012, p. 117), excessive value capture by the sponsor threatens not only the survival of the complementors but also the entire ecosystem (Iansiti & Levien, 2004; Simcoe, 2006).

Smartphone Platform Competition

The personal computer (PC) platform of Microsoft Windows and Intel processors ("Wintel") is perhaps the most often cited example of platform success. Here, we examine the efforts by mobile phone producers to replicate the Wintel adoption success while avoiding the economic rents captured by Microsoft and Intel. With its partners, Symbian Ltd. created the smartphone category and enjoyed rapid success as it sponsored the most popular smartphone platform from 2002 to 2010, accounting for nearly 450 million smartphones sold worldwide during that period.

Symbian's initial strategy followed many of the key principles of platform leadership defined by Gawer (2010): technology design, strong relations with complementors, internal organization, and firm scope. Symbian built a technical architecture that was the first one optimized for smartphones, that is, cellular phones that were also programmable mobile computing devices. Symbian also built a successful ecosystem that enabled a wide range of devices from multiple manufacturers, and had the largest supply of third-party application software. Its internal organization was focused on developing and distributing an advanced smartphone operating system (OS). Finally, Symbian worked with its shareholder-customers – the world's five largest handset makers – to provide a scope that included firms representing 80% of the industry.

However, Symbian's success did not prevent its own extinction, nor that of its platform. The company faced successive competition from two rival mobile computing platforms – Apple's iPhone and Google's Android – that more closely emulated PC capabilities, created a new dominant design, and by late 2010 had captured a majority of the market (Kenney & Pon, 2011; West & Mace, 2010). This led to a series of desperate attempts at retrenchment as Symbian licensees abandoned its platform for Android. Finally, the sole remaining customer (Nokia) orphaned Symbian in favor of a smartphone derivative of the same Microsoft Windows quasi-monopoly it had long sought to avoid.

While the story of the iPhone and Android success may be familiar to contemporary readers, less well known is that the Symbian platform had developed elements of the dominant design years before the iPhone or Android. The first Symbian smartphone from Ericsson in 1999 had a point-and-click interface with a (for its day) spacious LCD screen, while starting in 2006, Nokia phones on the Symbian platform used the same WebKit desktop browser technology as later shipped with the iPhone and Android. Even less well known is that Symbian had discussed creating its own application store in 2005 – three years before the iPhone App Store – but abandoned the project due (in part) to lack of resources.

Here, we document the rise and fall of Symbian Ltd. and its Symbian OS platform. We use this to describe how the firm built a complex ecosystem of stakeholders, evolved this ecosystem over its 10-year lifespan, and how limitations in its conception and leadership of this ecosystem limited its ability to respond to the iPhone and Android threats.

Research Design

Our study uses a case study research design, a widely accepted way to understand and explain complex interorganizational relationships to develop theoretical insights (e.g., Eisenhardt, 1989). Throughout our study period, we compiled data regarding the firm's platform strategies from a wide range of primary and secondary sources. Primary data from Symbian included current information on its website, archived press releases dating back to 1998 that were published on the website, and previous information from the company website stored on the Internet Archive (Archive.org). We also referenced unpublished company memos and presentations, particularly around the evolution of the company's formal ecosystem program during each of its phases. We utilized shareholder reports listing full audited financial statements during those years (2003-2006) when they were made available to employee-shareholders.

We conducted interviews with current and former Symbian employees who managed aspects of its ecosystem strategy spanning the company's entire existence from June 1998 to November 2008.[1] We were also guided by participant observation by the second author, who was the only senior executive to span the company's entire lifespan and was directly involved in the second phase of the ecosystem program. Finally, we supplemented our data with secondary data on the company and its ecosystem. We drew upon news coverage, particularly in *The Register*, a UK-based IT news site, and

summaries of the company's history and strategy in books by Symbian authors.

From this, we develop insights regarding the trade-offs in managing a complex ecosystem, including the cognitive limits to ecosystem design. We focus on the unique form of divided leadership between Symbian and its partners, and the resulting ambiguity in both perceived and actual leadership in the platform that contributed to its eventual difficulties.

CREATING THE SMARTPHONE INDUSTRY

Symbian Ltd. was founded as a spinoff of another London-based company, Psion PLC, but was co-owned and funded by the world's largest handset makers (see Table 1 for key dates).[2] Psion was created in 1980 to develop PC application software, but soon shifted to developing a family of keyboard-based pocket computer systems: Organiser I (1984), Organiser II (1986), Series 3 (1991), and Series 5 (1997). These products became part of a product category called "personal digital assistants" (PDA), which also included the Sharp Zaurus (1993), the Palm Pilot (1996), and various "Handheld PC" models using Microsoft's Windows CE (1996).[3]

In 1996, Nokia announced the Nokia Communicator, the first PDA-type phone, built upon software licensed from Geoworks Inc. A series of PDA makers followed by licensing their software for use in similar phones. The Palm OS was incorporated in the Qualcomm pdQ (1998), followed by a series of Treo phones from Handspring and later Palm. Microsoft licensed its Windows CE to phone makers – including Samsung (1998), HP (2001), and Sagem (2001) – but could not gain distribution by US network operators until 2002.

To create its own PDA-hybrid phone, Psion Software held licensing discussions with the world's largest handset makers from 1996 to early 1998. In June 1998, Psion, Nokia, Ericsson, and Motorola announced that they would be joint owners of a new company, Symbian Ltd.[4] A key goal shared by Symbian and its owners was preventing Microsoft from extracting proprietary rents from mobile devices as it had in PCs, where it commoditized the systems vendors. As the executive of one of the initial vendors told the *Financial Times*, "We knew what had happened in the PC market and were determined not to let that happen in the mobile phone market" (Price, 1999). By aligning Symbian with the three (later five) largest handset makers, they also hoped to limit Microsoft's eventual market share. In response, a few months later, Microsoft's CEO Bill Gates termed the

Table 1. Key Dates for Symbian Ecosystem.

Date	Event
1994	Psion begins developing a 32-bit PDA operating system, later known as EPOC
1996	Nokia ships Series 9000 Communicator, its first PDA phone, using software licensed from Geoworks
June 1997	Psion ships Series 5 PDA based on EPOC operating system
June 1998	Symbian Ltd. founded in London by Psion PLC, Nokia Oy, and Ericsson AB
October 1998	Motorola, Inc. becomes Symbian Ltd. investor, acquiring same 23.1% stake as Nokia and Ericsson
March 1999	NTT agrees to license Symbian OS to support its FOMA 3G network
April 1999	Symbian acquires Ericsson's Mobile Application Lab, which later becomes UIQ Technology AB
June 1999	Symbian holds first conference in London for developers and other ecosystem members
February 2000	Symbian's second developer conference is held in Silicon Valley
August 2000	Psion announces plans (later cancelled) to spin off Symbian shares in public offering
September 2000	Ericsson ships R380, the first Symbian OS phone
June 2001	Nokia ships its first Symbian OS phone, Nokia Communicator 9210, with Series 80 UI
October 2001	Sony and Ericsson combine mobile phone divisions into UK-based joint venture
November 2001	Nokia announces plans to license Series 60 user interface to other firms
April 2002	Symbian launches Symbian Platinum Partner Program, a revised ecosystem relations program
June 2002	Nokia 9290 Communicator is first Symbian phone sold in the United States
June 2002	Nokia ships 7650, the first Series 60 phone
December 2002	Sony Ericsson ships P800 camera phone, first Symbian handset with UIQ interface
January 2003	Fujitsu ships FOMA F2051 to NTT DoCoMo customers, first Symbian-enabled MOAP(S) phone
October 2003	Motorola sells Symbian shares to Nokia and Psion
November 2003	NTT DoCoMo licenses Symbian OS for distribution by its phone suppliers
July 2004	Psion sells shares to Nokia, Sony Ericsson, Matsushita, and Siemens for £138 million
February 2005	Nokia ships the Nokia 7710, the only Series 90 handset ever commercially available
February 2005	Panasonic ships FOMA P901i, NTT DoCoMo's first MOAP(L) phone utilizing Linux OS
April 2006	Nokia N91 is the first phone to ship with Symbian 9 operating system (v9.1)
November 2006	Symbian handset shipments reach 100 million

Table 1. (*Continued*)

Date	Event
November 2006	Symbian agrees to sell UIQ Technology AB to Sony Ericsson
October 2007	Motorola agrees to buy 50% of UIQ holdings from Sony Ericsson
March 2008	Symbian handset shipments reach 200 million
June 2008	Nokia announces plans to buy out remaining 52.1% of shares, and to create a single unified platform
July 2008	Symbian launches 3rd ecosystem program, the Symbian Partner Network
August 2008	Symbian Platinum Partner program discontinued
November 2008	Nokia completes acquisition of Symbian Ltd., which ceases to exist as an independent entity
February 2010	Symbian Foundation releases 40 million lines of Symbian code as open source
October 2010	Samsung and Sony Ericsson cancel Symbian plans, leaving only Nokia and DoCoMo-licensed handset makers
February 2011	Nokia selects Windows Phone as its future smartphone platform, announcing plans to phase out use of the Symbian platform
June 2012	Nokia releases Nokia 808 PureView, which becomes its last Symbian handset

Source: Symbian (2008a), news accounts.

Psion spinoff "serious competition" in a memo leaked to the *New York Times* (Markoff, 1998).

Symbian Ltd. was a software company whose primary focus was to license the Symbian OS to the world's leading handset makers to produce what it termed "smartphones." Beyond the ability to make voice calls on GSM mobile phone networks, the phones inherited the capabilities of Psion's organizers (such as calendar and address book), to which Symbian and its partners added features suitable for a mobile Internet device (such as e-mail and web browsing).

The firm was launched with approximately 160 employees transferred from Psion Software. By the time it had grown to 1,000 employees in 2004, technical employees – both R&D and technical consultants – comprised 77% of that total. Because it shipped no products directly to end users, it had a relatively small sales operation that worked with handset makers, while the marketing organization focused on generating industry visibility to attract end users and third-party developers.

Symbian's shareholders were its spinoff parent (Psion) and mobile phone makers that were also its customers. Psion and Symbian hoped for an IPO of the company, but it was blocked by the handset makers. Instead, shares

were bought by handset makers, with Nokia acquiring the largest stake (47.9%) in 2004 (West, 2014).

