

Windows Phone, Doomed or Pushed to fail?

A comprehensive analysis of smartphone market with respect to operating systems and scenario analysis for Microsoft to investigate the possibility of better future.

By

Anahita Didari

Thesis Submitted to the Department of Geography in Partial Fulfillment of the Requirements for the Degree of Master of Philosophy in System Dynamics



System Dynamics Group
Department of Geography
University of Bergen

June, 2017

Abstract:

The whole definition of communication has been changed since the time smartphones introduced to the world. Now these small devices play crucial roles in daily life of people around the world and lots of interactions are happening through them. This means there is a huge opportunity for smartphone manufacturers and operating system developers.

Since 2000 until now this market has evolved a lot, operating systems that used to be leaders in the market totally lost their edges and left the market while others join. Microsoft was one of the first entrant in this market and was able to maintain in a good position for a short period of time. This was predictable since Microsoft has a great success and reputation in providing OS for personal computers. Windows Phone had a very good start and a position in the market from 2003-2006 but after that its market share started to decrease and today it's below 1%.

In this thesis we explore reasons behind Windows Phone's failure. In order to be successful, first we focus on the smartphone market and make a comprehensive model of this market without any biases toward Windows Phone. Later we focus on the weak points of Windows Phone according to the model and apply scenario analysis to find out in what circumstances Windows Phone could end up in a better position.

Many endogenous dynamics are happening in this market through factors such as bandwagon effects, network diversity, complementary goods effects and brand loyalty effects. These factors help operating systems to build up their market, meanwhile the OS owners have important decisions to make. Strategic decisions such the license fee of the OS and the level of authority they give to manufacturers to customize the OS based on their needs. These are two very important decisions and have significant impacts on the market share of the OS.

Microsoft's business model in smartphone market is a traditional software business, and as the main source of revenue, Microsoft license Windows Phone to any smartphone manufacturers [1]. Therefore from the beginning they charged manufacturers, further Windows Phone is a closed-source operating system and manufacturers has no authority in

customizations. These factors together with the endogenous dynamics are main reasons behind Windows phone's failure.

We implement scenario analysis to investigate results of this competition for Windows Phone under different circumstances and various combination of appropriability and flexibility strategies. Our findings show that by changing appropriability strategy and providing Windows Phone free of charge for device manufacturers, Windows Phone could end up in a better position. But android was still a very powerful rival, because it is also an open-source operating system and it is a very important factor in attracting manufacturers. Further we prove that by changing this strategy at this moment, Windows phone maybe able to increase its market share from 1% to 5% but this is still a very small portion of the market and Microsoft needs an extensive plan and policies in this market if they intend to win.

ACKNOWLEDGEMENT

I would first like to thank my thesis advisor Associate Professor Muhammad Azeem Qureshi at Oslo and Akershus University Collage of Applied science for his supervision and advice during the preparation this research. He consistently allowed this thesis to be my own work, but steered me in the right the direction whenever he thought I needed it.

I would also like to acknowledge Professor Pal Davidsen, Professor David Wheat and Professor Erling Moxnes from the System dynamics group at University of Bergen for their great guidance and help during this thesis also for helping and educating me to become a part of system dynamics society.

I am also grateful to William A. Schoenberg, Aklilu T. Tadesse and Abotsi Prince for their insights and guides through modeling and writing process of this thesis.

Finally, I must express my very profound gratitude to my parents and to my sister for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis.

This accomplishment would not have been possible without them.

Thank you
Anahita Didari

Contents

Abstract:	II
ACKNOWLEDGEMENT	IV
1. Problem articulation	1
1.1. Theme selection:	1
1.2. Time horizon and dynamic problem definition:	2
1.3. Key variables:	4
1.3.1. Innovation and market diffusion:	4
1.3.2. Format battle:	4
2. Formulation of dynamic hypothesis	9
2.1. Initial hypothesis generation:	9
2.2. Endogenous explanation:	10
2.3. Mapping system structure:	12
2.4. Casual loop diagram:	13
2.4.1. Technology diffusion section:	14
2.4.2. OS market dynamics:	15
2.4.3. Operating System section:	16
2.5. Stock and Flow maps:	20
3. Formulation and results of the simulation	24
3.1. Formulation of the simulation	24
3.1.1. Market Section:	25
3.1.2. Operating system module:	31
3.2. Results of simulation:	65
3.2.1. Market section:	65
3.2.2. Operating system section:	69
4. Testing:	76
4.1. Boundary adequacy test:	76
4.2. Dimensional consistency:	77
4.3. Extreme condition test:	77
4.3.1. Adaption fraction:	77
4.3.2. Time to change:	78
4.3.3. Open Strategy extreme test:	80
4.4. Integration error test:	81

4.5.	Behavior reproduction:	82
4.6.	Behavior anomaly tests:	86
4.7.	Sensitivity test:	87
4.7.1.	Adaption Fraction sensitivity test	88
4.7.2.	Contact rate sensitivity test:	89
4.7.3.	Attribute weight sensitivity test	90
5.	Scenario Analysis:	92
5.1.	Windows Phone's strategies and performances:	92
5.2.	Scenarios:	94
5.2.1.	Scenario 1:	95
5.2.2.	Scenario 2:	96
5.2.3.	Scenario 3:	98
5.2.4.	Scenario 4:	99
5.2.5.	Scenario 5:	100
6.	Conclusions and recommendations:	102
	References:	106
	Equations:	108

1. Problem articulation

1.1. Theme selection:

Since Motorola released the first mobile phone in 1983 the mobile market has changed a lot. These changes include both technological improvement of the devices as well as changes in the market. One of the important events in this market was when smartphones entered the market, Smartphones are complex mobile phones which have some functionality of personal computers. Therefore one can use them both for voice calls or text messages, as well as to watch videos, listen to music, take photos and to run variety of applications. The development of mobile internet services represents an important milestone in the history of smartphones, as it constituted the main trigger for the introduction of devices that allowed a full convergence between computing and communication [2].

Smartphones are complicated devices like mini computers. The increase of device sophistication is challenging for the software running on the top of it. The OS (Operating system) is the heart of a smartphone software system. The OS determines features, performance, security and add-on applications of the smartphone[1] . According to Grazia Cecer, the turning point in the smartphone industry occurred between 2006 and 2007. .At the end of 2006, RIM launched a device for the business world based on BlackBerry, which enabled email and instant messaging, and HTML browsing. This was Curve 8100, which was aimed exclusively at business people. Apple entered this market segment in the beginning of 2007, when the first iPhone was announced to the world by Steve Jobs. The iPhone disrupted the traditional market concept by integrating the new phone with the OS, and the browser –Safari- and the iTunes store for downloading audio and video content. Even though it was not the first smartphone in the market, it soon became a point of reference for all products in the coming year in terms of design and user interface[2].

European companies did not respond to Apple’s challenge and continued to produce Symbian smartphones. Other competitors started to imitate and improving iPhone concepts,

Samsung and LG had accumulated competencies in touch screen phones, and of BlackBerry (RIM) and they had introduced the first touch screen device in 2008. From 2009, competition increased substantially, following the introduction of many devices. Which used the Android open source OS by different mobile device manufacturers [2].

The smartphone market is constantly evolving. E.g., the market leaders in 2003 were Microsoft, Palm and Symbian OS and Linux, which are completely different from those of today [1]. Microsoft entered the smartphone market in 2000, later in 2003 “Windows Mobile” became official. In 2010 “Windows Mobile” replaced by “Windows Phone” the name we recognize today. The market of Windows Phone has lots of up and downs and this was unpredictable, because Microsoft has a great reputation for providing Operating system of personal computers. Therefore, in this thesis we aim to find out why Microsoft lost this battle to Google (Android owner) and Apple (iOS owner) even though it was an early entrant with great deal of experience.

For analyzing and exploring the reason behind Microsoft’s low market share, we focus on two main areas. First we make a comprehensive model to understand dynamics of the smartphone market and find out reasons behind success of others. Second we apply scenario analysis for Windows Phone in order to investigate the possibility of better future for it.

1.2. Time horizon and dynamic problem definition:

Although the battle for a dominant position in the smartphone market increased in intensity from 2008, Microsoft entered the market in 2003 which means we need to start years before the competition reach its peak. Therefore, we chose the time horizon from 2000 to 2016, this way we can show the rise of all available operating systems in the market and analyze their effects on one another. Figure 1 shows the historical data of the market share in this market in time between 2000 and 2016.

The purpose of this thesis is to addresses three important issues:

(1) Understand the dynamics of smartphone market with respect to operating systems. Also identifying the major factors that have influenced this market over time.

(2) Using system dynamics to investigate the interaction between these factors over time and comparing simulation results with the historical data.

(3) Implementing scenario analysis for Windows Phone to find out how they could secure a better position in this market.

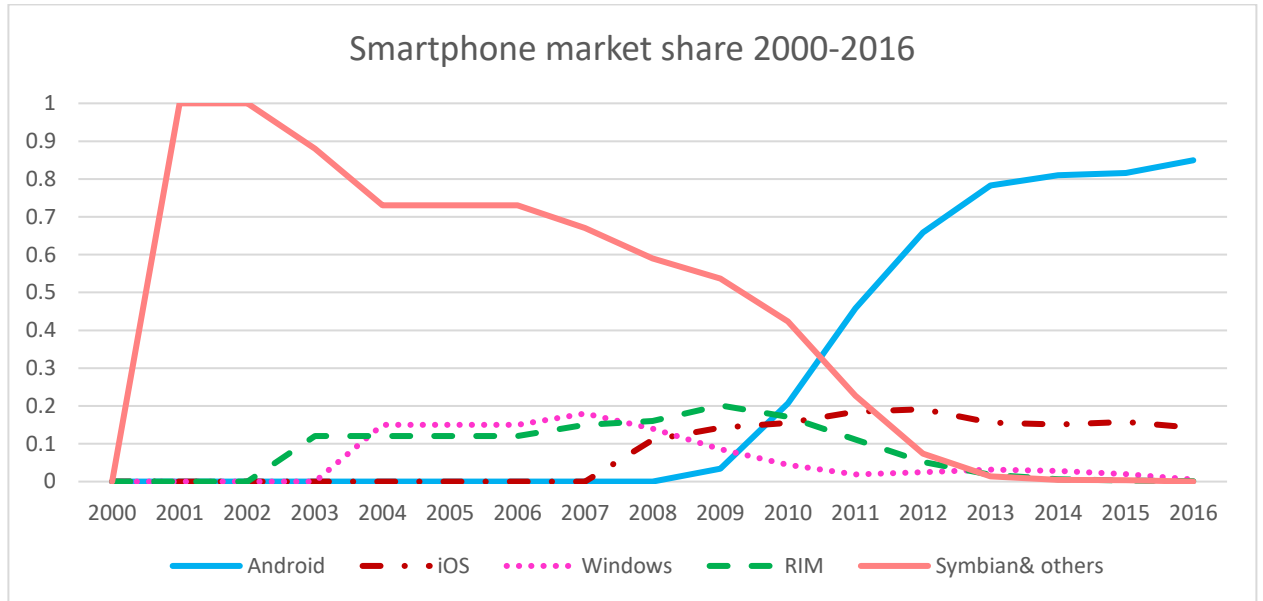


Figure 1 Time graph of Smartphone market share 2000-2016

The dynamics of the OS development is directly connected with the innovation diffusion of smartphones. In the beginning of 2000 there were lots of operating systems in the market and almost all phone manufacturers had their own small operating system. After Android introduced to the market due to its attractiveness, device manufacturers abandon their OS and adapted to it. Android has been the leader of the market for years, also a great portion of profit belongs to Apple even though the market share of iOS has not increased. BlackBerry is out of the market and Windows Phone is trying to innovate to compete with others. IDC (2012) predicts that Android will continue to be the most shipped smartphone operating system over the next five years. Based on the growth of Samsung sales and Google's strategy to allow different smartphone producers to use Android. Despite the decline in the iOS market share, analysts are positive about the future[2].

1.3. Key variables:

In this study two important sets of variables are being considered. First set is related to innovation and market diffusion factors, those explain how the smartphone technology has expanded since introduction of the first mobile. Whereas the second set comprises of the variables that determine format battles to analyze the OS market development over time.

1.3.1. Innovation and market diffusion:

According to John Sterman [3], The diffusion and adoption of new ideas and new products often follows S-shaped growth patterns. The spread of rumors and new ideas, the adoption of new technology, and the growth of new products can all be viewed as epidemics spreading by positive feedback as those who have adopted the innovation “infect” those who have not. New ideas spread as those who believe them come into contact with those who do not and persuade them to adapt the new belief.

In this thesis we use logistic model in order to explain the innovation diffusion. According to John Sterman [3] the logistic model is widely used to explain and predict the diffusion of innovations.. The logistic model often works well because it includes two feedback processes for every growth: a positive feedback loop that generates the initial period of accelerating growth and a negative feedback that causes the growth to slow as the carrying capacity is being approached. In most cases word of mouth is not the only positive feedback loop in the system, but in this thesis use logistic model for innovation diffusion and consider other factors in the format battle.

1.3.2. Format battle:

In 2011 Geerten van de Kaa [4] proposed a framework in order to help understanding and analyzing format battles. This framework includes 29 factors in five category of characteristics. It is important to know as he asserted “not all factors apply in each battle and per battle the importance of the relevant factors will be different. By applying the framework to different historical cases of format battles, weights for factors might be established”. Table 1, shows all the factors in five categories.

Later in 2014 G. van de Kaa [5] identified four categories of factors for standard dominance that can be directly influenced by the firm. These categories are characteristics of the standard, other stakeholders, characteristics of the standard supporters, and standard support strategy. In this study G. Van de Kaa investigated the usability of a multi attribute utility approach named fuzzy analytics hierarchy process to determine the relative weight of factors for standard dominance. In the following paragraphs we examine these factors.

Table 1

29 factors in format battle (G. Van de Kaa)

Factors		Factors	
	<i>Characteristics of the format supporter</i>		<i>Other stakeholders</i>
1	Financial strength	16	Current installed base
2	Brand reputation and credibility	17	Previous installed base
3	Operational supremacy	18	Big Fish
4	Learning orientation	19	Regulator
	<i>Characteristics of the format</i>	20	Antitrust laws
5	Technological superiority	21	Suppliers
6	Compatibility	22	Effectiveness of the format development process
7	Complementary goods	23	Network of stakeholders
8	Flexibility		<i>Market characteristics</i>
	<i>Format support strategy</i>	24	Bandwagon effect
9	Pricing strategy	25	Network externalities
10	Appropriability strategy	26	Number of options available
11	Timing of entry	27	Uncertainty in the market
12	Marketing Communications	28	Rate of change
13	Pre-emption of scarce assets	29	Switching costs
14	Distribution strategy		
15	Commitment		

A standard that has superior characteristics compared to other standards may have a higher chance of achieving dominance. This superiority may include technological superiority, compatibility, and availability of complementary goods [5]. Schumpeter [6] defines *technological superiority* of a standard as having superior features that makes this standard outperform other standards. However, the most technically advanced or the best standard does not necessary become the dominance one [7]. Another characteristic of a standard that can add to its superiority is the *compatibility* that the standard enables. Standards can be designed in such a way that they are backward compatible with the previous generation of the standard so that products that implement an old generation of

the standard can still be used together with products that implement the new generation of the standard [8]. Teece [9] defines *complementary goods* as those other goods needed to successfully commercialized a certain standards.

The second group of factors related to stakeholders other than the main standard supporter that affect the outcome of the standard battle [5]. The *current installed base* is a number of current users of a particular standard, when the number increase it will affect the format adaptions in a positive way. A self-reinforcing pattern can raise through network externalities, resulting in an initial advantage for the standard to achieve dominance [5]. *The Effectiveness of the format development process*, can be affected by the decision rules and processes, process management and stakeholders' involvement and all these factors can affect the duration and quality of resulting specifications. *Diversity of the network* refers to the extent to which relevant stakeholders are represented in the group of standard supporters. A standard that is supported by a diverse network in which stakeholders represent each relevant product market for which the standard serves a defining role will have a higher chance of achieving dominance [10, 11].

The third group is related to how strong the format supporter is and act in the market. *Financial strength* of the standard supporter is related to the current financial condition of the standard supporter and its future prospects, and positively effects standard dominance [12]. The brand reputation and credibility refers to the opinion people have about a group of standard supporters, based on what happened in the past. This plays a significant role in users' selection of the standard, since past performance in setting dominant standards has a positive impact on expectation of new proposals [13]. *Learning orientation*, with learning we refer both to the know-how; the core capabilities and the extent to which the firm can acquire new knowledge-absorptive capacity. The absorptive capacity refers to both technological know-how and market pioneering know-how [12].

According to G. Van de Kaa [5], a standard support strategy contains the range of strategies adopted in a market to win a standard battle. The regime of *appropriability* has been defined with respect of the commercial environment, excluding firm and market

structure, that govern a firm’s ability to capture the rent associated with an innovation [9]. In addition to price, a firm’s licensing policy has also been identified as a key driver in managing the relationship with producers of complementary goods. In the most extreme case, a firm may decide to make its technology completely available for free-“an open standard”- as did IBM when it entered PC market [14]. Timing of entry, for the particular context of technological battles, entry timing has been associated both with market entry and with R&D pioneering the start of systematic R&D activities [15]. Apparently early entrant has positive effects in helping to build a larger network of users also it increases learning within the firms, however there are some negative effects as well it can lock firm into particular technology and also if the time is so early the available market can be small. Christensen et al [16] argue that in fast-paced industries, very early entry does not lead firm to maximize their survival chances; there are maximized when firms enter the industry in a few years just prior to the emergence of dominance technology. Customer expectations play an important role in standard battles [17] and therefore, marketing communications are important to gain more market share.

Using fuzzy analytic hierarchy process, Van de Kaa [5] proposed a weights for each of the factors that influence the chance that standard achieve dominance.

Table 2
Factors for standard dominance ranked by importance

Factors	Ave. Weight	Factors	Ave. Weight
Diversity of the network	0.1897	complementary	0.0542
brand reputation	0.1825	marketing communications	0.0442
financial strength	0.1159	technological superiority	0.0437
current installed base	0.0904	effectiveness of development process	0.0424
timing of entry	0.0874	appropriability strategy	0.0422
compatibility	0.0685	learning orientation	0.0398

In 2015 G. van de Kaa proposed a model for standard dominance for converged systems [18]. “Six sets of factors for standard dominance were important in each technology battle

that we studied: complementary assets and strategies, size and diversity of the inter-organizational network of standard, commitment of the group of standard supports, availability and variety of complementary goods, installed base, and market mechanisms.”

Figure 2 shows this factors and their relationships.

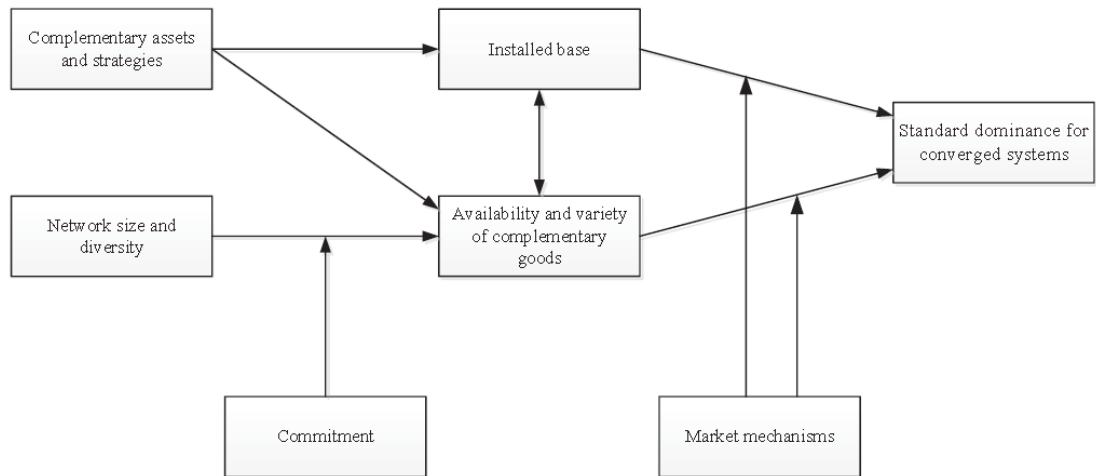


Figure 2 G. van de Kaa’s revised model for standard dominance of converged systems

In this thesis we use the combination of “four categories of factors for standard dominance that can be directly influenced by the firm” and “the model for standard dominance of converged systems”.

2. Formulation of dynamic hypothesis

2.1. Initial hypothesis generation:

The number of mobile phone users in the world is expected to pass five billion mark by 2019 (Figure 3). Nearly 60 percent of the world population already owned a mobile phone. There are two important events in this market, first the innovation diffusion and second the rise of smartphones. According to Statista.com [19] in 2012, about a quarter of all mobile users were smartphone users and by 2018, this number is expected to double.

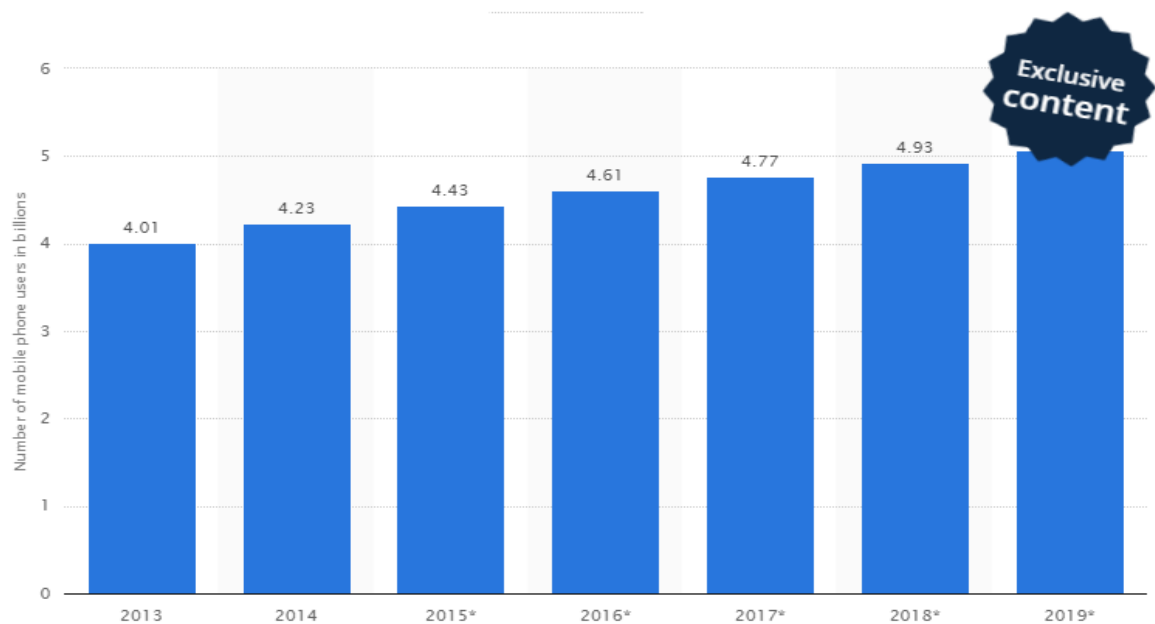


Figure 3 Number of mobile phone users worldwide

In the beginning of 2009 there were at least 6 operating systems available in the market; android, iOS, Symbian, Windows phone, RIM, Bada and ext. Right now there are two main competitors in the market (android and iOS), they can be considered as survivals.

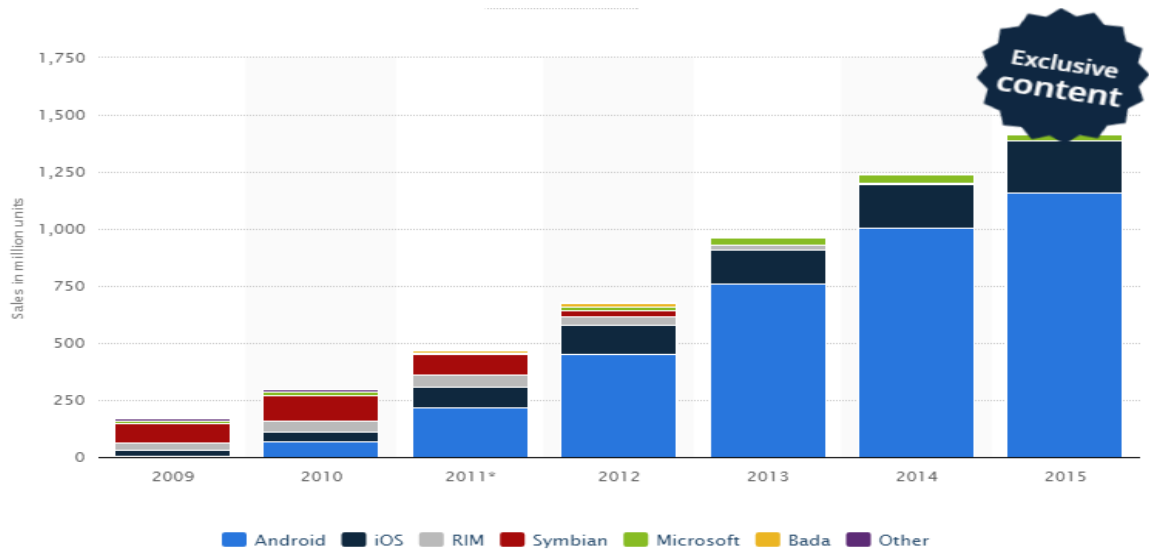


Figure 4 Global smartphone sales

As figure 4 shows, from 2009 to 2015 at least 4 operating systems vanished from the market, more precisely they lost the battle to other operating systems.

Looking through market and history, Google made a huge decision by making android "free-license" an "open-source OS". This strategy helped android to absorb 50 percent of the global smartphone sales by 2011 and increase it in following years. As we mentioned there are many other factors which act endogenously in favor of Android but that first strategy was the initiator of those endogenous dynamic.

2.2. Endogenous explanation:

As Sterman [3], said "system dynamics seeks endogenous explanation for phenomena". In this thesis the focus is on endogenous factors which are rising from inside, also there are other factors in the model which are exogenous or out of the scope of this thesis.

In this model there are 2 different modules, Market and Operating system (Figure 5). These modules are both affect and being affected by each other. In market module we are looking through market expansion via innovation and technology diffusion. "OS attractiveness" which is the main output of the OS module is influencing the market directly.

Sales rate and number of current installed base are two important factors from market module that influence the operating system module.

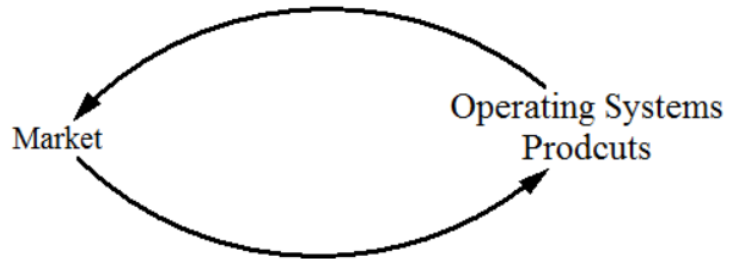


Figure 5 Market and Operating systems modules

Figure 6, shows the big picture of endogenous relations between two modules of this model. This figure shows how the behavior is being generated through the interaction of the variables presented in the model. We describe these loops and their dynamics in the following sections in details.

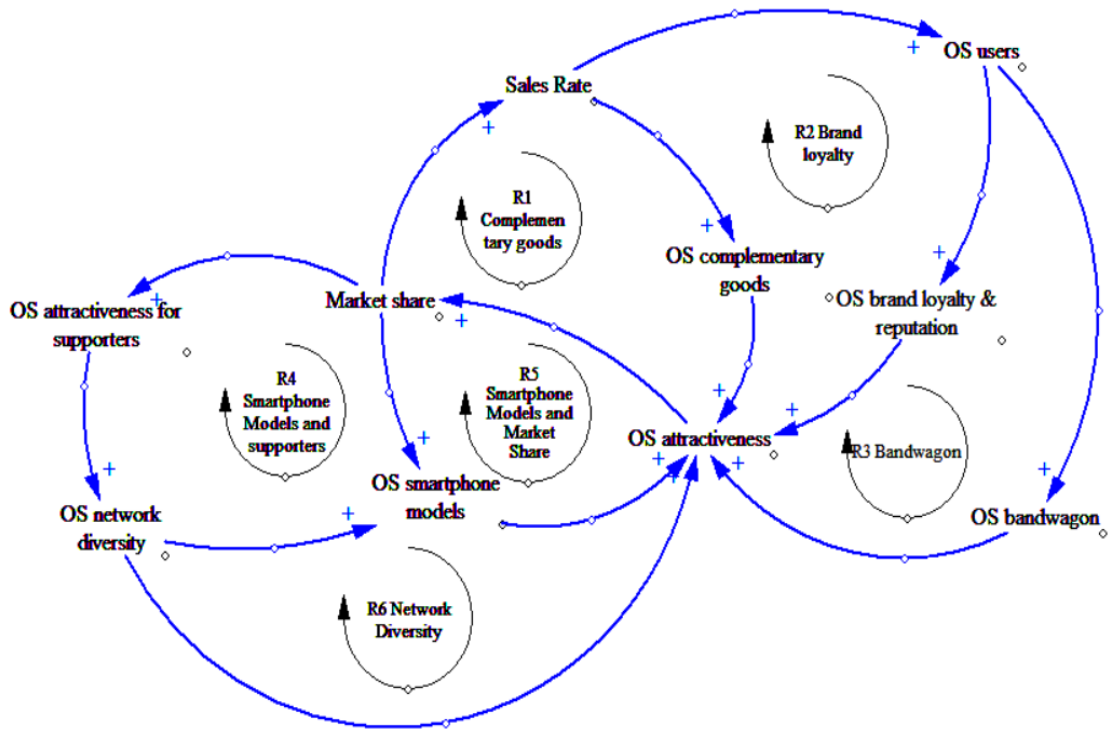


Figure 6 Overall CLD of the model

2.3. Mapping system structure:

Model Boundary Chart: A model boundary chart summarizes the scope of the model by listing which key variables are included endogenously, which are exogenous and which are excluded from the model [3]. In Table 4 you can explore these variables in the model.

Table 3

Model Boundary chart

	Endogenous	Exogenous	Excluded
Characteristics of the standard	Complementary goods		Technological superiority Compatibility
Other stakeholders	Current installed base Diversity of network		Effectiveness of development
Characteristics of standard supporter	Brand reputation		Financial strenght learning orientation
Standard supporter strategy		Appropriability strategy Timing of entry	Marketing Commiunication
Other factors	Smartphone model		Competitor's reactions

Subsystem Diagram: According to Sterman [3], A subsystem diagram show the overall architecture of a model. Subsystem diagrams convey information on the boundary and level of aggregation in the model by showing the number and type of different organizations or agents represented. Figure 7 represents the model boundary of this thesis and interactions between subsystems.

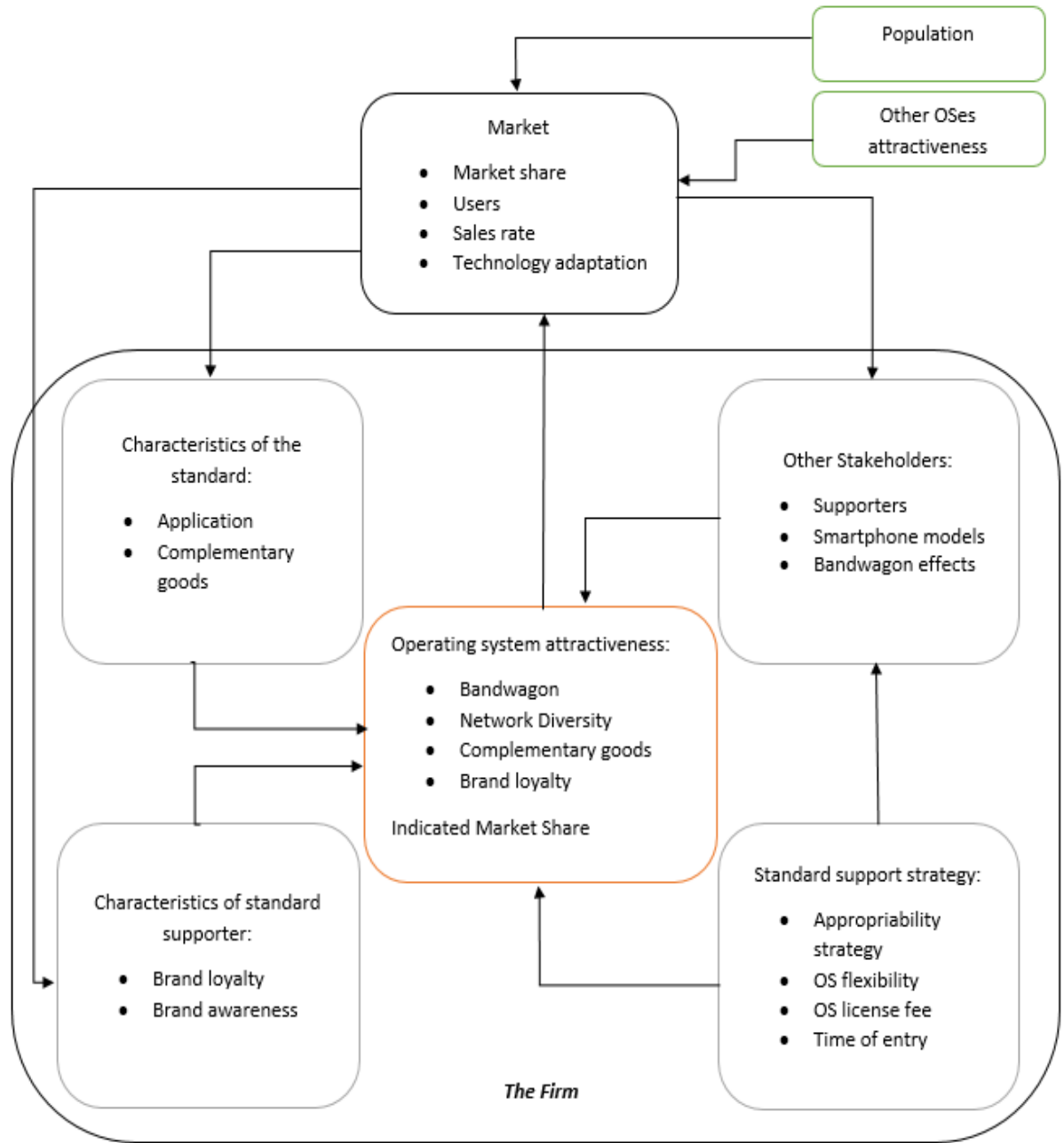


Figure 7 Subsystem diagram of the model

2.4. Casual loop diagram:

Model boundary charts and subsystem diagrams show the boundary and architecture of the model but do not show how the variables are related. Casual loop diagram (CLDs) are flexible and useful tools for diagramming the feedback structure of systems in any domain [3]. In this thesis we consider the proposed CLD [20] and the facts about the smartphone

market to make the CLD of this thesis. The CLD includes three main parts, technology diffusion, OS market dynamics and the third one which includes the contributory factors in OS attractiveness (being presented in 3 separated CLDs).

2.4.1. Technology diffusion section:

Figure 8 shows the development in the market of smartphone and innovation diffusion, this market has expanded since 2000. In this CLD there are two reinforcing loops and one balancing loop.

