A System Dynamics Approach to Data Center Capacity Planning

– A Case Study –

By
Kaveh Dianati

Thesis submitted in partial fulfillment of the requirements of the Master of Philosophy in System Dynamics from Universitetet i Bergen, Master of Philosophy in System Dynamics from Università degli Studi di Palermo, and Master of Science in Business Administration from Radboud Universiteit Nijmegen

Supervised by Professor Pål Davidsen

System Dynamics Group
Department of Geography
University of Bergen

June, 2012

Acknowledgements

I would like to express my sincere gratitude to Professor Pål Davidsen, who realized the opportunity for me to work on this exciting project, and who was an outstanding advisor and motivator all through the thesis period. I would also like to thank the client of this project, whom I unfortunately cannot name, for being always extremely helpful and generous with his time.

I am forever indebted to Professor Ali N. Mashayekhi, my first teacher in systems thinking. Having been his student is my timeless pride and one of the best things that ever happened to me.

Finally, I dedicate this result of my work to my father and mother, to whom I owe far more than I can ever give.

تقدیم به بهترین های زمین، پدر و مادرم

Abstract

This thesis is an empirical study where the System Dynamics methodology is applied to help the Chief Technical Officer of a Norwegian IT company, operating in the cloud computing industry, in planning for future data center capacity.

Put simply, cloud computing is the provisioning of centralized IT services and infrastructure to businesses in an on-demand, reliable, and inexpensive fashion, which is why it is sometimes loosely referred to as 'computing as a utility'.

The client's main interest in this project is to gain an analysis tool that can help in estimating the point in time at which the capacity limit of the company's data center in Oslo will be reached. This is a critical question for the business since setting up a new data center has a lead time of around one year, and it is essential to start planning for such an effort well beforehand. In this thesis, a System Dynamics model is built for this purpose, with its structure based on empirical knowledge elicited from the client of the project. Rigorous testing is applied to build confidence in the reliability and usefulness of the model. The model structure successfully replicates historical behavior of important variables in the system. The established robustness of the model qualifies it as suitable to use for policy and scenario testing. A few examples of such tests are carried out and documented in this report, including various tests regarding the central question of when the data center's capacity limit will be reached. This model can eventually become the basis of a management flight simulator that the client could use to try out different policies to see their consequences before implementing them in the real world.

This project has been carried out with two overarching purposes, one professional and one academic. The professional goal, as already mentioned, is to help the client in medium-term capacity planning. The academic aspiration of the thesis, however, is to establish the usefulness of the System Dynamics methodology in data center planning and cloud computing business fields. To the best of the author's knowledge, no previous System Dynamics works have been carried out in this area. Yet, being dominated by aging chains, co-flows, accumulations, delays, and feedbacks, data center management is in this thesis demonstrated to be a promising area for applying System Dynamics.

Table of Contents

List of Figures

Chapter One - Introduction

The subject of this thesis is a practical System Dynamics modeling project for a Norwegian IT company. The client is a leading provider of centralized IT operation services and net-based software distribution, established in 1997. The purpose of the modeling effort is to help the client in medium-term data center capacity planning. The main research question is to estimate the point in time at which, under different policies and environmental outcomes, the data center in question would run out of power capacity. This is a critical question for the client since setting up a new data center or upgrading the power capacity of the current data center is a very time-consuming and cost-intensive effort. The System Dynamics model was built in a period of around five months, February until June 2012, with regular interactive interviews with the client's contact person, who was the Chief Technical Officer of the firm, where model progress was presented regularly and questions were asked based on model structure and behavior, in order to complete the structure and analyze the behavior against historical data.

The industry within which the client company operates is the cloud computing industry. The American National Institute for Standards and Technology (NIST) defines *cloud computing* as 'a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction (The NIST Definition of Cloud Computing, 2011)'. In cloud computing, resources (e.g., CPU and storage) are provided as general utilities that can be leased and released by users through the Internet in an on-demand fashion (Zhang, Cheng, & Boutaba, 2010). Cloud computing is attractive to business owners as it gives them the possibility to outsource hardware, software, and management needed for IT services. A computing cloud gives scalable, quality-of-service (QoS) guaranteed, customizable, and inexpensive computing infrastructures on demand. If a business's IT services are outsourced to a cloud computing service provider, users will not have to manage various software installations, configuration and updates and computing resources and other hardware will no longer be prone to becoming outdated soon (Wang, et al., 2010).

According to Zhang, Cheng, and Boutaba (2010), the architecture of a cloud computing environment can be divided into four layers: the hardware/datacenter layer, the infrastructure layer, the platform layer and the application layer (Figure 1). The hardware layer, contained in the real world in data centers, is responsible for managing the physical resources of the cloud, including physical servers, routers, switches, power and cooling systems. A data center usually contains thousands of servers placed in racks and connected through switches, routers or other fabrics. Servers can be divided into various categories based on their usage, such as proxy servers, web servers, FTP servers, mail servers, and database servers. The infrastructure layer, or 'the virtualization layer', creates a pool of storage and computing resources by virtualizing the physical resources into several partitions each using virtualization technologies such as Xen and VMware. Virtualization is an essential component of cloud computing, since many key features, such as dynamic resource assignment, are only made available through virtualization technologies (Zhang, Cheng, & Boutaba, 2010).

The platform layer consists of operating systems and application frameworks, such as Java and .NET, and the application level, the highest level of the hierarchy, is responsible for delivering actual applications to end-users at a higher performance and lower cost level than traditional applications (Zhang, Cheng, & Boutaba, 2010). In this thesis, the focus is going to be on the two lower levels of the hierarchy, also known as Infrastructure as a Service (IaaS).

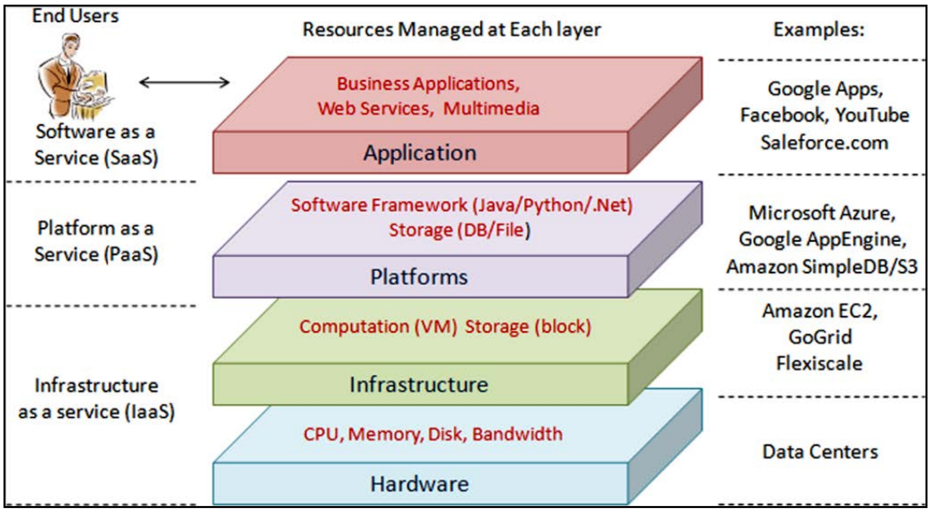


Figure 1 - Cloud Computing Architecture; borrowed from Zhang, Cheng, & Boutaba (2010)

The problem of interest for the client company of this project, which we will hereafter refer to as Abundant Bergen Clouds (ABC), is a problem of capacity planning. The main physical assets of ABC, as a cloud computing service provider, consists of data centers which accommodate racks for holding servers and storage spindles. Since data centers have a long set-up delay of about 12 months, it is very important for the company to be able to anticipate the need for capacity expansion. In the next chapter, we are going to outline the problem to be investigated in this thesis in a more detailed and formal manner.

In order to tackle this problem, we are going to use System Dynamics modeling to construct a model reflecting the dynamics of a data center. To the best of the author's knowledge, up to now there has not been any application of the System Dynamics methodology to the field of data center management. Yet, being rich in dynamic complexity, this field seems to be a promising area for applying System Dynamics. With this study, we hope to establish the usefulness of System Dynamics in this context.

The rest of this report is structured as follows: In Chapter Two we define our problem of interest formally while studying important reference modes. Chapter Three deals with our dynamic hypothesis and describes the structure of the model. In Chapter Four we conduct various tests to establish the usefulness of the model for its purpose. Finally, in Chapter Five we run several sample policy and scenario tests to demonstrate the kinds of analysis for which the model can be used. In the end we discuss the usefulness and the limitations of the model, suggesting paths for future work.

Chapter Two - Problem Outline

2.1. Introduction

As the result of this project, what ABC would like is a tool that would make it possible to anticipate *when* the limit of its data center in Oslo, in terms of capacity, would be reached and what would be the best way to respond to this in terms of different technological and market strategies. Main limiting factors within a data center are space and electric power. According to the experience of the client, the space limitation is seldom reached because before this ever happens the data center power cap is already surpassed. Therefore, to investigate the capacity of the data center in question, we focus our attention on developments in required electric power. Physical servers and storage spindles are the main consumers of electricity in a data center, both for operation and for cooling. Thus, to track and anticipate the required power in a data center, we need to study the development of the number of physical servers and storage spindles which are driving power demands. Furthermore, servers and spindles are the main physical assets that are used to provide cloud computing services, and so they determine the capacity of ABC in serving its clients.

2.2. Physical Servers and Processing Capacity

Physical servers are the predominant consumers of power in data centers, as opposed to storage spindles which typically use only less than ten percent of total power. Moreover, servers are the providers of processing power requirements of users, which arise before and above storage requirements. As a result, our primary variable of interest, both in relation to service capacity on the demand side and in relation to power requirements as a limit to supply, becomes the number of physical servers. Therefore, our dynamic hypothesis would above all need to explain the historical development of this variable, which is shown in Figure 2[1]. The number of physical servers, as can be seen, started at around 1000 units in the beginning of 2007 and has been growing steadily ever since, reaching around 3000 units at present.

[1] All data provided by the client start with January 2007, which is considered as the starting point of the timeframe of interest in this project.

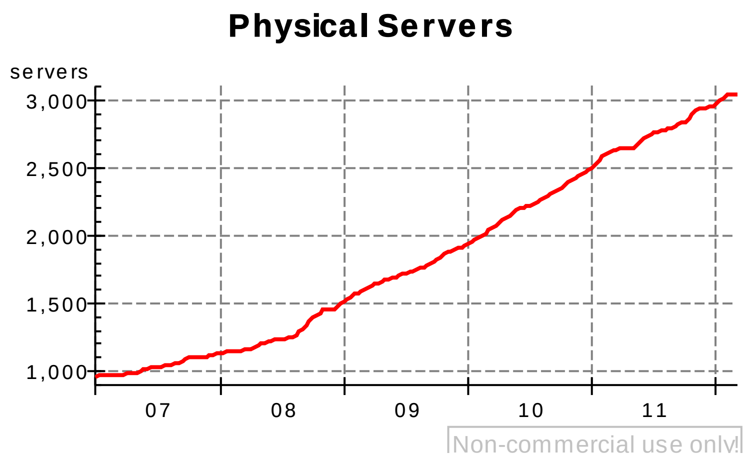


Figure 2 - Reference mode for Physical Servers

Among the physical servers' most crucial attributes are processing capacity and memory (RAM). Processing capacity depends on the number of Central Processing Units (CPUs) inside the physical servers and the number of virtual cores that those CPUs contain. Processing capacity technology has been improving ever since the advent of computers. Since 2007, the physical servers that ABC has been purchasing have constantly had two CPUs each, but the number of virtual cores on those CPUs has advanced from two to four and then to six. In other words, the performance of newly purchased physical servers in terms of processing power has evolved from 2x2 to 2x4 and further to 2x6, the first number representing number of CPUs on each server and the second one, the number of virtual cores on each CPU. With this in mind, the amounts of CPUs and CPU Cores have grown along with the growth in the number of physical servers, as can be seen in Figure 3 and Figure 4.

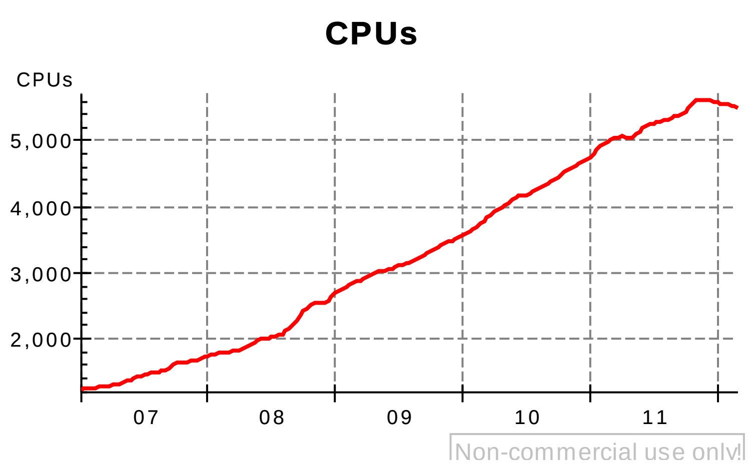


Figure 3 - Reference mode for number of CPUs

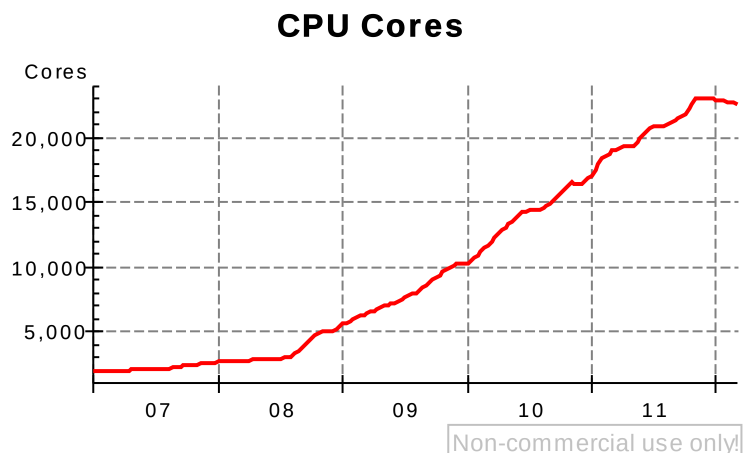


Figure 4 - Reference mode for number of CPU Cores

It can be noticed that the growth in CPU cores accelerates in mid-2008 as new CPUs being installed have twice as many virtual cores as the old ones.

As for Random Access Memory (RAM), the other important attribute of servers, no data was provided by the client of the project.

2.3. Hosts, Virtual Servers, and Logical Servers

Processing power requirements of users are responded by 'logical servers', with several users sharing the same logical server. A logical server can be either a stand-alone physical server, directly being used to run user applications, or a 'virtual server'. One physical server can be 'virtualized' into several virtual servers (also called 'virtual machines'). A virtual server is an abstraction layer or environment with an operating system between hardware components and the end-user. A host operating system can run many virtual servers that share system hardware components such as CPUs, controllers, disk, memory, and I/O (Daniels, 2009). "Virtualization technologies multiplex hardware and thus provide flexible and scalable platforms. Virtual machine techniques, such as VMware and Xen, offer virtualized IT-infrastructures on-demand (Wang, Kunze, Jie, Castellanos, Kramer, & Karl, 2008)".

The historical development of 'hosts', those physical servers used for hosting virtual servers, can be seen in Figure 5. In Figure 6 we can see the reference mode for the number of virtual servers. It can be instantly noticed that the use of virtualization technology in ABC drastically accelerates in 2010, as a result of advancements in virtualization technology. As virtual servers start growing more rapidly, ABC starts ramping up the amount of hosts in a step-wise manner. This means that hosts are added in chunks, as the maximum capacity for supporting the growth in virtual servers is being reached time and again.

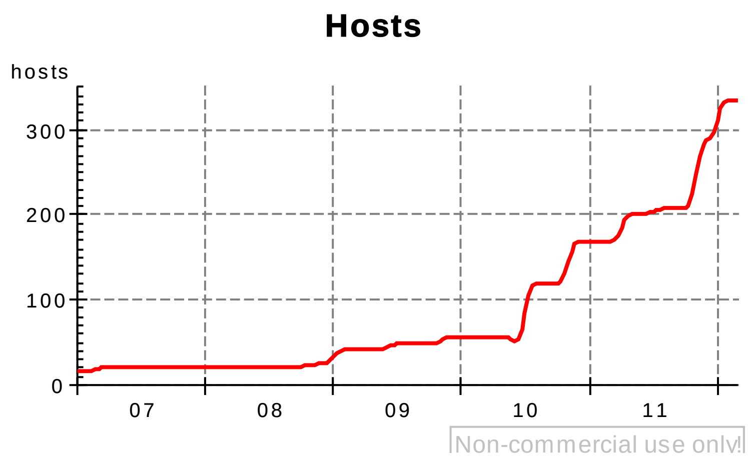


Figure 5 - Reference mode for number of Hosts

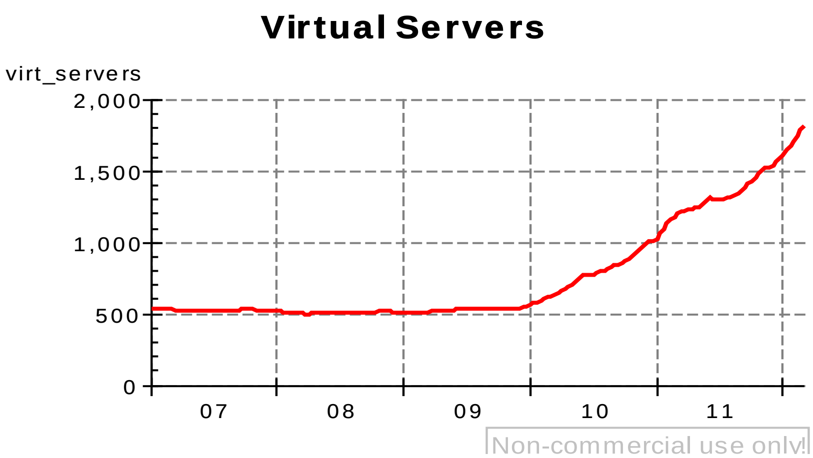


Figure 6 - Reference mode for number of Virtual Servers

The next graph shows the historical development of those physical servers which are *not* being used as hosts for virtual servers, but are directly serving user applications. This group of servers is referred to as 'Stand-alone Physical Servers'.

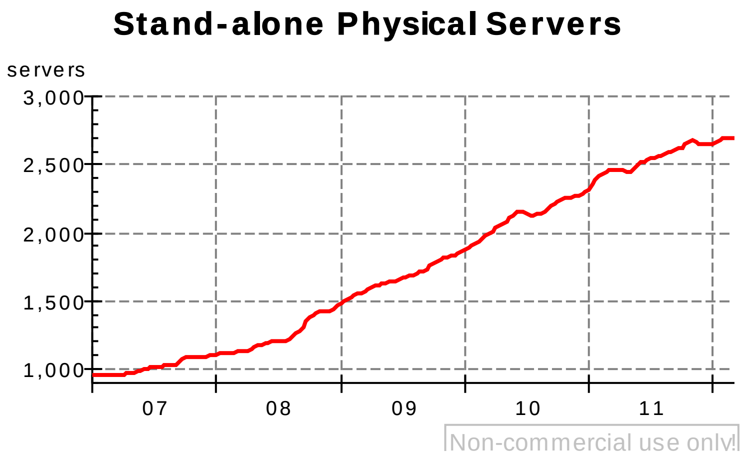


Figure 7 - Reference mode for Stand-alone Physical Servers

As explained earlier, a logical server, a server meeting the needs of users, can be either a stand-alone physical server or a virtual server. The growth in the amount of logical servers, along with the growth in its two components, is depicted in Figure 8. The number of logical servers is the sum of the number of stand-alone physical servers and the number of virtual servers at any point in time.

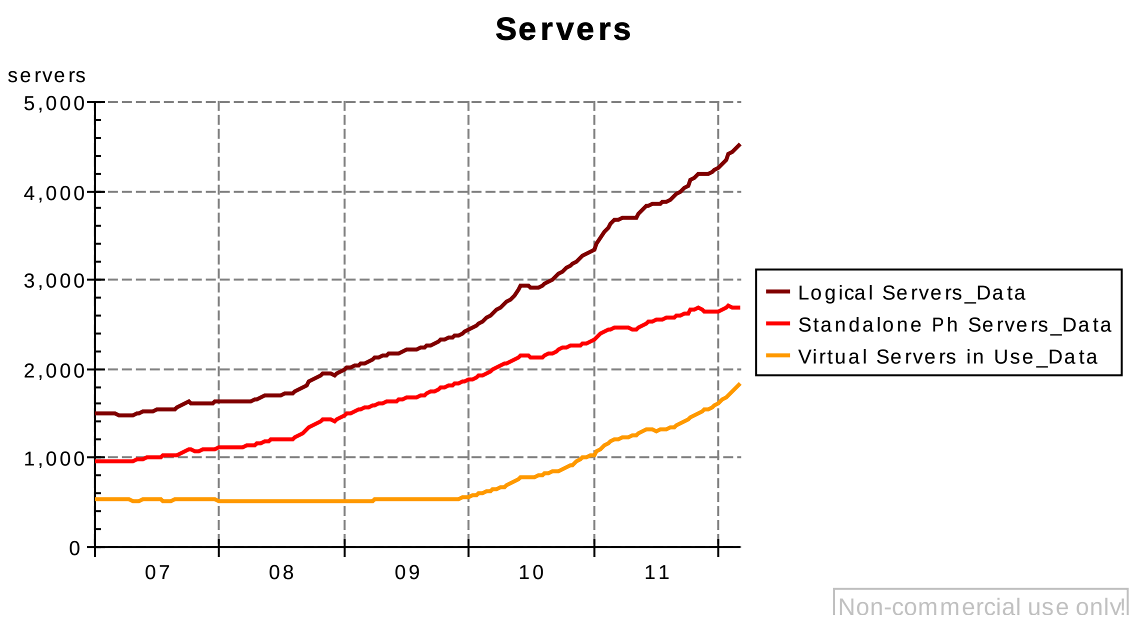


Figure 8 - Reference mode for Logical Servers

2.4. Storage Capacity

One of the main services that ABC offers to its clients is storage space for data. For this reason, storage spindles are another set of important assets that the company owns. Along with the growth in the number of users, ABC has had to acquire more and more spindles (Figure 9). Each one of these spindles adds a certain amount of storage capacity in terabytes (Figure 10). Spindles and their storage capacity have been growing steadily up until 2010, when there comes about a period of stagnation of about one year, after which they start growing almost twice as fast as before. This brings the current values for these two variables to a level where it would almost have been if the steady growth between 2007 and 2009 would have continued. The client did not have precise information about the reason for this change in trend in 2010-11.

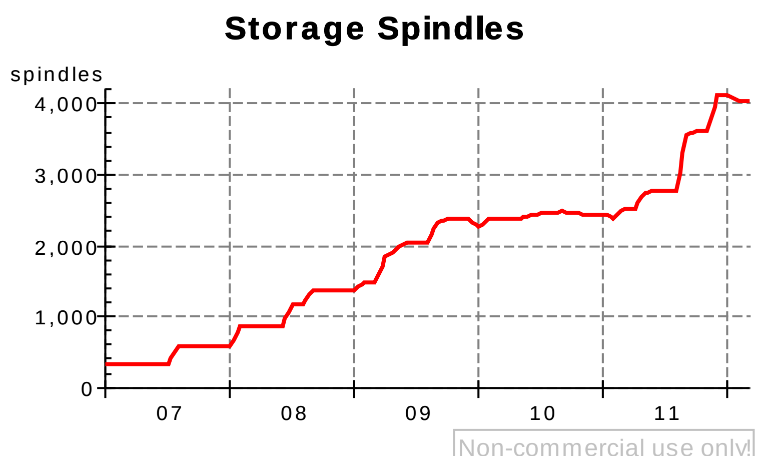


Figure 9 - Reference mode for the number of Spindles

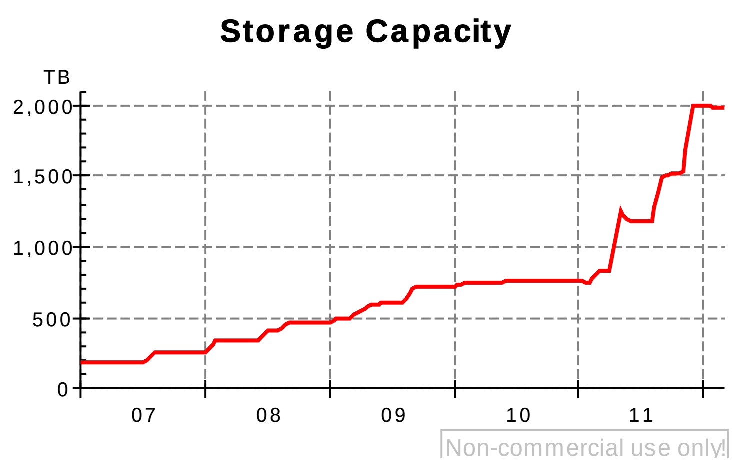


Figure 10 - Reference mode for Storage Capacity

2.5. Users and Revenues

The growth in all of the above reference modes is driven by the growth in ABC's user base. As the company is growing, it signs contracts with more and more clients, each of which adds to the user base served by ABC's servers. Hence, the number of users in the company's user base is the driving force behind the acquisition of more and more servers and storage spindles by ABC. Therefore, the trend in the number of users needs to be replicated at the outset in order to endogenously reproduce the growth in the number of servers. This trend, which is shown in Figure 11, is characterized with a more or less steady growth, faster towards the beginning, slower in 2009 and 2010, and starting to gain speed again in 2011. According to the client, the near-stagnation of growth in 2009 and 2010 should be attributed to the global financial crisis.

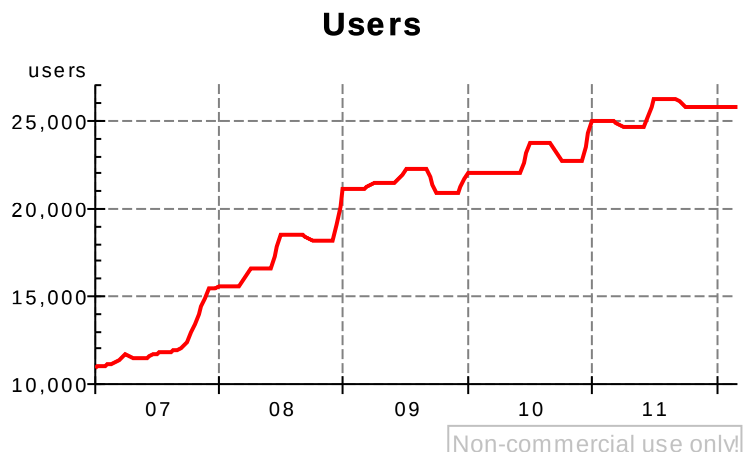


Figure 11 - Reference mode for number of Users

ABC segments its market into two distinct types of users which they call ‘full desktop’ users and ‘distributed services’ users. The former are high-requirement users who need significant processing power, memory, and storage, while the latter type of users are relatively low-requirement ones. Naturally, the price that ABC charges is higher for full desktop users than for distributed services users.

ABC charges its clients on a per-user per-month basis, in accordance with standard ‘computing as a utility’ business model in the cloud computing industry. Figure 12 depicts the growth in ABC’s recurring monthly revenues. For revenues in 2007, no data was available. Revenues are generally growing, but almost stagnant between 2009 and 2010, partly due to the financial crisis, and partly because the average prices for cloud computing services has been decreasing. Figure 13 shows the declining trend in the ‘Average Revenue per User’ indicator (known as ARPU).

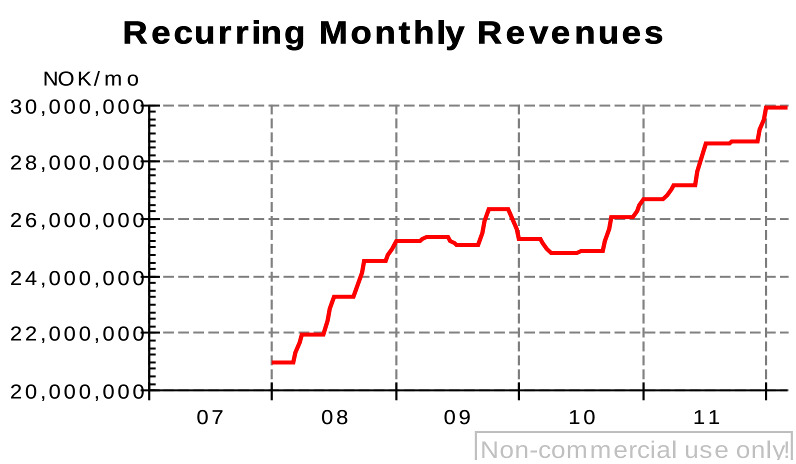


Figure 12 - Reference mode for Recurring Monthly Revenues

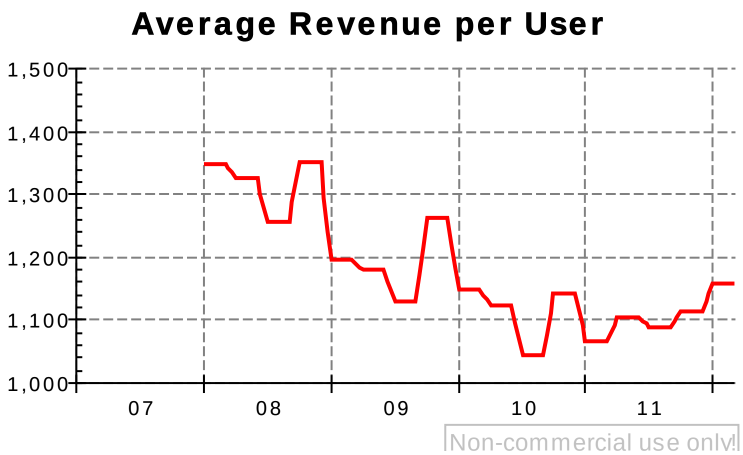


Figure 13 - Reference mode for ARPU

Chapter Three - Dynamic Hypothesis

3.1. Introduction

In order to analyze different scenarios and strategies for ABC, we need to first understand and model the structure of the internal processes of the company, and the way its assets and its user base correlate and develop through time. For this purpose, we construct a System Dynamics model that captures the structure of the firm, which we can later use for policy analysis. In this chapter, we are going to study the SD model built in the course of this project, which consists of several sectors, including the Servers sector, the Spindles sector, Data Center sector, the Users sector, and the Accounting & Finance sector. The backbone of the system is captured in the Servers sector, which includes structure for physical servers, their major attributes and electric power requirements, hosts, and virtual servers. In the next section, we are going to start studying the structure of the model with this main sector.

3.2. The Servers Sector

This sector of the model is based on an aging chain and co-flow pattern of structure, where the main flow is the flow of physical servers through successive age cohorts. This main pipeline is accompanied by several co-flows reflecting the capacity of these servers in terms of processing power and memory, their price, and also hosts and virtual servers. We'll begin by looking at the aging chain of physical servers.

3.2.1. Physical Servers

As physical servers are acquired to keep the 'Inventory of Physical Servers' (Figure 14) at the desired level ('Need-based Ph Server Acquisition' is described in 3.2.2 in more detail), they first reside in an inventory up until they are installed. Installation rate is governed by growing processor requirements of users ('Need-based Ph Server Installation' is described in 3.4.4). This rate is limited by the availability of surplus power in the data center (see section 3.5), by availability of installation engineers (3.6.3), and also naturally by inventory. Once physical servers are installed they start operating as the first year cohort 'Physical Servers Y1', where they reside for twelve months before they move to the second year cohort. Inside each cohort, elements shift forward on a monthly basis. This is made possible using an array with twelve elements (one for each month), along with the rate 'Shift Y1' which flows out of each stock and back into it, enabling the monthly shift of array elements. This structure allows tracking and managing the assets of the company on a monthly level of disaggregation, without having to build a messy aging chain with twelve stocks for each cohort. In this model, discrete inflows and outflows for each month are formulated using a monthly pulse function.

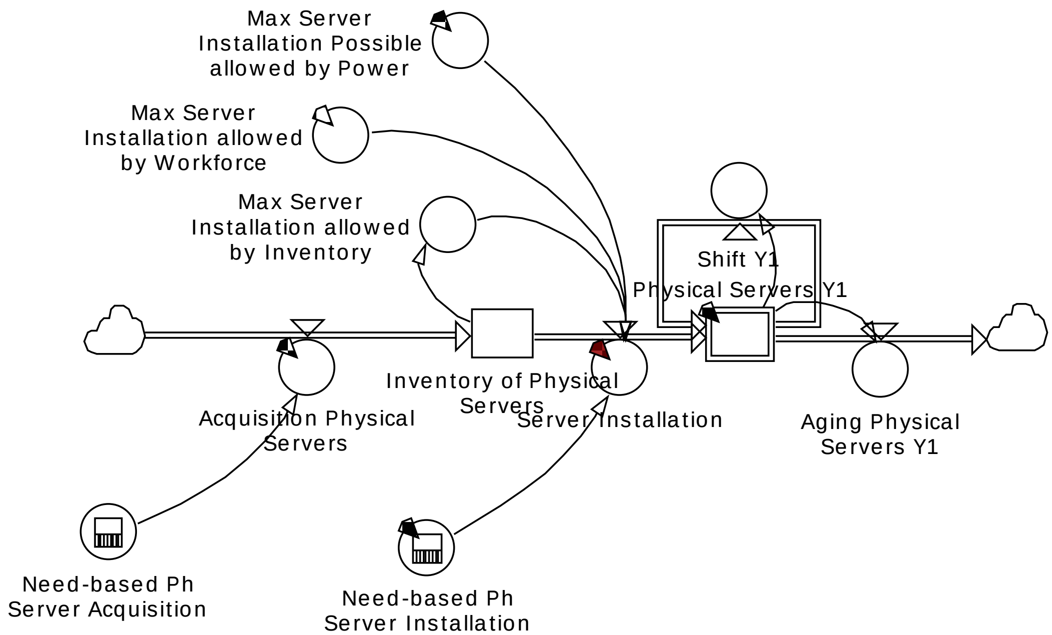


Figure 14 - Physical Server Acquisition & Installation

Physical servers move along the pipeline month by month inside each stock, and every twelve months they leap to the next cohort. This mechanism continues in the same way for five age cohorts, starting from the time when servers reside in the first year cohort ‘Physical servers Y1’ up until they reach the fifth year cohort ‘Physical Servers Y5’ (Figure 15) .

Average lifetime of physical servers was not a straightforward parameter to elicit. Accounting lifetime for physical servers is three years. In reality, though, physical servers live longer than that. Servers are typically replaced when the server service contract expires or when the cost-to-performance ratio gets too high. According to the client’s first estimate physical servers live around five years. The base number of cohorts for the model was based on this estimate. Nevertheless, after testing the behavior of this part of the model against historical data it became apparent that physical servers are used in fact longer than that, particularly in recent years. Simulating the model using the historical inflow of servers as exogenous input and setting the lifespan of servers to five years generated a curve for the total number of physical servers which was considerably lower than the data curve. The inflow being data-driven, it was established that the actual lifespan of physical servers has been longer than five years on average. The client considered this insight an important one because it pointed to a possibly unchecked policy for disengaging physical servers.

Since the length of this period of physical servers being used in excess of five years is on the one hand uncertain and on the other varying, we use a single stock to represent those servers that are older than five years.

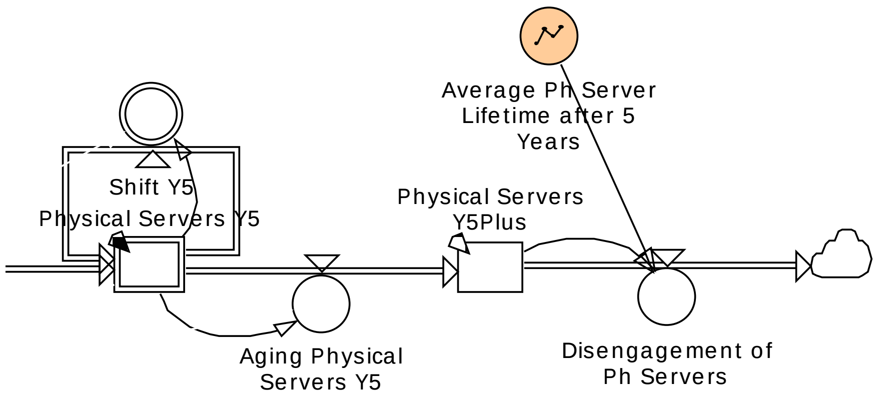


Figure 15 - Disengagement of Physical Servers

Unlike previous cohorts, this last stock is not an array since it does not represent a fixed number of months. As physical servers are disengaged, they flow out of this stock with a first order delay. The length of this delay is a graph function and is estimated using the client's knowledge and reference mode comparison tests once the structure has been conceptually validated (see section 4.1.1).

3.2.2. Need-based Server Acquisition

As pointed out earlier, acquisition of physical servers in ABC is based on a policy of keeping the inventory of physical servers at the desired level. The 'Desired Inventory of Physical Servers' (Figure 16) is set so as to make sure there are enough servers in the inventory for 'Desired Ph Server Inventory Coverage Period', assuming the current rate of 'Need-based Server Installation' (section 3.4.4). The variable 'Gap between Desired & Actual Inventory of Ph Servers' is self-explanatory. The next variable in line is simply this gap with the negative parts taken away. This provision is taken so that the rate of acquisition does not become negative in case our inventory is already higher than desired. Lastly, the target variable 'Need-based Ph Server Acquisition' is a delayed version of the non-negative gap between the desired and actual state of the inventory. The length of the delay is set to one month, and the order of the delay is set to six. This short high-order delay represents the time it takes from the moment ABC orders new machines until they receive it[1].

[1] In reality, acquisition is also limited based on inventory capacity. If there are already too many servers in inventory, no new orders will be put. This limit is not yet included in the model because I had no information about maximum inventory capacity.

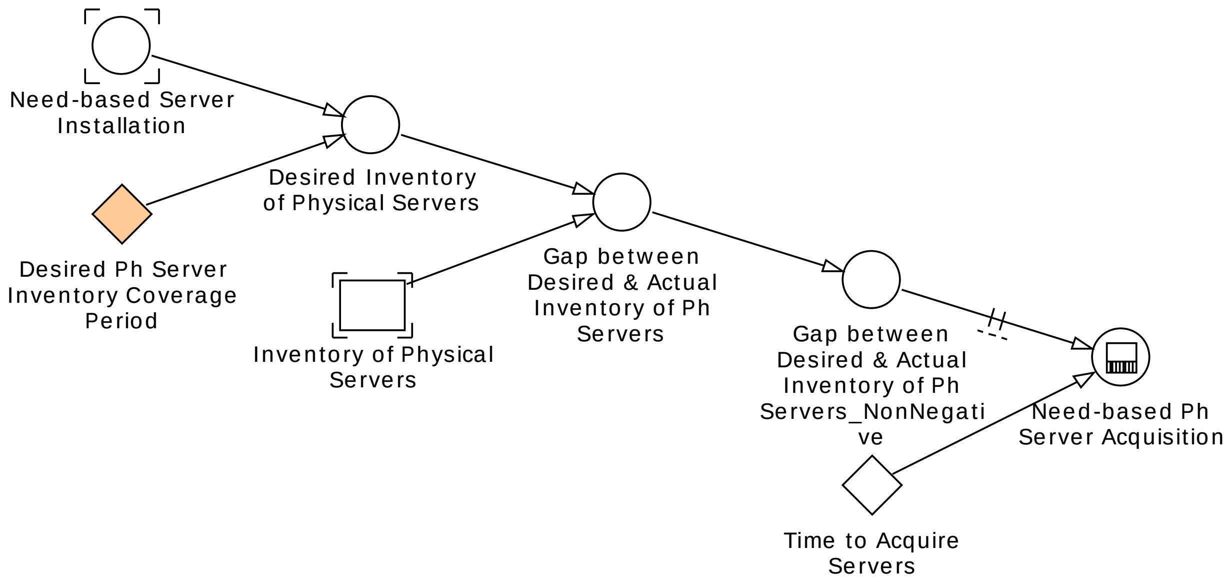


Figure 16 - Need-based Server Acquisition[1]

3.2.3. Stand-alone Physical Servers, Hosts, and Virtual Servers

In order to capture the fact that physical servers are utilized in two different configurations, either as stand-alone servers or as hosts for virtual servers, we use the co-flow structure portrayed in Figure 18. As new physical servers are purchased and put to use, a certain ratio of them are set up as hosts accommodating several virtual machines each. This 'Ratio of New Ph Servers being Converted to Hosts' is extracted from data, by dividing the yearly number of hosts added by the yearly number of physical servers added. This structure does not capture the exact moments in time when chunks of hosts are added to the fleet, but rather distributes these added chunks over the year. Nevertheless, this structure sufficiently replicates the growing trend in the number of hosts (see section 4.1.6). Every physical server being operated as a host can support a limited number of virtual servers. This number varies widely, depending on the number of virtual cores on the physical server, its RAM, the type and intensity of the user workloads, and the technology used for virtualization. The sum of the capacity of all our hosts puts a cap on the number of virtual servers that we can run. The dynamics of the possible number of virtual servers per host is highly complex, and beyond the scope of this thesis.

Those physical servers which are not converted to hosts are utilized as stand-alone. In other words, the variable 'Ratio of New Ph Servers starting as Stand-alone' in Figure 18 equals one minus the 'Ratio of New Ph Servers being Converted to Hosts'. These servers move along the pipeline with a conventional co-flow structure, with each of the flows in between stocks being formulated as the ratio of stand-alones over total physical servers of that cohort multiplied by the corresponding rate from the main pipeline of physical servers above. Within each stock, the elements of each array shift forward every month, just like they did for the original pipeline of physical servers. This monthly shift structure is in

[1] For a legend about what different colored variables, such as the one in this figure, are supposed to represent, please refer to Appendix B, p. 136.

place for all stocks in this sector, except the final cohort of each co-flow, which is a single element variable rather than an array.

A physical server can be set up as a host in the beginning of its lifetime, and in that case it will remain so until it is eventually disengaged. Nevertheless, physical servers can also be converted from stand-alone to hosts (but not in the other direction) later on in their lifetime. This constitutes a flow from a stock of stand-alone servers to the corresponding stock of hosts, governed by a policy decision. This structure will be added to the model as a policy structure in 5.2.2.

The pipeline of stand-alone physical servers moves in parallel to the main physical servers' pipeline in complete correspondence with it. For the pipeline of hosts, however, there is one important distinction: Hosts tend to have a shorter lifespan than physical servers directly utilized as stand-alone. This is due to several reasons, including the criticality of their role supporting several other virtual servers and the intensity and versatility of their utilization. According to the client's experience, average lifespan of hosts is three to four years. Therefore, the 'Hosts' co-flow has two stages fewer on the right end (Figure 17). After three years, hosts move to the stock of 'Hosts Y3Plus', where they reside for an (adjustable) average time of twelve more months.

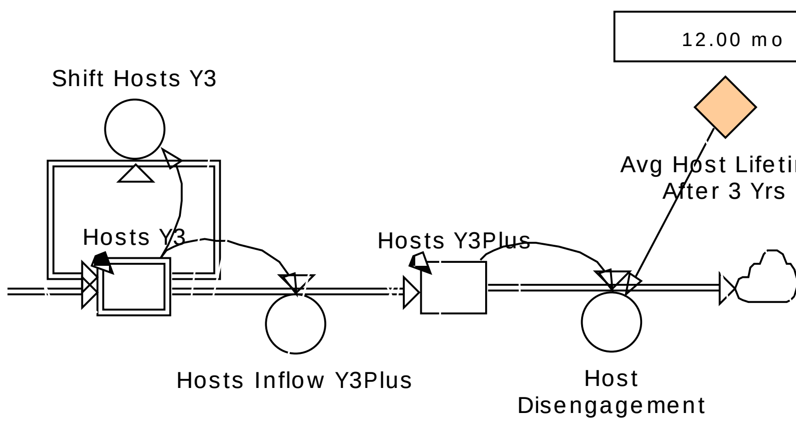


Figure 17 - Disengagement of Hosts

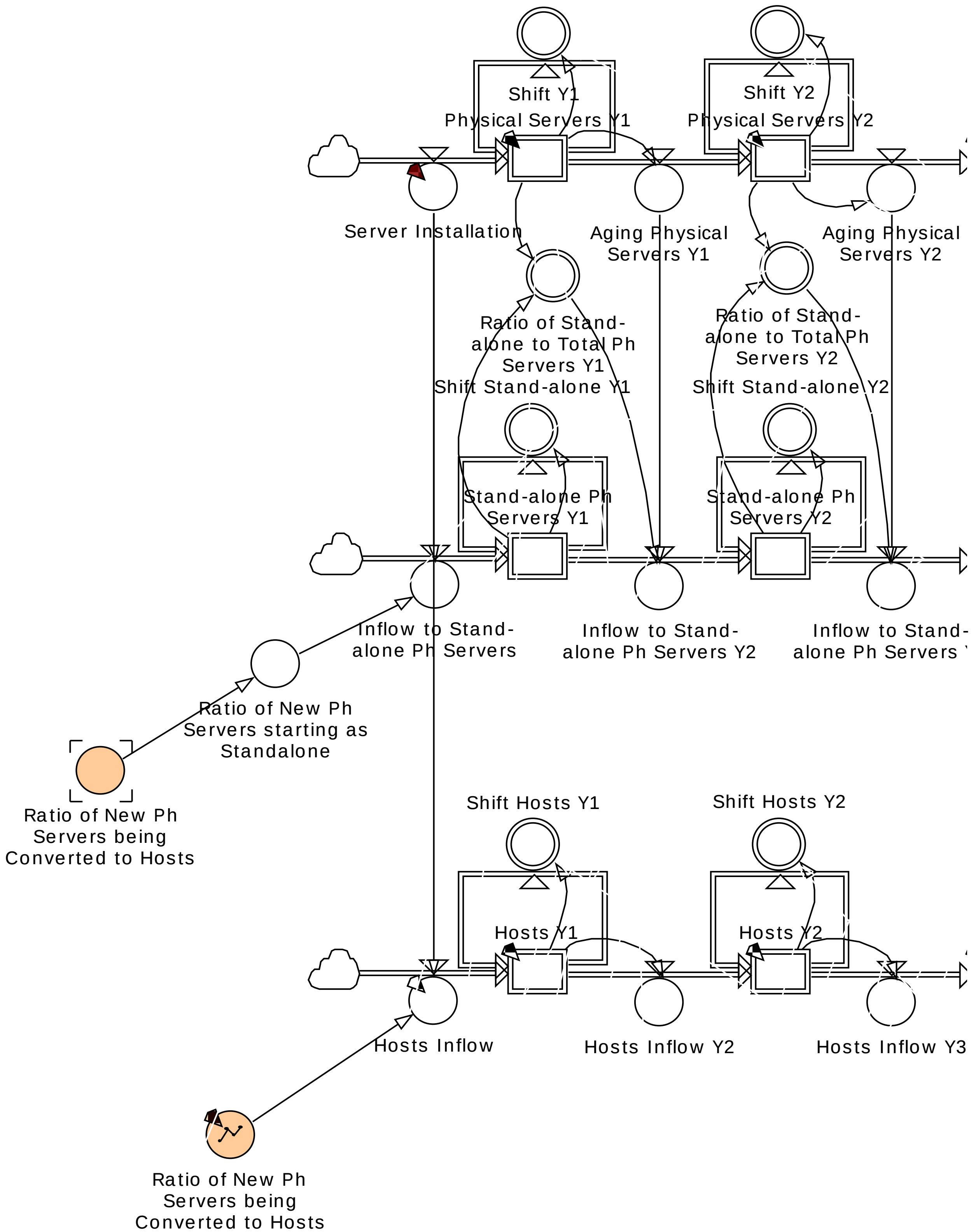


Figure 18 – Co-flows of Stand-alone Servers and Hosts

Parallel to the installation of physical servers to respond to user needs, virtual servers are added as well to satisfy a certain portion of user workloads. In recent years, there has been an accelerating growth in use of virtualization. "Recent advances in microprocessor technology and software have led to the increasing ability of commodity hardware to run applications within Virtual Machines (VMs) efficiently. VMs allow both the isolation of applications from the underlying hardware and other VMs, and the customization of the platform to suit the needs of the end-user (Buyyaa, Yeoa, Venugopala, Broberg, & Brandic, 2009, p. 600)." Virtual machines are not a new concept, but recent advances in hardware and software technology have brought virtualization to the forefront of IT management. Stability, cost savings, flexibility, and manageability are some of the reasons for the recent rise in virtualization (Daniels, 2009).

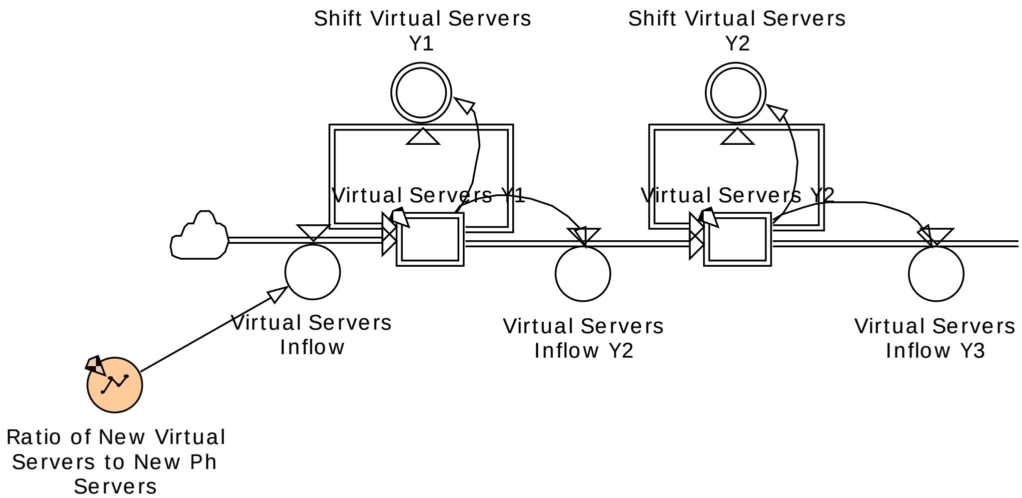


Figure 19 - Co-flow of Virtual Servers

Virtual servers constitute another pipeline in this sector of our model. In order to formulate the addition of virtual servers to ABC's assets, the 'Ratio of New Virtual Servers to New Ph Servers' (Figure 19) has been extracted from data. This ratio was obtained in the same way as previously mentioned ratios for hosts and stand-alones, by dividing the amount of virtual servers historically added each year by the amount of physical servers. This ratio is thereafter multiplied by the real-time flow of 'Server Installation' (Figure 14) to give the inflow to the aging chain of virtual servers. The inflow arrived at using this formulation replicates actual data for added virtual servers since 2007 pretty closely (Figure 20), empirically confirming this structural assumption.

In reality, this inflow of virtual servers is limited by the surplus capacity of our current number of hosts to support virtual machines. In this manner, adding new hosts to the fleet becomes a matter of weighing current capacity to accommodate more virtual servers against the need for more virtual servers in the near future, making it theoretically possible to 'endogenize' the inflow of hosts. However, as already mentioned, this link and the concept of number of virtual servers supported by a host is highly complex, involving several variables with difficult to quantify correlations, and is therefore not included in this thesis. Still, probably the most important predictor of this number is the number of virtual cores on a host server, which *can* be tracked using this model (see section 4.2.4). Furthermore, it is easily possible to at least track the ratio of 'virtual servers per host' for different age cohorts in this model.

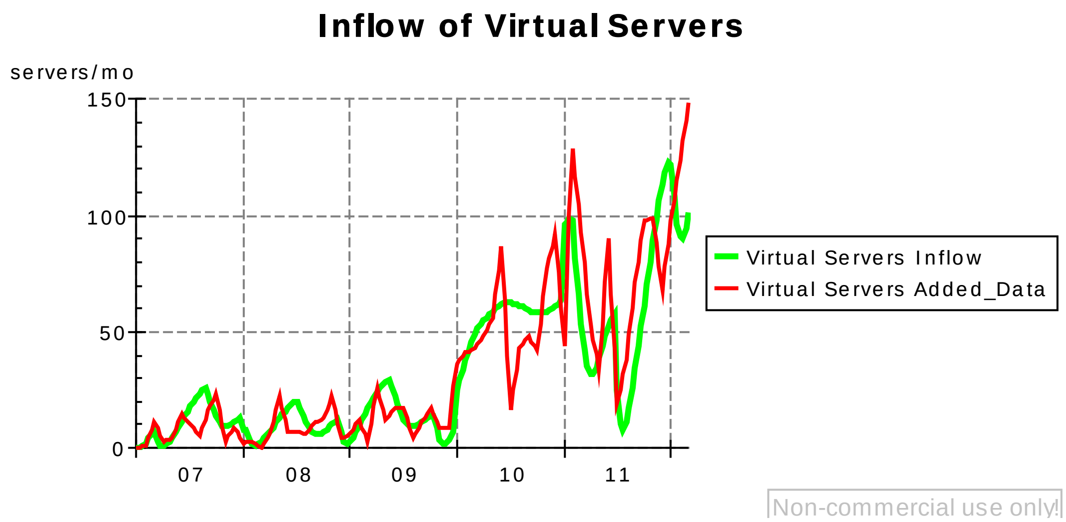


Figure 20 - Inflow of Virtual Servers, simulation (green) vs. data (red)

The outflow of virtual servers is modeled with a similar structure as for hosts: After the third year of their operation, virtual servers move to the final stock where they are depleted with a first-order delay. The length of this delay is varying and is estimated with help of reference mode comparison tests once the validity of the structure was established[1]. Total average estimated lifespan of virtual servers looks like the graph in Figure 21.

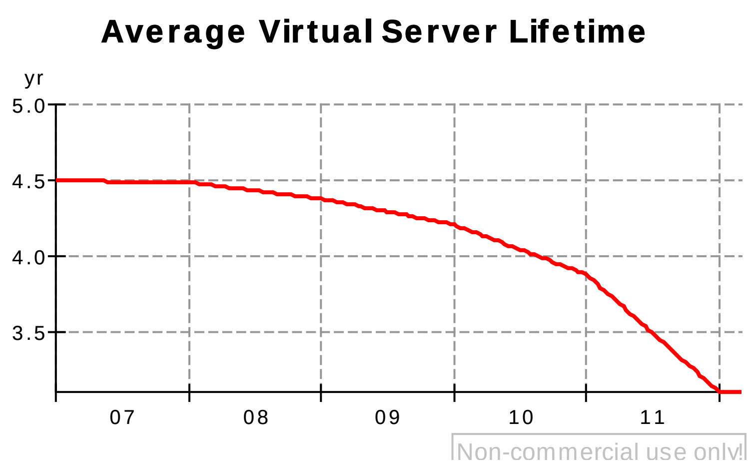


Figure 21 - Average Virtual Server Lifespan

Thus, according to the model, virtual machines have been dismounted increasingly sooner during the past five years.

3.2.4. CPUs, CPU Cores, and RAM

[1] In this report, where inputs to the model are estimated, the relevant reference mode comparison tests can be found in "model analysis" in Chapter Four.

Processing power, characterized by the number of CPUs on a physical server and the number of virtual cores on each CPU, and memory (RAM), as the major attributes of physical servers, are modeled as three separate co-flows.

The pipeline of CPUs is a straightforward co-flow where the input of 'Number of CPUs per New Server' is constantly set at two because since 2007 all physical servers purchased by ABC have had two CPUs on-board.

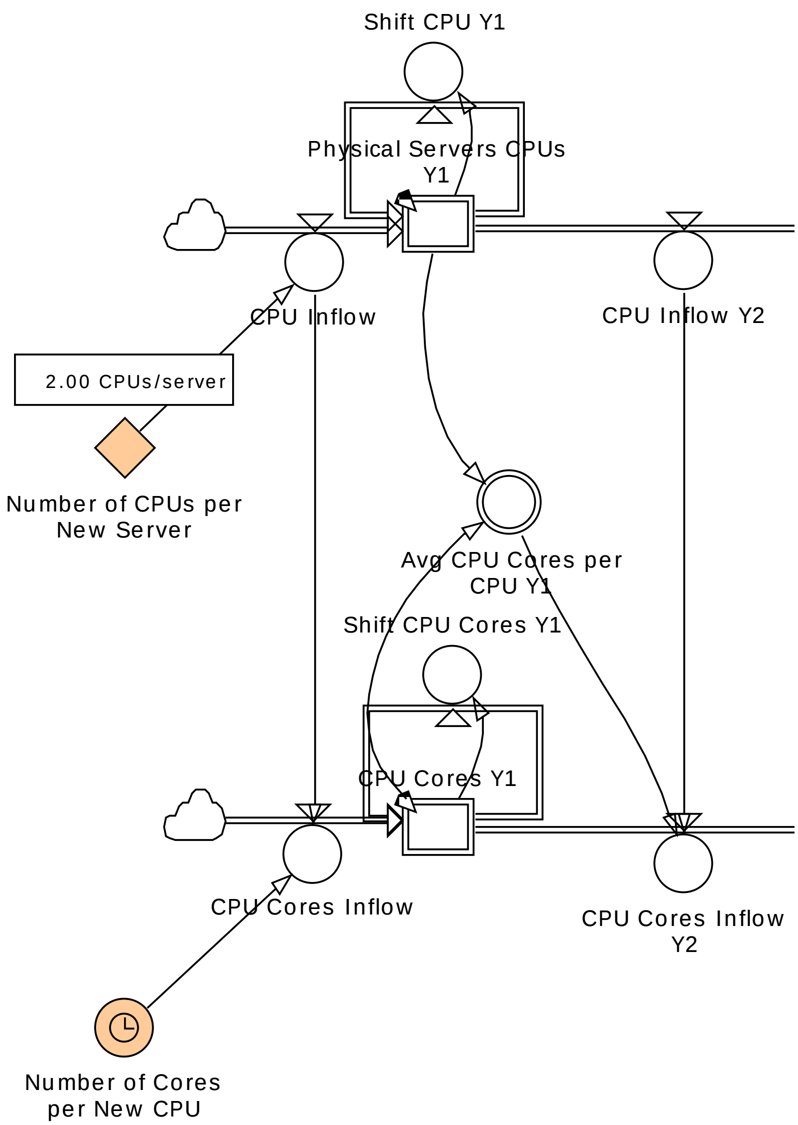


Figure 22 – Co-flows of CPUs & CPU Cores

The flow of CPU Cores is modeled as an ancillary co-flow of the CPU co-flow, rather than a co-flow of the physical servers pipeline. The input of 'Number of Cores per New CPU' signifies the growing number of virtual cores on each CPU for newly installed physical servers, and is shown in Figure 23. At the beginning of our timeframe, in 2007, purchased servers used to have CPUs with a single virtual core on them. This feature doubled once in 2007, and then again in 2008, and later reached six cores per CPU in

2009, at which level it has remained until today. Both the number of CPUs per new physical server and the number of cores per new CPU are seen to keep growing in the same step-wise fashion in the future.

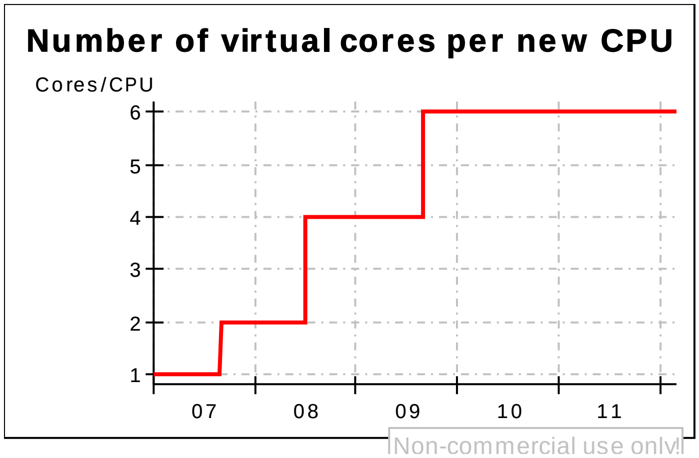


Figure 23 - Virtual cores per new CPU

In the model, a separate co-flow structure is included for tracking RAM per physical server for different age cohorts (Figure 24). This is an essential feature of servers influencing the number of end-users that it can serve as a stand-alone physical server and the number of virtual machines that it can host if it is converted to one. Unfortunately, there was no data provided by the client regarding RAM. Thus, this part of structure is not calibrated to real data and is not really used in model analysis. Nonetheless, it is kept there since it can quickly be calibrated once data becomes available.

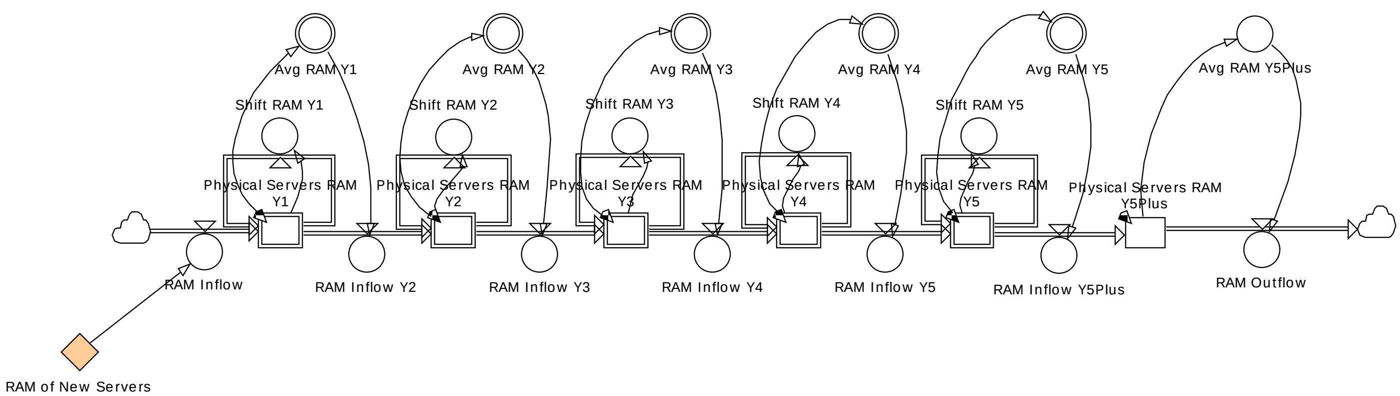


Figure 24 - Co-flow of RAM

3.2.5. Power Demand

Physical servers obviously require electricity. They need electricity not only for operation, but also for cooling. The amount of overhead power required, predominantly for cooling but also for lighting, switches, security, etc., is a feature of the data center. These overhead consumptions are normally included in energy cost measurements using a multiplier known as PUE (Power Usage Effectiveness). We are going to have a closer look at this concept in the 'Accounting & Finance' sector of the model (see section 3.6). For the time being, what we are interested in is being able to measure the operating power demand of our physical servers.