Like other OS companies, Symbian sought to maximize the supply of software supplied by third parties and thus the value created by that software. At the same time, as with any platform it was forced to trade off advancing the OS capabilities against providing continuity of interfaces for such software.

Symbian's ecosystem had key differences compared to the PC archetype. While the Windows ecosystem gradually emerged during the period 1981–1991, Symbian created an ecosystem strategy even before shipping its first product, a strategy that evolved across three distinct phases in its first decade. Without Microsoft's independent funding and control of key applications, Symbian had less supplier power and platform control. The management of the Symbian ecosystem was also constrained by the complexities of complements, systems architecture, distribution, and ownership relations not present in better-known computing architectures.

ARCHITECTING A SMARTPHONE OS

Modular Architecture

The Symbian platform consisted of the Symbian OS, a user interface framework, and an ARM-compatible central processing unit (CPU). ARM did not make smartphone CPUs, but licensed its reference designs to a wide range of semiconductor makers and worked closely with Symbian and the CPU makers to deploy each generation of its architecture (Chambers, 2006, pp. 100–103). ARM licensees Texas Instruments, ST Microelectronics, Renesas Technology (a Hitachi-Mitsubishi joint venture), and Ericsson Mobile Phones were leading CPU suppliers for Symbian-enabled phones.

The Symbian OS architecture was a hybrid between a general purpose computing architecture and that of a smartphone (Fig. 1). It included standard kernel services such as memory management and multitasking. General purpose middleware included networking, graphics, Internet, printing packages, as well as the user interface. Other middleware provided smartphone functionality, combining PDA features such as personal information manager services (for address book and calendar) with telephony services that enabled call management and logging.

The platform allowed for Java-based applications and (except for NTT DoCoMo customers) native C++ applications. Each handset maker

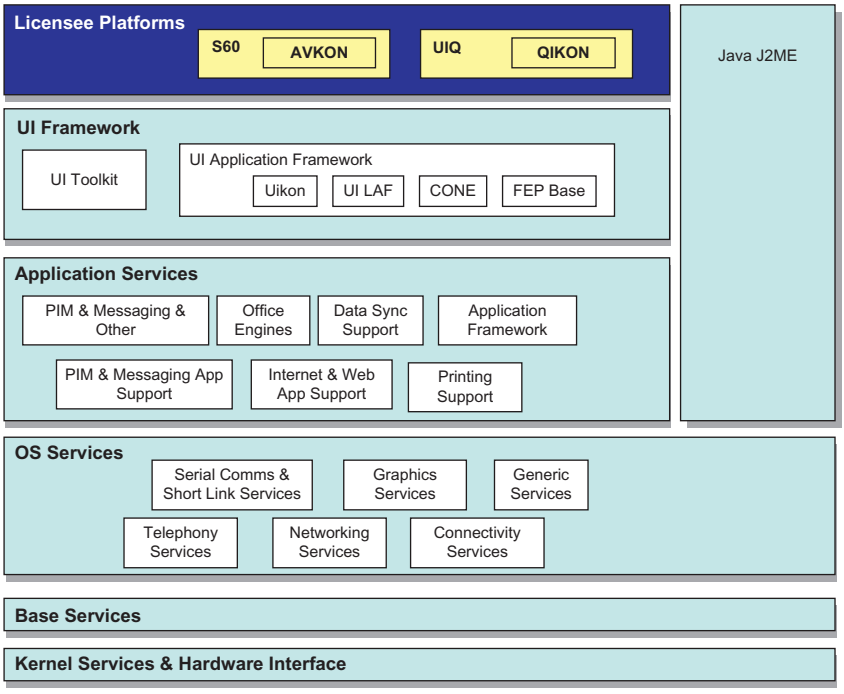

Fig. 1. Symbian Architecture. *Source:* Symbian.

licensed a Java interpreter as well as an engine for editing word processing and other office documents. Makers of CPU, graphics, and other chips customized the OS for maximal compatibility with their hardware.

Sub-platforms

While the first phone (Ericsson R380) had a custom user interface that was used only once, Symbian had designed its OS to make it easy to change the user interface "look and feel." Unlike the Windows (or later Android) mobile phone platform, these custom user interfaces (UIs) allowed each handset vendor to offer distinctive products. In the end, five different user interfaces were developed (Table 2), but only three shipped more than 5 million units.

Series 60 (later S60) was created by Nokia and licensed to other handset makers. This was the most popular user interface, both in terms of distinct

Table 2. Handset Model Production and Unit Sales for Symbian-Based Sub-platforms.

User Interface	MOAP (S)	Series 60[a]	Series 80	Series 90	UIQ	*(Emerald)*
Code name			Crystal	Hildon	Quartz	Emerald
Originator	NTT DoCoMo	Nokia	Symbian	Nokia	Symbian/Ericsson	Ericsson
Developer	Fujitsu	Nokia	Nokia	Nokia	UIQ Technology	Ericsson
Retail availability	2003–2013	2002–2013	2001–2007	2005–2006	2002–2009	2000–2001
Number of models[b] (2000–2012)	129	145	6 (Nokia)	1 (Nokia)	22	1 (Ericsson)
Best-selling models	n/a	Nokia Nseries	Nokia 9000 series	Nokia 7710	Sony Ericsson P800, P900, P910	Ericsson R380
Unit sales[c] (2000–2010)	65 million	370 million	<1 million	<100,000	10 million	<100,000

[a]Includes open source successors (Symbian[1], Symbian[2], Symbian[3], Anna, Belle) released in 2009–2012.
[b]For 304 models commercially released from September 2000 to June 2012; for sources, see end notes.
[c]Authors' sales estimates based on Symbian press releases and analyst reports.

models (145 designs[5] from nine vendors) and also unit sales (more than 350 million). It was character-and-icon based, with a cursor key and numeric keypad (later also a QWERTY keypad) as primary input devices. By virtue of its customer power, Nokia forced Symbian to accept Series 60 as a replacement for the "Pearl" interface, which was partially developed by Symbian but never used in any shipping product (Orlowski, 2010).

Series 80 was a UI developed by Symbian and maintained by Nokia, optimized for Nokia's 9000 series phones and exclusive to Nokia due to its patents (Orlowski, 2010). Like the original 1996 Nokia Communicator, these were among the heaviest and most expensive mobile phones, providing a larger screen, QWERTY keyboard, and folding "clamshell" design to substitute for a laptop computer. Because of the cost of maintaining a separate UI code base, Nokia dropped Series 80 in 2006 in favor of Series 90.

Series 90 was Nokia's most innovative interface, with stylus input and designed to support a larger color screen. However, it was used only by a handful of phones, only one of which (the Nokia 7710) ever reached the market. The user interface design was later re-used by Nokia in the Maemo Linux-based tablet computers that it released starting in 2005 (cf. Stuermer, Spaeth, & von Krogh, 2009).

UIQ was developed by Symbian using the former Ericsson Mobile Applications Lab in Ronneby, Sweden, and was primarily used by Sony Ericsson phones. The lab was sold off as UIQ Technology AB to Sony Ericsson in 2006; in 2007 Motorola bought half. With a stylus-based interface and a large supply of third-party software, some considered the UIQ interface to be the most modern smartphone design prior to the 2007 introduction of the iPhone.

MOAP (S) was developed by NTT DoCoMo and Fujitsu as the initial smartphone platform to support DoCoMo's "FOMA" 3G service (Yoshizawa, Ichikawa, & Kogetsu, 2006). (DoCoMo later contracted with Panasonic and NEC to create a rival Linux-based interface called MOAP(L)). Fujitsu shipped the first MOAP handset in 2003 and half of all MOAP (S) models, while Sharp, Mitsubishi, and Sony Ericsson began shipping handsets in 2005. Unlike other Symbian UIs, MOAP (S) did not allow downloadable native applications, and the application ecosystem was managed by DoCoMo, not Symbian.

Each user interface was in effect a sub-platform of the Symbian platform, each with its own UI-specific APIs. Because the UI makers had source code to the OS, they (particularly Nokia) added their own UI-specific APIs to the Symbian OS; most (but not all) APIs were eventually

migrated back to the shared Symbian code. Each user interface also had its own preferred web browser. At the same time, the proliferation of UIs increased Symbian's coordination costs and fragmented the application market. (In addition to controlling the user interfaces, handset makers also controlled the lowest level interfaces for the platform, the hardware adaptation layer.)

Programming Interfaces

Building upon Psion's EPOC, the Symbian OS APIs used a customized version of the C++ programming language to develop native applications, and it attracted experienced Psion software developers, especially from the United Kingdom. Application developers faced two challenges in writing software for Symbian devices, both of which increased developer learning time and thus specialization costs. First, the Symbian programming model (particular memory management) was not like any other device – unlike Windows Mobile, iPhone OS, or Android that includes smartphone versions of popular PC APIs. Second, the divided platform control – between Symbian and its UI companies – meant that developers often had trouble finding the right information about APIs or other development questions.

Two major efforts lowered learning costs by adding new APIs. In 2007, Symbian released POSIX-compatible C libraries that made it easier to port established Unix or Linux software – enabling versions of Quake (a multiplayer game), VNC (a desktop control client application), and Qt (a user interface library). In 2004, Nokia also released a Symbian S60 implementation of the Python programming language. As with Java on desktop computers in the 1990s (cf. West & Dedrick, 2000), adding a new shared API layer abstracted differences between mutually incompatible platforms, reducing switching and specialization costs. Although popular among Unix hackers for ease of rapid prototyping, it offered only a subset of the S60 APIs (Scheible & Tuulos, 2007).

Finally, at the behest of network operators (who distributed more than 90% of the world's mobile phones) in 2004, Symbian took steps that ended up making software development *more* difficult. The Symbian Signed initiative was intended to prevent viruses and other malware from taking over a handset and causing damage to a handset or the network (Morris, 2008). While security was widely seen as necessary, software developers voiced frustration over the resulting technical difficulty and bureaucratic approval delays.

SYMBIAN'S EVOLVING ECOSYSTEM STRATEGY

When Symbian was founded, it faced crucial challenges in building an ecosystem to support its technological innovation. Despite its inheritance from Psion, the new firm would need a new ecosystem. Its managers did not know what sort of ecosystem would be required: like its competitors, it assumed that smartphone ecosystems would be similar to those for PDAs.