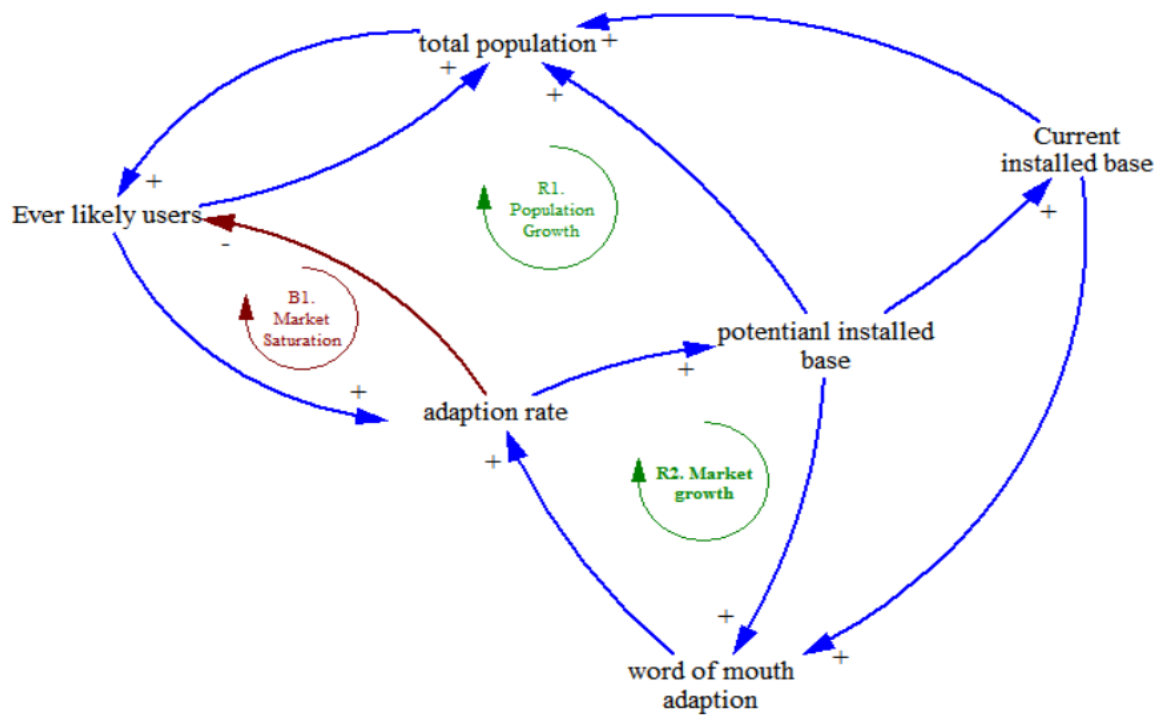


Figure 8 Innovation diffusion casual loop diagram

This loops are:

R1. Population growth. This loop is a simplified version of population growth in the society. It shows that increase in the total population is going to increase the net growth rate of population.

R2. Market growth. Both current installed base and potential installed base are people who are aware of the smartphone technology, and their role in this part of the model is like infectious population. Loop R2 shows how increase in this two variables is going to increase the adaption rate of the technology. Further increase in this rate positively increase both potential and current installed base.

B1. Market saturation. This loop shows increase in the ever likely users who are currently unaffected by the technology has positive effects on the adaption rate. Meanwhile the increase in adaption rate has a negative effect on the ever likely users because it decreases its level.

2.4.2. OS market dynamics:

Figure 9 shows the dynamics between OS attractiveness and the whole market, this CLD includes three important loops:

R3. OS attractiveness and current installed base. Current installed base variable is one of the inputs from market module to operating system module. This loop shows how increase in current installed base have a positive effect on OS attractiveness and market share. Increase in these two has positive impacts on current installed base.

R4. Other OS attractiveness. This loops shows the reinforcing effect of OS attractiveness on users' change rate. When the OS attractiveness is increasing, attractiveness of other OSes will decrease and that positively affect the change rate and it decreases. Since the change rate has negative effect on the current installed base, the overall loop shows a reinforcing behavior.

B3. OS market saturation. In this part of the model, we observe the relation between sales rate and potential users and see that increase in sales rate leads to decrease in potential users. Further decrease in potential users decreases the sales rate consequently.

B4. Other OS attractiveness and sales rate. This loop is a balancing loop that shows the relation between OS attractiveness, change rate and sales rate. As OS attractiveness

increases, the change rate decreases and decrease in change rate means decrease in potential users and further decrease in the sales rate.

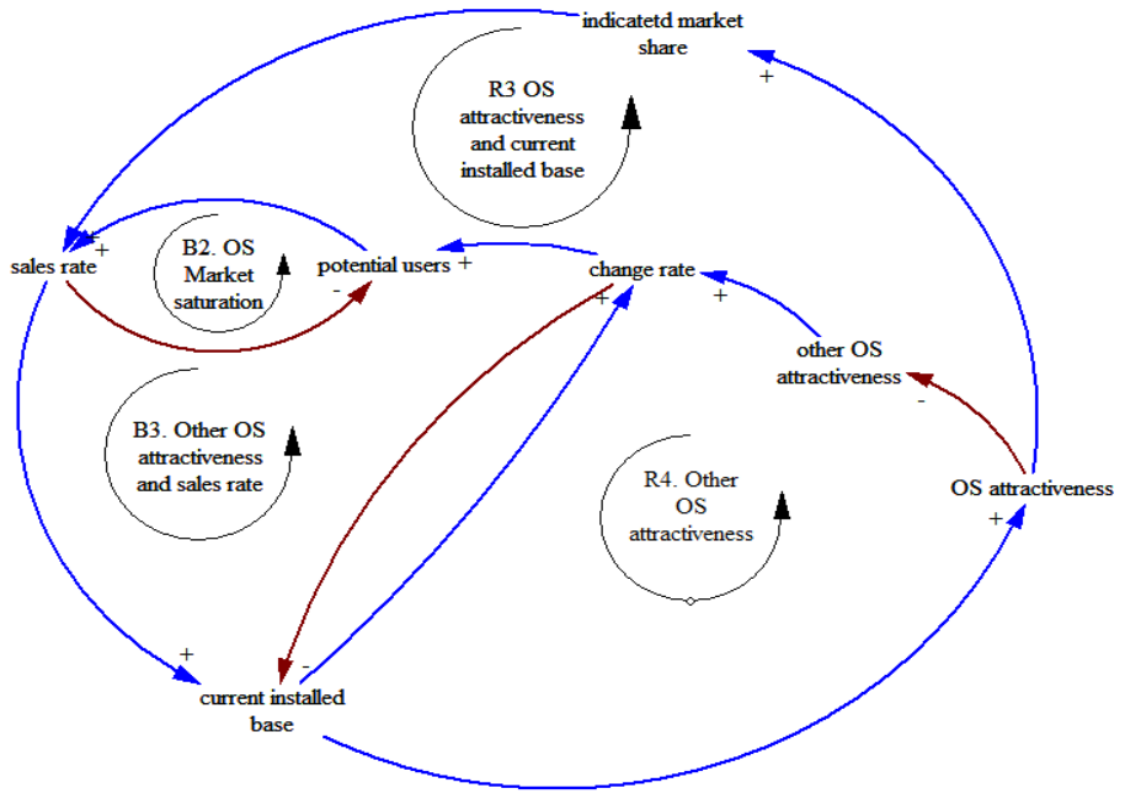


Figure 9 OS market dynamics CLD

2.4.3. Operating System section:

In this section we analyze interactions between 4 main categories of OS characteristics and their impact on the OS attractiveness.

First we explain two related categories, Standard support strategy and other stakeholders.

Figure 10 shows this CLD and it has three reinforcing loops;

market share of OS, increase in market share will positively affect the supports decision to join and produce smartphone with specific OS.

R7. Smartphone supporters: This loop is important because it shows that the increase in the total sales rate also increase the number of manufacturers in the market. So when sales rate increase, more companies are being establish to produce smartphones, and that increases the OS attractiveness and market share.

R8. OS model attractiveness: This loop represents that increase in market share positively affects supporters to produce variety of smartphone models. Variety of smartphone models is important because in most cases this model variety comes with price differentiation. That means wide range of potential users with different financial situations are being attracted to that OS.

Next CLD is related to characteristics of the OS and shown in figure 11, this CLD presents the relation between market share and complementary goods attractiveness, and has two reinforcing loops:

R9. Application attractiveness: This loops shows the interactions between sales rate and developers' decisions to develop applications for the OS. Meanwhile there are other factors that have impacts on this decision such as OS flexibility for developers, registration fee (that developers have to pay in order to publish their applications) and the availability of the app store.

R10. Complementary goods: In this loop the positive relation between the sales rate and number of supporters to produce complementary goods such as smartwatches, smartTVs is being shown. As the number of complementary goods increases so does the OS attractiveness and it will increase the willingness between potential installed bases.

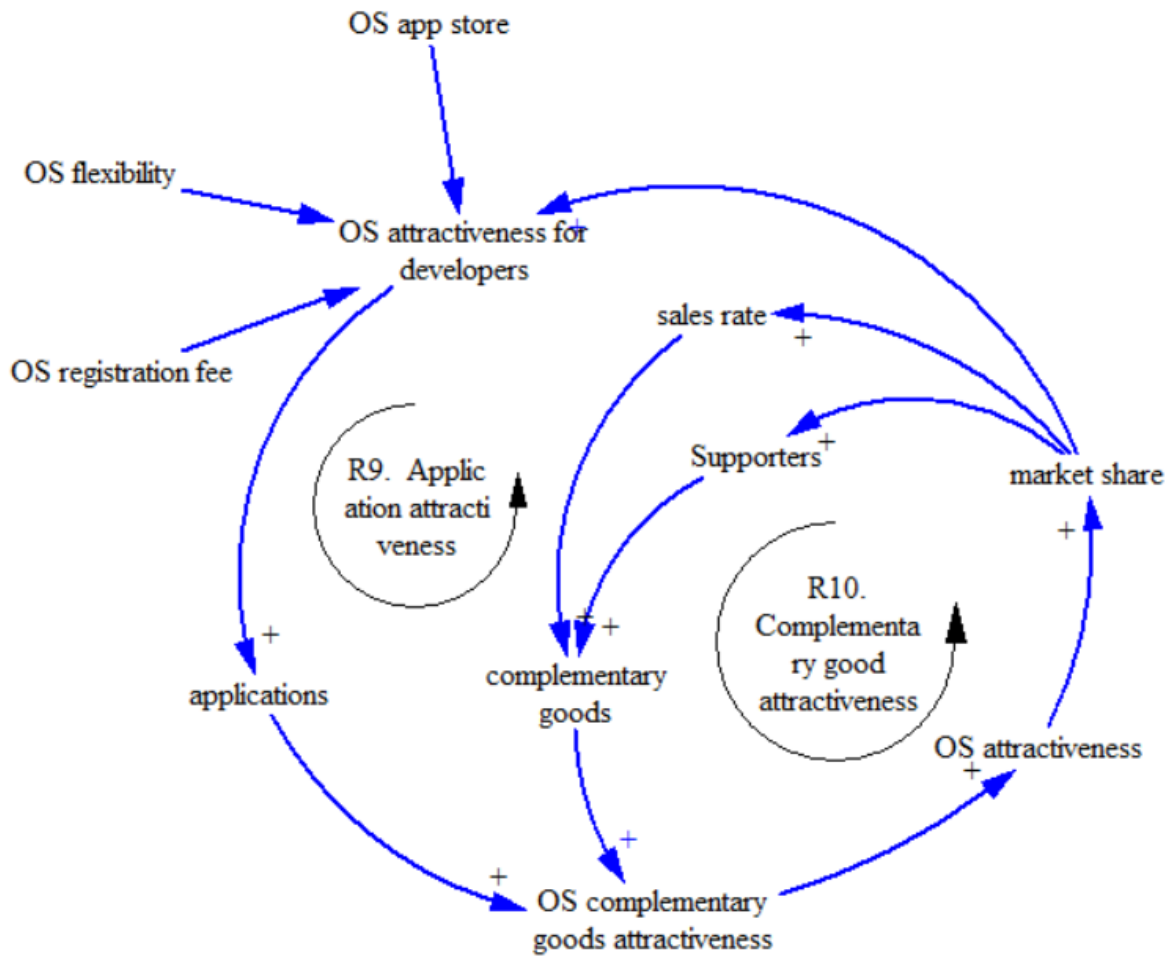


Figure 11 Complementary attractiveness CLD

The last part of OS CLD is related to characteristics of OS supporters; brand loyalty and reputation belongs to this category of characteristics, as figure 12 shows this CLD has two reinforcing loops:

R11. Complementary goods and brand loyalty: This loop shows that the increase in number of complementary goods has positive effects on the brand loyalty and further brand loyalty on OS attractiveness and market share and that leads to more complementary goods production.

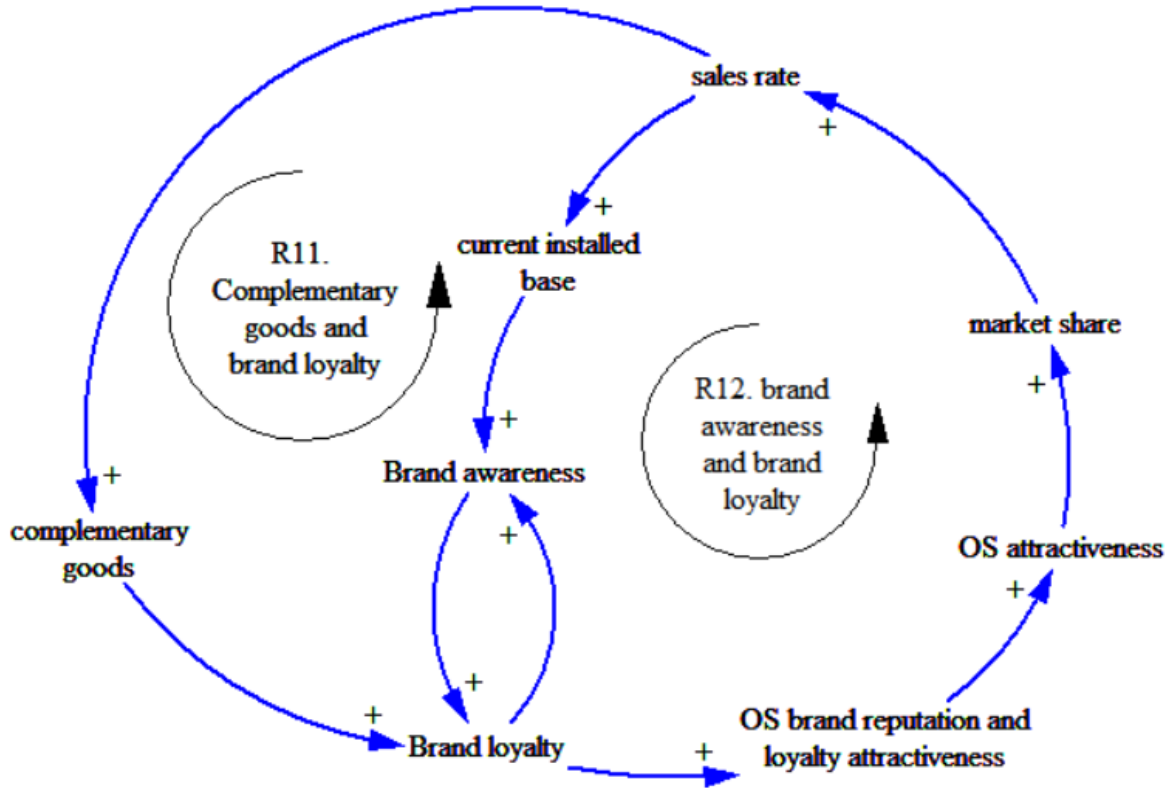


Figure 12 Brand loyalty CLD

R12. Brand awareness and brand loyalty: In this loop we can see how increase in current installed base positively increase the brand awareness and brand loyalty.

2.5. Stock and Flow maps:

To explore the big picture of the model we are using the stock and flow (SFD) map. There are so many different variables in the original SFD, but for purpose of better and clear understanding we present the highly aggregated and simplest version of the model in this section. The idea is to explain the whole model along with analysis in the next chapter.

Since this market includes at least 5 different operating systems, in the SFD modeling part we use arrays in order to avoid unnecessary repetition and complications. Based on Sterman [3] we made a basic structure of smartphone market (Figure 13). Though this is highly aggregated version of the model, this model shows the important feedback loops we described in the previous section.

Upper side of the figure 13 includes the market section of the model, this part shows the dynamics within the market and it has two important loops; Market diffusion and OS market. Population increases in the constant rate, and people who are not aware of the smartphone technology will adapt to this technology as they encounter people who are either using smartphone or undecided. Then they move to potential installed base, who need to make an important decision according to available data in the market and choose an appropriate operating system.

Sales rate is being affected by two main factors; indicated market share and the stock of potential installed base. As the potential installed base increases by the technology adaption rate, the sales rate increases as well. The important variable in the lower part of figure 13 is the OS attractiveness, according to Van de Kaa [5] there are 4 main categories of factors for standard dominance. In this model, complementary goods, current installed base, diversity of network, brand loyalty, appropriability strategy and time of entry are considered. These factors have impacts on the OS attractiveness in two different areas; first for supporter because there are different factors that affect supports to adapt to the specific standard. Second the OS attractiveness for users which includes complementary goods, current installed base, and diversity of network and brand loyalty.

Color coding is used to help readers to follow certain loops, common factors such as OS attractiveness and indicated market share are being connected with ticker and different color to show their multiple considerations.

First important factor in OS attractiveness is bandwagon effect, which means as the number of current installed base increase so does the contact rate between them and potential installed base. This will increase the bandwagon effectiveness of the OS.

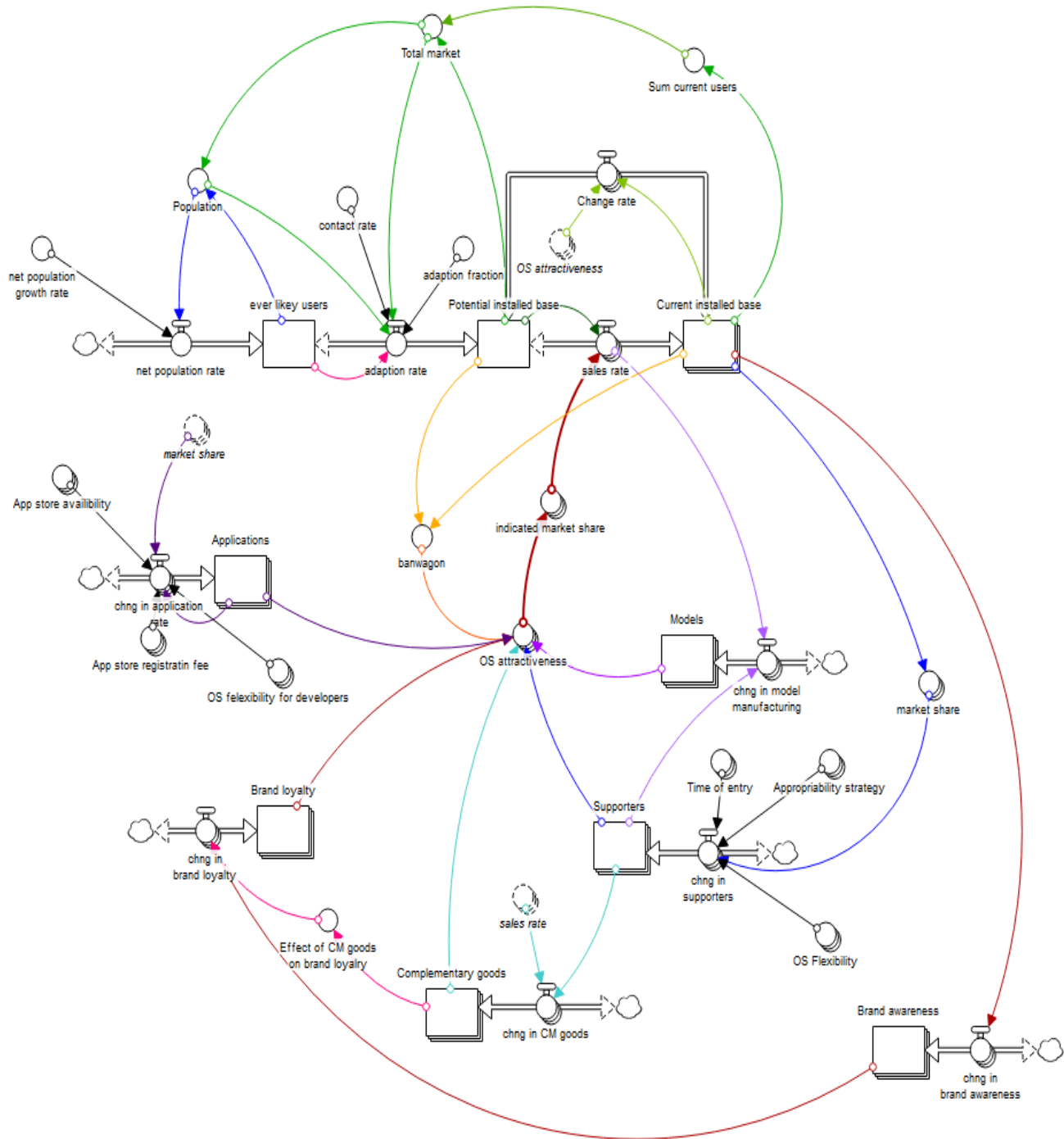


Figure 13 Stock and flow diagram big picture

Operating systems such as iOS and BlackBerry OS are produced to be used by their owners and smartphone manufacturer are not able to use them. Others are available for manufacturers to use but with different levels of customization authority. This factor along with OS flexibility, are factors that influence the decision of manufacturers to adapt to the specific OS. These supporters produce smartphones and complementary goods such as smart watches and smart TVs. Other operating systems just going to have one supporter which is the OS owner as well.

Complementary good includes both applications and goods, there are different factors that affect the developers' decision. Some of these factors such as availability of app stores, OS flexibility for developers and registration fee that needs to be paid in order to publish application are exogenous and change according to companies' policies. Application development also depends on the OS market share, thus increase in market share will increase application development rate. Complementary good also depends on the relative sales rate, and the number of supporters is also an important factor in the producing rate.

Another important part of the model is brand loyalty and awareness, as one can see in the SFD, the stock of current installed base and the brand awareness has a positive relationship so they increase and decrease together. It is worth mentioning that brands such as Google and Apple have some level of awareness before they introduced their operating systems. This awareness or reputation comes from their previous performance on the search engine and laptops production. Brand loyalty increases by the increase in brand awareness as well as the increase in complementary goods. Complementary goods are the category of products that encourage users to stay with a specific OS because they already have different products that are best functional with that OS.

3. Formulation and results of the simulation

3.1. Formulation of the simulation

In this chapter we use stock and flow diagrams to explain the structure of the model. We present and describe each sector in this model separately, also we show the feedback loops which we explained in the previous chapters. In each section we will explain the endogenous dynamics of the model as well as the assumptions made for each section.

It is worth mentioning that we are analyzing the market for 5 different OS; Symbian, BlackBerry, Windows phone, iOS and Android. The structure of market for all 5 OSes are quite similar, therefore in order to avoid unnecessary repetition we are using “array” function in Stella Architect. Figure 14 shows that how those variables and stocks that are containing different values for operating systems are being represented in the model.

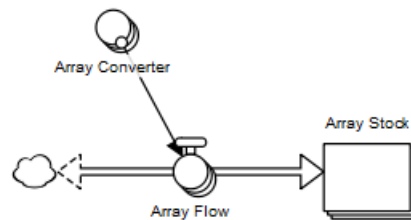


Figure 14 Array variables

Further in stock and flow diagrams reinforcing and balancing loops that we mentioned before are being shown. There are some other loops that were not major so we present those in the SFDs without numbers.

Also since there are important relations between this module and Market module we added variables and arrows just to show the dynamic in the SFD, these variables are located out of the sector. Since these relations are indirect we showed them with delay mark to avoid confusions.

3.1.1. Market Section:

Market section includes seven important loops, these loops are; Market growth, market saturation, OS attractiveness and current installed base, and others. To be able to focus clearly on different part of the model, we divided this section to two smaller yet important parts.

3.1.1.1. Market section part 1:

This part of market section represent the three main loops in the model: Market growth, market saturation and population growth. Figure 15 shows the relation between Current installed base, potential installed base and ever-likely users. In the following paragraphs we explain each one of these variables and the relation between them.

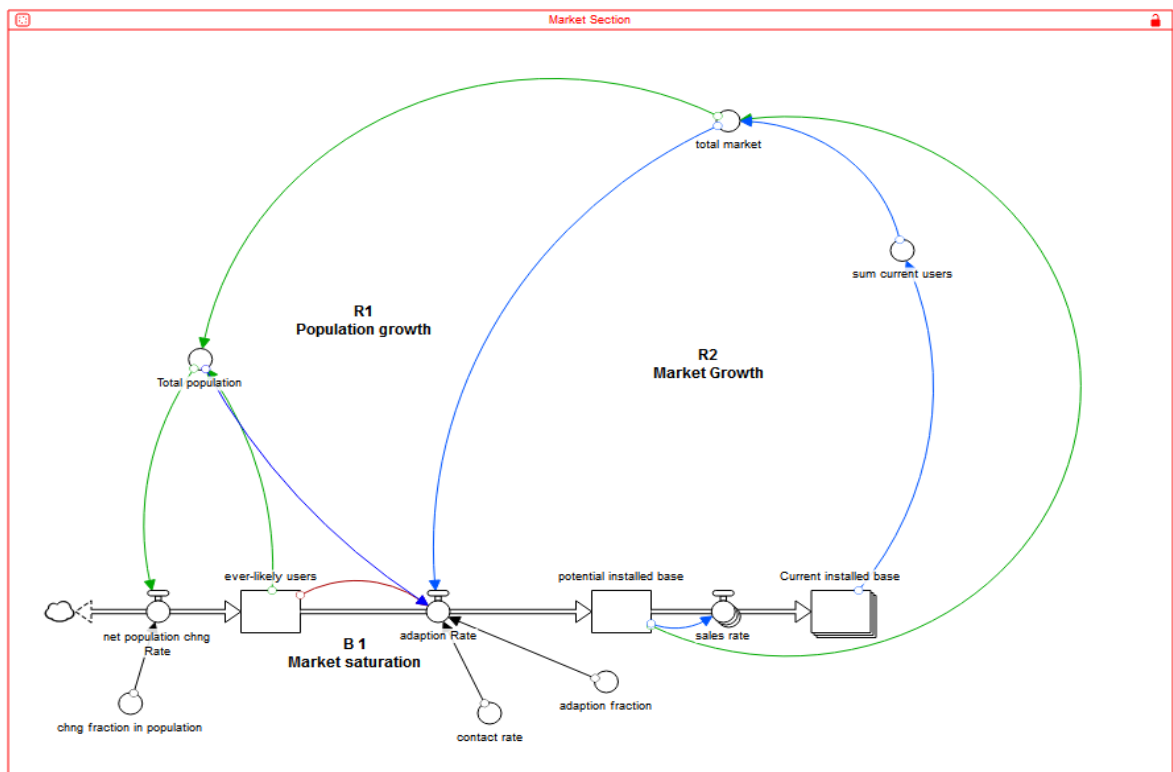


Figure 15 SFD of market section (part 1)

Ever-likely users (EU), presents the actual number of people in the world who are not aware of the smartphone technology. This stock increases as the total population in the world increases.

$$EU = \int (PR - AR)dt + EU(0) \quad (1)$$

This stock accumulates the difference between net population change rate, PR, and adaption rate, AR. The initial value for this stock is 5.9 billion people.

Net population change rate (PR), depends on total population and net population growth rate, Since this model's main concern is not population growth, we just used the average population net growth in the world to show the development of population in the world, the average net population growth is 1.2% [21].

$$PR = P * FP \quad (2)$$

Where P is the total population and FP is the net population fraction rate which is equal to 1.2%.

Total population (P), is the sum of the Total market (TM) and Ever likely (EU) stocks, this value shows the approximately number of people in the world. This variable is important and being used in other part of the model such as calculating total awareness.

$$P = TM + EU \quad (3)$$

Total market (TM) is a variable that shows the number of people who are aware of the smartphone technology, either they are using the technology or they have not made their choice about it.

$$TM = SCI + PI \quad (4)$$

Where SCI is the sum of current installed base and PI is the potential installed base.

As we mentioned in this model we are using arrays to show different operating systems, for showing technology diffusion we need the sum of all current installed bases (CI), Sum of current installed base (SCI) shows this variable.

$$SCI = Sum (CI) \quad (5)$$

Adaption rate (AR) shows how ever-likely people become aware of the smartphone technology and move to potential stock. This equation is same as logistic model, where AF is adaption fraction, CR is contact rate.

$$AR = \frac{AF*CR*EU*TM}{P} \quad (6)$$

AF is adaption fraction of technology which is constant rate of 1%.

CR is the average contact rate, in this model CR is equal to 30 contact per year.

The stock of potential installed base (PI) presents number people who are aware of smartphone technology but did not decide which OS they want to continue with.

$$PI = \int (AR + CR - SR)dt + PI(0) \quad (7)$$

In this equation, AR is the adaption rate, CR is the change rate and SR is the sales rate which we will explain the equation in the next section. The initial value for this stock is 200 million people.

3.1.1.2. Market section part 2:

In figure 16 shows the other main loop which includes smartphone market and installed base.

Current installed base (CI), is the arrayed stock and it shows the current number of people using each OS.

$$CI = \int (SR - ChR)dt + CI(0) \quad (8)$$

This stock accumulates the difference between Sales rate, SR, and adaption change rate, ChR. The initial value of this stock for all OSEs is zero.

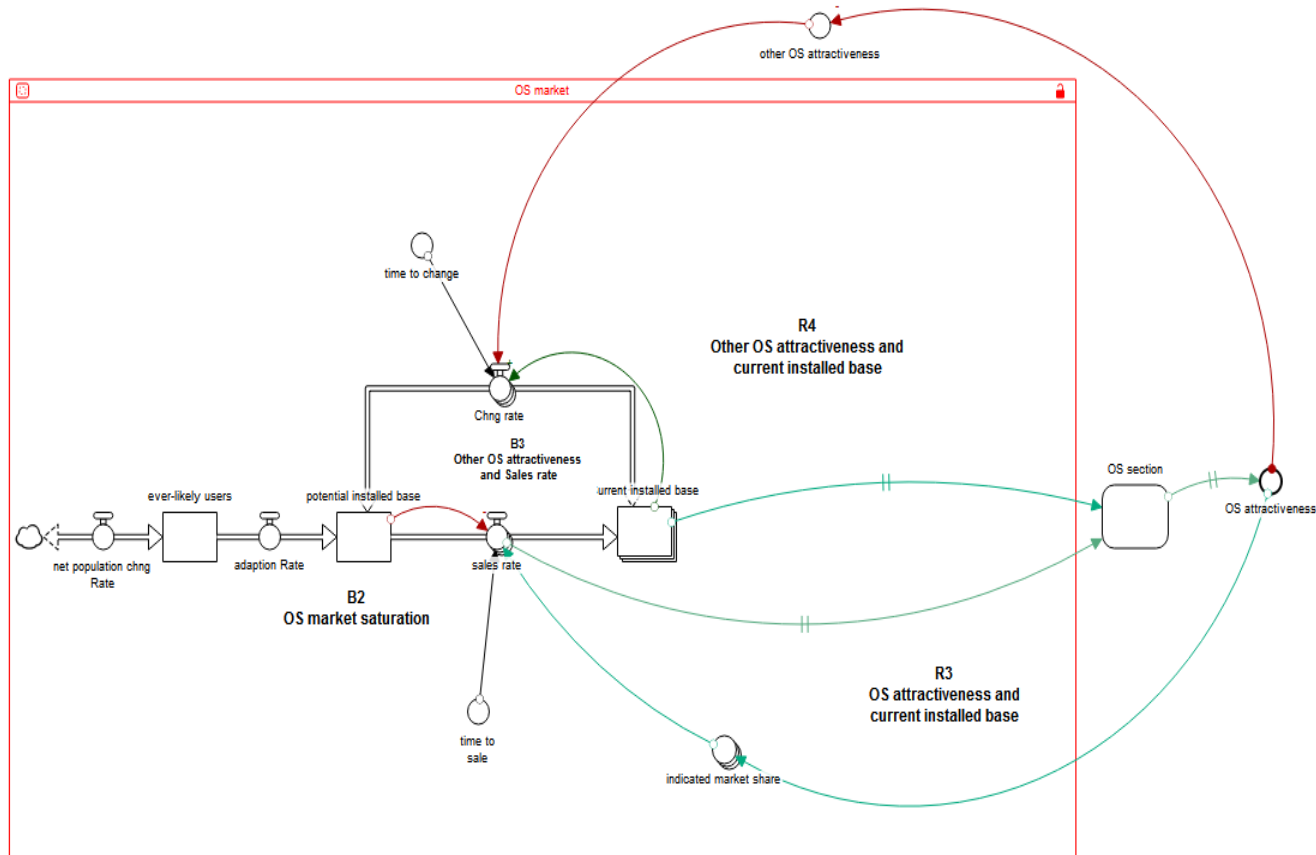


Figure 16 SFD of market section (part 2)

Sales rate (SR), depends on the most important variable in this model, “indicated market share”. This variable shows how potential users are going to make decision on choosing one operating system.

$$SR = \frac{PI * IMS}{TS} \quad (9)$$

Where PI is Potential installed base, IMS is indicated market share and TS is the time to sale which equals to two year.

Indicated market share (IMS), is the most important variable in this model, it shows how people in the category of potential installed base are going to make decision and choose their favorable OS.

$$IMS = SAFEDIV (OSA, SUM(OSA), 0) \quad (10)$$

Where OSA is the OS attractiveness, this value is another crucial value and the main output of operating system module, also in this equation we use safe divide equation because SUM of OS attractiveness at the beginning is zero.

Market Share (MS), concerns about the whole distribution of market not just in an exact time but in the whole period of time.

$$MS = SAFEDIV (CI, SUM(CI), 0) \quad (11)$$

Another loop includes the change rate of smartphone or operating system. Normally smartphone users change their phone every two years, due to either two-year contracts or two-year installment plans for their devices [22]. But the “other OS attractiveness” influence this time, if the OS attractiveness is much lower than others that most definitely will increase the smartphone change rate.

Change rate (CR):

$$CR = \left(\frac{CR}{TCh} \right) * (OOSA) \quad (12)$$

Where TCh is time to change which is constant and two years.

Other OS attractiveness (OOSA), this variable shows how other operating systems are more desirable than the OS.

$$OOSA = 1 - OSA \quad (13)$$

Where OSA is the OS attractiveness, we will describe this variable in the equation number 19.

Market section has three variables that are being used in the Operating System section, these variables are “OS relative sales rate”, “relative total sales rate” and “total awareness”.

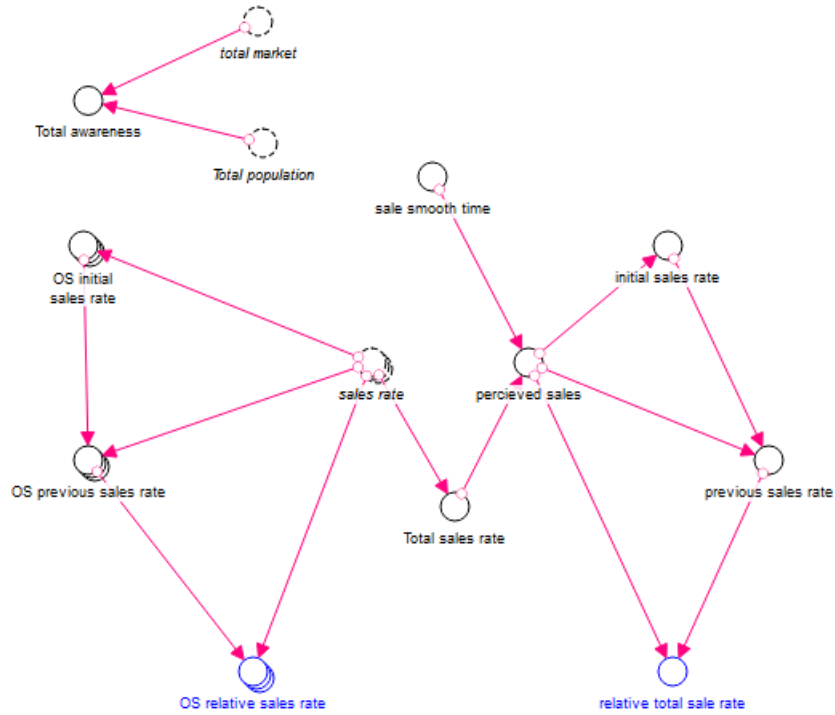


Figure 17 SFD of market section outputs for OS section

OS relative sales rate (OSRSR), shows the increase or decrease rate of sales rate, one year to another year for each OS separately.

$$OSRSR = SAFEDIV (SR, OSPSR, 0) \quad (14)$$

Where OSPSR, is the OS previous sales rate.