Physical servers operate at different levels of power usage depending on the purpose that they serve. Specifically, stand-alone physical servers are different from hosts in terms of average effective operating power. Hosts, being assigned relatively high loads, normally operate at a level close to their nominal power, while stand-alone servers take up a certain fraction of their nominal power. Since power usage profiles are different for these two main server categories, two separate co-flows are implemented for tracking their power requirements, which sum up to make total power needed to operate the data center's physical servers.

The variable 'Power Demand of New Servers' in Figure 26 is an input to these pipelines which was provided by the client (Figure 25). These power requirements depend on the type of server that is purchased. This input is multiplied by a factor of 99% for hosts, since they operate close to nominal power, and a factor of 90%, ten percent less than nominal, for stand-alone servers[1]. These factors are set as adjustable parameters in the model.

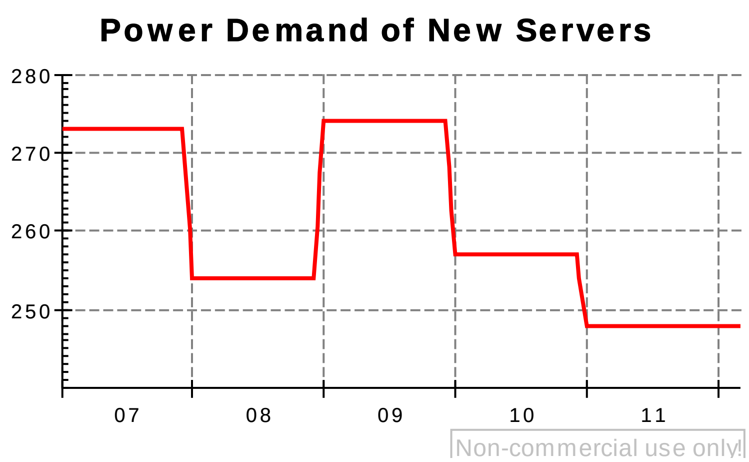


Figure 25 - Nominal Power Demand of New Physical servers

[1] Estimates provided by the client.

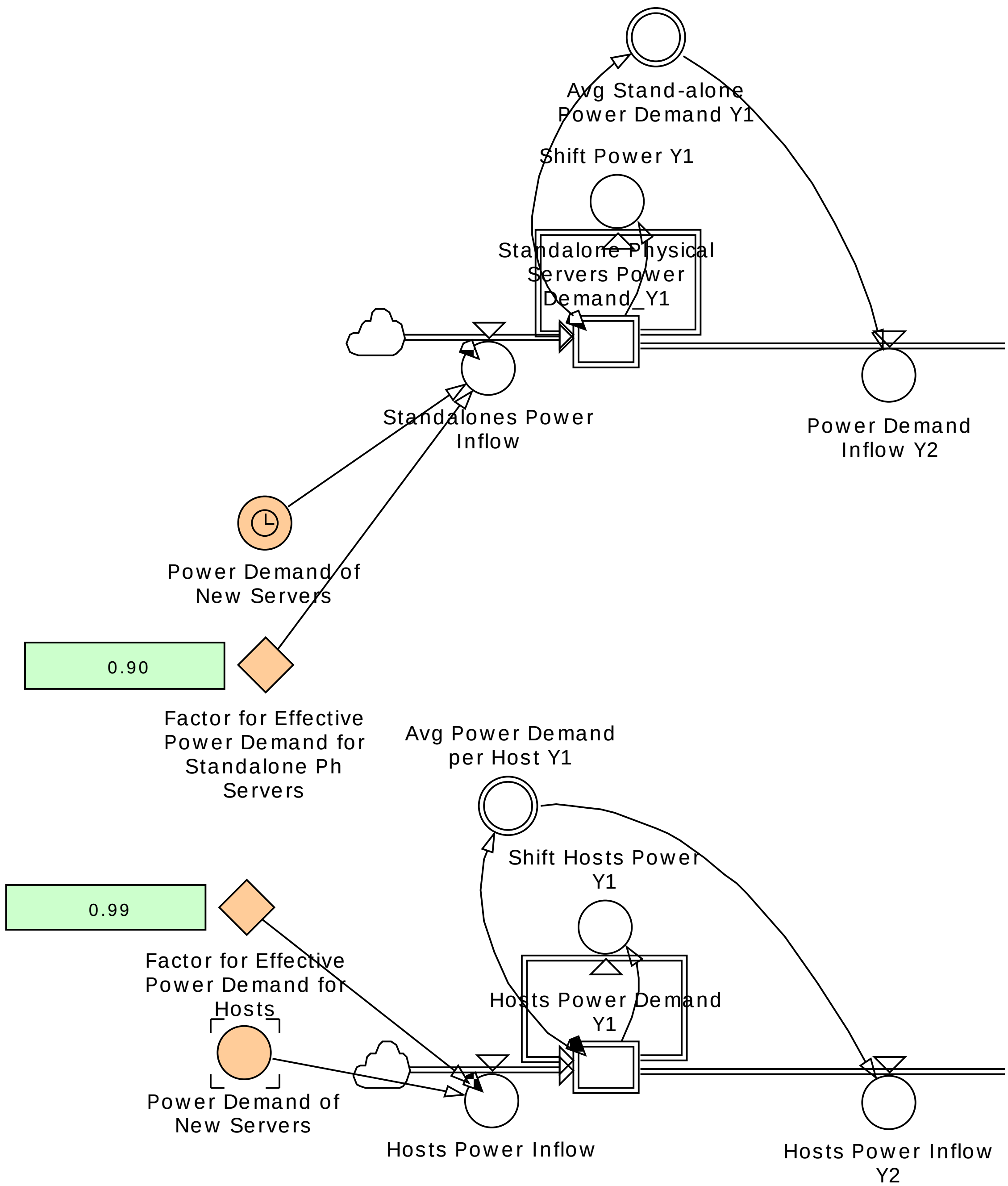


Figure 26 - Power co-flows

Thus, the inflows of power for the two pipelines in Figure 26 are formulated as the input graph in Figure 25 multiplied by the respective factor to convert nominal power demand into effective power demand,

and then multiplied by the inflow of the respective category of servers ('Inflow to Stand-alone Ph Servers or 'Hosts Inflow' as they appear in Figure 18)[1].

3.2.5 Price of Physical Servers

The last co-flow of this sector of the model is the pipeline of server prices. Need for this co-flow rose due to the necessity of tracking prices of different age cohorts of servers in order to calculate accounting depreciation. As mentioned earlier, in ABC's accounting system physical servers are depreciated during a period of 36 months. When prices of newly purchased servers vary through time, it is difficult to track average prices of different age cohorts without dynamic modeling.

The accounting aspect is further elaborated upon in the 'Accounting & Finance' sector (section 3.6.2). For now, the structure of the price pipeline is a basic co-flow structure with no complication (Figure 27). Historical 'Price of New Servers' is an external input to this pipeline (not yet provided by the client).

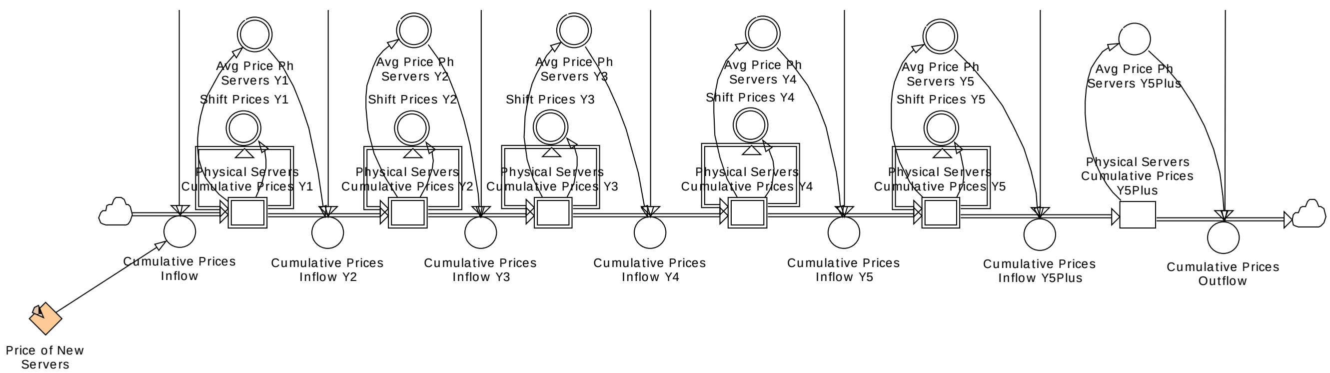


Figure 27 - Co-flow of Server Prices

3.3. The Spindles Sector

The sector for storage spindles is constructed similarly to the physical servers sector using co-flows of aging chains. However, this sector is smaller and less developed than the servers sector since it carries secondary importance compared to the servers sector. The sector consists of the main aging chain of storage spindles with three main attributes of spindles built in as co-flows: storage space, power, and price. Storage spindles have other important attributes besides storage, such as I/O capacity, and there are also different technologies associated with optimal use of spindles, such as spindle virtualization and RAID (Redundant Array of Independent Disks). However, in order to focus attention on building a good servers sector, the spindles sector of the model is kept simple and does not include these features.

3.3.1. Spindles

[1] Arrows from the inflows of the server pipelines into the inflows of power pipelines, and also arrows from stocks of servers into the 'average power per server' variables are not shown in Figure 26. In this document, arrows coming from other sections of the model are often not shown to avoid messy figures.

The aging chain of spindles is structured in exactly the same manner as the one for physical servers: Storage spindles are purchased to keep inventory at a certain desired level (with exactly the same structure for physical server acquisition, as described in section 3.2.2). Spindles are then installed based on user needs (see section 3.4.4). This installation rate is limited by the availability of surplus power in the data center (see section 3.5), and by availability of spindles in the inventory. Thereafter, they start going forward through successive age cohorts in the aging chain. They eventually reach the final stock, where they are depleted based on a first-order delay structure with (an estimated) time constant of two years, signifying an overall average lifetime of seven years.

3.3.2. Co-flows for Spindles

The three co-flows for storage spindles – power, storage, and prices – are modeled in almost exactly the same way as for physical servers. Therefore, we do not repeat the same descriptions over again, but we only point out the particularities of each co-flow:

- In terms of power demand, we have only one category of spindles, unlike physical servers where we distinguished between hosts and stand-alones. Furthermore, we don't have complete historical data regarding exact power demands of newly purchased spindles. We only know that spindles normally require around 18 watts of power each (which is smaller than the power demand of physical servers by more than one order of magnitude). Therefore, 'Power Demand of New Spindles' is taken as a constant and consequently, average power demand of different age cohorts of spindles should remain (and does remain) static.

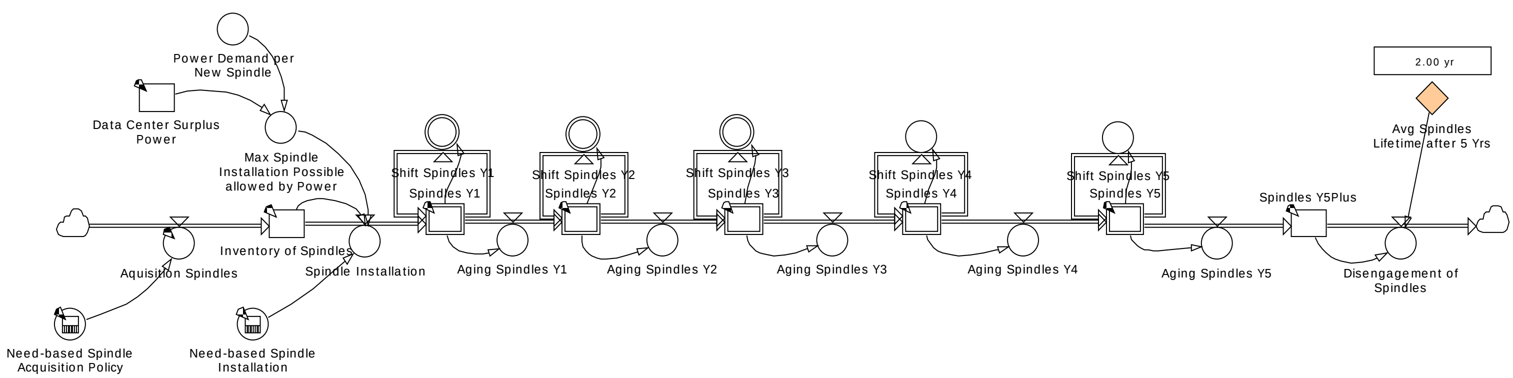


Figure 28 - Aging Chain of Storage Spindles

- As for the co-flow of storage capacity, this can be in a way regarded as the dual of the RAM co-flow for physical servers. In this case we have historical data for total storage, and thus we estimated 'Storage per New Spindle' using the empirically validated structure and historical data (4.1.8).
- Concerning prices, as of now we have received no historical data for prices of servers/spindles purchased, and 'Price per New Ph Server/Spindle' is a constant at the moment based on a wild assumption. Nevertheless, the structure is kept so that as soon as the data becomes available, it can easily be calibrated.

3.4. The Users Sector

The growth in the number of ABC's servers and storage spindles is driven by the growth in the number of end-users that the company is contractually obligated to serve and by the development in the requirements of these users at an aggregate level. Therefore, in order to be able to endogenously reproduce the growth in the ABC's assets we need to first capture the development in the company's user base. Each year, the company sets new targets for expanding its users based on overarching growth goals. Thereafter, new contracts are signed by ABC's sales department, while old contracts mature. Most clients renew their contracts, while only a small fraction leave the client base. These dynamics change the user base and these changes bring along changes in aggregate user requirements in terms of capacity, and also changes in total recurring revenues. All of this is the subject of the 'Users Sector' of the model being described in this section of the thesis.

ABC serves two different categories of users as mentioned before: 'full desktop' (FD) and 'distributed services' (DS) users. FD users have (roughly three times) higher capacity requirements, but the revenue that they generate is also much higher. While until 2011 the company's user base consisted of only FD users, since then the company has moved towards expanding its target market towards the DS segment of the market as well. In the model, we have constructed two identical structures with different parameter settings for the two different user types. Therefore, in the following model description we describe the structure for FD users only.

3.4.1. The User Base

The inflow to the stock of users is new users who are being signed up, while the outflows are on the one hand those users coming from a client who does not renew once the old contract expires, and on the other hand the users coming from a client which, for some reason, goes out of business. This latter outflow can be relatively significant in times of financial crisis.

In the model (Figure 29), 'Target for New Users' (left-most variable) comes from the Growth Sector (3.4.2). This could by itself be the inflow that is gradually fed to the stock, but as we know, if we ignore the expected outflow within our timeframe, we will encounter a steady-state error which constantly keeps a discrepancy between desired and actual states. For this reason, to the 'Target for New Users' we need to add the number of users that have flowed out of the stock during the previous year. This amount is represented by the variable 'DELAYINF_Ended Users of Last Year'. Since no information is available about this outflow for the year before the start of our simulation, we add an estimate of total outflow of users in 2006 to be added to the target for new users for 2007 (the variable 'Ended Users of 1st Year'). Thus, the inflow of users is formulated as the aforementioned sum being fed into the stock using a first order delay with a time constant of twelve months, the period within which a year's target is to be reached.

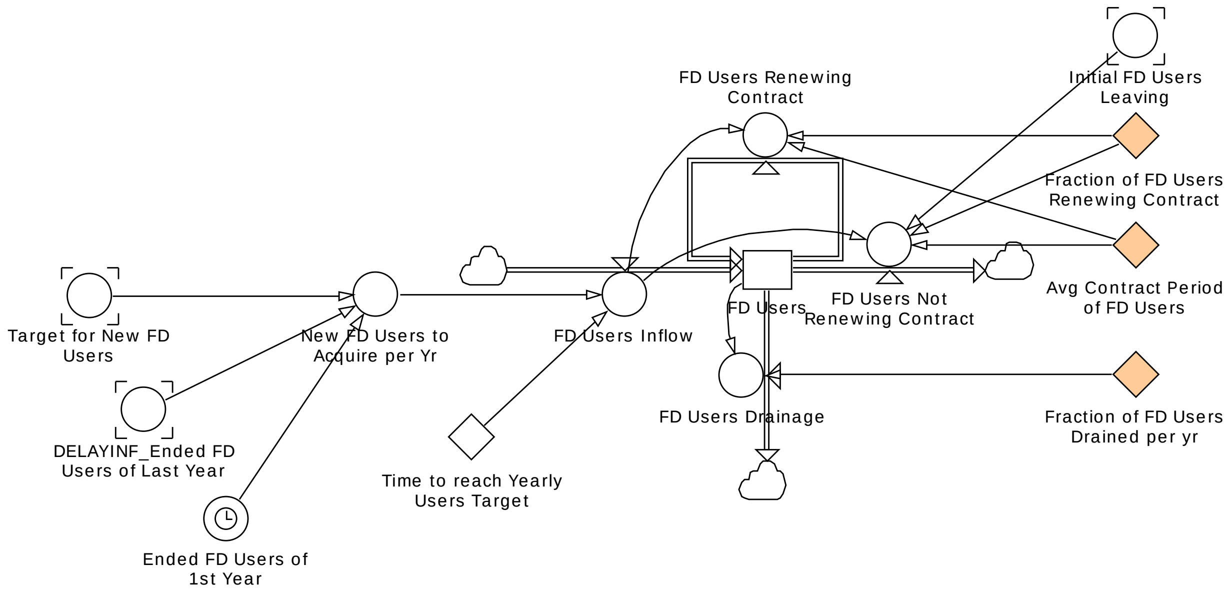


Figure 29 - The User Base

In this model, no market restrictions are imposed on the inflow of users. In other words, the company is assumed to achieve whatever goal that it sets for itself in terms of growth in the user base. This is due to the fact that the client believed that the market has so much free growth potential that, at least for the time being, this limitation is not a major point of concern. Moreover, our contact person in ABC was the company's Chief Technical Officer whose information was more complete in technical and operational aspects than in the market area. A more comprehensive model in the market area should incorporate concepts such as the stock of 'suspects', 'prospects', conversion rates from suspects to prospects and from prospects to contracts, contracts with different sizes in terms of number of users, etc. These details lie outside the scope of this thesis, and can present promising areas for future work.

Once the maturity dates for client contracts are reached, most clients (roughly 95%) renew their contract with ABC. This process is captured by the 'Users Renewing Contract' rate out of, and back into, the stock. In practice, this flow does not make any dynamic difference in behavior since whatever goes out of the stock instantly flows in. Nevertheless, the flow is included to make sure that the structure visually and conceptually reflects reality as closely as possible.

Roughly five percent of users leave the user-base once the client contract maturity date has been reached. This outflow is represented by 'Users Not Renewing Contract' and is formulated as a high-order delay of the inflow of users times the fraction of users not renewing contracts, with a delay time equal to average contract length (45 months). Since this formulation is based on the inflow, it does not deal with the initial user base already in the stock at 2007. Therefore, it was necessary to separately add to this outflow the outflow of those initial users which leave the user base every year. This is captured by adding the variable 'Initial Users Leaving' to this outflow.

The other outflow 'Users Drainage', representing the risk of default for a small fraction of ABC's clients, has a simpler formulation. This rate is an estimated two percent per year, simply multiplied by the stock level.

3.4.2. Growth in User Base

The growth engine in this system is driven by an annual growth target in the company's user base. At the first step, real historical annual growth rate is derived from the data for number of users: the variable 'User Growth Derived from Data' (Figure 30).

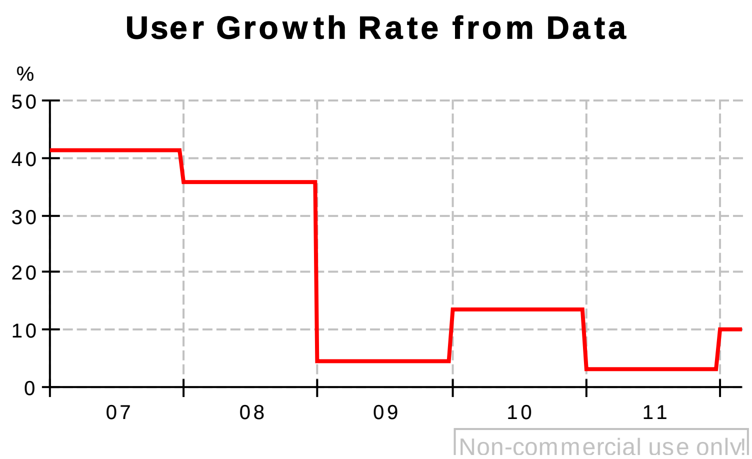


Figure 30 - User Growth Rate derived from Data

Secondly, this discrete year-by-year growth is translated into the equivalent continuous compounding exponential growth rate, to fit the almost-continuous (very small timestep) simulation carried out in the simulation software. As we know, for a nominal exponential growth rate N, effective growth rate E with continuous compounding is obtained with the following formula:

$$E = \lim_{dt \to 0} \left(1 + \frac{N}{dt}\right)^{dt} = e^{N} - 1$$

Therefore, to obtain the nominal growth rate for a desired effective growth rate, we have:

$$N = \ln(E + 1)$$

Thus, the variable 'Target for Percentage of Annual Growth' (Figure 31) is simply the natural logarithm of the effective growth rate obtained from data plus one. Next step is to multiply this target growth rate by the current 'Total Users' to obtain the target for the number of new users that need to join the user-base within the next year to reach the desired growth rate: the variable 'Target for Total New Users'. Finally, all we need to do is to translate this target into two separate targets for our two different user categories, FD & DS. This fraction is derived from data for the past and can be set to different values as a market policy lever for the future.

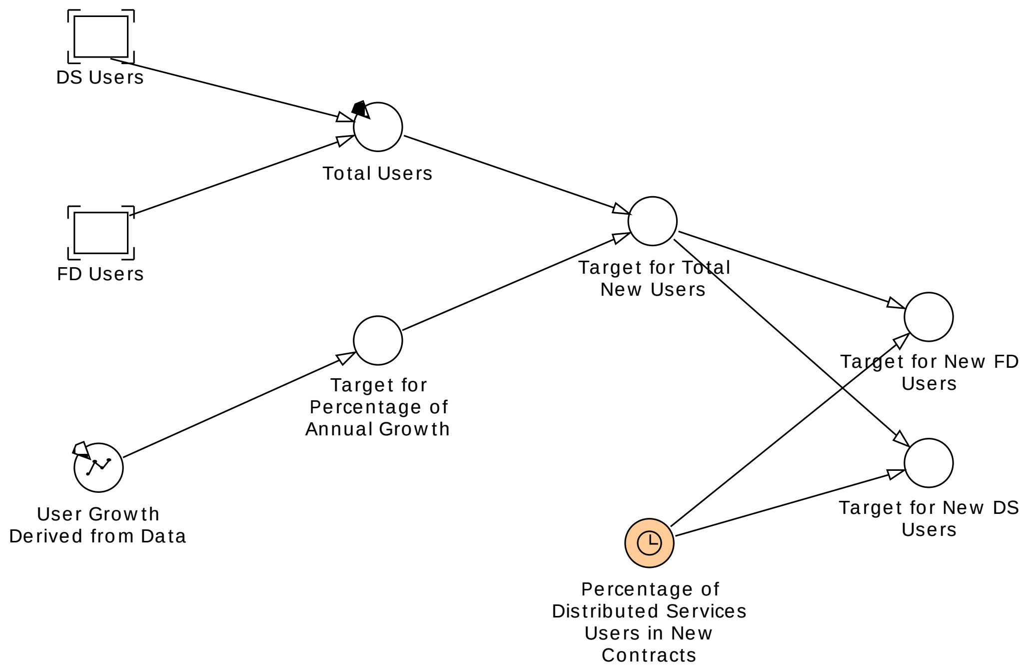


Figure 31 - Growth in User Base

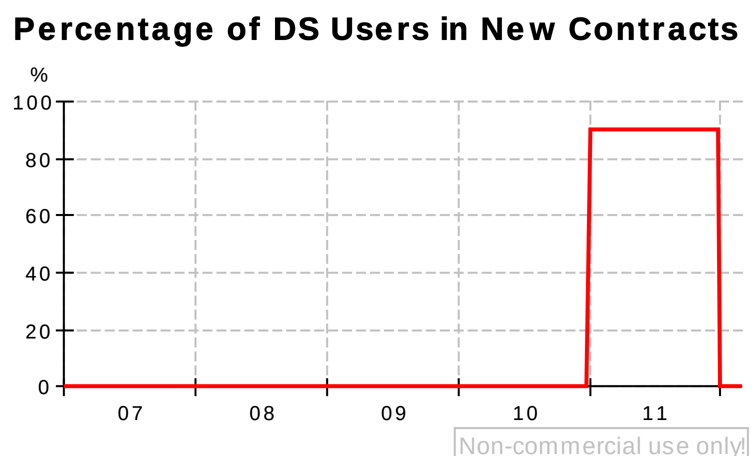


Figure 32 - Share of DS Users in New Contracts every year

ABC used to be targeting exclusively the 'full-desktop' segment of the market. This policy took a U-turn in 2011 where users joining from new contracts were almost wholly distributed-services users. ABC considers this market policy to be a major point of investigation for the future. With this parameter included in the model, the client can test different policies related to this, to observe the simulated effect of a change in policy on future development of assets and revenues.

3.4.3. User Requirements

The three major requirements of users - processing power, RAM, and storage space – are modeled as co-flows of the in- and outflows of users. The structure for user processor requirements can be seen in Figure 33 as an example.

The input of ‘Processor Requirements per New User’ to this co-flow comes from the intermediary variable ‘Users per Logical Server_New Users’, which is the same variable reversed. This reversed form is more intuitive to the client’s mind, possibly because each logical server serves several users while the amount of logical server per user is a fraction, and whole numbers are probably easier to think of than fractions. Therefore, we use “Users per Logical Server” for new users as the input. Once we were able to reproduce the growing trend in the number of users successfully, this variable is estimated so as to produce a development in aggregate server requirements which would stimulate the addition of new servers to the fleet in a way that would eventually replicate the historical behavior in server numbers (physical/virtual) closely. This estimate is shown in the graph in Figure 33. It can be seen that ‘Users per Logical Server’ has been steadily and significantly going down, from 18 to 1.6 users per server. This is partly due to the fact that ABC has deliberately improved its customer service level by assigning fewer and fewer users to share one logical server, and partly due to the fact that the average lifetime of physical servers has been increasing, causing higher accumulation in the stock of physical servers and pushing down the number of users per server.

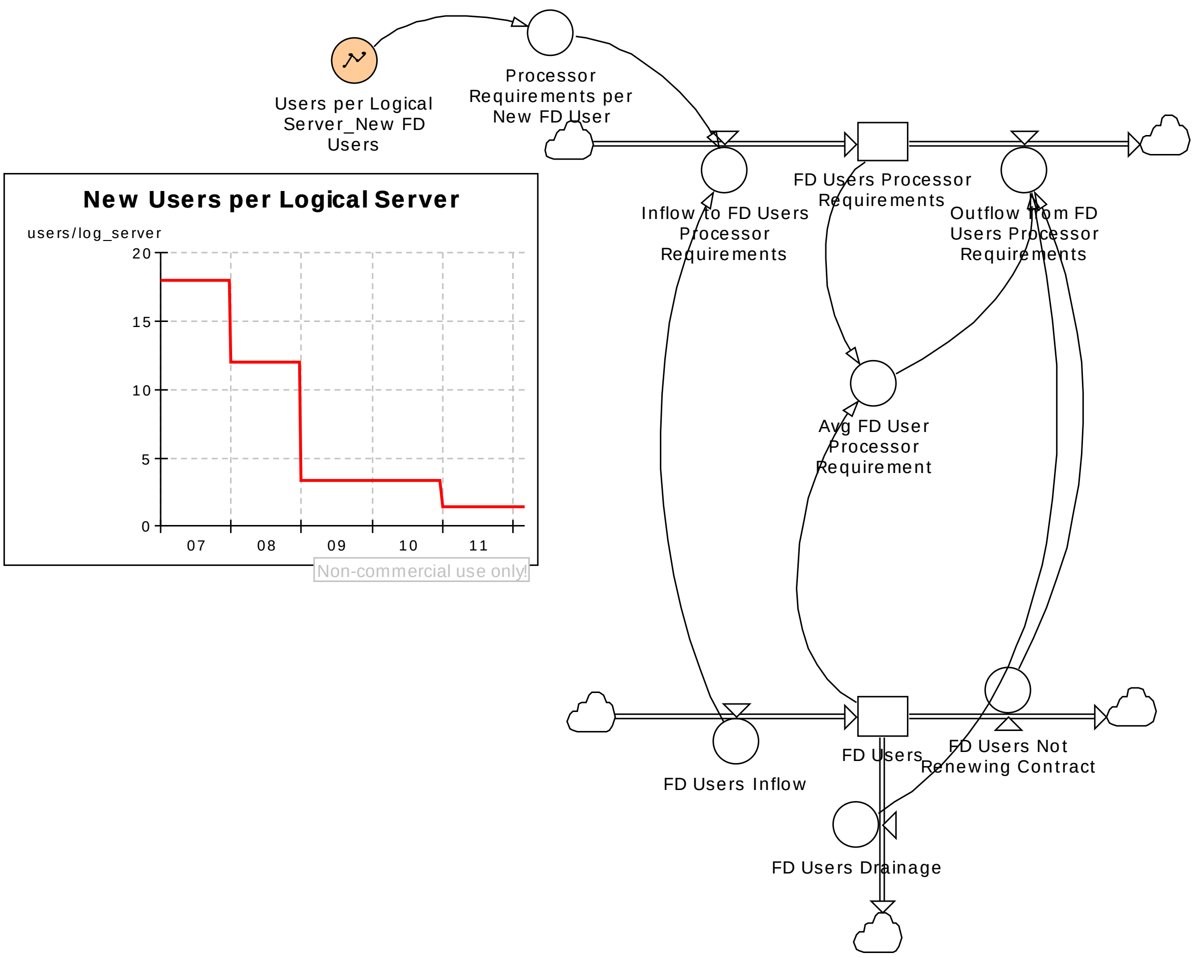
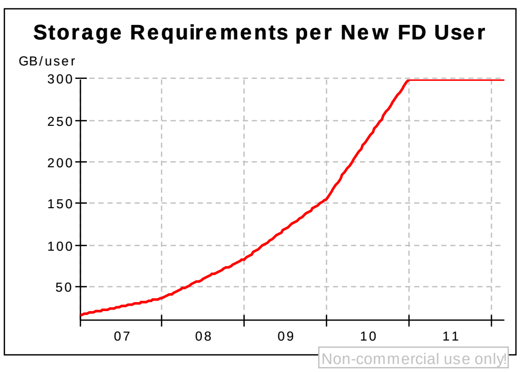


Figure 33 - User Processor Requirements

Thereafter, 'Users per Logical Server_New Users' is reversed and then multiplied by 'Users Inflow' to give the 'Inflow to Users Processor requirements'. The effects of the two outflows from the stock of users are combined into a single 'Outflow from Users Processor Requirements' by multiplying average user processor requirement by the sum of the two user outflows.

There are two co-flows structured in a similar way for storage and RAM requirements of users. The RAM co-flow is not calibrated since, as noted earlier, no data was made available regarding RAM. As for the storage requirements co-flow, the input of 'Storage per New User' was calibrated so as to reproduce the correct growing trend in the number of storage spindles given the development in 'storage capacity per spindle' according to the data. Below (Figure 34), we can see the structure and the input for full-desktop users. The input of storage requirements per new user is set to be three times smaller for distributed-services users, according to the client's estimate.



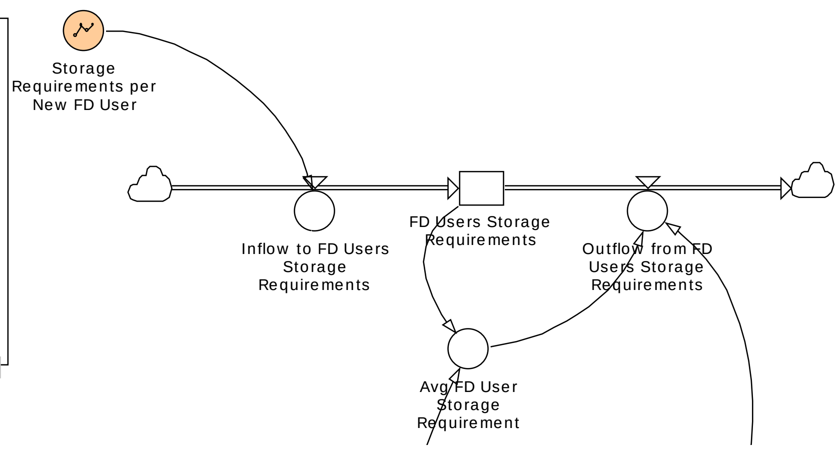


Figure 34 - The co-flow of Storage Requirements

3.4.4. Need-based Server/Spindle Installation

As pointed out before, while describing the physical servers aging chain and the spindles aging chain, the installation of physical servers and spindles is driven by the accumulating processor/storage requirements of users. The structure responsible for this relationship is described in this section. Here, we explain only 'Need-based Server Installation'. 'Need-based Spindles Installation' follows the same rules.

In reality, the need for installing new physical servers rises either when the data center needs more processing power or when it needs more RAM to serve its user base. In the model, however, this installation rate is dependent only on processor requirements because no data was provided by the client considering RAM. Processor requirements, as seen in the previous section, are measured in number of logical servers. Initially, processor requirements of the two user categories are added up to make 'Total Required Logical Server' (Figure 35). A gap will exist between this desired state and the current level of 'Total Logical Servers'. In the structure of the Servers sector of the model, we saw that when new physical servers are installed, a fraction of them are converted to hosts for virtual machines, and furthermore a number of virtual servers are also installed in parallel with new physical servers. Therefore, to formulate how a certain gap is closed in the number of logical servers, we need to use a bit of algebra:

New_L = New_Ph + New_V – New_H ➔ New_Ph = New_L / [1 + (New_V/New_Ph) – (New_H/New_Ph)]

(L: Logical Server; Ph: Physical Server; V: Virtual Server; H: Host)

Thus, to obtain the number of required new physical servers to be installed, we divide the gap in the number of logical servers by [1+'Ratio of New Virtual Servers to New Ph Servers'-'Ratio of New Ph Servers being Converted to Hosts']. The next variable is simply the non-negative version of this number, in order to prevent the model from automatically taking out servers from the fleet in case the desired number of servers is fewer than the actual number. The final resulting installation rate is a third-order material delay of the required physical servers to be installed, with a delay time of one month, representing the time it takes to set up and add a physical server to the fleet.

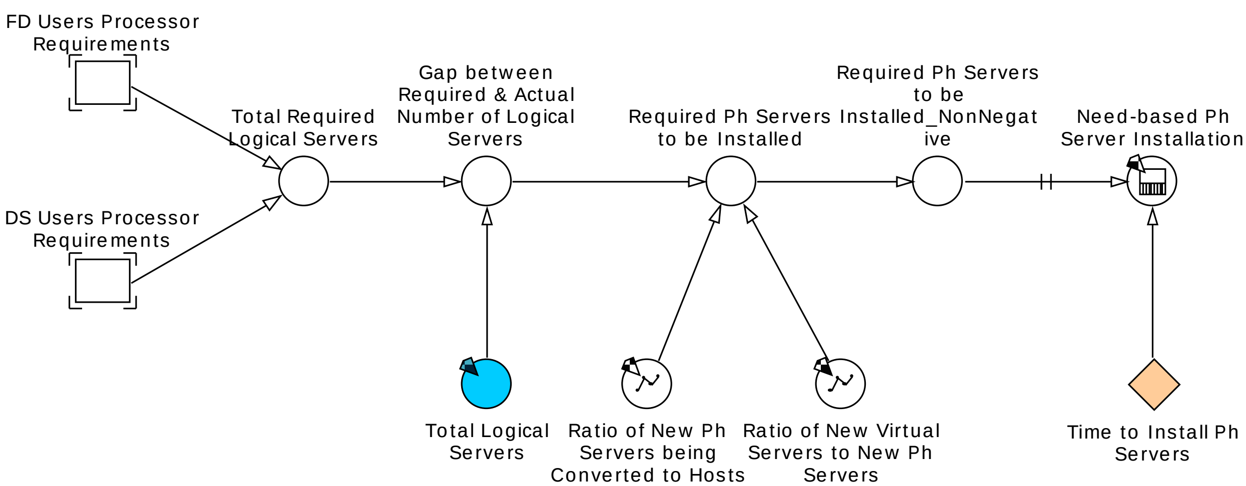


Figure 35 - Need-based Server Installation

'Need-based Spindle Installation' is obtained with a similar structure, except in this case calculating the number of spindles to be installed based on the gap in required storage capacity is more straightforward: We just divide the gap in storage by the 'Storage per New Spindle' to get the required number of spindles to be installed.

It is worth pointing out once again that in the model, the need-based installation rate of physical servers/spindles is not necessarily equal to the actual installation rate due to the fact the actual installation is limited by the availability of physical servers/spindles in the inventory, by the availability of installation engineers (HR), and also by the availability of surplus power capacity in the data center.

3.4.5. Revenues

A similar co-flow is put into place to capture the stock of recurring monthly revenues, which is fed by new users as they join ABC's user base and is depleted by those users leaving the user-base (Figure 36).

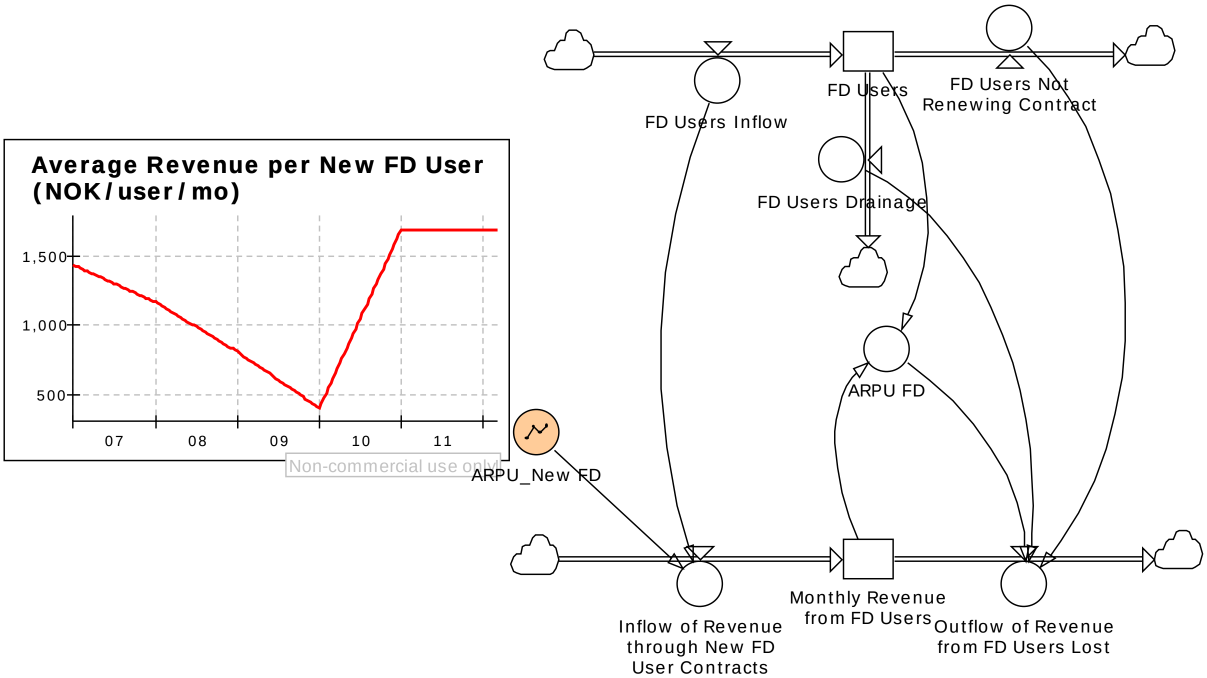


Figure 36 - Revenues from FD users

Historical data for the input variable 'average revenue per new user' (ARPU) was not available. Nevertheless, we had the overall monthly revenue data. Therefore, we were able to estimate the input through simulation, trying to get an acceptable fit with historical behavior of total revenue. This estimate is shown in the graph in Figure 36. As can be observed, price per user for new contracts should have experienced a major decline, up until around 2010 when it rapidly started recuperating previous loss (further discussed in 4.2.7.

3.5. The Data Center Sector

This relatively small sector (Figure 37) captures the concept of the power capacity of the data center under study, with one stock together with its in- and outflows. It should be noted that space is another main feature in data centers. However, as pointed out earlier, according to the client's experience, space is almost never a limiting factor in data center settings. There is normally plenty of space left when the cap of supported power is reached, especially with the trend towards higher density rack configurations in data centers. Therefore, modeling only the power feature of the data center in question was decided to sufficiently serve our purpose.

The single central stock of this sector, 'Data Center Surplus Power', is initiated by taking the 'Total Initial Data Center Power' (left-most variable) and subtracting from it the initial 'Total Power Demand of Servers & Spindles' (right-most variable). Surplus capacity flows out of this stock when newly installed physical servers or spindles take up more of the available power. These are the three outflows to the right of Figure 37. These outflows are the same as the inflows of power to the pipelines of stand-alone physical servers, hosts, or spindles already described in 3.2.5.

Eventually, when physical servers/spindles are disengaged, they free up the amount of power capacity that they were engaging. This released capacity flows back into the stock of Data Center Surplus Power (inflows on the left side). These inflows are the same as outflows of the power pipelines of physical servers/spindles.

Finally, ABC can always upgrade its data center in terms of supported power capacity. Indeed, the company has already done so once in the beginning of 2011 by doubling the data center's total power capacity from a total of one megawatt to two megawatts. This was done by installing new power lines in the ceiling of the data center. This possibility is captured by the inflow of 'Upgrade in Data Center Power'. This inflow can be used for potential similar upgrades in the future.

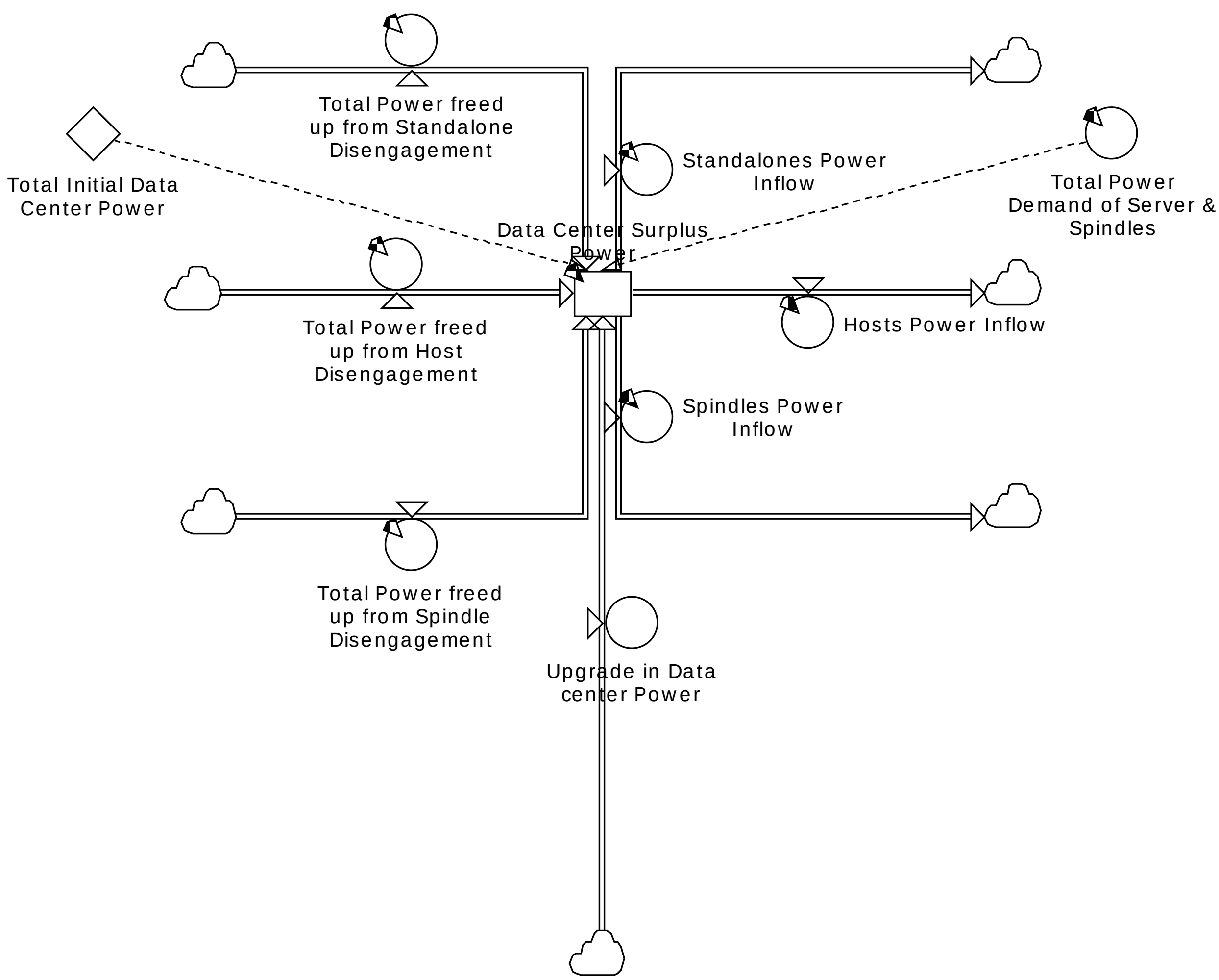


Figure 37 - The Data Center Sector

3.6. The Accounting & Finance Sector

Within this model, we have provisioned a small unsophisticated 'Accounting & Finance' sector, in order to be able to track the financial implications of the model's internal dynamics. In this sector, we have only included accounting and financial measures that could be drawn from endogenously produced model variables. In other words, we have not expanded the financial sector into new directions just to have a more realistic or sophisticated finance sector. This sector has two central goals: to calculate net income and to yield net cash flow.

3.6.1. Net Income and Expenses

Net income before taxes is simply total revenue minus total expenses. We already know total revenue from adding up revenue from FD users and revenue from DS users (section 3.4.5). Now we need to calculate total expenses. The types of expenses included in this model are: energy, rent, human resources, overhead, depreciation, and financial expenses.

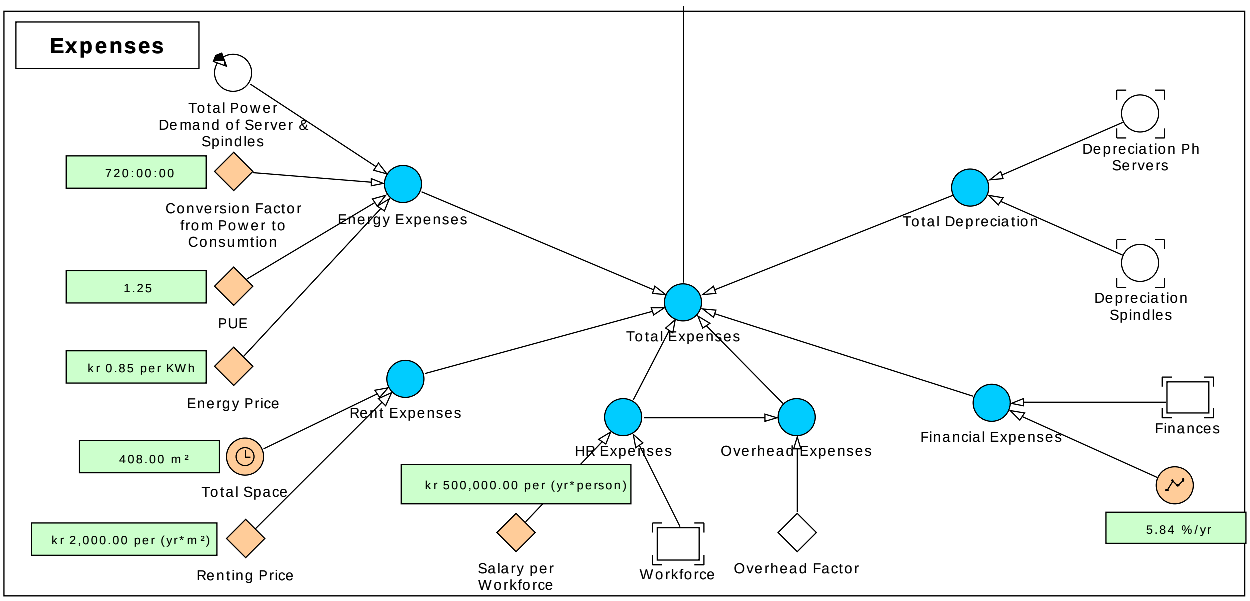


Figure 38 – Expenses

Let us begin with energy expenses. To obtain energy expenses we first need to sum up the effective power requirements of our physical servers and storage spindles: 'Total Power Demand of Servers & Spindles'. Secondly, this sum should be multiplied by a conversion factor to translate the concept of power into the concept of energy consumption, kilowatts into kilowatt-hours. This 'Conversion Factor from Power to Consumption' is simply the number of hours in a month: 720 hours. Thirdly, we must multiply this effective operational power consumption by the PUE factor. Power usage effectiveness (PUE) is a metric used to determine the energy efficiency of a data center. This measure is determined by dividing the amount of power entering a data center by the power used to run the computer infrastructure within it. PUE is therefore expressed as a ratio, with overall efficiency improving as the quotient decreases toward 1. PUE was created by the Green Grid, an industry group focused on data center energy efficiency (Green Grid Data Center Power Efficiency Metrics: PUE and DCiE, 2008). For the ABC's data center in this project, PUE is around 1.25. Lastly, we multiply this consumption by electricity price.

Rent expenses are simply the product of data center space by renting price. The data center in question used to expand over an area of 339 square meters, until the 4th quarter of 2011 when an additional area of 69 square meters was added. This step-up has been included in the 'Total Space' variable.
Next, we have HR (human resources) expenses, which is an average yearly salary times the number of operational employees. 'Operational Workforce' comes from a small HR sector, described later in 3.6.3. A constant 35% is added as 'Overhead Expenses' to this amount, as an estimate for costs related to administration and finance employees, office rental, office equipment, insurance, etc.
Financial expenses are calculated as the product of the stock of 'Finances' by the floating interest rate. The stock of Finances is imported from a small Finances sector described later in 3.6.4. Interest rate for ABC's finances is normally NIBOR (Norwegian InterBank Offered Rate) plus 0.5%. This data is imported from the website of Norges Bank (Norwegian central bank).
The last accounting cost element is depreciation, which is obtained as described in the following sector.

3.6.2. Depreciation

Previously, we built structure for tracking prices of different age cohorts of physical servers and storage spindles in the Servers Sector and the Spindles Sector. Here, we are going to use this provision to obtain accounting depreciation.

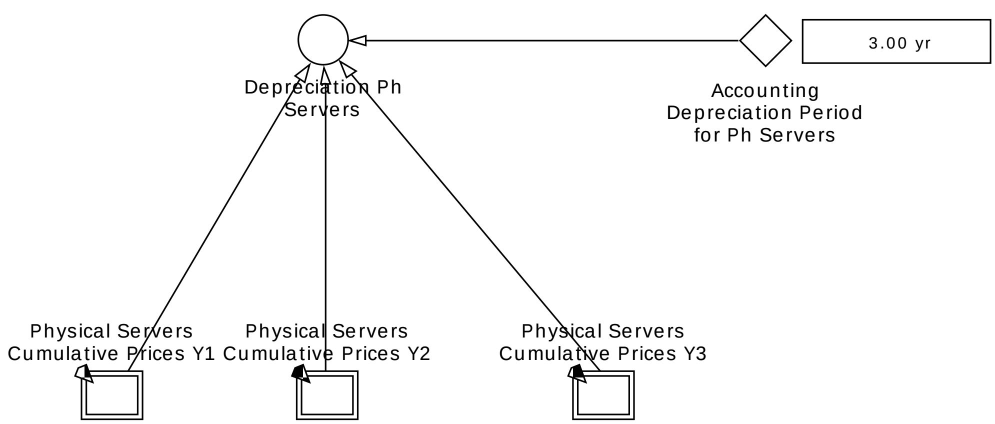


Figure 39 - Accounting Depreciation

In ABC, physical servers and spindles are depreciated in accounting books over a period of 36 months. To calculate depreciation we need to simply sum up the cumulative prices of the physical servers in the first three age cohorts (using the array sum function), reflecting the 36-month period, and divide the total price by this period. In this way we naturally obtain a number with a money-over-time dimension, which is the correct dimension for expenses.

Same structure is built for spindles. Total depreciation is the sum of physical servers' and spindles' depreciation.

3.6.3. Human Resources

Since HR expenses are a major cost driver, we recognized the need to include an HR sector, although highly simplified. This sector contains a single stock of 'Operational Workforce', with a first-order adjustment as inflow of hiring. It is assumed that operational employees are needed for server installations only, and that each employee can perform a certain number (in this case 15) of physical server installations in one month. It should be taken into account that the company would not desire any more installation engineers than the level of server installations that the power capacity of the data center permits. Thus, 'Desired Workforce' is obtained by dividing the minimum of 'Need-based Ph Server Installation' and 'Max Server Installation Possible allowed by Power' by 'Number of Ph Server Installations per Workforce per Month'. Thereafter, the stock adjusts itself to this desired level with a first-order delay, with a 'Time to Hire' of six months.

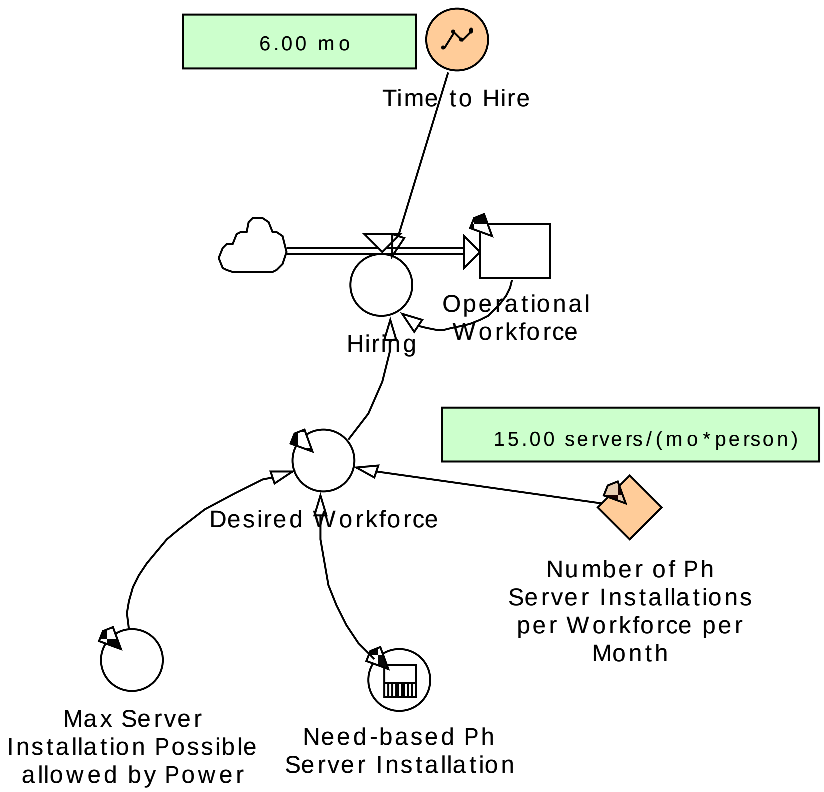


Figure 40 - Human Resources

As can be observed, this sector is very crude at this stage. To begin with, only physical server installations are taken as a proxy for operational HR needs. Other things such as virtual server installations, spindle installations, or server maintenance are not considered. Moreover, the only rate connected to the stock of workforce is hiring. Retirement, quitting, or firing rates are not modeled. Further, the distinction between rookies and experienced employees is not made.

Nevertheless, even at this level of incompleteness, this structure can give an estimate of HR expenses which is more reasonable than including this cost as a constant or ignoring it altogether.

3.6.4. Finances

In the following simple 'Finances' sector (Figure 41), financial needs for purchasing physical servers and spindles are taken into account. As ABC is invoiced for its purchases, the expenses first flow to a stock of credit, and are then converted to finances. The invoicing flow equals the acquisition rate of physical servers/spindles times their prices. ABC has a credit period of normally one month with its vendors, after which the company needs to pay the invoice amounts through other means of financing, which can be of various natures: long- or short-term loans, ploughback of net income, etc. One single stock is boldly put to capture these different means, with an average maturation time of three years, after which ABC has to pay back these loans out of cash. This is one element of cash outflow, modeled in the next section. The stock of Finances is assumed to be the origin of ABC's financial expenses, previously modeled in 3.6.1.

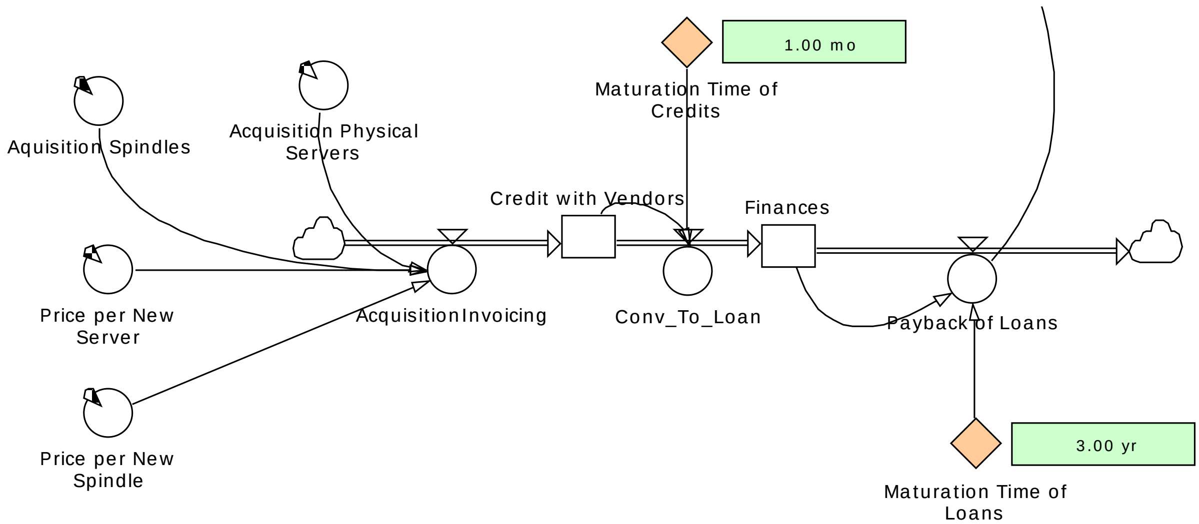


Figure 41 – Finances

3.6.5. Cash Flow

Probably the most vital financial indicator for a business is net cash flow. This concept, while fairly complicated in accounting, becomes very simple to visualize in System Dynamics terms: It is simply cash inflow minus cash outflow.

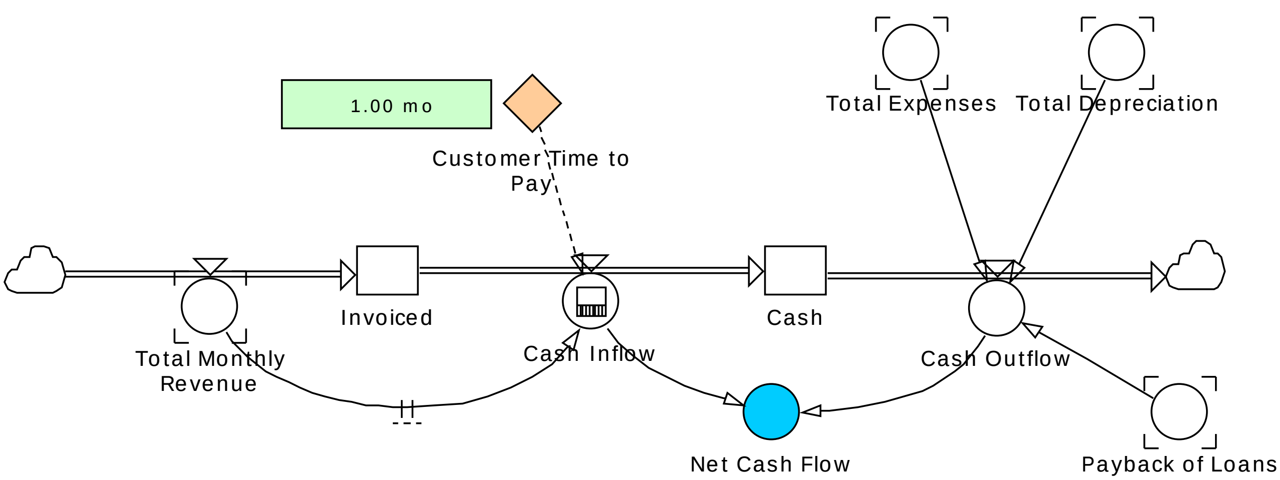


Figure 42 - Cash Flow

Cash inflow comes from customer payments. Customer payments is an outflow of the ‘Invoiced’ stock, which is fed by ‘Total Monthly Revenue’. This latter rate is converted to cash inflow with a pipeline delay of ‘Customer Time to Pay’, one month. ‘Cash Outflow’ consists of ‘Total Expenses’ plus ‘Payback of Loans’ minus ‘Total Depreciation’. Depreciation is taken into accounting when calculating accounting expenses, but is actually not a cash outflow. For this reason, this part is taken out of total expenses when calculating cash outflow.

Chapter Four - Model Analysis & Validation

Before a model can be used for policy analysis, enough confidence should be built in its validity, coherency, realisticness, and rigor. According to Barlas (1996), system dynamics validation tests fall under three categories: direct structure tests, structure-oriented behavior tests, and behavior pattern tests. “Direct structure tests assess the validity of the model structure, by direct comparison with knowledge about real system structure. (Barlas, 1996, p. 189)” These tests can be carried out both empirically and theoretically. Since this thesis has been a practical project of modeling a real business, the structural conceptualizations and relationships have been validated empirically, through talks and modeling sessions with the client. In addition, all variable dimensions and units have been chosen to be consistent with reality and with each other; no fudge factors have been added to enforce unit consistency. “Structure-oriented behavior tests assess the validity of the structure indirectly, by applying certain behavior tests on model-generated behavior patterns. (Barlas, 1996, p. 191)” This category of tests includes extreme condition and behavior sensitivity tests. In Chapter Five -, we are going to select a few critical policy variables for sensitivity tests. Lastly, “once enough confidence has been built in the validity of the model structure, one can start applying certain tests designed to measure how accurately the model can reproduce major behavior patterns exhibited by the real system. (Barlas, 1996, p. 193)” In these test, we will be comparing trends, average, and variations (section 4.2).