As its technology grew more popular, it attracted a growing number of potential ecosystem members, each wanting attention to solve their particular problems. At the same time, it was a small and (until 2005) a money-losing company with limited resources. Thus, a crucial challenge was prioritizing its scarce resources to build an ecosystem of unknown characteristics.

Finally, it faced two immediate competitors in Microsoft and Palm, at least three major future competitors largely unknown in 1998 (as well as indirect competition from in-house handset software development by potential customers). As with most software companies, the intellectual property of its software copyrights and trade secrets were its major assets and it worried about leakage of that IP to existing or potential competitors (cf. Cusumano, 2004). Ultimately, these IP concerns colored (and hindered), its willingness to transfer knowledge to ecosystem members, and thus its ability to attract new members and to help them create value.

SYMBIAN'S ECOSYSTEM: OVERVIEW

Symbian OS was only available to phone users pre-installed in a newly purchased Symbian-enabled phone. This meant that unlike a PC, Symbian could not sell end-user software upgrades and had effectively no direct relationship with customers. Instead, adoption of its latest technology – and revenues – depended on new adoption of smartphones and replacement purchases by existing owners.

Symbian described its network of customers and complementors as an "ecosystem"[6] (e.g., Northam, 2006). Different categories of licenses and partner relationships included the following:

- *System integrators or "Licensees"* (handset manufacturers) that integrated externally sourced and internally developed hardware and software to create new devices (i.e., handsets) for sale to end users.[7]

- *CPU vendors* worked to assure Symbian OS compatibility with their latest processors.
- *Other hardware suppliers* provided drivers for their respective hardware components.
- *User interface companies* were division of mobile phone companies or (for UIQ Technology) a separate company.
- *Other software developers*, sometimes referred to as independent software vendors (ISVs). This included developers of user applications and also middleware components such as databases.
- *Consultancies and training centers*. Symbian provided licensees a list of certified contract software development companies it called Symbian Competence Centers, whether mobile phone-specific consultants or the Symbian-oriented departments within large outsource software suppliers such as Satyam and Wipro.
- *Network operators*, which in most countries were the dominant distribution channel for phones, and also decided what software components were preloaded on phones.
- *Enterprise software developers*, for cases where a company developed Symbian-compatible software for its employees that used Symbian phones.

In many cases, members of Symbian's ecosystem were also members of competing mobile phone ecosystems, such as those surrounding Palm OS, Windows Mobile, and later Linux-based platforms such as the LiMo Foundation and Google's Open Handset Alliance (Android). Such divided loyalties were found not only in chipmakers and operators but (unlike with PCs) also in the system vendors who made phones that incorporated the Symbian OS.

Knowledge transferred to partners came in three forms: codified documentation, personalized technical support, and Symbian's source code. Most partners had access to only a subset of source code, while both UI companies and mobile phone operators asked Symbian to limit access to sensitive interfaces (such as those that might allow a wayward application to make expensive telephone calls). For employees who had full source code access, it came at a price: Symbian demanded a "refrigeration period" (typically six months) – during which the engineers were blocked from working on complements for competing platforms – for fear that they would unintentionally apply concepts of the Symbian code in a way that would improve the capabilities of Symbian's competitors.

Symbian's strategy for managing its ecosystem can be divided into three phases: "ad hoc" (1998-2002), "Platinum Program" (2002–2008), and "Symbian Partner Network" (2008).

Phase 1: An ad hoc *Ecosystem Strategy*

The initial structure and conception of the Symbian ecosystem was heavily influenced by its PDA forebears, and the well-known exemplars of PC and other computer ecosystems.

At the time of Symbian's founding, all three major PDA makers – Psion, Microsoft, and Palm – had active programs for attracting third-party application software. In fact, the initial Symbian OS carried over the APIs, technical documentation, ecosystem support staff, and supply of third-party software suppliers that had worked with Psion. Psion organized a series of developer conferences, starting in November 1992 with 10 talks to an audience of around 30 developers (Symbian, 2008a) and growing to around 200 developers at events in 1997. "One of the attractions of Symbian OS for Nokia and Ericsson was the reasonably big set of developers we had built up as Psion," recalled Simon East, first VP of technology for Symbian. Symbian's initial ecosystem strategy thus focused on working with third-party software developers.

This strategy is illustrated in Fig. 2 from a 2002 presentation made to a meeting of Symbian's Supervisory Board (Wood, 2002). Initially, Symbian used two different forms of formalized knowledge transfer to ecosystem members. The OEM Customization Kit (OCK) provided the Symbian OS, tools, and associated documentation for handset makers, while the "SDK" was the software development kit, with documentation used by ISVs to create applications and other add-on software.

However, the company soon realized that other potential ecosystem members needed their own specialized support. For this reason, a

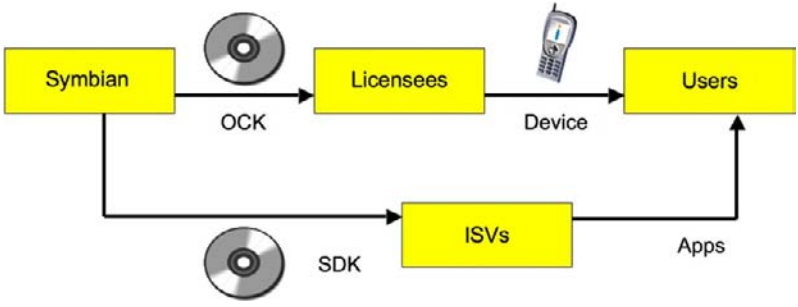

Fig. 2. Symbian's Ecosystem Concept as of 1998. OCK, OEM Customization Kit (for Handset Makers); SDK, Software Development Kit (for Software Developers); ISV, Independent Software Vendor. *Source:* Wood (2002).

number of separate partnering programs emerged *ad hoc* during the period 1998–2002:

- A program for Symbian Competence Centers, announced at the opening of February 2000 Symbian's Developer Conference in Silicon Valley;
- A program for Symbian training partners;
- A Symbian technology partner program, for companies providing technology (such as multimedia engines or compression modules) to run alongside Symbian OS;
- A semiconductor partner program, for companies providing hardware components to phone vendors;
- A tools partner program, for providers of compilers, integrated development environments, automated test facilities;
- A development partner program, for firms supplying technology to Symbian itself;
- A connectivity partner program, for companies providing solutions for synchronizing and backing up data between mobile devices and desktop computers.[8]

Each program tended to be managed by separate Symbian employees, often in different departments. Each program emerged following a separate motivation. It gradually became clear that each program had to meet complex needs, but that there were considerable commonalities.

Over the next few years, Symbian's actual ecosystem and the pattern of ecosystem coordination evolved from the original Psion-inspired model (shown in Fig. 2) to a new model (Fig. 3) that differed from the earlier conception in two key ways:

- Many partners were supplying software to phone manufacturers, but needed a greater amount of technical information and software than was contained in the SDKs designed for ISVs;
- These same partners (termed "Licensee Suppliers") needed two-way exchange of software with phone manufacturers, in ways that neither the OCK nor the SDK had envisioned or provided for.

Although Symbian had serious and well-managed programs to manage both phone manufacturers and ISVs from 1998 to 2002, in retrospect it underemphasized helping those companies that supported the phone manufacturers in creating devices, delaying the availability of new phones, and thus new Symbian customers.

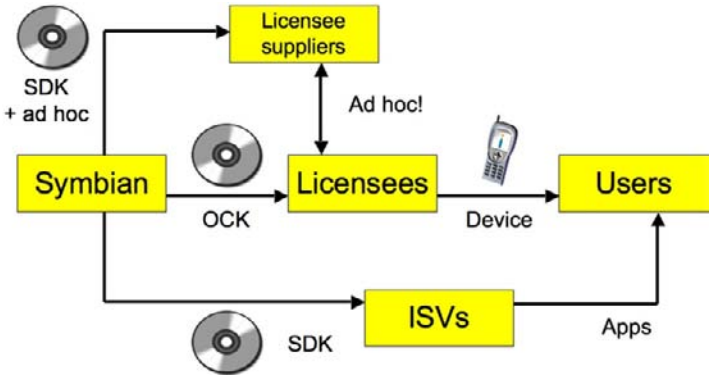

Fig. 3. Symbian's Ecosystem Concept ca. 2000. *Source:* Wood (2002).

Phase 2: Symbian Platinum Partners

Around July 2001, a proposal was created to unify many aspects of the previously separate partnering programs, into a new "Platinum Partner" program. The main differences were as follows:

- A deliberate preference for firms providing technology supplied in devices, rather than software added on afterwards. The "device creation" related partners included not only phone manufacturers (e.g., Nokia) but also providers of hardware components (e.g., Texas Instruments and Intel) and those that provided bundled middleware (e.g., Sun and Real Networks) or development tools (e.g., Borland and Metrowerks).
- A desire to systematize the efforts of running many different partner programs, and to obtain benefits of scale through having common development kits, event management, billing systems, and communications systems.

A key aspect of the new program was a new package of software provided to technology suppliers, known as the Development Kit ("DevKit"). This contained considerably more software than in the SDKs provided for ISVs, as well as additional licensing rights, but stopped short of the software and rights available to phone manufacturers. Meanwhile, the software package previously known as the OCK was re-designed as the "CustKit" (Customization Kit).

The program was discussed internally for nine months, before being announced in April 2002. Reasons for the delay before launching the program included

- Internal discussions over the appropriate membership fee for the program. Initially, annual fees of $15-25k were proposed. After some time, the concept emerged of a lower rate ($5k) for program membership, coupled with a surcharge if a partner wished to license the more extensive DevKit;
- A change in Symbian CEOs, when its first CEO, Colly Myers, resigned in February 2002 and was replaced two months later by David Levin.