Relative total sales rate (RTSR), shows the increase or decrease of total sales rate.

$$RTSR = SAFEDIV (PrSR, PPrSR, 0) \quad (15)$$

Where PrSR is the perceived sales rate and PPrSR is the previous perceived sales rate.

Perceived sales rate (PrSR):

$$PrSR = S_mthn1 (TSR, SST) \quad (16)$$

$$TSR = Sum (SR) \quad (17)$$

In this equation TSR is the total sales rate which is the sum of all sales rate, and SSR is the sales smooth time which is constant and two years.

Total awareness (TAW), this variable shows what fraction of total population are aware of the smartphone technology.

$$TAW = TM/P \quad (18)$$

3.1.2. Operating system module:

This section includes 4 different characteristics of the standard, we are going to present each category separately, plus the mutual part which is the OS attractiveness.

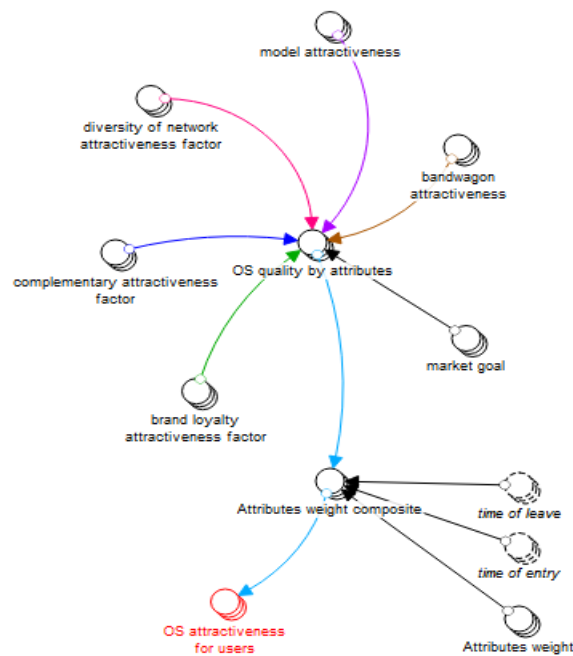


Figure 18 SFD of OS attractiveness for users

First we are going to start with the most important variable in this section which is:

OS attractiveness for users (OSA):

$$OSA = \frac{SUM(AWC[* , OS])}{SIZE(AWC[* , OS])} \quad (19)$$

Until now we used arrays for the different operating systems, in this part of the model we have another array named attributes. This array shows different attributes of the OS, such as network diversity attractiveness and bandwagon attractiveness. In OSA we want to summarize all those variables to one variable. AWC is attributes weight composite.

Attribute weight composite (AWC):

$$AWC = IF T \geq TL \text{ then } 0 \text{ else } IF T \geq TE \text{ then } OSQA * AW \text{ else } 0 \quad (20)$$

Where TL is time of leave, TE is time of entry, OSQA is the OS quality by attributes and AW is attributes weights. We will explain TE and TL in the OS standard support strategy section.

Attributes weights (AWe): For allocating weights to the factors we used Geerten van de kaa's research [5]. Table 4 is the original table from the paper.

In one hand not all of the factors directly influence the OS attractiveness and on the other hand some of these factors are out of scope of this project, we use the following process to allocate the appropriate weights in table 5:

- Considering the structure of the model, the final weight of network diversity is the sum of network diversity, effectiveness of development process and appropriability.
- Brand reputation is one of the factors from the characteristics of standard supporters so the weight of one other factor, learning orientation is being add to brand reputation factor.

Table 4

factors of standard dominance ranked by importance

Factors	Ave. Weight
Diversity of the network	0.1897
brand reputation	0.1825
financial strength	0.1159
current installed base	0.0904
timing of entry	0.0874
compatibility	0.0685
complementary	0.0542
marketing communications	0.0442
technological superiority	0.0437
effectiveness of development process	0.0424
appropriability strategy	0.0422
learning orientation	0.0398

- Complementary goods is the only factor being consider from the characteristics of the standard so the weight of two other factors, compatibility and technological superiority are being added to Complementary goods factor.

- The final weight of current installed base is half of the original weight because we add the model variety as an extra factor so we are going to allocate the rest of the weight to that.

- Model variety is the only factor we added to this model because of the nature of the market. The weight of this important factor is the sum of financial strength, Timing of entry, marketing and half of current installed base.

Table 5

Weights of final factors in the model

Factors	Ave. Weight
Diversity of network	0.2741
brand reputation	0.2222
current installed based	0.04515
complementary goods	0.1662
Model variety	0.29235

OS Quality attributes (OSQA): This variable is the collecting variables, it is collecting different factors like, complementary goods, bandwagon, and diversity of network, brand loyalty and model variety for each OS separately and multiply them with market goal of each OS. Figure 19 shows this table in the model:

Equation editor: OS quality by attributes

Select cells to enter equation for...

[attributes[*] OS[*]]

	Android	iOS	Windows	BlackBerry	Symbian&others
Complementary	complementary_att...	complementary_att...	complementary_att...	complementary_att...	complementary_att...
Network	diversity_of_netwo...	diversity_of_netwo...	diversity_of_netwo...	diversity_of_netwo...	diversity_of_netwo...
bandwagon	bandwagon_attract...	bandwagon_attract...	bandwagon_attract...	bandwagon_attract...	bandwagon_attract...
brand	brand_loyalty_attra...	brand_loyalty_attra...	brand_loyalty_attra...	brand_loyalty_attra...	brand_loyalty_attra...
Model	model_attractivene...	model_attractivene...	model_attractivene...	model_attractivene...	model_attractivene...

Figure 19 OS quality attributes table

Market Goal (MG): Each OS has a separate distribution strategy, meaning for example in some countries there is no Apple Store available, and Also BlackBerry phones were not distributed or fully functional worldwide. Table 6 shows the assumption values for market goal of each OS.

Table 6

Market Goal for each OS

Market Goal	
Symbian	1
Windows Phone	0.8
BlackBerry	0.7
iOS	0.7
Android	1

In the next 4 sections we are going to explain the formulation and stock and flow diagram of each category of characteristics.

3.1.2.1. Standard support strategy:

In this section we explain how decisions made by the OS owners are going to influence the OS attractiveness for supporters. There are some strategies such as Appropriability of the OS which are being made and in this model they being consider as exogenous factors. Most of variables in this section are exogenous, the loop that being shown at the top of figure 20 will be explain as loop R6 in the next section.

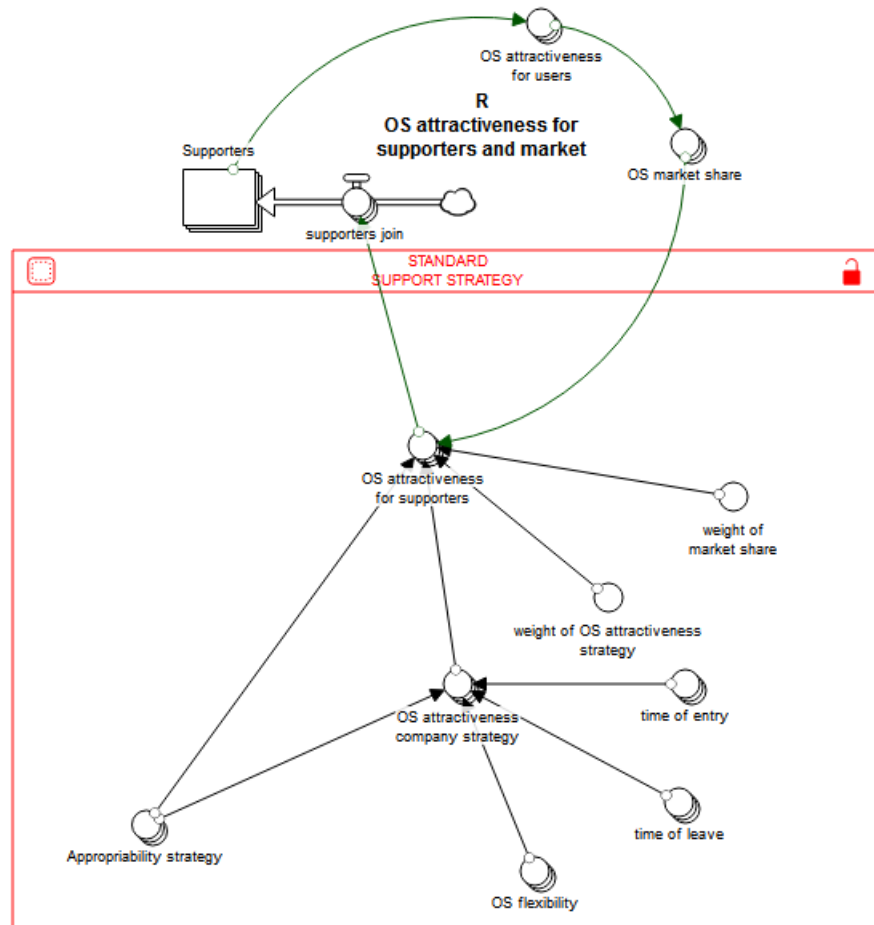


Figure 20 SFD of standard support strategy

OS attractiveness for supporters (OSAS), there are different variables that affect the manufacturers or supporters decisions to either adopt the specific OS or not. One variable is the OS owners’ strategy and the other is market share of that OS.

$$OSAS = IF T \geq TL \text{ then } 0 \text{ else } IF T > TE \text{ then } \left(\frac{MS}{100}\right) * WMS + OSACS * WOSACS * AP \text{ else } 0 \quad (21)$$

Where T is Time, TL is the time of leave, TE is the Time of entry, MS is market share, WMS is weight of market share, OSACS is the OS attractiveness company strategy and WOSACS is the weight of OS attractiveness company strategy.

Weight of market share (WMS) this is an assumption in this model and the weight is equal 30%.

Weight of OS attractiveness company strategy (WOSACS): this is an assumption in this model and the weight is equal 705%.

OS attractiveness company strategy (OSACS), this variable shows the importance of the OS owners' strategy to attract device manufacturers.

$$OSACS = IF T > TE \text{ then } IF T < TL \text{ then } OSF * APS \text{ else } 0 \text{ else } 0 \quad (22)$$

Time of entry (TE): The time horizon of the model is 2000-2016, and each OS entered the market in different time so we need the variable to show the time each of them entered the market. Table 7 shows the time when for each OS entered the market.

Table 7

Market entry time

	Year
Symbian	2000
WindowsPhone	2002
BlackBerry	2003
iOS	2007
Android	2008.5

Appropriability strategy (APS): According to Van de Kaa [5], appropriability strategy refers to an actor's ability to capture profits generated by a standard. An open licensing policy encourages imitation by competitors which will, in general, increase the chance of standard becoming dominant [23]. Table 8 shows the positive effects of appropriability strategy, the scale is 0-1 which 1 means supporters can adapt the OS and it is free and 0 means them only manufacturer that can use the OS is the main owner. The reason Windows phone has two value is for before they bought Lumia and after that.

Table 8

Appropriability strategy

Appropriability	
Symbian	0.25
WindowsPhone	0.5&0.1
BlackBerry	0
iOS	0
Android	1

OS flexibility (OSF): It is important to know that device manufacturers cannot choose iOS and blackberry OS as their operating system. So the comparison for this variable is between Symbian, Android and Windows phone. The Value is between 0-1, 1 means the most flexible and 0 means not available.

Table 9

Operating system flexibility

OS flexibility	
Symbian	0.25
Windows Phone	0.25
BlackBerry	0
iOS	0
Android	1

Android is an open source software, which means that the code is freely available under Apache license for modification and distribution by device manufacturers, wireless carriers and enthusiast developers. Windows Phone is a closed-sourced, which means that it is solely developed by the company and protected by copyright. The system offers new features such as: a whole new interface, text input by an on-screen virtual keyboard, threads messaging,

Internet Explorer browser, organized contacts via People Hub, Email access, Windows Phone App and Windows Phone Store [24].

Time of leave (TL): Two of the OS owners claimed their lost and left the market. This variable shows that time.

Table 10

Time of leave

Time of leave	
Symbian	2013
WindowsPhone	-
BlackBerry	2016
iOS	-
Android	-

3.1.2.2. Other stakeholders:

For the purpose of clearness, this category of characteristics divided to two parts; first network diversity and smartphone model, second bandwagon effects.

3.1.2.2.1. Network diversity and smartphone models:

In this part we explain the relation between OS attractiveness for supporter from the previous category, supporters and smartphone models. It is worth mentioning that supporters start using one operating system when its attractiveness increases, also the number of potential supporters increases as the market expands. Same mechanism happens for model production as well.

This part of the model has two main variables (Figure 21); diversity of network attractiveness factor and model diversity attractiveness which we are going to show separately in the following SFDs. The delayed arrows show indirect relations and they are not going to be explained.

Diversity of network attractiveness (DNA), this variable shows what fraction of the available supporters in the market are supporting the OS. For operating systems such as iOS and BlackBerry this factor is low because they do not let other supports to use their OS.

$$DNA = \frac{S}{SOS} \quad (23)$$

Where S is the number of supporters for each OS and SOS is the Sum of supporters for all operating systems.

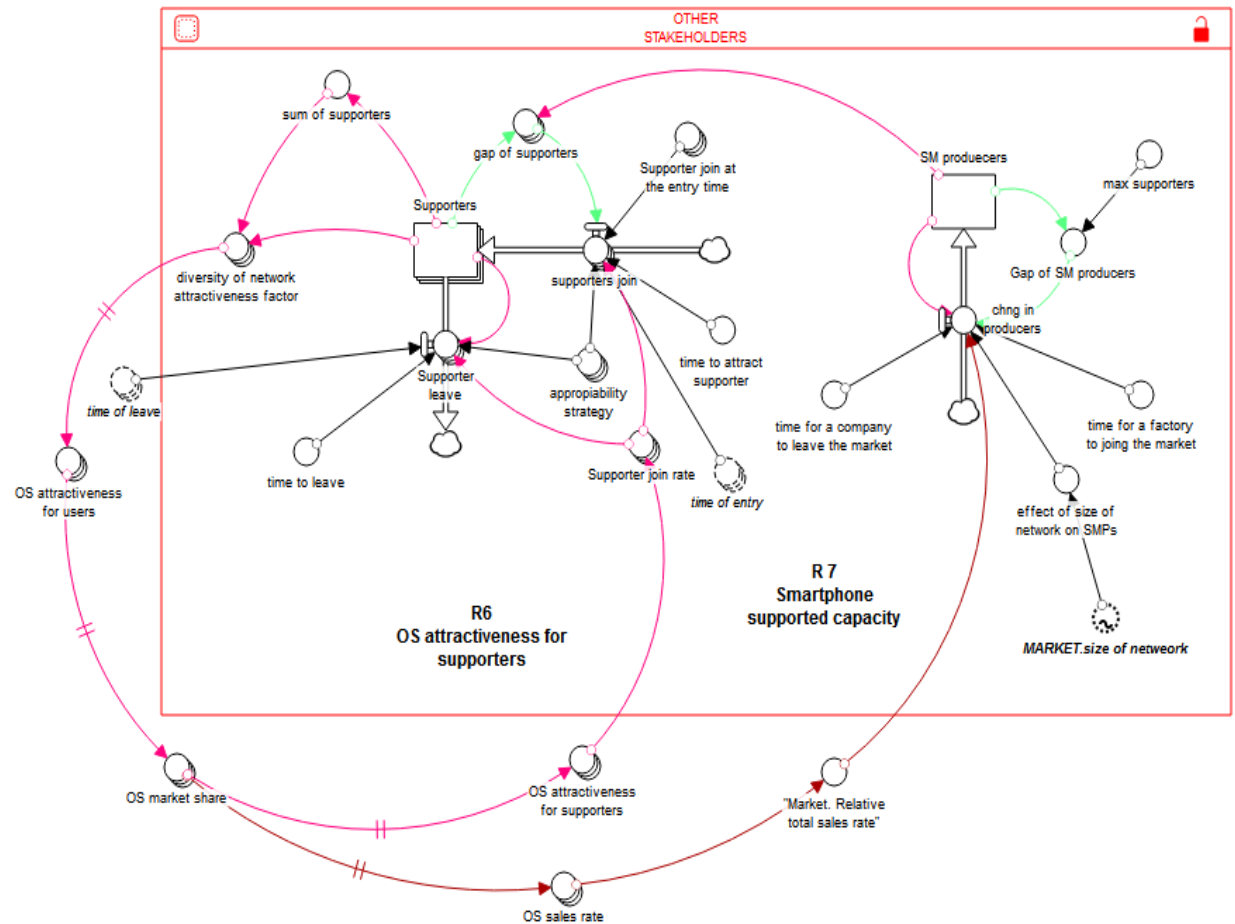


Figure 21 SFD of Other stakeholders (part 1)

Supporters (S) are a stock variable and shows the number of manufacturers that produce goods with that OS. Initial value of this stock for all OS is zero.

$$S = \int (SJ - SL)dt + CI(0) \quad (24)$$

Accumulates the difference between Supporters join, SJ, and Supporter leave, SL.

Sum of Supporters (SOS):

$$SOS = Sum(S) \quad (25)$$

Supporters join (SJ)'s equation is very important and it is different for each OS, for BlackBerry and iOS at the time of entry one supporter join which is the OS owner as well. For other OSes according to Supporter Join rate (SJR) and the gap of supporters (GS), supporters start to join and produce products.

$$SJ = \begin{cases} 0 & \text{IF } T < TE \\ SJET & \text{IF } T = TE \\ \frac{GS * SJR}{TTAS} & \text{IF } APS > 0 \end{cases} \quad (26)$$

Supporters Initial join rate (SJET), is being used in the model in favor of the iOS and BlackBerry, to show when they join the market just one supporters join with them and that's the owner. Since DT in this model is 4 then when the initial rate equals to 4 that mean one supporter joined.

Table 11

Supporter initial join rate

Supporter initial join rate	
Symbian	0
WindowsPhone	0
BlackBerry	4
iOS	4
Android	0

Supporter leave (SL), supporters may stop supporting the specific OS when the attractiveness of other OSes for companies is higher. Also it is important to know that when

the OS owner discontinue the OS that means all the supporters stop supporting the OS as well.

$$SL = \text{IF } T \geq TL \text{ then } \frac{S}{TTL} \text{ else if } APS > 0 \text{ then } (1 - SJR) * \frac{S}{TTL} \text{ else } 0 \quad (27)$$

If the appropriability is zero then the owner and supporter are same company and the SL is zero. Otherwise supporters leave when the other OSes attractiveness is higher.

Supporter join rate (SJR), this variable shows how supporters are making decisions on joining and supporting specific OSes.

$$SJR = \text{SAFEDIV}(OSAS, \text{SUM}(OSAS), 0) \quad (28)$$

Where OSAS is the OS attractiveness for supporters.

Gap of supporters (GS), this gap shows the difference between available producers in the market and the OS supporters.

$$GS = \text{IF } S < SMS \text{ then } (SMS - S) \text{ else } 0 \quad (29)$$

Where SMS is the smartphone supporters in the whole market.

SM producers (SMP), As we explained before in the beginning of the simulation time there were limited number of producers in the world and as the market grow so does the number of producers. The initial value of SM producers is five factories.

$$SMP = \int (ChgP) dt + CI(0) \quad (30)$$

Accumulates the change in producers (ChgP).

Change in producers (ChgP):

$$ChgP = \frac{(GSMP * RTSR * ENSOSMS)}{TFJ} - \frac{SMP}{TFL} \quad (31)$$

Where GSMP is the gap of Smartphone producers, RTSR is the relative total sales rate, ENSOSMS is the effect of network size on smartphone supporters. TFL and TFJ are time to leave and time to join variables and both equal to 5 years.

Gap of smartphone producers (GSMP):

$$GSMP = IF MAXP > SMP \text{ then } (MAXP - SMP) \text{ else } 0 \quad (32)$$

Maximum Producers (MAXP): in this model this variable shows a big smartphone manufacturer capacity, this is an assumption and is constant 20 factories.

As we discussed before, the other important variable in this section is Model attractiveness, figure 22 show the SFD for this part.

Model attractiveness (MA), this variable represent the fraction of available smartphone model in the market for each OS.

$$MA = SPM/SSPM \quad (33)$$

In this equation SPM is the smartphone model and SSPM is the sum of smartphone models.

The stock of smartphone Models (SPM), shows the number of available smartphone models in the market.

$$SPM = \int (PR - OR)dt + SPM(0) \quad (34)$$

This stock accumulates the difference between Production rate, PR, and Obsolesce rate, OR. Initial values for all operating systems except Symbian are zero and for Symbian the value is 10 widgets.

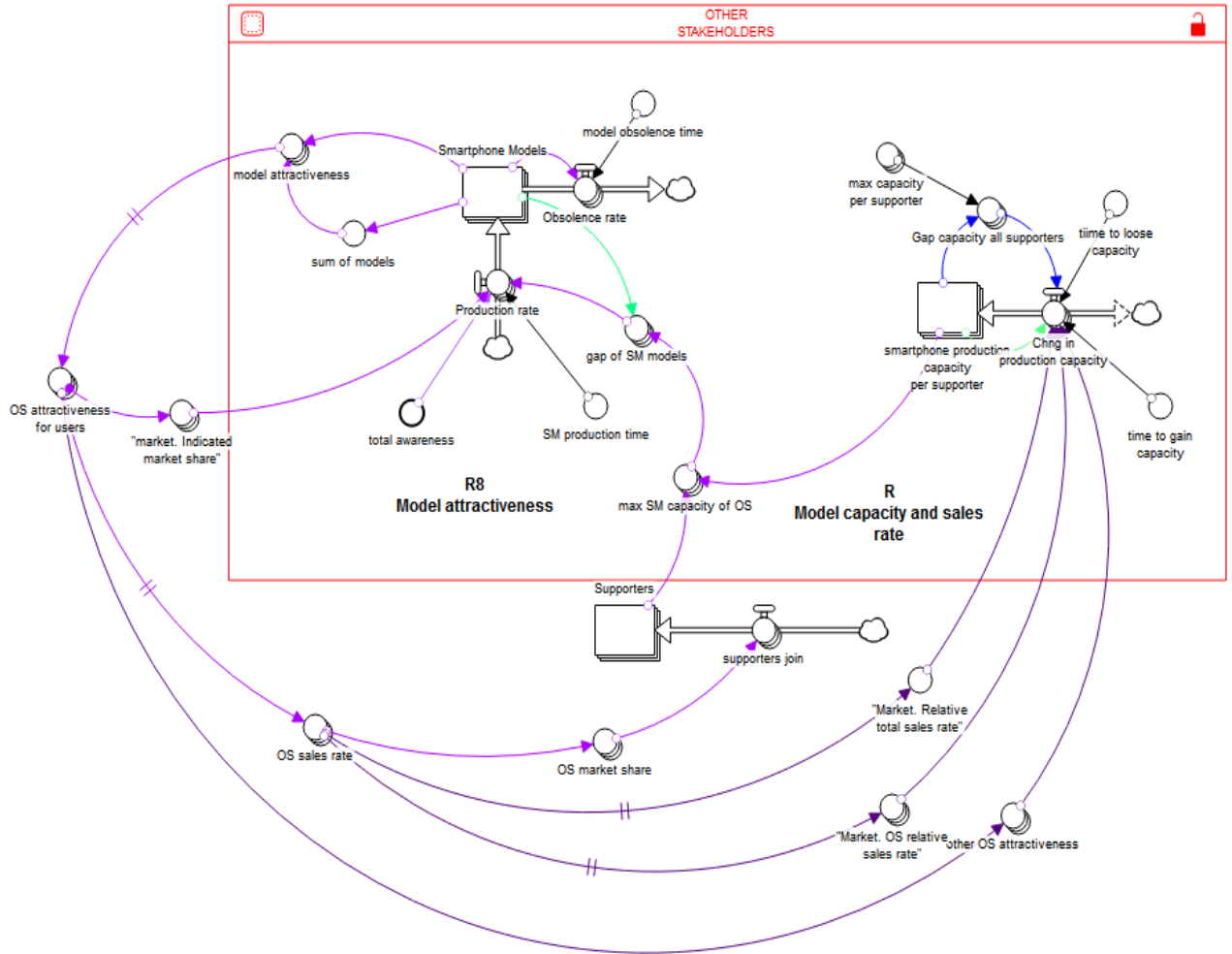


Figure 22 SFD of Other stakeholders (part 2)

Production rate (PR), this variables shows how the decisions over making new model of smartphones for specific OS is being made. This decision depends on different factors such as the gap of smartphone model (GSPM), the total awareness (TAW), and indicated market share IMS and the smartphone model production time (SMPT) which in this mode is equal to 1 year.

$$PR = \frac{GSPM * TAW * IMS}{SMPT} \quad (35)$$

Gap of smartphone model (GSPM), shows the difference between the whole manufacturers' capacity in producing new models and the models that are already being produce.

$$GSPM = IF MAXSPM > SPM \text{ then } (MAXSPM - SPM) \text{ else } 0 \quad (36)$$

In this equation MAXSPM is Maximum smartphone model capacity of OS.

Maximum smartphone model capacity of OS (MAXSPM), by multiplying the number of supporters and the smartphone production capacity per supporter (SPPC) this variable is being calculated.

$$MAXSPM = S * SPPC \quad (37)$$

Smartphone production capacity (SPPC), this stock represent the capacity that the manufacturer can produce different smartphone models.

$$SPPC = (ChgPC)dt + SPPC(0) \quad (38)$$

Change production capacity (ChgPC), This rate is different for Oses with different strategies, So a companies who are OS owners increase their capacity by observing their relative sales rate (OSRSR), while the others increase their capacity by observing the relative total sales (RTSR) of the market. The initial value for all operating systems is 3 widgets per factories.

- For Symbian, Windows phone and Android

$$ChgPC = \frac{GCa * RTSR}{TTIC} - \frac{(SPPC * OOSAT)}{TTLC} \quad (39)$$

Where GCa is the gap of capacity, RTSR is the relative total sales rate, OOSAT is the other OS attractiveness factor and TTIC is the time to increase capacity (3 years) and TTLC is the time to lose the capacity (2 years)

- For iOS and BlackBerry

$$ChgPC = \frac{GCa * OSRSR}{TTIC} - \frac{(SPPC * OOSAT)}{TTLC} \quad (40)$$

The only different is the companies that are responsible for their own smartphone production, are going to increase their capacity according to their OS sales rate not the total sales rate in this equation OSRSR is the OS relative sales rate.

Gap of capacity (GCa):

$$GCa = IF MAXCPS > SPPC \text{ then } (MAXCPS - SPPC) \text{ else } 0 \quad (41)$$

In this equation MAXCPS is the maximum capacity per supporter which are constant for each OS.

Maximum capacity per supporter (MAXCPS):

Different companies has different strategies, so a company like Apple does not want to produce so many models at the same time, while companies such as Samsung and LG want to produce a high range of models in order to dominant the market.

Table 12

Maximum capacity per supporter

Max Capacity per supporter	
Symbian	50
WindowsPhone	50
BlackBerry	50
iOS	20
Android	150

Size of network (SON): This variable shows the how many OSes are available in the market during times of simulations.

Table 13

Network Size

Network Size			
2000	1	2008	5
2001	1	2009	5
2002	2	2010	5
2003	3	2011	5
2004	3	2012	5
2005	3	2013	5
2006	3	2014	4
2007	4	2015	4
		2016	4

3.1.2.2.2. Bandwagon:

Figure 23 shows the relation between OS current installed base and bandwagon attractiveness. The indirect relations are being presented out of the sector in order to show the dynamics.

Bandwagon attractiveness (BA), is the fraction of bandwagon adaption for each OS to the all.

$$BA = SAFEDIV (BAd, SBAd, 0) \quad (42)$$

Where BAd is the bandwagon adaption, SBAd is the sum of bandwagon adaption.

Bandwagon adaption (BAd), shows that the increase in current installed base positively increase the contact between potential installed base and current installed base and that will lead to higher adaption rate.

$$BAd = CBIP * AF \quad (43)$$

Application (Ap), the stock of application which depends on the difference between Development rate, DvR, and Application Obsolesce rate, ApOR. The initial value of this stock for all Oses is zero.

$$AP = \int (DvR - ApOR)dt + AP(0) \quad (48)$$

Development Rate (DvR):

$$DvR = IF T > TE \text{ then } Ap * \frac{OSADv}{Sum(OSADv)} * \frac{1}{ApDvT} \text{ else IF } T \\ = TE \text{ then } X \text{ else } 0 \quad (49)$$

In this equation OSADv is the OS attractiveness for developer, ApDvT is the application development time which is constant and equal to 0.25 year. The important factor is X, which is different for each OS because this shows the number of application in the time of the OS publishing. The table 14 shows this value for each OS.

Table 14

Application Start rate (X)

Application start rate	
Symbian	4K
WindowsPhone	50K
BlackBerry	50K
iOS	120K
Android	120K

OS attractiveness for developers (OSADv), there are different factors that affect the developers' decision on choosing an OS for their applications. These factors are the OS flexibility for developers (OSFDv), the OS registration fee (OSRF), OS market share (OSMS) and the effect of app store on attractiveness for developers (ETOEa).

$OSADv = IF T$

$$\begin{aligned}
 &< TL \text{ then } \left(\frac{OSFDv}{5}\right) * 0.7 + \left(\frac{OSRF}{5}\right) * 0.2 + \left(\frac{OSMS}{100} * 0.1\right) \\
 &* ETOEA \text{ else } 0 \quad (50)
 \end{aligned}$$

OS flexibility for developers (OSFDv): This variable uses studies to compare and give weight to different characteristics of OS flexibility for developers, these are items such as development, Design guidelines, Fragmentation, publishing and restrictions and profit [25]. According to this study iOS and Android has the same flexibility for developer, but still developers prefer Android because the process of publishing the application is much faster in Google play. Other OSes have very low flexibility in this matter.

Table 15

OS flexibility for developers

Flexibility for developers	
Symbian	1
WindowsPhone	2
BlackBerry	2
iOS	4.5
Android	5

OS app store registration fee (OSRF): In this table we show the OS attractiveness according to the fee that members need to pay. iOS have the most expensive registration fee which is 99\$ per year, while the fee for others are different but the difference is not so much to effect the attractiveness.

Table 16

OS registration fee attractiveness

OS registration fee attractiveness

OS registration fee attractiveness	
Symbian	3
Windows Phone	5
BlackBerry	5
iOS	3
Android	5

Effect of TOE on attractiveness for developers (ETOEA):

$$ETOEA = IF T \geq TApS \text{ then } 2 \text{ else } 1 \quad (51)$$

As the company launch the app store for its OS the attractiveness of that OS for developers increase.

Time of app store (TApS): This is an exogenous variable that show what year each of OS owners launched the App store.

Table 17

App store launched year

App store Time

App store Time	
Symbian	2009
Windows Phone	2012
BlackBerry	2009
iOS	2008
Android	2008

Application Obsolesce rate (ApOR), applications usually have a life cycles and after a while they are being obsolete and useless or even being removed from App stores.

$$ApOR = Ap/AOT \quad (52)$$

In this equation A_p is the number of applications and AOT is the application obsolescence time which is the average of 10 years.

3.1.2.3.2. Complementary goods:

Complementary goods are those other goods needed to successfully commercialize a certain standard [5]. One important fact is that there are some manufacturers that produce complementary goods such as Smart watches or fitness band which are compatible with all the OSes (current ones such as Android, iOS and Windows Phone), those goods are not being included in this model structure. So complementary goods are ones which are being produced by the OS supporters specifically (Figure 25).

Complementary attractiveness factor (CAF), is the average of application attractiveness factors and complementary goods attractiveness factor.

$$CAF = (APAF + CGAF)/2 \quad (53)$$

Where $APAF$ is application attractiveness factor and $CGAF$ is Complementary goods attractiveness factor.

Complementary good attractiveness factor ($CGAF$), shows the fraction of complementary goods for each OS.

$$CGAF = SAFEDIV (CG, SCG, 0) \quad (54)$$

In this equation CG is the stock of complementary goods and SCG is sum of all complementary goods for all OSes.

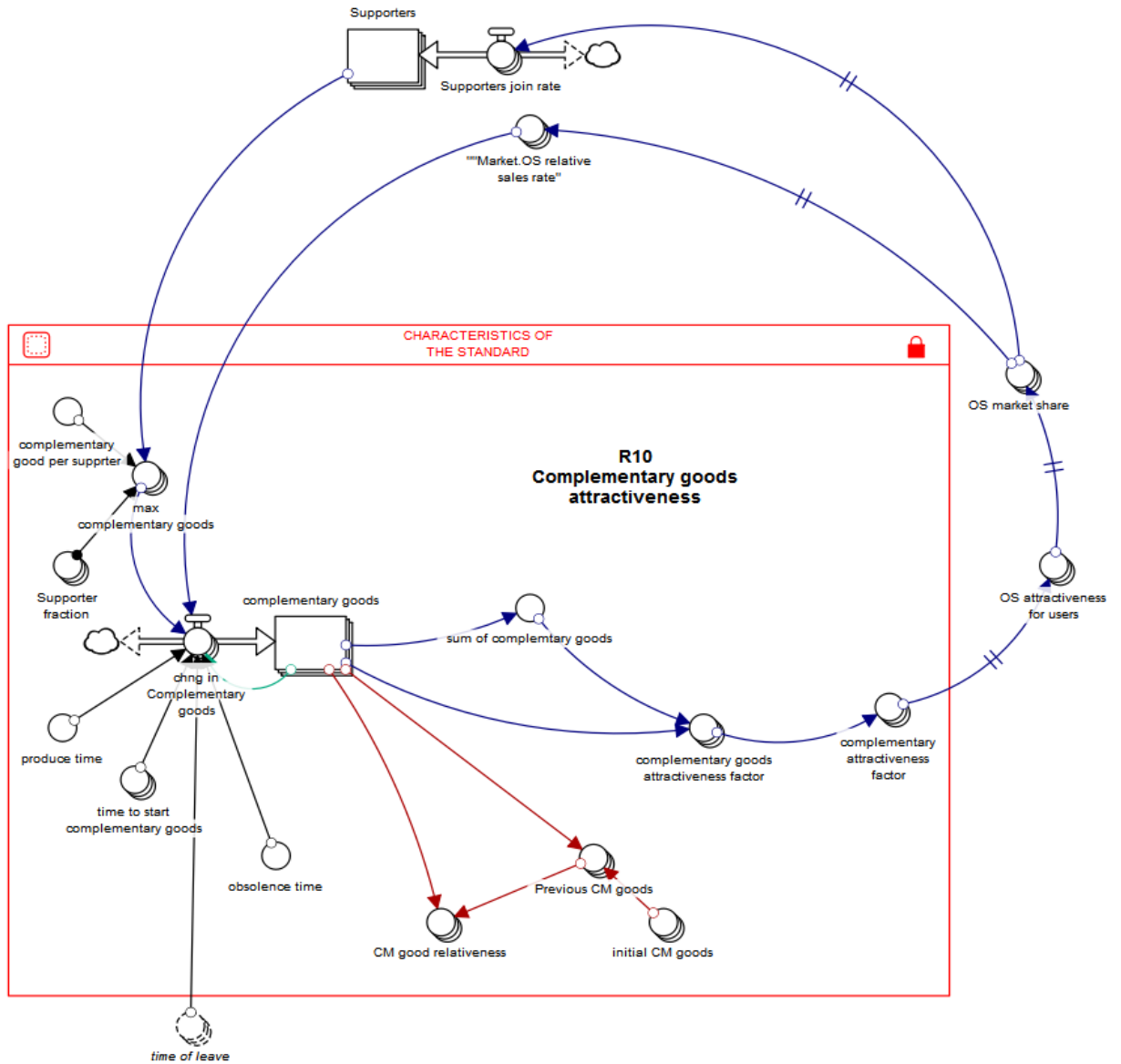


Figure 25 SFD of characteristics of the standard (Part 2)

Complementary Goods (CG), is the stock of complementary goods for each OS. The initial value of this stock is zero for each OS.

$$CG = ChgCG + CG(0) \quad (55)$$

Change in complementary goods rate (ChgCG):

$ChgCG = IF T < TEGG \text{ then } 0 \text{ else } T$

$$< TL \text{ then } \left((MaxCG - CG) * \frac{OSRSR}{PT} \right) - \left(\frac{CG}{OTCG} \right) \quad (56)$$

Where TEGG is the time supporters start to produce complementary goods, MaxCG is the maximum complementary goods, OSRSR is the OS relative sales rate and PT is production time (2 years), and OTCG is the complementary obsolesce time.

