

Ville-Pekka Peltonen

Software Asset Management

Current state and use cases

Helsinki Metropolia University of Applied Sciences

Master of Business Administration

Master's Degree Programme in Business Informatics

Thesis

24.4.2015

Author(s) Title	Ville-Pekka Peltonen Software Asset Management, Current state and use cases
Number of Pages Date	30 pages 24 March 2015
Degree	Master of Business Administration
Degree Programme	Master's Degree Programme in Business Informatics
Specialisation option	
Instructor(s)	Antti Hovi, Senior Lecturer
<p>The purpose of this study was to analyse the current state of asset management at a major telecom operator in Finland in order to create competitive edge out of software asset management for the company. The aim was to use the results of the current state analysis to improve the target company's software asset management.</p> <p>The study was carried out as qualitative research with interview workshops. This was done in order to get a clear picture of what the organisation has. The participants for these work shops were carefully chosen so that the entire company was represented in the work-shops.</p> <p>The results from these workshops showed that that a lot was already done regarding the software asset management in the company but the work was isolated and needed to be organised. There was a lot of room for improvement in software asset management. The most critical issue was the lack of proper tools for handling license masses in the company. Therefore, the improvement task in the present study was to create use cases that could be utilized as a communication tool illustrating the wanted outcome from software asset management and also to explain to other parties what should be done and why to improve the performance of the company in this area.</p> <p>The author proposes that the company should start a programme to build the capabilities required to have effective software asset management in place in the company.</p>	
Keywords	Software asset management, Current State Analysis, Use Case

Contents

1	Introduction	1
2	Software Asset Management	3
2.1	Entitlement	6
2.2	Inventory	7
3	Case Company	7
4	Pre-study about present state	8
4.1	Procurement workshop	11
4.2	Purchase and Contract management workshop	11
4.3	Software installation and inventory workshop	12
4.4	Configuration management workshop	12
4.5	Conclusions from workshops	13
5	Development model	13
5.1	RACI matrix	14
5.2	Rummler-Brache Diagram	15
6	Development Project	16
6.1	Use case License enquiry	18
6.2	Use Case License Purchase	19
6.3	Extend License Support	20
6.4	License support check	21
6.5	Use rights addition	22
6.6	License compliance check	23
6.7	Incompliance	24
6.8	Software decommission	25
6.9	Server decommission	26
6.10	License decommission	27
6.11	License optimization	28
7	Conclusions	28
8	Recommendations	29
	References	30

1 Introduction

This paper is done as part of my MBA studies at Metropolia University of Applied Sciences. It illustrates improvement project that was done to the Software Asset Management (SAM) area in order to improve SAM and achieve better understanding on risks involved in this area. SAM will be introduced in the second chapter of this paper.

Project is now finalized. The scope changed from original due to the re-organization during the project. The constant organizational changes in the case company make project work quite challenging since project resources can vanish from a project literally overnight.

During the project it had two different project managers who mainly concentrated steering and handling project governance issues. The core group in the project work was I and my colleague Merja Sorsa, neither of us did not have too deep knowledge of SAM. As Merja had started working in license management team just two months earlier and my normal work was not around License management area. We were fortunate to have senior expertise in our team in the license management team to point us to the right direction in the beginning. This saved us plenty of time at the beginning of the project.

Project consisted of two phases which were quite separate activities. First step was current state analysis that was conducted as an interview of stakeholder groups that are involved in different phases in SAM. We selected workshops to be the structure for these interviews in order to have rich discussions over the topics that were in a workshop in question. By dividing the workshops under separate topics helped us to keep the focus on that particular workshop in question because people have often tendencies to extend discussion outside the agreed topic in order by explaining problems they face. Those kinds of discussions could be easily then directed to appropriate workshop to keep the discussion sharp and to get to the point.

Second phase was improvement activity concentrated on improving management tools, mainly reconciliation and contract management tool enabling SAM to work in more agile way. Furthermore one of the main goals of the whole project was to enable SAM to respond to the coming pressure from outside the organization in a form of

software audits. By a proper SAM we could mitigate the risks and minimize the costs coming from those software audits. In addition to that it was obvious also to find tools that have the capability to optimize licenses. In audits software vendors investigate in co-operation with organization that they have the right amount of licenses compared into the amount of installations they have. The paid invoice is often needed as an evidence for user right verification. Motivation for vendors is purely financial; they want to be sure that customers have paid for what they are using. For customers the motivation is also financial as you don't want to pay for something that you don't even use.

The original intention of the project was to implement a tool for SAM purposes but in the early phase of the project of the setup it was understood by the project steering group that we had to narrow the scope to be able to deliver the current state analysis about the capabilities of organization. Reasoning for this was that we had to find out how the things are in real world instead of making assumptions. The work based on assumptions usually will actually do more harm than good.