Several key differences between the first and second phases of the ecosystem strategy are emphasized by comparing the previous diagram with Fig. 4:

- "Licensee Suppliers" were renamed as "Partners" (sometimes referred to as "Device Creation Partners"), and the emphasis on supporting them increased because of their important role in helping create new phones;
- Previously ad hoc support mechanisms from Symbian to different Partners were reorganized around the existence of the new DevKit;
- Previously ad hoc exchange of information and software between partners and licensees became governed by contractual terms in the new DevKit License (DKL);

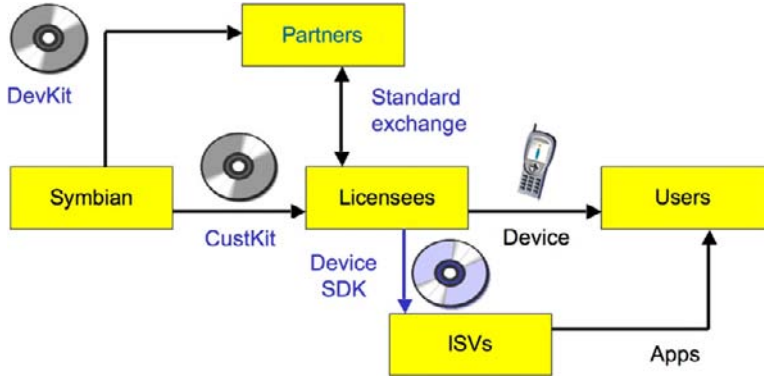

Fig. 4. Symbian's Ecosystem Concept as of 2002. DevKit, Development Kit (for Component Providers); CustKit, OS Customization Kit (for Handset Makers). *Source:* Wood (2002).

- Symbian put less priority on direct support of ISVs, on the assumption that the task of supporting applications developers would shift to the phone makers and those companies' UI systems, which after 2001 were located outside Symbian.

Instead of directly supporting ISVs, Symbian's Developer Network program would concentrate on being a hub of support for the developer networks in partner companies, who would in turn support ISVs.

Once the Platinum program structure was in place, it grew rapidly: by the end of 2002, it had attracted 100 companies, and nearly 300 by early 2006. Even as the program grew in size, Symbian management felt constant conflict between "quantity" and "quality" of partners:

- The "quality" approach involved a preference for the larger companies that seemed most likely to become winners in the Symbian space, or which had special endorsements from Symbian's customers.
- The "quantity" approach followed the principle of "level playing field" – avoiding picking winners, but giving an equal opportunity to small and unknown companies; the idea was that even though a given company might have the best technology of its type at one moment in time, this should not become a reason to imagine that company would remain indefinitely as the leader in its space.

Efforts to provide openness with a "level playing field" required more resources to administer a larger program – including keeping track of contacts, preparing and chasing invoices, providing technical support, and running larger partner events.

Other difficulties in running a large partner program were already anticipated at the time the Platinum program was created. An April 2002 analysis of the partner program (Wood, 2002) noted two potential problems. First, many firms were trying to become partners, but they varied widely in terms of their ability to deliver meaningful products. Second, Symbian did not have a large enough technical staff to provide the desired level of support for all possible partners.

For these reasons, a prioritization scheme was viewed as inevitable, and partners (including potential partners) were internally allocated to different tiers of importance: AA, A, B, and C. The AA partners were 15 companies deemed most critical to Symbian's success, A-level were 50 companies of high significance, the B level were those with at least one internal champion, and the C level comprised the remainder (Wood, 2003).

Finally, ecosystem members differed significantly in their rights to use Symbian's IP. Phone manufacturers received all source code to Symbian OS,[9] whereas partners did not receive so-called Category A source code that was deemed to be particularly sensitive. Based on an assumed "hub-and-spoke" model, partners could only distribute their changes to selected Symbian OS software to phone manufacturers (at the "hub"), not to another partner. Over time, it became clear to Symbian management that both restrictions hindered the free flow of valuable information and innovation among the ecosystem; both restrictions were eventually removed.

Symbian benchmarked the Platinum program on an ongoing basis: almost every year between 2004 and 2007, there were one or more internal review projects to consider major improvements in the partnering programs. These projects usually started optimistically: people would say things like "It should be easy to stop wasting effort on the low-value partner engagements and to put more effort onto the high-value partner engagements." But each time, the optimism changed to acceptance that easy optimizations of the program had already been made, and that partnering activity which initially looked low value was often highly valued by important Symbian stakeholders (key customers, internal strategists, etc.).

Phase 3: Symbian Partner Network

In November 2007, Symbian began developing a new partner program to meet two key objectives. The first was to increase the efficiency of the program through the enhanced use of IT, particularly increased web-based automation of common activities and creation of an improved extranet (called "SDN++") to communicate with ecosystem members. The second was to utilize that increased efficiency to lower the price and broaden the reach of the program, particularly with ISVs.

The most visible change was a reduction in the annual fee from $5,000 to $1,500. Part of the cost savings came by eliminating assigned partner managers for each registered partner, with ecosystem members instead being supported using extranet-based standard information and (paid) technical support. Symbian also lowered its expectations to a break-even basis; the Platinum program (designed when Symbian was losing money) had eventually generated a net profit.

Symbian unveiled the revised program on April 29, 2008, to ecosystem members at its semi-annual partner event, and encouraged its existing partners to migrate to the new program. It was not announced publicly until

July 7, with a concurrent announcement that the Platinum program would be discontinued six weeks later. However, the impact of the revised partner program was diminished by the surprise announcement two weeks earlier that Symbian would become a Nokia subsidiary and license its source code as open source software.

FIRM AND PLATFORM SUCCESS (2002-2007)

Platform Success

In the period of 30 months, three manufacturers shipped four Symbian phones with four different UIs: the Ericsson R380 (September 2000), Nokia 9210 (June 2001), Nokia 7650 (June 2002), and Fujitsu's FOMA F2051 (January 2003). The Symbian platform enjoyed uninterrupted exponential growth from 2002 to 2007; after three flat years, sales set a new record in 2010 with nearly 112 million units sold (Fig. 5).

Symbian's initial smartphone competitors – Palm and Microsoft – were also PDA based; but led by Nokia, the Symbian platform" or "were also PDA based, but led by Nokia, the Symbian platform passed both to achieve a majority of global mobile device sales (including PDAs) by mid-2004 (Canalys, 2004). Over time, the competitive threat from Palm faded, but phones licensing Windows Mobile and vertically integrated smartphones from Research in Motion (the Blackberry) and Apple (the iPhone) continued to grow in sales, particularly in North America. Two nonprofit consortia were formed in 2007 to standardize and promote new Linux-derived handset platforms: the LiMo Foundation (led by British operator Vodafone and Japanese operator NTT DoCoMo) and the Open Handset Alliance (created by Google to promote Android).

By the end of 2007, smartphones had grown to about 10% of all handsets sold worldwide, and Symbian OS was estimated to account for 63% of all smartphones – well ahead of Windows (12%), the Research in Motion Blackberry series (10%), Apple's iPhone (3%), and Linux (10%) (West, 2014; West & Mace, 2010). Nokia sold about half of all smartphones and nearly 80% of all Symbian-enabled handsets. Symbian accounted for a majority of global smartphone sales through 2008, and a clear plurality through 2010 with 37.6% of the market.

Symbian's major challenge was in North America. For example, in the summer of 2004, the Symbian platform had a 6% share of the US mobile device (smartphone and PDA) market, after 43% for Palm OS and 25% for

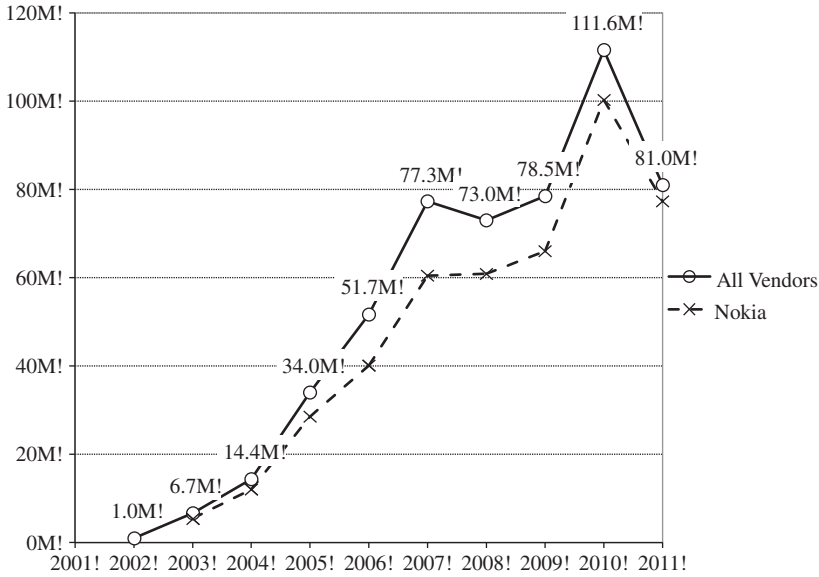

Fig. 5. Global Unit Sales of Symbian OS Phones (Millions).

Windows (Canalys, 2004). One major problem is that Symbian developed a version of Symbian for CDMA networks – which accounted for a majority of US subscribers – but Nokia cancelled its CDMA phones before they could be released and the CDMA modifications sat on a shelf, unused.[10]

Another obstacle was winning distribution for phones from the United States's three (later two) nationwide GSM operators. In particular, the largest – Cingular (later AT&T) – wanted weak suppliers and so rarely carried any phones from Nokia, the global mobile phone leader. As a consequence, Symbian was dependent on the relatively weak T-Mobile.

When Nokia brought its first Symbian phone to the United States, the $600 Nokia 9290 Communicator, without operator support it distributed the phone via computer dealers, IT consultants, and its website. Nokia even opened retail stores in New York and Chicago in 2006, but closed them in 2010.

Firm Success

Despite rapid organizational growth and market share success, Symbian faced severe resource constraints, suffering years of losses developing its

platform prior to achieving economies of scale sufficient to support its R&D efforts. Building on the Psion code base, Symbian spent (by our estimate) more than £200 million on R&D from 1999 to 2004 to develop three major Symbian OS releases. It achieved its first operating profit in 2005, a year in which its revenues and unit sales more than doubled, and the year Nokia launched its high margin Nseries phones (Table 3).

Symbian suffered from a lack of pricing power, particularly from 2004 onward, when Nokia accounted for more than 75% of unit sales. In early 2006, Symbian was pressed by shareholder-customers to adopt a reduced royalty schedule. The company would no longer receive a $2.25 surcharge on the first two million handsets of each major OS release. More significantly, the ordinary royalty shifted from a flat $5 fee to a graduated scale from $5 to $2.50 (Symbian, 2006). The latter provision benefited only Nokia – the only company to ship more than 5 million Symbian phones in a single year – and was in fact adopted at its behest.