Time to start complementary goods (TEGG), this an exogenous variables and we use the available data for that (table 18).

Table 18

Complementary good start time

Complementary goods Time

Symbian	-
Windows Phone	2010
BlackBerry	2008
iOS	2008
Android	2009

Maximum complementary goods (MaxCG), shows how many complementary can be produce for each OS. The equation is the multiply of OS supporters and the Complementary goods per supporter (CGPS).

$$MaxCG = S * CGPS * SFra \quad (57)$$

Complementary goods per supporter (CGPS): is the constant number of 5 widgets per factories.

Also since the number of available complementary goods has a positive effects on brand royalty through increase of switching cost. In this section we have one other important variable that will be used in the next category.

Supporter fraction (SFra), shows what percentage of the supporters are actually producing complementary goods, this values are assumption, for example for iOS and BlackBerry since there is only one supporters then that supporters is producing complementary goods as well. But for other OSes it is not 100% true, for example Samsung produce smartphones with Windows Phone as an OS but the company does not produce complementary goods for Windows Phone. Table 19 shows this values:

Table 19

Supporter fraction

Supporter fraction	
Symbian	0.5
WindowsPhone	0.5
BlackBerry	1
iOS	1
Android	0.5

Complementary goods relativeness (CGR):

$$CGR = SAFEDIV (CG, PreCG, 0) \quad (58)$$

Where PreCG is the previous number of available complementary goods.

Previous Complementary good (PreCG):

$$PreCG = PREVIOUS (CG, InitialCG) \quad (59)$$

3.1.2.4. Characteristics of standard supporters:

Managing brand equity involves decision making in various dimensions in order to create or sustain an active presence of a brand in a given market [26]. In this part of model we used the model that Peter A.Otto and J.Robert Bois [26] suggested, In that model they considered four components of the brand equity which are; brand awareness, brand loyalty, perceived quality and brand association. But for the purpose of this thesis we are using two parts, brand

awareness and brand loyalty. Two adjustments has been made by us, one the effect of complementary goods and OS ecosystem on the brand loyalty and the effect of fraction of awareness on the brand awareness (Figure 26).

The most important variable in this section is,

Brand loyalty attractiveness factor (BLAF), show the level brand loyalty for each OS in comparison with others.

$$BLAF = SAFEDIV (BL, SBL, 0) \quad (60)$$

Where BL is the brand loyalty and SBL is the Sum of brand loyalty for all OSes.

Brand Loyalty (BL), is stock with maximum level of 100. This stock shows how each of OSes' users are loyal to the brand. The level of brand loyalty increase by the complementary goods and brand awareness. This stock has zero loyalty as initial value for all operating systems.

$$BL = \int (ILR - LLR)dt + BL(0) \quad (61)$$

Accumulates the difference between Increase in loyalty rate, ILR, and Loose of loyalty rate, LLR.

Loose of loyalty rate (LLR):

$$LLR = IF T > TE then \frac{BL * OOSA}{TLBL} \quad (62)$$

Where OOSA is the other OS attractiveness and TLBL is the time to loose brand loyalty which is constant and equal to 2 years.

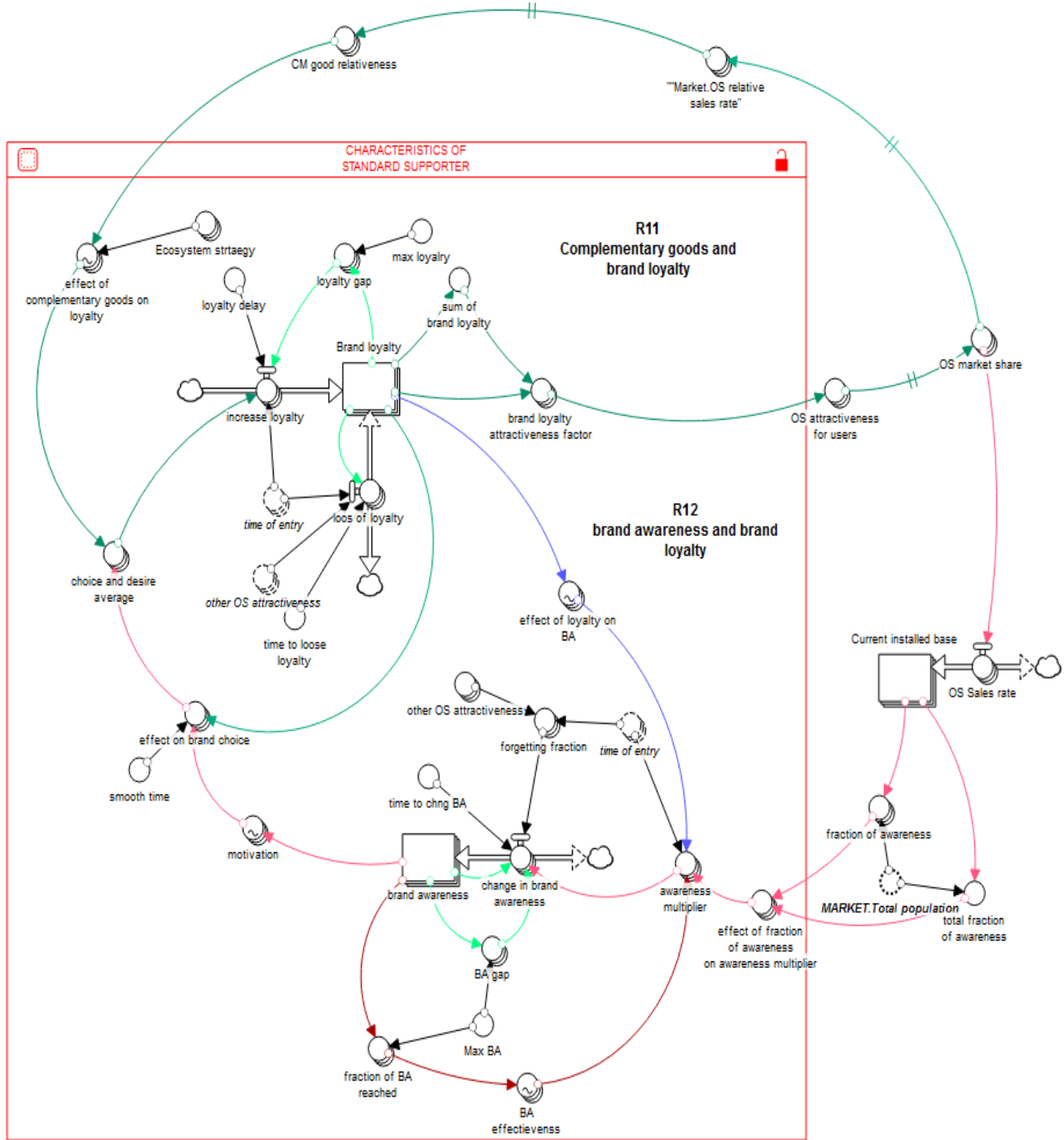


Figure 26 SFD of Characteristics of standard supporters

Increase in loyalty rate (ILR):

$$ILR = IF T > TE \text{ then } \frac{CDA * LG}{LD} \quad (63)$$

Where CDA is the choice and desire average, LG is the loyalty gap and LD is the loyalty delay (constant of 1 year).

Loyalty Gap (LG), this gap shows the difference between maximum brand loyalty and brand loyalty for each OS.

$$LG = MaxL - L \quad (64)$$

Maximum loyalty is constant for all OSes and equals to 100.

Choice and desire average (CDA):

$$CDA = \frac{(ECGL * 0.9) + (EBC * 0.1)}{2} \quad (65)$$

In this equation ECGL is effect of complementary goods on brand loyalty and EBC is the effect of brand choice. This variable have different weights because the wide range of complementary goods increase brand loyalty by increasing the switching cost which is so important in this market.

Effect of complementary goods on brand awareness (ECGL):

$$ECGL = Table \text{ function } (ESS * CGR) \quad (66)$$

This variable is a table function which shows the relation between availability of complementary good, OS ecosystem and brand loyalty. Where ESS is the ecosystem strategy of the OS owner and CGR is the complementary goods relativeness (figure 27).

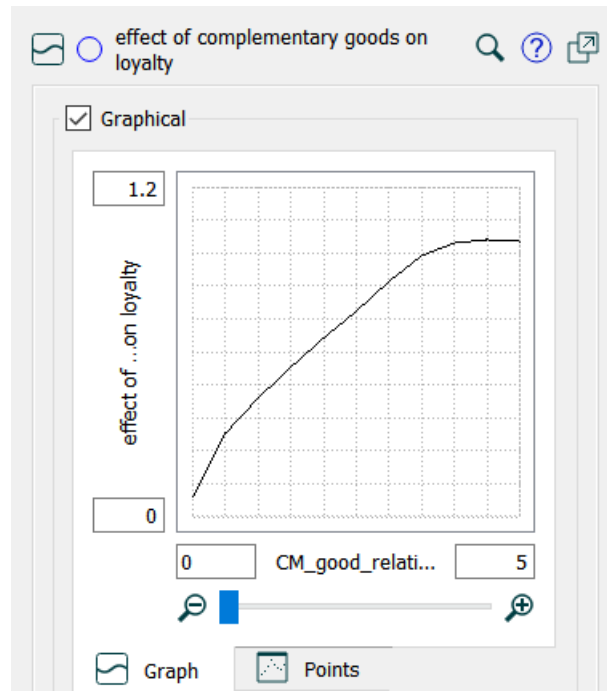


Figure 27 effect of complementary good on the brand loyalty

Ecosystem strategy (ESS):

When producing complementary good some of the manufacturers are willing to produce more compatible products while other prefer to use complementary goods to lock-in its users. These values are assumptions according to current situations in the market tale 20 shows these values.

Table 20

Ecosystem strategy

Ecosystem strategy	
Symbian	0
Windows Phone	1
BlackBerry	0
iOS	5
Android	3

Effect on brand choice (EBC):

$$EBC = SMTHN1((BL * M), ST)/100 \quad (67)$$

Where M is motivation and ST is the smooth time which is 0.5 year.

Motivation (M):

This equation is a table function which is made according previous study in brand management facilitation [26]. Using brand awareness as an input this table function is (Figure 28):

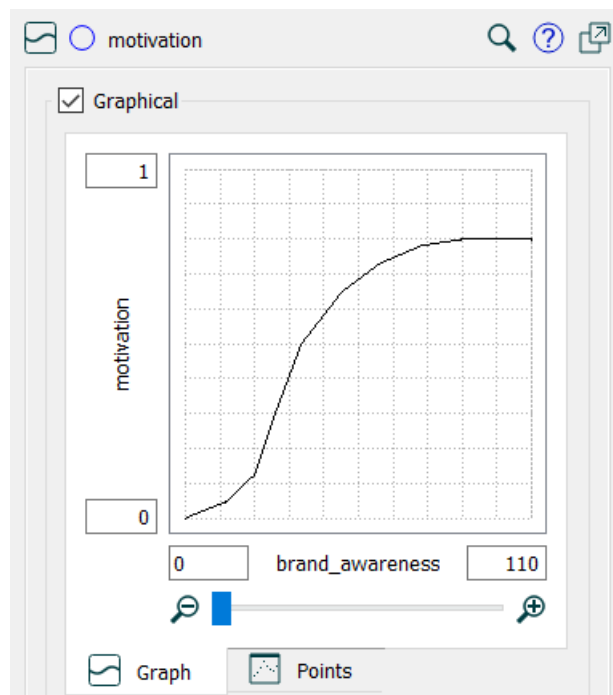


Figure 28 Motivation table function

Brand awareness (BA), this stock shows the level of brand awareness in the society as the number of current installed base increase so does the brand awareness.

$$BA = (ChgBA)dt + BA(0) \quad (68)$$

Change in Brand awareness (ChgBA):

$$ChgBA = \left(\frac{AM * BAG}{TCBA} \right) - \left(\frac{FF * BA}{TCBA} \right) \quad (69)$$

This equation shows the different between increase in brand awareness and loss of brand awareness. AM is the awareness multiply, BAG is brand awareness gap, TCBA is time to change brand awareness and FF is forgetting fraction.

Brand awareness gap (BAG):

$$BAG = MaxBA - BA \quad (70)$$

Where MaxBA is the maximum brand awareness and its equal to 100.

Awareness multiplier (AM), this is the main input that increase brand awareness and it depends on EFAAM which is the effect of fraction awareness on awareness multiplier, ELBA which is the effect of brand loyalty on brand awareness and BAE which is the brand awareness effectiveness.

$$BA = IF T > TE \text{ then } (EFAAM * ELBA * BAE) \text{ else } 0 \quad (71)$$

Brand awareness effectiveness (BAE): This variable is a table function made by Peter A. Otto and J. Robert Bois [26]. Using fraction of BA reached as an input to graph presented in figure 29.

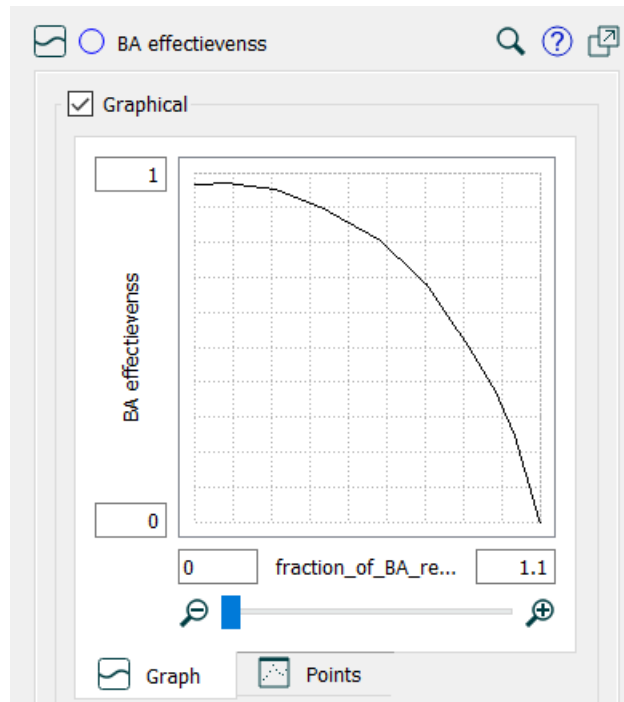


Figure 29 BA effectiveness

Fraction of BA reached (FBAR), is represent the fraction of brand awareness that reached to the maximum brand awareness that is possible.

$$FBAR = BA/MaxBA \quad (72)$$

Effect of loyalty on BA (ELBA): This variable is a table function made by Peter A. Otto and J. Robert Bois. Using brand loyalty as an input to graph presented in figure 30.

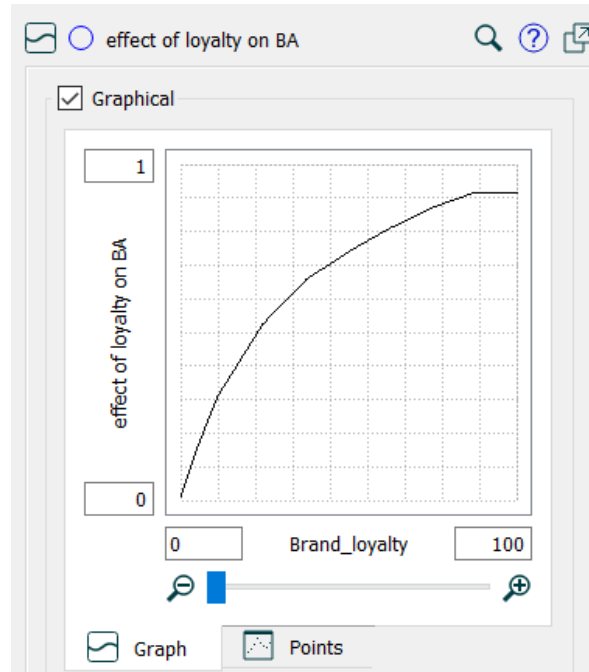


Figure 30 Effect of brand loyalty on Brand awareness

Effect of fraction awareness on awareness multiplier (EFAAM):

$$EFAAM = SAFEDIV (FA, TFA, 0) \quad (73)$$

Where FA is fraction awareness and TFA is the total fraction awareness.

Fraction awareness (FA), shows what fraction of the total population are the OS users.

$$FA = \frac{CI}{P} \quad (74)$$

Where CI is the current installed base and P is the total population.

Total Fraction awareness (TFA), shows what fraction of population are smartphone users.

$$TFA = \frac{Sum (CI)}{P} \quad (75)$$

3.2. Results of simulation:

In this section we are going through some of most important results from the model simulation, however we will discuss and compare the behavior with the reference data in the next chapter. The purpose of this section is to analyze the endogenous dynamics of the model and show how behavior of different variables leads to the final results in the market.

Table 21 shows the model setting for the main simulation of this thesis:

Table 21

Model setting

Model Setting	
Start time	2000
Stop time	2017
DT	1/4
Time units	year
Integration method	Euler

3.2.1. Market section:

In market section there are crucial variables with interesting behaviors, in the following paragraphs we find the reasons behind those behaviors.

Innovation diffusion:

As Sterman said, the diffusion and adaption of new ideas and new products often follows S-Shaped growth patterns. As figure 31 Shows the behavior of the smartphone diffusion in this model is also S-Shaped but in this time frame we can just observe the increasing increasingly part of growth. The increasing increasingly part happens because the unaware population is high so the process of they become aware of the technology while interacting

with others is very fast. Figure 32 shows the simulation when we extend the time frame in order to test the S-shaped structure.

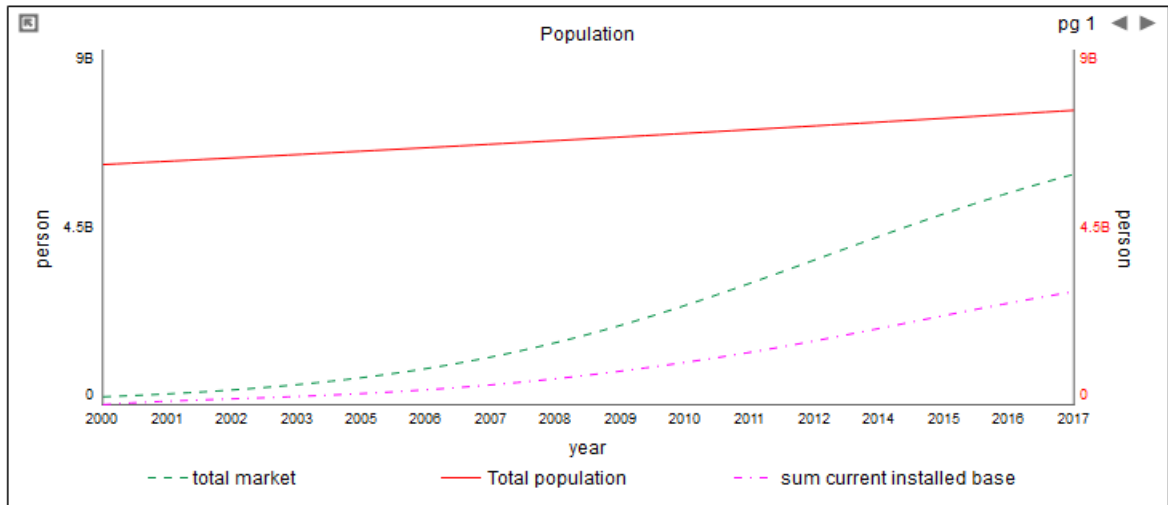


Figure 31 Comparing total population and total market 2000-2017

Total market is the number of people who are aware of the smartphone technology, either they are using one (SCI) or they are in the stage of choosing one (PI). As figure 31 shows the population is increasing with the fix growth rate of 1.2% and the total market is following that. According to figure 32 total market for the first 16 years is increasing increasingly, and after that its increasing decreasingly. The reason is in the beginning when more population are being aware of the technology, they influence the unaware population fast but that decreases the Ever-likely users so limited number of unaware people remains in the Ever-Likely user stock and shows goal-seeking behavior.

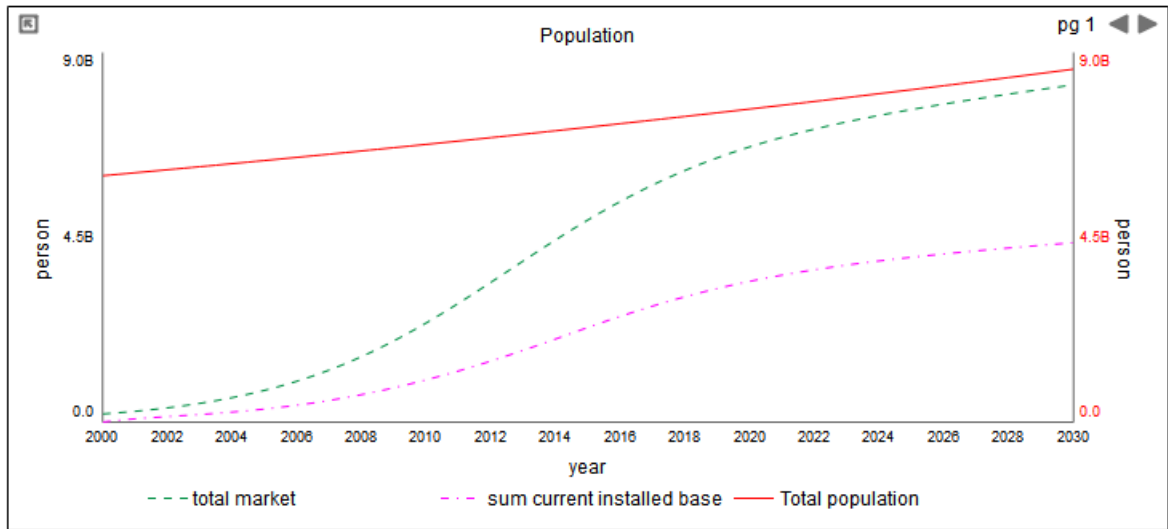


Figure 32 Comparing total population and total market 2000-2030

Figure 33 shows the behavior of Ever-likely users, potential installed base and sum of current installed base, one can see that while two later stock are increasing the former is depleting. The stock of potential installed base increases because the adaption rate increases, also this stock positively affects the sales rate and Sum of current installed base. Meanwhile the adaption rate which is the outflow from Ever-Likely users stock is much higher than net population growth rate and this causes this stock to be depleted.

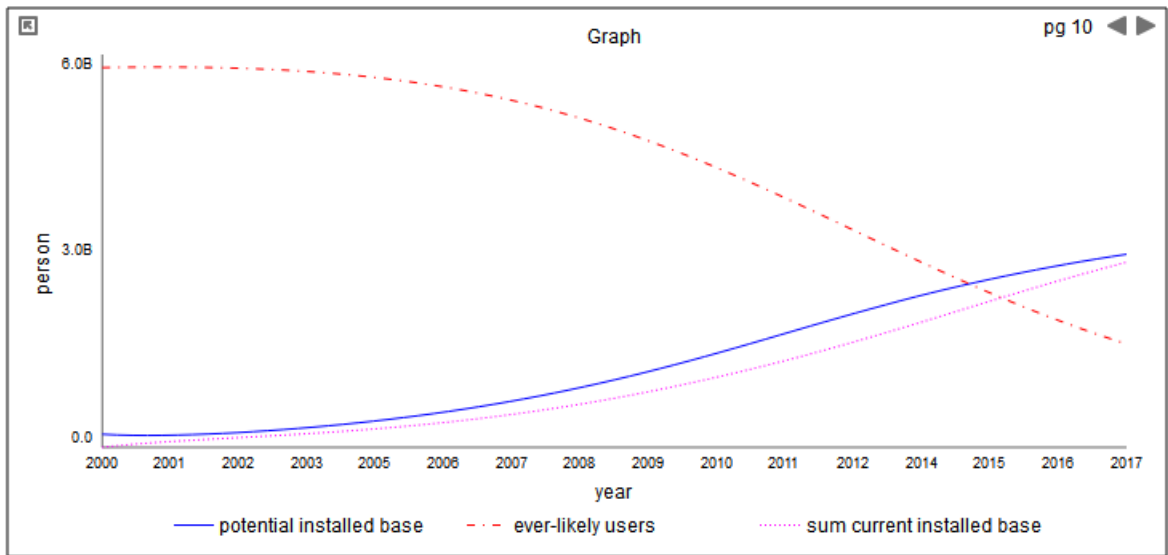


Figure 33 population stocks comparison

OS attractiveness, Indicated market share and sales rate:

These three crucial variables are highly affecting each other's, OS attractiveness depends on four categories of characteristics which are being influence by sales rate and market share. OS attractiveness has positive and direct effects on Indicated market share, and sales rate is being calculated by Indicated market share along with the stock of potential installed base. We will discuss OS attractiveness and its four categories in the next section, in this section we are going to analyze the behavior and effect of these three on each other's.

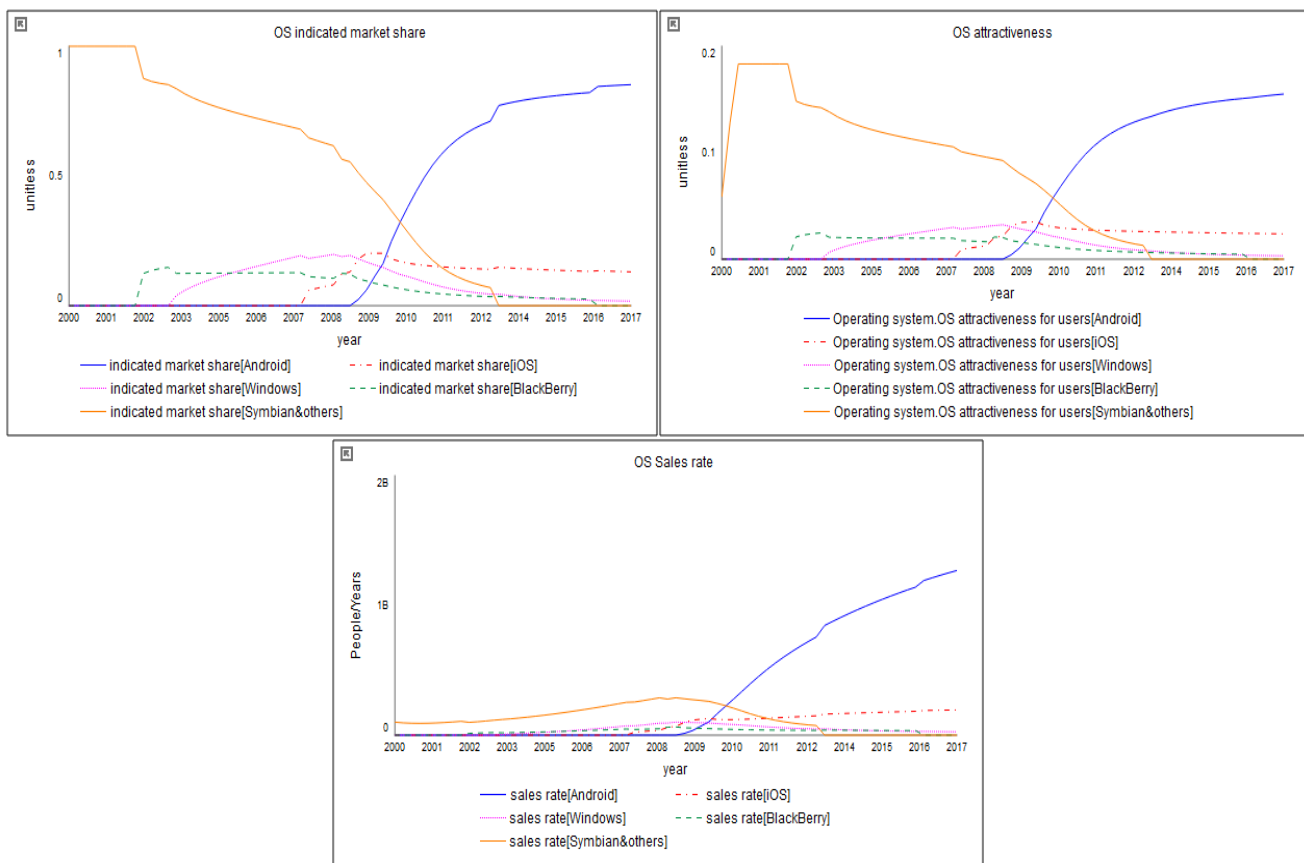


Figure 34 OS attractiveness, OS indicated market share and OS sales rate comparisons

As figure 34 Shows, the behavior of the OS attractiveness and OS indicated market share has so much similarities. While OS attractiveness is being calculated separately for each OS, OS indicated market share shows what fraction of OS attractiveness belongs to each OS. Sales rate also shows some similarities to others but since it also depends on the stock of potential

installed base, and that stock has different behavior so the sale rate graph is not totally follow the behavior of the OS indicated market share.

The interesting fact is how Android dominant the market shortly after introducing to the market, in the following section we are going to investigate reasons behind this fast growth.

3.2.2. Operating system section:

OS attractiveness:

Figure 35 shows the trend in the OS attractiveness for users in the market from 2000 to 2017 according to the simulation. As we mentioned before this variable depends on four category of OS characteristics which are; Standard support strategy, other stakeholders, characteristics of the standard and characteristics of standard supporters. Their effects can be direct such as effect of brand loyalty or indirect like the group of standard support strategies, and each factor has different weight.

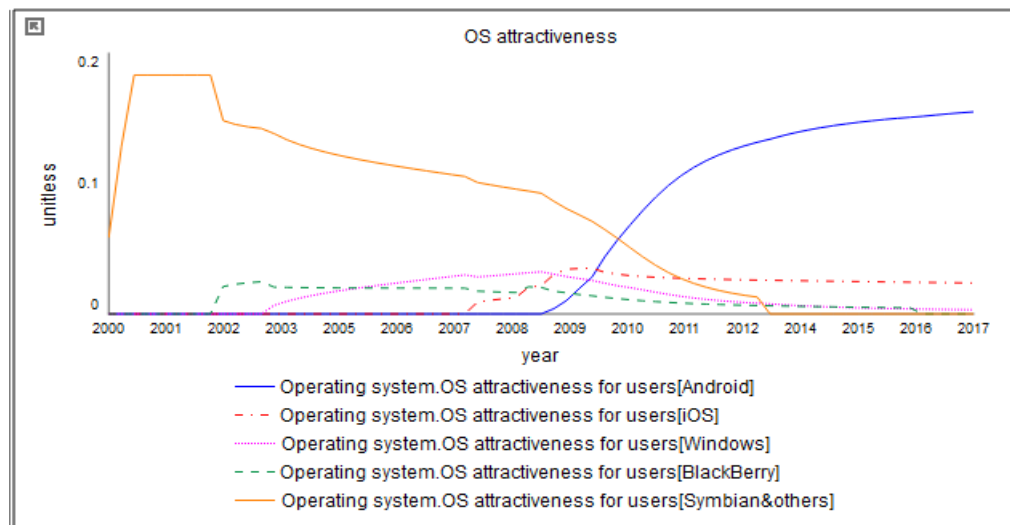


Figure 35 OS attractiveness for users

In following paragraphs we are going to analyze the behavior of each factors separately and their final effects on the OS attractiveness:

Standard support strategy category: The main outcome of this category is the “OS attractiveness for supports”, which shows the attractiveness of OS for supporters and how

they make decisions on adapting one. Since iOS and BlackBerry OS have a strong appropriability strategy, their OS attractiveness for supporters are zero all the time.

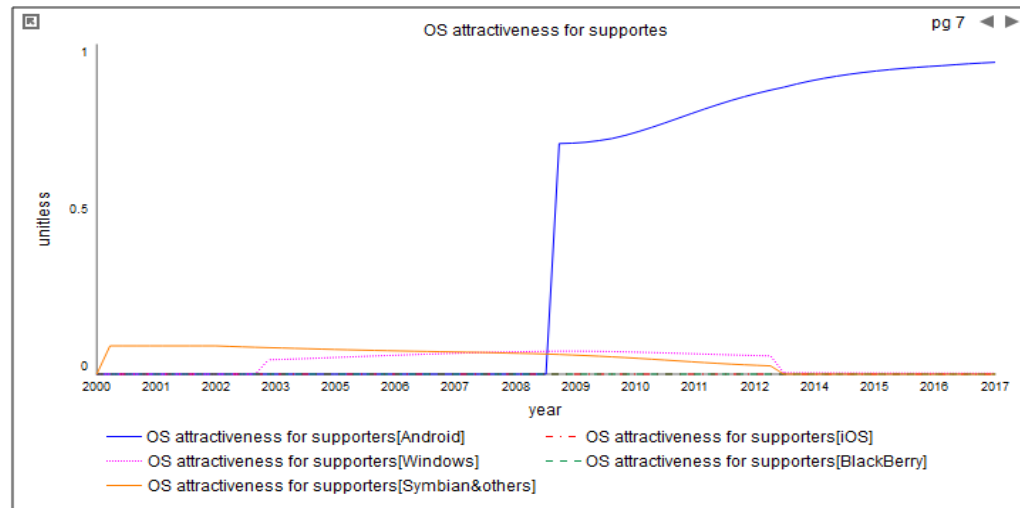


Figure 36 OS attractiveness for supporters

That leaves us with other three Operating systems, in the beginning of the simulation, Symbian is the only available OS in the market so it has the highest attractiveness. Later when Windows phone enter the market even though it has higher appropriability attractiveness for supporters it cannot overcome Symbian because Symbian has much higher market share. The interesting turnover happened when Android entered the market, despite the fact that Android has a low market share in the beginning since it is open-source and license-free, its attractiveness for supporters is high and the low market share does not affect supporters' decision in adapting Android.

Other stakeholders' category: There are three attractiveness factors in this category that directly influence the OS attractiveness, Bandwagon attractiveness, Diversity of network attractiveness and Model attractiveness.

Bandwagon attractiveness shows the importance of OS current installed base on the OS attractiveness. Figure 37 shows the behavior of this variable for each OS, as one can observe this value for Symbian was the highest until 2012, when Android took over the market.

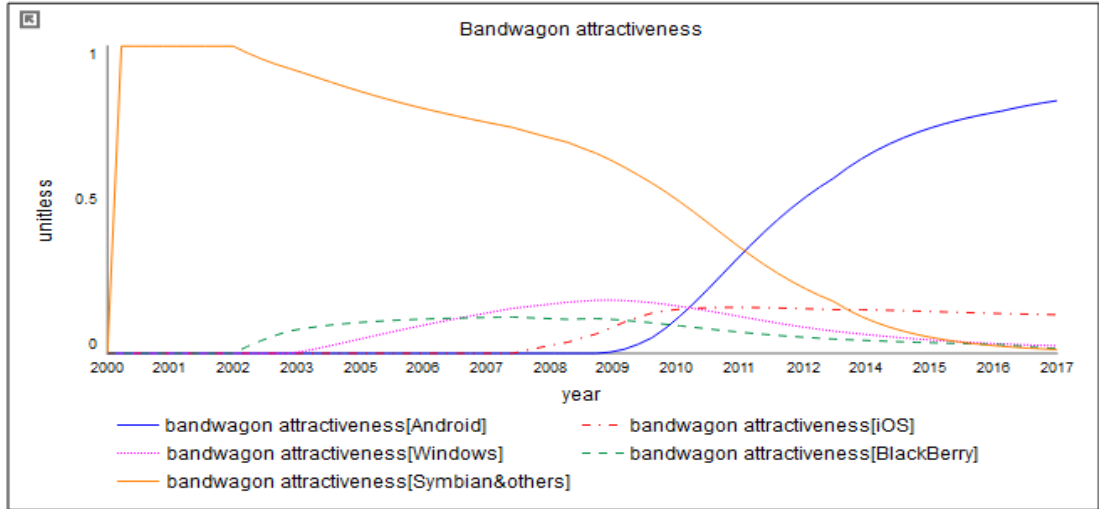


Figure 37 OS bandwagon attractiveness

Network Diversity attractiveness, this factor shows the importance of availability of OS supporters in the market. While this value for iOS and Blackberry is low due to their strategy, there is a competition for others. Figure 38 shows this competition and as we described before since the OS attractiveness for supporter for Android is high, it affects this factor and Android beats Symbian few years after entering the market.