We also decided to investigate how current tools in organization could support SAM activities during the whole software lifecycle. This would prevent possible investments to yet another IT tool before we were even sure that the need could have been covered within the existing tool. Not investing to new tool means that the money can be used to something else in the organization. This intention was also reduced later to be the definition work for use-cases of the SAM tool as we encountered problems with current tools. Problems related to the information model that was not done properly when the tool was taken into use. At that time the tool was only considered to be used in a limited use and with limited information.

At the end of project we ended up with very good current state analysis and tool independent use cases for Software Asset Management. Use cases will be a real asset in the next phase of License management improvement work. Use cases will be tangible for any future projects showing particular needs for processes and tools for controlled and efficient Software Asset Management.

2 Software Asset Management

Software Asset Management (SAM) is part of IT asset management (ITAM). Gartner defines ITAM as framework and set of processes tracking and monitoring the financial, physical, licensing and contractual aspects of IT assets throughout their life cycle.

(Snyder: 2012, p11)

IT assets cover all items that are used in organization for IT that has some kind of value. Assets can be tangible or intangible. Tangible assets include things like servers, storages and networks. So they are something that can be easily identified because they are physical assets. Intangible assets are usually harder to comprehend, because they are not something that can be calculated as easily as physical things. Licenses are intangible assets. Licenses produce right to use for certain software.

There are different kind of licensing models and depending on the licensing model one license can be used for instance whether for physical server installation purposes or to unlimited installations. In example case where servers has four Central Processing Units (CPU's) the licence need for one installation is as many licenses as the server hardware has. Like in Oracle database case that would mean that four CPU server would actually need four Oracle Database Licenses for a one installation in the server in question. At the other end of spectrum for example Microsoft offers an Enterprise agreement where the agreement gives company possibility to utilize unlimited installations of agreed Microsoft products during the agreement period.

SAM is concentrating managing as name suggests Software that is used by organization. Software is always owned by its publisher License grants to user organization right to use for that particular software. Reason for introducing SAM to organization is primarily cost savings that enable organization to have competitive edge against competitor that lack such capability in their own organization. (Snyder, 2012, p1)

Organizations that have started SAM programs reports according survey, done by Gartner, that 40% of companies achieve more than 10% savings in the first year of and more than 50% see above 10% savings between years two and five. (Snyder, 2012, p1)

Majority of SAM programs reporting successful result for cost saving are totalling 92,3% of respondents. This high success rate implies that there is real benefit for com-

pany to be achieved by putting SAM at the focal point in organization especially when Software costs for IT companies can be in tens or even in hundred million euro range. This suggest that saving cost are most certainly something that must have CIO and CFO level attention so that they will be run throughout organization smoothly.(Snyder, 2012,p2)

Degree of success	Total	Level 1 Maturity	Level 2 Maturity	Level 3 Maturity	Level 4 Maturity	Level 5 Maturity
Total (number of respondents)	143	10	17	63	28	25
1. Not at all successful	2 1,4%	-	-	2 3,2%	-	-
2.	1 0,7%	-	-	1 1,6%	-	-
3.	8 5,6%	3 30%	2 11,8%	3 4,8%	-	-
4.	29 20,3%	1 10%	5 29,4%	13 20,6%	7 25%	3 12%
5.	43 30,1%	4 40%	3 17,5%	21 33,3%	8 28,6%	7 28%
6.	42 29,4%	2 20%	6 35,3%	16 25,4%	8 28,6%	10 40%
7. Extremely successful	18 12,6%	-	1 5,9%	7 11,1%	5 17,9%	5 20%

Table 1: Cost Savings from ITAM

In table 1 above is shown results from Gartner survey done during 2011 among companies that have started ITAM programs. It illustrates how good organizations perceive their own success in ITAM programs. There is clear correlation between the level of maturity and cost savings. More mature the company is more likely it is to achieve saving for the company. Maturity model is explained in details below. (Snyder 2012, p5-6)

Maturity levels are described by Gartner:

Level 1: Ad hoc. ITAM exists but roles and processes are event- and transaction-driven. In this maturity level purchasing of licenses is done by individuals on current need without for example checking that someone might have free available license on their desk.

Level 2: Repeatable. ITAM roles and processes are defined at departmental level, but are not uniform across the organization. At this stage there might be agreement in the company that assistant might be ordering the Licenses, but the recording of assets is usually still at a low level.

Level 3: Defined. ITAM roles are defined, and processes are applied consistently across the organization. ITAM data is reliable and is starting to be used to guide tactical IT decisions. At this level usually there starts to be some kind of picture how many licenses are purchased by the organization.