In the meantime, it should be kept in mind that validation of a model is a gradual process which is dependent on the modeling project’s purpose (Barlas, 1996). The purpose of this model is mainly to help in medium-term planning of data center expansion, and to serve this purpose, the model should be able to correctly predict the future trend in power-requiring assets, mainly physical servers, given different future growth scenarios, different market policies in terms of type of new future users, and various extents of virtualization. To make sure that the model is going to be able to do so, first we need to see whether it can replicate well enough historical trends of main variables under study, and whether it does so for the right reasons. In order to check whether the model is producing the right behavior for the right reasons, in the next section we are going to carry out partial model tests by feeding sections of the model with exogenous data, and investigating the output of each section.

4.1. Partial Model Tests

In this section, we set out to try to validate the structure by trying to confirm major dynamic hypotheses individually. In order to do this, we pick major structural assumptions one by one, and while feeding other relevant variables with exogenous data but keeping the central variable in the selected section of structure endogenous, we check the behavior of relevant variables against their reference modes. As such, if these reference mode comparisons in the test are passed successfully, we can conclude with confidence that the selected part of model structure is consistent with reality. Once we perform this test on all major dynamic hypotheses, we can move to overall reference mode comparison tests where all our dynamic hypotheses are in place, and where we can find out whether or not the engine built out of all these tested parts runs smoothly.

Naturally, the extent to which partial model tests can be performed depends on the degree of availability of time-series data for various important variables. For instance, when we have data for the actual historical inflow of a certain stock, or aging chain of stocks, we are able to test the validity of the conceptualization and formulation of the outflow structure by feeding the model with the exogenous inflow data and observing the behavior of the stock against its reference mode. This is the case for some of our major aging chains such as the ones for physical servers, virtual servers, and hosts. Whereas in other cases, for example for the stock of 'Users', such disaggregate inflow/outflow time-series data is not available, and therefore this kind of partial model testing would not be possible there.

4.1.1. Testing the Physical Servers' Outflow Structure

For this test, we feed the aging chain of physical servers with actual historical data of number of servers added. In other words, in this case, the 'Server Installation' rate (Figure 14) is exogenously driven. Here, we can see the behavior of the 'Total Physical Servers' variable (sum of all physical servers in all stages of the aging chain) against its reference mode[1]:

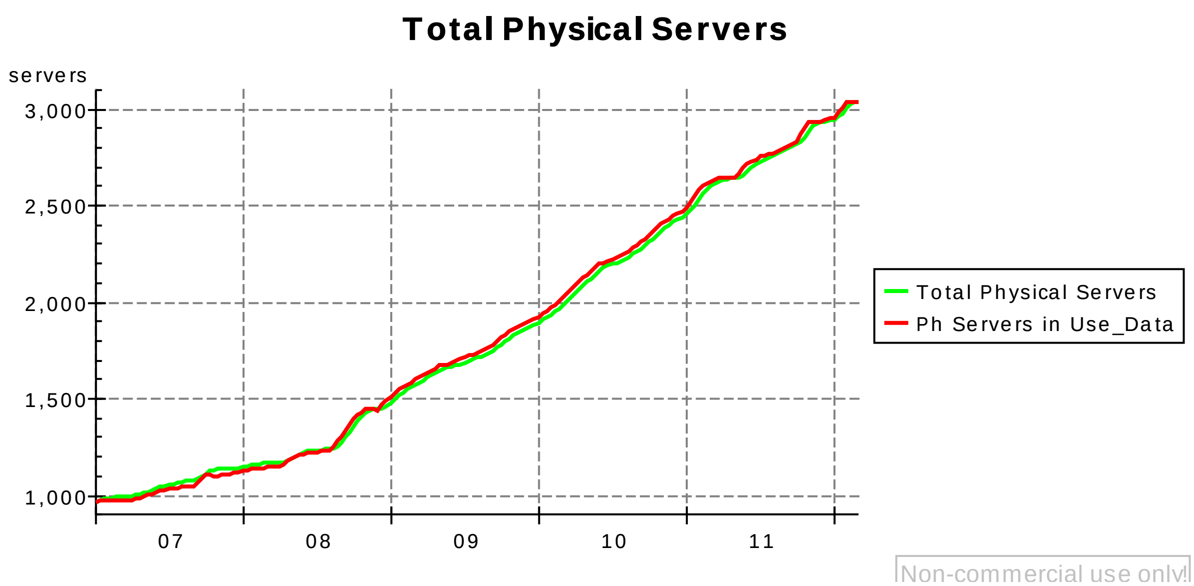


Figure 43 - Partial Model Test: Physical Servers Outflow

The fit is observed to be almost perfect. This validates our conceptualization of the dynamic of physical servers going through an aging pipeline, and further confirms our estimate for the total lifespan of physical servers, as seen in Figure 44.

---

[1] In this thesis, in all figures comparing simulated with historical behavior, the red line is data and the green line is simulated.

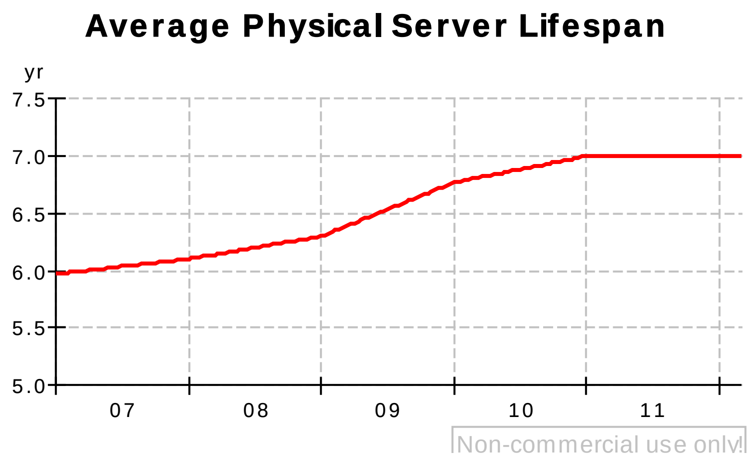


Figure 44 - Estimate for Average Physical Server Lifespan

4.1.2. Testing the Physical Servers Inflow Structure

For this part, we switch the inflow of the physical servers pipeline to endogenous formulation, while feeding other influencing variables with exogenous data. These include the outflow of the same aging chain, and the in- and outflows of the two related pipelines of virtual servers and hosts. These two other pipelines are directly relevant since they constitute the concept of 'logical servers' all together. We would set the inflow and outflow of our Users stock to actual historical if such data was available, but it's not. The result of this test can be seen in Figure 45.

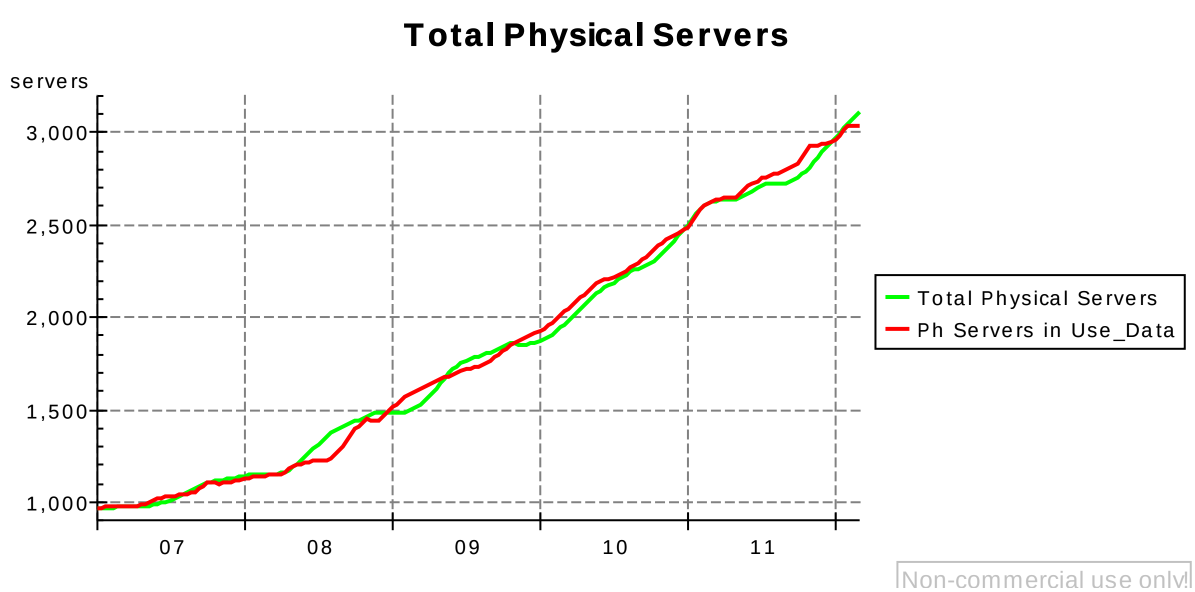


Figure 45 - Partial Model Test: Physical Servers Inflow

The fit is still very good, with a root-mean-square error that is 2.6 percent of the run average of the time-series data. This indicates that our dynamic hypothesis governing the inflow of physical servers is rigorous. This hypothesis is embodied in the Server Installation variable and influenced by 'Need-based

Ph Server Installation' and also by restrictions related to inventory, data center power, and human resources[1].

4.1.3. Testing the Virtual Servers Outflow Structure

In this section, in order to investigate the validity of the structure for disengagement of virtual servers, we simply input historical data into the 'Virtual Servers Inflow' and inspect the behavior of Total Virtual servers against the reference mode. The result is shown in Figure 46

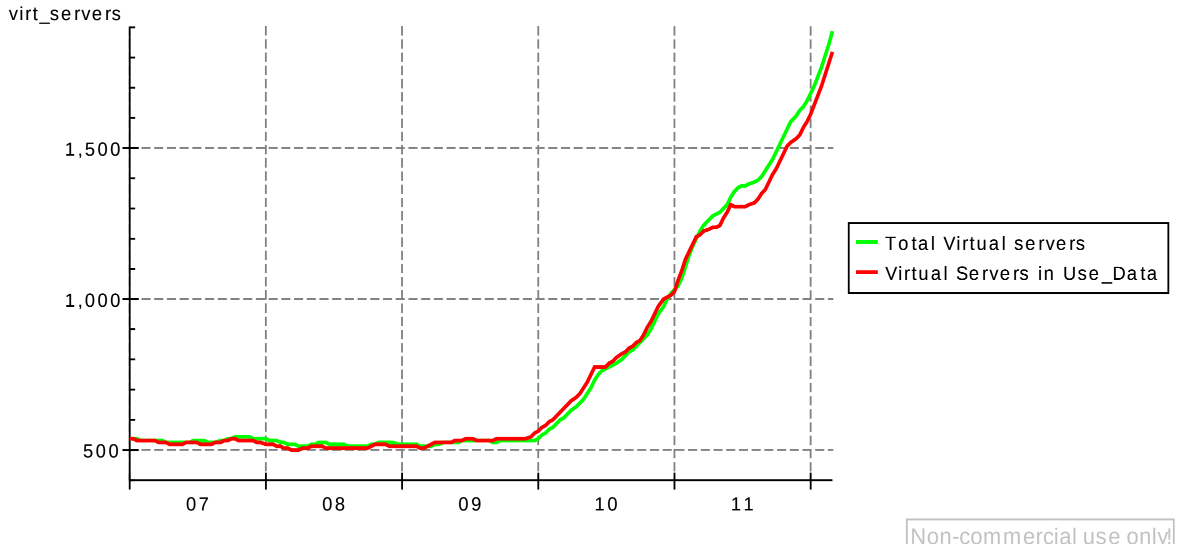


Figure 46 - Partial Model Test: Virtual Servers Outflow

This simulation is run with the following estimate for average virtual server lifespan:

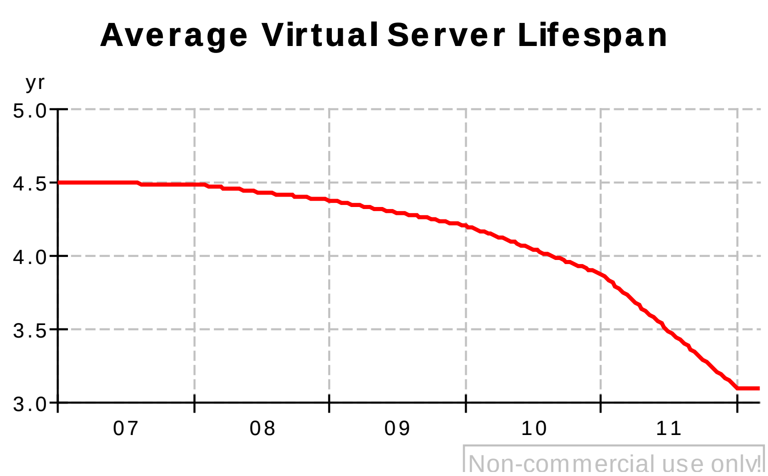


Figure 47 - Average Virtual Server Lifespan

[1] Particularly the inventory level restriction plays a role here, as seen in 4.1.2. The other two restrictions do not really come into play, since surplus data center power is always made available, and HR needs are complied with rapidly.

The observed fit is initially perfect, while a minor discrepancy forms from 2011 onwards. The logical implication is that the lifespan of virtual servers has probably shortened even further towards the end of the simulation period, increasing the outflow of virtual servers and pushing the simulated graph down closer to the reference mode. For us, this would imply eliminating one more stage of the virtual servers' pipeline, since with the current setting the lifespan cannot be made shorter than three years. However, since the current fit is still good enough for our purpose (root-mean-square error = 3.7% of run average of time-series data) we keep the current structure as it is.

4.1.4. Testing the Virtual Servers Inflow Structure

While we already conducted a direct-structure test on this hypothesis in 3.2.3 by comparing the actual historical inflow by the one generated by our assumption in Figure 20, we set out to corroborate our confidence by following the same procedure as we did for physical servers. We switch to exogenous inputs for flows into and out of relevant pipelines, namely the ones for physical servers and hosts, and observe the graphs for total virtual servers.

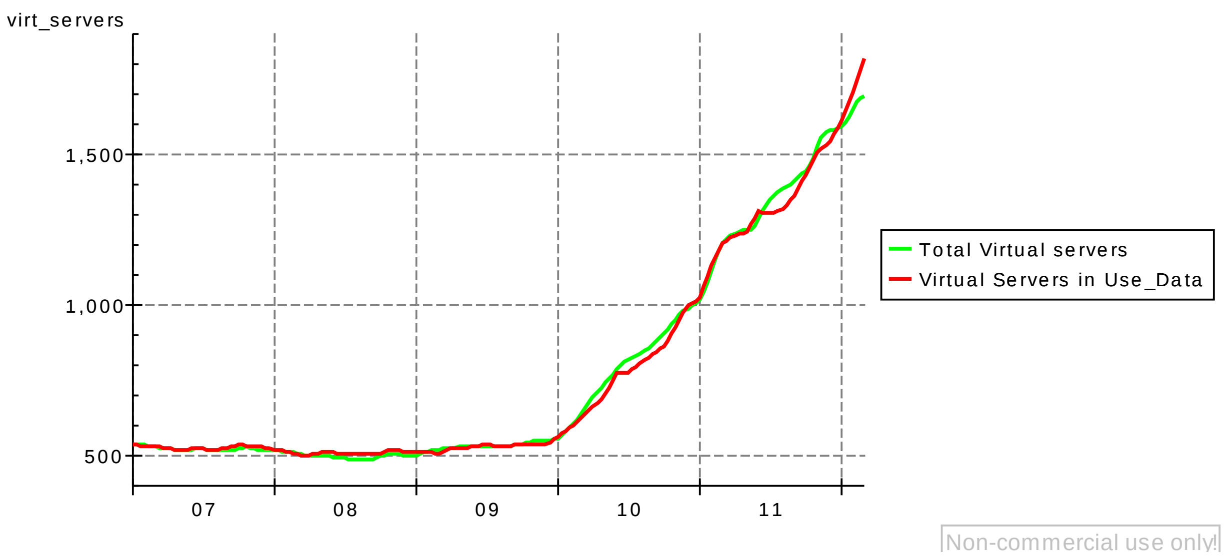


Figure 48 - Partial Model test: Virtual Servers Inflow

The results are favorable, with a root-mean-square error over run average of 3.1%. This gives further strength to our hypothesis that the process of adding new virtual servers can be directly linked to the addition of physical servers. This is only logical since both of these types of servers are added in parallel to respond to the requirements of new users coming in with each new contract.

4.1.5. Testing the Hosts Outflow Structure

Considering the aging chain of hosts, if we use historical time-series for the added hosts, assuming an average lifespan of four years for hosts, the following is the result that we obtain against the reference mode:

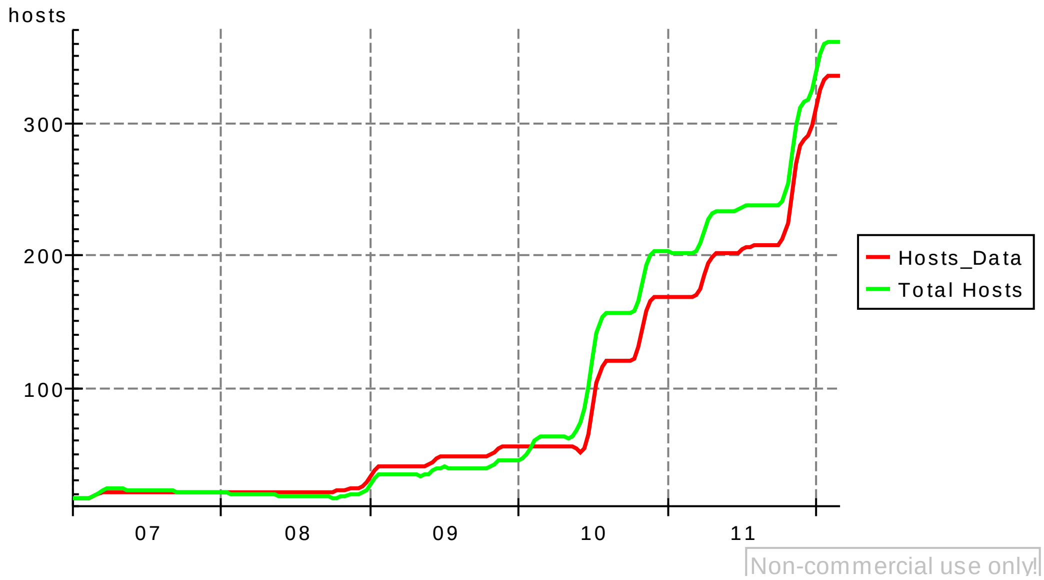


Figure 49 - Partial Model Test: Hosts Outflow_1

This time the result is not very encouraging; root-mean-square error (RMSE) divided by run average is more than 22%. Thus, we need to reconsider some of our structural assumptions.

Therefore, we need to refer to data for actual host disengagement to find out what is causing this incorrect behavior. The following graph plots simulated disengagement of hosts against the historical one:

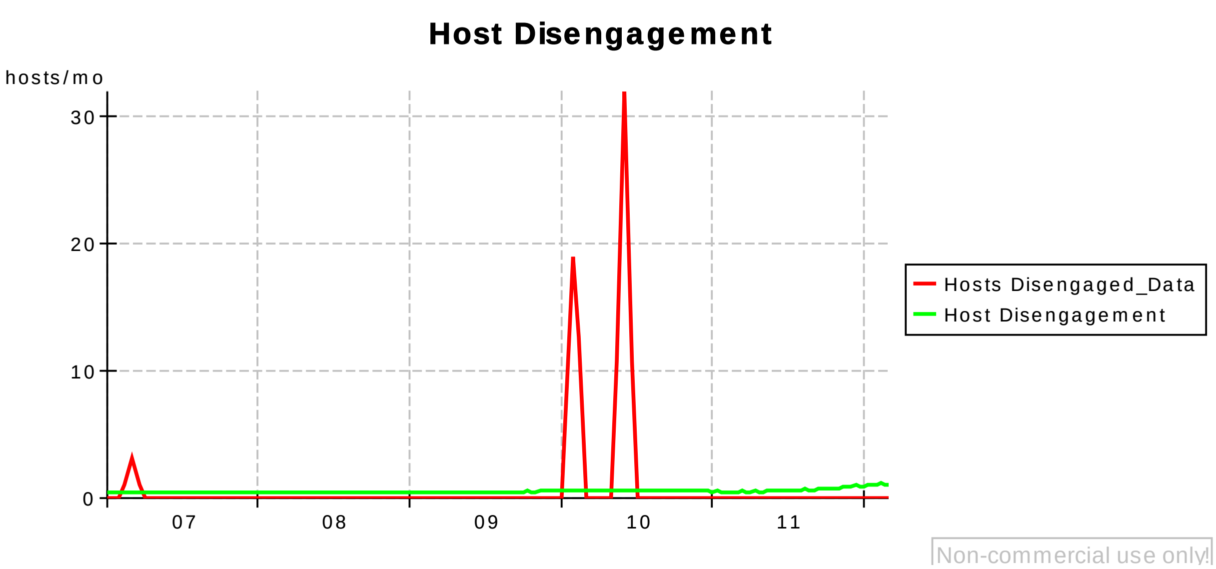


Figure 50 - Host Disengagement

The behaviors are far from congruous. In reality, hosts have been disengaged only in three instances; and mainly in two chunks in 2010. On the one hand we know that System Dynamics is not particularly suitable for modeling very discrete events like this, and on the other hand the client did not give a compelling explanation for this particularity in disengagement of hosts. The information given by the client regarding disengagement of hosts was that they live between three to four years. If we assume that the average host lifetime has been four years in the first three years of simulation, and three years after that, the error does decrease from around 22% to around 18%, but this is still substantial. Since we do not know the decision rule that is governing the peculiar outflow of hosts, the only way for us to get close to the reference mode is to exogenously force a sudden discarding of hosts in 2010, as in reality.

For instance, if in 2010 we force the discarding of 50% of second and third year hosts (using additional outflows of 'Hosts Discarded' for second and third stocks), we achieve the following acceptable fit with an RMSE divided by run average of just over 7%.

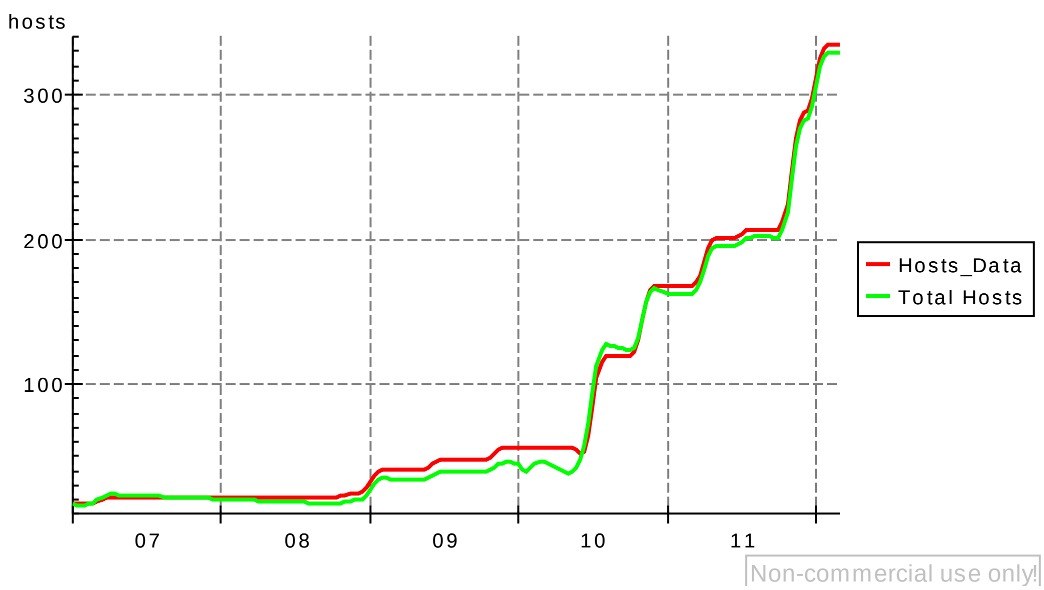


Figure 51 - Partial Model Test: Hosts Outflow_2

Since these estimates yield an acceptable result, we are going to use them in our final model.

4.1.6. Testing the Hosts Inflow Structure

At this stage we switch the side of the aging chain where we feed with exogenous data to the outflow side, and use the inflow rule based on a certain data-derived ratio of newly installed physical servers being converted to hosts every year. Figure 52 depicts the result.

As expected, we have successfully captured the general growing trend in the number of hosts, importantly getting very close to the actual number of hosts at the end of the simulation period. Nevertheless, in the model we add hosts in a gradual manner. However, unlike physical and virtual servers, hosts are not added gradually but in bulks. Therefore, we face an RMSE divided by run average of 20%. Yet, the good news is that the inflow policy rule used in the model is going to be much more representative of reality, and automatically yield a better fit, in the future. This is because of the fact that the new policy in ABC is to convert *all* new physical servers to hosts for virtual machines as they are installed. This implies that hosts will be added not in chunks but incrementally. This is reassuring since the main purpose of the model is future policy analysis.

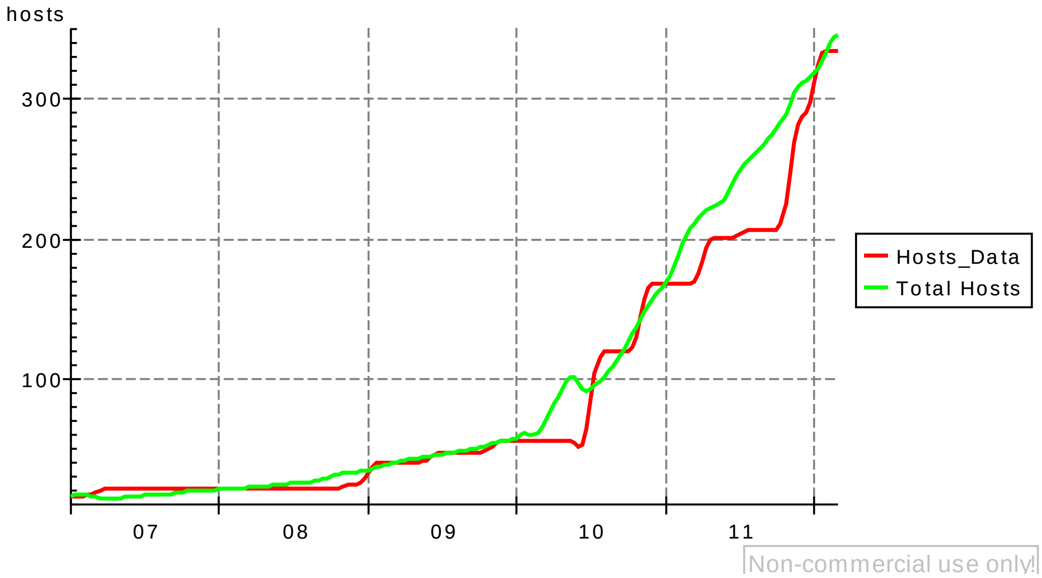


Figure 52 - Partial Model Test - Host Inflow

The main influences of the pipeline of hosts in the model are the following:

- Effect on the dynamics of the number of logical servers,
- Effect on the dynamics of total effective power demand of physical servers,
- Determining the maximum capacity for supporting virtual servers.

While the first two influences are captured in this model, the third one, which is a critical effect, is not modeled here. The reason for this, as pointed out earlier, is lack of information on the system governing the maximum possible virtual servers supported by one host. Ideally, once a modified version of the model would include this dynamic, the maximum capacity of hosts would set a limit on the inflow of virtual servers, while on the other hand the need to install new virtual servers would govern the conversion of more physical servers into hosts. This need would accumulate in a stock, where the outflow would be chunks of hosts being set up from time to time, when our virtual server capacity is close to complete depletion. Such dynamic is currently absent in the model because of lack of information, and this constitutes a major area of improvement for this work.

4.1.7. Testing the Outflow Structure of Spindles

As reviewed while describing the model structure, for storage spindles we have a similar aging chain to the one for physical servers. The depletion time constant for the final stock in this aging chain is a stable two years. The test result is shown in Figure 53. This result is compelling, with the two graphs almost coinciding. Therefore, we can go ahead with this structural assumption for the outflow of spindles.

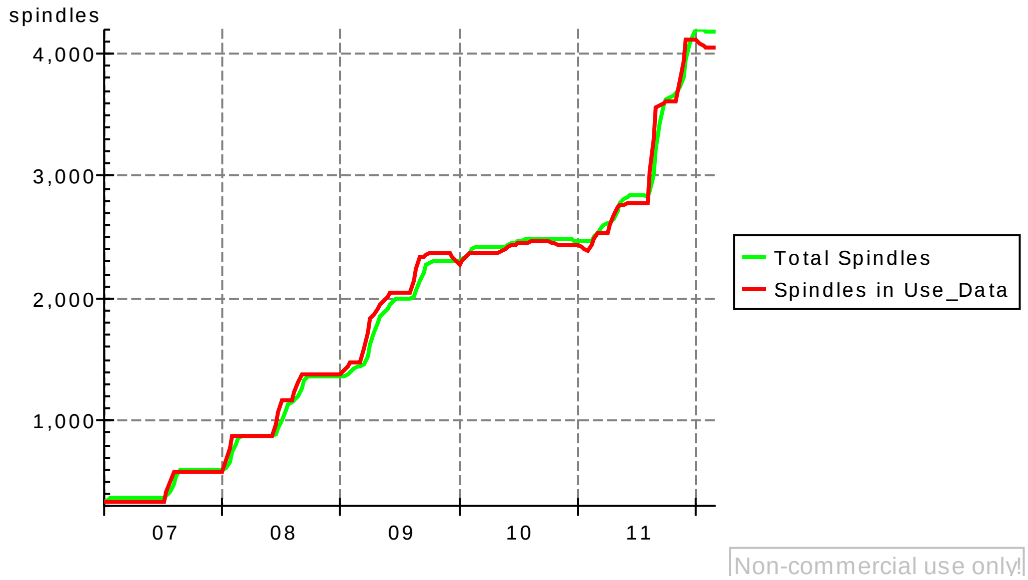


Figure 53 - Partial Model Test: Spindles Outflow

Since we are achieving good results with this test, we can use this opportunity to estimate the input of 'Storage per New Spindle', which has not been made available by the client. Using the estimate depicted in Figure 54 we attain the satisfactory fit for total storage shown in Figure 55. Therefore, we can consider the estimate in Figure 54 to be reliable enough to use in our analysis.

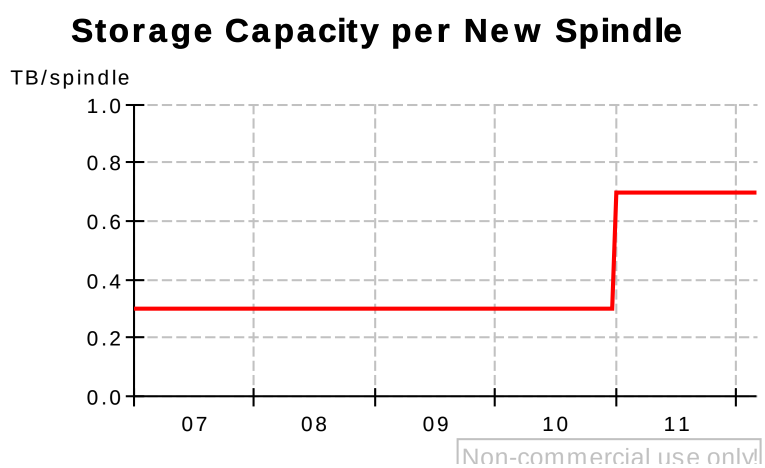


Figure 54 - Estimate for Storage Capacity per New Spindle

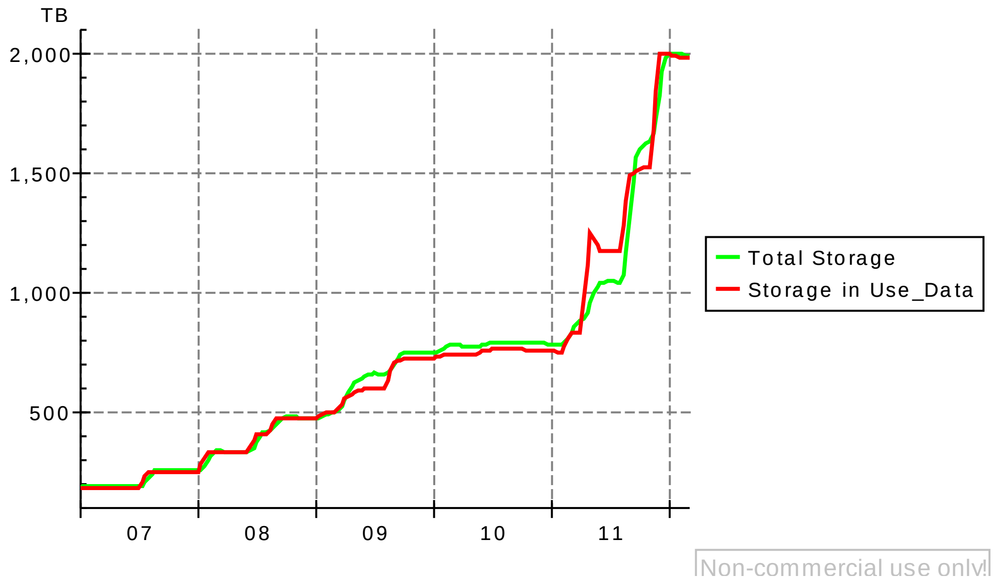


Figure 55 - Partial Model Test: Storage Outflow

4.1.8. Testing the Inflow Structure of Spindles

In this test, we let the inflow of spindles be determined endogenously based on user requirements, and we let the rate of storage capacity added flow alongside the inflow of spindles, whereas the outflow of spindles is set to be governed by historical data. The preliminary result shown in Figure 56 is not so reassuring.

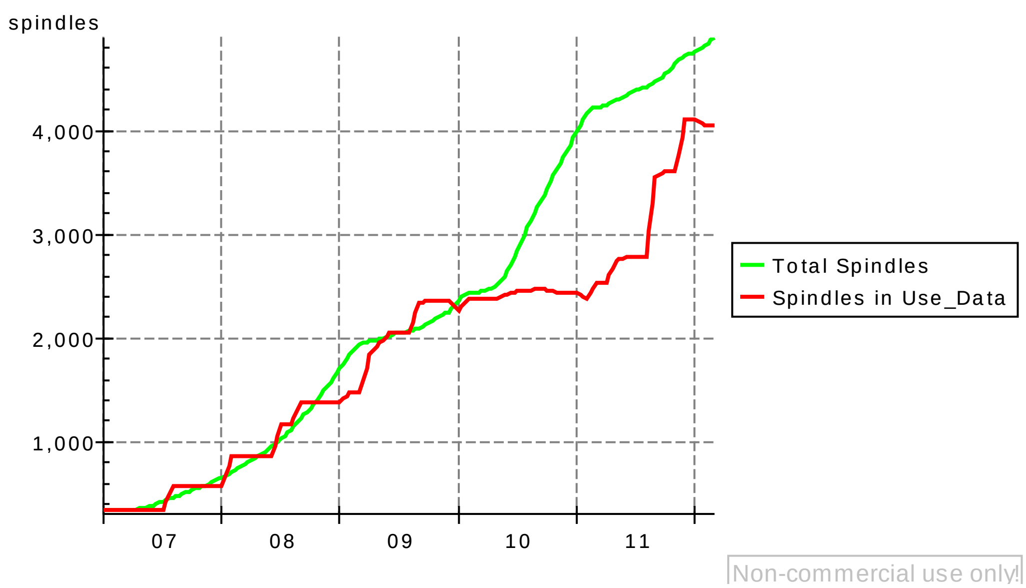


Figure 56 - Partial Model Test: Spindles Inflow_1

Looking at the data graph, it is easy to notice that the installation of new spindles has stalled in 2010. The cause of this is unknown to us. It can be deducted that this cannot be due to a stagnation in user storage requirements, because even if storage requirements of new users falls drastically in 2010, which is unlikely, total storage requirements of users will keep growing because the inflow of users is much higher than the outflow (user growth). Furthermore, number of spindles increase rapidly again in 2011, compensating for the stagnation 2010, which shows that need for storage has not stopped growing. Therefore, we can infer that in 2010 acquisition of new spindles has stopped for some reason, possibly an organizational one, even though need for spindles has kept growing throughout the simulation. This has led to the widening of the gap between required and actual storage, which has been sewn up again in 2011. Although we are not aware of the reason behind this, trying to implement this in the model may result in important insights.

To implement this, we need to introduce a very large delay in acquiring spindles in 2010 that would practically stop spindle acquisition, such as 999 weeks instead of 2. Afterwards, in 2011, we need to allow the model to recoup the gap in storage requirements by lowering back the delay time. Still, if we instantly decrease the delay time back to two weeks, the number of spindles adjusts itself too quickly as compared to historical data in 2011 (green line in Figure 57). Thus, we try using an acquisition delay of 99 weeks for 2011. This settings simulates as the blue line in the graph, which is a satisfactory fit (blue and green lines coincide before 2011). This longer than normal delay can be justified because in order to close the gap from 2010 we need to acquire a lot more spindles than normal routine and this will take longer than usual.

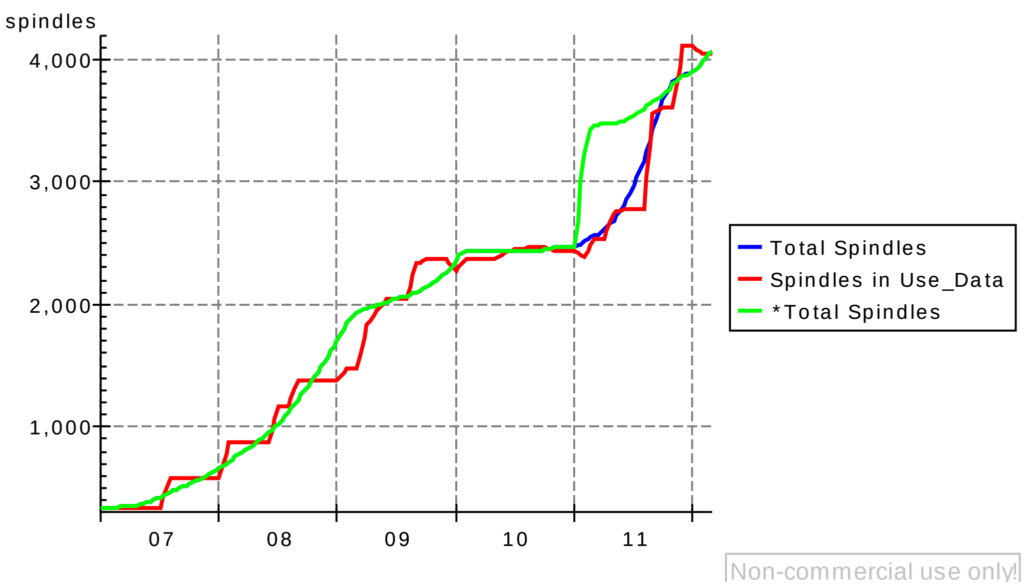


Figure 57 - Partial Model Test: Spindle Inflow_2; Red: data, Green: normal acquisition dleay in 2011, Blue: long acquisition delay in 2011.

Hence, for the inflow of spindles we can use the settings arrived at in this section for the rest of our analysis.

4.1.9. Storage Capacity: Data Inconsistency

Assuming we set both the inflow and the outflow of the aging chain of spindles to time-series data, and we initialize the model with total storage according to data at the beginning of the simulation period, the model must produce exactly the same numbers as time-series data for total storage. However, as we can see in Figure 58, there is a slight inconsistency in data for storage capacity. Therefore, among the time-series data for the inflow, the outflow, and the stock of storage capacity, one or more has to include incorrect data.

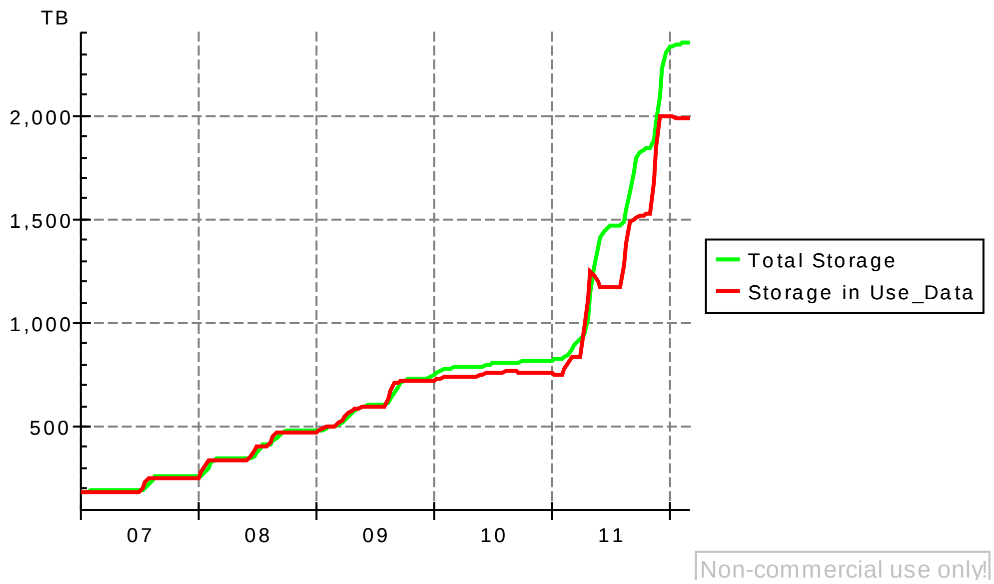


Figure 58 - Storage capacity data inconsistency

During the course of this project, some other instances of data inconsistency, such as inconsistency in data for the historical total number of CPUs, historical physical server disengagement, etc. were discovered. Discovering incorrect data, which can sometimes easily be achieved using System Dynamics, can be considered as one of the positive side-effects of an SD modeling project for a company.

4.2. Reference Mode Comparison Tests

Having built a certain extent of confidence in the model through running as many partial model tests as possible, considering data availability, we move over to testing the general behavior of important variables in the model against reference modes, where time-series data are available. Conducting such tests, we can make sure that the structure of the model, while being partially reliable according to partial model tests, is also coherent and consistent as whole.

4.2.1. Physical Servers Reference Mode Test

Figure 59 illustrates the simulated behavior of total physical servers against historical data for this variable. This variable reflects the sum of all physical servers installed within all age cohorts, utilizing the array sum function where our stocks are arrays of twelve elements.

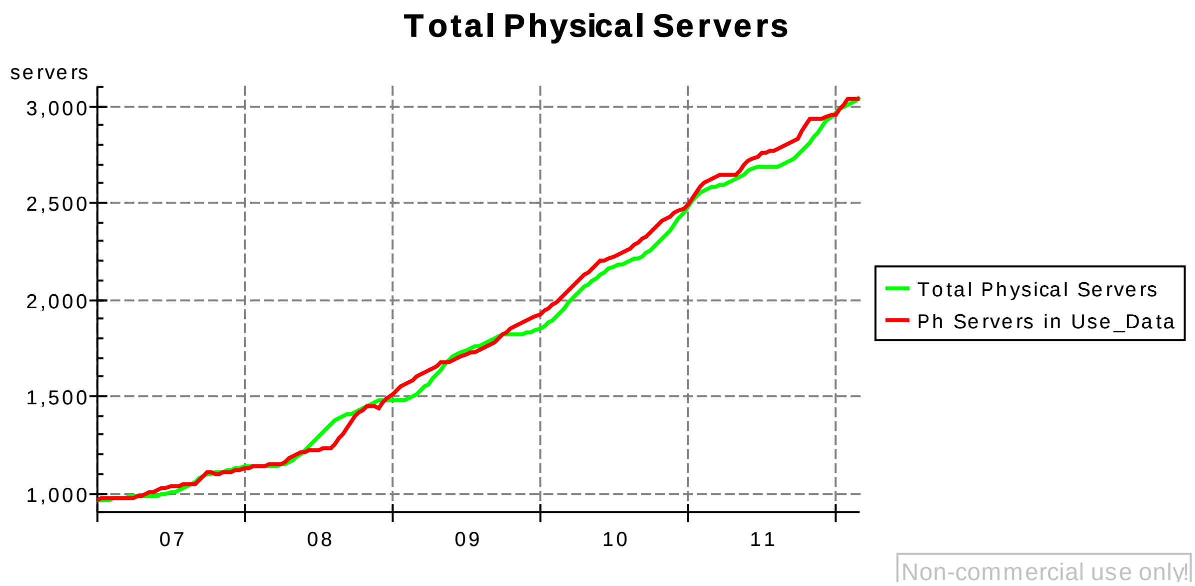


Figure 59 - Physical Servers: Simulated vs. Historical

The observed fit is quite good, considering the fact that the in- and out-flows are entirely generated by endogenously driven dynamics. The RMSE divided by run average is 2.9% and the correlation coefficient is 0.998. This good fit is mainly thanks to the conceptualization of the installation of new servers in the model. More and more servers come into the fleet because of the growing need for servers as the number of users grows. But in addition to that, there are some characteristic fluctuations in the historical graph as it is going up, which the model manages to emulate in a way. In other words, there are significant oscillations in the server installation rate, as portrayed in Figure 60. The model more or less replicates these seemingly random oscillations, and sometimes very closely so.

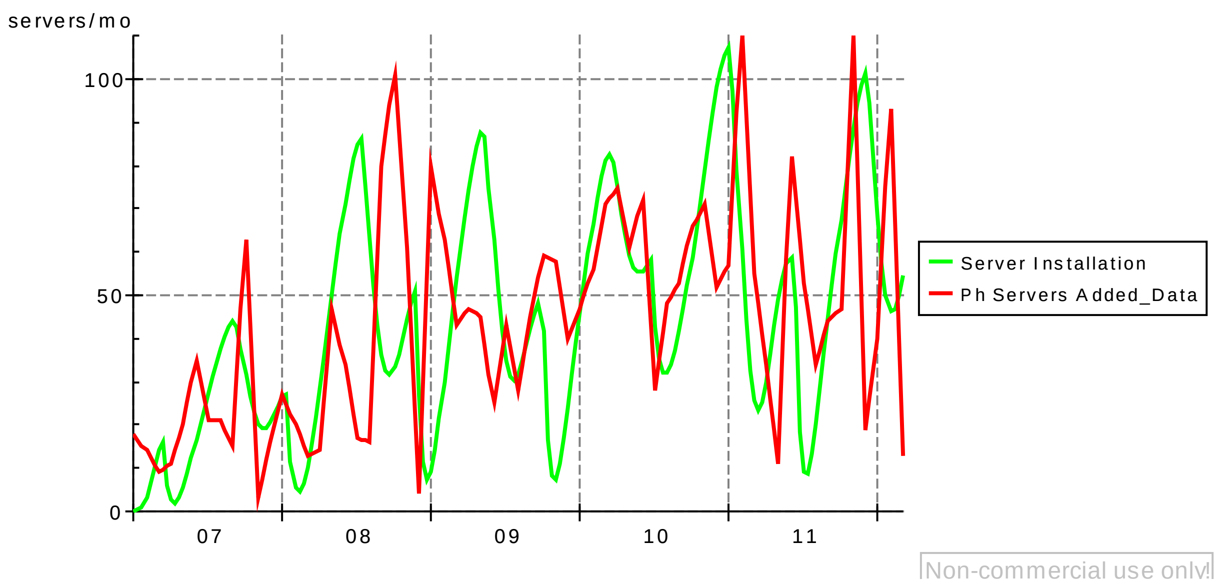


Figure 60 - Physical Server Installation Rate: Simulated vs. Historical (Inventory Coverage Period=one month)

The mechanism that leads to this model-generated behavior is linked to an important restriction on the installation rate: availability of physical servers in the inventory. The graph in Figure 61 reports the simulated amount of physical servers left in the inventory. For this variable, no historical data was available for comparison. The inventory of servers in ABC seems to be oscillating, as inventories in different businesses often do, sometimes reaching total depletion until new orders arrive. This dynamic is exactly what is behind the seemingly random oscillations in the installation of new servers.

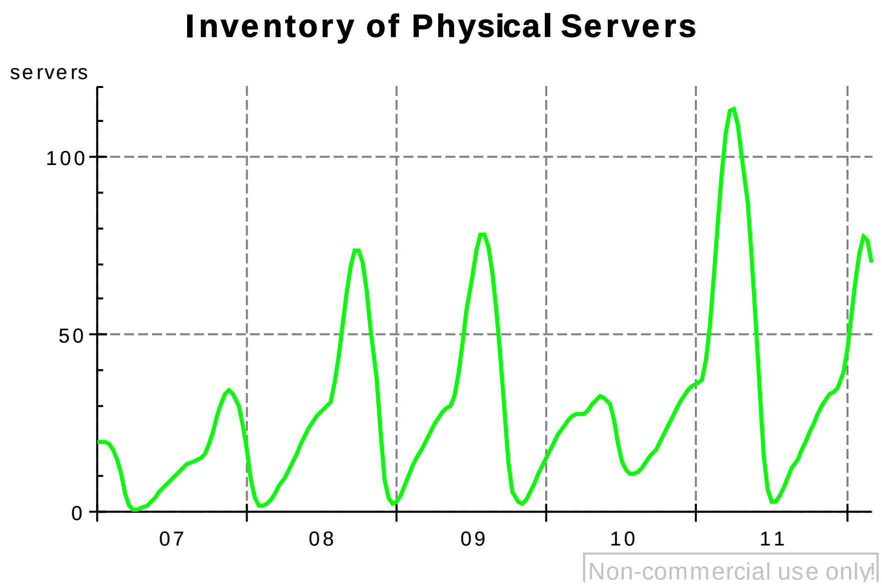


Figure 61 - Inventory of Physical Servers, Simulated

This behavior results from setting the 'Desired Ph Server Inventory Coverage Period' parameter (Figure 16) to one month[1]. As a matter of fact, when asked for an estimate for this parameter in reality, the client suggested that ABC keeps around two months worth of physical servers to be installed in inventory. Nevertheless, if we set this parameter to two months as suggested, the oscillations in inventory and consequently in installation rate are going to be smoothed out to a significant extent (Figure 62). With this parameter setting, the total number of physical servers keeps growing much more smoothly than before, losing the characteristic oscillations (Figure 63). Therefore, although this parameter setting actually gives a better fit for our main variable of total physical servers (RMSE divided by run average only 1.8%), I decided to keep the shorter length inventory coverage period because the shape of the graphs that it produces are more realistic.

[1] These oscillations are dampened to a certain extent when reducing the simulation timestep, but they never completely go away because of the intrinsic unstability described in the causal loop diagram a bit further.

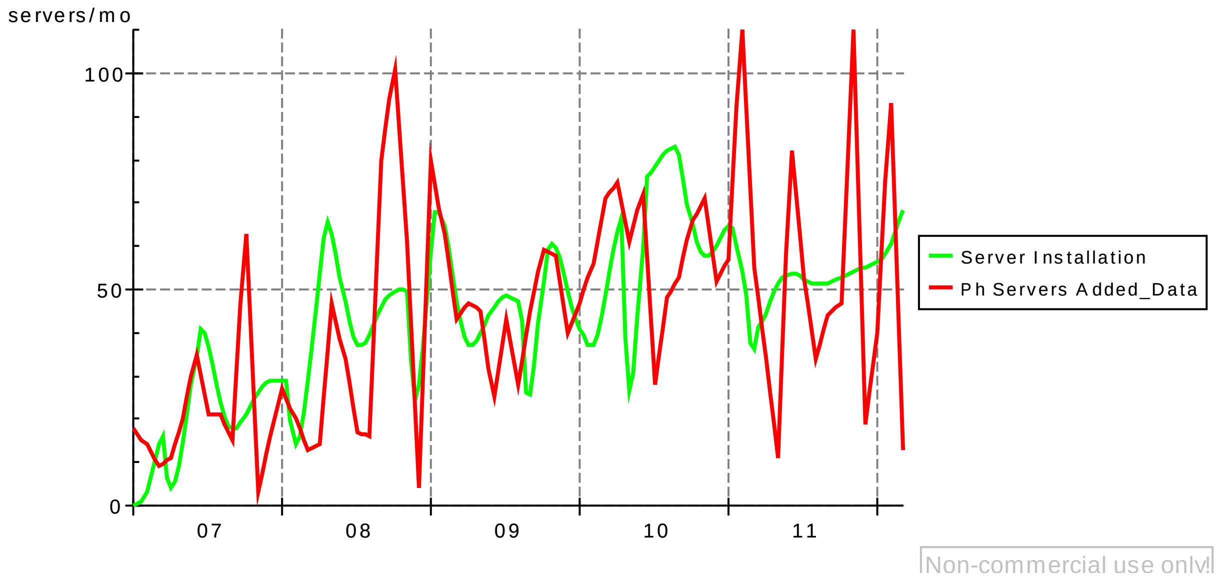


Figure 62 - Physical Server Installation Rate: Simulated vs. Historical (Inventory Coverage Period=two months)

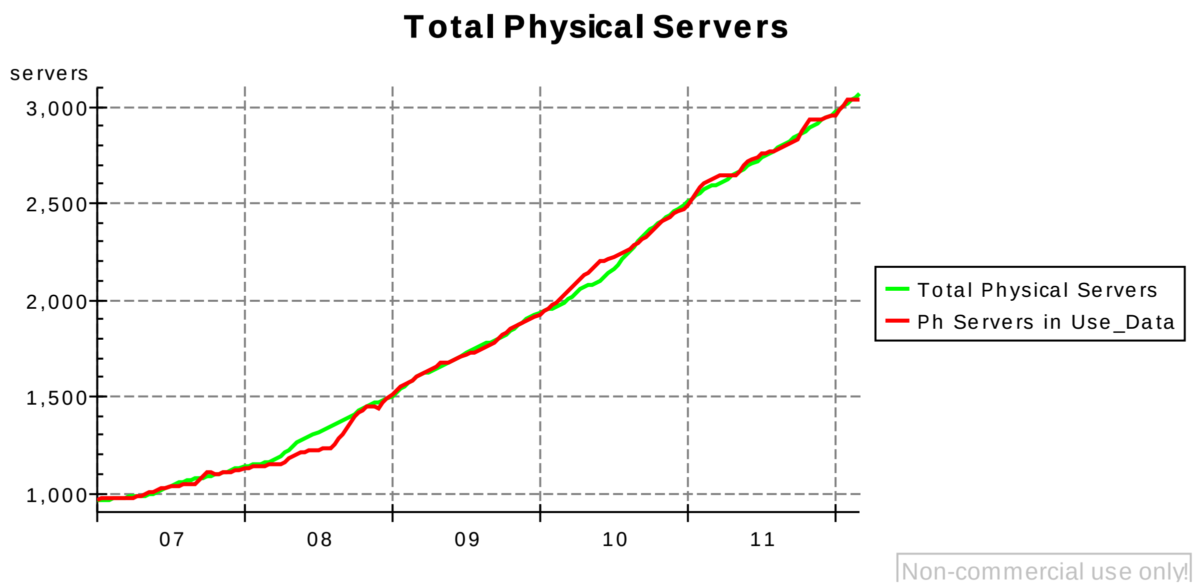


Figure 63 - Total Physical Servers: Simulated vs. Historical (Inventory Coverage Period=two months)

In order to study the structure responsible for the observed oscillations, let us have a look at the causal loop diagram in Figure 64. The diagram consists of four balancing feedback loops. The B1 loop involves the balancing structure which drives server acquisition in order to refill the inventory to close the gap between the desired and the actual level of the inventory. Loop B2 describes the depletion of the inventory due to servers being installed, and it also involves the effect of availability of servers in the inventory on server installation rate. Loop B3 portrays how server installation adds to the installed servers base and how the augmentation of the server base closes the gap in logical server requirements, which is driven by growth in the user base (not shown in the CLD). However, the major unstabilizing loop behind the oscillatory behavior in the rate of server installation and in the inventory is the B4 loop. This is the big loop whose trajectory goes around the figure, and which involves two stocks together with delays: classical requirements for an oscillatory behavior. This is how the loop works: A higher server installation closes the logical server requirements gap more rapidly (B3), lowering installation the next time round with a delay, which causes the inventory to be higher than it otherwise would have been, lessening the need for server acquisition, leading to a lower level in inventory than otherwise would have been (B1), and at last potentially hindering server installation; Thus, closing the loop. This balancing effect with a considerable delay is responsible for inventory oscillations.

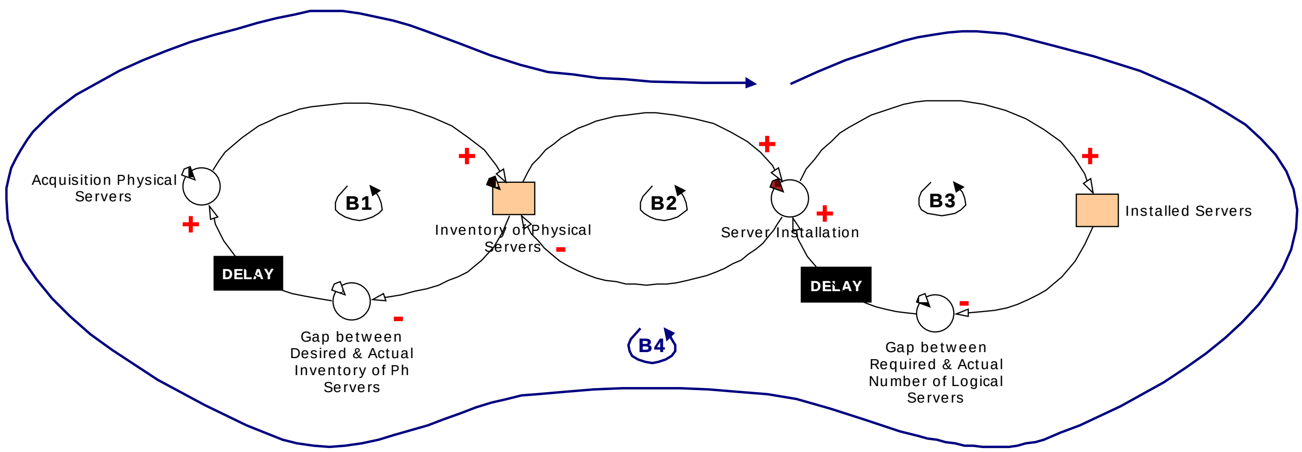


Figure 64 - Oscillations in Server Installation: Causal Loop Diagram[1]

4.2.2. Hosts and Virtual Servers Reference Mode Test

In Figure 65 we can compare the simulated graph for total number of hosts with the historical time-series graph. As already discussed, our model does not incorporate the mechanism for adding or discarding hosts in chunks, and therefore, we have to tolerate lack of point accuracy and a significant RMSE divided by run average of 17.7%. Nevertheless, we have succeeded in replicating the growing trend quite well, with a correlation coefficient of 0.991. As discussed earlier, this is more important than point accuracy, particularly in the future, since, according to recent policy, all physical servers are converted into hosts as they are installed.

[1] I can proudly describe this as the ugliest CLD I have ever made. Powersim is horrendous when it comes to drawing causal loops. Vensim is superior in this aspect (probably *only* in this aspect). However, I did not want to lose consistency in the looks of my causal loop diagrams with my stock and flow diagrams. That is why I kept using Powersim.

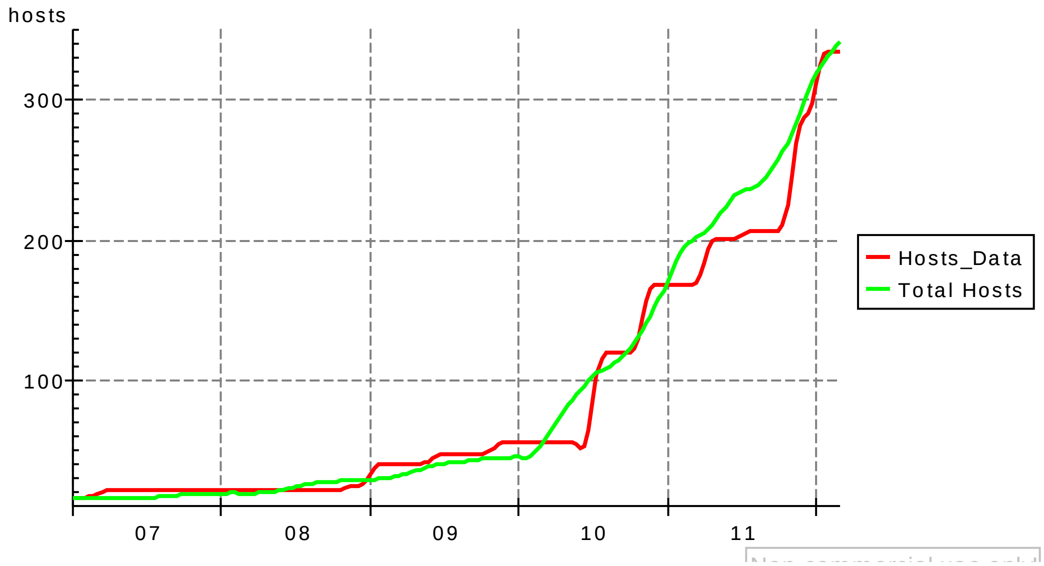


Figure 65 – Total Hosts: Simulated vs. Historical

The reference mode comparison test for total virtual servers can be studied in Figure 66. The two graphs clearly match very well, and the RMSE divided by run average is 4.4%. The model even closely imitates characteristic ups and downs, which can be easily noticed in 2011 where the two graphs separate a bit. This phase compliance is due to the realistic modeling of the inflow of virtual servers, as discussed in 3.2.3 and portrayed in Figure 20.

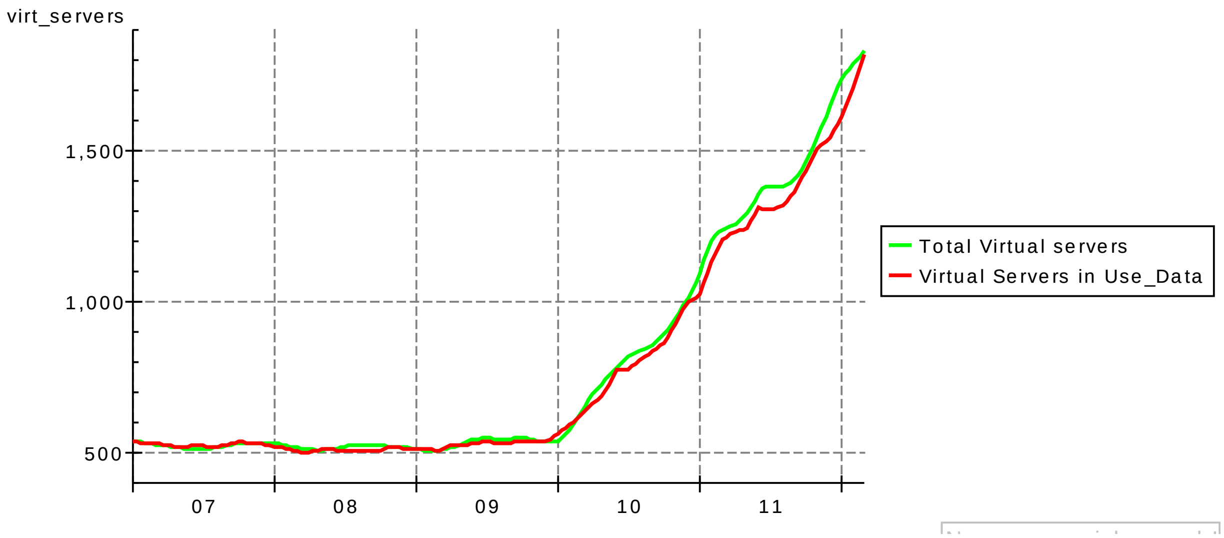


Figure 66 - Total Virtual Servers: Simulated vs. Historical

The following graph shows the simulated versus historical graph of the important indicator of 'Virtual Servers per Host', which is simply total virtual servers divided by total hosts at any point in time. Here again, we have got the trend right but not actual values, due to the discrepancy between simulated and historical number of hosts.