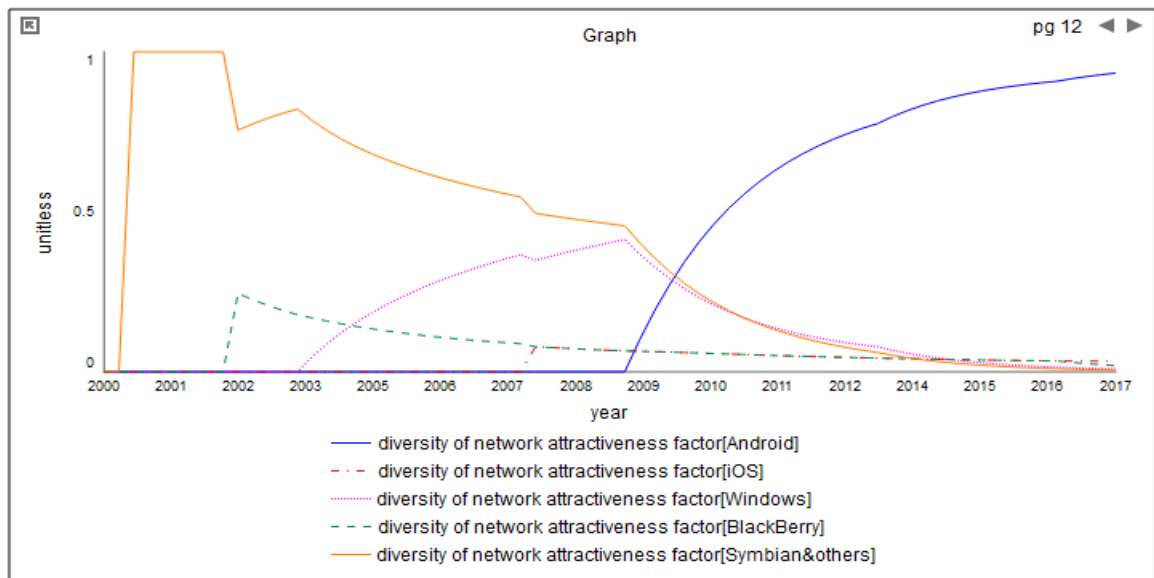


Figure 38 OS diversity of network attractiveness

Model attractiveness factor, in this variable we observe what portion of available models in the market belongs to each OS. This factor depends on both supporters and the OS sales rate in the market, as the sales rate increase the manufacturers increase and adjust their capacity to meet the market needs. Operating systems such as iOS and BlackBerry are being produce by one manufacturer so their attractiveness is less than others. Those are being produced by manufacturers with willingness to produce as many model as possible, such as Samsung and Sony.

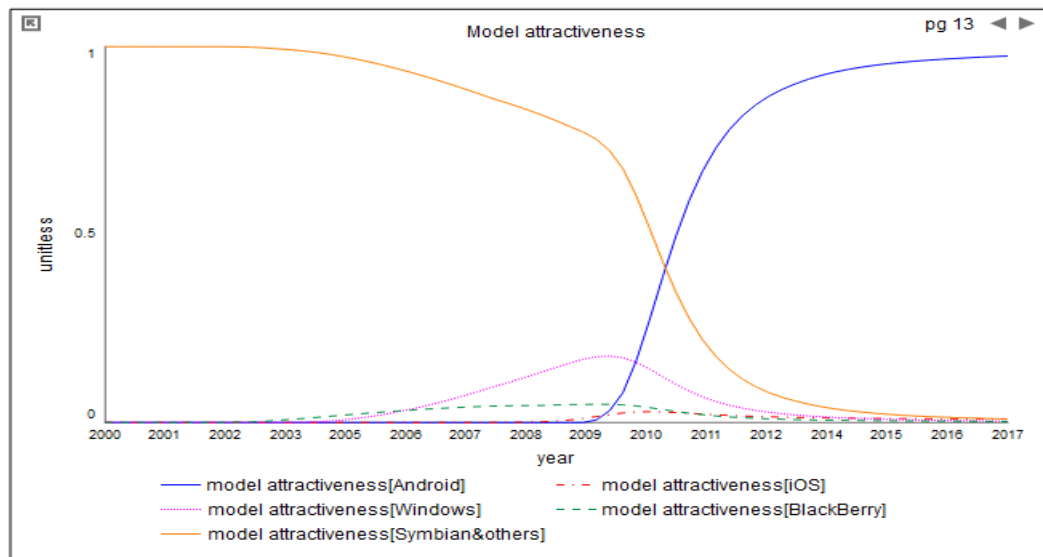


Figure 39 OS model attractiveness

The next category of characteristics is characteristics of the standard, it has one factor that effect the OS attractiveness. That factor itself depends on two other attractiveness factors; Application attractiveness factor and complementary good attractiveness factor.

Application attractiveness factor: This factor shows what fraction of available applications in the market that belongs to each OS, obviously OS with higher available applications attract more users. This stock depends on different factors such as, flexibility of OS for developers, registration fee, App store availability and sales rate. According to studies Symbian flexibility for developers was very low and it was hard to develop a functional application and that is one of the Symbian’s biggest failure reasons. According to Figure 40, this factor has lots of up and down for each OS, the reason is each time one OS enters the

market based on its attractiveness for developers it can reduce the application attractiveness of others.

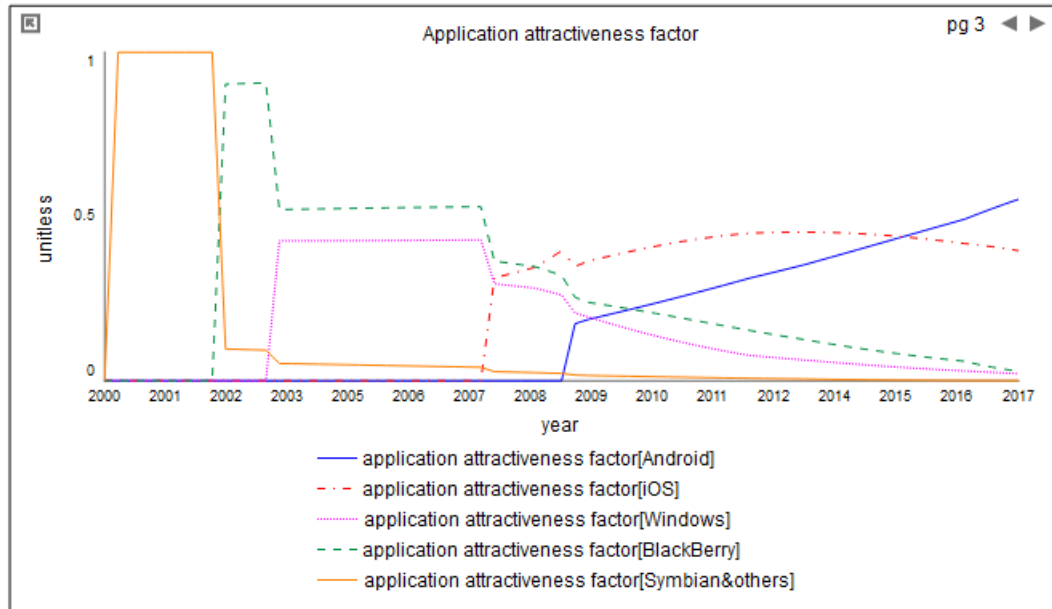


Figure 40 OS application attractiveness

Complementary goods attractiveness factor, this factor is being influence by two main variables, stock of supporters and OS sales rate. Figure 41 shows the behavior of this factor, most of the operating systems did not have Complementary goods at the beginning. In 2008 BlackBerry and Apple started to produce complementary goods, and shortly after that other manufacturers start producing complementary goods such as smartwatch, fitness band and smart TV for each OS.

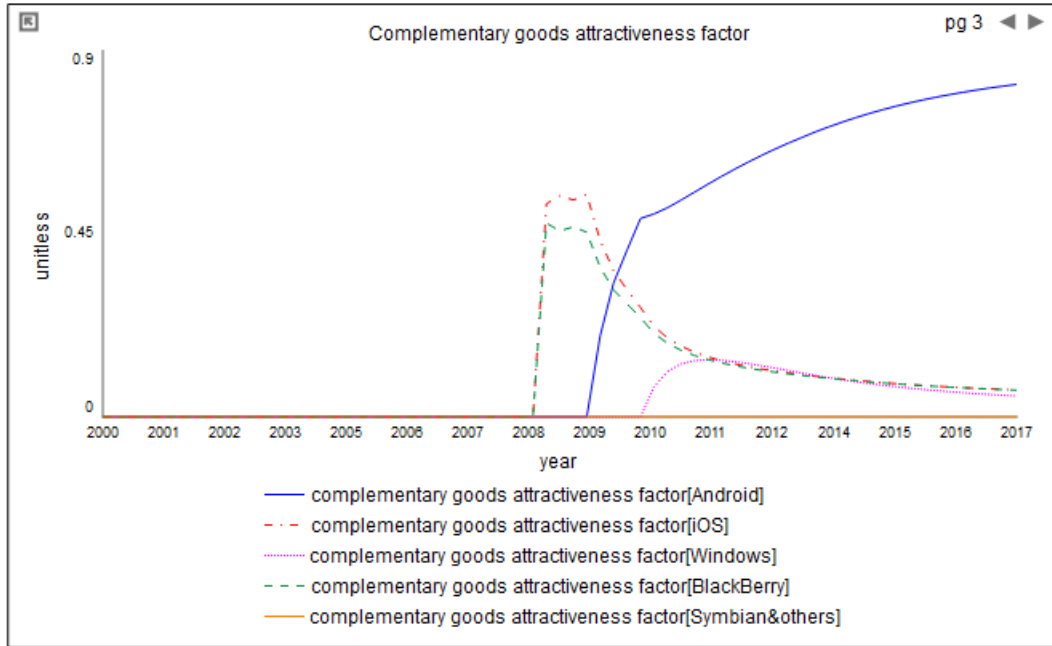


Figure 41 OS complementary goods attractiveness

Android has the biggest stock of supporters so clearly they produce higher number of complementary goods, and that is the reason of Android taking over in this factor as well. Figure 42 shows the average between application and complementary goods attractiveness, this value is the one that affect OS attractiveness.

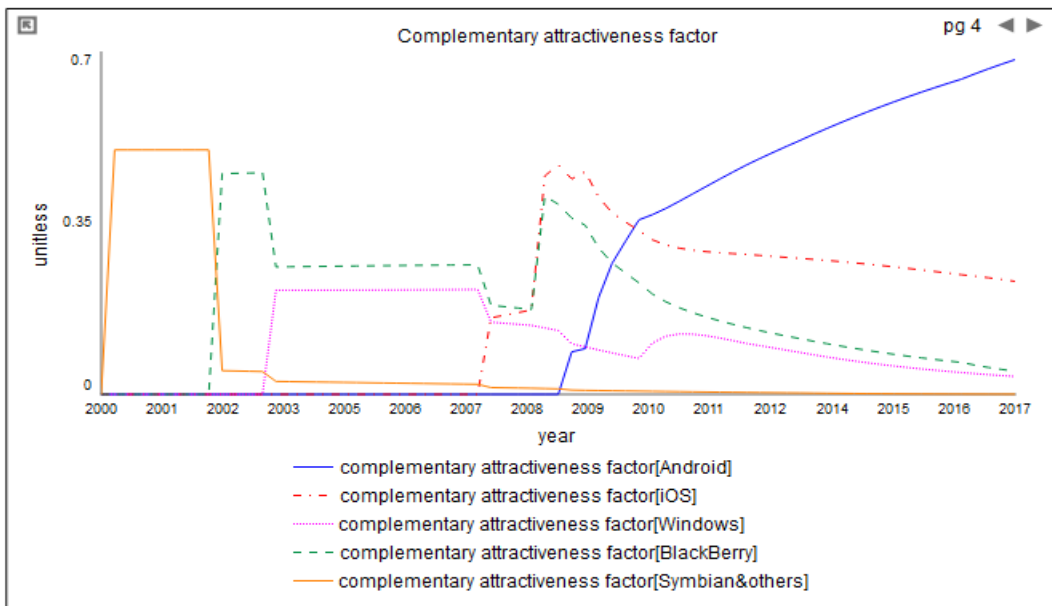


Figure 42 OS complementary attractiveness

The last category is characteristics of standard supporters, the important factor in this category is the brand loyalty attractiveness factor. The stock of complementary goods and brand awareness has positive effects on the brand loyalty. Another important factor is the ecosystem strategy of OS owners, by ecosystem we mean to what extent the complementary goods are compatible with the specific OS that they are being made for. For example Apple TV is just compatible with iOS, while Sony smart TV is better functional with Android but still it is possible to use iOS in order to connect to the TV. That explains the behavior of iOS and Android at the end, Android has the most complementary goods due to the highest number of supporters and iOS has the highest ecosystem for the users which increases the users' loyalty.

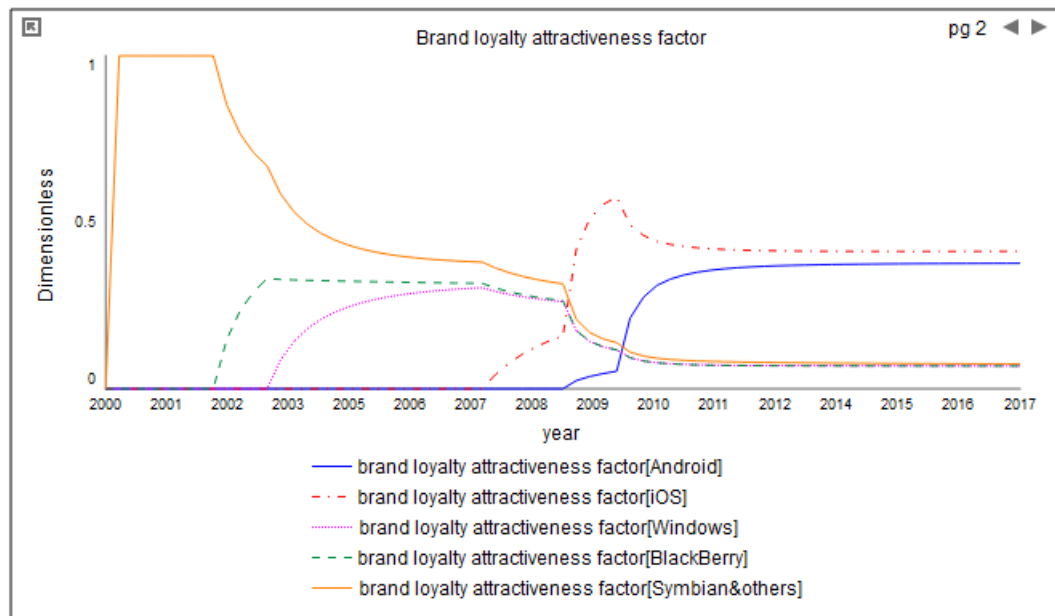


Figure 43 OS brand loyalty attractiveness

4. Testing:

4.1. Boundary adequacy test:

As Sterman [3] stated, Boundary adequacy tests assess the appropriateness of the model boundary for the purpose at hand. The first step is to determine what the boundary is, in the beginning of the thesis the table 3 And the figure 7 are showing the model boundary and the subsystem diagram of this model. Those are being made according to the framework suggested by G. Van de Kaa.

As we explained in recognizing the most important factors in this battle we chose factors G. Van de Kaa suggested in the his “revised model for standard dominance for converged systems” [18] also we used G. Van de Kaa’s analysis[5] for allocating weights to them.

In table 22, we included all the factors from the model and using the suggested weights. This table shows the distribution of the factors in both studies, the green cells shows the factors from the model [18] and yellow ones are factors from the framework that are being included in the thesis. There are six factors from the framework that we did not cover in the model, some of them may have minor effects on the results. Even they may explain minor inconsistency between data references and the simulation result. As Sterman [3] said in this test our concern is whether any feedbacks omitted from the model, if included, might be important given the purpose of the model.

Table 22

Comparison table for model boundary

	Weights	Model
Diversity of the network	0.1897	Yes
brand reputation	0.1825	Yes
financial strenght	0.1159	
curent installed base	0.0904	Yes
timing of entry	0.0874	Yes
compatibility	0.0685	
complementary	0.0542	Yes
marketing communications	0.0442	
technological superiority	0.0437	
effectiveness of development proces	0.0424	
appropriability strategy	0.0422	Yes
learning orientation	0.0398	

4.2. Dimensional consistency:

Dimensional consistency is one of the most basic tests and should be among the very first tests you do. Dimensional inconsistency may reveal nothing more than a typo-graphical error, an inverted ratio, or a missing time constant. More often, units errors reveal important flaws in the understanding of the structure or decision process we are trying to model [3].

Stella architect software is one of the system dynamics software packages that include dimensional analysis so in every steps of the modeling one can check the unit consistency of the model. According to this software this model has unit consistency, but since another test includes checking equations for variables with strange name and units so we included all the equation in the appendix in order to show all variables are carefully named and units are also carefully assigned.

4.3. Extreme condition test:

Models should be robust in extreme conditions. Robustness under extreme conditions means the model should behave in a realistic fashion no matter how extreme the inputs or policies imposed on it may be [3]. Below we are going to present the results of extreme condition tests for three of the exogenous variables.

4.3.1. Adaption fraction:

The normal value of this variable is 1% which shows what fraction of contact between Ever-likely users and current installed base is positive. Table 23 show the value for this variable in order to implement extreme condition test.

Table 23

Adaption rate for extreme condition test

	Normal value	Max Value
Adaption rate	1%	20%

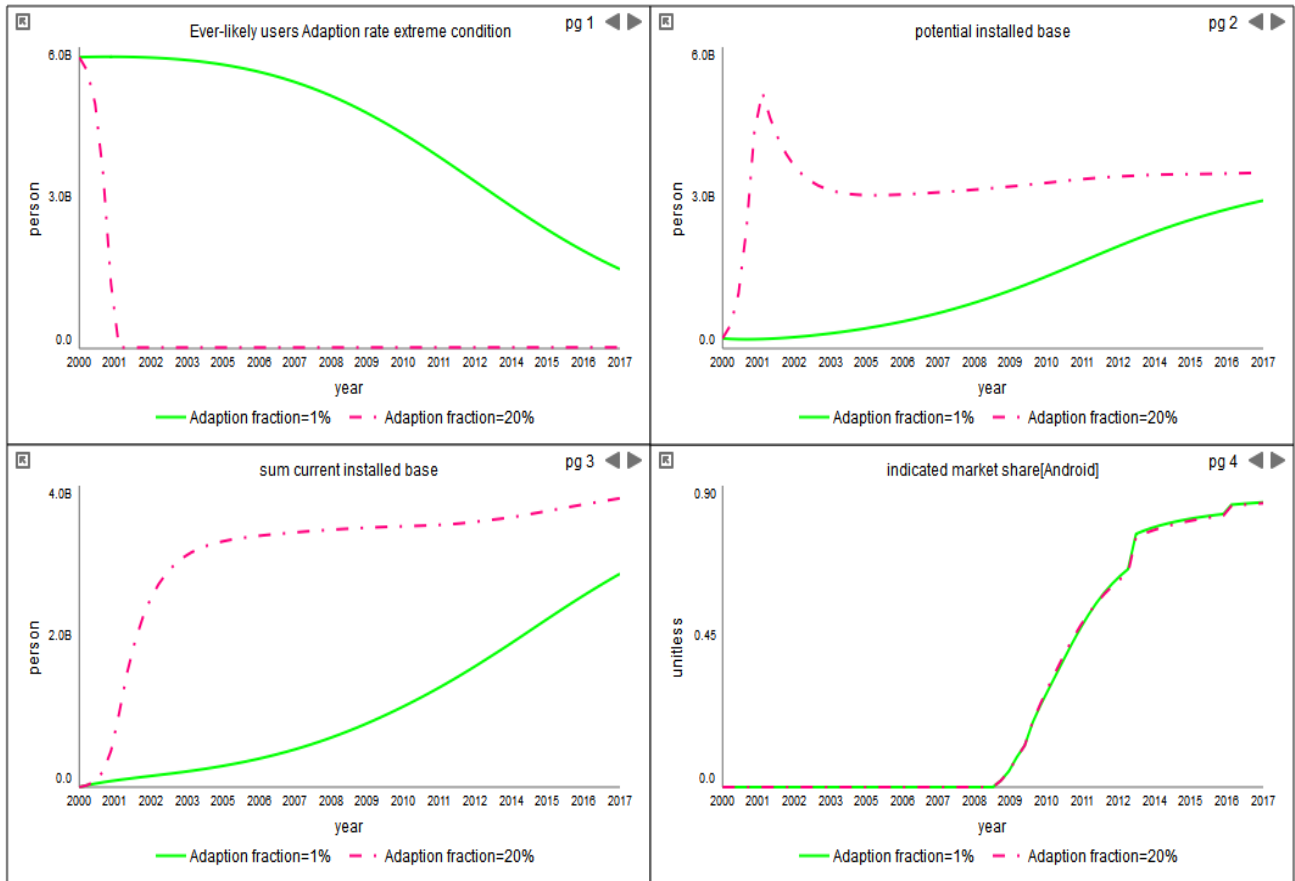


Figure 44 Extreme condition test- Adaption fraction

Figure 44 shows the result of this extreme test for four different values, obviously when adaption fraction increases unreasonably the stock of Ever-likely users deplete so much faster than usual because now the higher percentage of contacts are successful. Also with higher adaption the stock of potential installed base and sum of current installed base also grow faster. The interesting behavior is being shown by OS indicated market share, we show just one of the OSe in this case it's Android. As it is shown in the fourth graph the behavior of indicated market share follows the same pattern and does not depends on the adaption fraction because for calculating OS attractiveness we are using the ratio of current installed base or sales rate.

4.3.2. Time to change:

Normally users change their phones in 2 years and according to the model this can decrease when the other OS attractiveness is high. In this extreme test we are going to shock

the system by assuming the market is much faster and users change their phones within 6 months or less.

Table 24

Time to change for extreme condition test

	Normal value	Max Value
Time to change	2.00	0.5

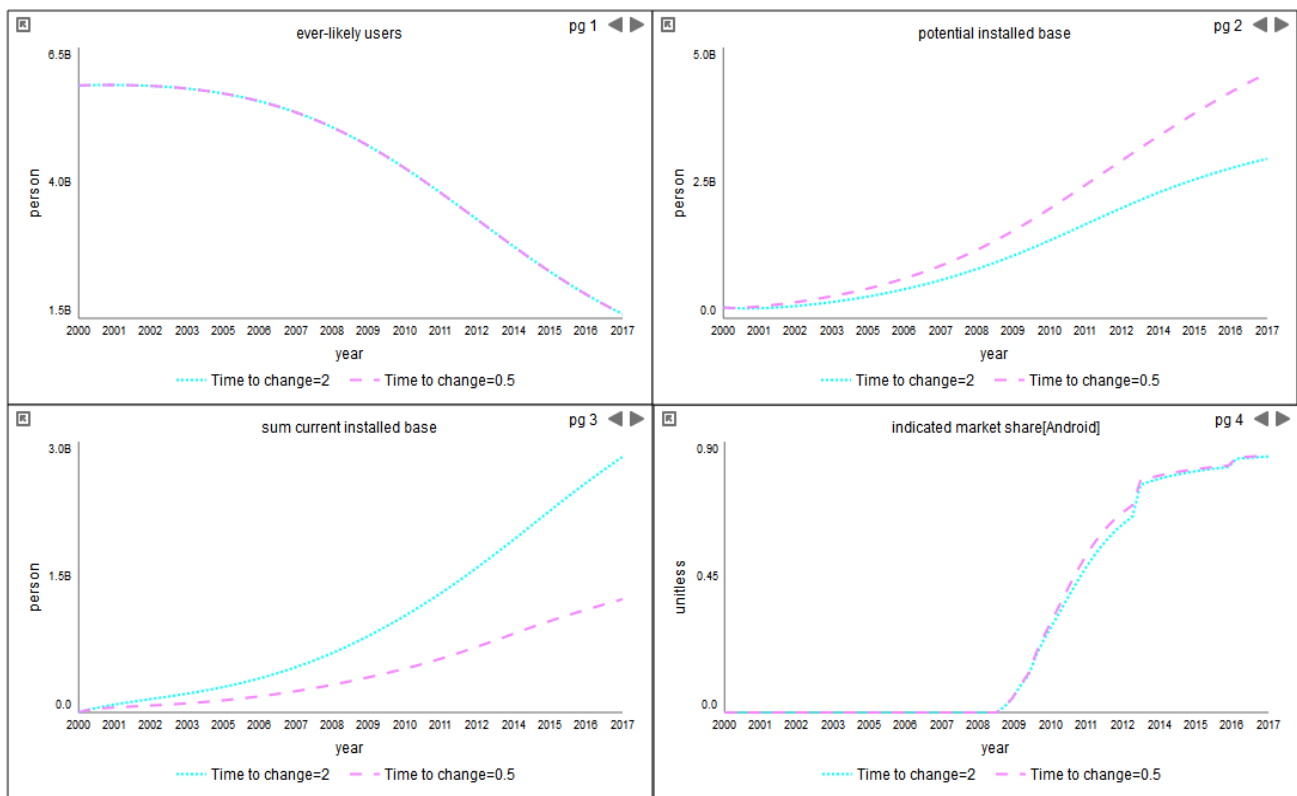


Figure 45 Extreme condition test-Time to change

By changing this variable we are increasing the speed of market that means people are willing to change their phones faster. That is the reason behind the fact that in this extreme condition sum of current installed based in much lower than normal situation and the potential installed base stock is higher because now people come back to this stock sooner than before. Change in this variable does not influence the ever-likely users stock, because according to its equation the adaption rate depends on the contact between total market

and ever-likely users and the fact that people are moving between the two stock of potential and installed base does not change the number of total market.

4.3.3. Open Strategy extreme test:

In this test we consider that all the available operating systems in the market have the highest appropriability, which means all of them are available free of charge for supporters. Figure 47 shows the behavior of current installed base stock for all operating systems.

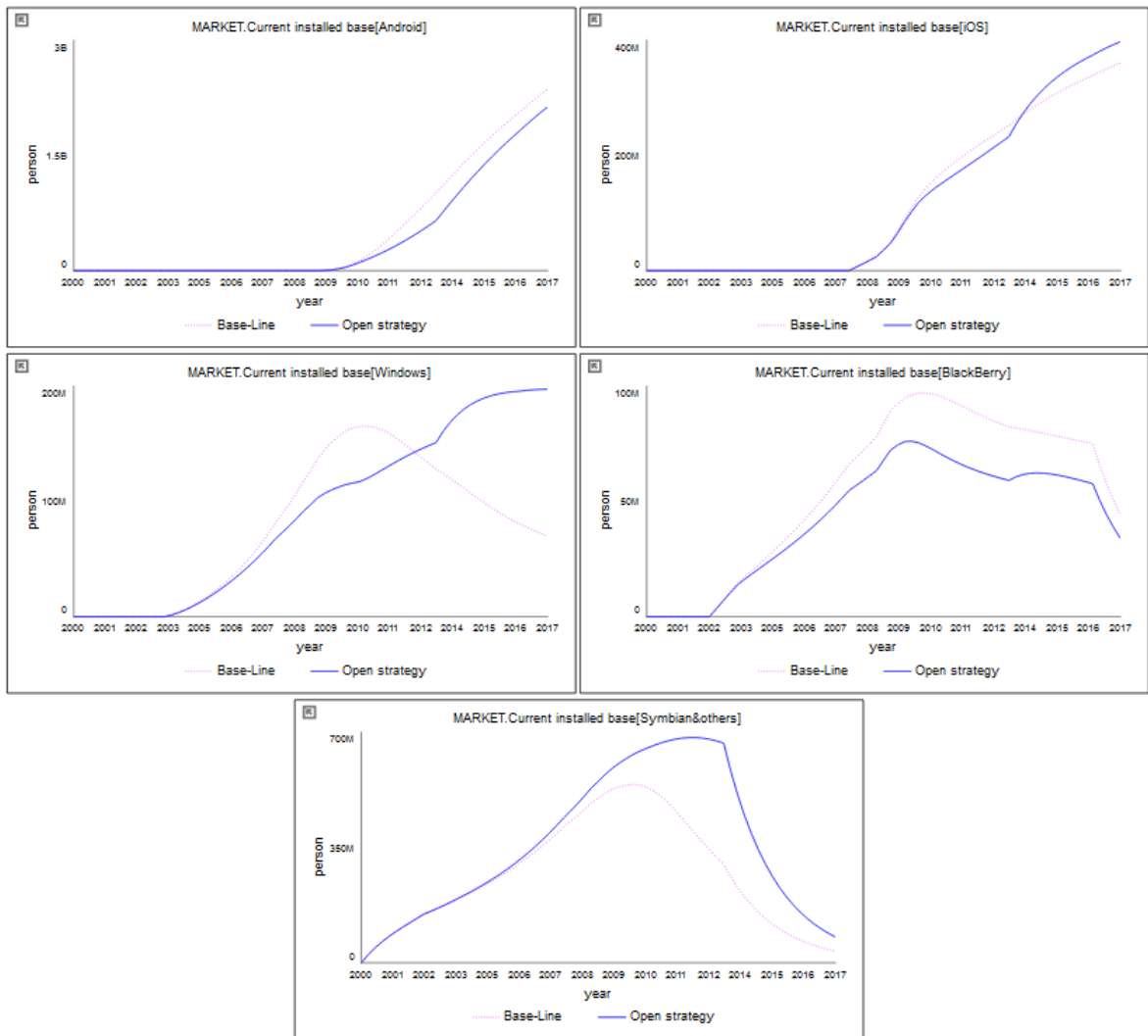


Figure 46 Extreme condition test-Open strategy

Obviously when all the operating systems are free for manufacturers to adapt, the market share and current installed base of Android (higher appropriability in base line) will

decrease while others grows. Keeping in mind that the technological flexibility of OS for wither supporters or developers stay at the same value.

4.4. Integration error test:

System dynamics models are usually formulated in continuous time and solved by numerical integration. You must select a numerical integration method and time step that yield an approximation of the underlying continuous dynamics accurate enough for your purpose [3]. According to Sterman [3], always test for such “DT error” by cutting the time step in half and running the model again. If the results change in ways that matters, the time step was too long. Continue until the results are no longer sensitive to the choice of time step.

For implementing this test we examine the model in 3 different DTs, these values are being shown in table 25:

Table 25

Integration error test

	Min DT	Normal DT	Max DT
DT	1/8	1/4	1/2

Figure 47 shows the comparison of OS indicated market share for OSes for different DTs, it can be imply from the graphs that the change in DT does not have a major impact on the final results of the model. Also according to the data when DT is equal to $\frac{1}{4}$ the results best match the reference data.

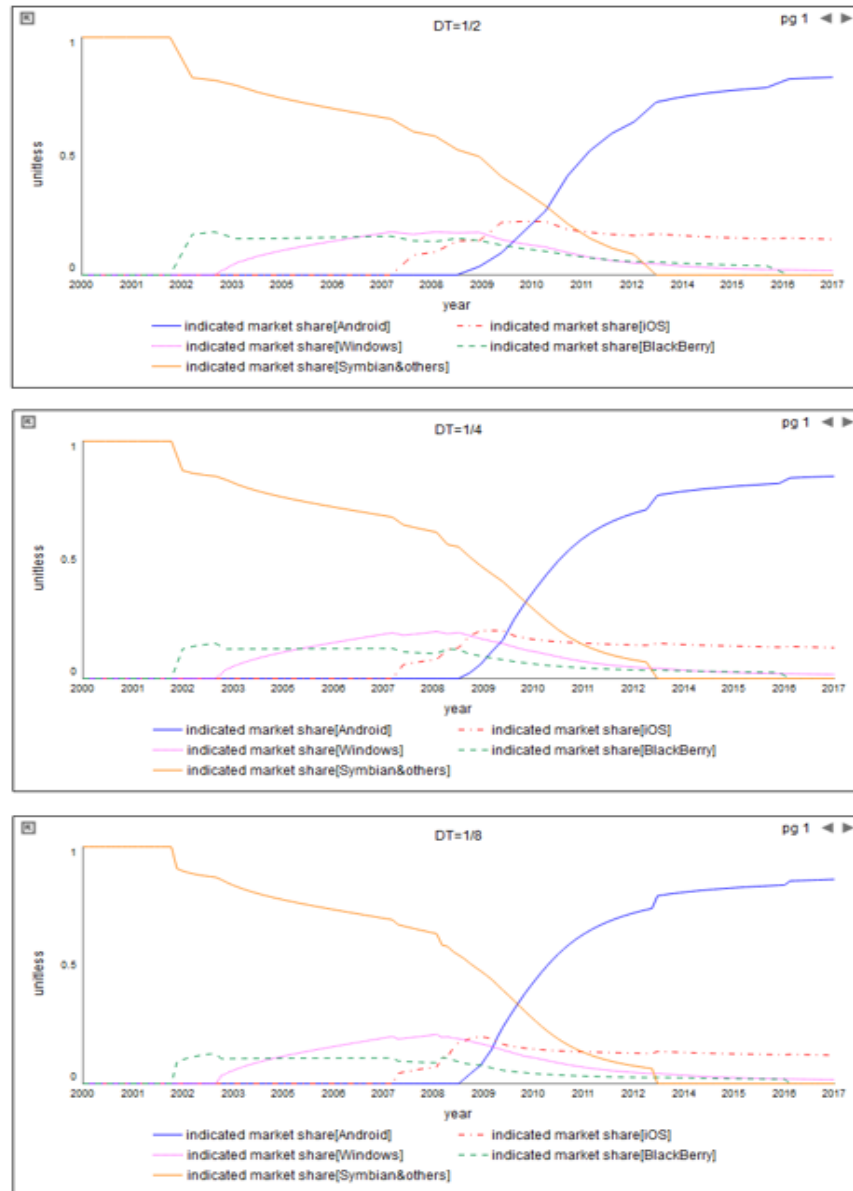


Figure 47 Integration error test

4.5. Behavior reproduction:

The proper use of the behavior reproduction test is to uncover flaws in the structure or parameters of the model and assess whether they matter relative to the purpose. Instead of showing how well your model fits, you should point out to your clients all the places it doesn't. These discrepancies mark the trails that can guide you to erroneous parameter estimates and inappropriate assumptions you should revise before using the model for policy analysis [3].

Before going through graphs that show the behavior of the simulation and data reference, it is important to know that during the modeling phase and after finishing the main structure we implemented this test. The results were so close in pattern but needs parameter adjustment (Figure 48 shows this comparison for Android).

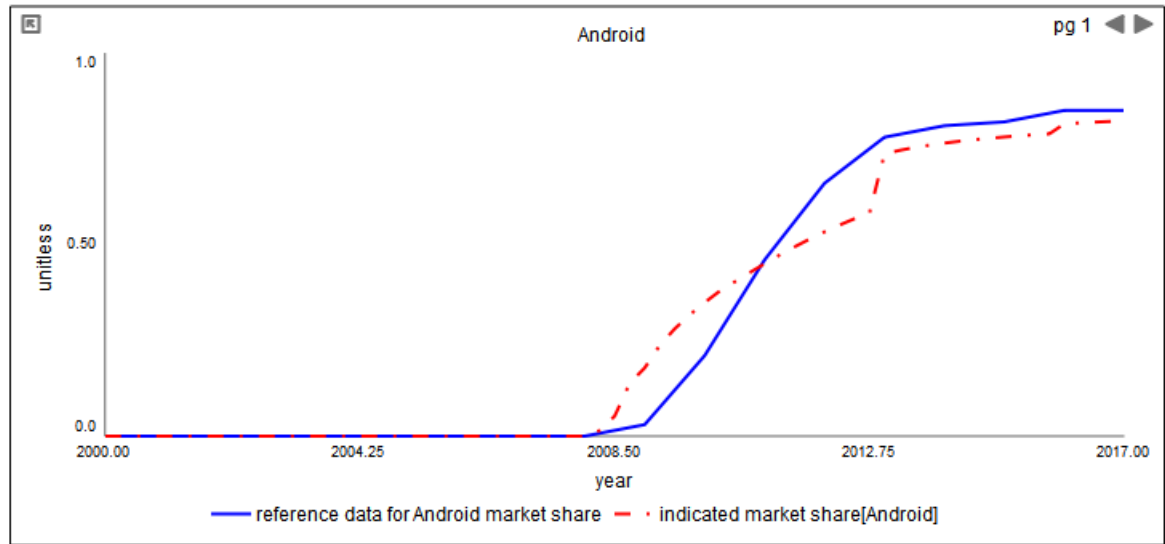


Figure 48 Android First comparison graph-without model section

After going through what is happening in the market, we found out that while there is no factor for the variety of model in the framework. This factor has a significant role in this battle, so we add the new structure and adjust attribute weights again as we described. As we mentioned it is almost impossible to consider all factors that effect this battle, so the results can be different or more reliable when we add all the factors to the model.