Finally, of Symbian's handset customers, only Nokia was able to achieve economies of scale for its product and UI platform development costs. While Nokia averaged sales of more than 3 million units per smartphone, Sony Ericsson averaged less than 1 million, and also had to support the UIQ sub-platform development with less than 10% of Nokia's smartphone revenues.

Assessing Symbian's Platform Success

At the beginning of 2008, Symbian's platform and ecosystem strategies had achieved great success. It had attracted 9,282 third-party software applications and in 7½ years, its OS had been shipped in 200 million phones, the most in the industry (Symbian, 2008b).

Throughout its ecosystem strategy, Symbian had ongoing debates over the balancing between competing goals such as quantity versus quality, fairness versus focus, and personal attention versus economies of scale. In making such decisions, the Symbian executives and ecosystem managers faced three major limitations.

The first was a cognitive blind spot toward the nature of the ecosystem. As part of the Psion PDA (and PC) legacy, Symbian's founders initially took for granted that its ecosystem would be like Psion's PDA ecosystem; the implicit assumption was that the major focus of ecosystem management was working with independent software vendors. In this regard, application software as the most important complement to general purpose computers

Table 3. Symbian Ltd. Financial Performance, 2002–2008.

	2008 Q1–Q2	2007 Q1–Q2	2007	2006	2005	2004	2003	2002
Royalties	£70.8m	£78.2m	£179.1m	£151.8m	£96.8m	£45.2m	£25.5m	£7.7m
Other income	£10.5m	£7.2m	£15.2m	£14.4m	£18.0m	£21.3m	£19.9m	£21.8m
Total revenue	£81.3m	£85.4m	£194.3m	£166.2m	£114.8m	£66.5m	£45.5m	£29.5m
R&D				£70.0m	£54.5m	£43.7m	£40.3m	£35.8m
SG&A				£34.1m	£27.0m	£23.5m	£19.9m	£16.8m
Net income				£55.1m	£15.3m	(£23.0m)	(£26.5m)	(£37.2m)
Gross margin				*84.4%*	*77.5%*	*70.7%*	*63.9%*	*43.8%*
Net margin				*33.1%*	*13.3%*	*−34.6%*	*−58.3%*	*−126.1%*
R&D intensity				*42.1%*	*47.5%*	*65.8%*	*88.7%*	*121.3%*
R&D (% of operating expense)				*67.4%*	*66.4%*	*60.3%*	*68.2%*	*67.8%*
Total employees				1191	1047	835	734	653
R&D employees				824	693	525	464	405
Average unit royalty ($)	$3.7	$4.4	$4.5	$5.30	$5.14	$5.7		
Average unit royalty (£)	*£1.86*	*£2.26*	*£2.32*	*£2.94*	*£2.85*	*£3.14*	*£3.81*	*£7.70*
Total unit sales	38.1m	34.6m	77.3m	51.7m	34.0m	14.4m	6.7m	1.0m
Nokia units[a]			*60.5m*	*40.1m*	*28.5m*	*12.0m*	*5.5m*	*0.5m*

Source: Symbian.com press releases via Archive.org and 2003–2006 Symbian annual reports to shareholders.
Notes: Authors' calculations shown in *italics*.
[a]Estimates based on Symbian press releases, Nokia annual report and analyst reports.

was the dominant logic (as defined by Prahalad & Bettis, 1986) of the computer industry of that era. A related assumption was that add-on applications were crucial to the value of a smartphone – true for PCs and game consoles, arguably false for PDAs, and demonstrably false for conventional mobile phones.

Second, new handset models were delayed because Symbian did not anticipate how hard it would be to create devices that were unmatched in complexity for a consumer electronics device. Wood (2005) identifies a number of potential pitfalls of mobile phone production, including changes in OS (or UI) APIs across new releases, problems with third-party software reliability and integration, and contractual delays in obtaining rights to distribute such software.

Finally, Symbian's ecosystem management had limited resources and had to be self-supporting – particularly until Symbian earned its first profit in 2005. Rather than maximizing partner access, the partner program was limited to providing services to those partners willing to pay enough money to support the cost of providing those services. These restrictions were gradually reduced through IT-enabled efficiencies, including shifting from paper to "click through" agreements and distributing information via an extranet rather than CD-ROM.

Symbian's shifting treatment of application software was also problematic. Under its initial ecosystem strategy, the company focused on applications at the expense of helping handset makers and those providing pre-installed software that had to be ready to ship with the handset. These early priorities delayed the availability and sale of smartphone handsets that would attract buyers away from conventional handsets, create an installed base for application developers, and also provide revenues to Symbian that would reduce its severe resource constraints.

During the second (Platinum Partner) phase, applications were de-emphasized and ISVs received less attention. Only limited progress was made on improving tools and broadening developer support, and the cost to ISVs remained relatively high. It was only after the release of the iPhone that Symbian began to develop the Symbian Partner Network to broaden the reach and lower the cost – a development that was rendered moot by Nokia's acquisition of Symbian.

Finally, while the size of the software ecosystem continued to grow with the number of applications, Symbian made little effort to ascertain the health of its ecosystem, or to question why there were no great successes akin to the Lotus, Borland, Ashton-Tate, and others of the early PC era. As it turns out, after-market software sales for Symbian smartphones remained

low, as did the software unit price – both more similar to PDAs than PCs. Unlike a platform leader who squeezes complementors for profits in a zero-sum game (Gawer & Henderson, 2007), Symbian did not intend to starve its complementors, but focused more on its own problems than theirs.

These difficulties suggest two modifications to the positive-feedback network effects model (cf. Gallagher & West, 2009). First, while theory asserts that more software increases hardware sales, this assumes the *ceteris paribus* condition that attracting software does not delay the development or sale of hardware. If the hardware has a direct utility without adding software complements – and if the hardware must compete with an established substitute to attract buyers – this suggests the early priority must be on creating an installed base of hardware.

Second, the attractiveness of a platform to complement providers is not merely the size of its installed base, but the installed base size times its propensity to buy complements. If a given platform (or product category) has a higher propensity to install complements – whether PCs versus smartphones or between competing videogame platforms – then that creates a larger addressable market. Similarly, a lower unit price for complements is attractive only with a large installed base, high purchase propensity, or low specialization cost (cf. Teece, 1986) – exactly the conditions later created by Apple with its iPhone App Store.

Another problem came with Symbian's tradeoffs of selective openness of technical disclosure to complementors. As the first mover, Symbian erred on the side of secretiveness to protect its technology (particularly source code) as a trade secret – through disclosure policies that required significant contractual, technical, and administrative overhead to evaluate and then meet information requests. In doing so, it left itself vulnerable to an open source platform challenger where (by definition) the source code provides nearly complete documentation at zero transaction cost (West, 2003). The industry discussions since 2001 of Linux-based mobile phone platforms became a threat to Symbian's existence with the debut of Android-based products in late 2008.

Finally, Symbian's entire "open" platform strategy arguably depended on it being an independent supplier not beholden to any one customer – which was plainly no longer true after 2004. The Wintel platform succeeded precisely because Microsoft and Intel aggressively courted and developed competitors to their initial customer IBM, which gave them new customers, grew the market, and provided incomparable economies of scale. Although Nokia lacked a controlling interest in Symbian, its *de facto* control of

Symbian's revenues allowed it to discourage (if not block) efforts by Symbian to create competitors to Nokia (West, 2014).

SUDDEN AND UNEXPECTED DECLINE

The year 2007 marked the high water mark for both Symbian unit sales, and also for its influence on buyers, complementors, handset makers, and public perceptions. A little more than three years later, Symbian Ltd. had ceased to exist as a legal entity and its technology was officially orphaned by Nokia, its only remaining customer. While Symbian achieved strong market share in Europe and much of the world, it had very low penetration in North America, enabling entry by three local platform sponsors: Research in Motion with its proprietary BlackBerry platform (2002), Apple with its iPhone platform (2007), and Google with its Android open source OS (2008).

The original 2007 iPhone found immediate success in North America. Its success was attributed to a technical architecture that sought to replicate the Internet experience of a PC in a handheld device, through a large screen with finger-touch input, a desktop-capable web browser, and then (in 2008) a direct distribution mechanism for third-party software applications (West & Mace, 2010). These characteristics were copied by a series of Android phones (Kenney & Pon, 2011), thus cementing the dominant design for a consumer-oriented smartphone.

Anticipating but not Meeting Architectural Challenges

With its clear focus on creating the smartphone category, Symbian and its partners had anticipated key elements of the dominant design before Apple, but failed to execute on bringing them to market or to combine them into a single product.

The UIQ interface was a stylus-based input method, and a full-sized display (albeit at lower resolution) was characteristic of the earliest Sony Ericsson phones – P800 (2002), P900 (2003), and P910 (2004) – as well as the Motorola A1000 (2004) and A1010 (2005). However, the phones were far less popular than Nokia's competing models, and Motorola abandoned UIQ (twice) while Sony Ericsson shifted to promoting a Walkman family of music-oriented phones.

Symbian had difficulty with its browser strategy, both due to underestimating the strategic importance of the browser and sheer bad luck. Starting in 1995, Psion and later Symbian worked to source a web browser from STNC, a small British company located less than 80 miles from Symbian's London headquarters. However, Microsoft purchased STNC and its Hitchhiker browser in July 1999 to create its first mobile web browser. Beginning in 2002, Symbian handset makers licensed a browser from Oslo-based Opera, which never provided website compatibility comparable to a desktop browser. Instead, the most successful and compatible smartphone browsers were based on the open source WebKit, created by Apple for its Macintosh PCs and later used by both the iPhone (2007) and Android (2008) platforms (West & Mace, 2010). Nokia announced its own WebKit-enabled browser as a research project in 2004, and in 2006 bundled it with the S60 phones, but its implementation lagged both Apple's and Google's. Due to fragmentation, a WebKit browser was never released for UIQ phones.

Finally, Symbian was hindered by its legacy code and installed base in meeting the challenge of more modern APIs and development tools provided for the iPhone and Android. The two new platforms offered more modern programming languages and widely disseminated tools. (Apple had a free online course that was viewed by 100,000 potential developers in the first year.) To address this, in 2008 Nokia bought Trolltech, maker of the widely used Qt cross-platform user interface library. In 2009, it announced plans to provide a common set of Qt-based APIs for programming Symbian phones and Meego tablets. However, Nokia was still implementing this transition in early 2011 when it announced it would abandon Symbian for Windows.