Figure 67 - Virtual Servers per Host: Simulated vs. Historical

There has been a general declining trend in average number of virtual servers per host. This is inspite of the fact that ABC's servers have grown stronger and stronger through time, as testified by the growing average number of virtual cores per physical server seen in section 4.2.4. This is an evidence of the growing trend in user requirements and service standards, as confirmed by our client, leading to fewer and fewer virtual machines being run on one an average host.

A compelling benefit of such a modeling effort for an organization is to point the management's attention toward important indicators such as this one. Companies such as our client's often have this kind of time-series data at their disposal, but rarely do they plot the data over time to inspect the dynamic development of important indicators.

4.2.3. Logical Servers Reference Mode Test

A logical server, the processing unit serving the end-user application's processing needs, is by definition either a stand-alone physical server or a virtual server. As such, the number of logical servers equals the total number of physical servers minus the number of hosts plus the number of virtual servers. Therefore, the graphs in Figure 68 are obtained using this simple arithmetic. The fit is as good as it gets, with an RMSE over run average of 2.2%. This is despite the relatively large point inaccuracy in simulation values of hosts. Then again, hosts comprise at most (in 2012) eleven percent of total physical servers. That is why point discrepancies between simulated and historical values do not make up a significant error in the total number of logical servers. This share will be increasing in the future, but at the same time, the ability of the model in closely reproducing the number of hosts will increase.

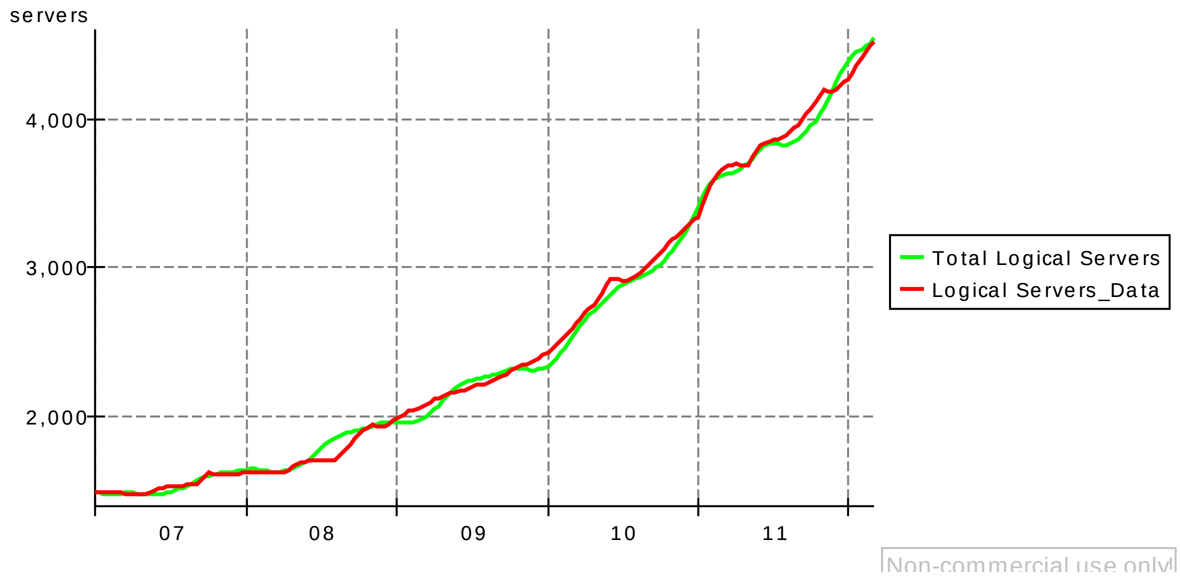


Figure 68 -Total Logical Servers: Simulated vs. Historical

In relation to logical servers, several informative indicators can be defined. Here, we study two of them: average logical servers per physical server, and average users per logical server.

The number of logical servers per physical server is an important performance indicator informing the decision-maker about the average level of utilization of physical servers (Figure 69). This ratio has been declining between 2007 and 2010, in line with rapid declining number of virtual servers per host, and then taking back off since 2010 along with accelerated virtualization and stabilizing ratio of virtual servers per host.

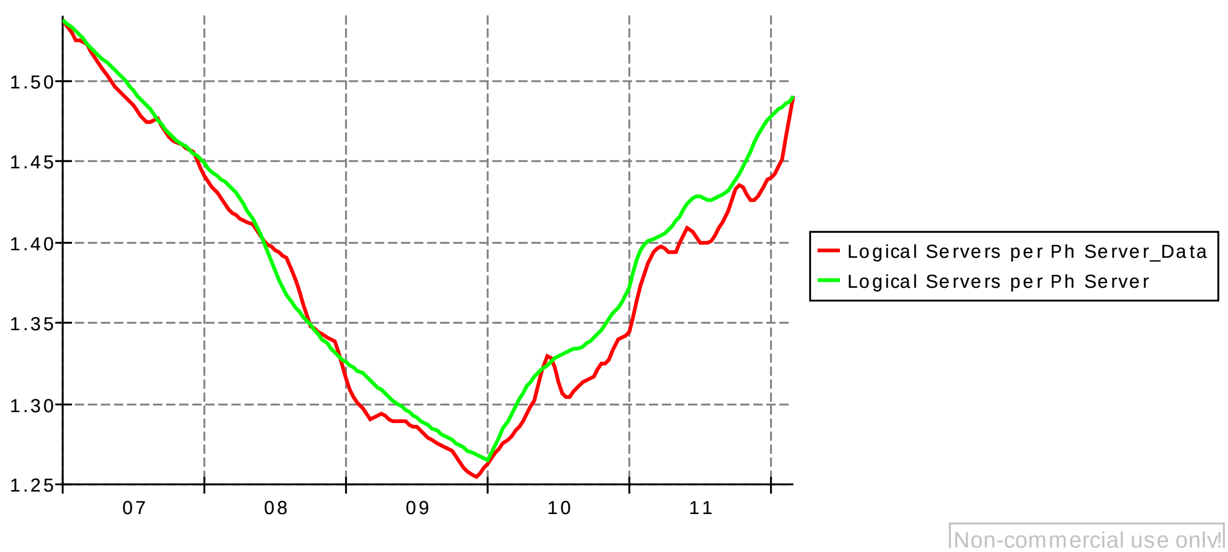


Figure 69 - Logical servers per Physical Server: Simulated vs. Historical

As for the proportion of average users per logical server, the trend is illustrated in Figure 70. We will come to the 'User's' variable in section 4.2.7. But for now, let us examine the behavior of the service-level-related ratio of average users assigned to one logical server.

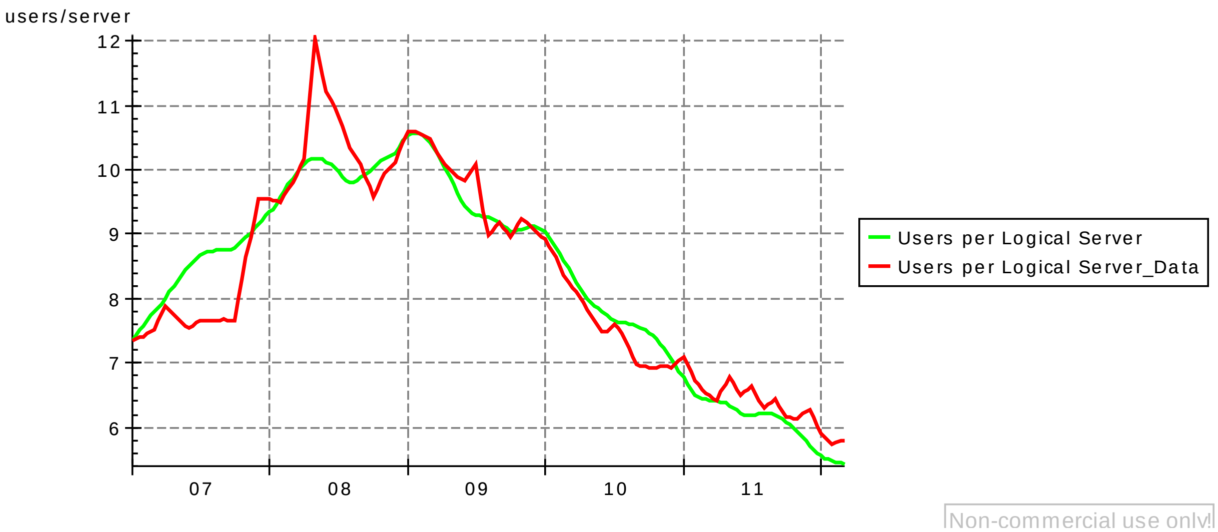


Figure 70 - Users per Logical Server - Simulated vs. Historical

As the graphs demonstrate, during the first couple of years the number of users sharing the same logical server unit was rising, implying a relatively relaxed standard in this dimension of service level, and showing that the number of users was growing faster than the number of logical servers. Thereafter, following more stringent service standards, the trend reverses, bringing the current average users per logical server to fewer than six users.

4.2.4. Data Processing Capacity Reference Mode Tests

The total of virtual CPU cores in physical servers is one of ABC's critical assets since it determines the total processing capacity of the firm. Furthermore, average number of cores per host is one of the determinants of maximum possible virtual machines hosted by one physical server. In this section we examine the development of variables linked to processing capacity, starting with Total CPUs.

Figure 71 portrays simulated versus historical total number of CPUs in all physical servers. There exists a gap between model-generated and data-driven behavior, which is wider than the gap in the Total Physical Servers comparison test. Considering the given information that since 2007 all new physical servers have been coming with dual CPUs, and taking into account the fact that simulated physical servers matched historical data quite well, it is logical to expect an equally good fit between the following graphs. A weaker fit can be speculated to point to a possible inconsistency in data.

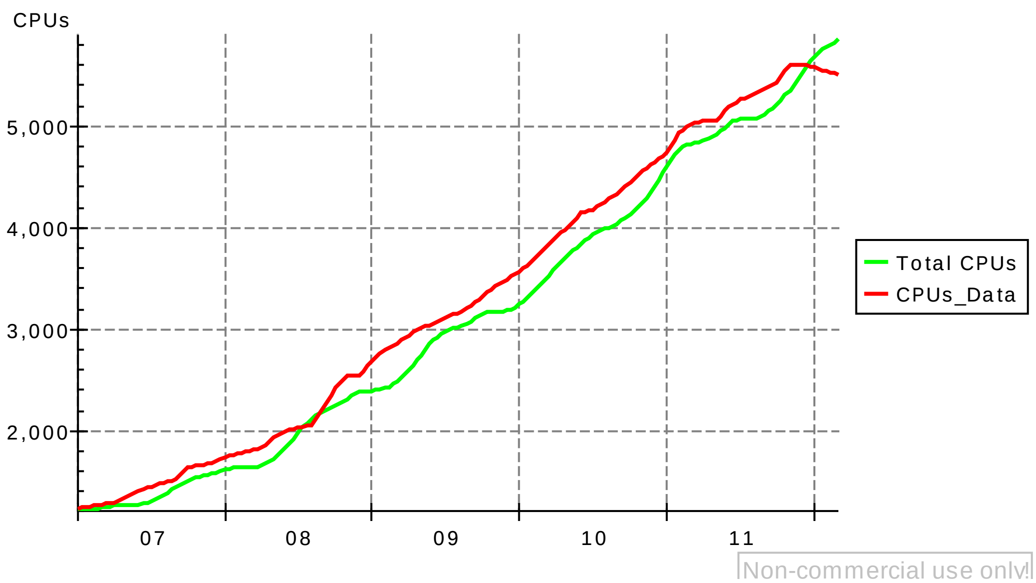


Figure 71 - Total CPUs: Simulated vs. Historical

This belief is invigorated after examining the reference mode comparison test for the number of cores in Figure 72. As we can see, the fit is quite impressive here again, except for in 2010, where a discrepancy is not unexpected because the same discrepancy exists for the physical servers test. This is good news because the number of cores is operationally more important than the number of CPUs for ABC.

In both reference modes for CPUs and for cores the graph suddenly heads down in the end of 2011. This seems counter-intuitive, or illogical, since the number of physical servers has not stopped growing. The client has to be consulted for a possible explanation about this[1].

The following graph illustrates the reference mode comparison test for the already discussed variable of virtual cores per physical server. This indicator, which is attained by simply dividing total cores by total physical servers, has been rising from 2 cores per physical server to around 8, reflecting the drastic improvement in the average performance of computers. The fit is satisfactory.

Still, looking at the overall average for cores per physical server is not sufficiently informative, because this variable has been growing very rapidly, and therefore physical servers of different age cohorts should have significantly different profiles regarding this feature. It is crucial that the company monitors this indicator at a level of servers from the same year cohort, or even servers from the same month cohort, which is a possibility featured in this model. Here, we are going to have a look at the dynamics created by feeding a stepwise increasing input to an aging chain of virtual cores, a classical textbook problem.

[1] This data is quite recent as of the time of this writing, and opportunity for a talk with the client about this has not come up yet.

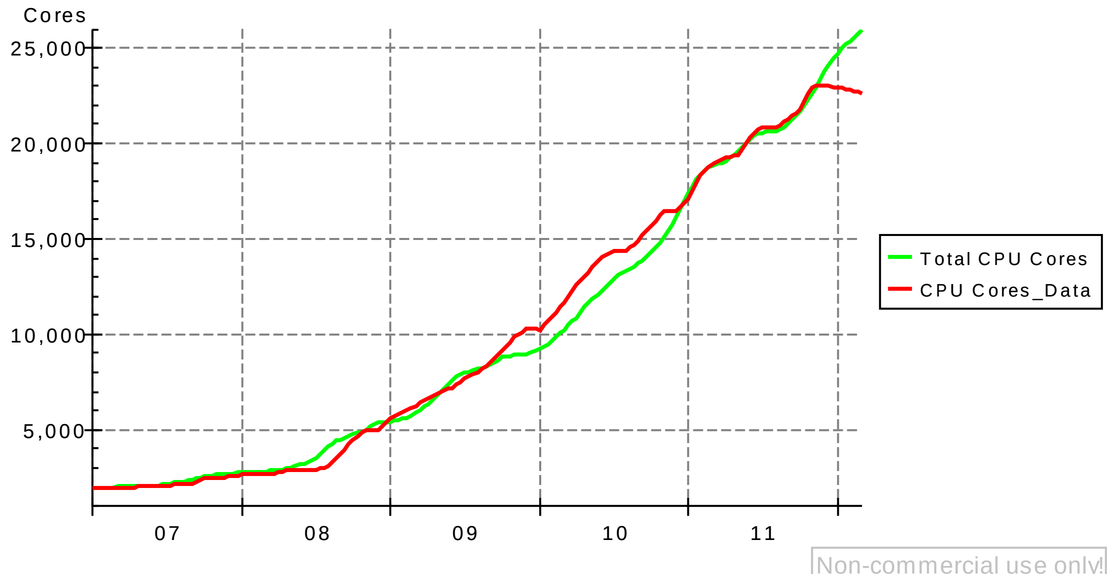


Figure 72 - Total CPU Cores: Simulated vs. Historical

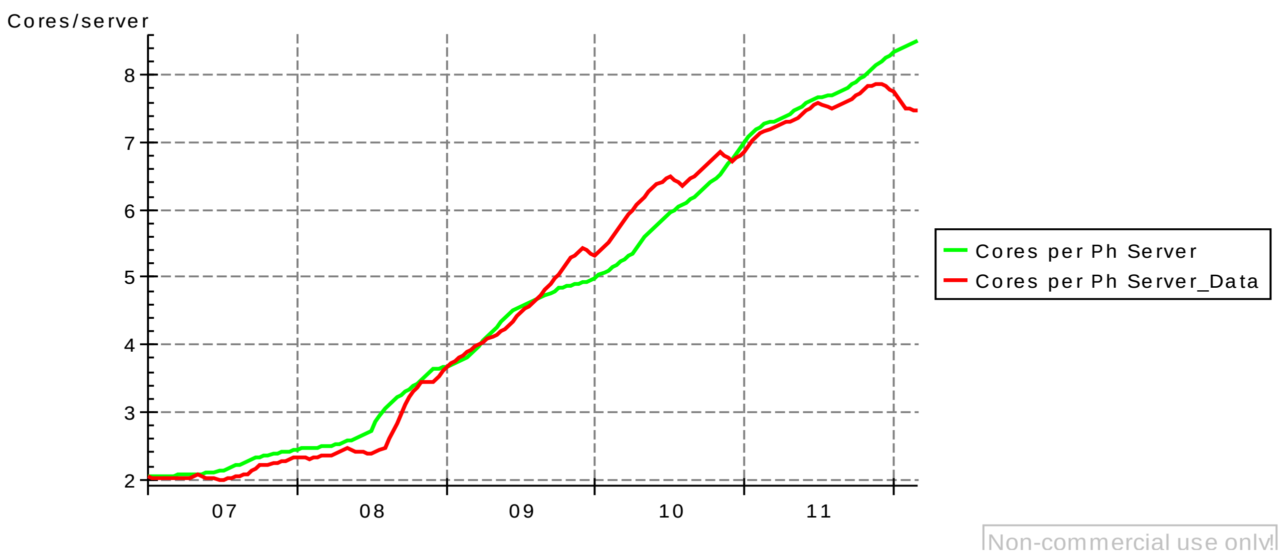


Figure 73 - Virtual Cores per Physical Server: Simulated vs. Historical

Average cores per physical server is plotted in the following graph for new servers coming in and for all successive age cohorts. The first-year average is a first-order adjustment to the input, the second year average a second-order adjustment, and so forth. As a striking demonstration of delays, at the beginning of 2012, when new physical servers have been coming with twelve cores each for already two years, and when the first-year cohort has just caught up with this new status, the older-than-five-years servers are still down at two, living in an era of the past. Older than three years server, with an average of 4.8 virtual cores per server, comprise more than a third of the total of our physical servers. At the end of the simulation, average cores per physical server is just over 8, while ABC has been acquiring servers with 12 cores each for more than two years. All this, emphasizes the high inertia in a system founded on assets with an important age attribute embedded in them.

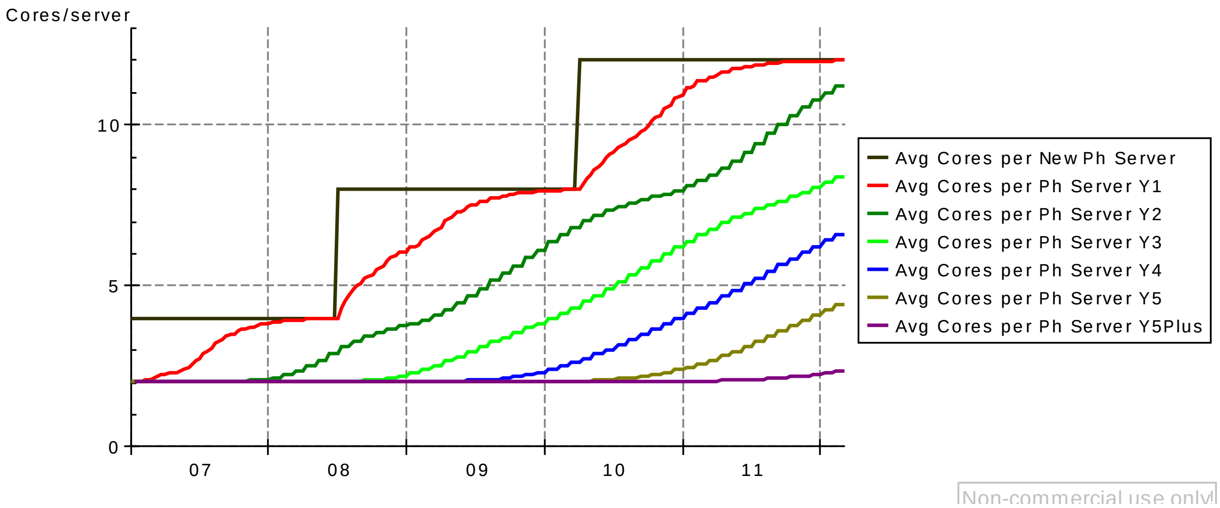


Figure 74 - Cores per Physical Server; Aging Chain Dynamics

In a further developed version of this model, maximum capacity of hosts in term of number of virtual machines supported would ideally be linked to features of hosts such as average number of cores. The decision-maker should keep a close eye on the 'average core per physical server' indicator for different age cohorts to make such potential decisions as converting older stand-alone servers to hosts.

4.2.5. Storage Capacity Reference Mode Tests

Storage requirements have been growing with an accelerated rate of growth, both due to growing number of users and due to a growing standard for the amount of redundancy necessary for higher data safety. There has been a temporary arrest in this growth in 2010, which we modeled using an extra-extra-large delay in the purchasing of spindles, as already discussed, which might be what actually has happened in reality.

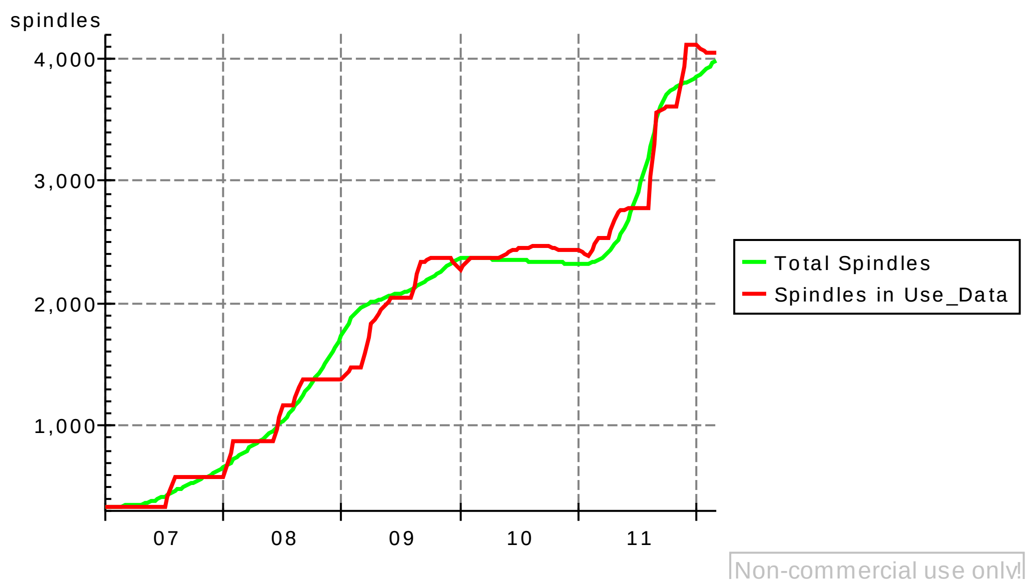


Figure 75 - Total Spindles: Simulated vs. Historical

Storage capacity (Figure 76) has grown in a likewise manner, with more than half of the growth taking place in 2011, when storage capacity of new spindles more than doubled. The estimate for 'Storage per New Spindle', attained by using the model, is shown in Figure 77.

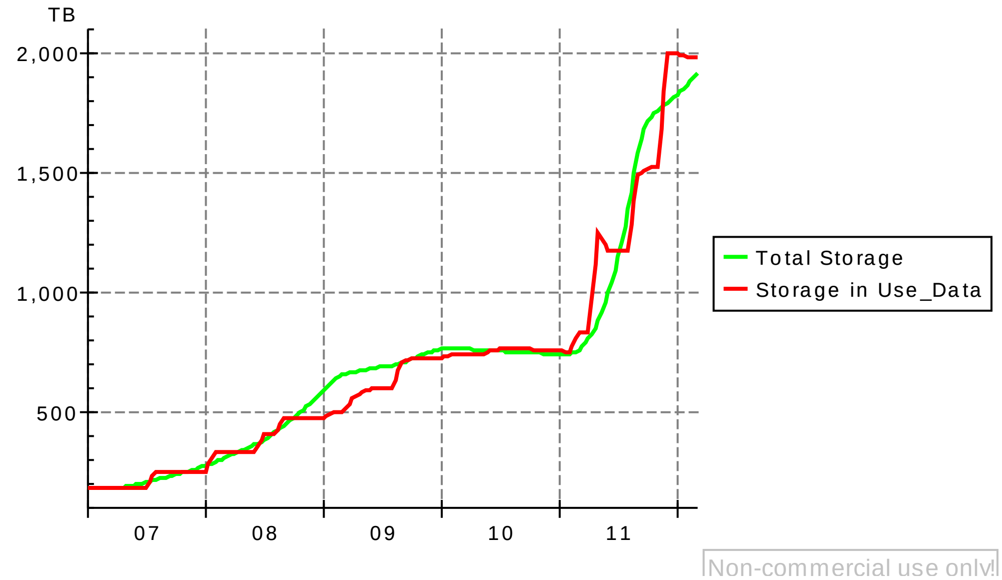


Figure 76 - Total Storage Capacity: Simulated vs. Historical

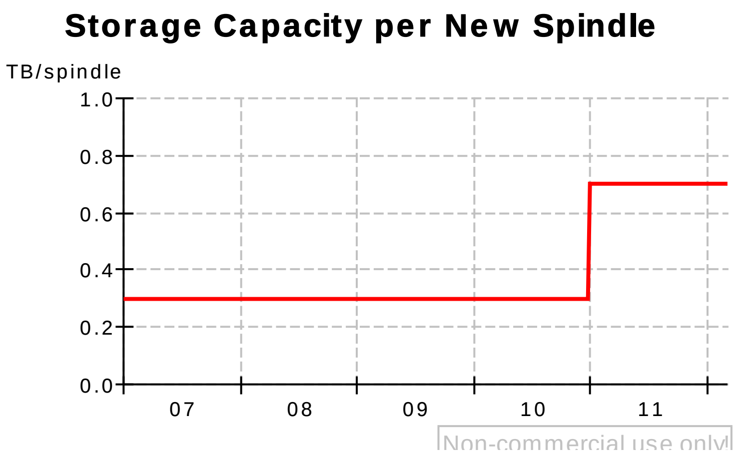


Figure 77 - Estimate for Storage per New Spindles

Average storage capacity for all of the company's spindles is declining and then surges back up in 2011, a behavior that the model successfully replicates (Figure 78). The reason for the fall is that the company has been acquiring relatively low-storage capacity spindles for some years since lack of storage has not been as much a point of concern as lack of I/O capacity. I/O capacity of spindles is outside the scope of this model. Then, starting with 2011, ABC is acquiring larger-storage capacity spindles, quickly increasing the average.

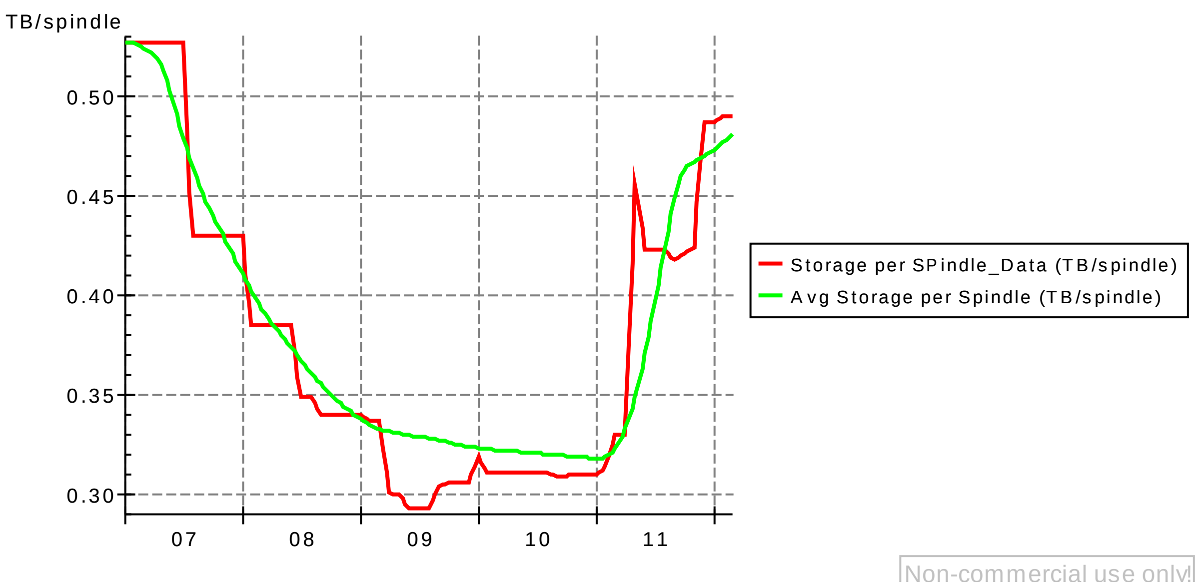


Figure 78 – Average Storage per Spindles: Simulated vs. Historical

Finally, it is worthwhile to investigate the dynamic behavior of another essential indicator: average amount of storage available per user. The reference mode comparison for this variable is plotted in Figure 79. This variable grows five times bigger during the simulation period, reflecting rapidly growing user storage requirements, rooted in both expected service level and change in data back-up technology. Discussion of this changing technology is beyond the scope of this report.

Figure 79 - Storage Capacity per User: Simulated vs. Historical

4.2.6. Power Demand – Simulated Behavior

Regarding total power demand, data reliability was an issue. For storage spindles, we had no data to begin with for total spindles' power demand. As for physical servers, the data that we had was not workable for two reasons: Firstly, it was data considering nominal, and not effective, power demand which we are interested in. And secondly, the data was secondary data generated based on primary knowledge of nominal power demand of new physical servers, using spreadsheet methods that cannot be completely trusted. The model uses the same primary data for 'Power Demand of New Physical Servers' and gives results different from data. Since the behavior of Total Physical Servers matched its reliable reference mode pretty well, and since data for power demand of new servers is reliable, model generated behavior must reflect reality closely enough and can be trusted. On these grounds, we are going to show the behavior of major power demand variables independently, rather than compared to any reference modes. Figure 80 shows simulated behavior for total power demand of spindles, total power demand of physical servers, and the sum of the two. In line with slight oscillations in numbers of physical servers and spindles, there are naturally slight oscillations in total power demand.

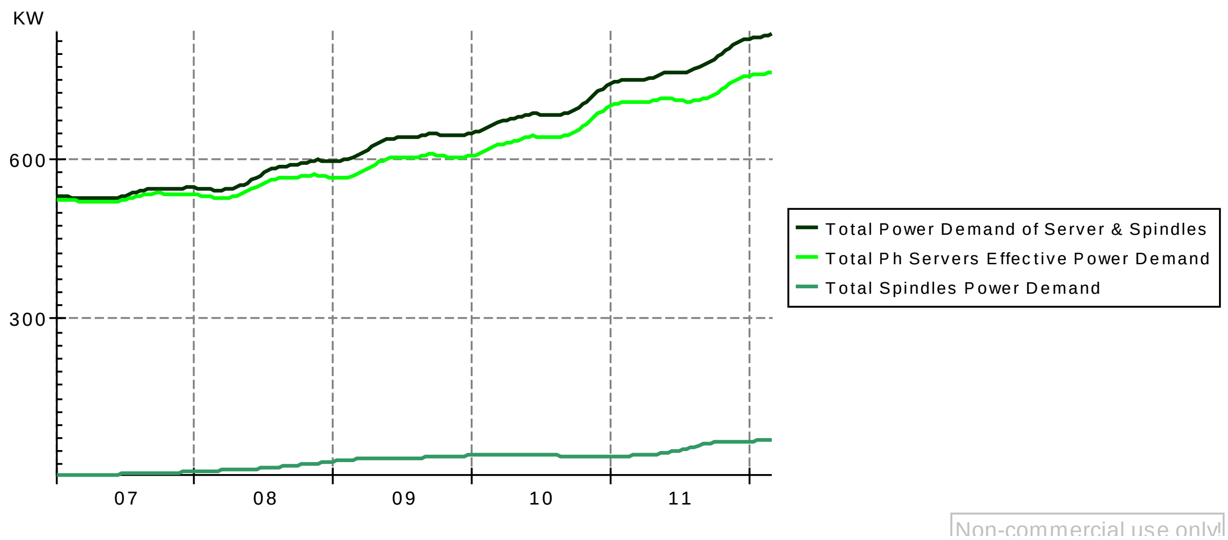


Figure 80 - Total Power Demand of Spindles (bottom graph); Total Effective Power Demand of Physical Server (middle graph); Total Power Demand of Servers and Spindles (top graph)

On the other hand, average power demand of physical servers is showing a declining goal-seek behavior, with newly installed servers utilizing higher technology to consume less power. Figure 81 depicts simulated behavior of average power demand of physical servers.

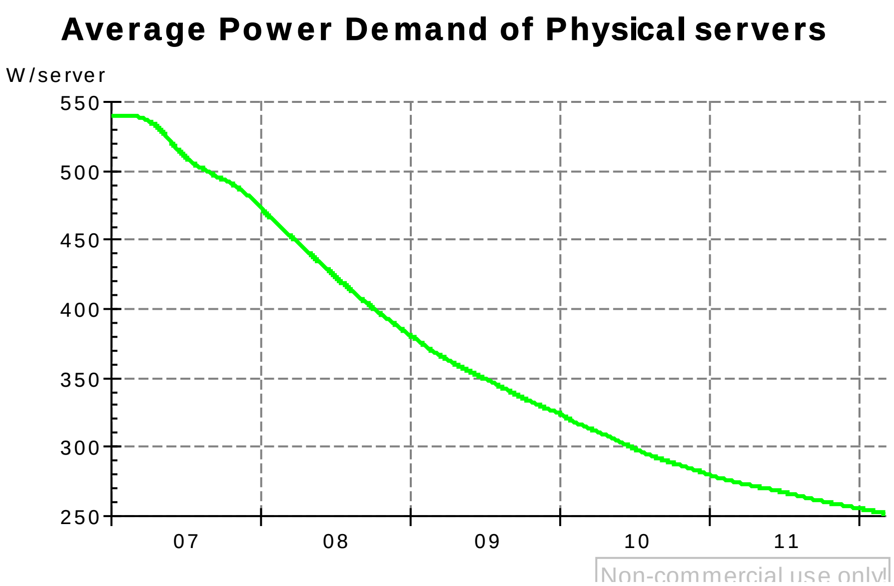


Figure 81 - Average Power Demand of Physical Servers: Simulated.

4.2.7. Users and Revenues Reference Mode Tests

The number of users in this model is generated by an internal exponential growth engine on the stock of 'Users', where the percentage of annual growth fraction is extracted from historical data. Therefore, the resulting simulated graph (Figure 82) will be a collage of yearly exponential growth patterns with different growth rates. These growth rates are very high in the beginning (around 42% for 2007 and 36%

for 2008), plummeting down to less than 3% during 2009, at least partly due to the global financial crisis, and striving to take off again after that, with 9.5% in 2010 and around 6% in 2011.

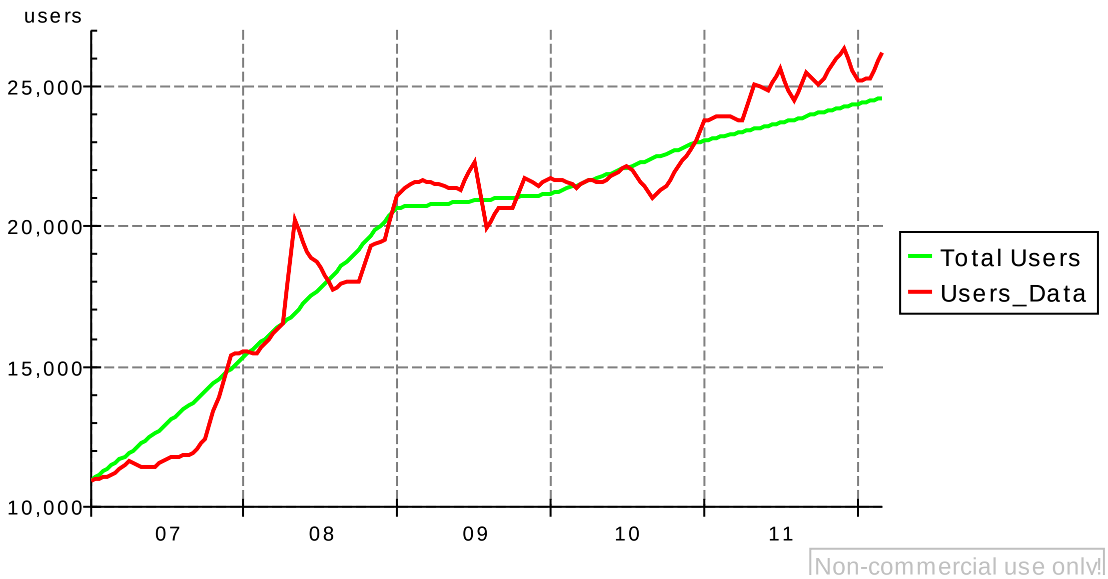


Figure 82 - Total Users - Simulated vs. Historical

Naturally, with this conceptualization, we are only able to replicate the general trend in behavior. The RMSE divided by run average is 4.9% and the correlation coefficient is 0.981.

In order to capture the dynamics of users more realistically, a proper 'market sector', capturing important market segments, conversion ratios from suspects to prospects and from prospects to actual contracts, and variables affecting these factors, different client sizes in terms of number of users, etc., needs to be developed and added to this model,. This can very well be an area of further development for this work.

At the moment, the annual growth rate of users is considered as a policy parameter for the future, a parameter that the user of the model can set to test different scenarios reflecting various growth conditions in the future. In the model, it is assumed that the company reaches its growth target with certainty. This is also not realistic. However, we need to take into account that this model is built with an operational mindset, the client being the Chief Technical Officer of the firm, and not with a market mindset. For a CTO, it is not unintuitive to consider the growth target as exogenous to the sector of the system that he/she manages.

Late in the project period, we received disaggregated time-series data for the two categories of users, full desktop user and distributed services users. The reference mode comparison tests for these two can be seen in Figure 83 and Figure 84. The 'FD Users' graph looks pretty much like the previous graph for total users, because for the most of the period the only type of users present have been FD users. DS users start coming in only in 2011, reflecting a change in market policy. This change of policy towards targeting client with distributed services users is likely to be reinforced in the future. In the growth sector of the model (section 3.4.2), we have included the parameter 'Percentage of Distributed Services

Users in New Contracts', which is set to historical conditions for the past, and can be set to any value in the future for policy testing.

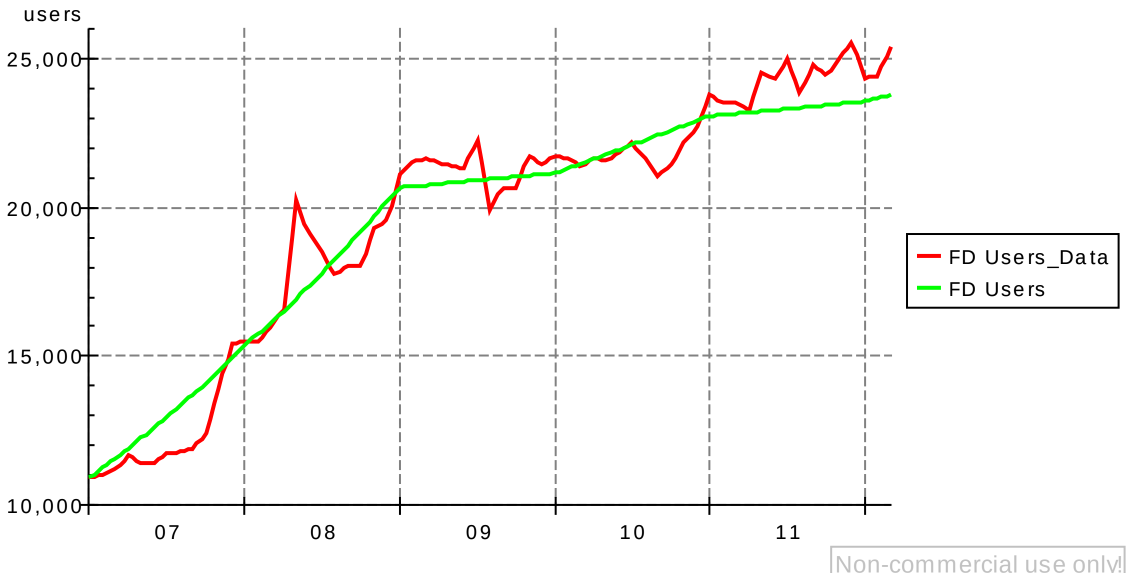


Figure 83 - Full Desktop Users: Simulated vs. Historical

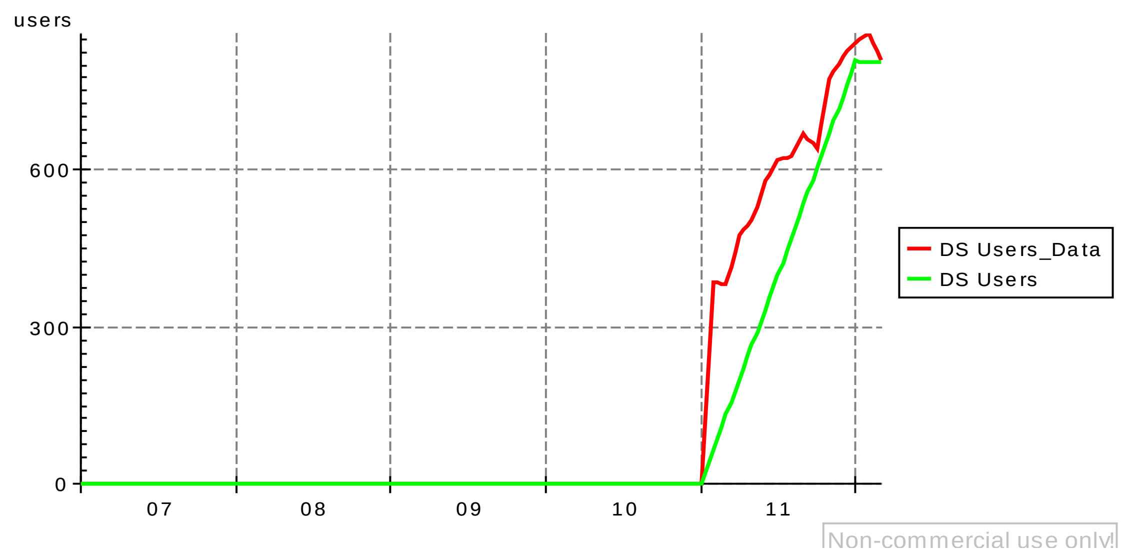


Figure 84 - Distributed Service Users: Simulated vs. Historical

Total monthly revenues generated by these users can be observed in Figure 85. As mentioned earlier, data considering price for new user contracts is not available. Therefore, the revenue variable had to be generated by trying to make an estimate for average revenue per new user that would give a satisfactory fit with revenue data. This estimate is shown in Figure 86.

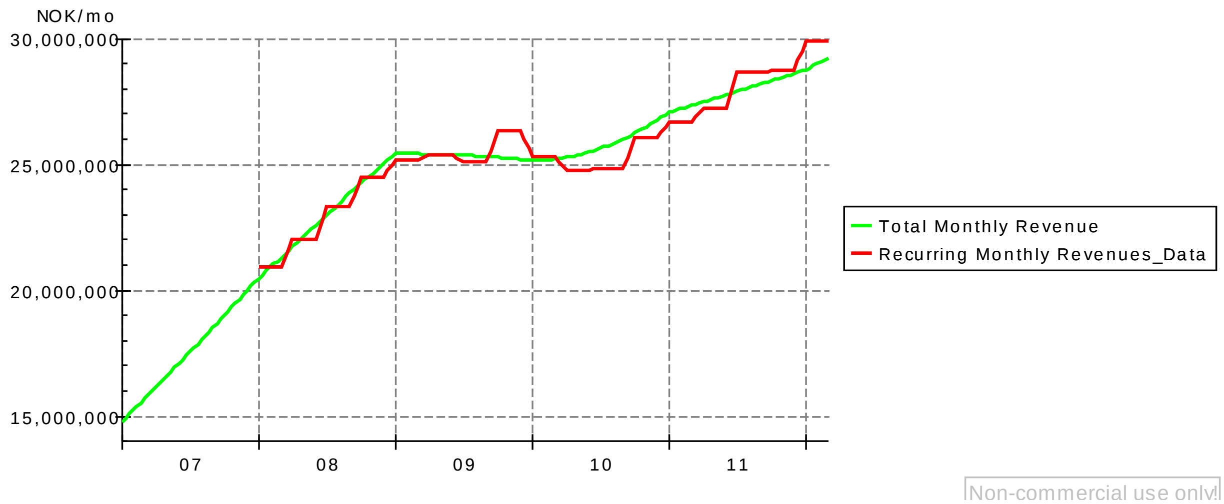


Figure 85 - Total Monthly Revenue: Simulated vs. Historical

According to this estimate, prices take a drastic dive between 2007 and 2010, when they rapidly rise again to more than compensate for the fall in the previous three years during only one year. This does not seem to be realistic according to the information elicited from the client. This information suggests that prices have been steadily declining in the past, but much more slowly than what is suggested in Figure 86. Therefore, this is concluded to point to an inconsistency between data for users and data for revenues. Since data for users has once been revised and refined by the client, it is reasonable to consider it relatively reliable. This seems to suggest that the client needs to re-evaluate the data provided for revenues.

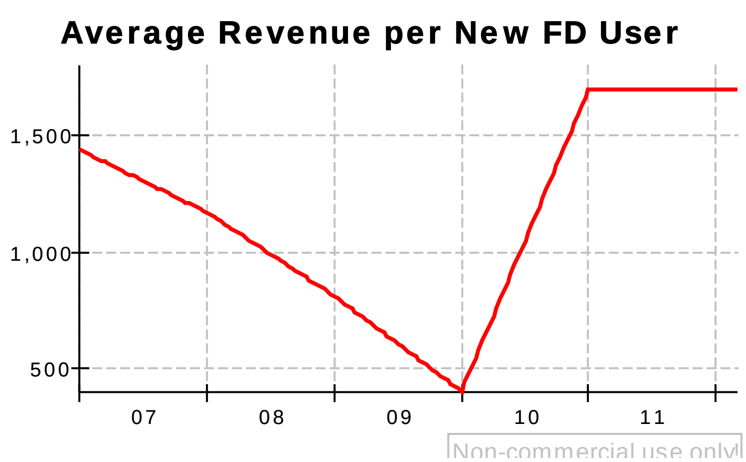


Figure 86 – Model Estimate for Average Revenue per New FD User

The client has been contacted about this probable flaw in data regarding revenues, but at the moment we have not received any reaction yet. Therefore, unfortunately, we have to keep this unrealistic estimate for average revenue per new user for the time being.

Chapter Five - Policy and Scenario Analysis

Now that we have established enough confidence in the ability of the model to endogenously replicate historical behavior of the real system, we are able to trust it with analyzing different policies and scenarios into the future. It should be noted that in this project, we are not dealing with a particular difficulty or problematic situation, and therefore, the deliverable is not going to be in the form of a single policy recommendation. Rather, the purpose is to explore different possibilities and help the client make more informed future projections. The model includes several parameters that can be considered either as policy levers or as scenario determinants. Through manipulating different combinations of these parameters, numerous scenarios can be analyzed. In this chapter, we are going to select a few different combinations of likely scenarios and policies for testing. This does not imply that these are the only possible tests. This model can be an analysis tool at the disposal of relevant decision-makers to continuously test different policies, while adjusting scenario parameters with the most up-to-date environmental information available.

Let us start by first going through the main policy and scenario parameters available to the user to manipulate.

5.1. Policy and Scenario Parameters

As used here, a policy parameter is one that is in direct control of the policymaker, while a scenario parameter is one that is mostly imposed on the organization by its environment. Scenario planning involves preparing appropriate measures for different scenarios. We shall start by pointing out those parameters in the model which reflect conditions outside the control of the organization, and involve uncertainty by nature.

5.1.1. Scenario Parameters

- *Power Demand of New Servers/Spindles*
  This parameter is to a limited extent in control of the decision-maker, who decides which brand/model of equipment to buy. However, power demand of new equipment is mostly determined by technology trends.
- *Price for new users*
  This depends on market trends. The overall average revenue per user is of course largely in control of the company, depending on which types of users are targeted. But the entry prices of the two different categories of users are determined in a competitive market.
- *Number of Cores per New CPU*
  Again, partly depends on purchasing decisions of ABC's managers, but is mostly a function of technology trends.
- *Storage per New Spindle*
  Same as above.

- *Cost Parameters (Server prices, Spindle prices, Energy, Rent, Interest Rate, etc.)*
  These are to a limited extent amenable through the choice of supplier/product type, but are mostly affected by environmental trends.

5.1.2. Policy Parameters

The following is a list of parameters representing important actionable levers at the decision-makers disposal:

- *User Growth Target*
  Setting a target is obviously governed by the management. Whether the company would eventually be able to reach that target is a different question. Within the limitations of this model, whatever growth target that is set is eventually reached. Therefore, the user should set only growth targets that are reasonable.
- *Server Lifetime*
  The operational manager can decide how long on average to keep servers, whether physical or virtual, in operation.
- *Ratio of New Physical Servers being Converted to Hosts*
  Along with the overwhelming trend in virtualization, this ratio, according to the client, is going to be close to one in the future.
- *Fraction of Stand-alone servers to Convert to Hosts*
  An existing possibility is to convert physical servers that have already started being operated as stand-alone to hosts later on in their lifespan. In the model, structure is built to reflect this policy, which is described as policy structure in 5.2.2.
- *Ratio of New Virtual Servers to New Physical Servers*
  This ratio, or the derived 'Share of Virtual Servers in Total New Logical Server Instances', is another decision that has to be made by ABC's management. This decision needs to be in harmony with the 'Ratio of New Physical Servers being Converted to Hosts'. In reality, maximum possible virtual servers that can be added depends on maximum virtual server capacity supported by hosts, a concept which is not incorporated in the model as discussed earlier.
- *Users per Logical Server for New (FD/DS) Users*
  The level of this variable is a determinant of the service level provided to users.
- *Storage per New (FD/DS) User*
  Another determinant of service level.
- *Percentage of Distributed Service Users in New Contracts*
  This parameter should be set according to future market policy.
- *Upgrade in Data center Power*
  This is the inflow described in 3.5, which captures the possibility of adding more power capacity to our data center.
- *Discard fractions for:*
  - *Ph Servers*
  - *Hosts*
  - *Spindles*

A possible policy to investigate is manually discarding older servers/spindles that possibly consume more energy than newer versions and provide lower capacity, and replace them with new ones, before their time has come. Separate discard structure is built into the model to reflect this possibility, as described in the next section.

5.2. Policy Structure

A couple of features in the model were not described in Chapter Three because they reflect strictly policy-related possibilities. In this section, we are going to quickly review these pieces of structure.

5.2.1. Discard Rates

So far we have assumed that all servers/spindles go through the full aging chain lifespan. What ABC can do in reality is to discard some of its servers/spindles from the pipeline before they reach the end of the pipeline. The management might see this as beneficial because of a possible drastic improvement in energy consumption or other features in new generations of servers/spindles. According to the following structure it becomes possible to discard a certain percentage of servers/spindles in a year, set by the user. This is formulated as a separate outflow of 'Discard Rate' for the final three age cohorts of equipment, as seen in Figure 87. The same structure is repeated for the next two final age cohorts, but not the first three age cohorts: it is assumed that ABC will not discard younger than four years old equipment. The same structure is also repeated for spindles. Obviously, along with each of these discard rates, other outflows have to be added to corresponding stocks in all dependent co-flows, such as power, CPU cores, etc, to represent the lost features associated with discarded equipment.

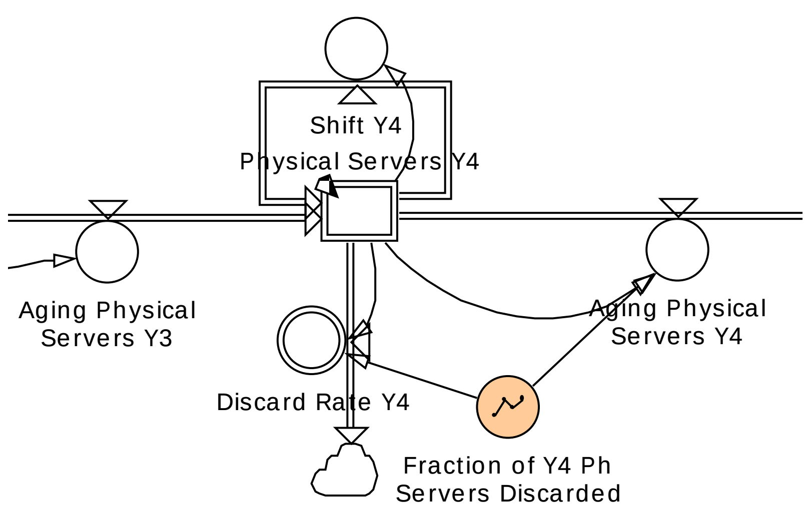


Figure 87 - Discard Rate

Therefore, the discard rate for each age cohort is formulated as the stock times the discard fraction, whose unit is percentage per year. Thus, the flow from the stock to the next cohort, which takes place at the beginning of each month, becomes the last element of the stock array times 'one minus the discard fraction'.

5.2.2. Converting Stand-alone Servers to Hosts

Another piece of structure not yet described is the flows reflecting the possibility of converting already operating stand-alone physical servers to hosts. Physical servers which are supposed to operate as hosts are usually set up to do so in the beginning of their operation. Nevertheless, it is also possible to convert an already installed stand-alone physical server to a host for virtual machines, if further virtual machine capacity is deemed necessary. The structure in Figure 88 captures this policy. It should be noted that the reverse, converting hosts to stand-alones, is not common practice in the business.

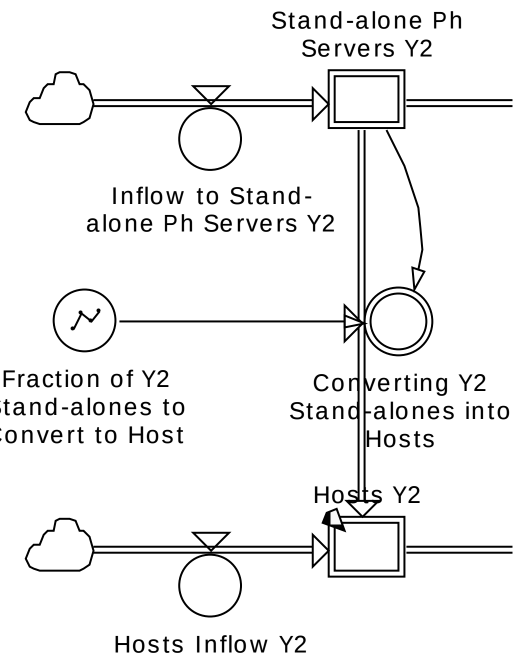


Figure 88 - Converting Stand-alone Servers to Hosts

Hence, the user can decide to convert a fraction of stand-alone servers residing in a particular age cohort to hosts during a certain year. This in-between flow is added to all age cohorts of servers, except the first one, since the first one already has an inflow of new hosts being installed. The conversion outflows from fourth, fifth, and final age cohorts in the stand-alones pipeline all flow into the fourth (final) cohort of hosts.

Naturally, along with each of these flows, there is a respective power demand flow between the pipeline of stand-alones' power demand and the pipeline of hosts' power demand.

5.3. Policy Tests

5.3.1. Momentum Policies into the Future

The first and most basic test that needs to be run is simply simulating into the future with all the current settings of the model kept unchanged. This test will let us foresee how important variables would develop in the highly unlikely case that all environmental variables and all policy variables remain constant.

Looking at future projections, the central question to be dealt with is how far into the future can the company grow within the current data center's environment, considering the power capacity available in the data center. In Figure 89 we can observe the projection of data center surplus power into the future, until 2020. Surplus power in the data center is being depleted with more servers and spindles being installed and taking up power (section 4.2.6).

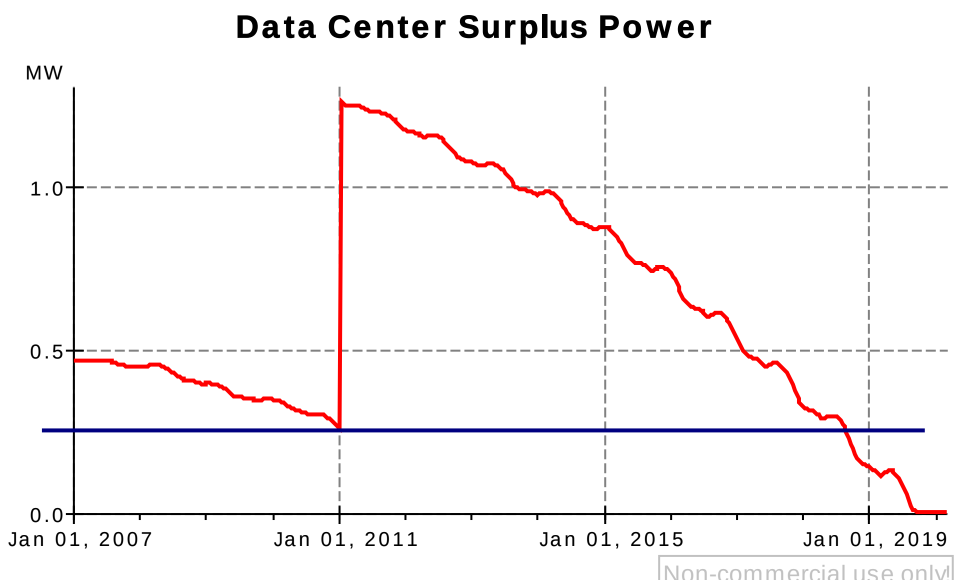


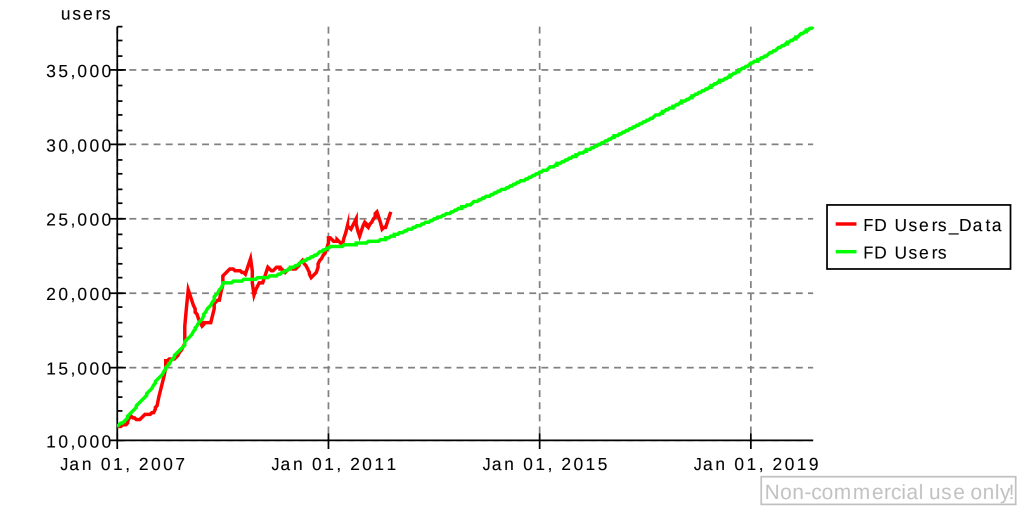
Figure 89 - Data Center Surplus Power: Projection into the future

The surplus power capacity, oscillating down in line with the slight oscillation in the number of physical servers, is foreseen to reach zero somewhere in 2020, all things constant. However, it is unlikely that the firm keeps growing unconcerned until the capacity hits absolute zero. For instance, historically, ABC upgraded the data center's power capacity in the beginning of 2011, when there was about 250 KW of power capacity left (the dark blue floor line in the figure). If we consider this amount as the safety margin that the company would like to keep, we can say that there has to be provisions planned for 2019, either as a second upgrade in power capacity if possible or renting and setting up new data center space. Since these provisions take around 12 months, according to the client's estimate, the effort should be initiated in 2018.

Figure 90 portrays the potential growth in the number of users and in revenues until 2020, under the current rate of user growth of 5.9% and the current average (model-estimated) revenue per FD user of 1700 NOK/user/month and per DS user of 400 NOK/user/month[1]. Assuming the limit of the data center

[1] These average price figures, especially the one for FD users, are too high compared to reality. Current prices according to the client are about 1250 NOK/user/month for FD users. However, we had to set prices for new users to unrealistic figures to be able to fit the reference mode for revenues, as discussed earlier, and for this policy test, we keep all parameters unchanged.

to be reached around the beginning of 2019, the limit in the number of users that ABC can support with the current settings will be around 35,000 and the maximum revenue to be reached will be slightly over 50,000,000 kroner per month (Figure 90). Furthermore, the maximum number of physical servers possible to install within the current situation would be slightly fewer than 9,000 (Figure 91), providing around 15000 logical servers in total.