Figure 48 to Figure 53 are showing comparisons of the reference data and results from the simulation. The data in the period from 2008-2016 was available but the data from 2000-2008 was inconsistent and not completely valid so we used different sources to collect data. One reason behind the different between simulation and data during 2000-2008 can be because of that.

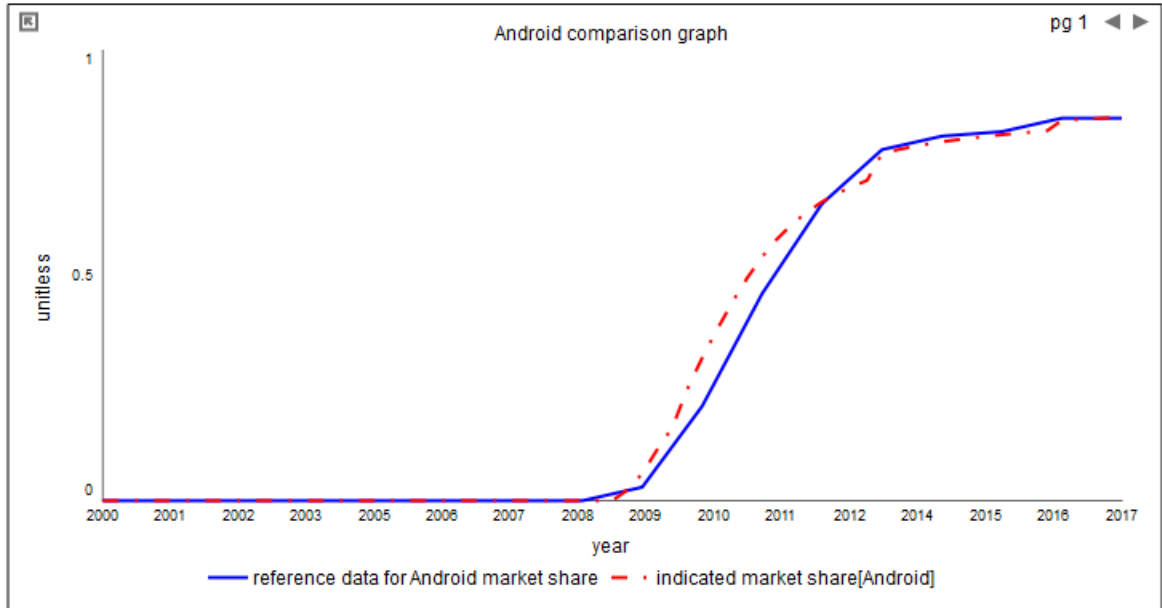


Figure 49 Android comparison graph- Data reference and simulation

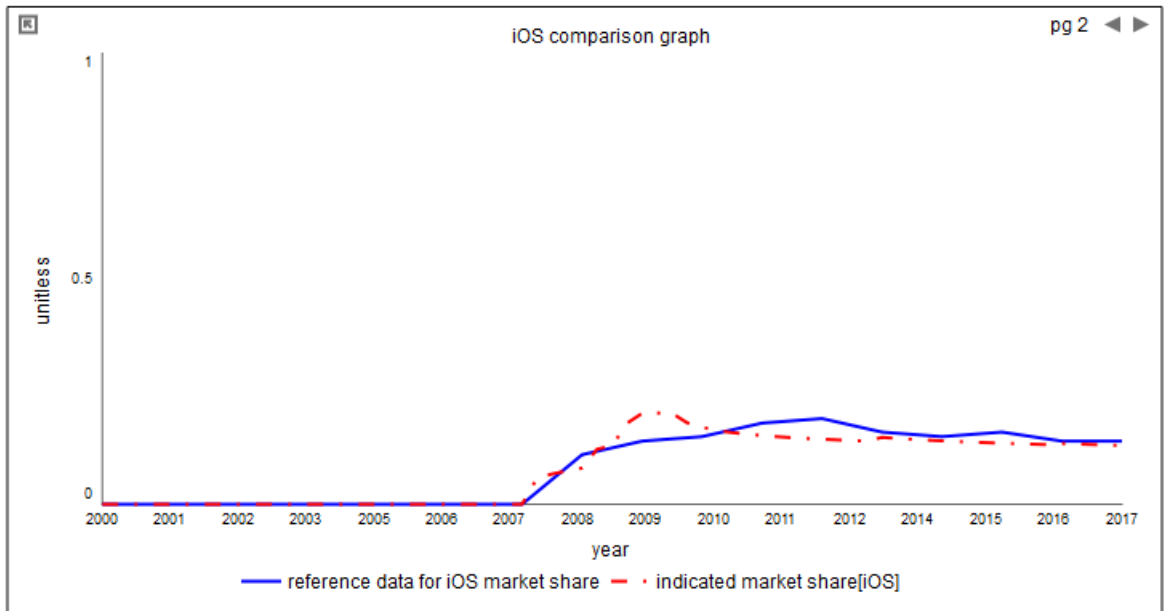


Figure 50 iOS comparison graph- Data reference and simulation

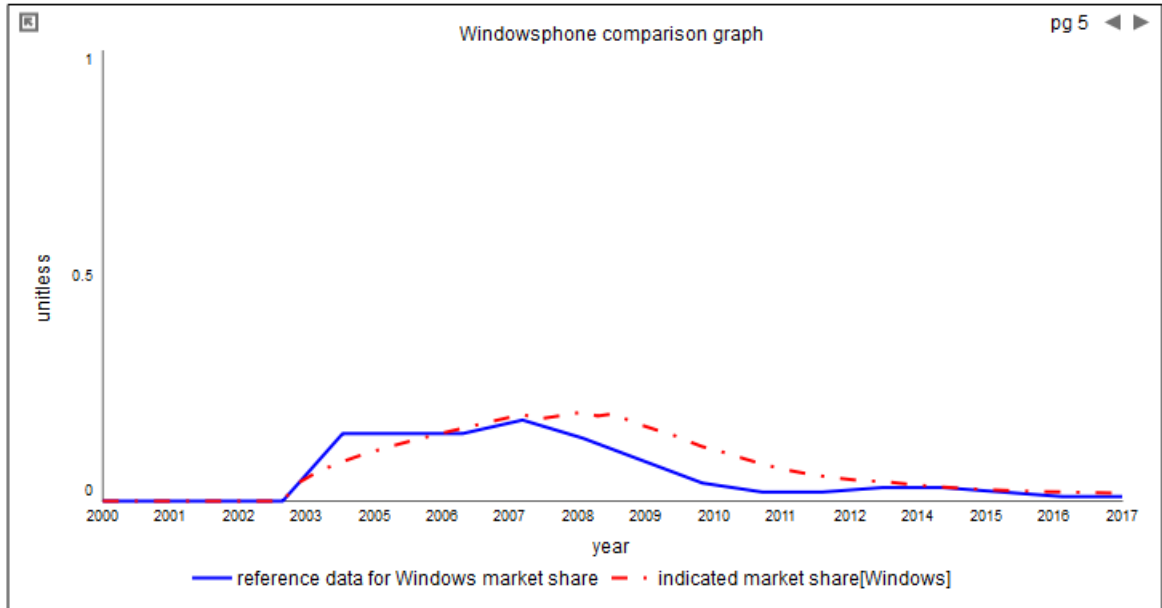


Figure 51 Windows Phone comparison graph- Data reference and simulation

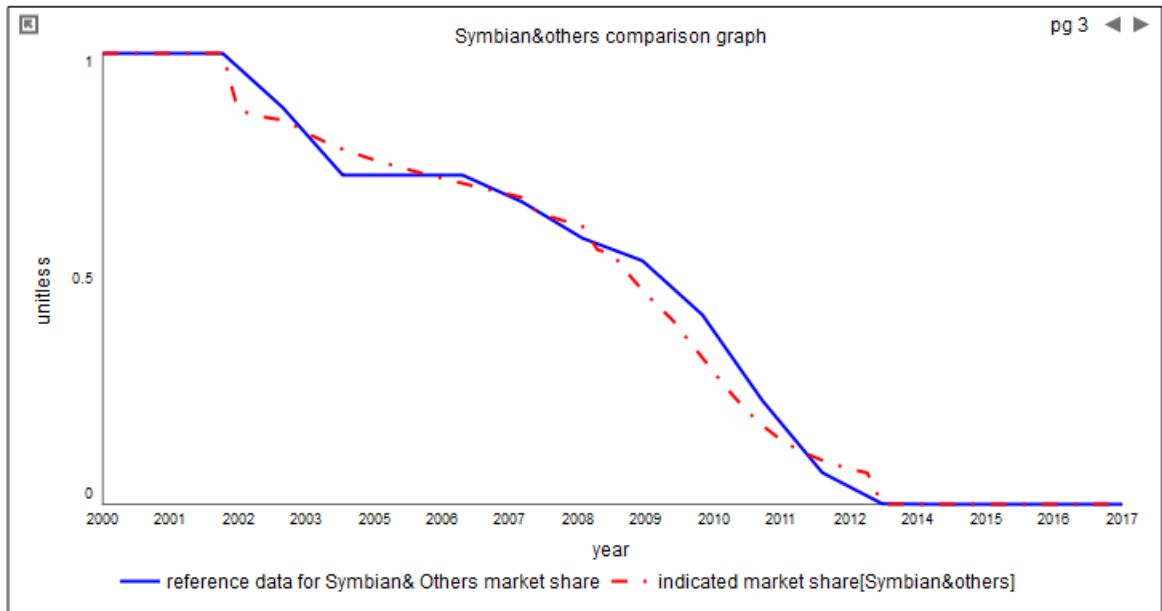


Figure 52 Symbian & Others comparison graph- Data reference and simulation

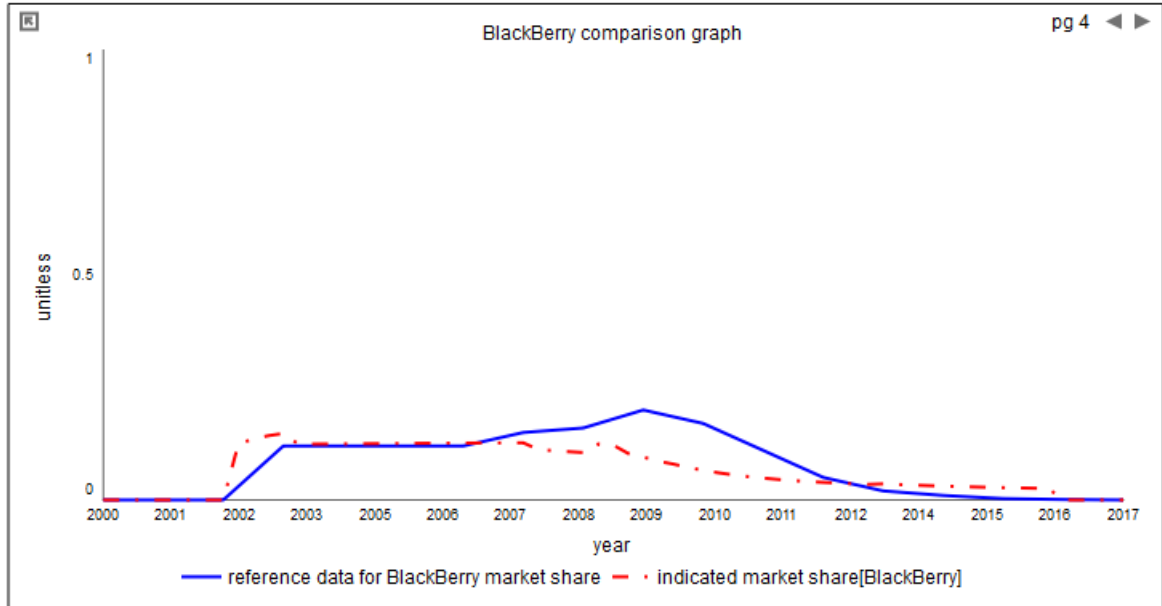


Figure 53 BlackBerry comparison graph- Data reference and simulation

According to figure 51 and 53, there is a small inconsistency between the data reference and model simulation, for market share of BlackBerry and Windows Phone during 2007 to 2011. Based on the available technological analysis in the market we believe this happened because of the technological superiority of BlackBerry and since it is not included in this model its share is not increasing in that period of time. Therefore, indicated market share of Windows Phone is higher than data reference since it absorbed the share of BlackBerry.

4.6. Behavior anomaly tests:

Behavior anomaly test examine the importance of structures by asking whether anomalous behavior arises when the relationship is deleted or modified. Anomalous behavior generated by deletion of a relationship provides you with some evidence for its importance. Loop knockout analysis is a common method to search for behavior anomalies. In loop knockout test you zero out a target relationship. For example, in decision rules of the form $\text{corrective action} = (\text{desired state} - \text{state}) / \text{adjustment time}$, you knock out the loop passing through the corrective action by setting the adjustment time to an essentially infinite value. If a loop knockout test generates bizarre or physically impossible behavior under extreme conditions you have evidence that the relationship is important and must be included [3].

To apply this test to the model, we decided to change the value of “Time to attract supporters” to almost infinite to investigate the role of supporters in the market. As it is obvious from the figure 54, applying this test leads to reasonable behavior (according to the change) in the market and iOS and BlackBerry are now the survivor of this battle (Considering the fact that if BlackBerry had enough market share they would never stop the production).

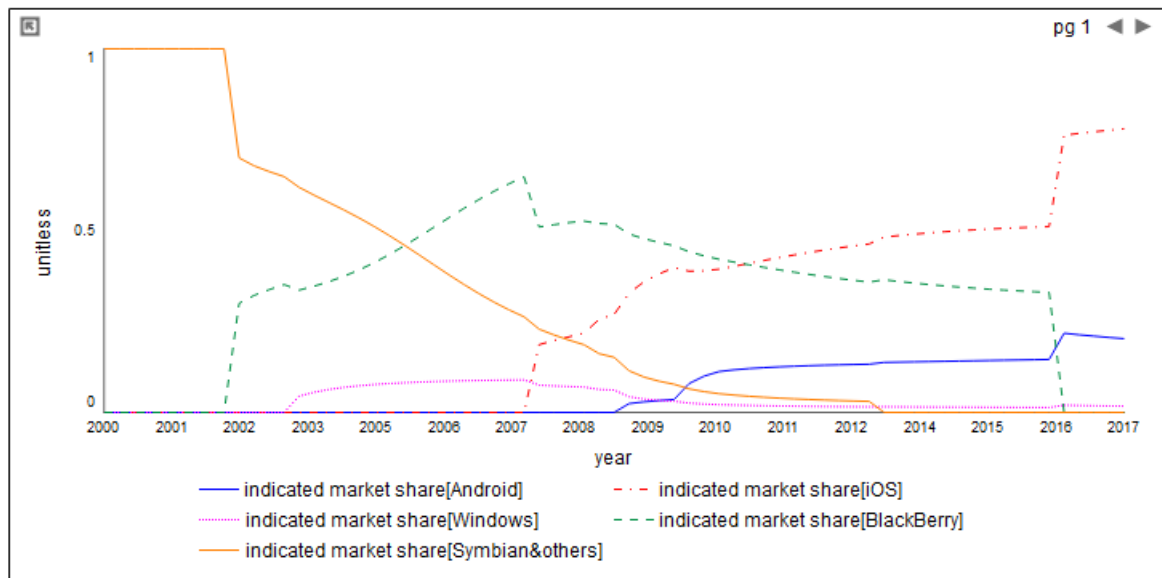


Figure 54 Behavior anomaly test

So the relation between standard support strategy and the diversity of network, model variety and complementary goods are very important to the market.

4.7. Sensitivity test:

According to Sterman [3], Since all models are wrong you must test the robustness of your conclusions to uncertainty in your assumptions. Sensitivity analysis asks whether your conclusions change in ways important to your purposes when assumptions are varied over the plausible range of uncertainty. He also mentioned that “Given the limited time and resources in any project, sensitivity analysis must focus in those relationships and parameters you suspect are both highly uncertain and likely to be influential”.

4.7.1. Adaption Fraction sensitivity test

One of the important factors that we are uncertain about in this model is the adaption fraction, which indicates what percentage of the contact between Ever-likely users and total market is successful. So this variable is our first one to check the sensitivity. Figure 55 shows some of the results of this analysis. One important outcome is that because for calculating OS attractiveness mostly we use the relative values, OS attractiveness factor and indicated market share are not sensitive to adaption fraction. This test shows how fast people are becoming potential users, and that have positive effect on the sales rate. The setting for this test in being shows in table 26.

Table 26

Sensitivity analysis-adaption fraction

	Distribution	Mean	Standard deviation	Run No.
Adaption fraction	Normal	0.01	0.01	10

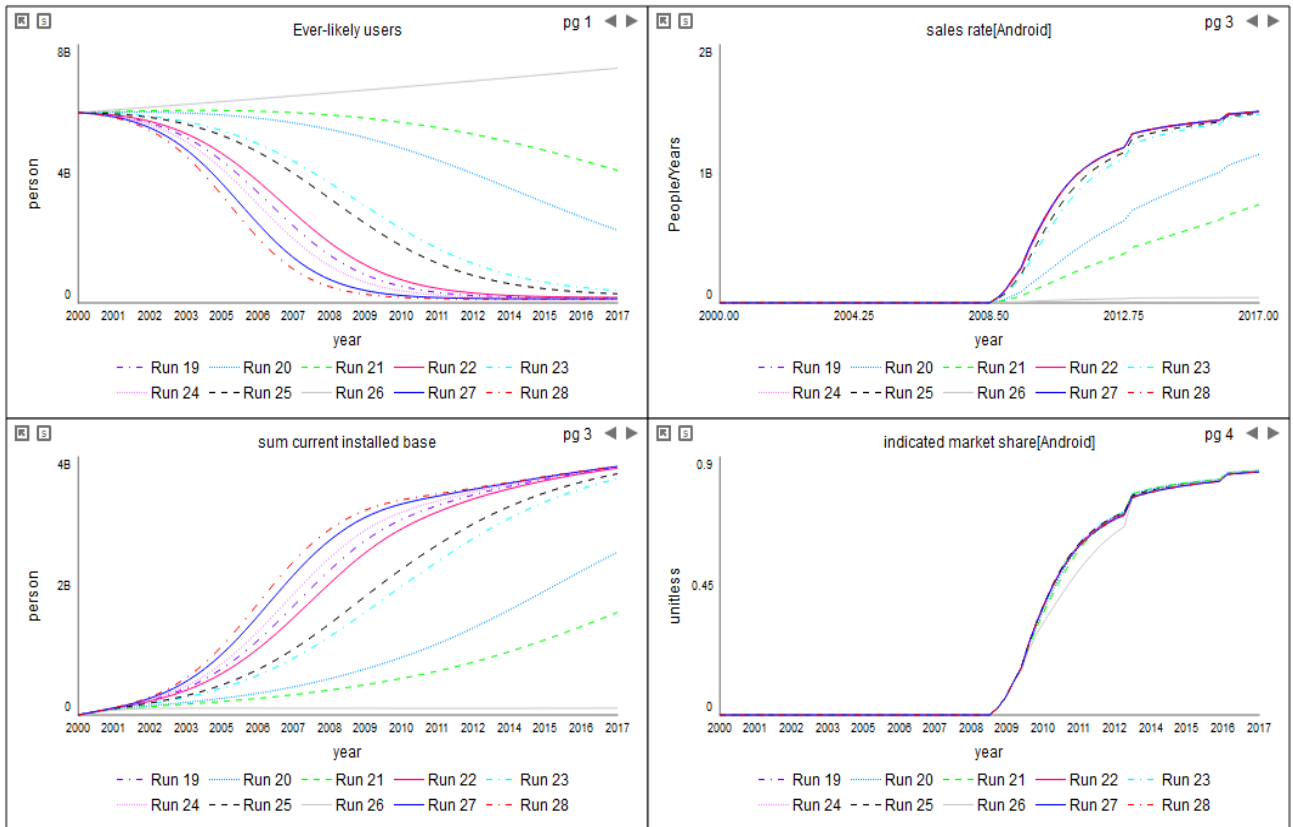


Figure 55 Sensitivity analysis-adaption fraction

4.7.2. Contact rate sensitivity test:

Another uncertain but crucial factor in the model is contact rate between ever-likely population and total market. For analyzing the sensitivity to this factor we use values in table 27. Change of this factor effects population stock to high degree which is predictable because with the contact rate being so high, so does the adaption rate and that means the ever-likely stock will deplete faster, But as figure 56 shows the indicated market share values stays almost the same.

Table 27

Sensitivity test-contact rate

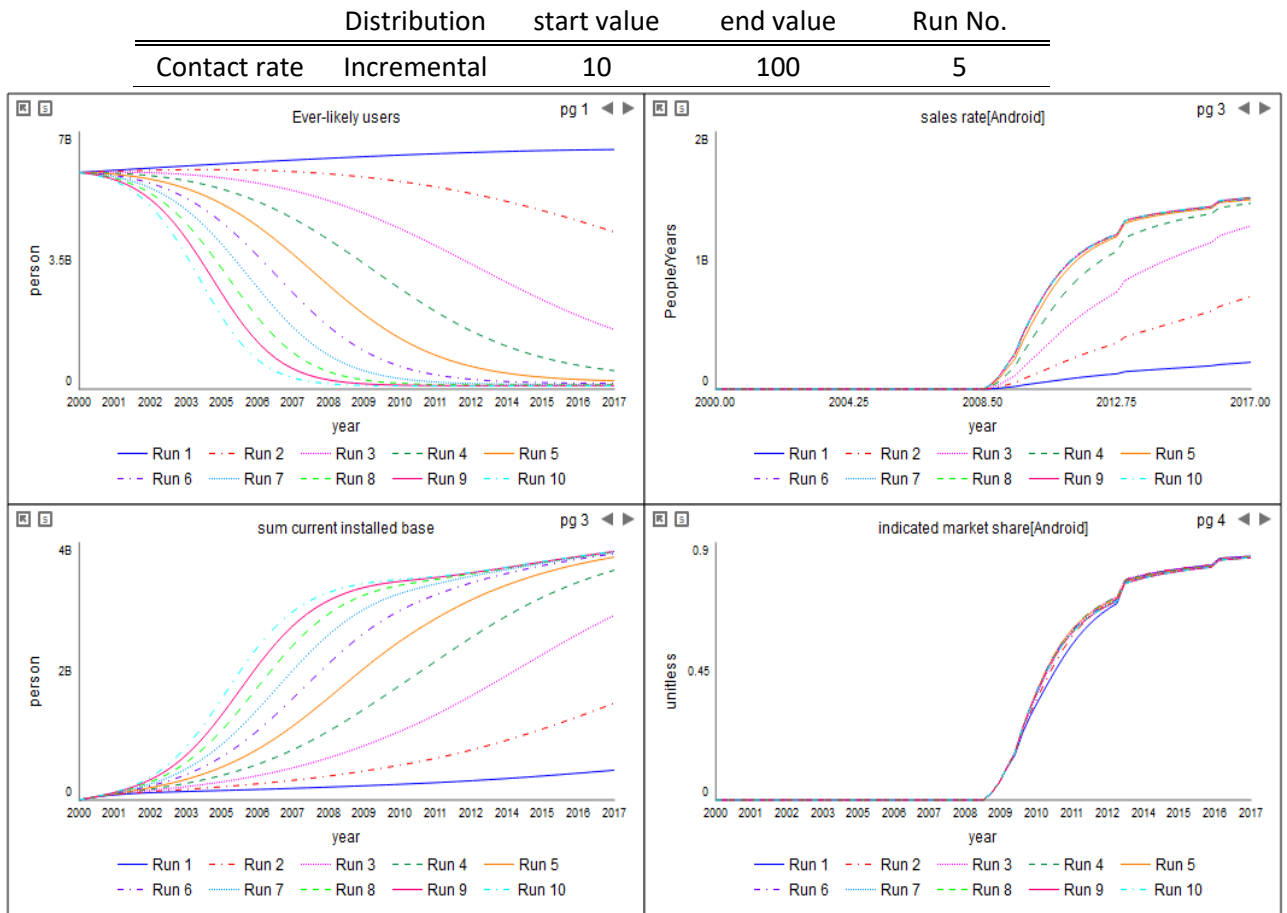


Figure 56 Sensitivity test-contact rate

4.7.3. Attribute weight sensitivity test

The other uncertain and influential parameter in this model is “Attribute weights”, As we discussed before we used the weight suggested by Geerten Van de Kaa [27] but not all the factors has direct impact on OS attractiveness and some of the factors are out of the scope of this thesis also one factor added to the list (model variety). We explained how we calculate the final weights from the table in the formulation of simulation chapter. In this part of sensitivity test we want to investigate to what extent the final results change when we imply different processes for weight allocations.

To calculate weights for Test 1, we normalized of the weights, we just consider the original weight of each factor from the study and normalized it. Test 2, we chose another processes so the weight of each category should maintain in that category and the weight of standard support strategy category (no direct effect on attractiveness) allocate to model variety is the new factor. Table 28 shows different values for attribute weights:

Table 28

Sensitivity analysis-Attribute weights

Factors	Base-line	Test 1	Test 2
Diversity of network	0.2741	0.36728	0.2321
brand reputation	0.2222	0.35334	0.3381
current installed based	0.04515	0.174831	0.0903
complementary goods	0.1662	0.10455	0.1662
Model variety	0.29235	0	0.1733
	1	1	1

Figure 57 shows the results of this test, apparently the results are very similar in the pattern for either of the weights, but for example since iOS has the highest brand reputation attractiveness when we implement tests its market share increase more than other situations.

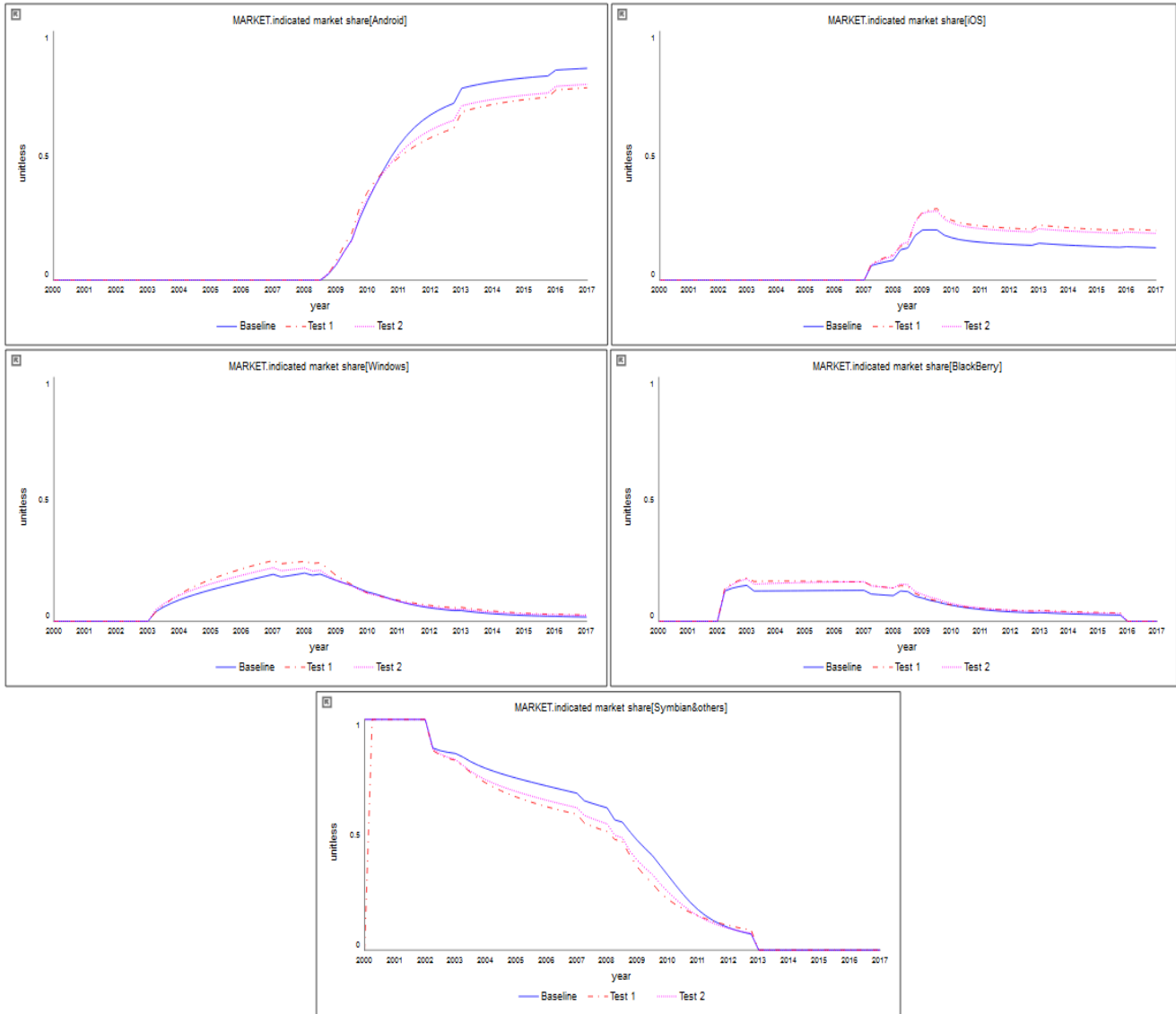


Figure 57 Sensitivity analysis-attribute weights

5. Scenario Analysis:

As we mentioned before the main purpose of this thesis is to understand why Windows Phone does not have a better position in the market, even though Microsoft as its owner has a great history of success in providing OS for personal computers. During the last chapters we tried to stay neutral and do not focus on the role of Windows Phone specifically, but in this chapter we concentrate on this OS and explore the possibility of success for Windows Phone.

We choose scenario analysis over policy analysis because, while there are lots of endogenous dynamics in the market, the OS owner's strategy has a major role in the victory and those variables depend on the owners' ideas and visions, therefore designing policies required high engagement from top managers of Microsoft. Further according to data and experts' opinion Windows Phone lost its market and there is almost no possibility of success in this market in the future.

5.1. Windows Phone's strategies and performances:

Ferida Lin and Weiguo Ye in 2009 [1] stated the fact that "Microsoft business model in smartphone market is a traditional software business. As the main source of revenue, Microsoft license Windows Mobile/Phone to any smartphone maker who is interested in putting Windows Mobile/Phone on its device. Windows Mobile/Phone model is a purer software business model compared with models of other major operating systems. It is Microsoft's ambition to copy its success in the personal computer industry to the smartphone world."

According to Ferida Lin and Weiguo Ye in 2009 [1], unlike the OS competition in the personal computer industry, the smartphone OS competition is not a battle among pure software vendors. Major OS owners all have different business models, and OS is just a part of each company's business. The business model of key players vary, but there is one point in common: device manufacturers and application developers are the only two actors providing direct resources for OS owner.

Microsoft is losing its position in this market, they tried to maintain their traditional role as a software business and charge manufacturers, meanwhile Google is proving Android free of charge for them. Moreover, Windows Phone is closed-sourced, meaning it developed by Microsoft and protected by copyright and Android is an open-source software and the code is available under the Apache license for modification and distribution for device manufacturers. In conclusion Android has the highest flexibility for manufacturers and it is free of charge.

According to Forbes [28], when Satya Nadella replaced Steve Ballmer as the CEO of Microsoft in 2014, he started his new job with the mission of making Microsoft the leading platform and productivity company for the mobile-first, cloud-first world. Jan Dawson [29] explains, to be cloud-first is to design new products and services with cloud back-ends in mind and to be mobile-first is to design them with mobile front-ends in mind.

Microsoft has a tiny share of the devices and a minority share of the platforms which run on these endpoints. The reality is Microsoft's products and services in a mobile environment will primarily be running on others' devices and platforms. That, in turn, means Microsoft's mobile-first focus will be implemented primarily in the user interfaces for individual products and services. But this is where your definition of mobile becomes really important: for certain of Microsoft's products, starting with the smartphone makes perfect sense: Skype, for instance. But for others – Office being an obvious example – focusing on a device where most of the use cases will likely never be employed doesn't make sense. Tablets and PCs should be the focus for Office, with a subset of the features available there available on smartphones [29].

To describe this new strategy of Microsoft based on the model, we can say Microsoft is no longer focus on its "sales rate" and "indicated market share" in the smartphone market. They are now interested in increasing their revenue through the installed base either Windows phone or others. It is very interesting and practical to focus on the stock of users rather than selling operating systems to phone manufacturers. Therefore we implement scenarios based on this strategy to analyze the results in different circumstances.

It is important to keep in mind the focus of this thesis is not on Microsoft revenue and the whole enterprise, but it is on the smartphone market.

5.2. Scenarios:

In this section we are going to test the results of different scenarios to find out how Windows Phone could survive in the market. As we mentioned before and based on Microsoft new strategy, the focus of the Microsoft is now on the stock of installed base. Microsoft wants to provide software for all available operating systems in the market. Considering the fact that Windows phone users have higher priority in this strategy, therefore one of the main goals in the scenario analysis is to increase Windows Phone installed base.

For implementing Scenario analysis we focused on two major strategies companies made for the OS. Appropriability strategy which shows whether or not OS is available for device manufacturers to use it and either it is free of charge or not. OS flexibility for companies that shows if the OS code is available for device manufacturers to do customizations or if OS in closed-source.

Table 29 shows the setting of 5 scenarios in the following sections, the time frame of these scenarios is 2000 to 2030.

Table 29

Scenarios setting

	Time to start	Appropriability strategy	OS flexibility
Scenario 1	2003	1	-
Scenario 2	2003	1	1
Scenario 3	2003	1	-
Scenario 4	2012	0.5	-
Scenario 5	2009	1	-
	2017	1	-

5.2.1. Scenario 1:

In this scenario the focus is not on the revenue through selling the license of windows phone, but it is on increasing market share and more importantly current installed base. This scenario shows the possibility of implementing the Mobile-First, Cloud-First plan from 2003, and focusing on the stock of Windows Phone installed base rather than just sales rate.

As figure 58 shows implementing this scenarios affect the indicated market share and it increase to 50% in 2010. This is the result of free operating system, now Windows Phone is more attractive than before for device manufacturers.

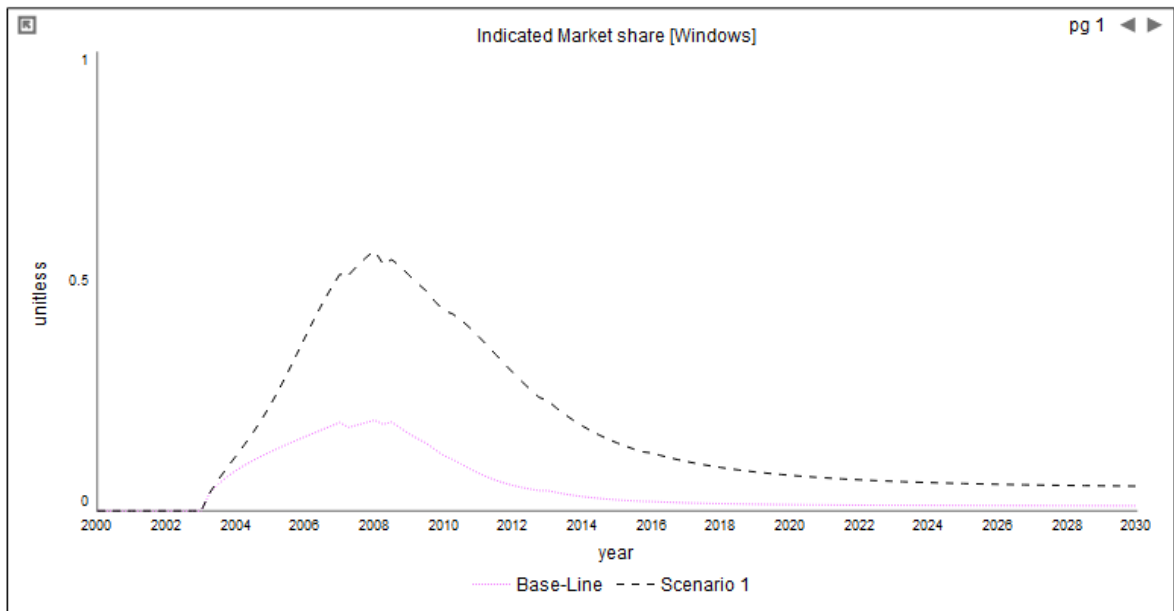


Figure 58 Scenario 1- Indicated market share

This does not take long and Windows Phone lose the market after Android coming to the market but in this scenario Windows Phone stays longer in a good position. The reason Android overcomes Windows Phone is that Android is open-source and their current ecosystem on complementary goods is better that Windows Phone. Maybe if Windows Phone change its other strategies the results would be better.