Adapting to a New Ecosystem Paradigm

Symbian also faced a challenge to its fundamental ecosystem strategy – first from iPhone on openness to complementors and then from Android on openness to handset vendors. Both posed a challenge that Symbian was unable to meet.

In July 2008, Apple launched the iPhone App Store, providing an application distribution mechanism that bypassed both third-party distributors and the operators' own application stores. The new store offered an unprecedented feature for a computing platform: a built-in way to directly sell and install all third-party applications. It also provided Apple with 30% of all download revenues, although a large proportion of the applications were provided free.

The new App Store grew dramatically: while Symbian had taken 7½ years to acquire nearly 10,000 applications, the iPhone App Store reached 15,000 apps after six months and 100,000 after 16 months (West & Mace, 2010). The success of the App Store attracted customers and complementors, bringing tremendous favorable publicity for Apple.

In response, sponsors of the Android, Windows Mobile, and BlackBerry platforms all announced their own app stores. Symbian took 15 months to launch its own app store and – constrained by both Nokia and its operator partners – was not allowed to sell directly to users. Instead, Symbian provided wholesale distribution via Nokia's Ovi store and the operators' stores, neither of which proved as well-implemented or popular as Apple's or Google's app stores.

By one estimate, total 2010 app store revenues reached $2.2 billion; Apple's store accounted for 83% of the total, with software developers receiving $1.25 billion and Apple's commission revenues reaching $535 million (Whitney, 2011) – more than Symbian's entire 2007 revenues.

As it turns out, Symbian had considered creating its own application store back in 2005. The proposal failed to attract support within Symbian and eventually died for several reasons. Symbian felt it had been successful in attracting third-party complements; lacking a comparison, it was unaware of the financial pressures its developers faced; and it saw the store as a distraction and an expense, rather than as a source of significant revenues to solve its own financial pressures. Because Symbian lacked direct access to customers, it would also require the cooperation of its handset customer-shareholders and carriers, both of which later resisted its efforts to create an app store even after Apple's success with the iPhone App Store.

The other challenge came from the Android platform, which shipped its first smartphone in 2008. While Symbian had bragged that it was an "open platform," source code was developed by employees of Symbian and its license holders, and was only available under nondisclosure and a royalty-bearing license. Meanwhile, Android offered a royalty-free license and full source code to any external partner.[11] At the 2007 launch of the Open Handset Alliance, Android's nominal sponsoring organization, founding members included two Symbian shareholders and Licensees – Motorola and Samsung – as well as NTT DoCoMo, Symbian's main sponsor in Japan; shareholders Ericsson and Sony Ericsson joined 13 months later (Table 4).

Symbian was founded with the intention of providing a platform shared by all handset licensees. However, by 2004, Nokia held nearly 48% of Symbian's equity and more than 75% of its annual handset sales – and

Table 4. Handset Models Developed by Major Symbian Licensees.

	Symbian					Android
Handset maker	Equity investment	UI	First model[a]	Last model[a]	No. of models	First model[a]
Nokia: 125 phones	1998–2012	S60[b]	2002	2012	117	*None*
		S80	2001	2005	6	
		S90	2004	2004	1	
		UIQ	2005	2005	1	
Ericsson: 1	1998–2008	*Emerald*	2000	2000	1	–
Sony Ericsson[c]: 26	2004–2008	UIQ	2002	2008	12	2008
		MOAP	2006	2008	11	
		S60	2009	2010	3	
Motorola: 7	1998–2003	UIQ	2003	2008	6	2007
		MOAP	2005	2005	1	
Fujitsu: 61		MOAP	2003	2012	61	2010
Sharp: 37		MOAP	2005	2012	37	2010
Mitsubishi: 19		MOAP	2005	2008	19	–
Samsung: 15	2003–2008	S60	2004	2009	15	2009
Matsushita (Panasonic): 3	1999–2008	S60	2004	2005	3	2012
Siemens: 1	2002–2008	S60	2004	2004	1	–
LG: 3		S60	2007	2009	3	2010
Other: 9		UIQ	2004	2006	4	–
		S60	2004	2011	5	–

Sources: For phone models: same as Table 2. For equity: West (2014).
[a]First shipment date for a handset model (announcement date where shipment date unavailable).
[b]Includes S60-compatible releases after 2008 Symbian acquisition.
[c]Joint venture of Sony and Ericsson (2001–2012); includes one Sony-branded MOAP handset.

increasing degrees of *de facto* platform control. While Sony Ericsson once placed all its smartphone bets with the Symbian platform, other licensees such as Samsung and Motorola placed only tentative bets that they later abandoned. By 2009, Android had achieved what the Symbian platform ultimately failed to do: provide an open platform shared by a wide range of handset makers and controlled by none of them.

Platform Extinction

Challenged by Apple and Google, Nokia made a series of increasingly desperate moves to preserve its smartphone market share, leading to the

phased elimination of Symbian Ltd. and its platform. In June 2008, it announced it would acquire the remaining 52% of Symbian Ltd. for €264 million, with most Symbian engineers becoming part of Nokia and integrated into the existing S60 user interface team. Nokia hoped that combining the Symbian and S60 engineers would simplify control and speed development of the platform. The plan would also merge the UIQ and MOAP user interfaces into S60, to create a single platform.

The acquisition killed Symbian's business model, because Symbian and S60 source code were donated to a new foundation that would manage it as royalty-free open source software.[12] By openly disclosing the Symbian source code and other technology, Nokia hoped to grow the Symbian ecosystem and stem smartphone market share losses to its two leading proprietary platform rivals, as well as blunt the enthusiasm that openness had brought to Android. Nokia completed its acquisition of Symbian Ltd. in November 2008, created the Symbian Foundation in early 2009, and transferred control of the Symbian OS source to it. In February 2010, the foundation released all of the Symbian OS source code: the estimated 40 million lines of code was said to be the largest open source release of formerly proprietary code.

However, the open source experiment soon proved a failure. With the rise of Android, other potential handset sponsors stopped funding the Symbian Foundation, leaving Nokia's contributions (and its internal R&D group) providing nearly all the resources to support the platform. In October 2010, both Samsung and Sony Ericsson announced they would no longer develop Symbian phones. While Nokia, Fujitsu, and Sharp continued to release new phones, both Nokia and Symbian continued to lose smartphone market share. In February 2011, Nokia announced that it would phase out Symbian in favor of Microsoft's Windows Phone as its smartphone platform, although its Symbian sales exceeded Windows Phone sales until mid-2012.

DISCUSSION

By some measures, Symbian was a tremendous success. It created the smartphone category, and built a complex ecosystem through a series of alliances with key stakeholders with very distinct value creation roles. It evolved that ecosystem strategy over time, in response to changes in its conception of the ecosystem, the expectations of its complementors, and in the availability of enabling technology (notably dissemination via the

Internet). Together, Symbian and its partners created the most popular smartphone platform, growing volume at a CAGR of 238% from 2002 to 2007 to capture two-thirds of the global smartphone market.

However, unlike ecosystems limited by technological challenges (e.g., Adner & Kapoor, 2010), the downfall of Symbian's ecosystem and platform can be traced to three (largely organizational) limitations of its ecosystem. First, Symbian created a computing ecosystem of unprecedented organizational and technical complexity. Second, the asymmetric dependencies of the various ecosystem members meant some stakeholders flourished while others struggled. Third, the divided leadership of the ecosystem limited the ability of Symbian and its ecosystem to respond to the new dominant design created by the iPhone.

Evolving a Complex Ecosystem

Prior research on ecosystem strategies has suggested how sponsors can control and manage their ecosystem to best advantage, by orchestrating the value creation and taking for itself the largest share of the value capture (Gawer & Cusumano, 2002; Iansiti & Levien, 2004; Maula, Keil, & Salmenkaita, 2006). While this may be a desirable steady state goal, our data suggest that finding the path to that state is far from trivial – both due to the complexity of the task and the information available to the ecosystem leader.

At best, optimizing the performance of an ecosystem built around some complex assembled product requires aligning the interests of a heterogeneous population of ecosystem members and partitioning (or self-assigning) the technical and business responsibilities among those members. Any actor has the choice to participate or not in the ecosystem; for many industries, this choice is influenced by the decision of whether to participate (non-exclusively) in one or more competing ecosystems. Finally, the actual (or prospective) failure of any ecosystem member may cause it to withdraw from an ecosystem, leaving a gap that may be filled only after a considerable delay.

To this complexity, we add the inherent uncertainties (and unknowability) that come with a new ecosystem around the firm's new platform. These uncertainties will be greater for a new firm, without prior firm-level reputation products or ecosystem experience. They will be even worse for a new-to-the-world technology or product category, where there is no direct precedent (known to any party) for partitioning the business and technical

responsibilities across ecosystem partners. A firm without products or an ecosystem will have to make assumptions about what technologies and business relationships it will need to create value. As Alan Roderick, onetime head of the Symbian Platinum Partners, summarized it: "In the early days, nobody knew where smartphones were going to go, what they were going to be capable of, or what it would take to make them sell." To use the Mintzberg (1978) formulation, any ecosystem strategy has its intentional and emergent aspects, with the former manifest by the firm's activities, structures, and programs to create and nurture an ecosystem, and the latter arising from firms that choose to join the ecosystem, pressures from competing ecosystems, and broader changes in the environment.

If a firm enters a market without an existing ecosystem, then where does the firm's initial ecosystem strategy come from? Our data – and the industry standard practice that influenced our subjects – suggests two possible cognitive heuristics that shape ecosystem formation. First, the firm and its managers will build upon the firm's (or their individual) prior ecosystem experience – as when Symbian founders learned from their Psion experience. Second, lacking a large body of formal knowledge on ecosystem management, firms adapt strategies from similar ecosystems: in this case, the Windows ecosystem was extremely influential.

Both Symbian and Nokia lacked the platform experience of an Apple or Microsoft in terms of managing a successful general purpose computing platform across multiple generations. Meanwhile, Symbian seriously underestimated the complexity required to transform an electronic pocket organizer to a general purpose, Internet-connected mobile computing device. And unlike in Mäkinen and Dedehayir (2013) – where the platform progress was limited by third-party software – here the limiting factor was the ability of handset makers to integrate software. Even after Symbian had shipped its first complete OS, the weaker software development capabilities of its handset makers (and their UI companies) meant they had difficulty keeping up with Symbian (and eventually, rival platforms) in implementing new platform features. As with other examples of loose coupling identified by Brusoni and Prencipe (2013), the entire ecosystem suffered when there was poor execution by one key party.