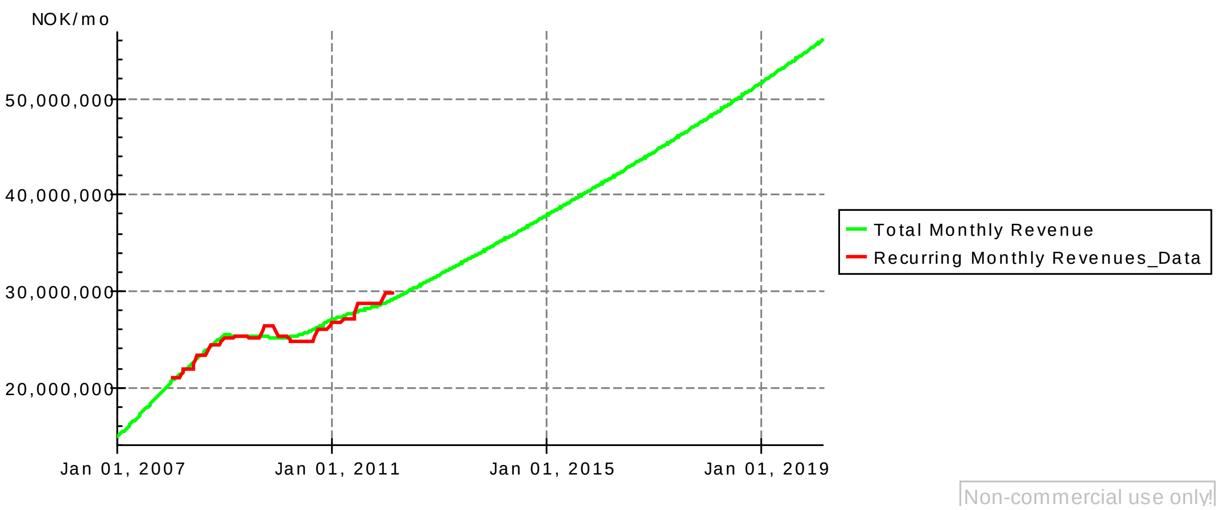


Figure 90 - Projected Users and Revenues until 2020

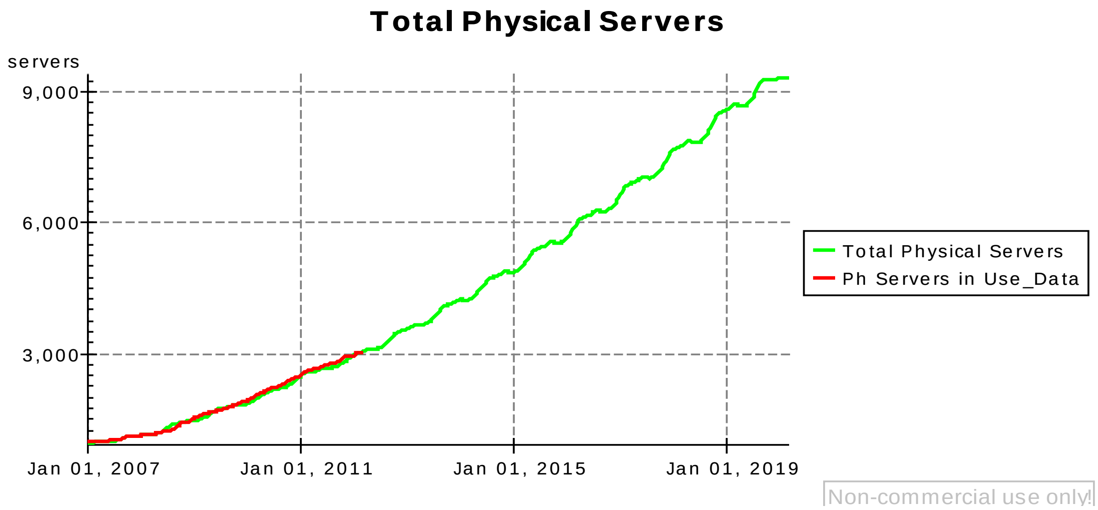


Figure 91 - Projected Total Physical Servers until 2020

Within the current state of the model, there is no restriction imposed on growth in the number of users. In principle, there are at least two factors in reality limiting infinite growth; a market factor and an operational factor. The market factor involves the ability of ABC to attract new clients, and is, as discussed earlier, outside the scope of this project. The operational factor, however, concerns whether or not ABC can serve the newcoming users in a timely manner. This is a question of whether or not the data center has adequate capacity to respond to the growing requirements, in terms of number of servers, brought in by new users. This restriction was not included in the version of model used for replicating historical reference mode, since such limitation has never really been in action in the past; ABC has historically foreseen a data center power shortage and planned for it in advance. However, now that we intend to use the model for future projections, new structure has to be implemented to reflect this limit to growth, in order for the model to behave realistically into the future.

The following piece of structure in Figure 92 was devised to capture the consideration of capacity limits when bringing in new users.

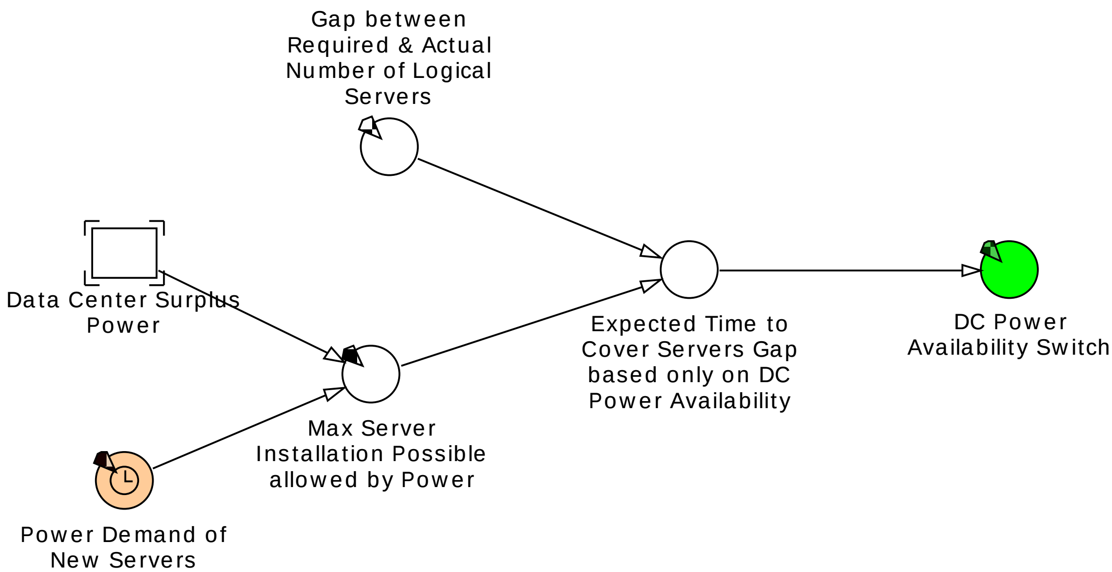


Figure 92 - Data Center Power Availability Switch

From our previous discussion, we remember the variable of 'Max Server Installation Possible allowed by Power', which is proportionate to Data Center Surplus Power and reversely proportionate to Power Demand of New Servers. The approaching of this variable to zero is taken to reflect a shortage in data center power. 'Expected Time to Cover Servers Gap based on DC Power Availability' is simply the gap in logical servers divided by maximum physical server installation with regard to power. The 'DC Power Availability Switch' is one when this Expected Time is shorter than four months, and zero otherwise. This implies that if, taking into consideration availability of power capacity in the data center, we do not expect to be able to close the gap in user requirements within four months then we switch off the data center power availability switch. This switch is then used in the Users sector to stop new users coming in when it is equal to zero[1]. Using this switch, we will be able to inflict a more meaningful limit to growth in the coming policy tests into the future.

5.3.2. Growth Rate Scenarios

The two most obvious scenarios to be studied are a relatively optimistic and a relatively pessimistic growth in demand for ABC's services in the future. Again, the first and foremost question to answer is when the limit of the data center power capacity would be reached. Having in mind the 2011 growth rate in the number of users of around six percent per year, we run the model once with fifty percent higher growth (9% per year) and once with fifty percent lower (3% per year).

5.3.2.1. High Growth Scenario

In this case, we would expect users to grow faster, effectuating faster growth in physical servers, and faster decline in data center surplus power. The result, shown in Figure 93, complies with this

[1] A more realistic way to formulate this would be to include a variable of Effect of DC Power Availability on User Inflow, which would be a graph function smoothly approaching zero as the Expected Time variable is going up. However, this zero-one switch is considered adequate at this point.

expectation, and shows that with a 50% higher growth rate, the data center limit is going to be reached around two years earlier.

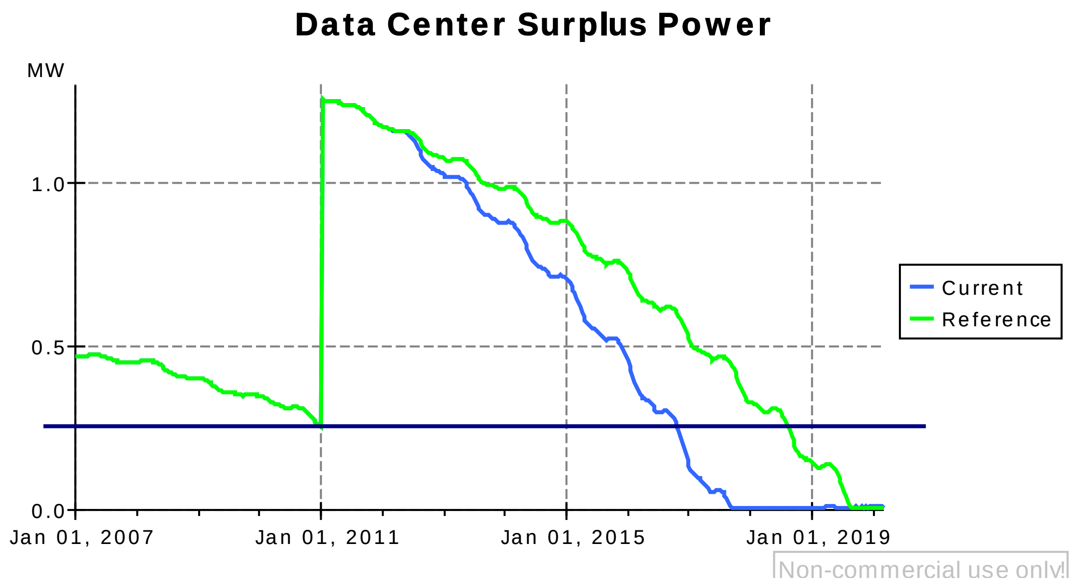


Figure 93 - Data Center Surplus Power: Base Policy Run (green), High Growth Scenario (blue)

With the new structure in place limiting infinite growth in user-base, the number of users in this scenario peaks rapidly at around 40,000 users, and starts a goal-seeking decline towards 35,000 after that since no more users are coming in but some users are flowing out along with ending contracts.

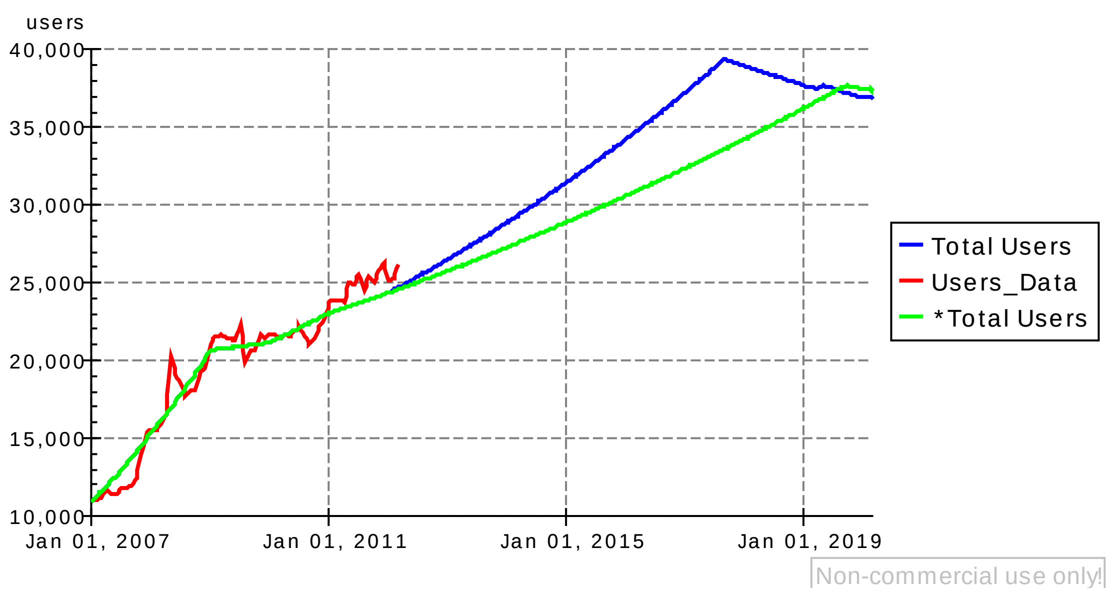


Figure 94 - Total Users: Reference Mode (red), Base Policy Run (green), High Growth Scenario (blue)

This analysis, however, assumes that the extent of using virtualization in ABC remains constant. Yet, the company can (and is planning to) ramp up the use of virtualization technology. Thus, we run the model one more time with the assumptions that one hundred percent of newly purchased physical servers are going to be set up for operation as hosts, and that the 'Ratio of New Virtual Servers to New Ph Servers' is going to be equal to four in the future, meaning that every new host will run four virtual machines. As seen in Figure 95, the decline in Data Center Surplus Power is going to be much slower under these circumstances.

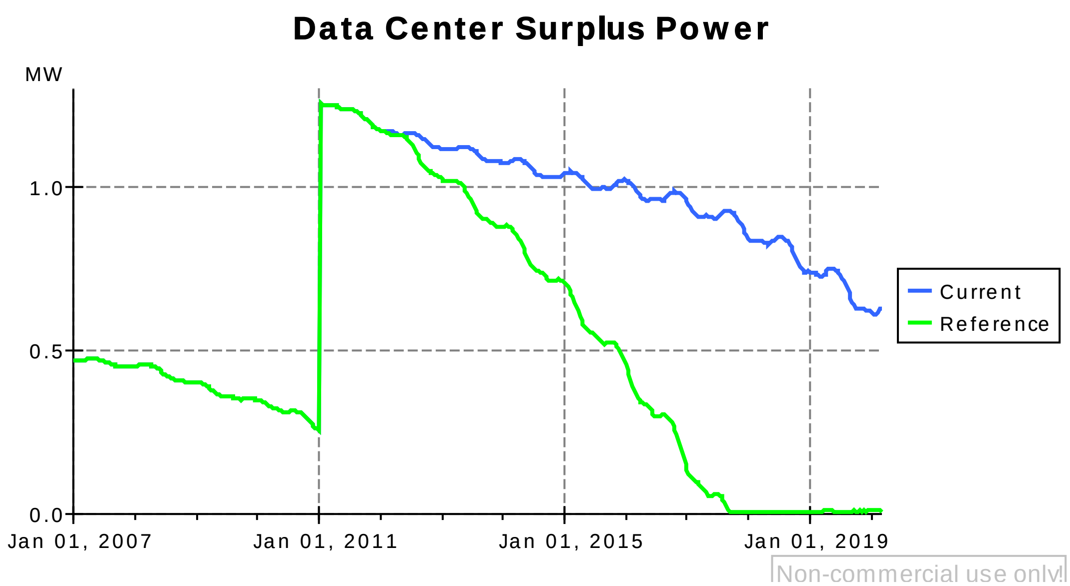


Figure 95 - Data Center Surplus Power: High Growth Scenario (green), Extensive Virtualization Policy (blue)

5.3.2.2. Low Growth Scenario

Supposing that due to unfavorable macroeconomic conditions, ABC would not be able to continue growing at the current rate, the expectation that the limit of our data center would be reached more slowly is almost trivial. Simulating the model under this assumption generates the expected behavior. In this case, capacity would not be a major concern. What ABC would need to do will be to stimulate more client contracts, and to be able to recommend policies in that area, we would need to build a market model.

5.3.2.3. Sensitivity of Data Center Power Run-out Time to Future Growth Rate

In order to make a more thorough analysis of the sensitivity of the point in time where the data center's power capacity runs out to the future growth rate in the number of users, we conduct a sensitivity analysis using the Latin Hypercube sampling method (which is a more advanced method than Monte Carlo) available in the Powersim Studio software. In this test, we set a normal probability distribution for future growth rate as the assumption variable, with an expected value of 5.9% (the value of the parameter for 2011] and a standard deviation of 3%, producing random values with the specified randomness over 30 runs . The result of this test is shown in Figure 96, including the 90, 75, 50, 25, and 10 percentiles. The 90 Percentile for example is a threshold curve with 90 percent of all time-series values over 30 runs lying under it.

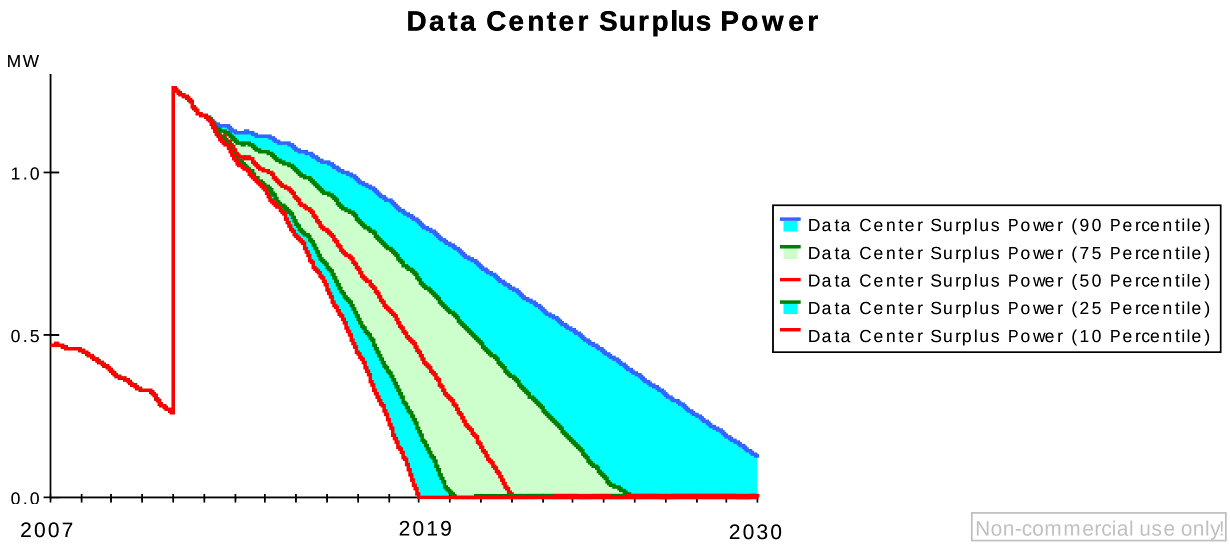


Figure 96 - Sensitivity of Data Center Power Run-out Time to Processor Future Growth Rate

The blue-green area is the 10-90% percentile range, with 80% of results of all simulations lying in this range. Likewise, 50% of results lie in the green 25-75% range. Not surprisingly, future growth rate is a highly sensitive parameter. With an unlikely very low growth rate, the data center can last beyond 2030, while with a quite likely growth rate of more than 8% per year power capacity may run out before 2019. This type of sensitivity testing gives realistic confidence ranges, instead of rather unrealistic point predictions, and can be very useful in planning.

5.3.3. Sensitivity of Data Center Power Run-out Time to New User Processor Requirements

In the model, the parameter reflecting processor requirements of new full-desktop users 'Users per Logical Server_New FD Users' is a sensitive parameter, which substantially changes the pace of growth in number of physical servers. This parameter is basically a policy parameter which has to do with the desired level of service provided to users. Furthermore, it is also decisive on the length of time over which our data center's power supply can respond to our needs. To test this sensitivity, we conduct a similar sensitivity analysis as in the previous section, setting a normal distribution for the 'Users per Logical Server_New FD Users' as the assumption variable, with an expected value of 1.6 [users/server] (the estimated value of the parameter for 2011] and a standard deviation of 1 [user/server]. The result of this test is shown in Figure 97.

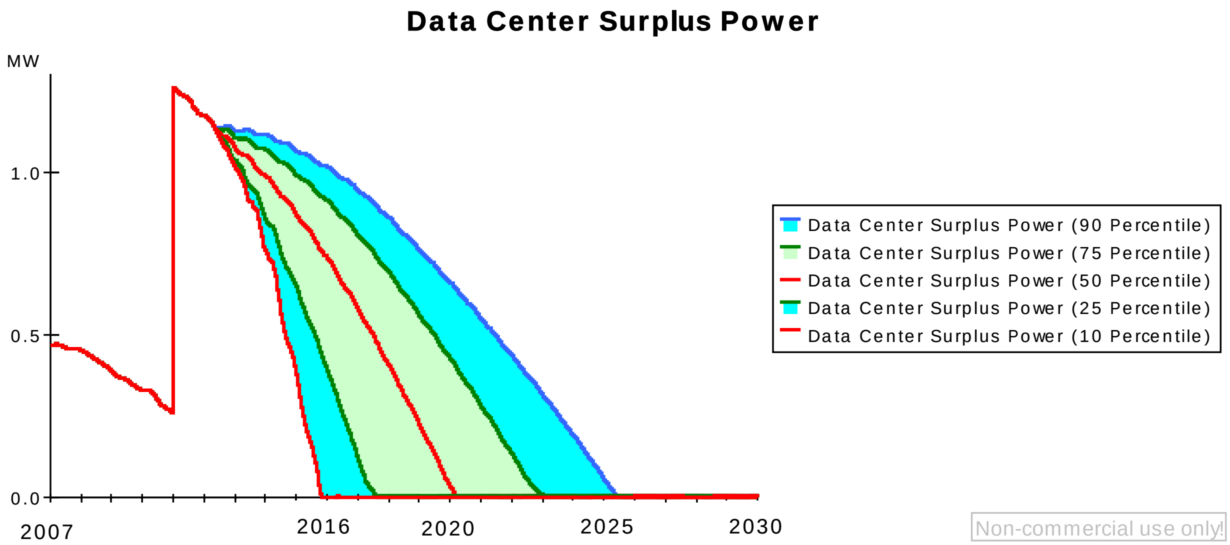


Figure 97 - Sensitivity of Data Center Power Run-out Time to Processor Requirements of New FD Users

As observed, randomness in the studied parameter can produce up to around ten years of difference in the expected time for the data center to run out of capacity. This is, of course, not surprising because a slight change in requirement *per* new user, which is a flow rate, can make a substantial change in total requirements, which is a stock, over time. Therefore, it is essential for the user of the model to expend maximum effort in calibrating this parameter with a high degree of precision. Alternatively, it is recommended that the client spends more time with the modeler to conceptualize this part of the model in a way which is more robust and less sensitive to parameterization.

5.3.4. Market Policy

At this stage, we are going to have a look at a combination of a higher growth scenario (10% per year) together with a redirection of market strategy towards signing contracts with more clients with distributed services users. In 2011, sixty percent of new users were DS users, but in the first three months of 2012, ABC has apparently switched back to bringing in only full desktop users. Here, we assume that the 60% share for DS users continues after 2011.

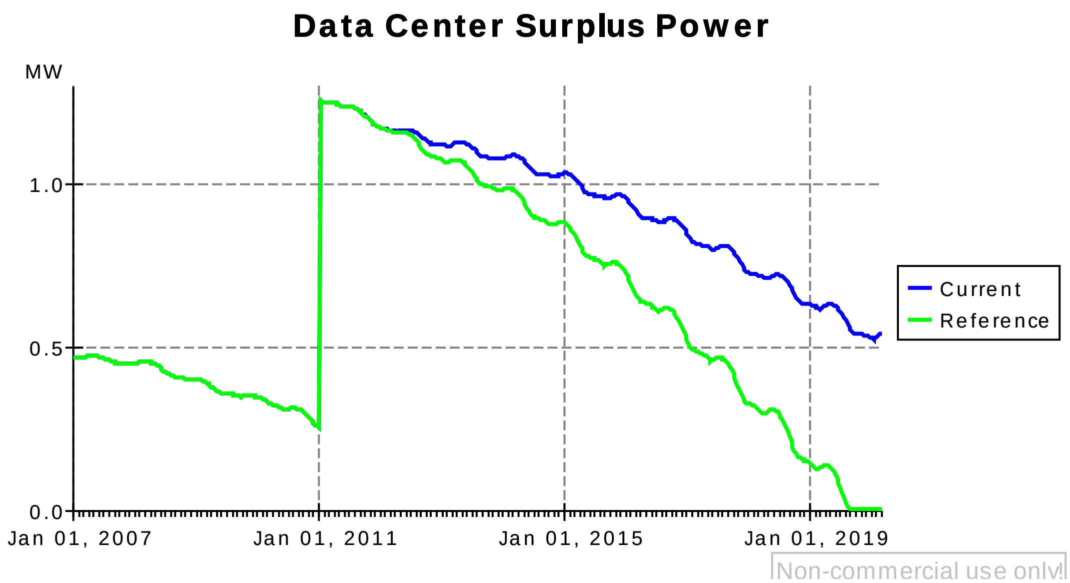


Figure 98 - Data Center Surplus Power - Base Policy Run (Green), Low Growth Scenario (Blue)

In this case, we will again observe that Data Center Surplus Power would run out later than when we have high growth and an inflow of only FD users. However, this longer lifetime of the data center does not come without a price. Naturally, with a higher percentage of DS users in new contracts, we are going to have lower average revenue per new user, and a slower growth in our revenue (Figure 100). Nevertheless, interestingly enough, the two graphs almost meet towards the end of the simulation. This is because ABC's growth is going to be more 'sustainable' with DS users which have lower requirements. With only FD users, the company will have to install servers at a high rate and will run out of capacity pretty soon. From then on, the user base is going to slowly decline towards steady-state stability.

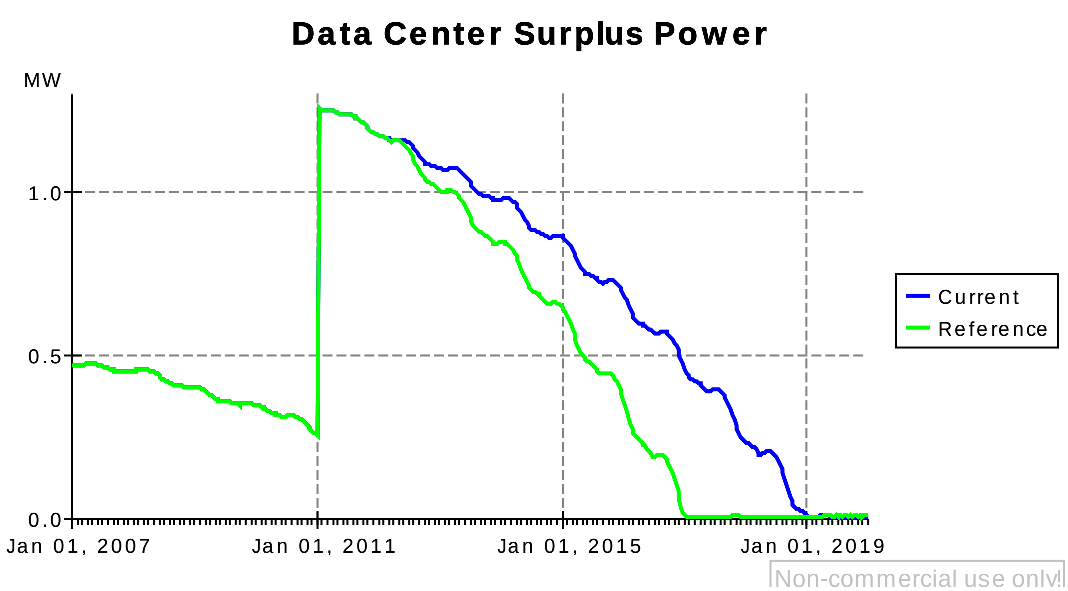


Figure 99 - Data Center Surplus Power, High Growth: 100% FD User Inflow (green), 40% FD User Inflow (blue)

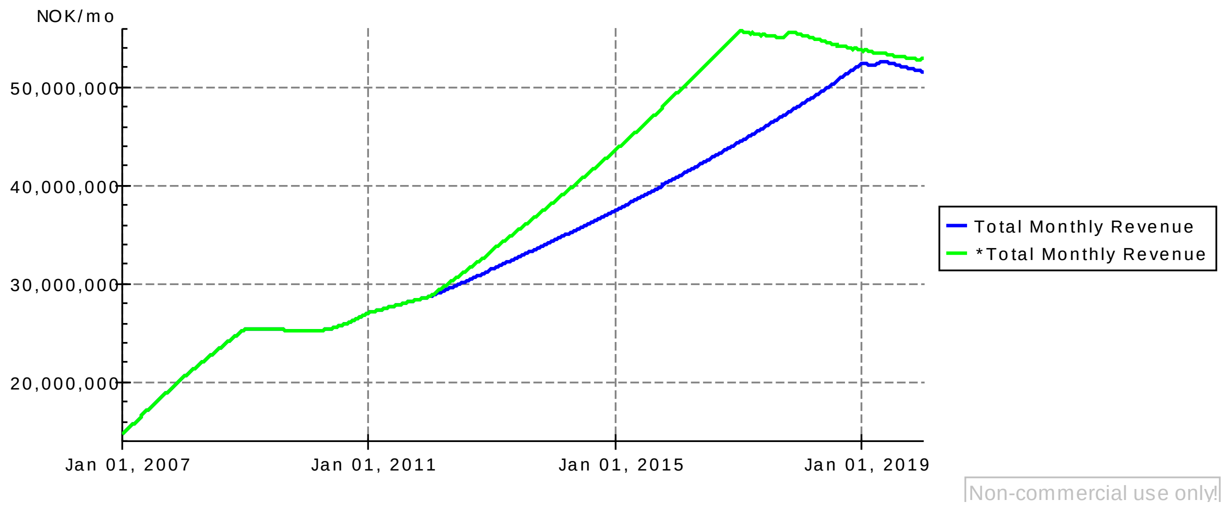


Figure 100 - Total Monthly Revenue, High Growth: 100% FD User Inflow (green), 40% FD User Inflow (blue)

5.3.4.1. Sensitivity of Data Center Power Run-out Time to Future Share of DS Users in New contracts

Same as before, in order to make a more thorough assessment of the level of sensitivity of our main problem variable, namely 'Data Center Surplus Power', to future market policy, we conduct a sensitivity analysis using 30 simulations with an expected value of 60% and a standard deviation of 30% for the share of distributed-services users in future contracts. Results are shown in Figure 101. While the model is not as sensitive to this parameter as the two parameters tested before, we can still see that large deviations can lead to deviations of around five years in the year in which power capacity runs out.

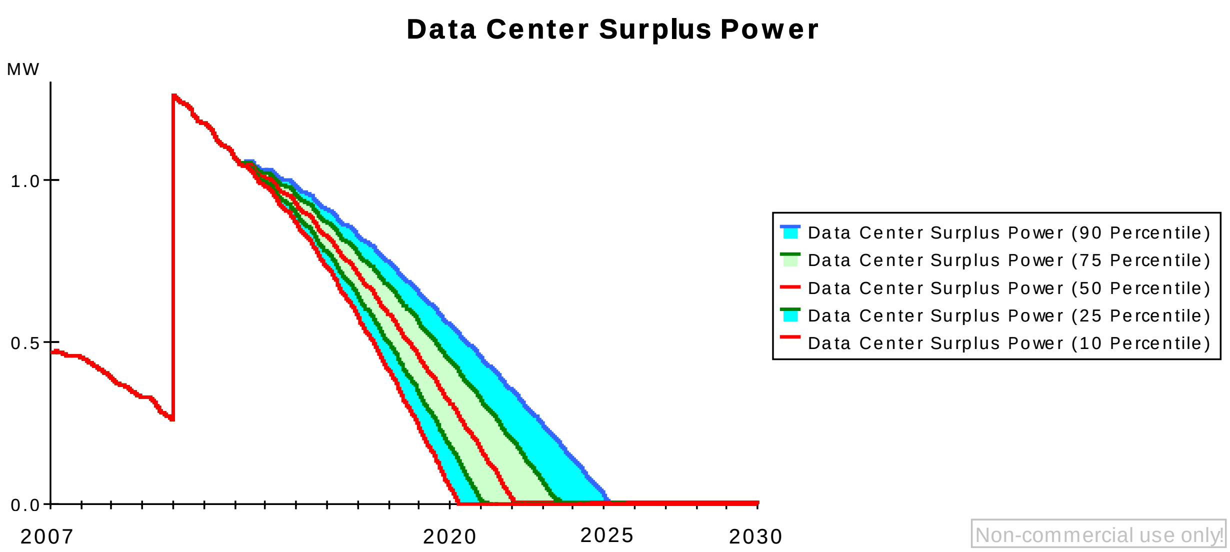


Figure 101 -Sensitivity of Data Center Power Run-out Time to Future Share of DS Users

5.3.5. Step-up in New User Processor Requirements coupled with HR Scarcity

In this section we are going to study a hypothetical scenario with rich dynamic implications. The scenario is based on two independent assumptions; one regarding processor requirements of new users and another concerning time to hire new server installation engineers.

At the first step, let us start our analysis by implementing the first assumption, a step increase in processor requirements of new users coming into the user base. Such scenario might happen due to increased competition in the industry. Until 2011, the estimated processor requirement of new users was around two thirds of a logical server per new FD users and a third of that for new DS users (In other words, on average, 1.6 new FD users or 4.8 new DS users shared one logical server). In this scenario, we assume that, starting from 2012, processor requirements of new users undergoes a step-up of around 60%, to reach one logical server for new FD users, and a third of a logical server for new DS users.

The expected outcome of this scenario should be an accelerated rate of server installation and an accelerated growth in number of servers, resulting in a fast decline in data center surplus power, which is exactly what the model predicts according to Figure 102 and Figure 103. But this data center power limitation is not the point of this test. Therefore, for the rest of this scenario test, we leave the 'DC Power Availability Switch' on all the time, to be able to solely inspect the effect of HR scarcity.

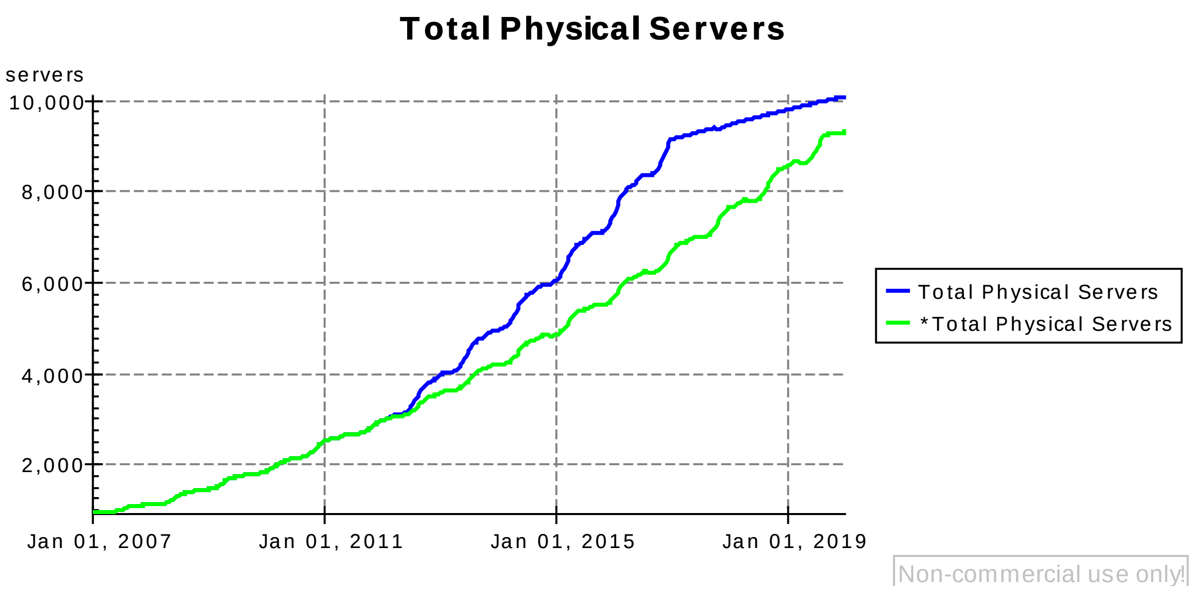


Figure 102 - Total Physical servers: Base Policy Run (green), Step-up in User Processor Requirements (blue)

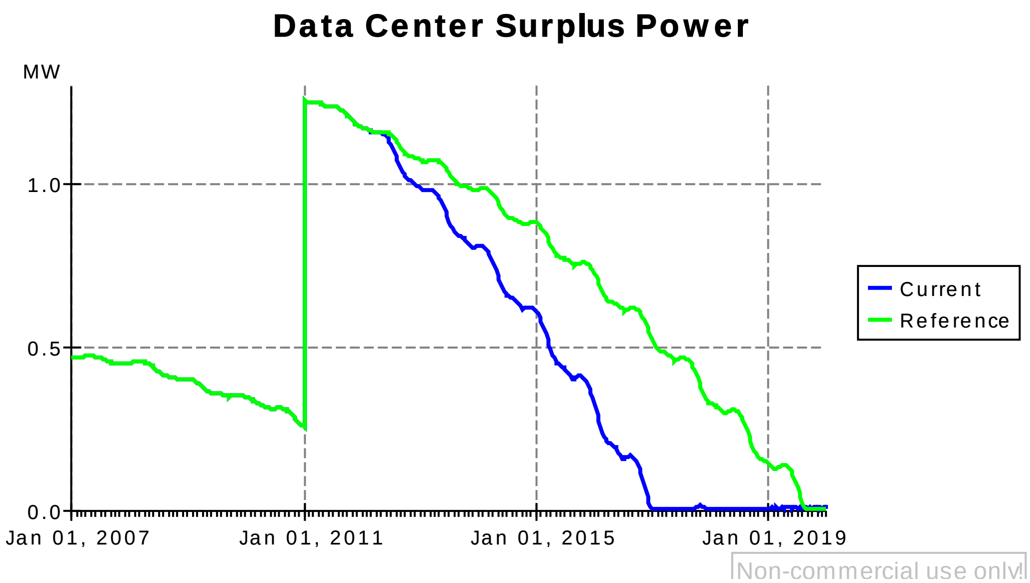


Figure 103 - Data Center Surplus Power: Policy Run (green), Step-up in User Processor Requirements (blue)

What we are interested in in this test is to explore the ramifications of such future potentiality on HR. As can be observed in Figure 104, the rate of server installation increases significantly with the recent development in new user processor requirements, and as a consequence of that, we are going to need more and more installation engineers to handle the increased demand (Figure 105).

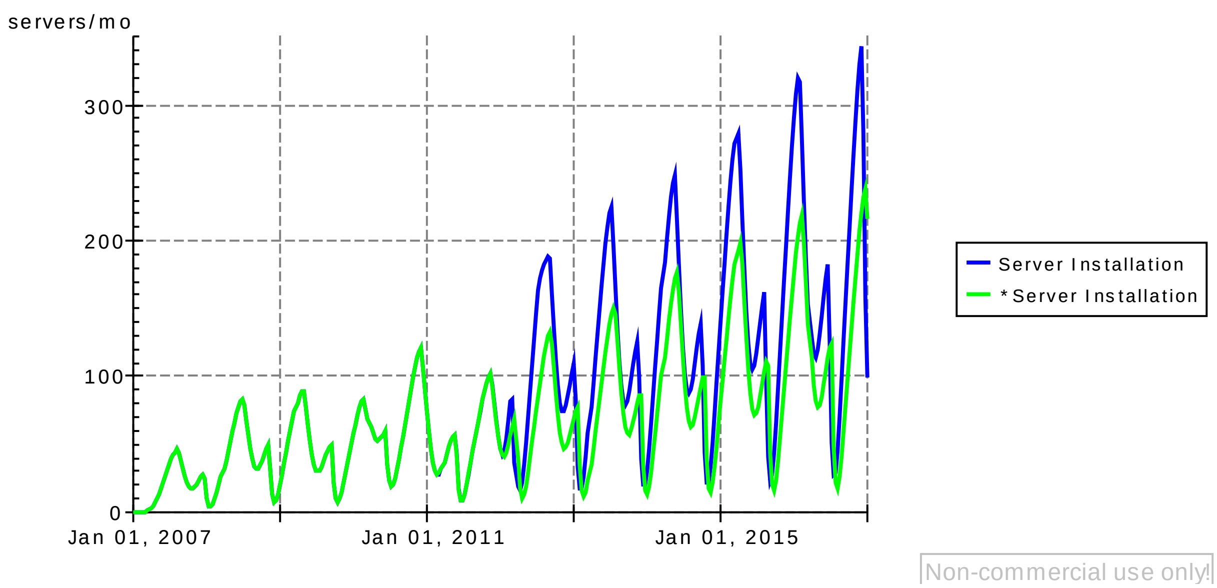


Figure 104 - Server Installation: Base Policy Run (green), Step-up in User Processor Requirements (blue)

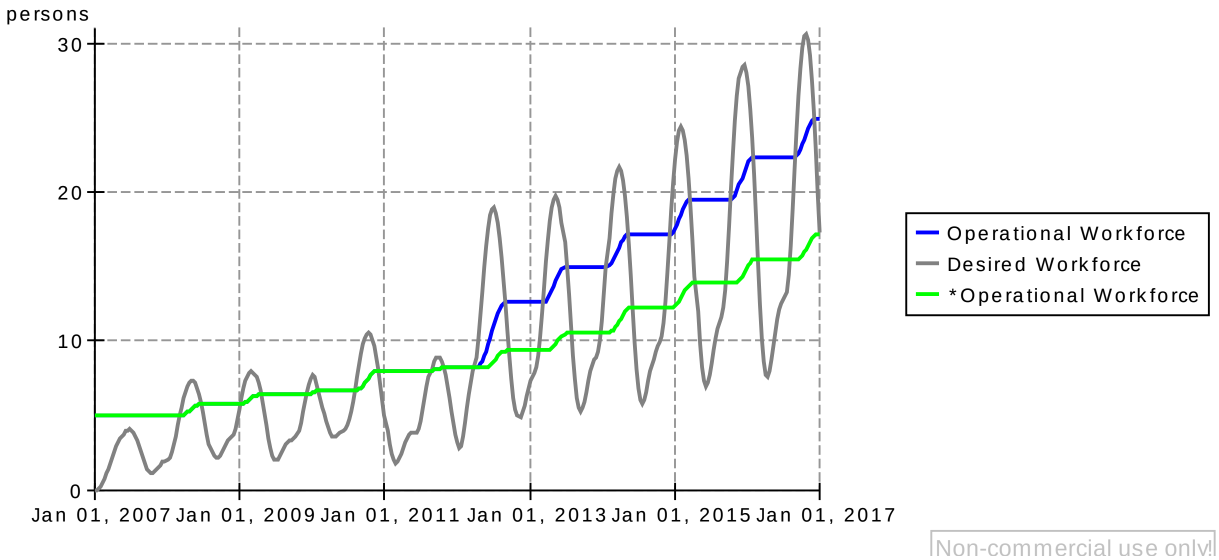


Figure 105 - Operational Workforce: Base Policy Run (green), Step-up in User Processor Requirements (blue) - Desired Workforce: Step-up in User Processor Requirements (grey)

In Figure 105, we can observe that operational workforce grows much faster in the scenario where there is a step-up in new user processor requirements (blue curve). We can also notice that the blue curve is agile in tracing the grey curve, the desired level of workforce which oscillates in line with oscillations in server installation. This agility is because 'Time to Hire' new installation engineers is set to six months, which is relatively short.

Now, picture a situation where, coincidentally, server installation engineers become very scarce in the job market at the same time, and ABC is not able to hire any new engineers for a couple of years. This situation can be simulated by setting Time to Hire to a very high value, e.g. 99 months, during 2012 and 2013. What would be the consequence of this sequence of events on the company's growth?

To explore this, obviously we need to implement in the model the effect that installation capacity has on growth. For this purpose, we include a structure similar to the 'Data center Power Availability Switch' which we introduced in Figure 92, this time for human resources (Figure 106).

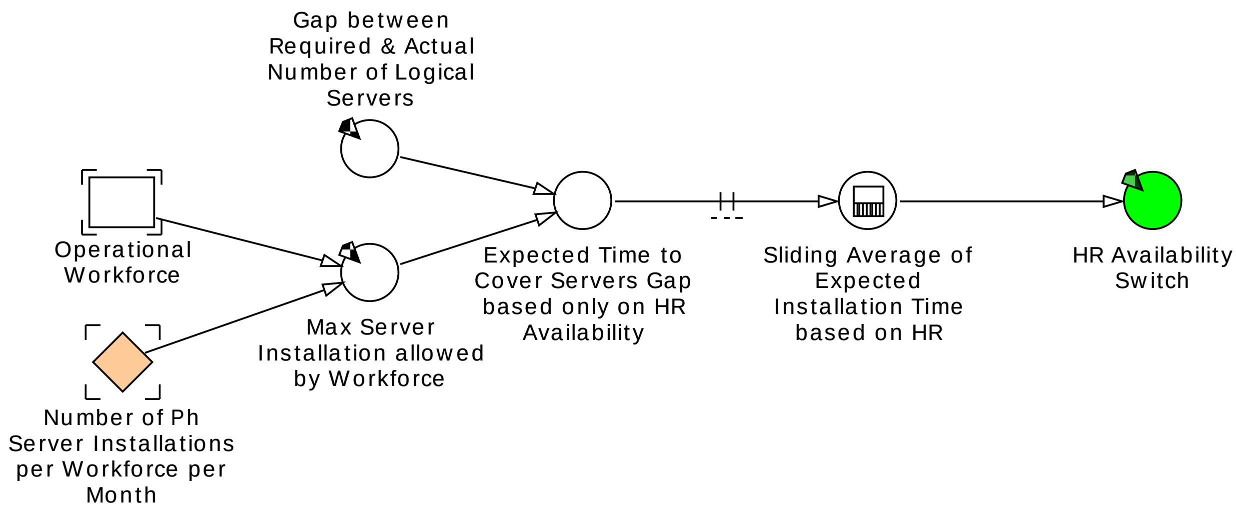


Figure 106 - HR Availability Switch

The HR Availability Switch is turned off if the average 'Expected Time to Cover Servers Gap based only on HR Availability' during the past three months rises beyond three months, a length of time which represents an 'unacceptable' customer response time. In this case the inflow to users is suspended. Of course, this conceptualization is not completely representative of reality. In reality, engineers can work harder under pressure, and thus the number of installations that an engineer can do in one month is not a constant. On the other hand, excessive work pressure can create burnout. These dynamics are not captured with this highly simplistic HR model. Developing the HR sector, possibly in collaboration with the HR manager of the company, can be a promising area for future work.

Figure 107 displays the graph for Total Users in the base run and in the run with this scenario's assumptions. Keep in mind that in this scenario we have cancelled the effect of running out of data center power on user growth. As we can see, HR shortage can potentially become an important limit to growth. In the causal loop diagram in Figure 108 we can see the interplay of these causalities. When processor requirements of new users (southeast corner of the CLD) go up, through widening the gap in processor requirements, it initiates two loops with opposite polarity at the same time. The balancing loop is put in action immediately, with the gap raising the expected time to cover the gap, which inhibits the inflow of users if it exceeds a certain limit. In other words, this is at the moment a 'conditional' balancing loop, which would turn into a normal balancing loop if the conditional function in the HR Availability Switch were made fuzzy instead of zero-one. On the other hand, the widening gap also plays a role in a reinforcing loop. A wider gap leads to higher Need-based Server Installation, higher Desired Workforce, and higher actual Operational Workforce with a delay, which eventually brings down the expected time to cover the gap, releasing the User Inflow again, and further widening the gap.

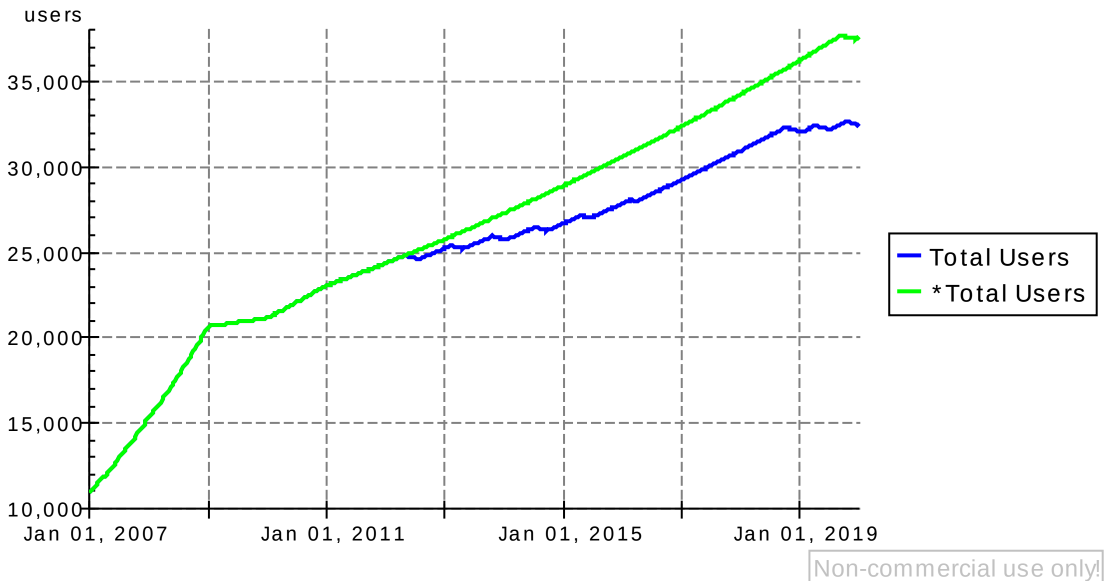


Figure 107 - Total Users: Base Policy Run (green), Step-up in User Processor Requirements coupled with HR Scarcity (blue)

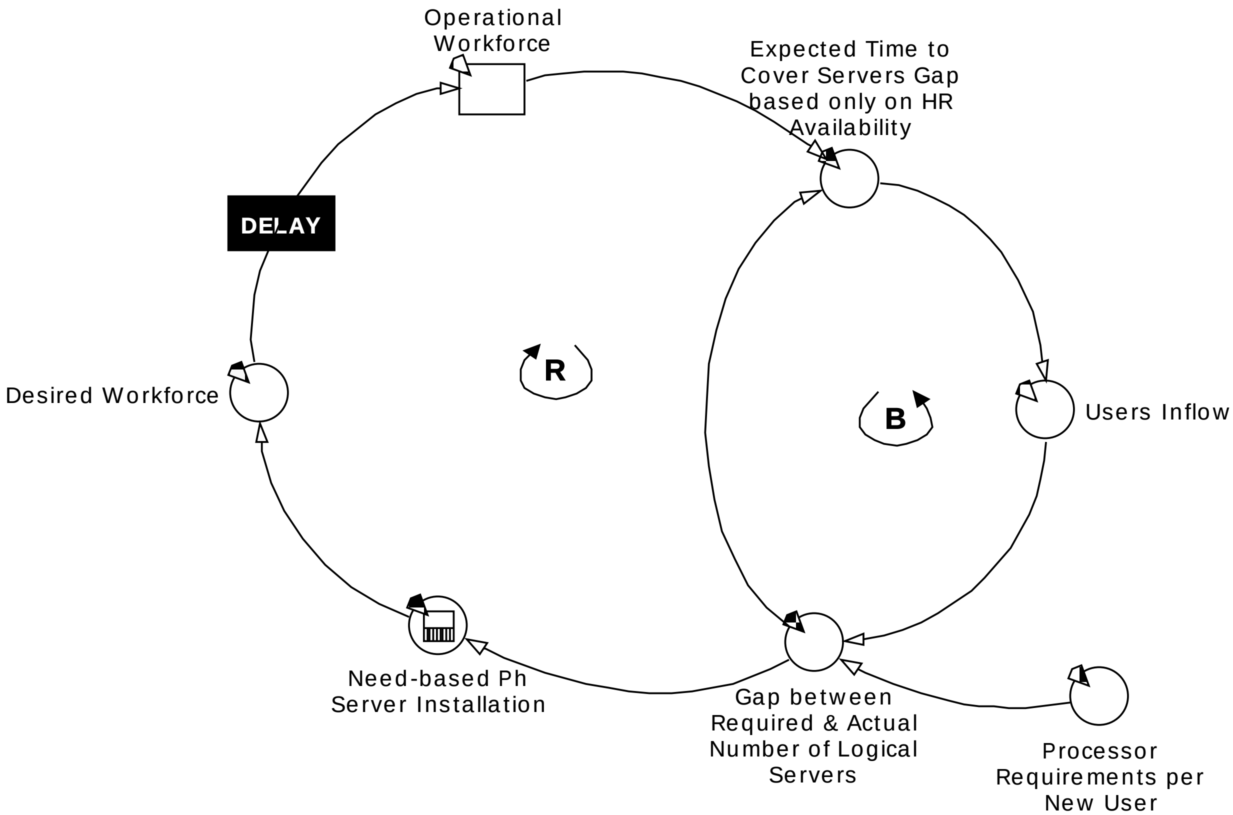


Figure 108 - Causal Loop Diagram, Human Resources

The delay in the reinforcing loop is an important factor, which turns this HR shortage into a potential limit to growth. This is because the reinforcing loops kick in with a delay relative to the balancing loop.

Therefore, the balancing loop hinders growth initially, before the reinforcing loop catches up to propel the growth engine again.

5.4. Conclusion, Discussion, and Future Work

The goal of this thesis was to build a System Dynamics model reflecting the operations of a cloud computing data center. The purpose of the model was to help the client of the project in medium-term data center capacity planning. The main research question was to estimate the time at which the data center in question would run out of power capacity, under different policies and environmental outcomes. This was a critical question for the client both since the data center's capacity to supply power to its servers and storage spindles is currently the most important limit to the company's growth, and since setting up a new data center or upgrading the power capacity of a current data center is a very time-consuming and cost-intensive effort.

The SD model was built in a period of around four months with regular interactive interviews with the client, who was the Chief Technical Officer of the firm, where model progress was presented regularly and questions were asked based on model structure and behavior, in order to complete the structure and analyze the behavior against known historical data. The model-building process entailed substantial learning for the modeler and for the client, according to his own testimony. The learning for the client included seeing their own business in a structured systemic visual representation, studying the development of business variables over time, discovering important performance indicators to track, and finding out about inconsistent or missing data. The conclusive added value of the model will of course happen when the client eventually uses the model to explore various policies and scenarios.

In this report, after developing confidence in the model with partial and overall tests where the model mostly succeeded in replicating important reference modes, we set out to run a number of sample policy and scenario tests. It should be noted that the subject of this thesis was not to recommend a policy to improve a problematic situation, as in many System Dynamics modeling theses. The subject was, rather, to provide a tool to help the manager make better projections into the future. Therefore, the result of this work is not in the form of a single recommended policy, but a sort of an engine for a potential flight simulator for ongoing policy analysis. The policies and scenarios analyzed in this chapter are but a few samples of a virtually unlimited number of possibilities.

In the last chapter of the report we observed how the answer to the client's central question is sensitive to several parameters internal or external to the firm, such as future growth, future market policy, and future desired level of customer service. We also noted how human resources – engineers responsible for the installation of servers, for instance – can potentially prove to be an important limit to growth in the future.

In conclusion, this thesis demonstrates the usefulness of the System Dynamics methodology in policymaking for data centers in the IT industry. Data center operations contain an intimidating degree of dynamic complexity, characterized by numerous aging chain and co-flow structures, delays, accumulations, and feedbacks. A compelling example of this complexity was given in 4.2.4

demonstrating the difficulty of intuitively estimating “virtual cores per server” for different age cohorts of servers. With all this dynamic complexity, System Dynamics modeling can be considered superlatively useful in comparison with other business modeling methods.

The model at its current state is still far from perfect. There are numerous areas for significant and meaningful improvement. Among the limitations and areas for improvement of this model are the following:

- As discussed in 4.1.6. The concept of capacity of hosts in terms of number of virtual machines supported is not modeled. This is an essential concept needed to model the inflow of hosts in a truly endogenous fashion.
- Due to lack of data, the RAM sector of the model was not calibrated, while RAM is a critical attribute of servers and is sometimes what is driving the installation of new servers. In other words, the current conceptualization of making server installations dependent only on processor requirements of users is incomplete since it is often RAM requirements that necessitate installing new servers.
- Storage spindles are not fully characterized. I/O capacity of spindles for instance is not modeled. According to the client’s knowledge, the area of providing storage space in cloud computing is so vast it could be a title for another thesis.
- The same holds for the market sector. There can be an aging chain built for users, where the stocks are ‘suspects’ turning into ‘prospects’ turning into cohorts of ‘clients’ at different stages of contract age. Clients can have different attributes as co-flows such as size of each client in terms of users, type of users for each client, etc. This is also a promising area for extending this work, possibly in collaboration with the market department of the firm.
- The data center sector can be modeled with much more detail. Space area was considered secondary in importance and was not modeled. Also, the concept of ‘racks’ that hold servers and spindles is not included. Data centers can be configured at different density levels in terms of rack concentration. This has implications on maximum number of servers that a rack can hold. This also seems to be a rich area to explore different policies with SD modeling.
- The ‘Accounting & Finance’ sector of the model is quite crude. The HR structure modeled within this sector can very well be developed into a whole detailed sector. Human resource dynamics are widely studied using System Dynamics, and SD has shown to be a promising methodology in this field. Other parts of the Finance sector could become more representative of reality, possibly in cooperation with the finance department of the company.
- This model would eventually prove most useful together with a user-friendly interface; a ‘management flight simulator’. Such interface has not yet been built in this project, and is a worthwhile effort for the future.

References[1]

Barlas, Y. (1996). Formal aspects of model validity and validation in system dynamics. *System Dynamics Review , 12* (3), 183-210.

Buyyaa, R., Yeoa, C. S., Venugopala, S., Broberg, J., & Brandic, I. (2009). Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. *Future Generation Computer Systems , 25*, 599-616.

Christian Belady (Microsoft). (2008). *Green Grid Data Center Power Efficiency Metrics: PUE and DCiE.* Retrieved from http://www.thegreengrid.org: http://www.thegreengrid.org/~/media/WhitePapers/White_Paper_6_-_PUE_and_DCiE_Eff_Metrics_30_December_2008.pdf?lang=en

Daniels, J. (2009). Server Virtualization Architecture and Implementation. *Crossroads , 16* (1), 8-12.

Mell, P., & Grance, T. (2011, September). *The NIST Definition of Cloud Computing.* Retrieved from National Institute of Standards and Technology (http://www.nist.gov): http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf

Wang, L., Kunze, M., Jie, T., Castellanos, A. C., Kramer, D., & Karl, W. (2008). Scientific Cloud Computing: Early Definition and Experience. *The 10th IEEE International Conference on High Performance Computing and Communications.*

Wang, L., von Laszewski, G., Younge, A., He, X., Kunze, M., Tao, J., et al. (2010). Cloud Computing: a Perspective Study. *New Generation Computing , 28*, 137-146.

Zhang, Q., Cheng, L., & Boutaba, R. (2010). Cloud computing: state-of-the-art and research challenges. *Journal of Internet Services and Applications , 1*, 7–18.

[1] The main reference for this work was of course the knowledge and experience elicited from the client of the project, CTO of ABC.

Appendix A - Model Equations[1]

Name	Dimensions	Unit	Definition
Accounting Depreciation Period for Ph Servers		yr	3
Accounting Depreciation Period for Spindles		yr	3
Accumulated Revenue of Last Yr		NOK	'Sampled Acc Revenue_Now'-'Sampled Acc Revenue_one year ago'
Accumulated Revenues_Data		NOK	0
Accumulated Servers Added		server	0
Accumulating Revenue		NOK	0
Acquisition Physical Servers		server	'Need-based Ph Server Acquisition'
AcquisitionInvoicing		NOK	'Acquisition Physical Servers'*'Price per New Server'+'Aquisition Spindles'*'Price per New Spindle'
Actual div by Target Annual Growth in Users		%/yr	'Annual Growth in Users'/'Target for Percentage of Annual Growth'
Added Hosts Acc		host	0
Added Hosts Acc_Data		host	0
Added Hosts div by Added Ph Servers of the Year			IF('Added Ph Servers of the Year'=0,0,'Added Hosts of the Year'/'Added Ph Servers of the Year')
Added Hosts div by Added Ph Servers of the Year_Data			IF('Added Ph Servers of the Year_Data'=0,0,'Added Hosts of the Year_Data'/'Added Ph Servers of the Year_Data')
Added Hosts of the Year		host	IF(TIME>1/1/2008,'Sampled Added Hosts Now'-'Sampled Added Hosts One Yr Ago','Sampled Added Hosts Now')
Added Hosts of the Year_Data		host	IF(TIME>1/1/2008,'Sampled Added Hosts Now_Data'-'Sampled Added Hosts One Yr Ago_Data','Sampled Added Hosts Now_Data')
Added Ph Servers Acc		server	0
Added Ph Servers Acc_Data		server	0
Added Ph Servers of the Year		server	IF(TIME>1/1/2008,'Sampled Added Ph Servers Acc Now'-'Sampled Added Ph Servers Acc One Yr Ago','Sampled

[1] Equations are presented here as imported from Powersim Studio 8.