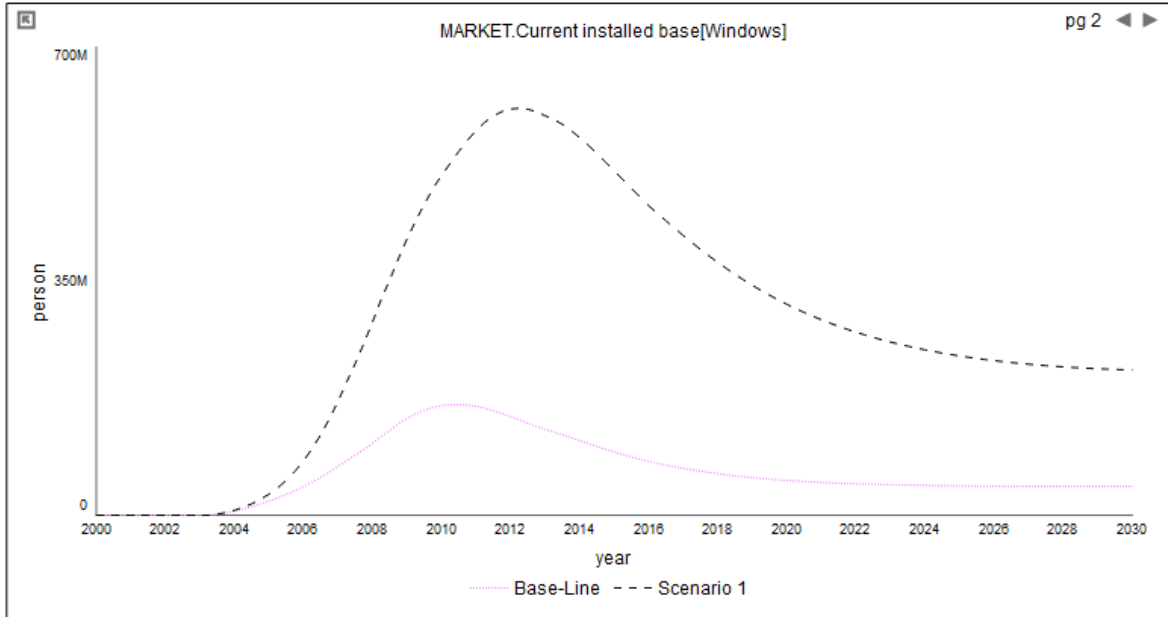


Figure 59 Scenario 2- Installed base

This scenario is very similar to the Google’s strategy for Android, Google’s plan is not making any profit directly through selling Android but by increasing the current installed base and making money from users constantly. While the stock of Windows Phone installed base eventually decreases, with this scenario the level of this stock is much higher than current situation and this means more revenue.

5.2.2. Scenario 2:

As we mentioned in the scenario 1, even when Window Phone is available free of charge for device manufacturers, after Android entering the market they lose both market share and current installed base. This decrease happens because Android is more attractive to device manufacturers since they can have their own customization on this operating system.

It is highly unlikely for Microsoft to provide Windows Phone’s codes available for device manufacturers and let them do all the customization. But in this scenario we assume that this is possible in order to see the results, so in scenario 2 Windows Phone is free of charge and open-source from 2003 also since its open-source this will increase the flexibility of OS for developers as well.

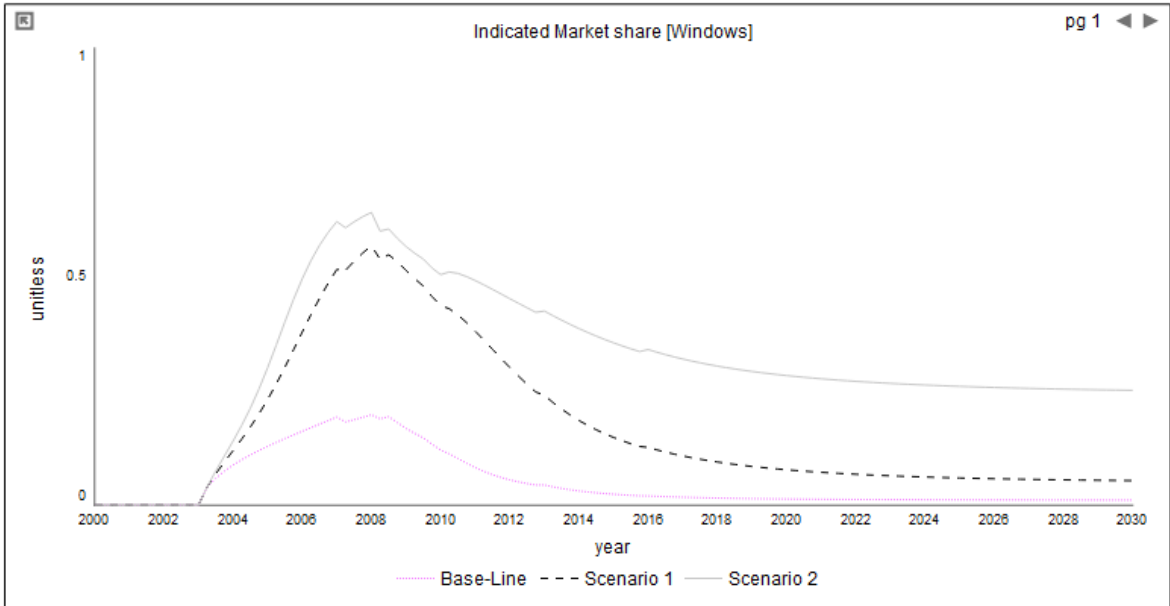


Figure 60 Scenario 2- Indicated market share

Figure 60 represents the result of this scenario and comparing it with scenario 1 and base-line. This scenario increases the indicated market share of Windows Phone, but this is still not enough for Windows Phone to reach the steady state in this market. One of the reasons can be the fact that Windows Phone does not have a good ecosystem for its complementary goods currently. Maybe if Microsoft had better position in this market they would worked on that area as well.

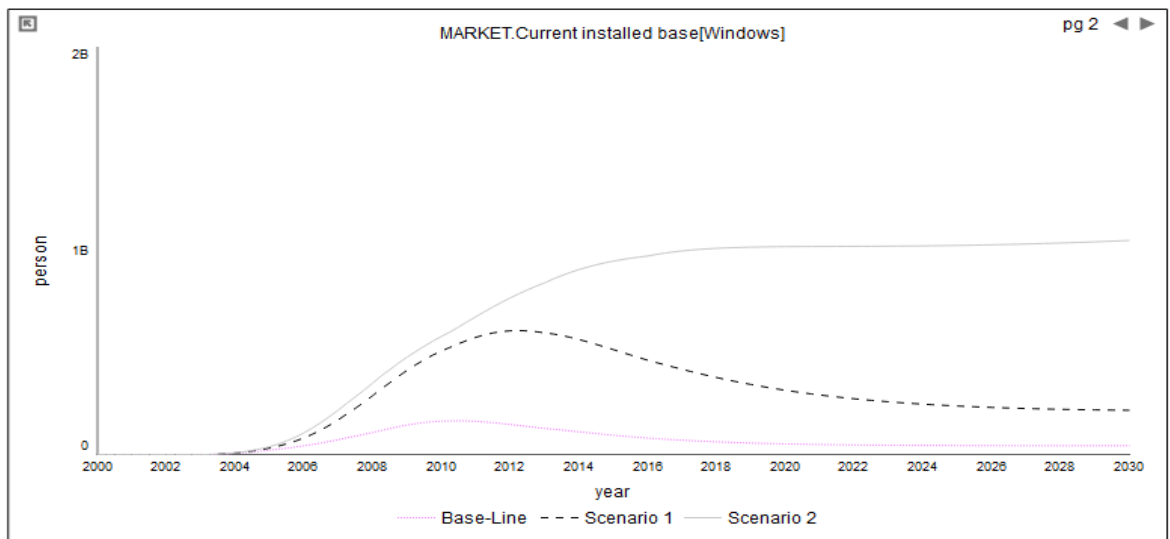


Figure 61 Scenario 2- Current installed base

Being open-source will increase the vulnerability and decrease the security of Windows Phone and the control of Microsoft, but we just want to show the results not recommending this scenario. By implementing this scenario, Windows Phone will end up with a high number of current installed base in future and this help them to successes in their Mobile-first plan since they have a larger installed base that are directly using Microsoft products.

5.2.3. Scenario 3:

In scenario 3, we investigate the results when Windows Phone is free of charge for manufacturers from 2003 to 2012 in order to build their own market. But from 2012 Microsoft start to charge for the license, to increase their revenue even more.

As one can interpret from figure 62, in this scenario the indicated market share decreases after the charging starts. Even the stock of current installed base and the bandwagon effect cannot prevent this from happening. As we explained before in the format battle the “diversity of network” is very important, not only this effect the OS attractiveness directly but also the stock of supporters positively affect the smartphone model variety and complementary goods.

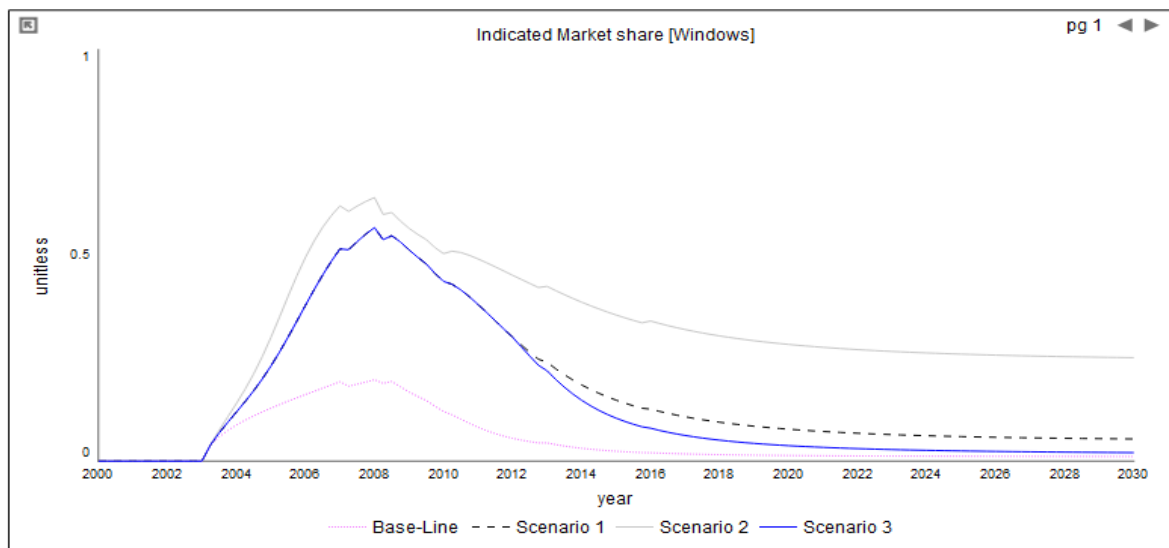


Figure 62 Scenario 3-Indicated market share

5.2.4. Scenario 4:

This is almost opposite of Scenario 3, in this scenario Microsoft implement the original plan (Baseline), until Android entered the market and start to attract more supporters and users. From 2009 Microsoft changes its appropriability strategy and provide windows phone free of charge for device manufacturers.

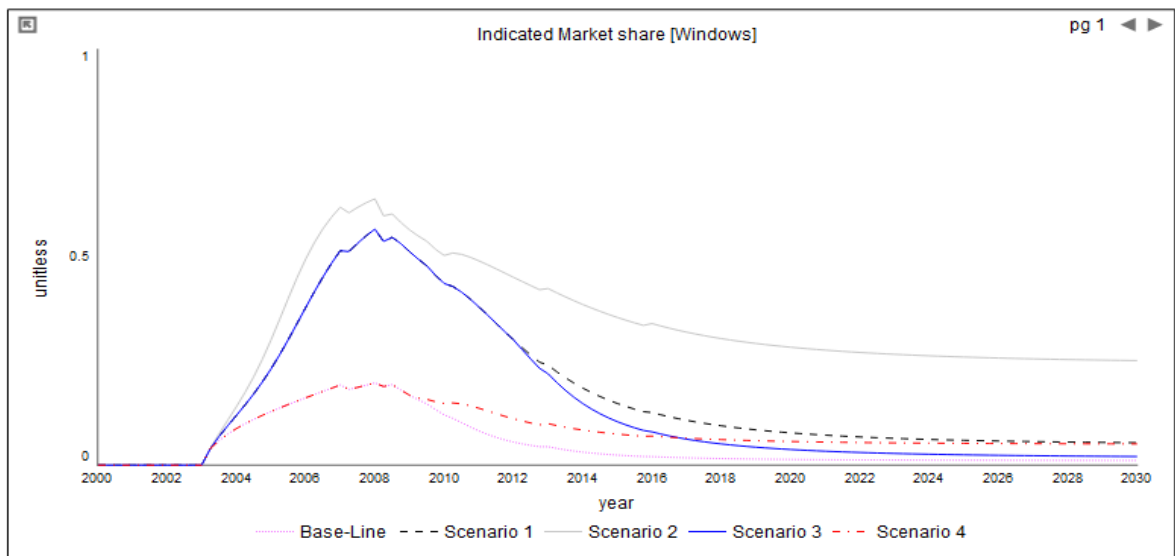


Figure 63 Scenario 4-indicated market share

As figure 63 shows, even though Microsoft respond to the market late and did not change the flexibility of the Windows Phone for device manufacturers this scenario works better than base-line in case of increasing market share. Also the stock of current installed base in this scenario is bigger than base-line scenario and scenario 3 (Figure 64).

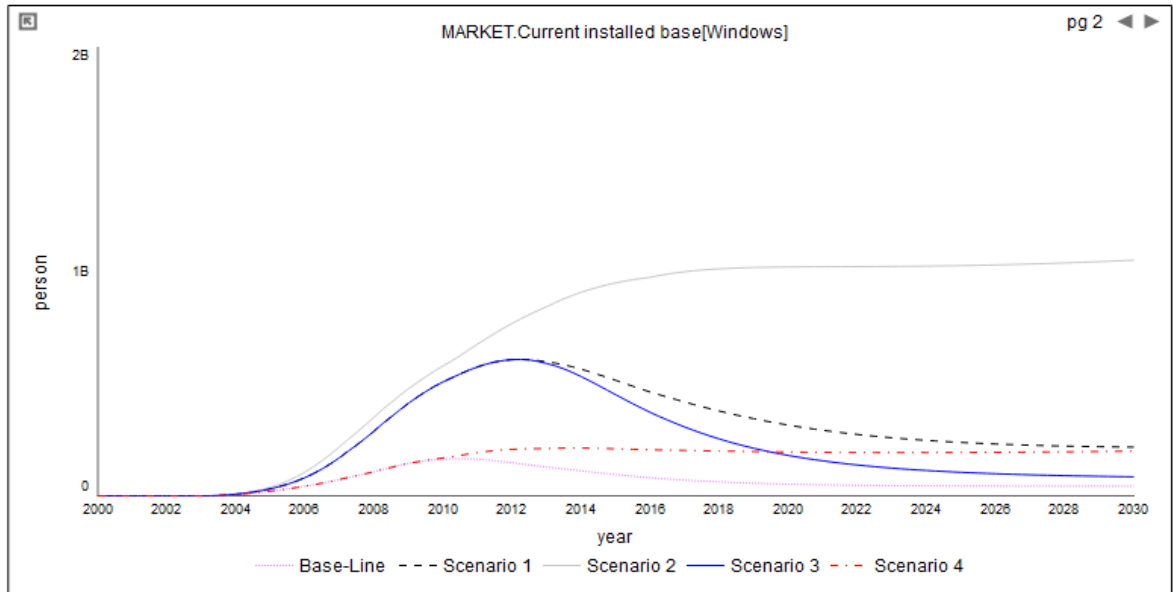


Figure 64 Scenario 4- Current Installed base

5.2.5. Scenario 5:

Scenarios 1 to 4, are happening in the past and shows how Microsoft could react in order to secure a better position in the market. In the last Scenario we intend to examine the effects of the change in appropriability in 2017 (now).

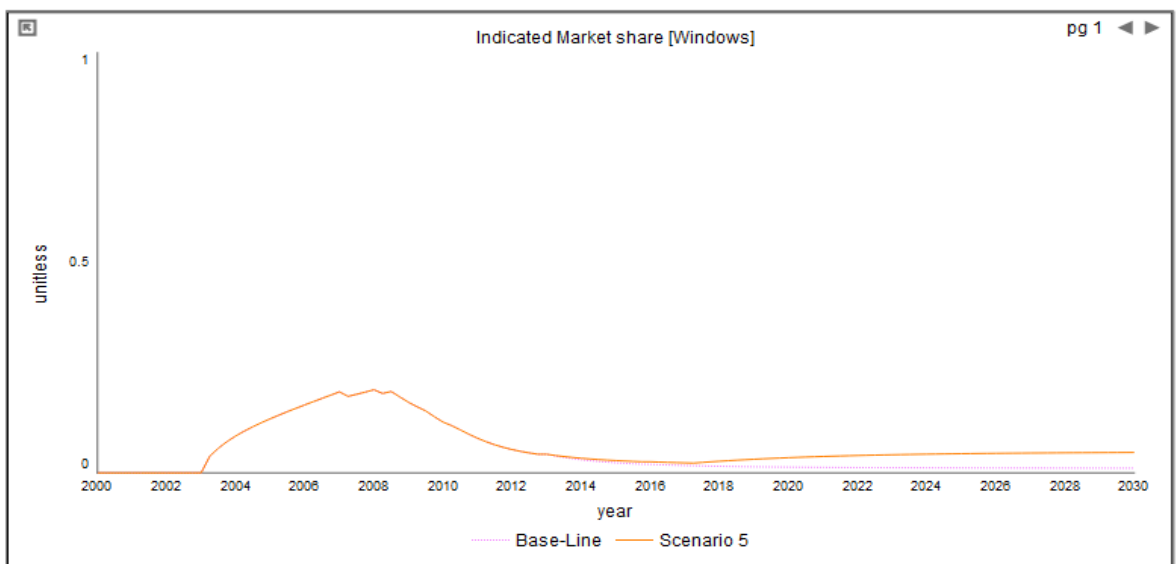


Figure 65 Scenario 5-Indicated market share

Figure 65 shows the simulation for indicated market share, market share increases almost around 5 times, this shows that even without any technical change or making Windows Phone an open-source, Microsoft has a chance to boost its market share in future with making this decision. The current installed base stock increases as well and this affect the success of Microsoft in implementing their Mobile-first, Cloud-first plan (Figure 66).

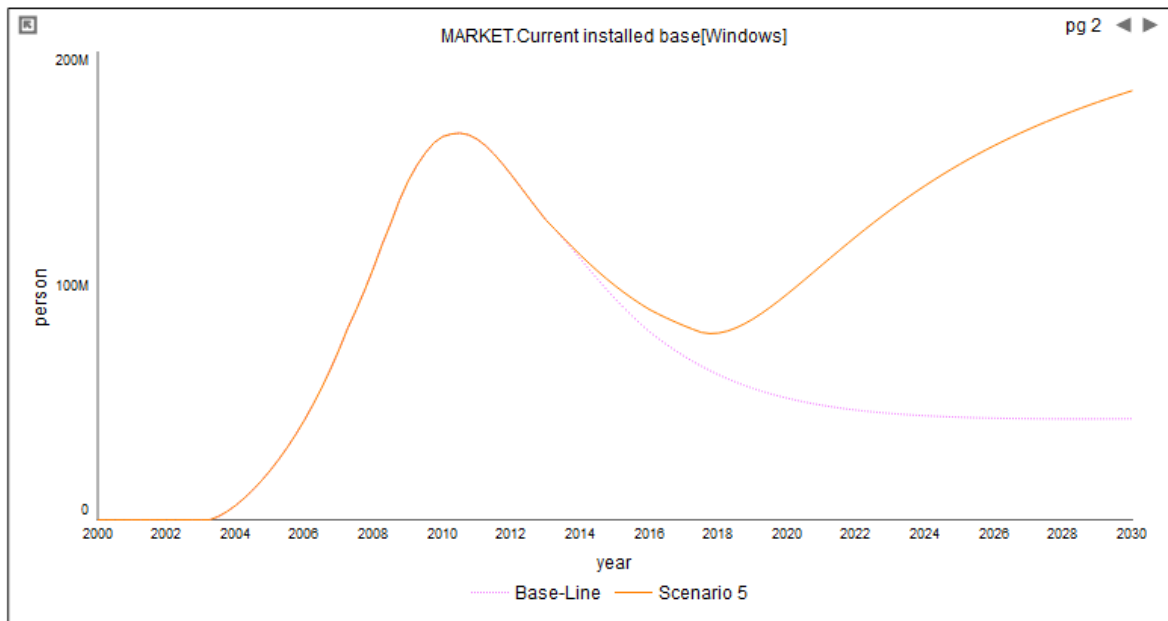


Figure 66 Scenario 5- Current installed base

6. Conclusions and recommendations:

This thesis's main concern was to analyze reasons behind Windows Phone's failure. In order to conduct this analysis we needed to cover two main areas, first to make a comprehensive model of the smartphone market including five other operating systems. Second to analyze the dynamics behind the market to distinguish main factors that led to Windows Phone low market share.

We build a model for the smartphone market and included Windows Phone along with other 5 important competitors, android, iOS, Symbian and BlackBerry. In the processes of making the model we used Geerten Van de Kaa's [5, 18] proposed model and framework. Four categories of characteristics are included in this model including, characteristics of standard supports, other stakeholders, characteristics of standard and standard support strategies.

Most of factors are dynamic and arise within the market, but there are two important variables that directly depends on the OS owner's strategy. These factors are appropriability strategy which demonstrate to what extent the OS is available for manufacturers to use and what the license fee is. Second is OS flexibility for manufacturers, this variable indicates the possibility of customization by manufacturers on the operating system. These two factors have significant role in attracting supporters to adapt the specific OS.

The role of OS supporters (or standard supporters) is the most important factor in winning this competition. Because this supporters are going to manufacture mobile models and complementary goods, so in order to reach the leader position in the market the OS needs supporters. While Android has been very successful to attract supporters to joining its network, Windows Phone was not successful in this mission. The reason is that Microsoft still charge phone manufacturers with the license fee, which is surprisingly not very low, also Windows Phone is a closed-source operating system which means manufacturers are not able to do any customization on it.

Microsoft has a great reputation in providing OS for personal computers, but they failed in smartphone market and they lost the market to Android. To an open-source software which is available free of charge for supporters, even though some may argue that Android had lots of technical issues in the beginning.

In the last chapter of this thesis we implemented five different scenarios in order to find out what combination of strategies would help Windows Phone to end up in better position. Unfortunately for Windows Phone the main competitor in this market has a great and clear strategy and the owner does not want to make any direct revenue through selling Android. The decision to make Windows Phone freely available can be justified by the fact that Microsoft can make more money through current installed base and making money through applications such as skype and Office package. But the decision to make a software open-source is huge and strategic and it will endanger the security and reliability of the OS and most probably it is not welcome by Microsoft.

The first scenario show the results when Windows Phone has been free of charge from its entry to the market. The result shows that even in this condition, Windows Phone would lost the majority of market share to Android, because Android open-source strategy is very attractive to supporters and sooner or later its supporters beats Windows Phone.

Second scenario is an unlikely scenario, because in this scenario Windows Phone is also an open-source operating system. In this scenario Windows Phone will end up with the better position in the market, but still Android is the market leader due to its better ecosystem strategy. This is the area Windows Phone would have worked if they had a chance of becoming the leader in the first place.

Third scenario represents the situation where Windows Phone is free of charge from 2003 to 2012 in order to build the market but after 2012 manufacturers have to pay the license fee. This scenario will not end in a good way either, because when charging starts manufacturers have Android as an alternative operating system and they will switch to it.

In the fourth scenario we investigate what would have happened when Microsoft decided to provide Windows Phone free of charge for manufacturers after Android entered the market in 2009. In this scenario Windows Phone will end up in slightly better situation than current situation, that's because Android open-source strategy still beats Windows Phone.

The last scenario analyze the results when Microsoft changes its appropriability strategy in 2017, and does not charge manufacturers with license fee. While in this scenario the indicated market share of Windows Phone will increase almost 5 times, the final result is not very impressive.

According to the model and scenario analysis, Microsoft cannot win the market or improve its position just by changing its appropriability strategy. Microsoft needs an extensive strategy and policy analysis to cover all four categories, appropriability strategy attracts more supporters but it is not enough to win. Microsoft requires to focus on the availability of applications for Windows Phone, currently this stock is very low in comparison with others. Further Microsoft has to build a great ecosystem for its users in order to increase their loyalty to Windows Phone, this can be achieved by high number of complementary goods and a better ecosystem strategy. Applying these policies in the model demands high contributions from Microsoft Company and all engaging managers, so this is one of the areas that one can follow in the future.

As John Sterman stated in the book Business Dynamics [3], no model is perfect and this model is not far from that definition and the model itself has a potential to be expanded. We have few recommendation for further studies, one can investigate the role of technological superiority in order to find out its importance in this battle. Moreover supports are important from another perspective in this market as well and that is their role in marketing and advertising both in technology diffusion and OS attractiveness.

Another recommendation for future study is to expand this model to cover all five phases in Suarez's framework [14]. These phases are, R&D build up, technical feasibility,

creating the market, the decisive battle and post-dominance phases. By applying this framework one can predict variables such as “entry time” and “leave time” endogenously in the model which is very challenging and interesting.

References:

1. Lin, F. and W. Ye, *Operating system battle in the ecosystem of smartphone industry*. 2009.
2. Cecere, G., N. Corrocher, and R.D. Battaglia, *Innovation and competition in the smartphone industry: Is there a dominant design?* Telecommunications Policy, 2015. **39**(3-4): p. 162-175.
3. Sterman, J.D., *Business dynamics : systems thinking and modeling for a complex world*. 2000, Boston: Irwin McGraw-Hill.
4. van de Kaa, G., et al., *Factors for winning interface format battles: A review and synthesis of the literature*. Technological Forecasting & Social Change, 2011. **78**(8): p. 1397-1411.
5. van de Kaa, G., et al., *Supporting Decision Making in Technology Standards Battles Based on a Fuzzy Analytic Hierarchy Process*. Engineering Management, IEEE Transactions on, 2014. **61**(2): p. 336-348.
6. Schumpeter, J.A., *The Theory of Economic Development: An Inquiry into Profits, Capital, Credit, Interest, and the Business Cycle*. 1934.
7. David, P.A., *Clio and the Economics of QWERTY*. The American Economic Review, 1985. **75**(2): p. 332-337.
8. Lee, J., J. Lee, and H. Lee, *Exploration and Exploitation in the Presence of Network Externalities*. Management Science, 2003. **49**(4): p. 553-570.
9. Teece, D., *Profiting from technological innovation: Implications for integration, collaboration, licensing and public policy*. 1993. p. 112-113.
10. Gomes-Casseres, B., *Group versus group: how alliance networks compete*. 1994.
11. Foray, D., *Users, standards and the economics of coalitions and committees*. Information Economics and Policy, 1994. **6**(3): p. 269-293.
12. Agarwal, R., et al., *Knowledge transfer through inheritance: Spinout generation, development, and survival*, in *Acad. Manage. J.* 2004. p. 501-522.
13. Axelrod, R., et al., *Coalition Formation in Standard-Setting Alliances*. Management Science, 1995. **41**(9): p. 1493-1508.
14. Suarez, F.F., *Battles for technological dominance: an integrative framework*. Research Policy, 2004. **33**(2): p. 271-286.
15. Klepper, S., *Entry, Exit, Growth, and Innovation over the Product Life Cycle*. The American Economic Review, 1996. **86**(3): p. 562-583.
16. Clayton M. Christensen, F.F.S., James M. Utterback, *Strategies for Survival in Fast-Changing Industries*. Management Science, 1998. **44**(12-part-2): p. S207-S220.
17. Shapiro, C. and H.R. Varian, *Information Rules, a Strategic Guide to the Network economy*. 1999.
18. van de Kaa, G., H. de Vries, and J. van Den Ende, *Strategies in network industries: the importance of inter-organisational networks, complementary goods, and commitment*. Technology Analysis & Strategic Management, 2015. **27**(1): p. 73.
19. Statista.com. *Forecast of mobile phone users worldwide*. 2012; Available from: <https://www.statista.com/statistics/274774/forecast-of-mobile-phone-users-worldwide/>.
20. Papachristos, G. and G. Van de kaa, *Factors for winning Format Battles modeling approach*. 2014.
21. Roser, M. and E. Ortiz-Ospina. *World population growth*. 2015; Available from: <https://ourworldindata.org/world-population-growth/>.
22. Kline, D.B. *How often does the average american replace his or her smartphone*. 2015; Available from: <https://www.fool.com/investing/general/2015/07/15/how-often-does-the-average-american-replace-his-or.aspx>.

23. Garud, R., S. Jain, and A. Kumaraswamy, *Institutional entrepreneurship in the sponsorship of common technological standards: The case of Sun Microsystems and Java*, in *Acad. Manage. J.* 2002. p. 196-214.
24. *Differences between android and windows phone.*
25. Sinicki, A. *Developing for Android and iOS*. 2016; Available from: <http://www.androidauthority.com/developing-for-android-vs-ios-697304/>.
26. Peter A. Otto, J.R.B., *Brand management facilitation: A system dynamics approach for decision makers*. 2001.
27. Smit, F.C. and C.W.I. Pistorius, *Implications of the Dominant Design in Electronic Initiation Systems in the South African Mining Industry*. *Technological Forecasting & Social Change*, 1998. **59**(3): p. 255-274.
28. MSV, J. *Microsoft is marching ahead in its cloud-first, mobile-first journey*. 2016; Available from: <https://www.forbes.com/sites/janakirammsv/2016/04/02/microsoft-is-marching-ahead-in-its-cloud-first-mobile-first-journey/#2758ea113f6c>.
29. Dawson, J. *What does cloud-first Mobile-first mean for Microsoft*. 2014; Available from: <https://techpinions.com/what-does-cloud-first-mobile-first-mean-for-microsoft/33496>.

Equations:

Market Module:

Current_installed_base[Android](t) = Current_installed_base[Android](t - dt) + (sales_rate[Android] - Chng_rate[Android]) * dt

INIT Current_installed_base[Android] = 0

Current_installed_base[iOS](t) = Current_installed_base[iOS](t - dt) + (sales_rate[iOS] - Chng_rate[iOS]) * dt

INIT Current_installed_base[iOS] = 0

Current_installed_base[Windows](t) = Current_installed_base[Windows](t - dt) + (sales_rate[Windows] - Chng_rate[Windows]) * dt

INIT Current_installed_base[Windows] = 0

Current_installed_base[BlackBerry](t) = Current_installed_base[BlackBerry](t - dt) + (sales_rate[BlackBerry] - Chng_rate[BlackBerry]) * dt

INIT Current_installed_base[BlackBerry] = 0

Current_installed_base[Symbian&others](t) = Current_installed_base[Symbian&others](t - dt) + (sales_rate[Symbian&others] - Chng_rate[Symbian&others]) * dt

INIT Current_installed_base[Symbian&others] = 0

INFLOWS:

sales_rate[OS] = (potential_installed_base*indicated_market_share/time_to_sale)

OUTFLOWS:

Chng_rate[OS] = (Current_installed_base/time_to_change)*other_OS_attractiveness

"ever-likely_users"(t) = "ever-likely_users"(t - dt) + (net_population_chng_Rate - adaption_Rate) * dt

INIT "ever-likely_users" = 5900000000

INFLOWS:

net_population_chng_Rate = Total_population*chng_fraction_in_population

OUTFLOWS:

adaption_Rate = (contact_rate*adaption_fraction*"ever-likely_users"*total_market)/Total_population

potential_installed_base(t) = potential_installed_base(t - dt) + (adaption_Rate + Chng_rate[Android] + Chng_rate[iOS] + Chng_rate[Windows] + Chng_rate[BlackBerry] + Chng_rate[Symbian&others] - sales_rate[Android] - sales_rate[iOS] - sales_rate[Windows] - sales_rate[BlackBerry] - sales_rate[Symbian&others]) * dt

INIT potential_installed_base = 200000000

INFLOWS:

adaption_Rate = (contact_rate*adaption_fraction*"ever-likely_users"*total_market)/Total_population

Chng_rate[Android] = (Current_installed_base/time_to_change)*other_OS_attractiveness

Chng_rate[iOS] = (Current_installed_base/time_to_change)*other_OS_attractiveness

Chng_rate[Windows] = (Current_installed_base/time_to_change)*other_OS_attractiveness

Chng_rate[BlackBerry] =

(Current_installed_base/time_to_change)*other_OS_attractiveness

Chng_rate[Symbian&others] =

(Current_installed_base/time_to_change)*other_OS_attractiveness

OUTFLOWS:

sales_rate[Android] = (potential_installed_base*indicated_market_share/time_to_sale)

```

sales_rate[iOS] = (potential_installed_base*indicated_market_share/time_to_sale)
sales_rate[Windows] = (potential_installed_base*indicated_market_share/time_to_sale)
sales_rate[BlackBerry] = (potential_installed_base*indicated_market_share/time_to_sale)
sales_rate[Symbian&others] =
(potential_installed_base*indicated_market_share/time_to_sale)
adaption_fraction = Normal_adaption_fraction*(1-
switch_adaption_fraction)+Extreme_adaption_fraction*(switch_adaption_fraction)
chng_fraction_in_population = 0.012
contact_rate = 30
Extreme_adaption_fraction = 0.2
Extreme_test_time_to_change = 0.5
indicated_market_share[OS] = SAFEDIV(Operating_system.OS_attractiveness_for_users,
SUM(Operating_system.OS_attractiveness_for_users), 0)
initial_sales_rate = INIT(percieved_sales)
Market_Share[OS] = SAFEDIV(Current_installed_base*100,
SUM(Current_installed_base), 0)
Normal_adaption_fraction = 0.01
Normal_time_to_change = 2
OS_initial_sales_rate[OS] = INIT(sales_rate)
OS_previous_sales_rate[OS] = PREVIOUS(sales_rate, OS_initial_sales_rate)
OS_relative_sales_rate[OS] = SAFEDIV(sales_rate, OS_previous_sales_rate, 0)
other_OS_attractiveness[OS] = 1-Operating_system.OS_attractiveness_for_users
percieved_sales = SMTH1(Total_sales_rate, sale_smooth_time)
previous_sales_rate = PREVIOUS(percieved_sales, initial_sales_rate)
reference_data_for_Android_market_share = GRAPH(TIME)
(2000.00, 0.000), (2001.00, 0.000), (2002.00, 0.000), (2003.00, 0.000), (2004.00, 0.000),
(2005.00, 0.000), (2006.00, 0.000), (2007.00, 0.000), (2008.00, 0.000), (2009.00, 0.030),
(2010.00, 0.210), (2011.00, 0.460), (2012.00, 0.660), (2013.00, 0.780), (2014.00, 0.810),
(2015.00, 0.820), (2016.00, 0.850), (2017.00, 0.850)
reference_data_for_BlackBerry_market_share = GRAPH(TIME)
(2000.00, 0.000), (2001.00, 0.000), (2002.00, 0.000), (2003.00, 0.120), (2004.00, 0.120),
(2005.00, 0.120), (2006.00, 0.120), (2007.00, 0.150), (2008.00, 0.160), (2009.00, 0.200),
(2010.00, 0.170), (2011.00, 0.110), (2012.00, 0.050), (2013.00, 0.020), (2014.00, 0.010),
(2015.00, 0.003), (2016.00, 0.001), (2017.00, 0.000)
reference_data_for_iOS_market_share = GRAPH(TIME)
(2000.00, 0.000), (2001.00, 0.000), (2002.00, 0.000), (2003.00, 0.000), (2004.00, 0.000),
(2005.00, 0.000), (2006.00, 0.000), (2007.00, 0.000), (2008.00, 0.110), (2009.00, 0.140),
(2010.00, 0.150), (2011.00, 0.180), (2012.00, 0.190), (2013.00, 0.160), (2014.00, 0.150),
(2015.00, 0.160), (2016.00, 0.140), (2017.00, 0.140)
reference_data_for_Symbian&_Others_market_share = GRAPH(TIME)
(2000.00, 1.000), (2001.00, 1.000), (2002.00, 1.000), (2003.00, 0.880), (2004.00, 0.730),
(2005.00, 0.730), (2006.00, 0.730), (2007.00, 0.670), (2008.00, 0.590), (2009.00, 0.540),
(2010.00, 0.420), (2011.00, 0.230), (2012.00, 0.070), (2013.00, 0.001), (2014.00, 0.000),
(2015.00, 0.000), (2016.00, 0.000), (2017.00, 0.000)
reference_data_for_Windows_market_share = GRAPH(TIME)
(2000.00, 0.000), (2001.00, 0.000), (2002.00, 0.000), (2003.00, 0.000), (2004.00, 0.150),
(2005.00, 0.150), (2006.00, 0.150), (2007.00, 0.180), (2008.00, 0.140), (2009.00, 0.090),