Asymmetric Dependencies with the Ecosystem

Symbian depended fully on the success of its platform, as did many of its application suppliers. However, this was not true of other members of its

ecosystem, such as semiconductor makers, handset makers, and network operators. Unlike Symbian, this second group generated revenue from other mobile phones, not just Symbian. Nokia emphasized premium prices (of up to €1,000 for its best phones), maximizing its gross margins while limiting the number of customers available to Symbian and application providers. When other manufacturers (Motorola, Sony Ericsson) had less success selling Symbian phones, they sold non-smartphones or phones using Windows or (later on) Android.

Additionally, competition between handset makers within the ecosystem undercut efforts to build a common platform, and align the interests of the entire ecosystem to its shared success. Nokia, Ericsson, and NTT DoCoMo each built separate sub-platforms to support their respective aims, and Ericsson's sub-platform never attained the scale necessary to support its R&D costs. Outsiders such as Samsung, Panasonic, LG, and Siemens had difficulty developing for Nokia's S60 and grew wary of depending on the Nokia-controlled platform – much as IBM's rivals were wary of OS/2 (cf. Grove, 1996). Compared to Frankort's (2013) optimistic example of intra-industry cooperation and knowledge transfer, here the knowledge transfer was much less effective.

Finally, Symbian's success in attracting third-party applications masked the difficulty its partners had in profiting from those applications. Relatively weak application sales were not a priority for Symbian and ignored by the rest of the ecosystem – until Apple's iPhone created a new distribution paradigm that dramatically increased developer unit sales and proceeds.

Challenges of Divided Ecosystem Leadership

Normally identifying the ecosystem leader is clear-cut. When there are rival claims, Adner (2012, p. 116) argues that leadership can be inferred from the actions of others: "The leader is not the one who says, 'I'm the leader.' He's the one about whom everyone else says, 'He's the leader.' This is the litmus test of leadership."

From the date of its public unveiling, Symbian Ltd. was proclaimed as the leader of a new ecosystem by investor-customers who sought to transfer their legitimacy to the Symbian platform. However, over its 10-year lifespan, the actions of these customers served to undermine that leadership as manifest by technical control, market control, and financial control of the platform. At least, four factors contributed to Symbian's declining *de facto* leadership.

First, its technical leadership – through control of application-facing APIs (cf. West & Dedrick, 2000) – was intentionally preempted by handset makers who asserted API control by creating custom UI layers. As Simon East recalled: "It became clear to us that Nokia had woken up to the fact that if these guys own the UI and the developer model – then that's where the value is going to migrate to" (Orlowski, 2010). Relinquishing leadership both at the top (UI) and bottom (hardware) layers, Symbian and its platform suffered both from coordination problems with its licensees and also their generally weaker level of software development capabilities.

Second, because it did not sell to consumers, Symbian both lacked a direct source of revenues and a marketing relationship to assert its leadership with the intended beneficiaries of the ecosystem, that is, smartphone buyers. The top mobile computing analyst for one market research firm was stark in his 2001 warning:

> [T]he main obstacle Symbian faces, [Ken] Dulaney said, is brand awareness.
>
> "They've really done a poor job of really raising (the image of) their company," he said, adding that Palm and Microsoft have been much better at branding their names.
>
> Symbian has generally let the device maker do the talking while staying hidden in the background, Dulaney said – a strategy that isn't in the company's best interest.
>
> "I think they really need to reverse that strategy," he said (Dano, 2001).

Third, because its shareholders (other than Psion) had inherent conflicts between their roles as investors and customers, the investors controlled Symbian for their benefit as customers rather than to maximize the value of the company and their investment (West, 2014). By vetoing Symbian initiatives, Nokia and Ericsson weakened Symbian's financial health, its control of the platform, and the overall vibrancy of the ecosystem.

The final challenge to Symbian's leadership came with the end of the multilateral balance of power among the handset makers. Compared to its co-founders and ecosystem partners – notably Ericsson and Motorola[13] – Nokia proved the most consistent its ability to release new Symbian smartphones (Orlowski, 2010). Its global cellphone market share doubled from 1997 to a peak of 38.6% in 2008, while that of Ericsson and Motorola fell by half. As Symbian's dominant investor and customer after 2004, Nokia increasingly asserted leadership of the ecosystem – and others followed its lead.

According to Adner (2012, p. 117), "Successful ecosystem leaders capture their outsized returns in the end, after the ecosystem is established and

running. But in the beginning they build, sacrifice, and invest to ensure everyone else's participation." Who realized the financial gains of leadership? Nokia did but Symbian did not, having lost leadership at a time when it should have been harvesting profits.

We believe that Symbian's eventual failure suggests key difficulties of divided leadership of an ecosystem and a platform. The platform literature has largely ignored the potential tensions of shared (or divided) platforms: while Gawer and Henderson (2007) defined both Microsoft and Intel as Wintel platform "owners," they focus on their cooperative rather than competitive platform efforts.[14] When comparing the Symbian and Wintel platforms, it was clear that individual CPU suppliers played much less of a role, whereas the system integrators (handset makers) played a greater – at times crucial – role. Symbian's relationship with these integrators demonstrated a greater degree of rivalry than Microsoft faced either with its integrators or Intel. Although Microsoft competed with its application vendors, it enjoyed far more technical and market power than Symbian did in negotiating with handset makers.

This divided leadership ultimately hurt the Symbian platform and the ecosystem members, delaying its ability to respond to the iPhone, Android, and the App Store challenges. As market leaders, Symbian and its partners (especially Nokia) initially discounted the iPhone threat. Symbian was quicker than its partners to react, but lacked both the resources and technical control to react unilaterally. A key vulnerability – the browser – was not controlled by Symbian, but its sub-platform partners (particularly the market-leading Nokia smartphones) were vulnerable to direct comparison. Other aspects of the user experience (e.g., preloaded applications) were left to the handset maker or even the operator – a model rejected by Apple although later adopted with Android.

Future Research

There are inherent limitations as to the generalizability from a single case – in this case, a business ecosystem. The Symbian ecosystem differs from that for Windows Mobile, and significantly different from other mobile phone and computing ecosystems. Ecosystem relationships are considerably simpler in an industry where there is only one major class of complement such as with videogames, an industry where the degree of variation between firms and successive console generations cries out for a systematic study of ecosystem management.

In the tradition of Carliss Baldwin (Baldwin & Clark, 2000; Baldwin & Woodard, 2010), this work suggests further research as to the interdependence of the technical and economic relationships within an ecosystem. The Symbian ecosystem suggests that the technical structure is more enduring than the business structure. A piece of add-on software may initially be created as a complementary product sold independently to users by its developer, but in mobile phones such complementary software often became software components that are later integrated as part of the platform capabilities. What influence does an ecosystem leader play in developing and cultivating such potential value-adding components? How can we predict the difference in the value creation when a complement (adopted by a few) becomes an integrated component (provided to many)? And does this generalize to ecosystems beyond mobile phones where users have search or use difficulties acquiring complements – or to ecosystems where a small number of parties (here manufacturers and operators) play a disproportionate role in product distribution?

Since Symbian was launched, the practice of ecosystem management appears to have become better understood and more mature, with new examples such as mobile phones and videogame consoles as additional exemplars beyond the PC. Will this reduce the problem of a dominant logic for a new ecosystem that forces strategies to fit a single well-known exemplar? Or will it merely shift the definition of the dominant logic for ecosystem management?

NOTES

1. The first author interviewed Simon East, VP of Technology from 1998 to 2001; David Wood, head of Symbian Platinum Partners from 2002 to 2004; Alan Roderick, head of the Symbian Competency Centers from 2001 to 2004 and the Platinum program from 2004 to 2007; Patricia Correa, who succeeded Roderick, developing and launching the Symbian Partner Network program in 2008; as well as Isaac Dela Pena, who was a senior manager (among other positions) at Nokia from 2000 to 2010.

2. The history of Symbian and its relationship to Psion can be found in Tasker (2000) and Northam (2006).

3. Here, we focus on the pocket-sized PDAs that eventually proved to the be dominant design for the product category, rather the unsuccessful, tablet-sized PDAs such as the Apple Newton (1992–1997) and AT&T EO (1993–1994) that first gave the name to the category.

4. Motorola did not actually finalize its investment in the company until October 1998 (West, 2014).

5. We created a database of 304 Symbian handsets shipped 2000–2013 using the official public list of handsets on Symbian's website (both in 2008 and as stored on the Internet Archive), databases of phones on GSMArena.com and Japanese phones on Wikipedia.org, and press releases and news stories about handset releases.

6. The word "community" was sometimes used as an alternative, but "ecosystem" was generally preferred since "ecosystem" recognizes the reality that companies have competitive relationships, not just the "friendly" relations implied by the word "community."

7. Within Symbian, handset manufacturers were handled by the sales division rather than ecosystem management, but manufacturers also had full access to all partner information and events.

8. The state of the partner program in early 2000 can be found on the Internet Archive (Archive.org) using the March 6, 2000, backup of www.symbian.com/partners.

9. This source code excluded a very small portion that had been licensed in by Symbian from third-party suppliers under a contract preventing any other company from seeing the source.

10. Nokia's decision to cancel its CDMA handsets was seen as tied to its patent disputes with Qualcomm (originator of the CDMA mobile standard) that continued until the two firms settled in July 2008.

11. Despite its (largely) open source code, Android scored lower than Symbian, Linux, Mozilla, and five other mobile-related open source projects in an independent 2011 study of openness in open source communities (Laffan, 2011).

12. Such open source foundations were originally created to strengthen the negotiating position of volunteer individual contributors, but Nokia's intended use was deliberately modeled on the Eclipse Foundation, in which the foundation increases legitimacy by providing limited autonomy from its main corporate sponsor (see O'Mahony, 2003; West & O'Mahony, 2008).

13. Two other investors – Panasonic and Siemens – exited handsets, while licensee Sendo went bankrupt. For Samsung and LG, Symbian was always one of several smartphone platforms, and in 2010 the Korean firms later became the first and second largest makers of Android smartphones.