			Added Ph Servers Acc Now')
Added Ph Servers of the Year_Data		server	IF(TIME>1/1/2008,'Sampled Added Ph Servers Acc Now_Data'-'Sampled Added Ph Servers Acc One Yr Ago_Data','Sampled Added Ph Servers Acc Now_Data')
Added Virtual Servers Acc		virt_server	0
Added Virtual Servers Acc_Data		virt_server	0
Added Virtual Servers div by Added Hosts of the Year			'Added Virtual Servers of the Year'/'Added Hosts of the Year'
Added Virtual Servers div by Added Hosts of the Year_Data			IF('Added Hosts of the Year_Data'=0,0,'Added Virtual Servers of the Year_Data'/'Added Hosts of the Year_Data')
Added Virtual Servers div by Added Ph Servers of the Year			'Added Virtual Servers of the Year'/'Added Ph Servers of the Year'
Added Virtual Servers div by Added Ph Servers of the Year_Data			IF('Added Ph Servers of the Year_Data'=0,0,'Added Virtual Servers of the Year_Data'/'Added Ph Servers of the Year_Data')
Added Virtual Servers of the Year		virt_server	IF(TIME>1/1/2008,'Sampled Added Virtual Servers Now'-'Sampled Added Virtual Servers One Yr Ago','Sampled Added Virtual Servers Now')
Added Virtual Servers of the Year_Data		virt_server	IF(TIME>1/1/2008,'Sampled Added Virtual Servers Now_Data'-'Sampled Added Virtual Servers One Yr Ago_Data','Sampled Added Virtual Servers Now_Data')
Addition of Power Capacity		MW	0
Aging Physical Servers Y1		server/mo	IF('Monthly Pulse'>0,'Physical Servers Y1'[12]/TIMESTEP,0)
Aging Physical Servers Y2		server/mo	IF('Monthly Pulse'>0,'Physical Servers Y2'[12]/TIMESTEP,0)
Aging Physical Servers Y3		server/mo	IF('Monthly Pulse'>0,'Physical Servers Y3'[12]/TIMESTEP,0)
Aging Physical Servers Y4		server/mo	IF('Monthly Pulse'>0,'Physical Servers Y4'[12]/TIMESTEP,0)*(1-'Fraction of Y4 Ph ServersDiscarded')
Aging Physical Servers Y5		server/mo	IF('Monthly Pulse'>0,'Physical Servers Y5'[12]/TIMESTEP,0)*(1-'Fraction of Y5 Ph ServersDiscarded')
Aging Spindles Y1		spindle/mo	IF('Monthly Pulse_Copy'>0,'Spindles Y1'[12]/TIMESTEP,0)

Aging Spindles Y2		spindle/yr	IF('Monthly Pulse_Copy'>0,'Spindles Y2'[12]/TIMESTEP,0)
Aging Spindles Y3		spindle/mo	IF('Monthly Pulse_Copy'>0,'Spindles Y3'[12]/TIMESTEP,0)
Aging Spindles Y4		spindle/mo	IF('Monthly Pulse_Copy'>0,'Spindles Y4'[12]/TIMESTEP,0)*(1-'Fraction of Y4 Spindles Discarded')
Aging Spindles Y5		spindle/mo	IF('Monthly Pulse_Copy'>0,'Spindles Y5'[12]/TIMESTEP,0)*(1-'Fraction of Y5 Spindles Discarded')
Annual Growth in Users		%/yr	'Net Flow of Users'/'Total Users'
Annual Growth in Users in Percentage		%	'Annual Growth in Users_Discrete'/Users_Oslo_One_Yr_Ago
Annual Growth in Users in Percentage_Data		%	'Annual Growth in Users_Data'/Users_Oslo_Data_One_Yr_Ago
Annual Growth in Users_Data		user	Users_Data_Raw-Users_Oslo_Data_One_Yr_Ago
Annual Growth in Users_Discrete		user	'Total Users'-Users_Oslo_One_Yr_Ago
Aquisition Spindles		spindle/mo	'Need-based Spindle Acquisition Policy'
ARPU		NOK/(mo*user)	'Total Monthly Revenue'/'Total Users'
ARPU DS		NOK/(mo*user)	IF('DS Users'=0, 0,'Monthly Revenue from DS Users'/'DS Users')
ARPU FD		NOK/(mo*user)	'Monthly Revenue from FD Users'/'FD Users'
ARPU_Data		NOK/(mo*user)	'Recurring Monthly Revenues_Data'/Users_Data
ARPU_New DS		NOK/(mo*user)	400
ARPU_New FD		NOK/(mo*user)	GRAPH(TIME, STARTTIME, 1, {1440,1170,810,400,1700,1700//Min:400;Max:2000//})
Average Ph Server Lifetime after 5 Years		yr	GRAPH(TIME, STARTTIME,1,{0.97,1.1,1.3,1.77,2,2//Min:0;Max:5//})
Avg Contract Age of Init Users		mo	21
Avg Contract Period of DS Users		mo	45
Avg Contract Period of FD Users		mo	45
Avg Cores per New Ph Server		Core/server	'Number of CPUs per New Server'*'Number of Cores per New CPU'

Avg Cores per Ph Server Y1		Core/serve r	ARRSUM('CPU Cores Y1')/ARRSUM('Physical Servers Y1')
Avg Cores per Ph Server Y2		Core/serve r	ARRSUM('CPU Cores Y2')/ARRSUM('Physical Servers Y2')
Avg Cores per Ph Server Y3		Core/serve r	ARRSUM('CPU Cores Y3')/ARRSUM('Physical Servers Y3')
Avg Cores per Ph Server Y4		Core/serve r	ARRSUM('CPU Cores Y4')/ARRSUM('Physical Servers Y4')
Avg Cores per Ph Server Y5		Core/serve r	ARRSUM('CPU Cores Y5')/ARRSUM('Physical Servers Y5')
Avg Cores per Ph Server Y5Plus		Core/serve r	'CPU Cores Y5Plus'/'Physical Servers Y5Plus'
Avg CPU Cores Error		Core	RUNAVERAGE(ABS('CPU Cores Error'))
Avg CPU Cores per CPU Y1	1..12	Core/CPU	IF('Physical Servers CPUs Y1'=0,0,'CPU Cores Y1'/'Physical Servers CPUs Y1')
Avg CPU Cores per CPU Y2	1..12	Core/CPU	IF('Physical Servers CPUs Y2'=0,0,'CPU Cores Y2'/'Physical Servers CPUs Y2')
Avg CPU Cores per CPU Y3	1..12	Core/CPU	IF('Physical Servers CPUs Y3'=0,0,'CPU Cores Y3'/'Physical Servers CPUs Y3')
Avg CPU Cores per CPU Y4	1..12	Core/CPU	IF('Physical Servers CPUs Y4'=0,0,'CPU Cores Y4'/'Physical Servers CPUs Y4')
Avg CPU Cores per CPU Y5	1..12	Core/CPU	IF('Physical Servers CPUs Y5'=0,0,'CPU Cores Y5'/'Physical Servers CPUs Y5')
Avg CPU Cores per CPU Y5Plus		Core/CPU	IF('Physical Servers CPUs Y5Plus'=0,0,'CPU Cores Y5Plus'/'Physical Servers CPUs Y5Plus')
Avg CPU Y1	1..12	CPU/server	IF('Physical Servers Y1'=0,0,'Physical Servers CPUs Y1'/'Physical Servers Y1')
Avg CPU Y2	1..12	CPU/server	IF('Physical Servers Y2'=0,0,'Physical Servers CPUs Y2'/'Physical Servers Y2')
Avg CPU Y3	1..12	CPU/server	IF('Physical Servers Y3'=0,0,'Physical Servers CPUs Y3'/'Physical Servers Y3')
Avg CPU Y4	1..12	CPU/server	IF('Physical Servers Y4'=0,0,'Physical Servers CPUs Y4'/'Physical Servers Y4')

Avg CPU Y5	1..12	CPU/server	IF('Physical Servers Y5'=0,0,'Physical Servers CPUs Y5'/'Physical Servers Y5')
Avg CPU Y5Plus		CPU/server	IF('Physical Servers Y5Plus'=0,0,'Physical Servers CPUs Y5Plus'/'Physical Servers Y5Plus')
Avg DS User Processor Requirement		log_server /user	IF('DS Users'=0, 0,'DS Users Processor Requirements'/'DS Users')
Avg DS User RAM Requirement		GB/user	IF('DS Users'=0, 0,'DS Users RAM Requirements'/'DS Users')
Avg DS User Storage Requirement		GB/user	IF('DS Users'=0, 0,'DS USers Storage Requirements'/'DS Users')
Avg Error div by Avg Number of CPU Cores		%	'Avg CPU Cores Error'/'Avg Number of CPU Cores_Data'
Avg Error div by Avg Number of Hosts		%	'Avg Hosts Error'/'Avg Number of Hosts_Data'
Avg Error div by Avg Number of Logical Servers		%	'Avg Logical Servers Error'/'Avg Number of Logical Servers_Data'
Avg Error div by Avg Number of Ph Servers		%	'Avg Ph Server Error'/'Avg Number of Ph Servers_Data'
Avg Error div by Avg Number of Spindles		%	'Avg Spindles Error'/'Avg Number of Spindles_Data'
Avg Error div by Avg Number of Users		%	'Avg Users Error'/'Avg Number of Users_Data'
Avg Error div by Avg Number of V Server		%	'Avg V Server Error'/'Avg Number of V Server_Data'
Avg Error div by Avg Storage		%	'Avg Storage Error'/'Avg Storage_Data'
Avg Error plus STD Devi div by Avg Storage		%	'Avg Error div by Avg Storage'+'STD Deviation of Error div by Avg Storage'
Avg Error plus STD Devi div by Avg Number of CPU Cores		%	'Avg Error div by Avg Number of CPU Cores'+'STD Deviation of Error div by Avg Number of CPU Cores'
Avg Error plus STD Devi div by Avg Number of Hosts		%	'Avg Error div by Avg Number of Hosts'+'STD Deviation of Error div by Avg Number of Hosts'
Avg Error plus STD Devi div by Avg Number of Logical Servers		%	'Avg Error div by Avg Number of Logical Servers'+'STD Deviation of Error div by Avg Number of Logical Servers'
Avg Error plus STD Devi div by Avg Number of Ph Servers		%	'Avg Error div by Avg Number of Ph Servers'+'STD Deviation of Error div by Avg Number of Ph Servers'
Avg Error plus STD Devi div by Avg Number of Spindles		%	'Avg Error div by Avg Number of Spindles'+'STD Deviation of Error div by Avg Number of Spindles'
Avg Error plus STD Devi div by Avg Number of Users		%	'Avg Error div by Avg Number of Users'+'STD Deviation of Error div by Avg Number of Users'

Avg Error plus STD Devi div by V Server		%	'Avg Error div by Avg Number of V Server'+'STD Deviation of Error div by Avg Number of V Server'
Avg FD User Processor Requirement		server/user	'FD Users Processor Requirements'/'FD Users'
Avg FD User RAM Requirement		GB/user	'FD Users RAM Requirements'/'FD Users'
Avg FD User Storage Requirement		TB/user	'FD Users Storage Requirements'/'FD Users'
Avg Host Lifetime After 3 Yrs		yr	GRAPHSTEP(TIME, STARTTIME,1,{1,1,1,0.05,.05,.05})
Avg Host per Ph Server Y1	1..12		IF('Physical Servers Y1'=0,0,'Hosts Y1'/'Physical Servers Y1')
Avg Host per Ph Server Y2	1..12		IF('Physical Servers Y2'=0,0,'Hosts Y2'/'Physical Servers Y2')
Avg Host per Ph Server Y3	1..12		IF('Physical Servers Y3'=0,0,'Hosts Y3'/'Physical Servers Y3')
Avg Host per Ph Server Y3Plus			'Hosts Y3Plus'/ARRSUM('Physical Servers Y4'+'Physical Servers Y5'+'Physical Servers Y5Plus')
Avg Hosts Error		host	RUNAVERAGE(ABS('Hosts Error'))
Avg Initial Power Demand of Hosts_Data		W/server	'Avg Power Demand of Servers_Data'*'Factor for Effective Power Demand for Hosts'
Avg Initial Power Demand of Stand-alone Servers_Data		W/server	'Avg Power Demand of Servers_Data'*'Factor for Effective Power Demand for Standalone Ph Servers'
Avg KW Demand of Disengaged Servers		W/server	IF('Total Ph Server Disengagement'=0,0,'Total Power freed up from Standalone Disengagement'/'Total Ph Server Disengagement')
Avg Logical Servers Error		server	RUNAVERAGE(ABS('Logical Servers Error'))
Avg Number of CPU Cores_Data		Core	RUNAVERAGE('CPU Cores_Data')
Avg Number of Hosts_Data		host	RUNAVERAGE(Hosts_Data)
Avg Number of Logical Servers_Data		server	RUNAVERAGE('Logical Servers_Data')
Avg Number of Ph Servers_Data		server	RUNAVERAGE('Ph Servers in Use_Data')
Avg Number of Spindles_Data		spindle	RUNAVERAGE('Spindles in Use_Data')
Avg Number of Users_Data		user	RUNAVERAGE(Users_Data)
Avg Number of V Server_Data		virt_server	RUNAVERAGE('Virtual Servers in Use_Data')
Avg Ph Server Error		server	RUNAVERAGE(ABS('Ph Server Error'))
Avg Ph Server Lifetime		yr	Average Ph Server Lifetime after 5 Years'+5

Avg Ph Servers Effective Power Demand		W/server	'Total Ph Servers Effective Power Demand'/'Total Physical Servers'
Avg Power Demand of New Servers_Data		W/server	IF(TIME>STARTTIME+62,NAN,1)*'Avg Power Demand of New Servers_Data_Raw'
Avg Power Demand of New Servers_Data_Raw		W/server	0
Avg Power Demand of Servers_Data		W/server	'Total KW Demand of Servers_Data'/'Ph Servers in Use_Data'
Avg Power Demand per Host Y1	1..12	W/host	'Hosts Power Demand Y1'/'Hosts Y1'
Avg Power Demand per Host Y2	1..12	W/host	'Hosts Power Demand Y2'/'Hosts Y2'
Avg Power Demand per Host Y3	1..12	W/host	'Hosts Power Demand Y3'/'Hosts Y3'
Avg Power Demand per Host Y3Plus		W/host	'Hosts Power Demand Y3Plus'/'Hosts Y3Plus'
Avg Power Demand Spindles Y1	1..12	KW/spindle	IF('Spindles Y1'=0,0,'Spindles Power_Y1'/'Spindles Y1')
Avg Power Demand Spindles Y2	1..12	KW/spindle	IF('Spindles Y2'=0,0,'Spindles Power_Y2'/'Spindles Y2')
Avg Power Demand Spindles Y3	1..12	KW/spindle	IF('Spindles Y3'=0,0,'Spindles Power_Y3'/'Spindles Y3')
Avg Power Demand Spindles Y4	1..12	KW/spindle	IF('Spindles Y4'=0,0,'Spindles Power_Y4'/'Spindles Y4')
Avg Power Demand Spindles Y5	1..12	KW/spindle	IF('Spindles Y5'=0,0,'Spindles Power_Y5'/'Spindles Y5')
Avg Power Demand Spindles Y5Plus		KW/spindle	IF('Spindles Y5Plus'=0,0,'Spindles Power_Y5Plus'/'Spindles Y5Plus')
Avg Power of Disengaged Servers_Data		W/server	IF(TIME>STARTTIME+62,NAN,1)*'Avg Power of Disengaged Servers_Data_Raw'
Avg Power of Disengaged Servers_Data_Raw		W/server	0
Avg Price of Spindles Y1	1..12	NOK/spindle	IF('Spindles Y1'=0,0,'Spindles Cumulative Prices Y1'/'Spindles Y1')
Avg Price of Spindles Y2	1..12	NOK/spindle	IF('Spindles Y2'=0,0,'Spindles Cumulative Prices Y2'/'Spindles Y2')
Avg Price of Spindles Y3	1..12	NOK/spindle	IF('Spindles Y3'=0,0,'Spindles Cumulative Prices Y3'/'Spindles Y3')

Avg Price of Spindles Y4	1..12	NOK/spindle	IF('Spindles Y4'=0,0,'Spindles Cumulative Prices Y4'/'Spindles Y4')
Avg Price of Spindles Y5	1..12	NOK/spindle	IF('Spindles Y5'=0,0,'Spindles Cumulative Prices Y5'/'Spindles Y5')
Avg Price of Spindles Y5Plus		NOK/spindle	IF('Spindles Y5Plus'=0,0,'Spindles Cumulative Prices Y5Plus'/'Spindles Y5Plus')
Avg Price Ph Servers Y1	1..12	NOK/server	IF('Physical Servers Y1'=0,0,'Physical Servers Cumulative Prices Y1'/'Physical Servers Y1')
Avg Price Ph Servers Y2	1..12	NOK/server	IF('Physical Servers Y2'=0,0,'Physical Servers Cumulative Prices Y2'/'Physical Servers Y2')
Avg Price Ph Servers Y3	1..12	NOK/server	IF('Physical Servers Y3'=0,0,'Physical Servers Cumulative Prices Y3'/'Physical Servers Y3')
Avg Price Ph Servers Y4	1..12	NOK/server	IF('Physical Servers Y4'=0,0,'Physical Servers Cumulative Prices Y4'/'Physical Servers Y4')
Avg Price Ph Servers Y5	1..12	NOK/server	IF('Physical Servers Y5'=0,0,'Physical Servers Cumulative Prices Y5'/'Physical Servers Y5')
Avg Price Ph Servers Y5Plus		NOK/server	IF('Physical Servers Y5Plus'=0,0,'Physical Servers Cumulative Prices Y5Plus'/'Physical Servers Y5Plus')
Avg RAM per Ph Server		GB/server	'Total RAM'/'Total Physical Servers'
Avg RAM Y1	1..12	GB/server	IF('Physical Servers Y1'=0,0,'Physical Servers RAM Y1'/'Physical Servers Y1')
Avg RAM Y2	1..12	GB/server	IF('Physical Servers Y2'=0,0,'Physical Servers RAM Y2'/'Physical Servers Y2')
Avg RAM Y3	1..12	GB/server	IF('Physical Servers Y3'=0,0,'Physical Servers RAM Y3'/'Physical Servers Y3')
Avg RAM Y4	1..12	GB/server	IF('Physical Servers Y4'=0,0,'Physical Servers RAM Y4'/'Physical Servers Y4')
Avg RAM Y5	1..12	GB/server	IF('Physical Servers Y5'=0,0,'Physical Servers RAM Y5'/'Physical Servers Y5')
Avg RAM Y5Plus		GB/server	IF('Physical Servers Y5Plus'=0,0,'Physical Servers RAM Y5Plus'/'Physical Servers Y5Plus')
Avg Spindle per Server		spindle/server	'Total Spindles'/'Total Physical Servers'

Avg Spindles Error		spindle	RUNAVERAGE(ABS('Spindles Error'))
Avg Spindles Lifetime after 5 Yrs		yr	2
Avg Stand-alone Power Demand Y1	1..12	KW/server	IF('Physical Servers Y1'=0,0,'Standalones Power_Y1'/'Physical Servers Y1')
Avg Stand-alone Power Demand Y2	1..12	KW/server	IF('Physical Servers Y2'=0,0,'Standalones Power_Y2'/'Physical Servers Y2')
Avg Stand-alone Power Demand Y3	1..12	KW/server	IF('Physical Servers Y3'=0,0,'Standalones Power_Y3'/'Physical Servers Y3')
Avg Stand-alone Power Demand Y4	1..12	KW/server	IF('Physical Servers Y4'=0,0,'Standalones Power_Y4'/'Physical Servers Y4')
Avg Stand-alone Power Demand Y5	1..12	KW/server	IF('Physical Servers Y5'=0,0,'Standalones Power_Y5'/'Physical Servers Y5')
Avg Stand-alone Power Demand Y5Plus		KW/server	IF('Physical Servers Y5Plus'=0,0,'Standalones Power_Y5Plus'/'Physical Servers Y5Plus')
Avg Storage Error		TB	RUNAVERAGE(ABS('Storage Error'))
Avg Storage of Spindles Y1	1..12	GB/spindle	IF('Spindles Y1'=0,0,'Storage Capacity Spindles Y1'/'Spindles Y1')
Avg Storage of Spindles Y2	1..12	GB/spindle	IF('Spindles Y2'=0,0,'Storage Capacity Spindles Y2'/'Spindles Y2')
Avg Storage of Spindles Y3	1..12	GB/spindle	IF('Spindles Y3'=0,0,'Storage Capacity Spindles Y3'/'Spindles Y3')
Avg Storage of Spindles Y4	1..12	GB/spindle	IF('Spindles Y4'=0,0,'Storage Capacity Spindles Y4'/'Spindles Y4')
Avg Storage of Spindles Y5	1..12	GB/spindle	IF('Spindles Y5'=0,0,'Storage Capacity Spindles Y5'/'Spindles Y5')
Avg Storage of Spindles Y5Plus		GB/spindle	IF('Spindles Y5Plus'=0,0,'Storage Capacity Spindles Y5Plus'/'Spindles Y5Plus')
Avg Storage per Spindle		TB/spindle	'Total Storage'/'Total Spindles'
Avg Storage_Data		TB	RUNAVERAGE('Storage in Use_Data')
Avg Users Error		user	RUNAVERAGE(ABS('Users Error'))
Avg V Server Error		virt_server	RUNAVERAGE(ABS('V Server Error'))
Avg Virt Server Lifetime		yr	Avg Virt Server Lifetime after 3 Yrs'+3

Avg Virt Server Lifetime after 3 Yrs		yr	GRAPH(TIME, STARTTIME,1,{1.5,1.49,1.38,1.21,0.88,0.10//Min:0;Max:2.1//})
Avg Virtual Server per Host Y1	1..12		IF('Hosts Y1'=0,0,'Virtual Servers Y1'/'Hosts Y1')
Avg Virtual Server per Host Y2	1..12		IF('Hosts Y2'=0,0,'Virtual Servers Y2'/'Hosts Y2')
Avg Virtual Server per Host Y3	1..12		IF('Hosts Y3'=0,0,'Virtual Servers Y3'/'Hosts Y3')
Avg Virtual Server per Host Y4			IF('Hosts Y3Plus'=0,0,'Virtual Servers Y3Plus'/'Hosts Y3Plus')
Cash		NOK	0
Cash Inflow		NOK/mo	DELAYPPL('Total Monthly Revenue','Customer Time to Pay')
Cash Outflow		NOK/yr	'Payback of Loans'+'Total Expenses'-'Total Depreciation'
Conv_To_Loan		NOK/mo	'Credit with Vendors'/'Maturation Time of Credits'
Conversion Factor from Power to Consumtion		hr	30*24
Converting Y2 Stand-alones into Hosts	1..12	server/yr	'Stand-alone Ph Servers Y2'*'Fraction of Y2 Stand-alones to Convert to Host'
Converting Y3 Stand-alones into Hosts	1..12	server/yr	'Fraction of Y3 Stand-alones to Convert to Host'*'Stand-alone Ph Servers Y3'
Converting Y4 Stand-alones into Hosts	1..12	server/yr	'Fraction of Y4 Stand-alones to Convert to Host'*'Stand-alone Ph Servers Y4'
Converting Y5 Stand-alones into Hosts	1..12	server/yr	'Stand-alone Ph Servers Y5'*'Fraction of Y5 Stand-alones to Convert to Host'
Converting Y5Plus Stand-alones into Hosts		server/yr	'Stand-alone Ph Servers Y5Plus'*'Fraction of Y5Plus Stand-alones to Convert to Host'
Cores per Ph Server		Core/server	'Total CPU Cores'/'Total Physical Servers'
Cores per Ph Server_Data		Core/server	'CPU Cores_Data'/'Ph Servers in Use_Data'
CPU Cores Discarded Y4	1..12	Core/yr	'CPUs Discarded Y4'*'Avg CPU Cores per CPU Y4'[12]
CPU Cores Discarded Y5	1..12	Core/mo	'CPUs Discarded Y5'*'Avg CPU Cores per CPU Y5'[12]
CPU Cores Discarded Y5Plus		Core/yr	'CPUs Discarded Y5Plus'*'Avg CPU Cores per CPU Y5Plus'
CPU Cores Error		Core	'Total CPU Cores'-'CPU Cores_Data'
CPU Cores Inflow		Core/mo	'CPU Inflow'*'Number of Cores per New CPU'

CPU Cores Inflow Y2		Core/mo	'CPU Inflow Y2'*'Avg CPU Cores per CPU Y1'[12]
CPU Cores Inflow Y3		Core/mo	'CPU Inflow Y3'*'Avg CPU Cores per CPU Y2'[12]
CPU Cores Inflow Y4		Core/mo	'CPU Inflow Y4'*'Avg CPU Cores per CPU Y3'[12]
CPU Cores Inflow Y5		Core/mo	'CPU Inflow Y5'*'Avg CPU Cores per CPU Y4'[12]
CPU Cores Inflow Y5Plus		Core/mo	'CPU Inflow Y5Plus'*'Avg CPU Cores per CPU Y5'[12]
CPU Cores Outflow		Core/yr	'CPU Outflow'*'Avg CPU Cores per CPU Y5Plus'
CPU Cores per CPU_Data		Core/CPU	'CPU Cores_Data'/CPUs_Data
CPU Cores Y1	1..12	Core	'CPU Cores_Data'/12/5
CPU Cores Y2	1..12	Core	'CPU Cores_Data'/12/5
CPU Cores Y3	1..12	Core	'CPU Cores_Data'/12/5
CPU Cores Y4	1..12	Core	'CPU Cores_Data'/12/5
CPU Cores Y5	1..12	Core	'CPU Cores_Data'/12/5
CPU Cores Y5Plus		Core	0
CPU Cores_Data		Core	IF(TIME>STARTTIME+62,NAN,1)*'CPU Cores_Data_Raw'
CPU Cores_Data_Raw		Core	0
CPU Inflow		CPU/mo	'Server Installation'*'Number of CPUs per New Server'
CPU Inflow Y2		CPU/mo	'Aging Physical Servers Y1'*'Avg CPU Y1'[12]
CPU Inflow Y3		CPU/mo	'Aging Physical Servers Y2'*'Avg CPU Y2'[12]
CPU Inflow Y4		CPU/mo	'Aging Physical Servers Y3'*'Avg CPU Y3'[12]
CPU Inflow Y5		CPU/mo	'Aging Physical Servers Y4'*'Avg CPU Y4'[12]
CPU Inflow Y5Plus		CPU/mo	'Aging Physical Servers Y5'*'Avg CPU Y5'[12]
CPU Outflow		CPU/yr	'Ph Servers Outflow'*'Avg CPU Y5Plus'
CPU per Ph Server		CPU/server	'Total CPUs'/'Total Physical Servers'
CPU per Ph Server_Data		CPU/server	CPUs_Data/'Ph Servers in Use_Data'
CPUs Discarded Y4	1..12	CPU/yr	'Discard Rate Y4'*'Avg CPU Y4'[12]
CPUs Discarded Y5	1..12	CPU/mo	'Discard Rate Y5'*'Avg CPU Y5'[12]
CPUs Discarded Y5Plus		CPU/yr	'Discard Rate Y5Plus'*'Avg CPU Y5Plus'
CPUs_Data		CPU	IF(TIME>STARTTIME+62,NAN,1)*CPUs_Data_Raw
CPUs_Data_Raw		CPU	0

Credit with Vendors		NOK	0
Cumulative Prices Inflow		NOK/mo	'Server Installation'*'Price per New Server'
Cumulative Prices Inflow Y2		NOK/mo	'Aging Physical Servers Y1'*'Avg Price Ph Servers Y1'[12]
Cumulative Prices Inflow Y3		NOK/mo	'Aging Physical Servers Y2'*'Avg Price Ph Servers Y2'[12]
Cumulative Prices Inflow Y4		NOK/mo	'Aging Physical Servers Y3'*'Avg Price Ph Servers Y3'[12]
Cumulative Prices Inflow Y5		NOK/mo	'Aging Physical Servers Y4'*'Avg Price Ph Servers Y4'[12]
Cumulative Prices Inflow Y5Plus		NOK/mo	'Aging Physical Servers Y5'*'Avg Price Ph Servers Y5'[12]
Cumulative Prices Inflow_Spindles Y2		NOK/mo	IF('Monthly Pulse_Copy'>0,'Aging Spindles Y1'*'Avg Price of Spindles Y1'[12],0)
Cumulative Prices Inflow_Spindles Y3		NOK/yr	IF('Monthly Pulse_Copy'>0,'Aging Spindles Y2'*'Avg Price of Spindles Y2'[12],0)
Cumulative Prices Inflow_Spindles Y4		NOK/mo	IF('Monthly Pulse_Copy'>0,'Aging Spindles Y3'*'Avg Price of Spindles Y3'[12],0)
Cumulative Prices Inflow_Spindles Y5		NOK/mo	IF('Monthly Pulse_Copy'>0,'Aging Spindles Y4'*'Avg Price of Spindles Y4'[12],0)
Cumulative Prices Inflow_Spindles Y5Plus		NOK/mo	IF('Monthly Pulse_Copy'>0,'Aging Spindles Y5'*'Avg Price of Spindles Y5'[12],0)
Cumulative Prices Outflow		NOK/yr	'Ph Servers Outflow'*'Avg Price Ph Servers Y5Plus'
Cumulative Prices Outflow_Spindles		NOK/yr	'Disengagement of Spindles'*'Avg Price of Spindles Y5Plus'
Customer Time to Pay		mo	1
Data Center Surplus Power		MW	'Total Initial Data Center Power'-'Total Power Demand of Server & Spindles'
Date to Add New Power Capacity		@yr	TIME
DC Power Availability Switch			IF('Expected Time to Cover Servers Gap based only on DC Power Availability'>4,0,1)
DELAYINF_Ended DS Users of Last Year		user	DELAYINF('Ended DS Users of Last Year',1,6)
DELAYINF_Ended FD Users of Last Year		user	DELAYINF('Ended FD Users of Last Year',1,6)
DELAYINF_Revenue of Last Year		NOK	DELAYINF('Revenue of Last 1 yr',1,3,00000000)
Depreciation Ph Servers		NOK/yr	ARRSUM('Physical Servers Cumulative Prices Y1'+'Physical Servers Cumulative Prices Y2'+'Physical Servers Cumulative Prices Y3')/'Accounting Depreciation Period for Ph Servers'

Depreciation Spindles		NOK/yr	ARRSUM('Spindles Cumulative Prices Y1'+'Spindles Cumulative Prices Y2'+'Spindles Cumulative Prices Y3')/'Accounting Depreciation Period for Spindles'
Desired Inventory of Physical Servers		server	'Need-based Ph Server Installation'*'Desired Ph Server Inventory Coverage Period'
Desired Inventory of Spindles		spindle	'Need-based Spindle Installation'*'Desired Spindle Inventory Coverage Period'
Desired Ph Server Inventory Coverage Period		mo	1
Desired Spindle Inventory Coverage Period		mo	1
Desired Workforce		person	MIN('Need-based Ph Server Installation','Max Server Installation Possible allowed by Power')/'Number of Ph Server Installations per Workforce per Month'
Discard Rate Spindles Y4		spindle/mo	IF('Monthly Pulse_Copy'>0,'Spindles Y4'[12]/TIMESTEP,0)*'Fraction of Y4 Spindles Discarded'
Discard Rate Spindles Y5		spindle/mo	IF('Monthly Pulse_Copy'>0,'Spindles Y5'[12]/TIMESTEP,0)*'Fraction of Y5 Spindles Discarded'
Discard Rate Y4	1..12	server/yr	'Physical Servers Y4'*'Fraction of Y4 Ph ServersDiscarded'
Discard Rate Y5	1..12	server/mo	'Physical Servers Y5'*'Fraction of Y5 Ph ServersDiscarded'
Discard Rate Y5Plus		server/yr	'Physical Servers Y5Plus'*'Fraction of Y5Plus Ph Servers Discarded'
Disengagement of Spindles		spindle/yr	'Spindles Y5Plus'/'Avg Spindles Lifetime after 5 Yrs'
DS Users		user	Users_Data*'Init Percentage of DS USers'
DS Users Drainage		user/yr	'DS Users'*'Fraction of DS Users Drained per yr'
DS Users having ended contract		user	0
DS Users Inflow		user/mo	'New DS Users to Acquire'/'Time to reach Yearly Users Target'
DS Users Not Renewing Contract		user/mo	DELAYMTR('DS Users Inflow'*(1-'Fraction of DS Users Renewing'),'Avg Contract Period of DS Users',7)+'Initial DS Users Leaving'
DS Users Processor Requirements		log_server	0
DS Users RAM Requirements		GB	0
DS Users Renewing Contract		user/mo	DELAYMTR('DS Users Inflow'*'Fraction of DS Users

			Renewing','Avg Contract Period of DS Users',7)
DS USers Storage Requirements		GB	0
DS Users_Data		user	IF(TIME>STARTTIME+62,NAN,1)*'DS Users_Data_Raw'
DS Users_Data_Raw		user	0
Ended DS Users of 1st Year		user	0
Ended DS Users of Last Year		user	'Sampled Ended DS Users Now'-'Sampled Ended DS Users One Year Ago'
Ended FD Users of 1st Year		user	STEP(600,STARTTIME)-STEP(600,STARTTIME+13)
Ended FD Users of Last Year		user	'Sampled Ended FD Users Now'-'Sampled Ended FD Users One Year Ago'
Energy Expenses		NOK/yr	'Total Power Demand of Server & Spindles'*'Conversion Factor from Power to Consumtion'*PUE*'Energy Price'
Energy Price		NOK/KWh	0.85
Error in RunAvg_CPU Cores		%	ABS('RunAvg_CPU Cores_Data'-'RunAvg_CPU Cores')/'RunAvg_CPU Cores_Data'
Error in RunAvg_Hosts		%	ABS(RunAvg_Hosts_Data-RunAvg_Hosts)/RunAvg_Hosts_Data
Error in RunAvg_Logical Servers		%	ABS('RunAvg_Logical Servers_Data'-'RunAvg_Logical Servers')/'RunAvg_Logical Servers_Data'
Error in RunAvg_Ph Servers		%	ABS('RunAvg_Ph Servers in Use_Data'-'RunAvg_Total Ph Servers')/'RunAvg_Ph Servers in Use_Data'
Error in RunAvg_Spindles		%	ABS(RunAvg_Spindles_Data-RunAvg_Spindles)/RunAvg_Spindles_Data
Error in RunAvg_Storage		%	ABS('RunAvg_Storage in Use_Data'-'RunAvg_Total Storage')/'RunAvg_Storage in Use_Data'
Error in RunAvg_Users		%	ABS(RunAvg_Users_Data-RunAvg_Users)/RunAvg_Users_Data
Error in RunAvg_V servers		%	ABS('RunAvg_V Server in Use_Data'-'RunAvg_V Server')/'RunAvg_V Server in Use_Data'
Error in STDEV_CPU Cores		%	ABS('STD_Deviation_CPU Cores_Data'-'STD_Deviation_CPU Cores')/'STD_Deviation_CPU Cores_Data'
Error in STDEV_Hosts		%	ABS(STD_Deviation_Hosts_Data-

			STD_Deviation_Hosts)/STD_Deviation_Hosts_Data
Error in STDEV_Logical Servers		%	ABS('STD_Deviation_Logical Servers_Data'-'STD_Deviation_Logical Servers')/'STD_Deviation_Logical Servers_Data'
Error in STDEV_Ph Servers		%	ABS('STD_Deviation_Ph Servers in Use_Data'-'STD_Deviation_Total Ph Servers')/'STD_Deviation_Ph Servers in Use_Data'
Error in STDEV_Spindles		%	ABS('STD_Deviation_Spindles in Use_Data'-STD_Deviation_Spindles)/'STD_Deviation_Spindles in Use_Data'
Error in STDEV_Storage		%	ABS('STD_Deviation_Storage in Use_Data'-'STD_Deviation_Total Storage')/'STD_Deviation_Storage in Use_Data'
Error in STDEV_Users		%	ABS(STD_Deviation_Users_Data-STD_Deviation_Users)/STD_Deviation_Users_Data
Error in STDEV_V Servers		%	ABS('STD_Deviation_V Server in Use_DataV Server'-'STD_Deviation_V Server')/'STD_Deviation_V Server in Use_DataV Server'
Expected Time to Cover Servers Gap based only on DC Power Availability		mo	'Gap between Required & Actual Number of Logical Servers'/'Max Server Installation Possible allowed by Power'
Expected Time to Cover Servers Gap based only on HR Availability		mo	'Gap between Required & Actual Number of Logical Servers'/'Max Server Installation allowed by Workforce'
Factor for Effective Power Demand for Hosts			0.99
Factor for Effective Power Demand for Standalone Ph Servers			0.9
FD Not Renewing by Percentage of Inflow		%	'FD Users Not Renewing Contract'/'FD Users Inflow'
FD User_Data_Raw		user	0
FD Users		user	Users_Data*(1-'Init Percentage of DS USers')
FD Users Drainage		user/yr	'FD Users'*'Fraction of FD Users Drained per yr'
FD Users Drained by Percentage		%/yr	'FD Users Drainage'/'FD Users'
FD Users having ended contract		user	0
FD Users Inflow		user/mo	'New FD Users to Acquire per Yr'/'Time to reach Yearly

			Users Target'
FD Users Not Renewing by Percentage		%/yr	'FD Users Not Renewing Contract'/'FD Users'
FD Users Not Renewing Contract		user/mo	DELAYMTR('FD Users Inflow'*(1-'Fraction of FD Users Renewing Contract'),'Avg Contract Period of FD Users',7,0)+'Initial FD Users Leaving'
FD Users Processor Requirements		server	'FD Users'*'Initial Avg User Processor Req'
FD Users RAM Requirements		GB	FD Users'*2
FD Users Renewing Contract		user/mo	DELAYMTR('FD Users Inflow'*'Fraction of FD Users Renewing Contract','Avg Contract Period of FD Users',7,0)
FD Users Storage Requirements		TB	FD Users'*16
FD Users_Data		user	IF(TIME>STARTTIME+62,NAN,1)*'FD User_Data_Raw'
Finances		NOK	0
Financial Expenses		NOK/yr	Finances*IR
Fraction of DS Users Drained per yr		yr^-1	0.02
Fraction of DS Users Renewing			0.95
Fraction of FD Users Drained per yr		yr^-1	0.02
Fraction of FD Users Renewing Contract			0.95
Fraction of Revenue for Server Acquisition			0.1
Fraction of Stand-alone Servers to Convert to Hosts		%/yr	GRAPHSTEP(TIME,STARTTIME,1,{0,0,0,0,0,0//Min:-1;Max:11//})
Fraction of Y2 Hosts Discarded		%	GRAPHSTEP(TIME, STARTTIME, 1,{0,0,0,50,0,0})
Fraction of Y2 Stand-alones to Convert to Host		%/yr	'Fraction of Stand-alone Servers to Convert to Hosts'
Fraction of Y3 Hosts Discarded		%	GRAPHSTEP(TIME, STARTTIME, 1,{0,0,0,50,0,0})
Fraction of Y3 Stand-alones to Convert to Host		%/yr	'Fraction of Stand-alone Servers to Convert to Hosts'
Fraction of Y4 Hosts Discarded		%/yr	0
Fraction of Y4 Ph ServersDiscarded		%/yr	GRAPHSTEP(TIME,STARTTIME,1,{0,0,0,0,0,0//Min:-1;Max:11//})
Fraction of Y4 Spindles Discarded			0
Fraction of Y4 Stand-alones to Convert to Host		%/yr	'Fraction of Stand-alone Servers to Convert to Hosts'
Fraction of Y5 Ph ServersDiscarded		%/yr	GRAPHSTEP(TIME,STARTTIME,1,{0,0,0,0,0,0//Min:-

			1;Max:11//})
Fraction of Y5 Spindles Discarded			0
Fraction of Y5 Stand-alones to Convert to Host		%/yr	'Fraction of Stand-alone Servers to Convert to Hosts'
Fraction of Y5Plus Ph Servers Discarded		%/yr	GRAPHSTEP(TIME,STARTTIME,1,{0,0,0,0,0,0//Min:-1;Max:11//})
Fraction of Y5Plus Stand-alones to Convert to Host		%/yr	'Fraction of Stand-alone Servers to Convert to Hosts'
Future Growth			5.9
Future Share of DS Users in New Contracts			60
Gap between Desired & Actual Inventory of Ph Servers		server	'Desired Inventory of Physical Servers'-'Inventory of Physical Servers'
Gap between Desired & Actual Inventory of Ph Servers_NonNegative		server	('Gap between Desired & Actual Inventory of Ph Servers'+ABS('Gap between Desired & Actual Inventory of Ph Servers'))/2
Gap between Desired & Actual Inventory of Spindles		spindle	'Desired Inventory of Spindles'-'Inventory of Spindles'
Gap between Desired & Actual Inventory of Spindles_NonZero		spindle	('Gap between Desired & Actual Inventory of Spindles'+ABS('Gap between Desired & Actual Inventory of Spindles'))/2
Gap between Required & Actual Number of Logical Servers		server	'Total Required Logical Servers'-'Total Logical Servers'
Gap between Required & Actual Storage		TB	'Total Storage Requirements'-'Total Storage'
Hiring		person/mo	(('Desired Workforce'-'Operational Workforce')+ABS('Desired Workforce'-'Operational Workforce'))/2/'Time to Hire'
Host Disengagement		host/yr	'Hosts Y3Plus'*(1-'Fraction of Y4 Hosts Discarded')/'Avg Host Lifetime After 3 Yrs'
Hosts Added Divided by Servers Added_Data			'Hosts Added_Data_Raw'/'Ph Servers Added_Data_Raw'
Hosts Added_Data		host/mo	IF(TIME>STARTTIME+62,NAN,1)*'Hosts Added_Data_Raw'
Hosts Added_Data_Raw		host/mo	0
Hosts Discarded Y2	1..12	host	'Hosts Y2'*'Fraction of Y2 Hosts Discarded'
Hosts Discarded Y3	1..12	host	'Hosts Y3'*'Fraction of Y3 Hosts Discarded'
Hosts Discarded Y4		host/yr	'Hosts Y3Plus'*'Fraction of Y4 Hosts Discarded'

Hosts Disengaged_Data		host/mo	IF(TIME>STARTTIME+62,NAN,1)*'Hosts Disengaged_Data_Raw'
Hosts Disengaged_Data_Raw		host/mo	0
Hosts Error		host	'Total Hosts'-Hosts_Data
Hosts Inflow		server/mo	'Server Installation'*'Ratio of New Ph Servers being Converted to Hosts'
Hosts Inflow Y2		host/mo	IF('Monthly Pulse'>0,'Hosts Y1'[12]/TIMESTEP,0)
Hosts Inflow Y3		host/mo	IF('Monthly Pulse'>0,'Hosts Y2'[12]/TIMESTEP,0)*(1-'Fraction of Y2 Hosts Discarded')
Hosts Inflow Y3Plus		host/mo	IF('Monthly Pulse'>0,'Hosts Y3'[12]/TIMESTEP,0)*(1-'Fraction of Y3 Hosts Discarded')
Hosts Power Demand Y1	1..12	W	'Total Initial Power Demand of Hosts_Data'/12/4
Hosts Power Demand Y2	1..12	W	'Total Initial Power Demand of Hosts_Data'/12/4
Hosts Power Demand Y3	1..12	W	'Total Initial Power Demand of Hosts_Data'/12/4
Hosts Power Demand Y3Plus		W	'Total Initial Power Demand of Hosts_Data'/4
Hosts Power freed up_Discard Y2	1..12	W	'Hosts Discarded Y2'*'Avg Power Demand per Host Y2'
Hosts Power freed up_Discard Y3	1..12	W	'Hosts Discarded Y3'*'Avg Power Demand per Host Y3'
Hosts Power freed up_Discard Y3Plus		m²*kg/yr^4	'Hosts Discarded Y4'*'Avg Power Demand per Host Y3Plus'
Hosts Power Inflow		m²*kg/mo^4	'Hosts Inflow'*'Power Demand of New Servers'*'Factor for Effective Power Demand for Hosts'
Hosts Power Inflow Y2		m²*kg/mo^4	'Hosts Inflow Y2'*'Avg Power Demand per Host Y1'[12]
Hosts Power Inflow Y3		m²*kg/mo^4	'Hosts Inflow Y3'*'Avg Power Demand per Host Y2'[12]
Hosts Power Inflow Y3Plus		m²*kg/mo^4	'Hosts Inflow Y3Plus'*'Avg Power Demand per Host Y3'[12]
Hosts Power Outflow		m²*kg/yr^4	'Host Disengagement'*'Avg Power Demand per Host Y3Plus'
Hosts Y1	1..12	host	Hosts_Data/12/4
Hosts Y2	1..12	host	Hosts_Data/12/4

Hosts Y3	1..12	host	Hosts_Data/12/4
Hosts Y3Plus		host	Hosts_Data/4
Hosts_Data		host	IF(TIME>STARTTIME+62,NAN,1)*Hosts_Data_Raw
Hosts_Data_Raw		host	17
HR Availability Switch			IF('Sliding Average of Expected Installation Time based on HR'>3,0,1)
HR Expenses		NOK/yr	'Operational Workforce'*'Salary per Workforce'
Inflow by Percentage		%/yr	'FD Users Inflow'/'FD Users'
Inflow of Revenue through New DS User Contracts		NOK/mo²	'DS Users Inflow'*'ARPU_New DS'
Inflow of Revenue through New FD User Contracts		NOK/mo²	'FD Users Inflow'*'ARPU_New FD'
Inflow to Cumulative Prices of Spindles		NOK/mo	'Spindle Installation'*'Price per New Spindle'
Inflow to DS Users Processor Requirements		log_server /mo	'Processor Requirement per DS User'*'DS Users Inflow'
Inflow to DS Users RAM Requirements		GB/mo	'RAM Requirement per New DS User'*'DS Users Inflow'
Inflow to DS USers Storage Requirements		GB/mo	'DS Users Inflow'*'Storage Requirement per New DS User'
Inflow to FD Users Processor Requirements		log_server /mo	'Processor Requirements per New FD User'*'FD Users Inflow'
Inflow to FD Users RAM Requirements		GB/mo	'RAM Requirement per New FD User'*'FD Users Inflow'
Inflow to FD Users Storage Requirements		GB/mo	'FD Users Inflow'*'Storage Requirements per New FD User'
Inflow to Stand-alone Ph Servers		server/mo	'Server Installation'*'Ratio of New Ph Servers starting as Standalone'
Inflow to Stand-alone Ph Servers Y2		server/mo	'Aging Physical Servers Y1'*'Ratio of Stand-alone to Total Ph Servers Y1'[12]
Inflow to Stand-alone Ph Servers Y3		server/mo	IF('Monthly Pulse'>0,'Aging Physical Servers Y2'*'Ratio of Stand-alone to Total Ph Servers Y2'[12],0)
Inflow to Stand-alone Ph Servers Y4		server/mo	'Aging Physical Servers Y3'*'Ratio of Stand-alone to Total Ph Servers Y3'[12]
Inflow to Stand-alone Ph Servers Y5		server/mo	'Aging Physical Servers Y4'*'Ratio of Stand-alone to Total Ph Servers Y4'[12]
Inflow to Stand-alone Ph Servers Y5Plus		server/mo	'Aging Physical Servers Y5'*'Ratio of Stand-alone to Total Ph Servers Y5'[12]

Init Percentage of DS USers		%	0%
Initial Avg User Processor Req		server/user	1/'Users per Logical Server_Data'
Initial DS Users		user	Users_Data_Raw*'Init Percentage of DS USers'
Initial DS Users Leaving		user/yr	'Initial DS Users'*'Fraction of DS Users Drained per yr'+'Initial DS Users'*(1-'Fraction of DS Users Renewing')/('Avg Contract Period of DS Users'-'Avg Contract Age of Init Users')
Initial FD Users		user	Users_Data_Raw*(1-'Init Percentage of DS USers')
Initial FD Users Leaving		user/yr	'Initial FD Users'*'Fraction of FD Users Drained per yr'+'Initial FD Users'*(1-'Fraction of FD Users Renewing Contract')/('Avg Contract Period of FD Users'-'Avg Contract Age of Init Users')
Initial FD Users Leaving by Percentage of Inflow		%	'Initial FD Users Leaving'/'FD Users Inflow'
Inventory of Physical Servers		server	0
Inventory of Spindles		spindle	'Spindles in Use_Data_Raw'/5/12
Invoiced		NOK	0
IR		%/yr	GRAPHSTEP(TIME,STARTTIME,1,{3.75,4,3.5,3.37,6.75,5.84//Min:0;Max:7//})
Logical Servers Error		server	'Total Logical Servers'-'Logical Servers_Data'
Logical Servers per Ph Server			'Total Logical Servers'/'Total Physical Servers'
Logical Servers per Ph Server_Data			'Logical Servers_Data'/'Ph Servers in Use_Data'
Logical Servers_Data		server	'Standalone Ph Servers_Data'+'Virtual Servers in Use_Data'
Maturation Time of Credits		mo	1
Maturation Time of Loans		yr	3
Max CPU Cores Error		Core	RUNMAX(ABS('CPU Cores Error'))
Max Error div by Avg Number of CPU Cores		%	'Max CPU Cores Error'/'Avg Number of CPU Cores_Data'
Max Error div by Avg Number of Hosts		%	'Max Hosts Error'/'Avg Number of Hosts_Data'
Max Error div by Avg Number of Logical Servers		%	'Max Logical Servers Error'/'Avg Number of Logical Servers_Data'
Max Error div by Avg Number of Ph Servers		%	'Max Ph Server Error'/'Avg Number of Ph Servers_Data'

Max Error div by Avg Number of Spindles		%	'MaxSpindles Error'/'Avg Number of Spindles_Data'
Max Error div by Avg Number of Users		%	'Max Users Error'/'Avg Number of Users_Data'
Max Error div by Avg Number of V Server		%	'Max V Server Error'/'Avg Number of V Server_Data'
Max Error div by Avg Storage		%	'Max Storage Error'/'Avg Storage_Data'
Max Hosts Error		host	RUNMAX(ABS('Hosts Error'))
Max Logical Servers Error		server	RUNMAX(ABS('Logical Servers Error'))
Max Ph Server Error		server	RUNMAX(ABS('Ph Server Error'))
Max Server Installation allowed by Inventory		server/mo	'Inventory of Physical Servers'/TIMESTEP
Max Server Installation allowed by Workforce		server/mo	'Operational Workforce'*'Number of Ph Server Installations per Workforce per Month'
Max Server Installation Possible allowed by Power		server/mo	('Data Center Surplus Power'/'Power Demand of New Servers')/TIMESTEP
Max Spindle Installation allowed by Inventory		spindle/mo	'Inventory of Spindles'/TIMESTEP
Max Spindle Installation Possible allowed by Power		spindle/mo	('Data Center Surplus Power'/'Power Demand of New Spindles')/TIMESTEP
Max Storage Error		TB	RUNMAX(ABS('Storage Error'))
Max Users Error		user	RUNMAX(ABS('Users Error'))
Max V Server Error		virt_server	RUNMAX(ABS('V Server Error'))
MaxSpindles Error		spindle	RUNMAX(ABS('Spindles Error'))
MIN Budget for Servers		NOK	50000000
Monthly Pulse			PULSE(TIMESTEP,STARTTIME,1)
Monthly Pulse_Copy			PULSE(1*TIMESTEP,STARTTIME,1)
Monthly Revenue from DS Users		NOK/mo	0
Monthly Revenue from FD Users		NOK/mo	FD Users'*1350
Need-based Ph Server Acquisition		server	DELAYMTR('Gap between Desired & Actual Inventory of Ph Servers_NonNegative','Time to Acquire Servers',6)
Need-based Ph Server Installation		server/mo	DELAYMTR('Required Ph Servers to be Installed_NonNegative','Time to Install Ph Servers',3,0)
Need-based Spindle Acquisition Policy		spindle/mo	DELAYMTR('Gap between Desired & Actual Inventory of Spindles_NonZero','Time to Acquire Spindles',3,0)

Need-based Spindle Installation		spindle/mo	DELAYMTR('Required Spindles to be Installed_NonNegative','Time to Install Spindles',1,0)
Net Cash Flow		NOK/mo	'Cash Inflow'-'Cash Outflow'
Net Flow of DS Users		user/mo	'DS Users Inflow'-'Total Outflow of DS Users'
Net Flow of FD Users		user/mo	'FD Users Inflow'-'Total Outflow of FD Users'
Net Flow of Users		user/mo	'Net Flow of FD Users'+'Net Flow of DS Users'
Net Growth divided by Target Growth		%/yr	'Net Growth of FD Users'/'Target for Percentage of Annual Growth'
Net Growth of DS Users		%/yr	'Net Flow of DS Users'/'DS Users'
Net Growth of FD Users		%/yr	'Net Flow of FD Users'/'FD Users'
Net Income before Taxes		NOK/yr	'Total Annual Revenue'-'Total Expenses'
New DS Users to Acquire		user	IF('DC Power Availability Switch'=1 AND 'HR Availability Switch'=1,'Target for New DS Users'+'DELAYINF_Ended DS Users of Last Year'+'Ended DS Users of 1st Year',0)
New FD Users to Acquire per Yr		user	IF('DC Power Availability Switch'=1 AND 'HR Availability Switch'=1,'Target for New FD Users'+'DELAYINF_Ended FD Users of Last Year'+'Ended FD Users of 1st Year',0)
Number of Cores per New CPU		Core/CPU	2+STEP(2,STARTTIME+18)+STEP(2,STARTTIME+39)
Number of CPUs per New Server		CPU/server	2
Number of Ph Server Installations per Workforce per Month		server/(mo *person)	15
Number of Stand-alone Servers to Convert to Hosts		server/mo	'Server Installation'*MAX(('Hosts Added Divided by Servers Added_Data'-'Ratio of New Ph Servers being Converted to Hosts'),0)
Older than Y2 Hosts			'Total Hosts'-'Y1and2 Hosts'
Older than Y2 Servers			'Total Physical Servers'-'Y1and2 Ph Servers'
Older than Y2 Stand-alone Ph Servers			'Older than Y2 Servers'-'Older than Y2 Hosts'
Operational Workforce		person	5
Outflow by Percentage		%/yr	'Total Outflow of FD Users'/'FD Users'
Outflow from DS Users Processor Requirements		log_server /yr	'Avg DS User Processor Requirement'*('DS Users Drainage'+'DS Users Not Renewing Contract')

Outflow from DS Users RAM Requirements		GB/yr	'Avg DS User RAM Requirement'*('DS Users Drainage'+'DS Users Not Renewing Contract')
Outflow from DS USers Storage Requirements		GB/yr	'Avg DS User Storage Requirement'*('DS Users Drainage'+'DS Users Not Renewing Contract')
Outflow from FD Users Processor Requirements		server/yr	'Avg FD User Processor Requirement'*('FD Users Drainage'+'FD Users Not Renewing Contract')
Outflow from FD Users RAM Requirements		GB/yr	'Avg FD User RAM Requirement'*('FD Users Drainage'+'FD Users Not Renewing Contract')
Outflow from FD Users Storage Requirements		TB/yr	'Avg FD User Storage Requirement'*('FD Users Drainage'+'FD Users Not Renewing Contract')
Outflow of Revenue from DS Users Lost		NOK/mo^2	'ARPU DS'*('DS Users Drainage'+'DS Users Not Renewing Contract')
Outflow of Revenue from FD Users Lost		NOK/mo^2	'ARPU FD'*('FD Users Drainage'+'FD Users Not Renewing Contract')
Overhead Expenses		NOK/yr	'HR Expenses'*'Overhead Factor'
Overhead Factor			0.35
Payback of Loans		NOK/yr	IF(Finances=0,0,Finances/'Maturation Time of Loans')
Percentage of Distributed Services Users in New Contracts		%	'Percentage of Distributed Services Users in New Contracts_No Unit'*1%
Percentage of Distributed Services Users in New Contracts_No Unit			GRAPHSTEP(TIME, STARTTIME, 1,{0,0,0,0,60,0,'Future Share of DS Users in New Contracts'//Min:-1;Max:100//})
Ph Server Error		server	'Total Physical Servers'-'Ph Servers in Use_Data'
Ph Servers Added_Data		server/mo	IF(TIME>STARTTIME+62,NAN,1)*'Ph Servers Added_Data_Raw'
Ph Servers Added_Data_Raw		server/mo	0
Ph Servers Disengaged_Data		server/mo	IF(TIME>STARTTIME+62,NAN,1)*'Ph Servers Disengaged_Data_Raw'
Ph Servers Disengaged_Data_Raw		server/mo	0
Ph Servers in Use_Data		server	IF(TIME>STARTTIME+62,NAN,1)*'Ph Servers in Use_Data_Raw'
Ph Servers in Use_Data_Raw		server	0
Ph Servers Outflow		server/yr	'Physical Servers Y5Plus'*(1-'Fraction of Y5Plus Ph Servers

			Discarded')/'Average Ph Server Lifetime after 5 Years'
Physical Servers CPUs Y1	1..12	CPU	CPUs_Data/12/5
Physical Servers CPUs Y2	1..12	CPU	CPUs_Data/12/5
Physical Servers CPUs Y3	1..12	CPU	CPUs_Data/12/5
Physical Servers CPUs Y4	1..12	CPU	CPUs_Data/12/5
Physical Servers CPUs Y5	1..12	CPU	CPUs_Data/12/5
Physical Servers CPUs Y5Plus		CPU	0
Physical Servers Cumulative Prices Y1	1..12	NOK	11e6/12/5
Physical Servers Cumulative Prices Y2	1..12	NOK	11e6/12/5
Physical Servers Cumulative Prices Y3	1..12	NOK	11e6/12/5
Physical Servers Cumulative Prices Y4	1..12	NOK	11e6/12/5
Physical Servers Cumulative Prices Y5	1..12	NOK	11e6/12/5
Physical Servers Cumulative Prices Y5Plus		NOK	0
Physical Servers RAM Y1	1..12	GB	12/5/7600
Physical Servers RAM Y2	1..12	GB	12/5/7600
Physical Servers RAM Y3	1..12	GB	12/5/7600
Physical Servers RAM Y4	1..12	GB	12/5/7600
Physical Servers RAM Y5	1..12	GB	12/5/7600
Physical Servers RAM Y5Plus		GB	0
Physical Servers Y1	1..12	server	'Ph Servers in Use_Data'/12/5
Physical Servers Y2	1..12	server	'Ph Servers in Use_Data'/12/5
Physical Servers Y3	1..12	server	'Ph Servers in Use_Data'/12/5
Physical Servers Y4	1..12	server	'Ph Servers in Use_Data'/12/5
Physical Servers Y5	1..12	server	'Ph Servers in Use_Data'/12/5
Physical Servers Y5Plus		server	0
Power Demand Inflow Y2		m^2*kg/mo ^4	'Aging Physical Servers Y1'*'Avg Stand-alone Power Demand Y1'[12]
Power Demand Inflow Y3		m^2*kg/mo ^4	'Aging Physical Servers Y2'*'Avg Stand-alone Power Demand Y2'[12]

Power Demand Inflow Y4		m^2*kg/mo^4	'Aging Physical Servers Y3'*'Avg Stand-alone Power Demand Y3'[12]
Power Demand Inflow Y5		m^2*kg/mo^4	'Aging Physical Servers Y4'*'Avg Stand-alone Power Demand Y4'[12]
Power Demand Inflow Y5Plus		m^2*kg/mo^4	'Aging Physical Servers Y5'*'Avg Stand-alone Power Demand Y5'[12]
Power Demand Inflow_Spindles Y2		m^2*kg/mo^4	IF('Monthly Pulse_Copy'>0,'Aging Spindles Y1'*'Avg Power Demand Spindles Y1'[12],0)
Power Demand Inflow_Spindles Y3		m^2*kg/yr^4	IF('Monthly Pulse_Copy'>0,'Aging Spindles Y2'*'Avg Power Demand Spindles Y2'[12],0)
Power Demand Inflow_Spindles Y4		m^2*kg/mo^4	IF('Monthly Pulse_Copy'>0,'Aging Spindles Y3'*'Avg Power Demand Spindles Y3'[12],0)
Power Demand Inflow_Spindles Y5		m^2*kg/mo^4	IF('Monthly Pulse_Copy'>0,'Aging Spindles Y4'*'Avg Power Demand Spindles Y4'[12],0)
Power Demand Inflow_Spindles Y5Plus		m^2*kg/mo^4	IF('Monthly Pulse_Copy'>0,'Aging Spindles Y5'*'Avg Power Demand Spindles Y5'[12],0)
Power Demand of New Servers		KW/server	IF(TIME<3/1/2012,'Avg Power Demand of New Servers_Data_Raw', 248)
Power Demand of New Spindles		W/spindle	18
Power freed up - Standalones Y5Plus		m^2*kg/yr^4	'Ph Servers Outflow'*'Avg Stand-alone Power Demand Y5Plus'
Power freed up from Discarded Servers Y5Plus		m^2*kg/yr^4	'Stand-alone Ph Servers Discarded Y5Plus'*'Avg Stand-alone Power Demand Y5Plus'
Power freed up from Servers Y4	1..12	m^2*kg/yr^4	'Stand-alone Ph Servers Discarded Y4'*'Avg Stand-alone Power Demand Y4'
Power freed up from Servers Y5	1..12	m^2*kg/mo^4	'Stand-alone Ph Servers Discarded Y5'*'Avg Stand-alone Power Demand Y5'
Power freed up from Spindles Y5Plus		m^2*kg/yr^4	'Disengagement of Spindles'*'Avg Power Demand Spindles Y5Plus'
Power freed up Spindles Y4		m^2*kg/mo^4	IF('Monthly Pulse_Copy'>0,'Discard Rate Spindles Y4'*'Avg Power Demand Spindles Y4'[12],0)
Power freed up Spindles Y5		m^2*kg/mo^4	IF('Monthly Pulse_Copy'>0,'Discard Rate Spindles Y5'*'Avg Power Demand Spindles Y5'[12],0)

Power Outflow from Conversion Y2	1..12	m²*kg/yr^4	'Converting Y2 Stand-alones into Hosts'*'Avg Stand-alone Power Demand Y2'
Power Outflow from Conversion Y3	1..12	m²*kg/yr^4	'Converting Y3 Stand-alones into Hosts'*'Avg Stand-alone Power Demand Y3'
Power Outflow from Conversion Y3Plus		m²*kg/yr^4	ARRSUM('Power Outflow from Conversion Y4'+'Power Outflow from Conversion Y5')+'Power Outflow from Conversion Y5Plus'
Power Outflow from Conversion Y4	1..12	m²*kg/yr^4	'Converting Y4 Stand-alones into Hosts'*'Avg Stand-alone Power Demand Y4'
Power Outflow from Conversion Y5	1..12	m²*kg/yr^4	'Converting Y5 Stand-alones into Hosts'*'Avg Stand-alone Power Demand Y5'
Power Outflow from Conversion Y5Plus		m²*kg/yr^4	'Converting Y5Plus Stand-alones into Hosts'*'Avg Stand-alone Power Demand Y5Plus'
Price of Discarded Ph Servers Y4	1..12	NOK/yr	'Discard Rate Y4'*'Avg Price Ph Servers Y4'
Price of Discarded Ph Servers Y5	1..12	NOK/mo	'Discard Rate Y5'*'Avg Price Ph Servers Y5'
Price of Discarded Ph Servers Y5Plus		NOK/yr	'Discard Rate Y5Plus'*'Avg Price Ph Servers Y5Plus'
Price of Discarded Spindles Y4		NOK/mo	IF('Monthly Pulse_Copy'>0,'Discard Rate Spindles Y4'*'Avg Price of Spindles Y4'[12],0)
Price of Discarded Spindles Y5		NOK/mo	IF('Monthly Pulse_Copy'>0,'Discard Rate Spindles Y5'*'Avg Price of Spindles Y5'[12],0)
Price per New Server		NOK/server	10000
Price per New Spindle		NOK/spindle	5000
Price per Server		NOK/server	50000
Processor Requirement per DS User		log_server/user	'Processor Requirements per New FD User'/3
Processor Requirements per New FD User		log_server/user	1/'Users per Logical Server_New FD Users'
PUE			1.25
RAM Discarded Y4	1..12	GB/yr	'Discard Rate Y4'*'Avg RAM Y4'[12]