```

```

(2010.00, 0.040), (2011.00, 0.020), (2012.00, 0.020), (2013.00, 0.030), (2014.00, 0.030),
(2015.00, 0.020), (2016.00, 0.010), (2017.00, 0.010)
relative_total_sale_rate = SAFEDIV(percieved_sales, previous_sales_rate, 0)
sale_smooth_time = 0.5
size_of_network = GRAPH(TIME)
(2000.00, 1.000), (2001.00, 1.000), (2002.00, 2.000), (2003.00, 3.000), (2004.00, 3.000),
(2005.00, 3.000), (2006.00, 3.000), (2007.00, 4.000), (2008.00, 5.000), (2009.00, 5.000),
(2010.00, 5.000), (2011.00, 5.000), (2012.00, 5.000), (2013.00, 5.000), (2014.00, 4.000),
(2015.00, 4.000), (2016.00, 4.000)
sum_current_installed_base = SUM(Current_installed_base)
switch_adaption_fraction = 0
Switch_Time_to_change = 0
time_to_change = Normal_time_to_change*(1-
Switch_Time_to_change)+Extreme_test_time_to_change*Switch_Time_to_change
time_to_sale = 2
total_awareness = total_market/Total_population
total_market = sum_current_installed_base+potential_installed_base
Total_population = "ever-likely_users"+total_market
Total_sales_rate = SUM(sales_rate)
{ The model has 41 (81) variables (array expansion in parens).
In this module and 0 additional modules with 2 sectors.
Stocks: 3 (7) Flows: 4 (12) Converters: 34 (62)
Constants: 10 (10) Equations: 28 (64) Graphicals: 6 (6)
There are also 30 expanded macro variables.
}

```

Operating system Module:

```

Operating_system.Attributes_weight[Complementary] = 0.1662
Operating_system.Attributes_weight[Network] = 0.2741
Operating_system.Attributes_weight[bandwagon] = 0.04515
Operating_system.Attributes_weight[brand] = 0.2222
Operating_system.Attributes_weight[Model] = 0.29235
Operating_system.Attributes_weight_composite[attributes, OS] = IF TIME >=
time_of_leave[OS] THEN 0 ELSE IF TIME >=time_of_entry[OS] THEN
OS_quality_by_attributes*Attributes_weight[attributes] ELSE 0
Operating_system.brand_loyalty_attractiveness_factor[OS] = SAFEDIV(Brand_loyalty,
sum_of_brand_loyalty, 0)
Operating_system.market_goal[Android] = 1
Operating_system.market_goal[iOS] = 0.8
Operating_system.market_goal[Windows] = 0.6
Operating_system.market_goal[BlackBerry] = 0.6
Operating_system.market_goal[Symbian&others] = 1
Operating_system.OS_attractiveness_for_users[OS] =
SUM(Attributes_weight_composite[* ,OS])/SIZE(Attributes_weight_composite[* ,OS])
Operating_system.OS_quality_by_attributes[Complementary, Android] =
complementary_attractiveness_factor[Android]*market_goal[Android]

```

Operating_system.OS_quality_by_attributes[Complementary, iOS] =
 complementary_attractiveness_factor[iOS]*market_goal[iOS]
 Operating_system.OS_quality_by_attributes[Complementary, Windows] =
 complementary_attractiveness_factor[Windows]*market_goal[Windows]
 Operating_system.OS_quality_by_attributes[Complementary, BlackBerry] =
 complementary_attractiveness_factor[BlackBerry]*market_goal[BlackBerry]
 Operating_system.OS_quality_by_attributes[Complementary, Symbian&others] =
 complementary_attractiveness_factor[Symbian&others]*market_goal[Symbian&others]
 Operating_system.OS_quality_by_attributes[Network, Android] =
 diversity_of_network_attractiveness_factor[Android]*market_goal[Android]
 Operating_system.OS_quality_by_attributes[Network, iOS] =
 diversity_of_network_attractiveness_factor[iOS]*market_goal[iOS]
 Operating_system.OS_quality_by_attributes[Network, Windows] =
 diversity_of_network_attractiveness_factor[Windows]*market_goal[Windows]
 Operating_system.OS_quality_by_attributes[Network, BlackBerry] =
 diversity_of_network_attractiveness_factor[BlackBerry]*market_goal[BlackBerry]
 Operating_system.OS_quality_by_attributes[Network, Symbian&others] =
 diversity_of_network_attractiveness_factor[Symbian&others]*market_goal[Symbian&others]
 Operating_system.OS_quality_by_attributes[bandwagon, Android] =
 bandwagon_attractiveness[Android]*market_goal[Android]
 Operating_system.OS_quality_by_attributes[bandwagon, iOS] =
 bandwagon_attractiveness[iOS]*market_goal[iOS]
 Operating_system.OS_quality_by_attributes[bandwagon, Windows] =
 bandwagon_attractiveness[Windows]*market_goal[Windows]
 Operating_system.OS_quality_by_attributes[bandwagon, BlackBerry] =
 bandwagon_attractiveness[BlackBerry]*market_goal[BlackBerry]
 Operating_system.OS_quality_by_attributes[bandwagon, Symbian&others] =
 bandwagon_attractiveness[Symbian&others]*market_goal[Symbian&others]
 Operating_system.OS_quality_by_attributes[brand, Android] =
 brand_loyalty_attractiveness_factor[Android]*market_goal[Android]
 Operating_system.OS_quality_by_attributes[brand, iOS] =
 brand_loyalty_attractiveness_factor[iOS]*market_goal[iOS]
 Operating_system.OS_quality_by_attributes[brand, Windows] =
 brand_loyalty_attractiveness_factor[Windows]*market_goal[Windows]
 Operating_system.OS_quality_by_attributes[brand, BlackBerry] =
 brand_loyalty_attractiveness_factor[BlackBerry]*market_goal[BlackBerry]
 Operating_system.OS_quality_by_attributes[brand, Symbian&others] =
 brand_loyalty_attractiveness_factor[Symbian&others]*market_goal[Symbian&others]
 Operating_system.OS_quality_by_attributes[Model, Android] =
 model_attractiveness[Android]
 Operating_system.OS_quality_by_attributes[Model, iOS] = model_attractiveness[iOS]
 Operating_system.OS_quality_by_attributes[Model, Windows] =
 model_attractiveness[Windows]
 Operating_system.OS_quality_by_attributes[Model, BlackBerry] =
 model_attractiveness[BlackBerry]

```

Operating_system.OS_quality_by_attributes[Model, Symbian&others] =
model_attractiveness[Symbian&others]
Operating_system.Scenario_1 = 0
Operating_system.Scenario_2 = 0
Operating_system.Scenario_3 = 0
Operating_system.Scenario_4 = 0
Operating_system.Scenario_5 = 0
Operating_system.Scenario_total_switch =
Scenario_1+Scenario_4+Scenario_2+Scenario_3+Scenario_5
Operating_system.sum_of_brand_loyalty = SUM(Brand_loyalty)

```

Operating_system.CHARACTERISTICS_OF_STANDARD_SUPPORTER:

```

Operating_system.awareness_multiplier[OS] = IF TIME>=time_of_entry THEN
effect_of_fraction_of_awareness_on_awareness_multiplier*effect_of_loyalty_on_BA*BA
_effectieveness ELSE 0
Operating_system.BA_effectieveness[OS] = GRAPH(fraction_of_BA_reached)
(0.000, 0.966), (0.107645, 0.969298), (0.259021, 0.951754), (0.407034, 0.899123),
(0.590214, 0.807018), (0.740061, 0.679825), (0.874618, 0.500), (0.957187, 0.377193),
(1.01927, 0.250), (1.100, 0.0001)
Operating_system.BA_gap[OS] = Max_BA-brand_awareness
Operating_system.brand_awareness[Android](t) = brand_awareness[Android](t - dt) +
(change_in_brand_awareness[Android]) * dt
INIT Operating_system.brand_awareness[Android] = 30
Operating_system.brand_awareness[iOS](t) = brand_awareness[iOS](t - dt) +
(change_in_brand_awareness[iOS]) * dt
INIT Operating_system.brand_awareness[iOS] = 15
Operating_system.brand_awareness[Windows](t) = brand_awareness[Windows](t - dt) +
(change_in_brand_awareness[Windows]) * dt
INIT Operating_system.brand_awareness[Windows] = 5
Operating_system.brand_awareness[BlackBerry](t) = brand_awareness[BlackBerry](t - dt)
+ (change_in_brand_awareness[BlackBerry]) * dt
INIT Operating_system.brand_awareness[BlackBerry] = 1
Operating_system.brand_awareness[Symbian&others](t) =
brand_awareness[Symbian&others](t - dt) +
(change_in_brand_awareness[Symbian&others]) * dt
INIT Operating_system.brand_awareness[Symbian&others] = 1
INFLOWS:
Operating_system.change_in_brand_awareness[OS] =
((awareness_multiplier*BA_gap)/time_to_chng_BA)-
(forgetting_fraction*brand_awareness)/time_to_chng_BA
Operating_system.Brand_loyalty[Android](t) = Brand_loyalty[Android](t - dt) +
(increase_loyalty[Android] - loose_loyalty[Android]) * dt
INIT Operating_system.Brand_loyalty[Android] = 0
Operating_system.Brand_loyalty[iOS](t) = Brand_loyalty[iOS](t - dt) +
(increase_loyalty[iOS] - loose_loyalty[iOS]) * dt

```

```

INIT Operating_system.Brand_loyalty[iOS] = 0
Operating_system.Brand_loyalty[Windows](t) = Brand_loyalty[Windows](t - dt) +
(increase_loyalty[Windows] - loose_loyalty[Windows]) * dt
INIT Operating_system.Brand_loyalty[Windows] = 0
Operating_system.Brand_loyalty[BlackBerry](t) = Brand_loyalty[BlackBerry](t - dt) +
(increase_loyalty[BlackBerry] - loose_loyalty[BlackBerry]) * dt
INIT Operating_system.Brand_loyalty[BlackBerry] = 0
Operating_system.Brand_loyalty[Symbian&others](t) = Brand_loyalty[Symbian&others](t
- dt) + (increase_loyalty[Symbian&others] - loose_loyalty[Symbian&others]) * dt
INIT Operating_system.Brand_loyalty[Symbian&others] = 0
INFLOWS:
Operating_system.increase_loyalty[OS] = IF TIME >=time_of_entry THEN
((choice_and_desire_average*loyalty_gap)/loyalty_delay) ELSE 0
OUTFLOWS:
Operating_system.loose_loyalty[OS] = IF TIME >=time_of_entry THEN
((Brand_loyalty*MARKET.other_OS_attractiveness)/time_to_loose_loyalty) ELSE 0
Operating_system.choice_and_desire_average[OS] =
(effect_on_brand_choice*0.1)+(effect_of_complementary_goods_on_loyalty*0.9)
Operating_system.Ecosystem_strtaegy[Android] = 2
Operating_system.Ecosystem_strtaegy[iOS] = 5
Operating_system.Ecosystem_strtaegy[Windows] = 0
Operating_system.Ecosystem_strtaegy[BlackBerry] = 0
Operating_system.Ecosystem_strtaegy[Symbian&others] = 0
Operating_system.effect_of_complementary_goods_on_loyalty[OS] =
GRAPH(CM_good_relativeness*Ecosystem_strtaegy)
(0.000, 0.070), (0.500, 0.303), (1.000, 0.432), (1.500, 0.544), (2.000, 0.650), (2.500, 0.748),
(3.000, 0.856), (3.500, 0.950), (4.000, 0.996), (4.500, 1.008), (5.000, 1.002)
Operating_system.effect_of_fraction_of_awareness_on_awareness_multiplier[OS] =
SAFEDIV(fraction_of_awareness, total_fraction_of_awareness, 0)
Operating_system.effect_of_loyalty_on_BA[OS] = GRAPH(Brand_loyalty)
(0.0, 0.010), (4.58716, 0.149123), (11.0092, 0.311404), (24.4648, 0.526316), (37.3089,
0.657895), (50.1529, 0.74228), (62.3853, 0.811404), (75.2294, 0.872807), (87.4618,
0.916667), (100.0, 0.9167)
Operating_system.effect_on_brand_choice[OS] = ((SMTH1((Brand_loyalty*motivation),
smooth_time)))/100
Operating_system.forgetting_fraction[OS] = IF TIME >= time_of_entry THEN
0.1*(MARKET.other_OS_attractiveness) ELSE 0
Operating_system.fraction_of_awareness[OS] =
MARKET.Current_installed_base/MARKET.Total_population
Operating_system.fraction_of_BA_reached[OS] = brand_awareness/Max_BA
Operating_system.loyalty_delay = 1
Operating_system.loyalty_gap[OS] = max_loyalry-Brand_loyalty
Operating_system.Max_BA = 100
Operating_system.max_loyalry = 100
Operating_system.motivation[OS] = GRAPH(brand_awareness)
(0.0, 0.000), (13.14, 0.04824), (22.32, 0.1271), (28.44, 0.2938), (37.0, 0.500), (49.84,
0.6491), (61.46, 0.728), (74.92, 0.7807), (87.76, 0.7982), (100.0, 0.800), (110.0, 0.800)

```

Operating_system.smooth_time = 0.5
 Operating_system.time_to_chng_BA = 2
 Operating_system.time_to_loose_loyalty = 2
 Operating_system.total_fraction_of_awareness =
 (SUM(MARKET.Current_installed_base))/MARKET.Total_population

Operating_system.CHARACTERISTICS_OF_THE_STANDARD:

Operating_system.Application[OS](t) = Application[OS](t - dt) + (Development_rate[OS] -
 application_obsolescence_rate[OS]) * dt

INIT Operating_system.Application[OS] = 0

INFLOWS:

Operating_system.Development_rate[Android] = IF TIME>time_of_entry THEN
 Application[Android]*(OS_attractiveness_for_developers[Android]/SUM(OS_attractiveness_for_developers))/ application_dev_time ELSE IF TIME=time_of_entry THEN 120000 ELSE 0

Operating_system.Development_rate[iOS] = IF TIME>time_of_entry THEN
 Application[iOS]*(OS_attractiveness_for_developers[iOS]/SUM(OS_attractiveness_for_developers))/ application_dev_time ELSE IF TIME=time_of_entry THEN 120000 ELSE 0

Operating_system.Development_rate[Windows] = IF TIME>time_of_entry THEN
 Application[Windows]*(OS_attractiveness_for_developers[Windows]/SUM(OS_attractiveness_for_developers))/ application_dev_time ELSE IF TIME=time_of_entry THEN 50000 ELSE 0

Operating_system.Development_rate[BlackBerry] = IF TIME>time_of_entry THEN
 Application[BlackBerry]*(OS_attractiveness_for_developers[BlackBerry]/SUM(OS_attractiveness_for_developers))/ application_dev_time ELSE IF TIME=time_of_entry THEN 50000 ELSE 0

Operating_system.Development_rate[Symbian&others] = IF TIME>time_of_entry THEN
 Application[Symbian&others]*(OS_attractiveness_for_developers[Symbian&others]/SUM(OS_attractiveness_for_developers))/ application_dev_time ELSE IF TIME=time_of_entry THEN 4000 ELSE 0

OUTFLOWS:

Operating_system.application_obsolescence_rate[OS] = Application/application_obs_time

Operating_system.application_attractiveness_factor[OS] = SAFEDIV(Application, sum_of_application, 0)

Operating_system.application_dev_time = 0.5

Operating_system.application_obs_time = 10

Operating_system.CM_good_relativeness[OS] = SAFEDIV(complementary_goods, Previous_CM_goods, 0)

Operating_system.complementary_attractiveness_factor[OS] = (application_attractiveness_factor+complementary_goods_attractiveness_factor)/2

Operating_system.complementary_good_per_supppter = 10

Operating_system.complementary_goods[OS](t) = complementary_goods[OS](t - dt) + (chng_in_Complementary_goods[OS]) * dt

INIT Operating_system.complementary_goods[OS] = initial_CM_goods

INFLOWS:


```

Operating_system.chng_in_Complementary_goods[Android] = IF
TIME>=time_of_leave[Android] THEN 0 ELSE IF
TIME>=time_to_start_complementary_goods[Android] THEN
((Gap_of_CM_goods[Android]*MARKET.OS_relative_sales_rate[Android])/produce_time)-
(complementary_goods[Android]/obsolence_time) ELSE 0
Operating_system.chng_in_Complementary_goods[iOS] = IF TIME>=time_of_leave[iOS]
THEN 0 ELSE IF TIME>=time_to_start_complementary_goods[iOS] THEN
((Gap_of_CM_goods[iOS]*MARKET.OS_relative_sales_rate[iOS])/produce_time)-
(complementary_goods[iOS]/obsolence_time) ELSE 0
Operating_system.chng_in_Complementary_goods[Windows] = IF
TIME>=time_of_leave[Windows] THEN 0 ELSE IF
TIME>=time_to_start_complementary_goods[Windows] THEN
((Gap_of_CM_goods[Windows]*MARKET.OS_relative_sales_rate[Windows])/produce_time)-
(complementary_goods[Windows]/obsolence_time) ELSE 0
Operating_system.chng_in_Complementary_goods[BlackBerry] = IF
TIME>=time_of_leave[BlackBerry] THEN 0 ELSE IF
TIME>=time_to_start_complementary_goods[BlackBerry] THEN
((Gap_of_CM_goods[BlackBerry]*MARKET.OS_relative_sales_rate[BlackBerry])/produce_time)-
(complementary_goods[BlackBerry]/obsolence_time) ELSE 0
Operating_system.chng_in_Complementary_goods[Symbian&others] = IF
TIME>=time_of_leave[Symbian&others] THEN 0 ELSE IF
TIME>=time_to_start_complementary_goods[Symbian&others] THEN
((Gap_of_CM_goods[Symbian&others]
*MARKET.OS_relative_sales_rate[Symbian&others])/produce_time)-
(complementary_goods[Symbian&others]/obsolence_time) ELSE 0
Operating_system.complementary_goods_attractiveness_factor[OS] =
SAFEDIV(complementary_goods, sum_of_complementary_goods, 0)
Operating_system.effects_of_TOE_on_attractivness_for_developers[OS] = IF
TIME>=Time_of_app_store THEN 2 ELSE 1
Operating_system.Gap_of_CM_goods[OS] = IF
max_complementary_goods>complementary_goods THEN max_complementary_goods-
complementary_goods ELSE 0
Operating_system.initial_CM_goods[OS] = 0
Operating_system.max_complementary_goods[OS] =
Supporters*complementary_good_per_supppter*Supporters_fractions
Operating_system.obsolence_time = 15
Operating_system.OS_attractiveness_for_developers[OS] = IF TIME < time_of_leave
THEN
(((OS_flexibility_for_developrs/5)*0.7)+((OS_Reg_fee/5)*0.2)+((MARKET.Market_Share/100)*0.1))*effects_of_TOE_on_attractivness_for_developers ELSE 0
Operating_system.OS_flexibility_for_developrs[Android] = 5
Operating_system.OS_flexibility_for_developrs[iOS] = 4.5
Operating_system.OS_flexibility_for_developrs[Windows] = 2
Operating_system.OS_flexibility_for_developrs[BlackBerry] = 2
Operating_system.OS_flexibility_for_developrs[Symbian&others] = 1
Operating_system.OS_Reg_fee[Android] = 5
Operating_system.OS_Reg_fee[iOS] = 3

```

Operating_system.OS_Reg_fee[Windows] = 4
 Operating_system.OS_Reg_fee[BlackBerry] = 4
 Operating_system.OS_Reg_fee[Symbian&others] = 3
 Operating_system.Previous_CM_goods[OS] = PREVIOUS(complementary_goods,
 initial_CM_goods)
 Operating_system.produce_time = 2
 Operating_system.sum_of_application = SUM(Application)
 Operating_system.sum_of_complementary_goods = SUM(complementary_goods)
 Operating_system.Supporters_fractions[Android] = 0.5*(1-
 Switch_Open_strategy)+0.5*Switch_Open_strategy
 Operating_system.Supporters_fractions[iOS] = 1*(1-
 Switch_Open_strategy)+0.5*Switch_Open_strategy
 Operating_system.Supporters_fractions[Windows] = 0.5*(1-
 Switch_Open_strategy)+0.5*Switch_Open_strategy
 Operating_system.Supporters_fractions[BlackBerry] = 1*(1-
 Switch_Open_strategy)+0.5*Switch_Open_strategy
 Operating_system.Supporters_fractions[Symbian&others] = 0.5*(1-
 Switch_Open_strategy)+0.5*Switch_Open_strategy
 Operating_system.Time_of_app_store[Android] = 2008
 Operating_system.Time_of_app_store[iOS] = 2008
 Operating_system.Time_of_app_store[Windows] = 2012
 Operating_system.Time_of_app_store[BlackBerry] = 2009
 Operating_system.Time_of_app_store[Symbian&others] = 2009
 Operating_system.time_to_start_complementary_goods[Android] = 2009
 Operating_system.time_to_start_complementary_goods[iOS] = 2008
 Operating_system.time_to_start_complementary_goods[Windows] = 2010
 Operating_system.time_to_start_complementary_goods[BlackBerry] = 2008
 Operating_system.time_to_start_complementary_goods[Symbian&others] = 2020

Operating_system.OTHER_STAKEHOLDERS:

Operating_system.bandwagon_adaption[OS] =
 MARKET.adaption_fraction*contact_between_current_and_potential_users
 Operating_system.bandwagon_attractiveness[OS] = SAFEDIV(bandwagon_adaption,
 sum_bandwagon, 0)
 Operating_system.contact_between_current_and_potential_users[OS] =
 contact_of_potential_users*probability_of_contact
 Operating_system.contact_of_potential_users =
 MARKET.potential_installed_base*MARKET.contact_rate
 Operating_system.diversity_of_network_attractiveness_factor[OS] =
 SAFEDIV(Supporters, sum_of_supporters, 0)
 Operating_system.effect_of_size_of_network_on_SMPs = MARKET.size_of_network/5
 Operating_system.Gap_capacity_all_supporters[OS] = IF
 max_capacity_per_supporter>smartphone_production_capacity_per_supporter THEN
 (max_capacity_per_supporter-smartphone_production_capacity_per_supporter) ELSE 0

```

Operating_system.gap_of_SM_models[OS] = IF
max_SM_capacity_of_OS>Smartphone_Models THEN max_SM_capacity_of_OS-
Smartphone_Models ELSE 0
Operating_system.Gap_of_SM_producers = IF max_supporters>SM_producers THEN
(max_supporters-SM_producers) ELSE 0
Operating_system.gap_of_supporters[OS] = IF Supporters<SM_producers THEN
SM_producers-Supporters ELSE 0
Operating_system.max_capacity_per_supporter[Android] =
Max_capacity_per_supporter_Normal[Android]*(1-
Switch_Open_strategy)+Open_strategy_capacity*Switch_Open_strategy
Operating_system.max_capacity_per_supporter[iOS] =
Max_capacity_per_supporter_Normal[iOS]*(1-
Switch_Open_strategy)+Open_strategy_capacity*Switch_Open_strategy
Operating_system.max_capacity_per_supporter[Windows] =
Max_capacity_per_supporter_Normal[Windows]*(1-
Switch_Open_strategy)+Open_strategy_capacity*Switch_Open_strategy
Operating_system.max_capacity_per_supporter[BlackBerry] =
Max_capacity_per_supporter_Normal[BlackBerry]*(1-
Switch_Open_strategy)+Open_strategy_capacity*Switch_Open_strategy
Operating_system.max_capacity_per_supporter[Symbian&others] =
Max_capacity_per_supporter_Normal[Symbian&others]*(1-
Switch_Open_strategy)+Open_strategy_capacity*Switch_Open_strategy
Operating_system.Max_capacity_per_supporter_Normal[Android] = 150
Operating_system.Max_capacity_per_supporter_Normal[iOS] = 50
Operating_system.Max_capacity_per_supporter_Normal[Windows] = IF
Scenario_total_switch=0 THEN 50 ELSE IF Scenario_5=1 THEN IF TIME < 2017 THEN
50 ELSE 150 ELSE IF Scenario_4=1 THEN IF TIME<2009 THEN 50 ELSE 150 ELSE
150
Operating_system.Max_capacity_per_supporter_Normal[BlackBerry] = 50
Operating_system.Max_capacity_per_supporter_Normal[Symbian&others] = 50
Operating_system.max_SM_capacity_of_OS[OS] =
Supporters*smartphone_production_capacity_per_supporter
Operating_system.max_supporters = 40
Operating_system.model_attractiveness[OS] = Smartphone_Models/sum_of_models
Operating_system.model_obsolescence_time = 5
Operating_system.Open_strategy_capacity = 100
Operating_system.probability_of_contact[OS] =
MARKET.Current_installed_base/MARKET.potential_installed_base
Operating_system.SM_production_time = 1
Operating_system.SM_producers(t) = SM_producers(t - dt) + (chng_in_producers) * dt
INIT Operating_system.SM_producers = 5
INFLOWS:
Operating_system.chng_in_producers =
((Gap_of_SM_producers*MARKET.relative_total_sale_rate*effect_of_size_of_network_o
n_SMPs)/time_for_a_factory_to_joining_the_market)-
(SM_producers/time_for_a_company_to_leave_the_market)*0

```

$$\text{Operating_system.Smartphone_Models[Android]}(t) = \text{Smartphone_Models[Android]}(t - dt) + (\text{Production_rate[Android]} - \text{Obsolence_rate[Android]}) * dt$$

$$\text{INIT Operating_system.Smartphone_Models[Android]} = 0$$

$$\text{Operating_system.Smartphone_Models[iOS]}(t) = \text{Smartphone_Models[iOS]}(t - dt) + (\text{Production_rate[iOS]} - \text{Obsolence_rate[iOS]}) * dt$$

$$\text{INIT Operating_system.Smartphone_Models[iOS]} = 0$$

$$\text{Operating_system.Smartphone_Models[Windows]}(t) = \text{Smartphone_Models[Windows]}(t - dt) + (\text{Production_rate[Windows]} - \text{Obsolence_rate[Windows]}) * dt$$

$$\text{INIT Operating_system.Smartphone_Models[Windows]} = 0$$

$$\text{Operating_system.Smartphone_Models[BlackBerry]}(t) = \text{Smartphone_Models[BlackBerry]}(t - dt) + (\text{Production_rate[BlackBerry]} - \text{Obsolence_rate[BlackBerry]}) * dt$$

$$\text{INIT Operating_system.Smartphone_Models[BlackBerry]} = 0$$

$$\text{Operating_system.Smartphone_Models[Symbian\&others]}(t) = \text{Smartphone_Models[Symbian\&others]}(t - dt) + (\text{Production_rate[Symbian\&others]} - \text{Obsolence_rate[Symbian\&others]}) * dt$$

$$\text{INIT Operating_system.Smartphone_Models[Symbian\&others]} = 10$$

INFLOWS:

$$\text{Operating_system.Production_rate[Android]} = (\text{gap_of_SM_models[Android]} * \text{MARKET.indicated_market_share[Android]}) * \text{MARKET.total_awareness} / \text{SM_production_time}$$

$$\text{Operating_system.Production_rate[iOS]} = (\text{gap_of_SM_models[iOS]} * \text{MARKET.indicated_market_share[iOS]} * \text{MARKET.total_awareness}) / \text{SM_production_time}$$

$$\text{Operating_system.Production_rate[Windows]} = (\text{gap_of_SM_models[Windows]} * \text{MARKET.indicated_market_share[Windows]}) * \text{MARKET.total_awareness} / \text{SM_production_time}$$

$$\text{Operating_system.Production_rate[BlackBerry]} = (\text{gap_of_SM_models[BlackBerry]} * \text{MARKET.total_awareness} * \text{MARKET.indicated_market_share[BlackBerry]}) / \text{SM_production_time}$$

$$\text{Operating_system.Production_rate[Symbian\&others]} = (\text{gap_of_SM_models[Symbian\&others]} * \text{MARKET.indicated_market_share[Symbian\&others]}) * \text{MARKET.total_awareness} / \text{SM_production_time}$$

OUTFLOWS:

$$\text{Operating_system.Obsolence_rate[OS]} = \text{Smartphone_Models} / \text{model_obsolence_time}$$

$$\text{Operating_system.smartphone_production_capacity_per_supporter[OS]}(t) = \text{smartphone_production_capacity_per_supporter[OS]}(t - dt) + (\text{Chng_in_production_capacity[OS]}) * dt$$

$$\text{INIT Operating_system.smartphone_production_capacity_per_supporter[OS]} = 3$$

INFLOWS:

$$\text{Operating_system.Chng_in_production_capacity[Android]} = ((\text{Gap_capacity_all_supporters[Android]} * \text{MARKET.relative_total_sale_rate}) / \text{time_to_gain_capacity}) - (\text{smartphone_production_capacity_per_supporter[Android]} * \text{MARKET.other_OS_attractiveness[Android]} / \text{tiime_to_loose_capacity})$$

```

Operating_system.Chng_in_production_capacity[iOS] =
((Gap_capacity_all_supporters[iOS]*MARKET.OS_relative_sales_rate[iOS])/time_to_gain_capacity)
Operating_system.Chng_in_production_capacity[Windows] =
((Gap_capacity_all_supporters[Windows]*MARKET.relative_total_sale_rate)/time_to_gain_capacity)-
(smartphone_production_capacity_per_supporter[Windows]*MARKET.other_OS_attractiveness[Windows]/time_to_loose_capacity)
Operating_system.Chng_in_production_capacity[BlackBerry] =
(Gap_capacity_all_supporters[BlackBerry]*MARKET.OS_relative_sales_rate[BlackBerry])/time_to_gain_capacity
Operating_system.Chng_in_production_capacity[Symbian&others] =
((Gap_capacity_all_supporters[Symbian&others]*MARKET.relative_total_sale_rate)/time_to_gain_capacity)-
(smartphone_production_capacity_per_supporter[Symbian&others]*MARKET.other_OS_attractiveness[Symbian&others]/time_to_loose_capacity)
Operating_system.sum_bandwagon = SUM(bandwagon_adaption)
Operating_system.sum_of_models = SUM(Smartphone_Models)
Operating_system.sum_of_supporters = SUM(Supporters)
Operating_system.Supporter_join_at_the_entry_time[Android] = 0
Operating_system.Supporter_join_at_the_entry_time[iOS] = 4
Operating_system.Supporter_join_at_the_entry_time[Windows] = 0
Operating_system.Supporter_join_at_the_entry_time[BlackBerry] = 4
Operating_system.Supporter_join_at_the_entry_time[Symbian&others] = 0
Operating_system.Supporter_join_rate[OS] =
SAFEDIV(OS_attractiveness_for_supporters, SUM(OS_attractiveness_for_supporters), 0)
Operating_system.Supporters[OS](t) = Supporters[OS](t - dt) + (supporters_join[OS] - Supporter_leave[OS]) * dt
INIT Operating_system.Supporters[OS] = 0
INFLOWS:
Operating_system.supporters_join[OS] = IF TIME < time_of_entry THEN 0 ELSE IF TIME=time_of_entry THEN Supporter_join_at_the_entry_time ELSE IF appropriability_strategy>0 THEN ((gap_of_supporters*Supporter_join_rate)/time_to_attract_supporter) ELSE 0
OUTFLOWS:
Operating_system.Supporter_leave[OS] = IF TIME >= time_of_leave THEN Supporters/time_to_leave ELSE IF appropriability_strategy>0 THEN (1-Supporter_join_rate)*Supporters/time_to_leave ELSE 0
Operating_system.time_to_loose_capacity = 2
Operating_system.time_for_a_company_to_leave_the_market = 5
Operating_system.time_for_a_factory_to_joining_the_market = 8
Operating_system.time_to_attract_supporter = 3
Operating_system.time_to_gain_capacity = 3
Operating_system.time_to_leave = 2

*****
Operating_system.STANDARD_SUPPORT_STRATEGY:

```

```
Operating_system.appropriability_strategy[Android] = 1
Operating_system.appropriability_strategy[iOS] = 0*(1-
Switch_Open_strategy)+1*(Switch_Open_strategy)
Operating_system.appropriability_strategy[Windows] = IF Scenario_5=1 THEN IF
TIME>2017 THEN 1 ELSE 0.5 ELSE IF Scenario_4=1 THEN IF TIME< 2009 THEN 0.5
ELSE 1 ELSE IF Scenario_3=1 THEN IF TIME<2012 THEN 1 ELSE 0.5 ELSE IF
Scenario_2=1 THEN 1 ELSE IF Scenario_1=1 THEN 1 ELSE IF Switch_Open_strategy=0
THEN IF TIME < 2013 THEN 0.5 ELSE 0.1 ELSE 1
Operating_system.appropriability_strategy[BlackBerry] = 0*(1-
Switch_Open_strategy)+1*(Switch_Open_strategy)
Operating_system.appropriability_strategy[Symbian&others] = IF Switch_Open_strategy=0
THEN IF TIME < 2008 THEN 0.25 ELSE 0.25 ELSE 1
Operating_system.OS_attractiveness_company_strategy[OS] = IF TIME > time_of_entry
THEN IF TIME < time_of_leave THEN OS_flexibility*appropriability_strategy ELSE 0
ELSE 0
Operating_system.OS_attractiveness_for_supporters[OS] = IF TIME>=time_of_leave
THEN 0 ELSE IF TIME>time_of_entry THEN
((MARKET.Market_Share/100)*weight_of_market_share+OS_attractiveness_company_str
ategy*weight_of_OS_attractiveness_strategy)*appropriability_strategy ELSE 0
Operating_system.OS_flexibility[Android] = 1
Operating_system.OS_flexibility[iOS] = 0
Operating_system.OS_flexibility[Windows] = IF Scenario_2=1 THEN 1 ELSE 0.25
Operating_system.OS_flexibility[BlackBerry] = 0
Operating_system.OS_flexibility[Symbian&others] = 0.25
Operating_system.Switch_Open_strategy = 0
Operating_system.time_of_entry[Android] = 2008.5
Operating_system.time_of_entry[iOS] = 2007
Operating_system.time_of_entry[Windows] = 2003
Operating_system.time_of_entry[BlackBerry] = 2002
Operating_system.time_of_entry[Symbian&others] = 2000
Operating_system.time_of_leave[Android] = 3000
Operating_system.time_of_leave[iOS] = 3000
Operating_system.time_of_leave[Windows] = 3000
Operating_system.time_of_leave[BlackBerry] = 2016
Operating_system.time_of_leave[Symbian&others] = 2013
Operating_system.weight_of_market_share = 0.3
Operating_system.weight_of_OS_attractiveness_strategy = 0.7
{ The model has 131 (495) variables (array expansion in parens).
In this module and 0 additional modules with 4 sectors.
Stocks: 8 (36) Flows: 12 (56) Converters: 111 (403)
Constants: 40 (84) Equations: 83 (375) Graphicals: 5 (21)
There are also 30 expanded macro variables.
}
```