14. West and Dedrick (2001) were one of the first to identify the potential divergence of Microsoft and Intel's interests, when Intel supported Linux as a server operating system competing with Windows. Since then, Microsoft has offered Windows implementations for mobile phones and (with Windows 8) even PCs that do not require Intel-compatible chips.

ACKNOWLEDGMENTS

Earlier versions of this paper were presented at the 2008 User and Open Innovation Conference, the 1st Tilburg Conference on Innovation, and the Stanford Social Science and Technology Seminar. We thank the conference participants and the editors for their many helpful suggestions.

REFERENCES

Adner, R. (2012). *The wide lens: A new strategy for innovation*. New York: Penguin.

Adner, R., & Kapoor, R. (2010). Value creation in innovation ecosystems: How the structure of technological interdependence affects firm performance in new technology generations. *Strategic Management Journal*, *31*(3), 306–333.

Arthur, W. B. (1996). Increasing returns and the new world of business. *Harvard Business Review*, *74*(4), 100–109.

Baldwin, C. Y., & Clark, K. B. (2000). *Design rules, Vol. 1: The power of modularity*. Cambridge, MA: MIT Press.

Baldwin, C. Y., & Woodard, C. J. (2010). The architecture of platforms: A unified view. In A. Gawer (Ed.), *Platforms, Markets and Innovation*. Cheltenham, UK: Elgar.

Bresnahan, T. F., & Greenstein, S. (1999). Technological competition and the structure of the computer industry. *Journal of Industrial Economics*, *47*(1), 1–40.

Brusoni, S., & Prencipe, A. (2013). The organization of innovation in ecosystems: Problem framing, problem solving, and patterns of coupling. In R. Adner, J. Oxley & B. S. Silverman (Eds.), *Collaboration and competition in business ecosystems*. Advances in Strategic Management (Vol. 30, pp. 167–194). Bingley, UK: Emerald Group Publishing Limited.

Canalys (2004) *Global smart phone shipments treble in Q3 - Worldwide handheld market falls, but growth continues outside US*, press release, Canalys Ltd., October 27, 2004.

Chambers, T. (2006). *How smartphones work: Symbian and the mobile phone industry*. Chichester, UK: Wiley.

Cusumano, M. A. (2004). *The business of software*. New York: Free Press.

Dano, M. (2001). Is Symbian OK? *RCR Wireless News*, April 16, p. 1.

Eisenhardt, K. M. (1989). Building theories from case study research. *Academy of Management Review*, *14*(4), 532–550.

Eisenmann, T. R. (2007). Managing proprietary and shared platforms: A life-cycle view. Working Paper No. 07-105; Harvard Business School. Retrieved from http://ssrn.com/abstract=996919.

Farrell, J., & Klemperer, P. (2007). Coordination and lock-in: Competition with switching costs and network effects. In M. Armstrong & R. K. Porter (Eds.), *Handbook of Industrial Organization* (Vol. 3, pp. 1967–2072). Amsterdam: North Holland.

Frankort, H. T. W. (2013). Open innovation norms and knowledge transfer in interfirm technology alliances: Evidence from information technology, 1980–1999. In R. Adner, J. Oxley & B. S. Silverman (Eds.), *Collaboration and competition in business ecosystems*. Advances in Strategic Management (Vol. 30, pp. 239–282). Bingley, UK: Emerald Group Publishing Limited.

Gabel, H. L. (1987). Open standards in computers: The case of X/OPEN. In H. Landis Gabel (Ed.), *Product standardization and competitive strategy*. Amsterdam, The Netherlands: North-Holland.

Gallagher, S., & West, J. (2009). Reconceptualizing and expanding the positive feedback network effects model: A case study. *Journal of Engineering and Technology Management*, *26*(3), 131–147.

Gawer, A. (2010). The organization of technological platforms. In N. Phillips, G. Sewell & D. Griffiths (Eds.), *Research in the Sociology of Organizations*, (vol. 29, pp. 287–296).

Gawer, A., & Cusumano, M. A. (2002). *Platform leadership: How Intel, Microsoft, and Cisco drive industry innovation*. Boston, MA: Harvard Business School Press.

Gawer, A., & Henderson, R. (2007). Platform owner entry and innovation in complementary markets: Evidence from Intel. *Journal of Economics and Management Strategy*, *16*(1), 1–34.

Grove, A. S. (1996). *Only the Paranoid survive: How to exploit the crisis points that challenge every company and career*. New York: Doubleday.

Iansiti, M., & Levien, R. (2004). *The Keystone advantage: What the new dynamics of business ecosystems mean for strategy, innovation, and sustainability*. Boston, MA: Harvard Business School Press.

Katz, M. L., & Shapiro, C. (1985). Network externalities, competition, and compatibility. *American Economic Review*, *75*(3), 424–440.

Kenney, M., & Pon, B. (2011). Structuring the smartphone industry: Is the Mobile Internet OS Platform the key? *Journal of Industry, Competition and Trade*, *11*(3), 239–261.

Laffan, L. (2011). *Open Governance Index: Measuring the true openness of open source projects from Android to Webkit*. London: Vision Mobile Ltd.

Mäkinen, S. J., & Dedehayir, O. (2013). Business ecosystems' evolution – An ecosystem clockspeed perspective. In R. Adner, J. Oxley & B. S. Silverman (Eds.), *Collaboration and competition in business ecosystems*. Advances in Strategic Management (Vol. 30, pp. 99–125). Bingley, UK: Emerald Group Publishing Limited.

Markoff, J. (1998). Microsoft memo offers a glimpse of Gates 2.0, *New York Times*, Oct. 12, p. C1.

Maula, M., Keil, T., & Salmenkaita, J.-P. (2006). Open innovation in systemic innovation contexts. In H. Chesbrough, W. Vanhaverbeke & J. West (Eds.), *Open innovation: Researching a new paradigm* (pp. 241–257). Oxford: Oxford University Press.

Mintzberg, H. (1978). Patterns in strategy formation. *Management Science*, *24*(9), 934–948.

Morris, B. (2008). Platform security and Symbian Signed: Foundation for a secure platform," white paper, Symbian Developer Network Version: 1.4, January.

Morris, C. R., & Ferguson, C. H. (1993). How architecture wins technology wars. *Harvard Business Review*, *71*(2), 86–96.

Nathani, P. (2008). "Happy Birthday to PyS60!" forums.nokia.com. Retrieved from http://blogs.forum.nokia.com/blog/pankaj-nathanis-forum-nokia-blog/2008/12/26/happy-birthday-pys60. Accessed on December 26.

Northam, P. (Ed.). (2006). *How smartphones work: Symbian and the mobile phone industry*. Chichester, UK: Wiley.

Orlowski, A. (2010). Symbian's secret history: The battle for the company's soul; How Nokia took charge, and never let go, The Register. Retrieved from: http://www.theregister.co.uk/2010/11/29/symbian_history_part_two_ui_wars/. Accessed on November 29.

O'Mahony, S. (2003). Guarding the commons: How community managed projects protect their work. *Research Policy*, *32*(7), 1179–1198.

Prahalad, C. K., & Bettis, R. A. (1986). The dominant logic: A new linkage between diversity and performance. *Strategic Management Journal*, *7*(6), 485–501.

Price, C. (1999). Symbian partners exploit Epoc to transform phones, *Financial Times*, June 2, p. 1.

Scheible, J., & Tuulos, V. (2007). *Mobile Python: Rapid prototyping of applications on the mobile platform*. Chichester, UK: Wiley.

Simcoe, T. (2006). Open standards and intellectual property rights. In H. Chesbrough, W. Vanhaverbeke & J. West (Eds.), *Open innovation: Researching a new paradigm* (pp. 161–183). Oxford: Oxford University Press.

Stuermer, M., Spaeth, S., & von Krogh, G. (2009). Extending private-collective innovation: A case study. *R&D Management*, *39*(2), 170–191.

Symbian (2008a). Symbian @ 10: History of Symbian, Retrieved from http://tenyears.symbian.com/timeline.php, Accessed June 23, 2008.

Symbian (2008b). Symbian reports first quarter results for 2008, press release, May 20. Retrieved from http://www.symbian.com/news/pr/2008/pr20089950.html, Accessed on July 9, 2008.

Tasker, M. (2000). A new EPOC. In *Professional Symbian Programming: Mobile Solutions on the EPOC Platform* (Chapter 1). Birmingham, UK: Wrox Press.

Teece, D. (1986). Profiting from technological innovation: Implications for integration, collaboration, licensing and public policy. *Research Policy*, *15*(6), 285–305.

West, J. (2003). How open is open enough? Melding proprietary and open source platform strategies. *Research Policy*, *32*(7), 1259–1285.

West, J. (2014). Challenges of funding open innovation platforms: Lessons from Symbian Ltd. In H. Chesbrough, W. Vanhaverbeke & J. West (Eds.), *Open innovation: New research and developments*. Oxford: Oxford University Press.

West, J., & Dedrick, J. (2000). Innovation and control in standards architectures: The rise and fall of Japan's PC-98. *Information Systems Research*, *11*(2), 197–216.

West, J., & Dedrick, J. (2001). Open source standardization: The rise of Linux in the Network Era. *Knowledge, Technology & Policy*, *14*(2), 88–112.

West, J., & Mace, M. (2010). Browsing as the killer app: Explaining the rapid success of Apple's iPhone. *Telecommunications Policy*, *34*(5-6), 270–286.

West, J., & O'Mahony, S. (2008). The role of participation architecture in growing sponsored open source communities. *Industry & Innovation*, *15*(2), 145–168.

Whitney, L. (2011). *Report: Apple remains king of app-store market, CNET News.com*. Retrieved from http://news.cnet.com/8301-13579_3-20032012-37.html. Accessed on February 11.

Wood, D. (2002). "Symbian Developer Expo 2002 - in context" internal presentation, Symbian Ltd., London, April 2002.

Wood, D. (2003). "Partnering goals, 1H 2003" internal document, Symbian Ltd., London, January 2003.

Wood, D. (2005). *Symbian for smartphone leaders: principles of successful smartphone development projects*. Chichester, UK: Wiley.

Yoshizawa, M., Ichikawa, Y., & Kogetsu, Y. (2006). Expansion of "MOAP" software platform for mobile terminals. *NTT DoCoMo Technical Journal*, *8*, 15–18.