RAM Discarded Y5	1..12	GB/mo	'Discard Rate Y5'*'Avg RAM Y5'[12]
RAM Discarded Y5Plus		GB/yr	'Discard Rate Y5Plus'*'Avg RAM Y5Plus'
RAM Inflow		GB/mo	'Server Installation'*'RAM of New Servers'
RAM Inflow Y2		GB/mo	'Aging Physical Servers Y1'*'Avg RAM Y1'[12]
RAM Inflow Y3		GB/mo	'Aging Physical Servers Y2'*'Avg RAM Y2'[12]
RAM Inflow Y4		GB/mo	'Aging Physical Servers Y3'*'Avg RAM Y3'[12]
RAM Inflow Y5		GB/mo	'Aging Physical Servers Y4'*'Avg RAM Y4'[12]
RAM Inflow Y5Plus		GB/mo	'Aging Physical Servers Y5'*'Avg RAM Y5'[12]
RAM of New Servers		GB/server	10
RAM Outflow		GB/yr	'Ph Servers Outflow'*'Avg RAM Y5Plus'
RAM Requirement per New DS User		GB/user	'RAM Requirement per New FD User'/3
RAM Requirement per New FD User		GB/user	2
Ratio of New Ph Servers being Converted to Hosts			GRAPHSTEP(TIME, STARTTIME, 1,{0.03,0.03,0.04,0.23,.23//Min:-1;Max:1//})
Ratio of New Ph Servers starting as Standalone			1-'Ratio of New Ph Servers being Converted to Hosts'
Ratio of New Servers Converted to Hosts			MAX(MIN('Hosts Added Divided by Servers Added_Data',1),0)
Ratio of New Virtual Servers to New Ph Servers			GRAPHSTEP(TIME, STARTTIME, 1,{.4,.22,.31,.9,1.34//Min:-1;Max:1//})
Ratio of Stand-alone to Total Ph Servers Y1	1..12		IF('Physical Servers Y1'=0,0,'Stand-alone Ph Servers Y1'/'Physical Servers Y1')
Ratio of Stand-alone to Total Ph Servers Y2	1..12		IF('Physical Servers Y2'=0,0,'Stand-alone Ph Servers Y2'/'Physical Servers Y2')
Ratio of Stand-alone to Total Ph Servers Y3	1..12		IF('Physical Servers Y3'=0,0,'Stand-alone Ph Servers Y3'/'Physical Servers Y3')
Ratio of Stand-alone to Total Ph Servers Y4	1..12		IF('Physical Servers Y4'=0,0,'Stand-alone Ph Servers Y4'/'Physical Servers Y4')
Ratio of Stand-alone to Total Ph Servers Y5	1..12		IF('Physical Servers Y5'=0,0,'Stand-alone Ph Servers Y5'/'Physical Servers Y5')
Ratio of Stand-alone to Total Ph Servers Y5Plus			IF('Physical Servers Y5Plus'=0,0,'Stand-alone Ph Servers Y5Plus'/'Physical Servers Y5Plus')

Recurring Monthly Revenues_Data		NOK/mo	IF(TIME>STARTTIME+62 OR TIME<STARTTIME+12,NAN,1)*'Recurring Monthly Revenues_Data_Raw'
Recurring Monthly Revenues_Data_Raw		NOK/mo	0
Recurring Revenues with Delay		NOK/mo	DELAYPPL('Recurring Monthly Revenues_Data_Raw',2)
Rent Expenses		NOK/yr	'Total Space'*'Renting Price'
Renting Price		NOK/(yr*m^2)	2000
Required Ph Servers to be Installed		server	'Gap between Required & Actual Number of Logical Servers'/(1+'Ratio of New Virtual Servers to New Ph Servers'-'Ratio of New Ph Servers being Converted to Hosts')
Required Ph Servers to be Installed_NonNegative		server	('Required Ph Servers to be Installed'+ABS('Required Ph Servers to be Installed'))/2
Required Spindles to be Installed		spindle	'Gap between Required & Actual Storage'/'Storage per New Spindle'
Required Spindles to be Installed_NonNegative		spindle	('Required Spindles to be Installed'+ABS('Required Spindles to be Installed'))/2
Revenue based Acquisition		server	MAX('Accumulated Revenue of Last Yr'*'Fraction of Revenue for Server Acquisition'/'Price per Server','MIN Budget for Servers'/'Price per Server')
Revenue of 1st Yr		NOK	STEP(270000000,STARTTIME)-STEP(270000000,STARTTIME+2)
Revenue of Last 1 yr		NOK	'Sampled Acc Revenue Now'-'Sampled Acc Revenue 1 yr ago'
RMSE_Hosts		%	(SQRT(RUNAVERAGE((Hosts_Data-'Total Hosts')^2)))/'Avg Number of Hosts_Data'
RMSE_Logical Servers		%	(SQRT(RUNAVERAGE(('Logical Servers_Data'-'Total Logical Servers')^2)))/'Avg Number of Logical Servers_Data'
RMSE_Ph Servers		%	(SQRT(RUNAVERAGE(('Ph Servers in Use_Data'-'Total Physical Servers')^2)))/'Avg Number of Ph Servers_Data'
RMSE_Spindles		%	(SQRT(RUNAVERAGE(('Spindles in Use_Data'-'Total Spindles')^2)))/'Avg Number of Spindles_Data'

RMSE_Storage			(SQRT(RUNAVERAGE(('Storage in Use_Data'-'Total Storage')^2)))/'Avg Storage_Data'
RMSE_Users		%	(SQRT(RUNAVERAGE((Users_Data-'Total Users')^2)))/'Avg Number of Users_Data'
RMSE_Virt Servers		%	(SQRT(RUNAVERAGE(('Virtual Servers in Use_Data'-'Total Virtual servers')^2)))/'Avg Number of V Server_Data'
RunAvg_CPU Cores		Core	RUNAVERAGE('Total CPU Cores')
RunAvg_CPU Cores_Data		Core	RUNAVERAGE('CPU Cores_Data')
RunAvg_Hosts		host	RUNAVERAGE('Total Hosts')
RunAvg_Hosts_Data		host	RUNAVERAGE(Hosts_Data)
RunAvg_Logical Servers		server	RUNAVERAGE('Total Logical Servers')
RunAvg_Logical Servers_Data		server	RUNAVERAGE('Logical Servers_Data')
RunAvg_Ph Servers in Use_Data		server	RUNAVERAGE('Ph Servers in Use_Data')
RunAvg_Spindles		spindle	RUNAVERAGE('Total Spindles')
RunAvg_Spindles_Data		spindle	RUNAVERAGE('Spindles in Use_Data')
RunAvg_Storage in Use_Data		TB	RUNAVERAGE('Storage in Use_Data')
RunAvg_Total Ph Servers		server	RUNAVERAGE('Total Physical Servers')
RunAvg_Total Storage		TB	RUNAVERAGE('Total Storage')
RunAvg_Users		user	RUNAVERAGE('Total Users')
RunAvg_Users_Data		user	RUNAVERAGE(Users_Data)
RunAvg_V Server		virt_server	RUNAVERAGE('Total Virtual servers')
RunAvg_V Server in Use_Data		virt_server	RUNAVERAGE('Virtual Servers in Use_Data')
Salary per Workforce		NOK/(yr*person)	500000
Sampled Acc Revenue 1 yr ago		NOK	DELAYPPL('Sampled Acc Revenue Now',12,0000000)
Sampled Acc Revenue Now		NOK	SAMPLE('Accumulated Revenues_Data',STARTTIME,1)
Sampled Acc Revenue_Now		NOK	SAMPLE('Accumulating Revenue',STARTTIME,1)
Sampled Acc Revenue_one year ago		NOK	DELAYPPL('Sampled Acc Revenue_Now',1)
Sampled Acc Servers Added		server	SAMPLE('Accumulated Servers Added',STARTTIME,1)
Sampled Acc Servers added 1 year ago		server	DELAYPPL('Sampled Acc Servers Added',12,0000000)

Sampled Added Hosts Now		host	SAMPLE('Added Hosts Acc',STARTTIME,1)
Sampled Added Hosts Now_Data		host	SAMPLE('Added Hosts Acc_Data',STARTTIME,1)
Sampled Added Hosts One Yr Ago		host	DELAYPPL('Sampled Added Hosts Now',1)
Sampled Added Hosts One Yr Ago_Data		host	DELAYPPL('Sampled Added Hosts Now_Data',1)
Sampled Added Ph Servers Acc Now		server	SAMPLE('Added Ph Servers Acc',STARTTIME,1)
Sampled Added Ph Servers Acc Now_Data		server	SAMPLE('Added Ph Servers Acc_Data',STARTTIME,1)
Sampled Added Ph Servers Acc One Yr Ago		server	DELAYPPL('Sampled Added Ph Servers Acc Now',1)
Sampled Added Ph Servers Acc One Yr Ago_Data		server	DELAYPPL('Sampled Added Ph Servers Acc Now_Data',1)
Sampled Added Virtual Servers Now		virt_server	SAMPLE('Added Virtual Servers Acc',STARTTIME,1)
Sampled Added Virtual Servers Now_Data		virt_server	SAMPLE('Added Virtual Servers Acc_Data',STARTTIME,1)
Sampled Added Virtual Servers One Yr Ago		virt_server	DELAYPPL('Sampled Added Virtual Servers Now',1)
Sampled Added Virtual Servers One Yr Ago_Data		virt_server	DELAYPPL('Sampled Added Virtual Servers Now_Data',1)
Sampled Ended DS Users Now		user	SAMPLE('DS Users having ended contract',STARTTIME,1)
Sampled Ended DS Users One Year Ago		user	DELAYPPL('Sampled Ended DS Users Now',1)
Sampled Ended FD Users Now		user	SAMPLE('FD Users having ended contract',STARTTIME,1)
Sampled Ended FD Users One Year Ago		user	DELAYPPL('Sampled Ended FD Users Now',1)
Server Installation		server/mo	MIN('Need-based Ph Server Installation','Max Server Installation allowed by Inventory','Max Server Installation Possible allowed by Power','Max Server Installation allowed by Workforce')
Servers Added Last Year		server	'Sampled Acc Servers Added'-'Sampled Acc Servers added 1 year ago'
Share of Hosts from Total Power Demand		%	'Total Hosts Effective Power Demand'/'Total Power Demand of Server & Spindles'
Share of Hosts in Ph Servers		%	'Total Hosts'/'Total Physical Servers'
Share of Hosts in Ph servers_Data		%	Hosts_Data/'Ph Servers in Use_Data'
Share of Ph Servers from Total Power Demand		%	'Total Ph Servers Effective Power Demand'/'Total Power Demand of Server & Spindles'
Share of Spindles from Total Power Demand		%	'Total Spindles Power Demand'/'Total Power Demand of Server & Spindles'

Share of Stand-alones from Total Power Demand		%	'Total Stand-alones Effective Power Demand'/'Total Power Demand of Server & Spindles'
Share of Stand-alones in Ph Servers		%	'Total Ph servers Available as Standalone'/'Total Physical Servers'
Share of Stand-alones in Ph Servers_Data		%	'Standalone Ph Servers_Data'/'Ph Servers in Use_Data'
Share of Virtual Servers in Total New Logical Server Instances		%	'Virtual Servers Inflow'/('Inflow to Stand-alone Ph Servers'+'Virtual Servers Inflow')
Shift CPU Cores Y1	1..11	Core/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'CPU Cores Y1'[i]/TIMESTEP),0)
Shift CPU Cores Y2	1..11	Core/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'CPU Cores Y2'[i]/TIMESTEP),0)
Shift CPU Cores Y3	1..11	Core/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'CPU Cores Y3'[i]/TIMESTEP),0)
Shift CPU Cores Y4	1..11	Core/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'CPU Cores Y4'[i]/TIMESTEP),0)
Shift CPU Cores Y5	1..11	Core/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'CPU Cores Y5'[i]/TIMESTEP),0)
Shift CPU Y1	1..11	CPU/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Physical Servers CPUs Y1'[i]/TIMESTEP),0)
Shift CPU Y2	1..11	CPU/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Physical Servers CPUs Y2'[i]/TIMESTEP),0)
Shift CPU Y3	1..11	CPU/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Physical Servers CPUs Y3'[i]/TIMESTEP),0)
Shift CPU Y4	1..11	CPU/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Physical Servers CPUs Y4'[i]/TIMESTEP),0)
Shift CPU Y5	1..11	CPU/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Physical Servers CPUs Y5'[i]/TIMESTEP),0)
Shift Hosts Power Y1	1..11	m^2*kg/mo^4	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Hosts Power Demand Y1'[i]/TIMESTEP),0)
Shift Hosts Power Y2	1..11	m^2*kg/mo^4	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Hosts Power Demand Y2'[i]/TIMESTEP),0)
Shift Hosts Power Y3	1..11	m^2*kg/mo^4	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Hosts Power Demand Y3'[i]/TIMESTEP),0)

Shift Hosts Y1	1..11	host/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Hosts Y1'[i]/TIMESTEP),0)
Shift Hosts Y2	1..11	host/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Hosts Y2'[i]/TIMESTEP),0)
Shift Hosts Y3	1..11	host/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Hosts Y3'[i]/TIMESTEP),0)
Shift Power Spindles Y1	1..11	m^2*kg/mo^4	IF('Monthly Pulse_Copy'>0,FOR(i=1..11 | 'Spindles Power_Y1'[i]/TIMESTEP),0)
Shift Power Spindles Y2	1..11	m^2*kg/mo^4	IF('Monthly Pulse_Copy'>0,FOR(i=1..11 | 'Spindles Power_Y2'[i]/TIMESTEP),0)
Shift Power Spindles Y3	1..11	m^2*kg/mo^4	IF('Monthly Pulse_Copy'>0,FOR(i=1..11 | 'Spindles Power_Y3'[i]/TIMESTEP),0)
Shift Power Spindles Y4	1..11	m^2*kg/mo^4	IF('Monthly Pulse_Copy'>0,FOR(i=1..11 | 'Spindles Power_Y4'[i]/TIMESTEP),0)
Shift Power Spindles Y5	1..11	m^2*kg/mo^4	IF('Monthly Pulse_Copy'>0,FOR(i=1..11 | 'Spindles Power_Y5'[i]/TIMESTEP),0)
Shift Power Y1	1..11	m^2*kg/mo^4	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Standalones Power_Y1'[i]/TIMESTEP),0)
Shift Power Y2	1..11	m^2*kg/mo^4	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Standalones Power_Y2'[i]/TIMESTEP),0)
Shift Power Y3	1..11	m^2*kg/mo^4	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Standalones Power_Y3'[i]/TIMESTEP),0)
Shift Power Y4	1..11	m^2*kg/mo^4	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Standalones Power_Y4'[i]/TIMESTEP),0)
Shift Power Y5	1..11	m^2*kg/mo^4	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Standalones Power_Y5'[i]/TIMESTEP),0)
Shift Prices Spindles Y1	1..11	NOK/mo	IF('Monthly Pulse_Copy'>0,FOR(i=1..11 | 'Spindles Cumulative Prices Y1'[i]/TIMESTEP),0)
Shift Prices Spindles Y2	1..11	NOK/mo	IF('Monthly Pulse_Copy'>0,FOR(i=1..11 | 'Spindles Cumulative Prices Y2'[i]/TIMESTEP),0)
Shift Prices Spindles Y3	1..11	NOK/mo	IF('Monthly Pulse_Copy'>0,FOR(i=1..11 | 'Spindles Cumulative Prices Y3'[i]/TIMESTEP),0)
Shift Prices Spindles Y4	1..11	NOK/mo	IF('Monthly Pulse_Copy'>0,FOR(i=1..11 | 'Spindles

			Cumulative Prices Y4'[i]/TIMESTEP),0)
Shift Prices Spindles Y5	1..11	NOK/mo	IF('Monthly Pulse_Copy'>0,FOR(i=1..11 | 'Spindles Cumulative Prices Y5'[i]/TIMESTEP),0)
Shift Prices Y1	1..11	NOK/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Physical Servers Cumulative Prices Y1'[i]/TIMESTEP),0)
Shift Prices Y2	1..11	NOK/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Physical Servers Cumulative Prices Y2'[i]/TIMESTEP),0)
Shift Prices Y3	1..11	NOK/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Physical Servers Cumulative Prices Y3'[i]/TIMESTEP),0)
Shift Prices Y4	1..11	NOK/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Physical Servers Cumulative Prices Y4'[i]/TIMESTEP),0)
Shift Prices Y5	1..11	NOK/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Physical Servers Cumulative Prices Y5'[i]/TIMESTEP),0)
Shift RAM Y1	1..11	GB/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Physical Servers RAM Y1'[i]/TIMESTEP),0)
Shift RAM Y2	1..11	GB/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Physical Servers RAM Y2'[i]/TIMESTEP),0)
Shift RAM Y3	1..11	GB/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Physical Servers RAM Y3'[i]/TIMESTEP),0)
Shift RAM Y4	1..11	GB/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Physical Servers RAM Y4'[i]/TIMESTEP),0)
Shift RAM Y5	1..11	GB/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Physical Servers RAM Y5'[i]/TIMESTEP),0)
Shift Spindles Y1	1..11	spindle/mo	IF('Monthly Pulse_Copy'>0,FOR(i=1..11 | 'Spindles Y1'[i]/TIMESTEP),0)
Shift Spindles Y2	1..11	spindle/yr	IF('Monthly Pulse_Copy'>0,FOR(i=1..11 | 'Spindles Y2'[i]/TIMESTEP),0)
Shift Spindles Y3	1..11	spindle/mo	IF('Monthly Pulse_Copy'>0,FOR(i=1..11 | 'Spindles Y3'[i]/TIMESTEP),0)
Shift Spindles Y4	1..11	spindle/mo	IF('Monthly Pulse_Copy'>0,FOR(i=1..11 | 'Spindles Y4'[i]/TIMESTEP),0)
Shift Spindles Y5	1..11	spindle/mo	IF('Monthly Pulse_Copy'>0,FOR(i=1..11 | 'Spindles Y5'[i]/TIMESTEP),0)

Shift Stand-alone Y1	1..11	server/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Stand-alone Ph Servers Y1'[i]/TIMESTEP),0)
Shift Stand-alone Y12	1..11	server/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Stand-alone Ph Servers Y2'[i]/TIMESTEP),0)
Shift Stand-alone Y13	1..11	server/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Stand-alone Ph Servers Y3'[i]/TIMESTEP),0)
Shift Stand-alone Y4	1..11	server/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Stand-alone Ph Servers Y4'[i]/TIMESTEP),0)
Shift Stand-alone Y5	1..11	server/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Stand-alone Ph Servers Y5'[i]/TIMESTEP),0)
Shift Storage Y1	1..11	TB/mo	IF('Monthly Pulse_Copy'>0,FOR(i=1..11 | 'Storage Capacity Spindles Y1'[i]/TIMESTEP),0)
Shift Storage Y2	1..11	TB/mo	IF('Monthly Pulse_Copy'>0,FOR(i=1..11 | 'Storage Capacity Spindles Y2'[i]/TIMESTEP),0)
Shift Storage Y3	1..11	TB/mo	IF('Monthly Pulse_Copy'>0,FOR(i=1..11 | 'Storage Capacity Spindles Y3'[i]/TIMESTEP),0)
Shift Storage Y4	1..11	TB/mo	IF('Monthly Pulse_Copy'>0,FOR(i=1..11 | 'Storage Capacity Spindles Y4'[i]/TIMESTEP),0)
Shift Storage Y5	1..11	TB/mo	IF('Monthly Pulse_Copy'>0,FOR(i=1..11 | 'Storage Capacity Spindles Y5'[i]/TIMESTEP),0)
Shift Virtual Servers Y1	1..11	virt_server /mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Virtual Servers Y1'[i]/TIMESTEP),0)
Shift Virtual Servers Y2	1..11	virt_server /mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Virtual Servers Y2'[i]/TIMESTEP),0)
Shift Virtual Servers Y3	1..11	virt_server /mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Virtual Servers Y3'[i]/TIMESTEP),0)
Shift Y1	1..11	server/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Physical Servers Y1'[i]/TIMESTEP),0)
Shift Y2	1..11	server/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Physical Servers Y2'[i]/TIMESTEP),0)
Shift Y3	1..11	server/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Physical Servers Y3'[i]/TIMESTEP),0)
Shift Y4	1..11	server/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Physical Servers

			Y4'[i]/TIMESTEP),0)
Shift Y5	1..11	server/mo	IF('Monthly Pulse'>0,FOR(i=1..11 | 'Physical Servers Y5'[i]/TIMESTEP),0)
Sliding Average of Expected Installation Time based on HR		mo	SLIDINGAVERAGE('Expected Time to Cover Servers Gap based only on HR Availability',3)
Spindle Installation		spindle/mo	MIN('Need-based Spindle Installation','Max Spindle Installation allowed by Inventory','Max Spindle Installation Possible allowed by Power')
Spindles Cumulative Prices Y1	1..12	NOK	17e5/12/5
Spindles Cumulative Prices Y2	1..12	NOK	17e5/12/5
Spindles Cumulative Prices Y3	1..12	NOK	17e5/12/5
Spindles Cumulative Prices Y4	1..12	NOK	17e5/12/5
Spindles Cumulative Prices Y5	1..12	NOK	17e5/12/5
Spindles Cumulative Prices Y5Plus		NOK	0
Spindles Disengaged_Data		spindle	IF(TIME>STARTTIME+62,NAN,1)*'Spindles Disengaged_Data_Raw'
Spindles Disengaged_Data_Raw		spindle	0
Spindles Error		spindle	'Total Spindles'-'Spindles in Use_Data'
Spindles in Use_Data		spindle	IF(TIME>STARTTIME+62,NAN,1)*'Spindles in Use_Data_Raw'
Spindles in Use_Data_Raw		spindle	0
Spindles Investment_Data		spindle	IF(TIME>STARTTIME+62,NAN,1)*'Spindles Investment_Data_Raw'
Spindles Investment_Data_Raw		spindle	0
Spindles per Ph Server_Data		spindle/server	'Spindles in Use_Data'/'Ph Servers in Use_Data'
Spindles Power Inflow		KW/mo	'Spindle Installation'*'Power Demand of New Spindles'
Spindles Power_Y1	1..12	W	Spindles Y1'*18
Spindles Power_Y2	1..12	W	Spindles Y2'*18
Spindles Power_Y3	1..12	W	Spindles Y3'*18
Spindles Power_Y4	1..12	W	Spindles Y4'*18

Spindles Power_Y5	1..12	W	Spindles Y5'*18
Spindles Power_Y5Plus		KW	0
Spindles Y1	1..12	spindle	'Spindles in Use_Data'/12/5
Spindles Y2	1..12	spindle	'Spindles in Use_Data'/12/5
Spindles Y3	1..12	spindle	'Spindles in Use_Data'/12/5
Spindles Y4	1..12	spindle	'Spindles in Use_Data'/12/5
Spindles Y5	1..12	spindle	'Spindles in Use_Data'/12/5
Spindles Y5Plus		spindle	0
Stand-alone Ph Servers Discarded Y4	1..12	server/yr	'Discard Rate Y4'*'Ratio of Stand-alone to Total Ph Servers Y4'
Stand-alone Ph Servers Discarded Y5	1..12	server/mo	'Discard Rate Y5'*'Ratio of Stand-alone to Total Ph Servers Y5'
Stand-alone Ph Servers Discarded Y5Plus		server/yr	'Discard Rate Y5Plus'*'Ratio of Stand-alone to Total Ph Servers Y5Plus'
Stand-alone Ph Servers Outflow		server/yr	'Ph Servers Outflow'*'Ratio of Stand-alone to Total Ph Servers Y5Plus'
Stand-alone Ph Servers Y1	1..12	server	'Standalone Ph Servers_Data'/12/5
Stand-alone Ph Servers Y2	1..12	server	'Standalone Ph Servers_Data'/12/5
Stand-alone Ph Servers Y3	1..12	server	'Standalone Ph Servers_Data'/12/5
Stand-alone Ph Servers Y4	1..12	server	'Standalone Ph Servers_Data'/12/5
Stand-alone Ph Servers Y5	1..12	server	'Standalone Ph Servers_Data'/12/5
Stand-alone Ph Servers Y5Plus		server	0
Standalone Ph Servers_Data		server	'Ph Servers in Use_Data'-Hosts_Data
Standalones Power Inflow		KW/mo	'Inflow to Stand-alone Ph Servers'*'Power Demand of New Servers'*'Factor for Effective Power Demand for Standalone Ph Servers'
Standalones Power_Y1	1..12	KW	'Total Initial Power Demand of Stand-alone Servers_Data'/12/5
Standalones Power_Y2	1..12	KW	'Total Initial Power Demand of Stand-alone Servers_Data'/12/5
Standalones Power_Y3	1..12	KW	'Total Initial Power Demand of Stand-alone

			Servers_Data'/12/5
Standalones Power_Y4	1..12	KW	'Total Initial Power Demand of Stand-alone Servers_Data'/12/5
Standalones Power_Y5	1..12	KW	'Total Initial Power Demand of Stand-alone Servers_Data'/12/5
Standalones Power_Y5Plus		KW	0
STD Deviation of CPU Cores Error		Core	RUNSTDEV('CPU Cores Error')
STD Deviation of Error div by Avg Storage		%	'STD Deviation of Storage Error'/'Avg Storage_Data'
STD Deviation of Error div by Avg Number of CPU Cores		%	'STD Deviation of CPU Cores Error'/'Avg Number of CPU Cores_Data'
STD Deviation of Error div by Avg Number of Hosts		%	'STD Deviation of Hostsr Error'/'Avg Number of Hosts_Data'
STD Deviation of Error div by Avg Number of Logical Servers		%	'STD Deviation of Logical Servers Error'/'Avg Number of Logical Servers_Data'
STD Deviation of Error div by Avg Number of Ph Servers		%	'STD Deviation of Ph Server Error'/'Avg Number of Ph Servers_Data'
STD Deviation of Error div by Avg Number of Spindles		%	'STD Deviation of Spindles Error'/'Avg Number of Spindles_Data'
STD Deviation of Error div by Avg Number of Users		%	'STD Deviation of Users Error'/'Avg Number of Users_Data'
STD Deviation of Error div by Avg Number of V Server		%	'STD Deviation of V Server Error'/'Avg Number of V Server_Data'
STD Deviation of Hostsr Error		host	RUNSTDEV('Hosts Error')
STD Deviation of Logical Servers Error		server	RUNSTDEV('Logical Servers Error')
STD Deviation of Ph Server Error		server	RUNSTDEV('Ph Server Error')
STD Deviation of Spindles Error		spindle	RUNSTDEV('Spindles Error')
STD Deviation of Storage Error		TB	RUNSTDEV('Storage Error')
STD Deviation of Users Error		user	RUNSTDEV('Users Error')
STD Deviation of V Server Error		virt_server	RUNSTDEV('V Server Error')
STD_Deviation_CPU Cores		Core	RUNSTDEV('Total CPU Cores')
STD_Deviation_CPU Cores_Data		Core	RUNSTDEV('CPU Cores_Data')
STD_Deviation_Hosts		host	RUNSTDEV('Total Hosts')
STD_Deviation_Hosts_Data		host	RUNSTDEV(Hosts_Data)

STD_Deviation_Logical Servers		server	RUNSTDEV('Total Logical Servers')
STD_Deviation_Logical Servers_Data		server	RUNSTDEV('Logical Servers_Data')
STD_Deviation_Ph Servers in Use_Data		server	RUNSTDEV('Ph Servers in Use_Data')
STD_Deviation_Spindles		spindle	RUNSTDEV('Total Spindles')
STD_Deviation_Spindles in Use_Data		spindle	RUNSTDEV('Spindles in Use_Data')
STD_Deviation_Storage in Use_Data		TB	RUNSTDEV('Storage in Use_Data')
STD_Deviation_Total Ph Servers		server	RUNSTDEV('Total Physical Servers')
STD_Deviation_Total Storage		TB	RUNSTDEV('Total Storage')
STD_Deviation_Users		user	RUNSTDEV('Total Users')
STD_Deviation_Users_Data		user	RUNSTDEV(Users_Data)
STD_Deviation_V Server		virt_server	RUNSTDEV('Total Virtual servers')
STD_Deviation_V Server in Use_DataV Server		virt_server	RUNSTDEV('Virtual Servers in Use_Data')
Storage Capacity Decrease Y5Plus		GB/yr	'Disengagement of Spindles'*'Avg Storage of Spindles Y5Plus'
Storage Capacity Spindles Y1	1..12	TB	'Storage in Use_Data'/12/5
Storage Capacity Spindles Y2	1..12	TB	'Storage in Use_Data_Raw'/12/5
Storage Capacity Spindles Y3	1..12	TB	'Storage in Use_Data_Raw'/12/5
Storage Capacity Spindles Y4	1..12	TB	'Storage in Use_Data_Raw'/12/5
Storage Capacity Spindles Y5	1..12	TB	'Storage in Use_Data_Raw'/12/5
Storage Capacity Spindles Y5Plus		TB	0
Storage Discarded Y4		GB/mo	IF('Monthly Pulse_Copy'>0,'Discard Rate Spindles Y4'*'Avg Storage of Spindles Y4'[12],0)
Storage Discarded Y5		GB/mo	IF('Monthly Pulse_Copy'>0,'Discard Rate Spindles Y5'*'Avg Storage of Spindles Y5'[12],0)
Storage Disengaged_Data		TB	IF(TIME>STARTTIME+62,NAN,1)*'Storage Disengaged_Data_Raw'
Storage Disengaged_Data_Raw		TB	0
Storage Error		TB	'Total Storage'-'Storage in Use_Data'
Storage in Use_Data		TB	IF(TIME>STARTTIME+62,NAN,1)*'Storage in Use_Data_Raw'
Storage in Use_Data_Raw		TB	0

Storage Increase Y1		TB/mo	'Spindle Installation'*'Storage per New Spindle'
Storage Increase Y2		GB/mo	IF('Monthly Pulse_Copy'>0,'Aging Spindles Y1'*'Avg Storage of Spindles Y1'[12],0)
Storage Increase Y3		GB/yr	IF('Monthly Pulse_Copy'>0,'Aging Spindles Y2'*'Avg Storage of Spindles Y2'[12],0)
Storage Increase Y4		GB/mo	IF('Monthly Pulse_Copy'>0,'Aging Spindles Y3'*'Avg Storage of Spindles Y3'[12],0)
Storage Increase Y5		GB/mo	IF('Monthly Pulse_Copy'>0,'Aging Spindles Y4'*'Avg Storage of Spindles Y4'[12],0)
Storage Increase Y5Plus		GB/mo	IF('Monthly Pulse_Copy'>0,'Aging Spindles Y5'*'Avg Storage of Spindles Y5'[12],0)
Storage Investment_Data		TB	IF(TIME>STARTTIME+62,NAN,1)*'Storage investment_Data_Raw'
Storage investment_Data_Raw		TB	0
Storage per New Spindle		TB/spindle	GRAPHSTEP(TIME, STARTTIME, 1,{0.3,0.3,0.3,0.3,0.7//Min:0;Max:1//}) //'TB per New Spindle_Data'
Storage per New Spindle_Data		TB/spindle	SAMPLEIMM('Spindles Investment_Data'<>0,'Storage Investment_Data'/'Spindles Investment_Data')
Storage per Ph Server_Data		TB/server	'Storage in Use_Data'/'Ph Servers in Use_Data'
Storage per SPindle_Data		TB/spindle	'Storage in Use_Data'/'Spindles in Use_Data'
Storage per User		TB/user	'Total Storage'/'Total Users'
Storage per User_Data		GB/user	'Storage in Use_Data'/Users_Data
Storage Requirement per New DS User		GB/user	'Storage Requirements per New FD User'/3
Storage Requirements per New FD User		GB/user	GRAPH(TIME, STARTTIME, 1,{16,36,83,156,300,300,300//Min:-1;Max:310//})
Target for New DS Users		user	'Target for Total New Users'*'Percentage of Distributed Services Users in New Contracts'
Target for New FD Users		user	'Target for Total New Users'*(1-'Percentage of Distributed Services Users in New Contracts')
Target for Percentage of Annual Growth			LN('User Growth Derived from Data'+1)
Target for Total New Users		user	'Total Users'*'Target for Percentage of Annual Growth'

Time to Acquire Servers		mo	1
Time to Acquire Spindles		wk	GRAPHSTEP(TIME,STARTTIME,1,{2,2,2,999,99,2})
Time to Hire		mo	GRAPHSTEP(TIME,STARTTIME,1,{6,6,6,6,6})
Time to Install Ph Servers		mo	1
Time to Install Spindles		wk	2
Time to reach Yearly Users Target		mo	12
Total Annual Revenue		NOK/yr	'Total Monthly Revenue'
Total CPU Core Outflow from Ph Server Diseng		Core/yr	ARRSUM('CPU Cores Discarded Y4'+'CPU Cores Discarded Y5')+'CPU Cores Discarded Y5Plus'+'CPU Cores Outflow'
Total CPU Cores		Core	ARRSUM('CPU Cores Y1'+'CPU Cores Y2'+'CPU Cores Y3'+'CPU Cores Y4'+'CPU Cores Y5')+'CPU Cores Y5Plus'
Total CPU Outflow from Ph Server Diseng		CPU/yr	ARRSUM('CPUs Discarded Y4'+'CPUs Discarded Y5')+'CPUs Discarded Y5Plus'+'CPU Outflow'
Total CPUs		CPU	ARRSUM('Physical Servers CPUs Y1'+'Physical Servers CPUs Y2'+'Physical Servers CPUs Y3'+'Physical Servers CPUs Y4'+'Physical Servers CPUs Y5')+'Physical Servers CPUs Y5Plus'
Total Cumulative Price Outflow from Ph Server Diseng		NOK/yr	ARRSUM('Price of Discarded Ph Servers Y4'+'Price of Discarded Ph Servers Y5')+'Price of Discarded Ph Servers Y5Plus'+'Cumulative Prices Outflow'
Total Cumulative Price Outflow from Spindle Disengagement		NOK/mo	'Price of Discarded Spindles Y4'+'Price of Discarded Spindles Y5'+'Cumulative Prices Outflow_Spindles'
Total Depreciation		NOK/yr	'Depreciation Ph Servers'+'Depreciation Spindles'
Total Expenses		NOK/yr	'Energy Expenses'+'Rent Expenses'+'Total Depreciation'+'Financial Expenses'+'HR Expenses'+'Overhead Expenses'
Total Host Lifetime		yr	3+'Avg Host Lifetime After 3 Yrs'
Total Hosts		host	ARRSUM('Hosts Y1'+'Hosts Y2'+'Hosts Y3')+'Hosts Y3Plus'
Total Hosts Disengaged		host	ARRSUM('Hosts Discarded Y2'+'Hosts Discarded Y3'+'Hosts Discarded Y4'+'Host Disengagement')
Total Hosts Effective Power Demand		KW	ARRSUM('Hosts Power Demand Y1'+'Hosts Power Demand Y2'+'Hosts Power Demand Y3')+'Hosts Power Demand

			Y3Plus'
Total Initial Data Center Power		MW	1
Total Initial Power Demand of Hosts_Data		W	Hosts_Data*'Avg Initial Power Demand of Hosts_Data'
Total Initial Power Demand of Stand-alone Servers_Data		KW	'Standalone Ph Servers_Data'*'Avg Initial Power Demand of Stand-alone Servers_Data'
Total KW Demand of Servers_Data		KW	IF(TIME>STARTTIME+62,NAN,1)*'Total Power Demand of Servers_Data_Raw'
Total Logical Servers		server	'Total Ph servers Available as Standalone'+'Total Virtual servers'
Total Monthly Revenue		NOK/mo	'Monthly Revenue from FD Users'+'Monthly Revenue from DS Users'
Total Not Renewing minus Initial Leaving equal New Users Not Renewing		%	'FD Not Renewing by Percentage of Inflow'-'Initial FD Users Leaving by Percentage of Inflow'
Total Outflow of DS Users		user/mo	'DS Users Drainage'+'DS Users Not Renewing Contract'
Total Outflow of FD Users		user/mo	'FD Users Drainage'+'FD Users Not Renewing Contract'
Total Ph Server Disengagement		server/yr	ARRSUM('Discard Rate Y4'+'Discard Rate Y5')+'Discard Rate Y5Plus'+'Ph Servers Outflow'
Total Ph servers Available as Standalone		server	'Total Physical Servers'-'Total Hosts'
Total Ph Servers Effective Power Demand		KW	'Total Stand-alones Effective Power Demand'+'Total Hosts Effective Power Demand'
Total Physical Servers		server	ARRSUM('Physical Servers Y1'+'Physical Servers Y2'+'Physical Servers Y3'+'Physical Servers Y4'+'Physical Servers Y5')+'Physical Servers Y5Plus'
Total Power Demand of Server & Spindles		KW	'Total Ph Servers Effective Power Demand'+'Total Spindles Power Demand'
Total Power Demand of Servers_Data_Raw		KW	0
Total Power freed up from Host Disengagement		W	ARRSUM('Hosts Power freed up_Discard Y2'+'Hosts Power freed up_Discard Y3')+'Hosts Power freed up_Discard Y3Plus'+'Hosts Power Outflow'
Total Power freed up from Spindle Disengagement		m^2*kg/mo^4	'Power freed up Spindles Y4'+'Power freed up Spindles Y5'+'Power freed up from Spindles Y5Plus'
Total Power freed up from Standalone Disengagement		m^2*kg/yr^	ARRSUM('Power freed up from Servers Y4'+'Power freed up

		4	from Servers Y5')+'Power freed up - Standalones Y5Plus'+'Power freed up from Discarded Servers Y5Plus'
Total RAM		GB	ARRSUM('Physical Servers RAM Y1'+'Physical Servers RAM Y2'+'Physical Servers RAM Y3'+'Physical Servers RAM Y4'+'Physical Servers RAM Y5')+'Physical Servers RAM Y5Plus'
Total RAM Outflow from Ph Server Diseng		GB/yr	ARRSUM('RAM Discarded Y4'+'RAM Discarded Y5')+'RAM Discarded Y5Plus'+'RAM Outflow'
Total Required Logical Servers		server	'FD Users Processor Requirements'+'DS Users Processor Requirements'
Total Server Disengagement to report		server	'Total Ph Server Disengagement'*TIMESTEP
Total Space		m²	270+69+STEP(69,STARTTIME+57)
Total Spindle Disengagement		spindle/mo	'Discard Rate Spindles Y4'+'Discard Rate Spindles Y5'+'Disengagement of Spindles'
Total Spindles		spindle	ARRSUM('Spindles Y1'+'Spindles Y2'+'Spindles Y3'+'Spindles Y4'+'Spindles Y5')+'Spindles Y5Plus'
Total Spindles Power Demand		W	ARRSUM('Spindles Power_Y1'+'Spindles Power_Y2'+'Spindles Power_Y3'+'Spindles Power_Y4'+'Spindles Power_Y5')+'Spindles Power_Y5Plus'
Total Stand-alone Ph Server Disengagement		server/yr	ARRSUM('Stand-alone Ph Servers Discarded Y4'+'Stand-alone Ph Servers Discarded Y5')+'Stand-alone Ph Servers Discarded Y5Plus'+'Stand-alone Ph Servers Outflow'
Total Stand-alones Effective Power Demand		KW	ARRSUM('Standalones Power_Y1'+'Standalones Power_Y2'+'Standalones Power_Y3'+'Standalones Power_Y4'+'Standalones Power_Y5')+'Standalones Power_Y5Plus'
Total Storage		TB	ARRSUM('Storage Capacity Spindles Y1'+'Storage Capacity Spindles Y2'+'Storage Capacity Spindles Y3'+'Storage Capacity Spindles Y4'+'Storage Capacity Spindles Y5')+'Storage Capacity Spindles Y5Plus'
Total Storage Capacity Decrease from Spindle Disengagement		GB/mo	'Storage Discarded Y4'+'Storage Discarded Y5'+'Storage Capacity Decrease Y5Plus'
Total Storage Requirements		TB	'FD Users Storage Requirements'+'DS USers Storage

			Requirements'
Total Users		user	'FD Users'+'DS Users'
Total Virtual servers		virt_server	ARRSUM('Virtual Servers Y1'+'Virtual Servers Y2'+'Virtual Servers Y3')+'Virtual Servers Y3Plus'
Upgrade in Data center Power		KW/mo	PULSE(1,STARTTIME+4,1000)+PULSE('Addition of Power Capacity','Date to Add New Power Capacity',1000)
User Growth Derived from Data		%	'User Growth Derived from Data_No Unit'*1%
User Growth Derived from Data_No Unit			GRAPHSTEP(TIME,STARTTIME,1,{41.7,35.9,2.8,9.5,5.9,'Future Growth'})
Users Error		user	'Total Users'-Users_Data
Users per Logical Server		user/server	'Total Users'/'Total Logical Servers'
Users per Logical Server_Data		user/server	Users_Data/'Logical Servers_Data'
Users per Logical Server_Future FD Users			1.6
Users per Logical Server_New FD Users		user/log_server	Users per Logical Server_New FD Users_No Unit'*1
Users per Logical Server_New FD Users_No Unit			GRAPHSTEP(TIME, STARTTIME,1,{18,13,2.7,2.5,1.6,'Users per Logical Server_Future FD Users'//Min:-1;Max:16//})
Users_Data		user	'FD Users_Data'+'DS Users_Data'
Users_Data_Raw		user	0
Users_Oslo_Data_One_Yr_Ago		user	DELAYPPL(Users_Data_Raw,12)
Users_Oslo_One_Yr_Ago		user	DELAYPPL('Total Users',12)
V Server Error		virt_server	'Total Virtual servers'-'Virtual Servers in Use_Data'
Virt Servers per Ph Server_Data			'Virtual Servers in Use_Data'/'Ph Servers in Use_Data'
Virt_Server per Host_Data			'Virtual Servers in Use_Data'/Hosts_Data
Virtual Server Disengagement		virt_server/yr	'Virtual Servers Y3Plus'/'Avg Virt Server Lifetime after 3 Yrs'
Virtual Server per Host			'Total Virtual servers'/'Total Hosts'
Virtual Servers Added		server/mo	'Server Installation'*'Ratio of New Virtual Servers to New Ph Servers'

Virtual Servers Added_Data		virt_server /mo	IF(TIME>STARTTIME+62,NAN,1)*'Virtual servers Added_Data_Raw'
Virtual servers Added_Data_Raw		virt_server /mo	0
Virtual Servers Disengaged_Data		virt_server /mo	IF(TIME>STARTTIME+62,NAN,1)*'Virtual Servers Disengaged_Data_Raw'
Virtual Servers Disengaged_Data_Raw		virt_server /mo	0
Virtual Servers in Use_Data		virt_server	IF(TIME>STARTTIME+62,NAN,1)*'Virtual Servers in Use_Data_Raw'
Virtual Servers in Use_Data_Raw		virt_server	0
Virtual Servers Inflow		server/mo	'Server Installation'*'Ratio of New Virtual Servers to New Ph Servers'
Virtual Servers Inflow Y2		virt_server /mo	IF('Monthly Pulse'>0,'Virtual Servers Y1'[12]/TIMESTEP,0)
Virtual Servers Inflow Y3		virt_server /mo	IF('Monthly Pulse'>0,'Virtual Servers Y2'[12]/TIMESTEP,0)
Virtual Servers Inflow Y3Plus		virt_server /mo	IF('Monthly Pulse'>0,'Virtual Servers Y3'[12]/TIMESTEP,0)
Virtual Servers Y1	1..12	virt_server	'Virtual Servers in Use_Data'/12/4
Virtual Servers Y2	1..12	virt_server	'Virtual Servers in Use_Data'/12/4
Virtual Servers Y3	1..12	virt_server	'Virtual Servers in Use_Data'/12/4
Virtual Servers Y3Plus		virt_server	'Virtual Servers in Use_Data'/4

Appendix B - Stock and Flow Diagram

The most important parts of the stock and flow diagram were described section by section in Chapter Three -. Nevertheless, in order to have a full view of the model without having to open the file in Powersim, in this appendix we have pasted pictures of the full model. The model consists of six tabs for: Servers, Spindles, Users, Data Center, Accounting & Finance, and Operational Accounting. The last tab (Oper. Accounting) is not presented in this section because it involves only arithmetically derived variables from main model variables and there is no dynamic involved. These variables include ones for calculating total sums, averages, etc.

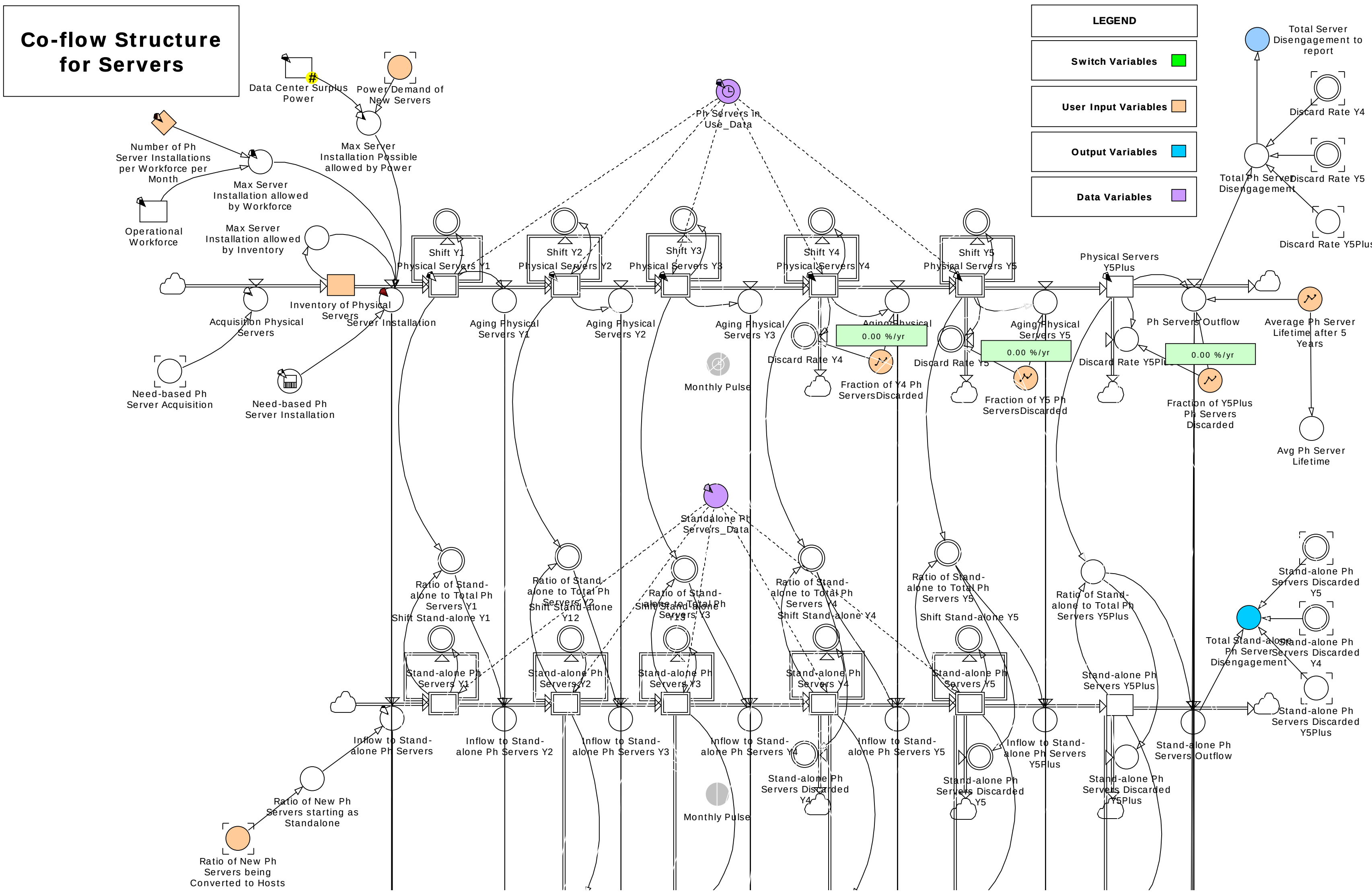


Figure 109 - Stock and Flow Diagram: Physical Servers and Stand-alones

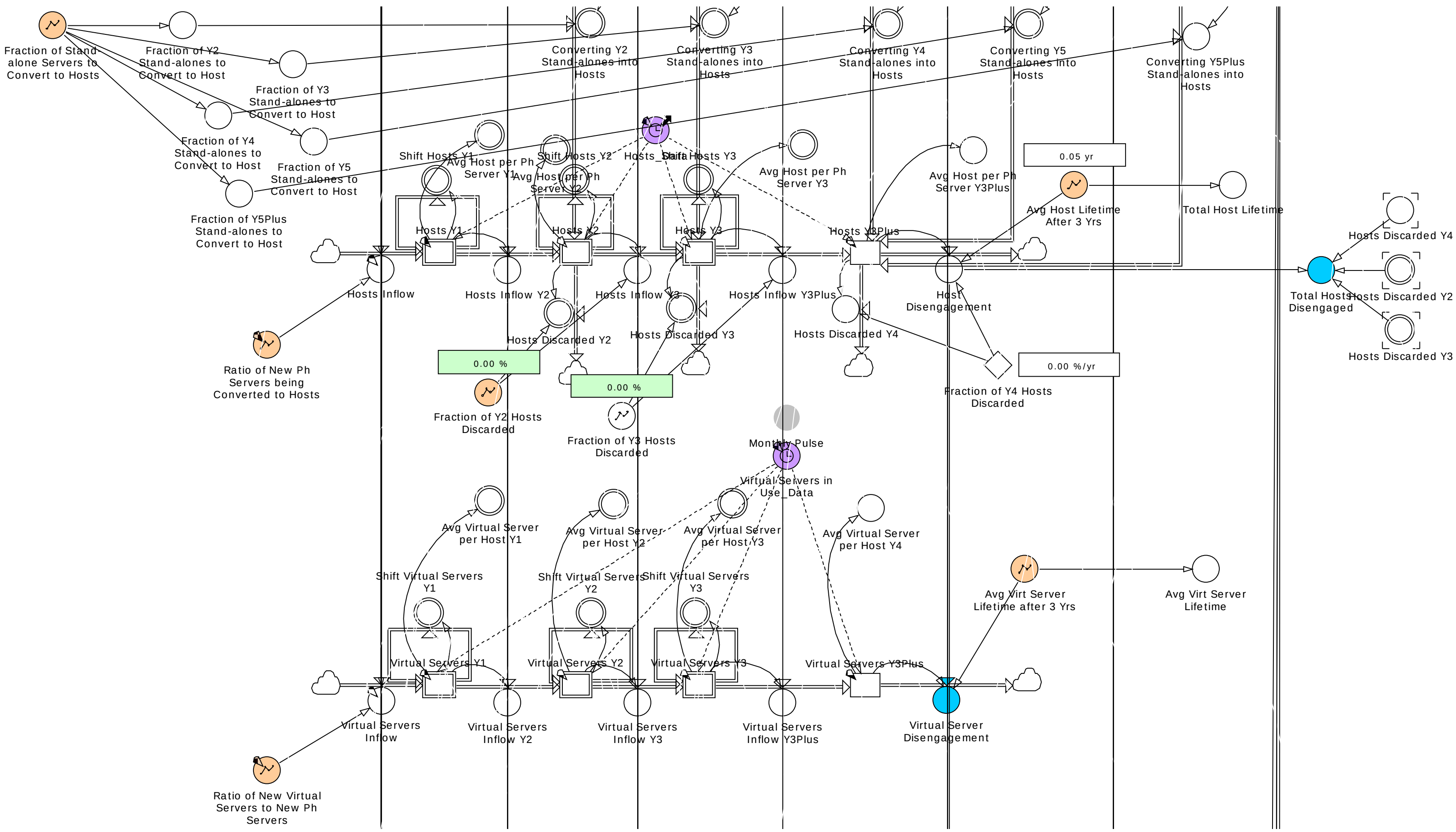


Figure 110 - Stock and Flow Diagram: Hosts and Virtual Servers

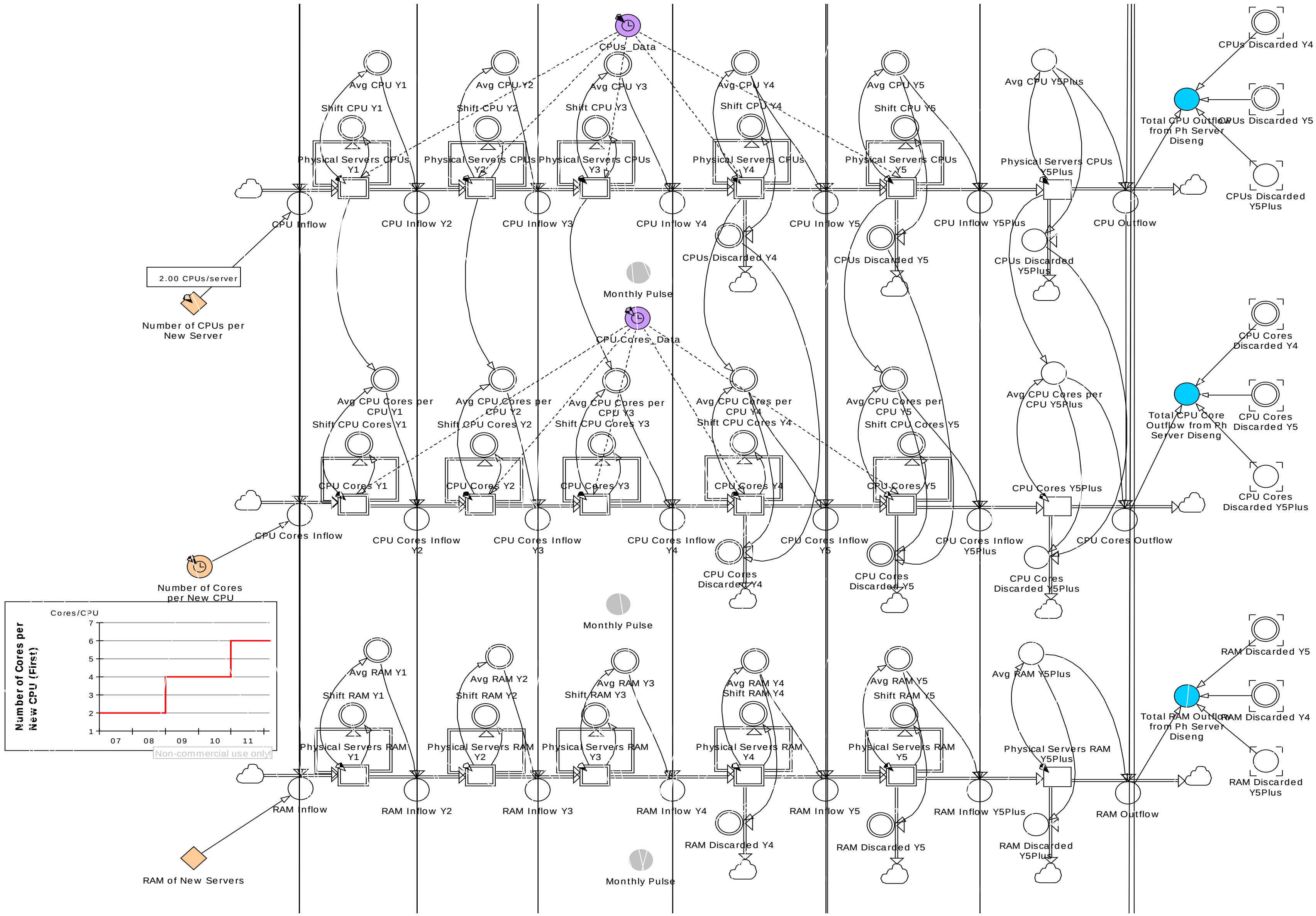


Figure 111 - Stock and Flow Diagram: CPUs, CPU Cores, and RAM

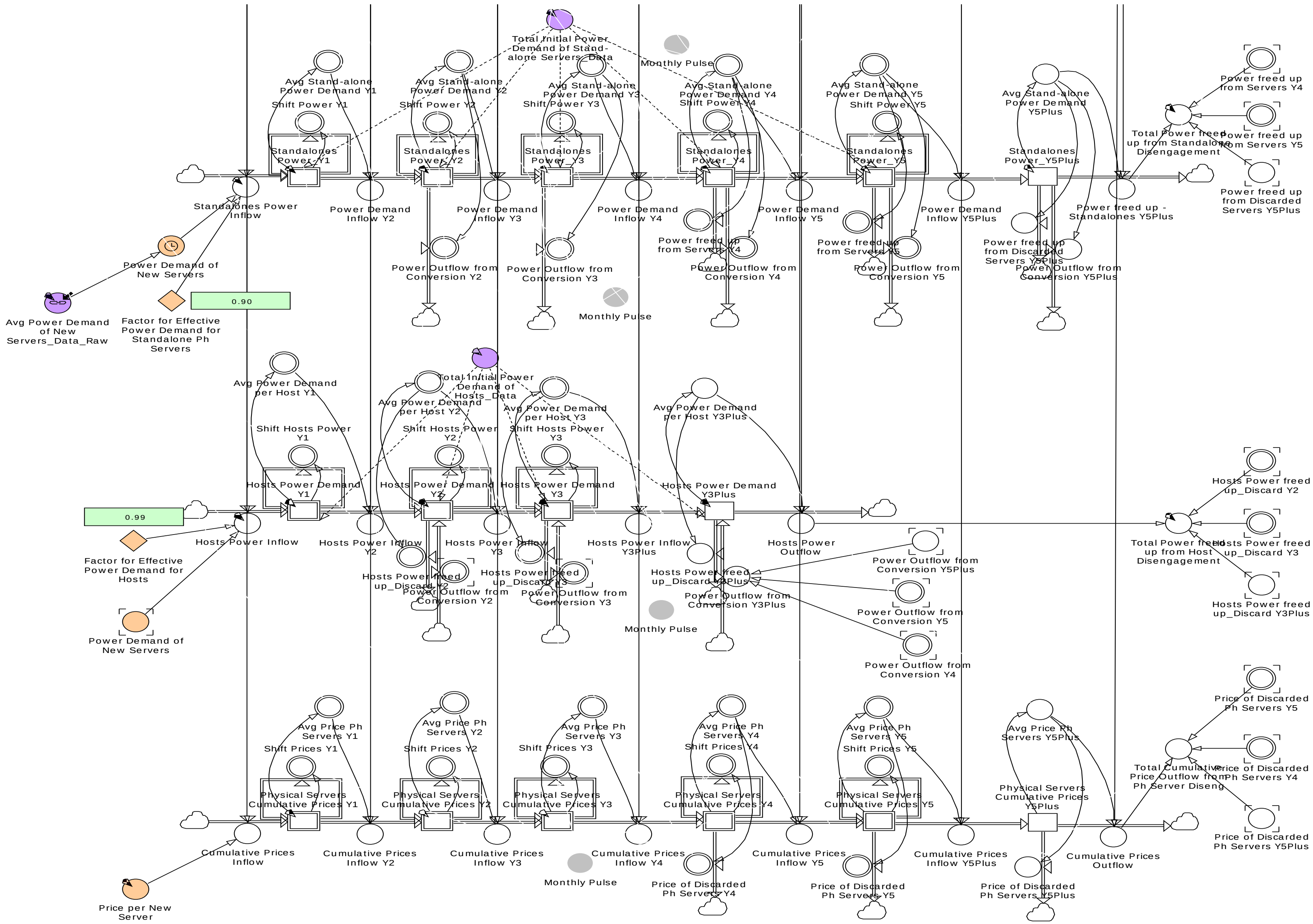


Figure 112 - Stock and Flow Diagram: Power Demand and Server Prices

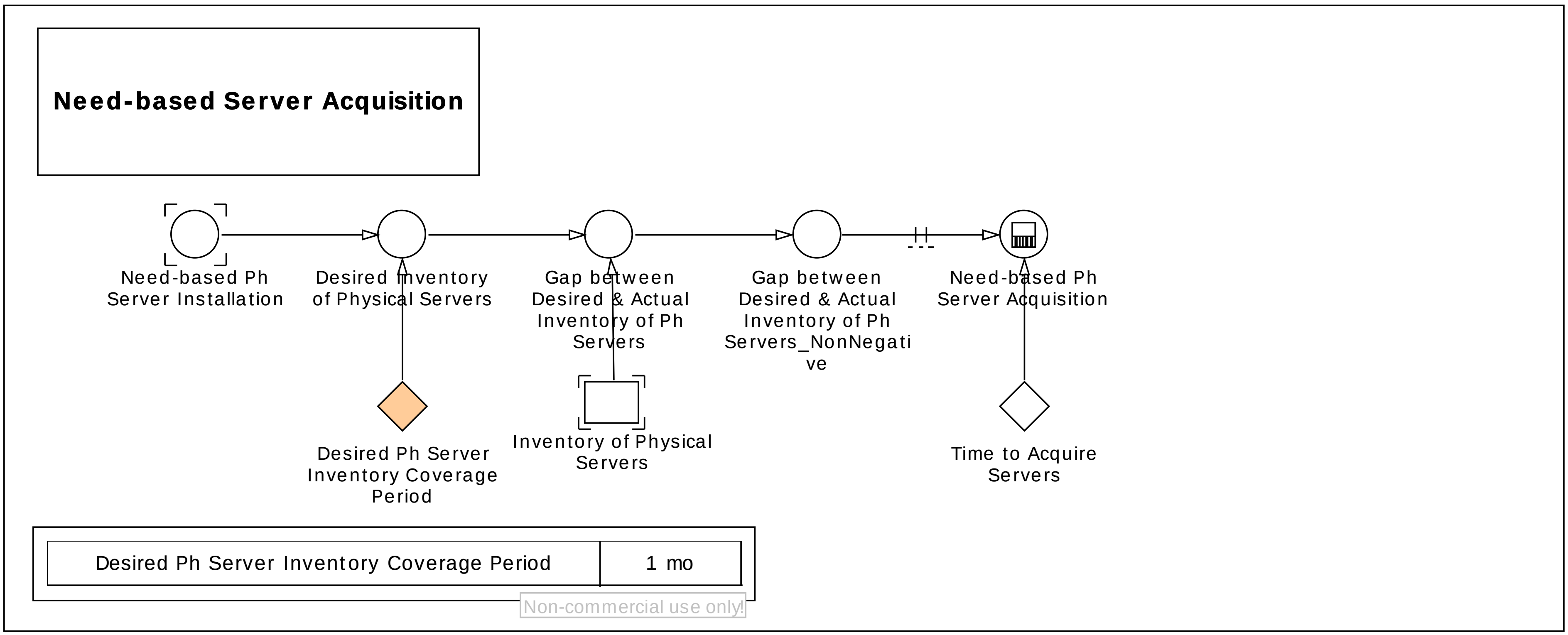


Figure 113 - Need-based Server Acquisition

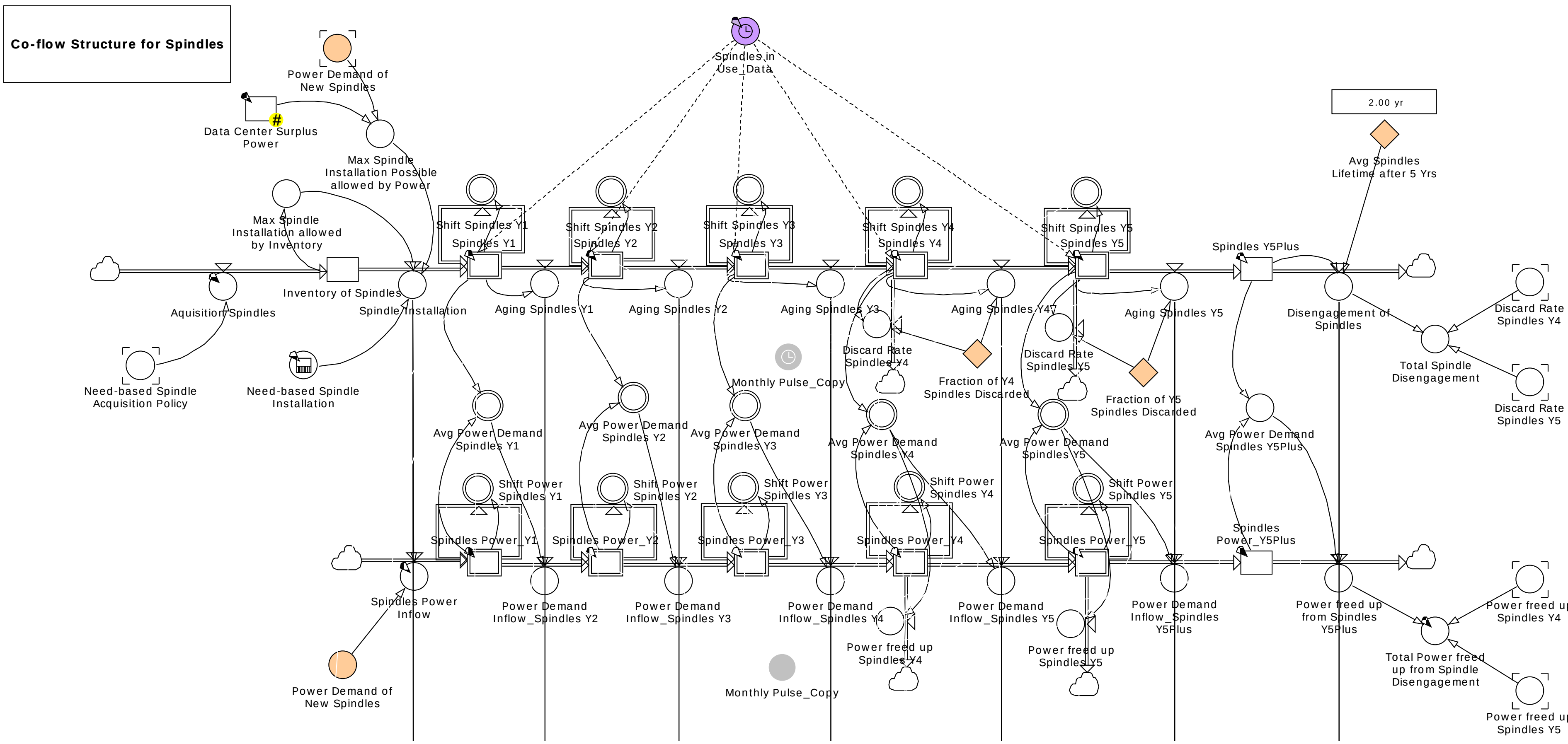


Figure 114 - Stock and Flow Diagram: Spindles and Spindles' Power Demand

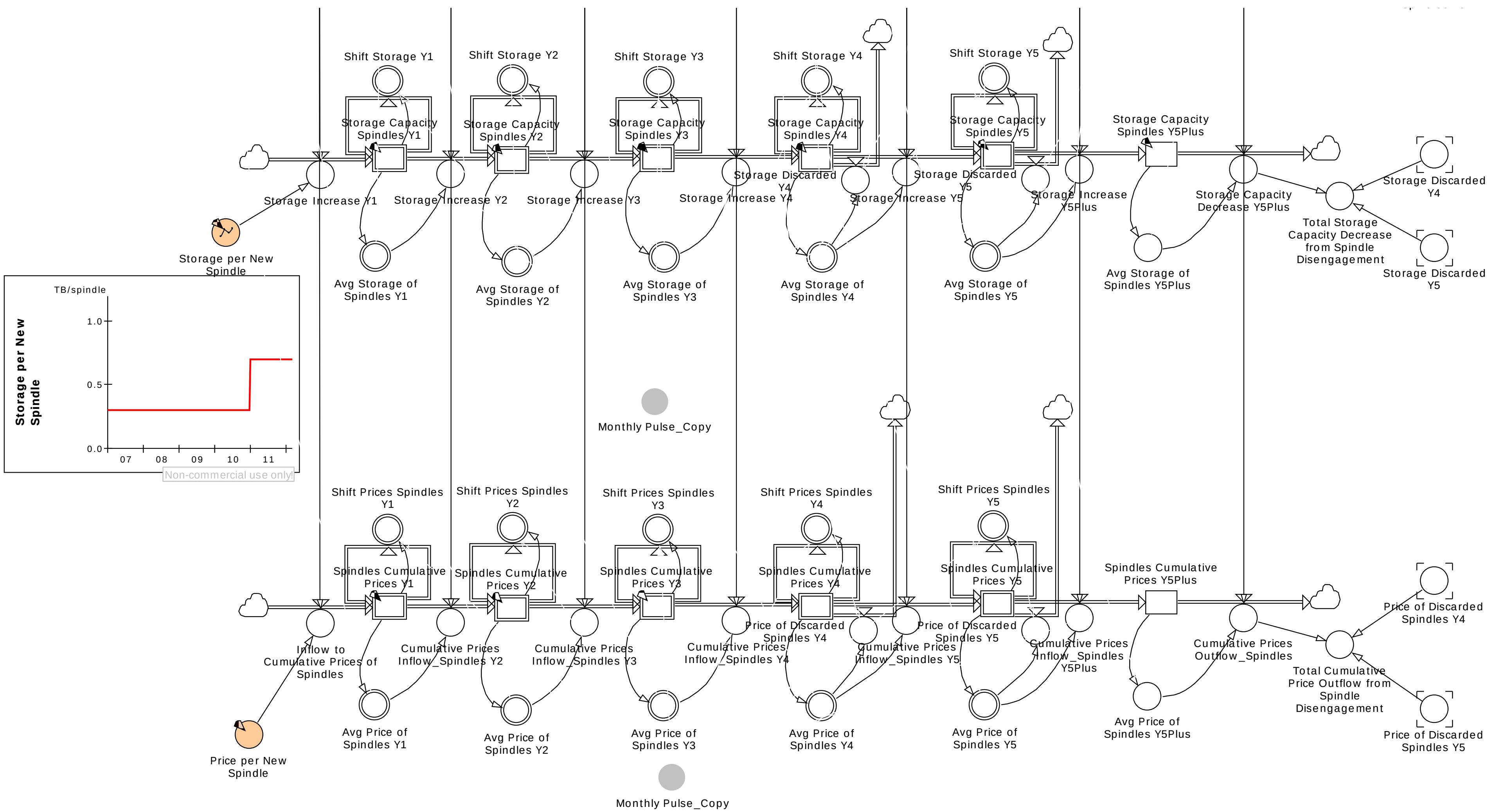


Figure 115 - Spindles' Storage Capacity and Prices

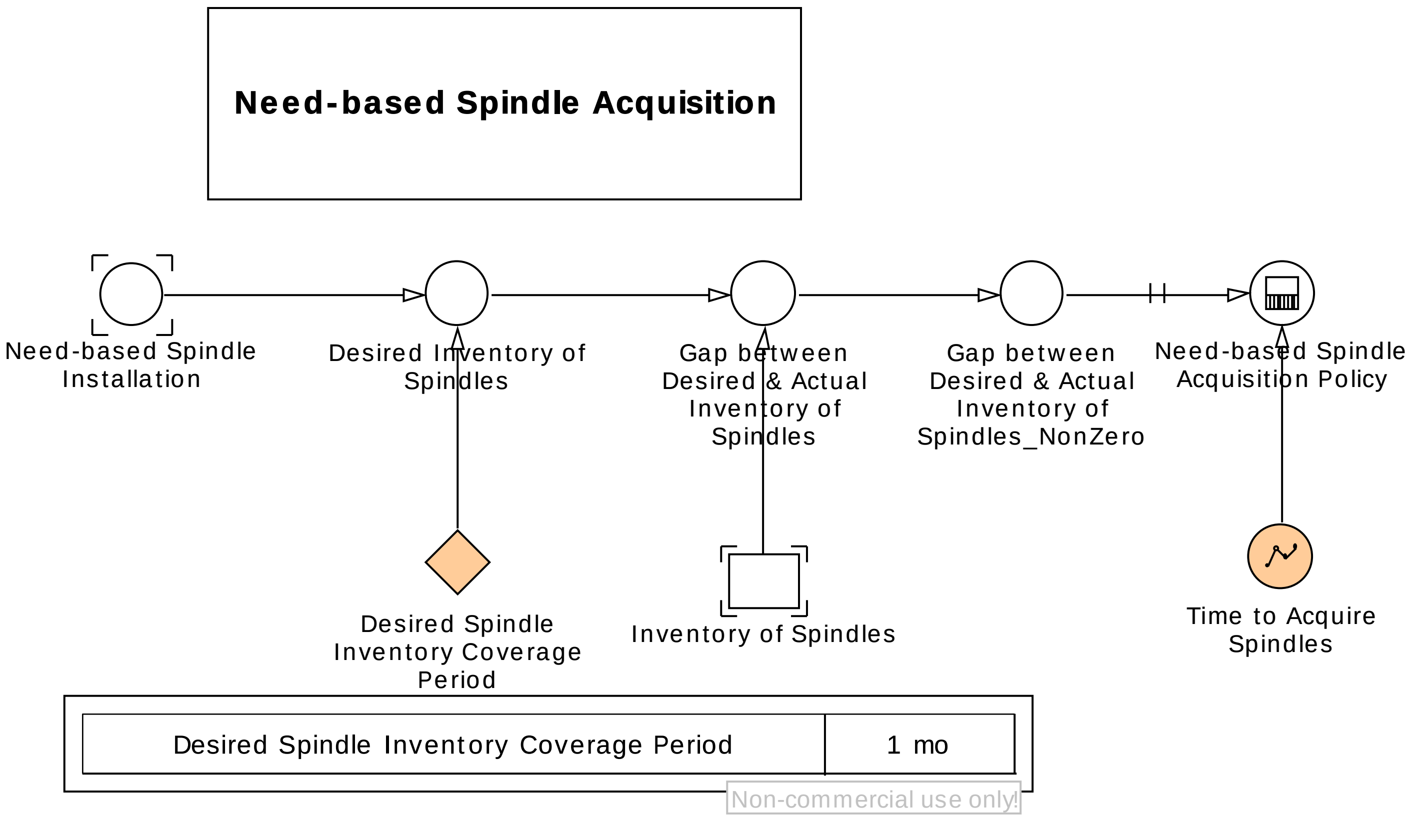


Figure 116 - Need-based Spindle Acquisition

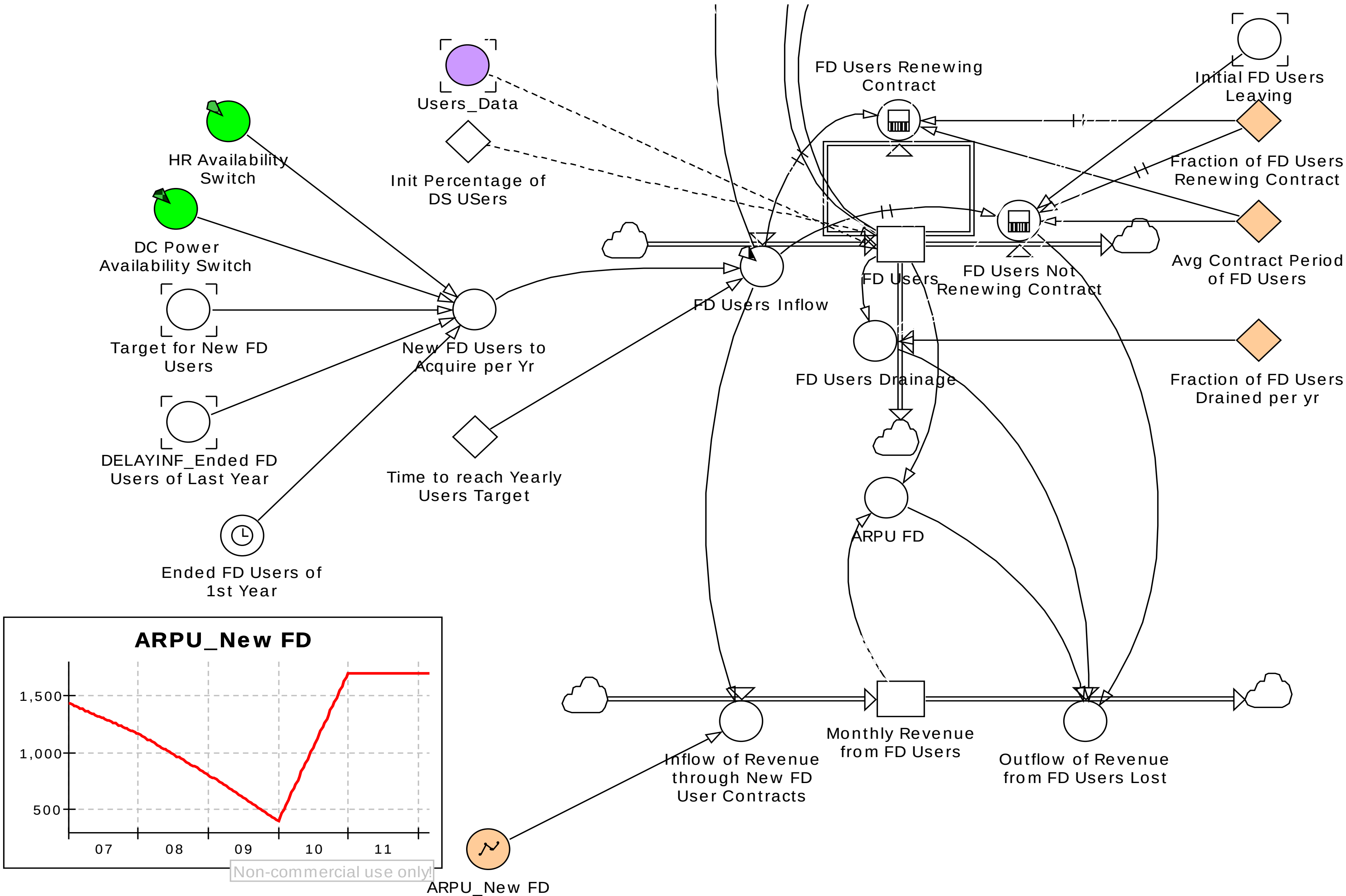


Figure 117 - FD Users and Revenues

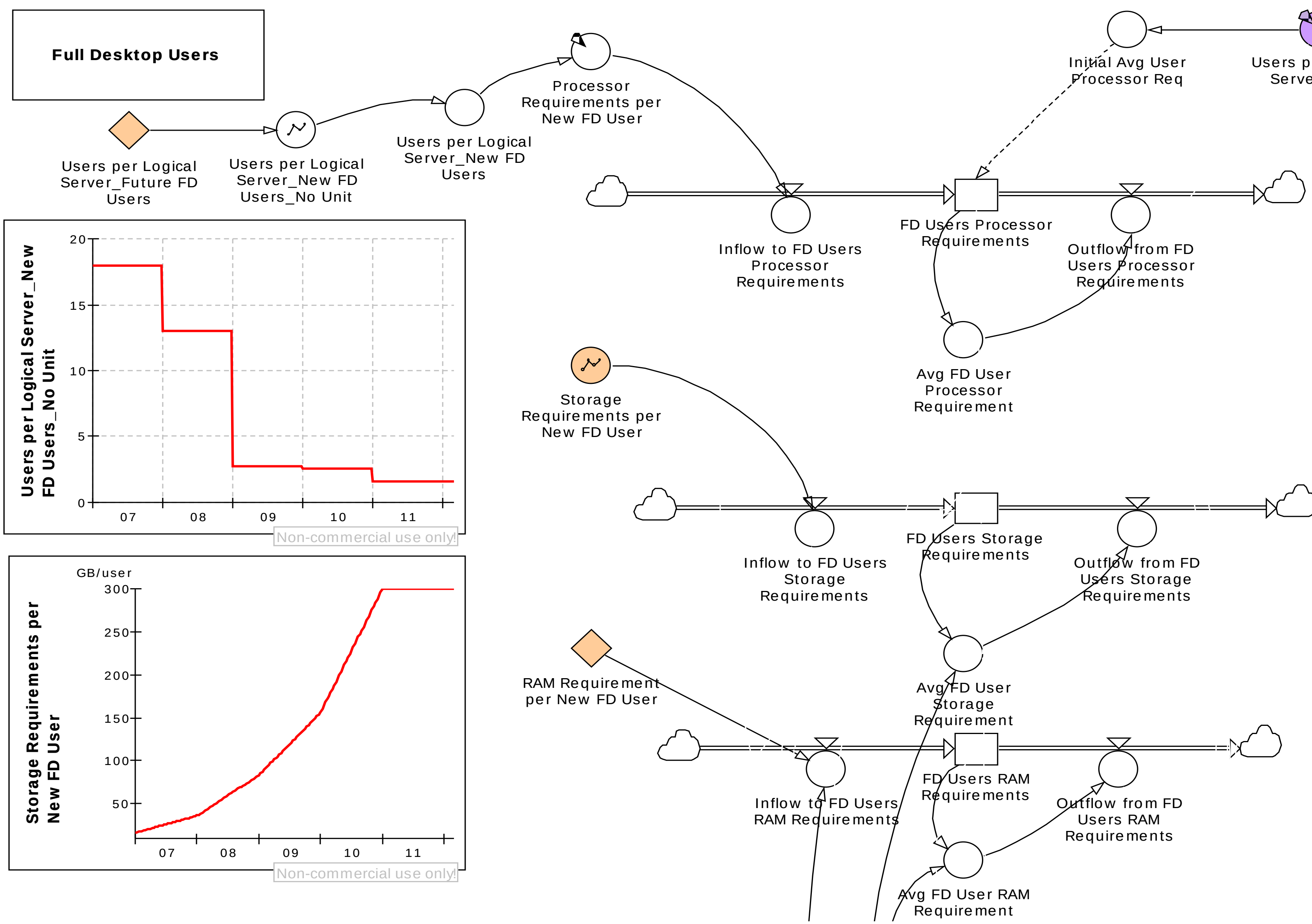


Figure 118 - FD Users' Requirements

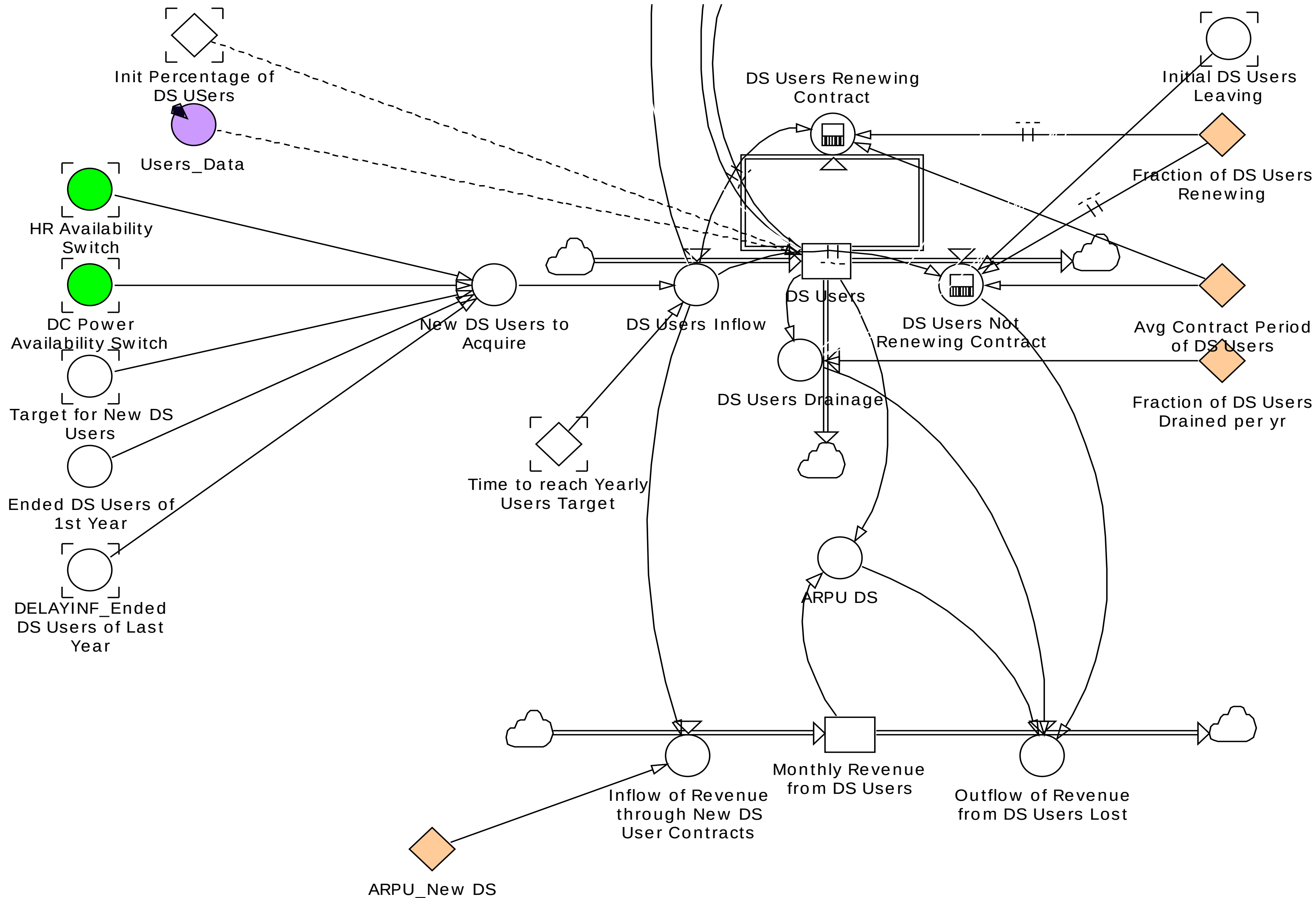


Figure 119 - DS Users and Revenues

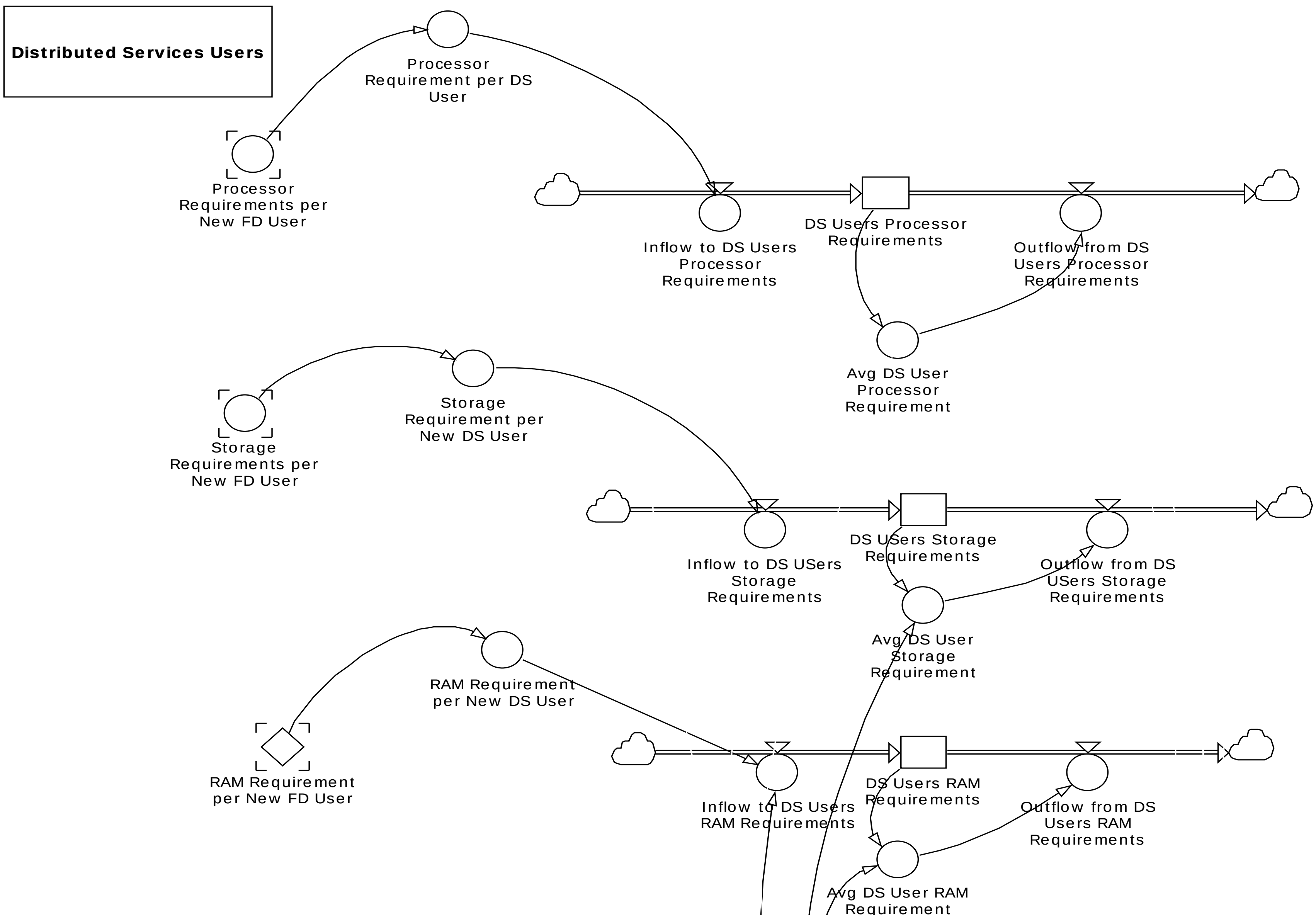


Figure 120 - DS Users' Requirements

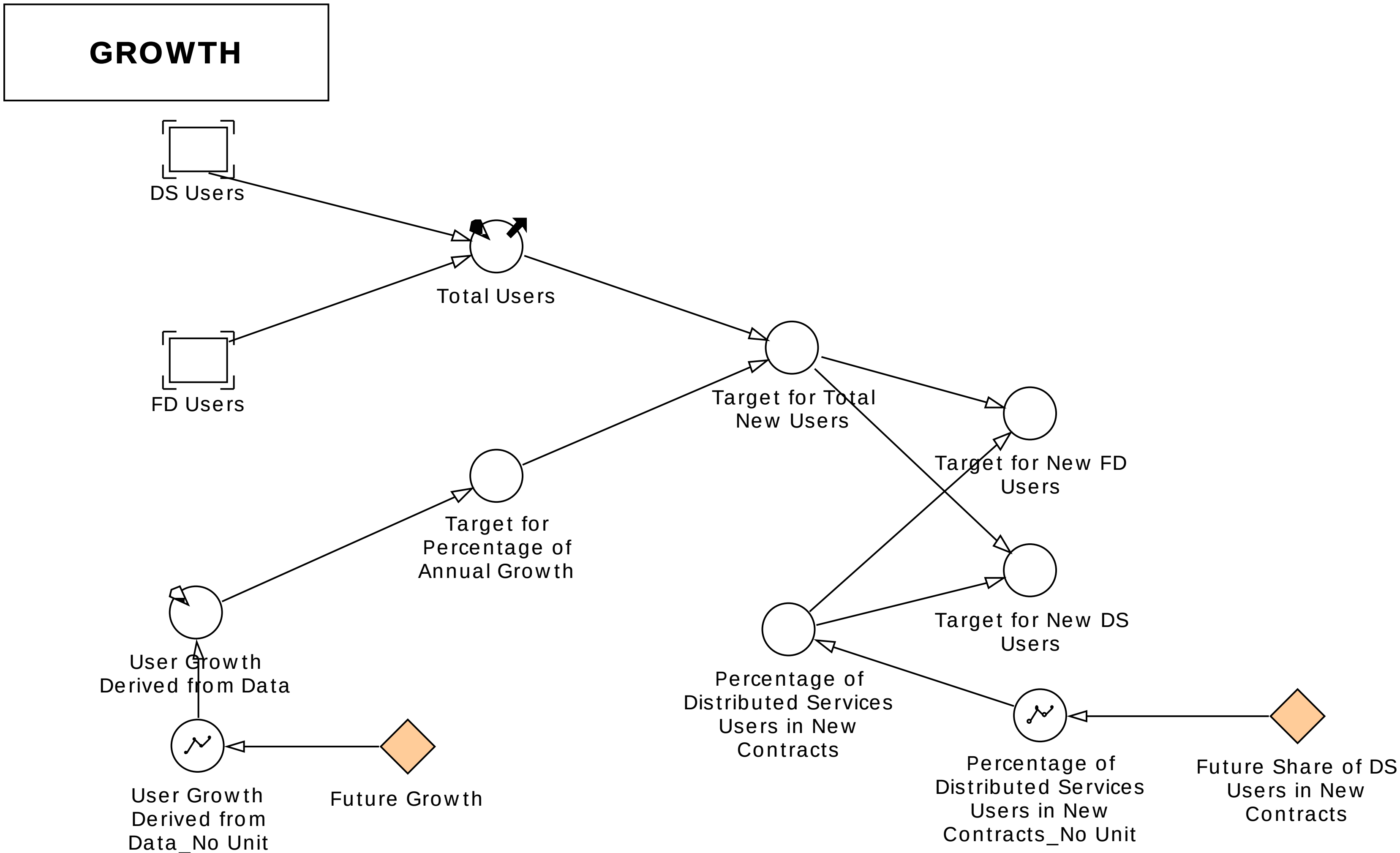


Figure 121 - Users' Growth

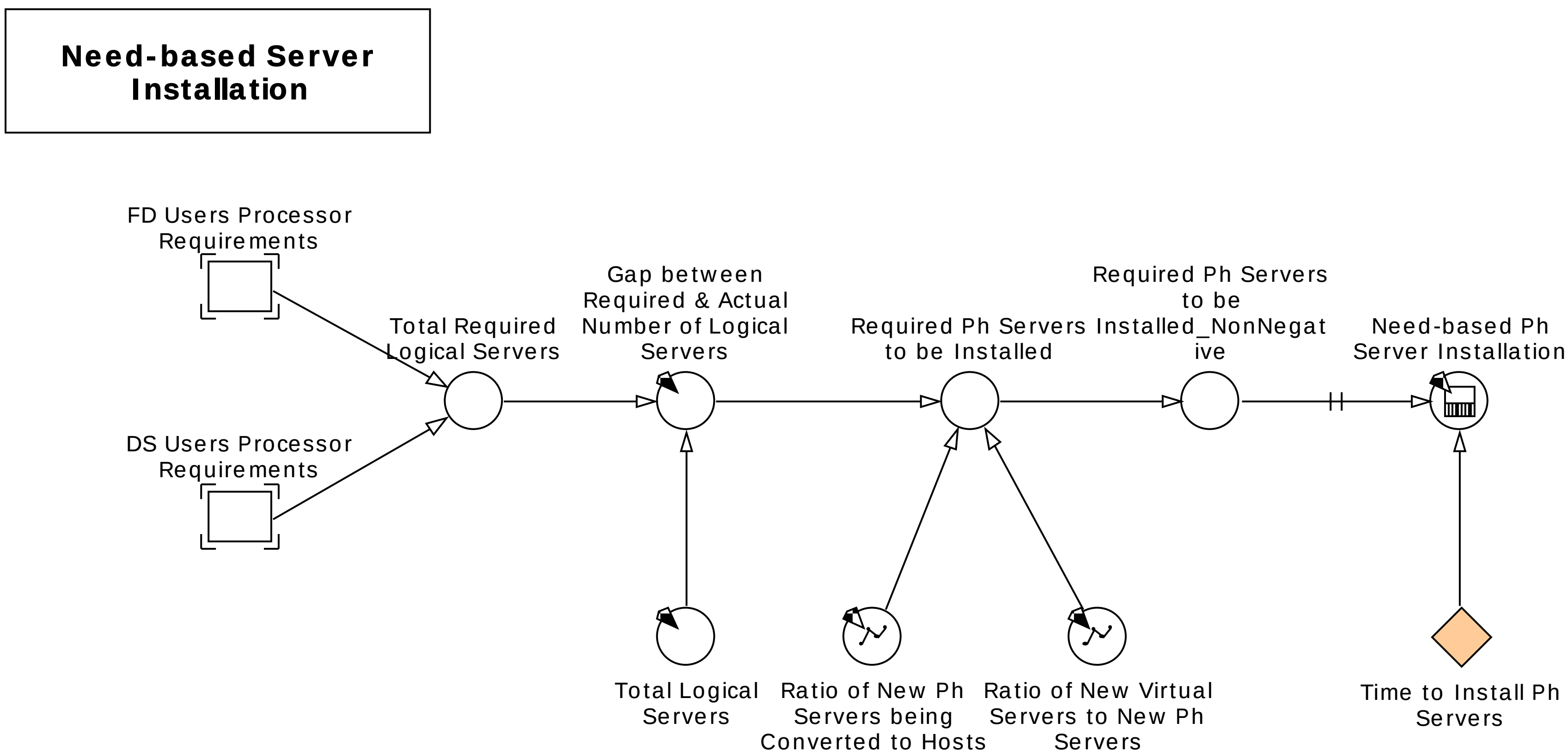


Figure 122 - Need-based Server Installation

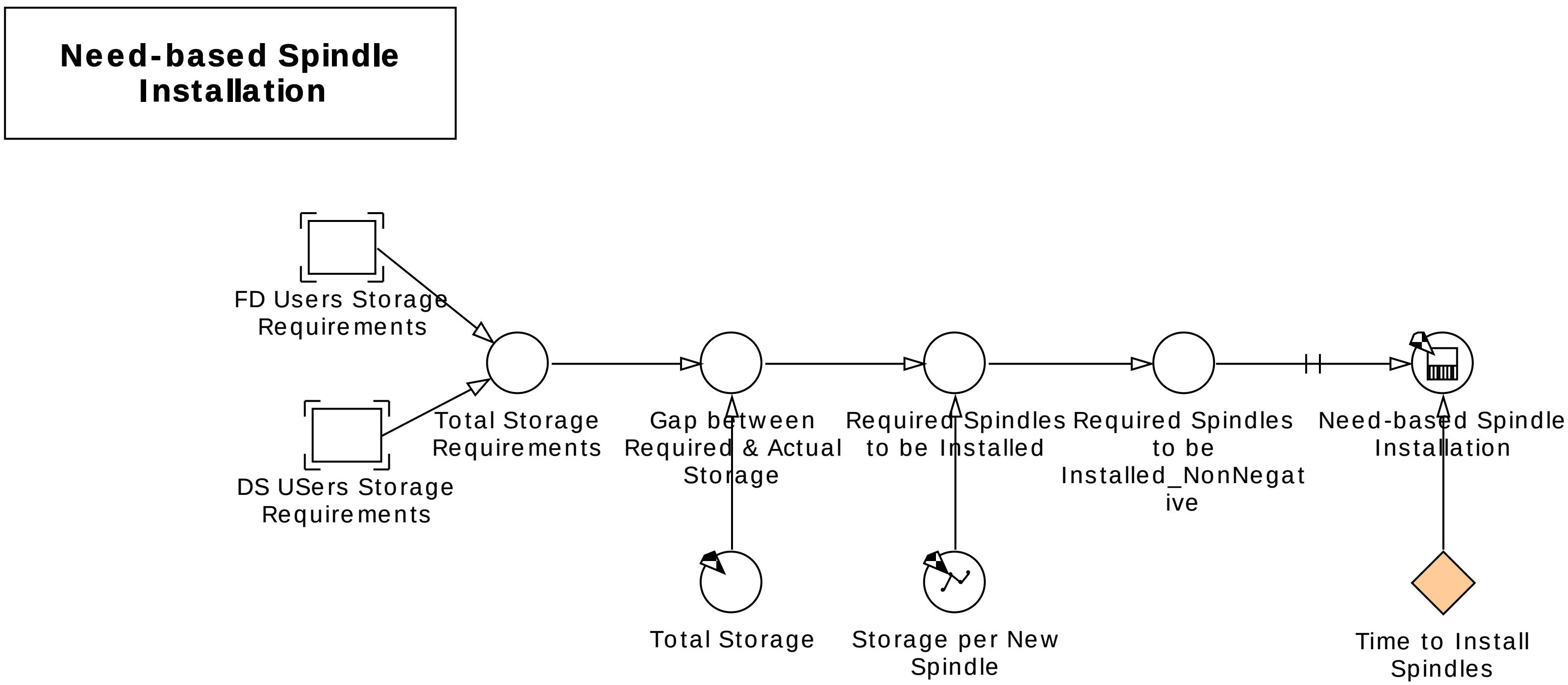


Figure 123 - Need-based Spindle Installation

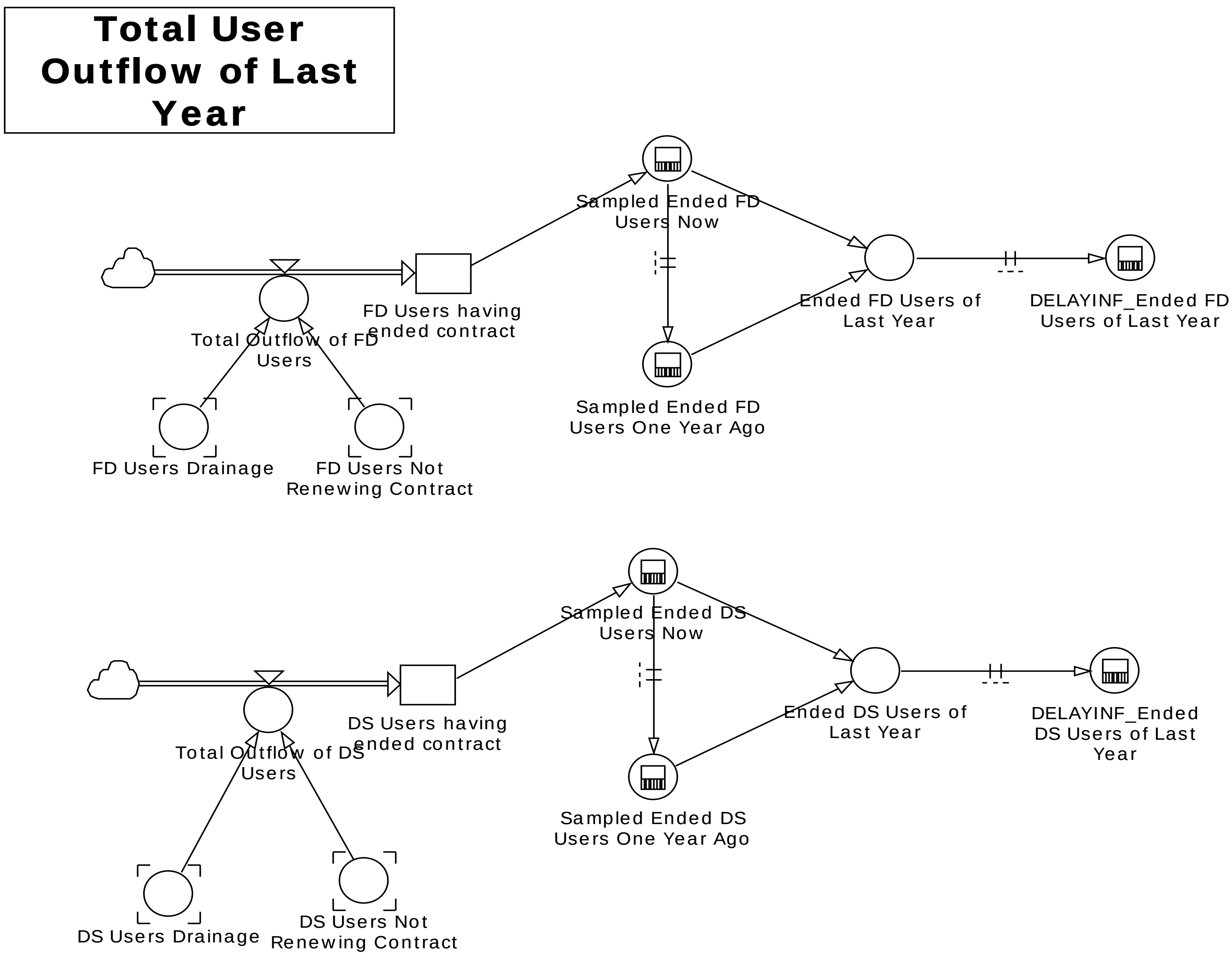


Figure 124 - Total User Outflow of Last Year

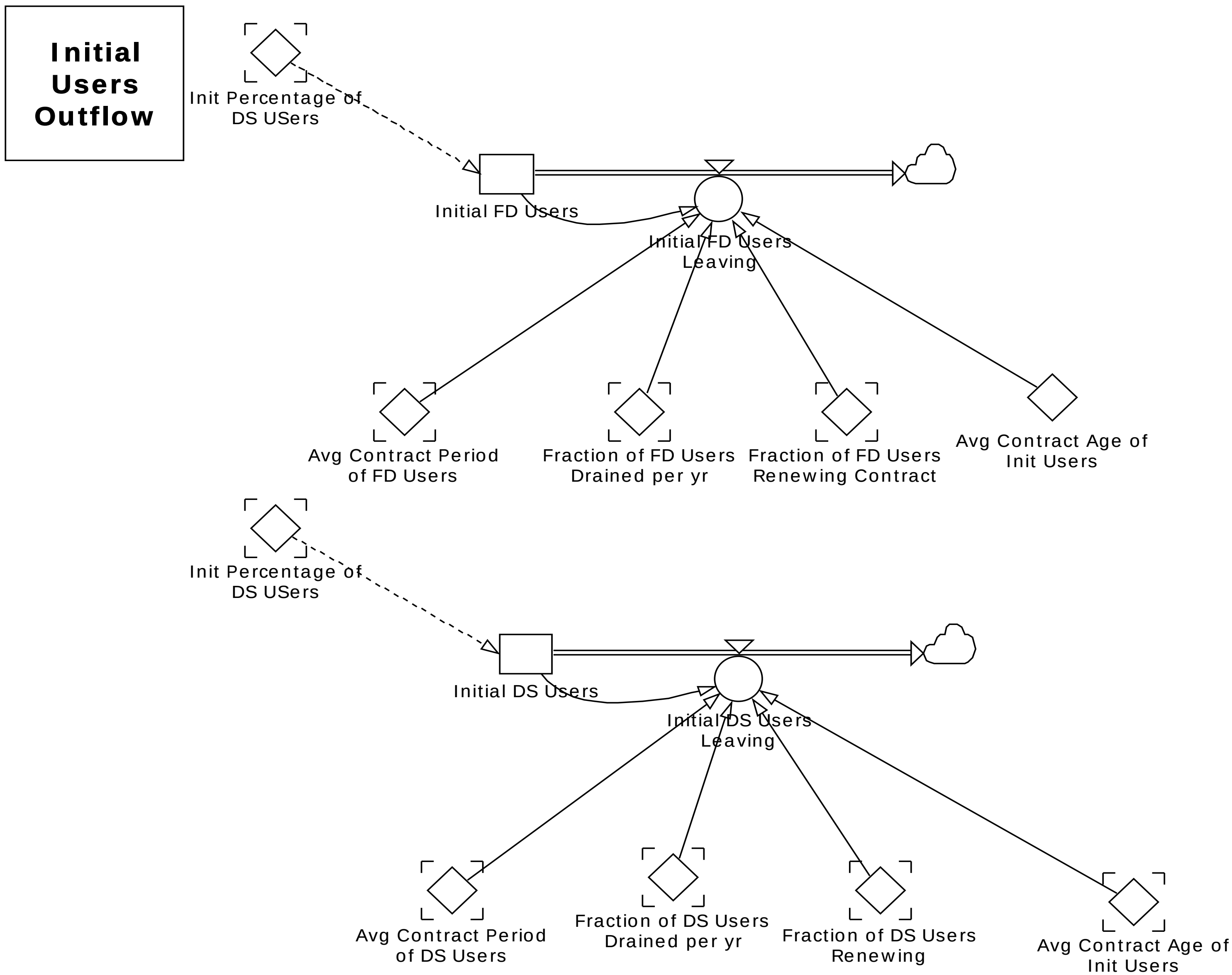


Figure 125 - Initial Users Outflow

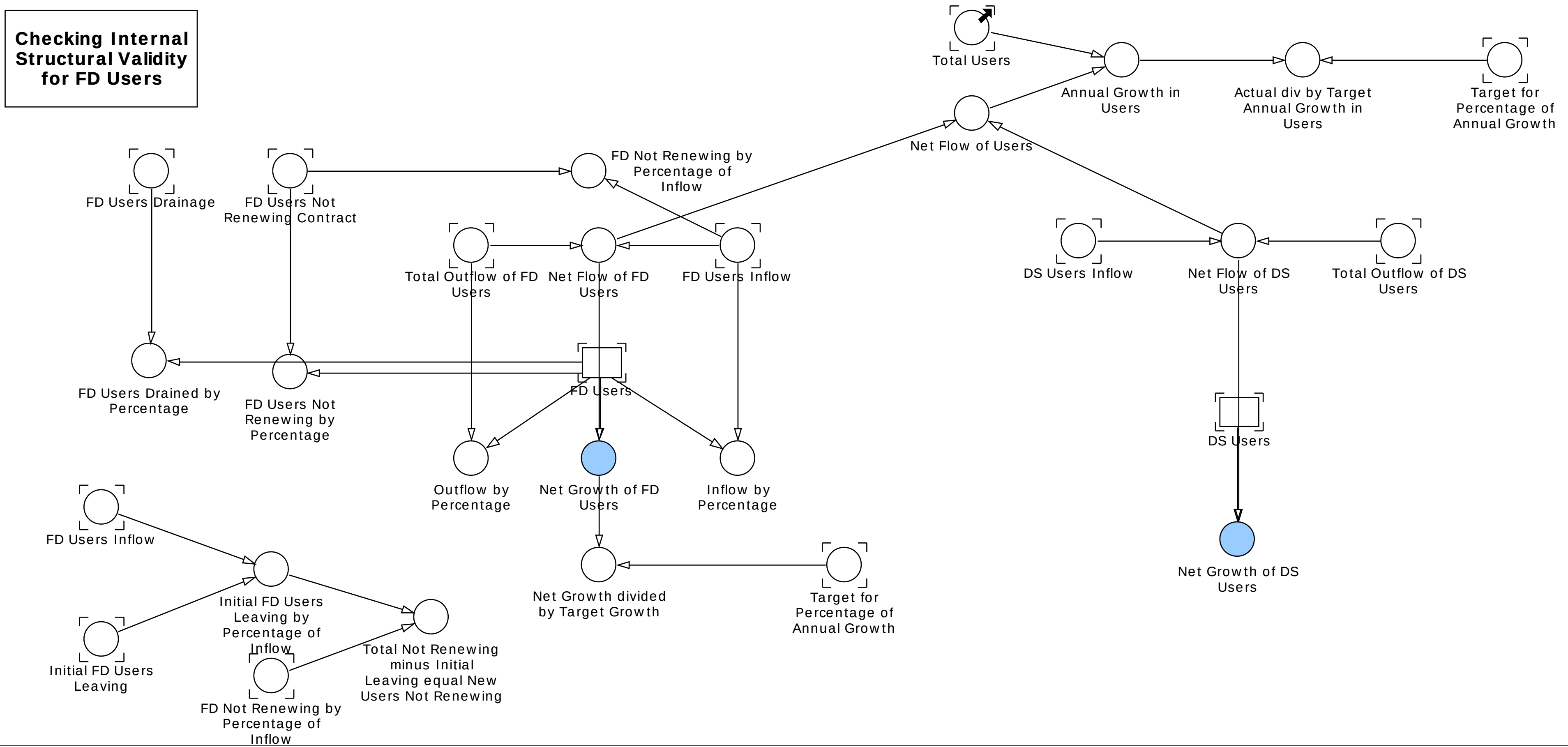
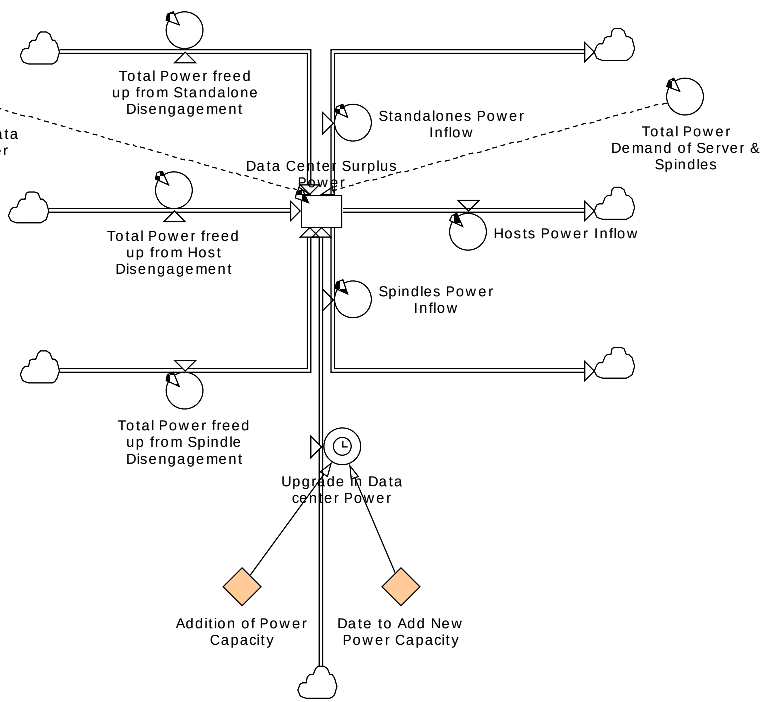
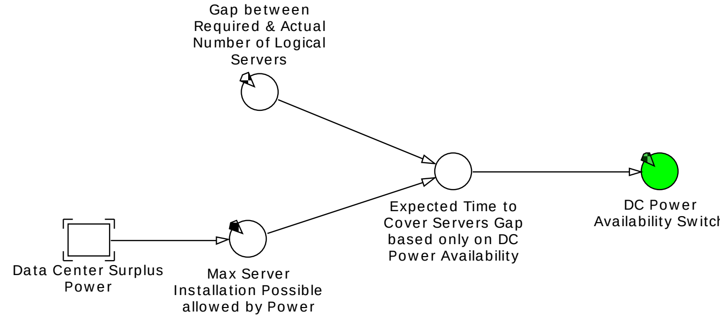
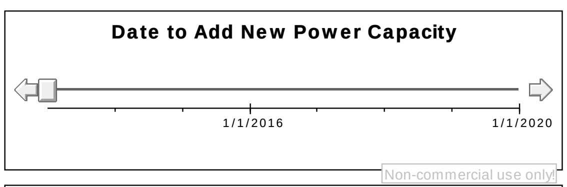
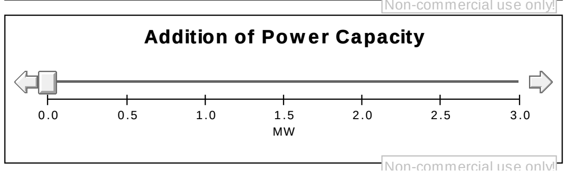


Figure 126 - Checking Internal Structural Validity for Users Inflow









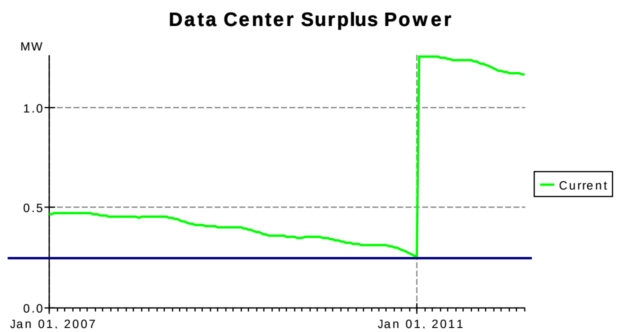


Figure 127 - The Data Center Sector

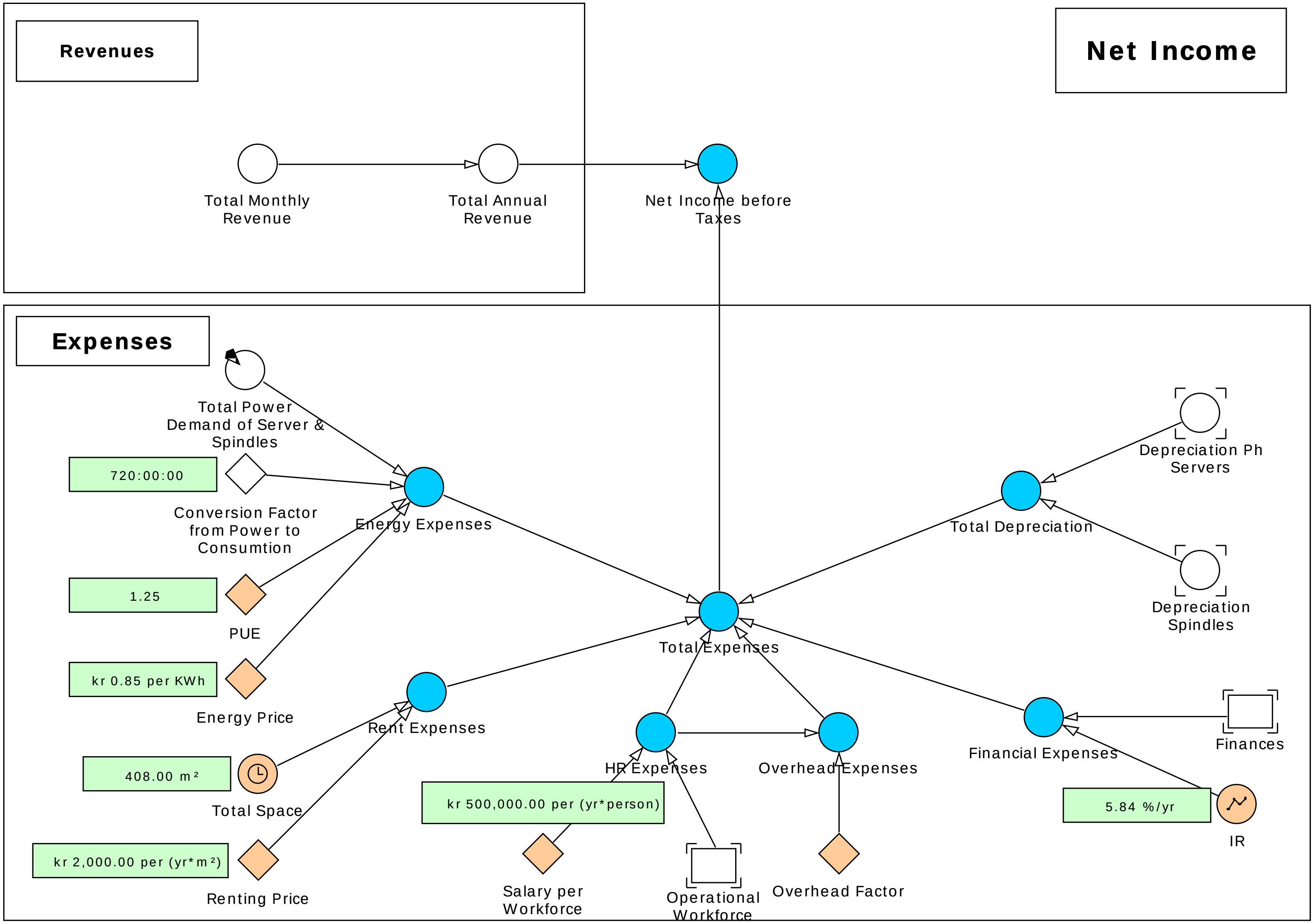


Figure 128 - Net Income

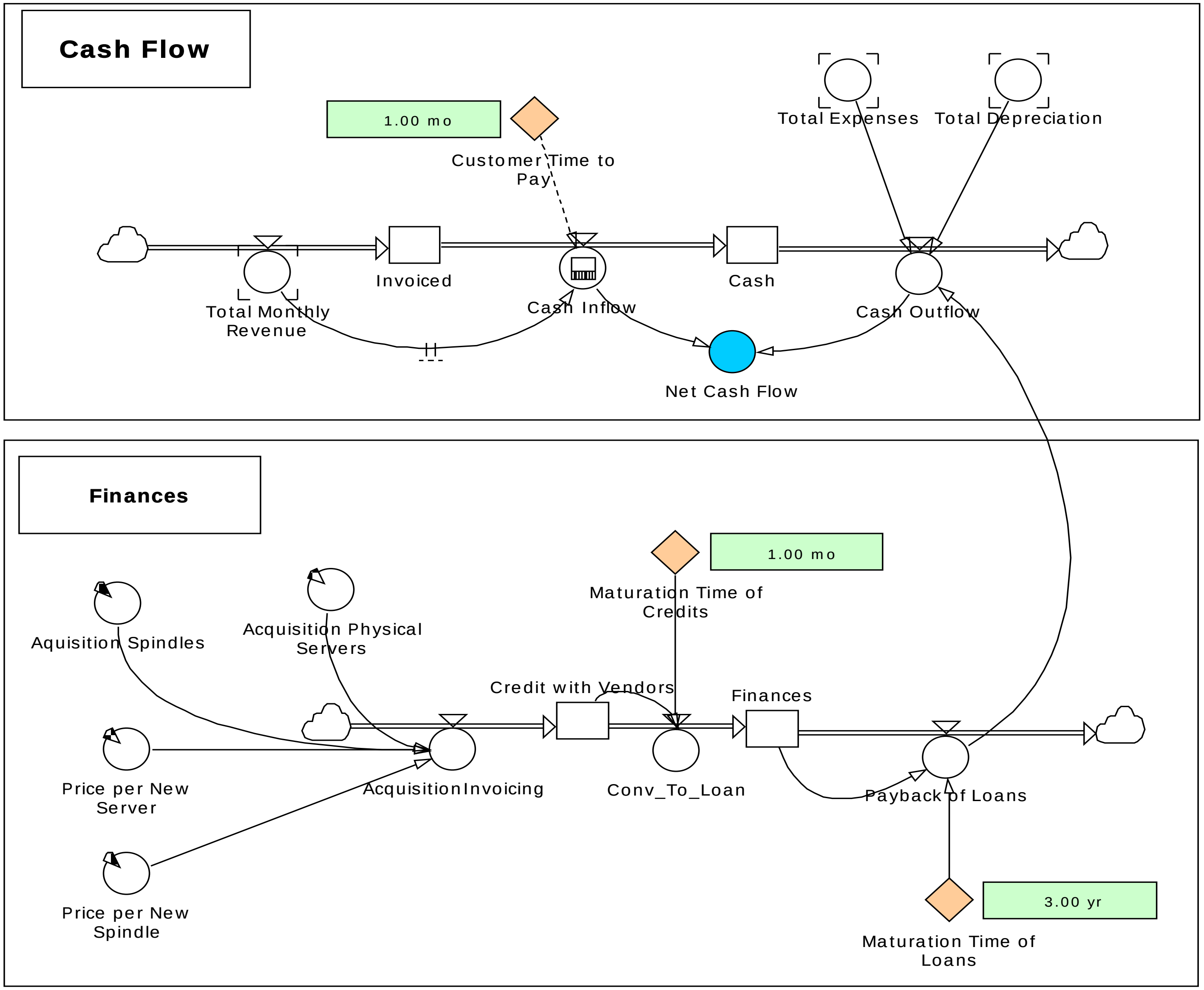


Figure 129 - Cash Flow and Finances

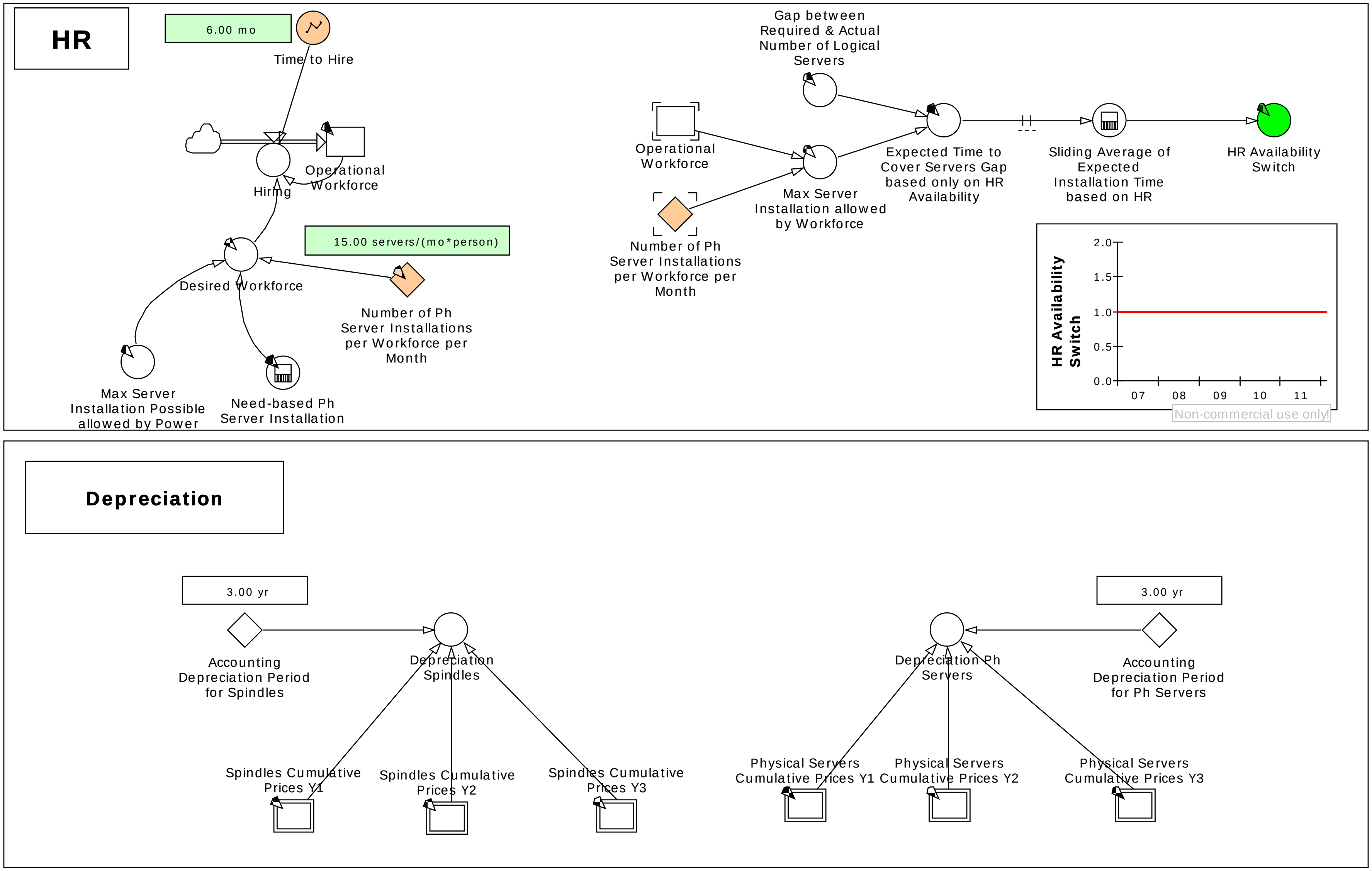


Figure 130 - HR and Equipment Depreciation