Level 4: Quantitatively managed. ITAM responsibility and process integration extends into financial management and other adjacent areas. ITAM reporting is used by senior executives to manage cross-IT operational costs via reporting (for example, IT budget calculations, budget variance reconciliation and total cost of ownership).

Level 5: Optimizing. There is a proactive capacity to "sense" the business and its strategy, to look forward and plan future actions, and to support IT strategies (for example, cost modelling, IT service cost alignment and long-term IT planning). ITAM data is used at the most senior levels of the organization to influence and guide IT strategies (sourced vs. outsourced, for example, or cloud vs. on-premises). This is facilitated by proactive cross-organizational engagement (with, for example, strategic sourcing, and enterprise architecture and project and portfolio management).

Gartner has two maturity models that overlap in the area of ITAM. These maturity models are directed at differing audiences (IT operations and more centralized ITAM programs), but are generally consistent in the levels they use to measure how an ITAM program matures.

Main focus in SAM is to have compliant situation, where entitlement to use software meets the actual situation of installed and used software by the organization. Status of

a company can be in three stages when it comes to reconciliation of license status. These three statuses are compliant, over-licensed and under-licensed. (Barber, 2010, p5)

Compliance means that organization is exactly right licensed, which is extremely rare. At any time it is likely that there are purchase orders processed and installations waiting for the implementation. This will lead to a small margin of error with the exemption of legacy products that are no longer purchased or deployed. Therefore A small tolerance can be applied where the situation can be considered to be compliant. (Barber, 2010, p5)

Over-licensing is situation where organization utilizes fewer licenses than it actually owns. In this situation usually licenses are placed in pool where they can be utilized later if needed. In over-licensing situation is also organization must also think if retiring licences can bring savings in form of avoided support fees, of course this kind of decision must be done in that way that possible future needs for this particular license are also gone through so that organization can also avoid possible future purchase needs if some kind of usage is foreseen for future. (Barber, 2010, p5)

Under-licensing is the most dangerous situation for organization as it is breach of licence conditions and terms. Furthermore it can jeopardize organizations relationship with licence vendor resulting hike for license prices. This result in internal audit should always trigger investigation and remediation of this situation so that the risk of losing both money and image of organization can be removed. (Barber, 2010, p5)

The Information utilized in SAM consists of entitlement data and inventory data. They both have good use in the SAM. In next chapter we will go through what is meant by entitlement and inventory.

2.1 Entitlement

The entitlement data that company has about the software licences such as contracts, license certificates, invoices, CDs, boxes and electronic license keys. Entitlement data shows and proof that organization has right to use these software's and it also describes what constraints usage might have, like geographical restrictions or possibility to only utilize licenses in certain legal entity.

It is quite common that this information is quite scattered in organization and standards don't support well SAM processes in the beginning. It is a task for SAM to influence other processes like purchase to pay (P2P) in that way the P2P will start to provide information that is needed in SAM processes later on in life cycle of software asset in organization.

2.2 Inventory

Inventory data is the data that shows to the company what kind of assets they have and what kind of configurations those assets have. The data is gathered from existing IT environment either manually or by any available inventory tool. In corporate environments the amount of data renders away the possibility for using manual methods. The changes to environment will happen more rapidly than it is possible to record the changes manually without having literally hundred people to keep records up to date.

There are many inventory tools in the markets today and finding the one that will suite your organizations needs and also can be used as evidence in the case of possible licence review is quite limited.

Next we will take a look on the case company. We will look the size of company and the impact that SAM can have on company.

3 Case Company

Company that this project is done is large multinational telecom-operator functioning in 17 countries. It has 25.000 employees and with 190 million subscriptions. Net sales of the company were 10.000 Million Euros in the year 2013.

The corporate consists on many independent companies that have different ways to handle their operations. From Group point of view aim is to have economies of scale for all where it is possible. In Software licence area this benefit comes from negotiation volumes in frame agreements between software vendors and case company. One will get better discounts when combining 17 companies needs compared to situation where each of those companies try to negotiate their own deals individually.

Also other big emerging reality is that software vendors are losing their revenue due to the down turn in global economic situation. Their solution to decreasing revenues is auditing customers in order to find possible installed software without proper licencing and then sell those licences to companies that have broken the license rules. Big software companies have said that they expect to have 30% of their revenue in future coming from this auditing stream. As IT budgets in companies are quite big spending and the software is the fastest growing segment in the IT budget. The effect from unwanted license costs can be really significant, even millions of euros.

If rough estimation of value of the licenses in scope is 450 Million Euros and yearly maintenance cost is around 20% of that value. It makes saving potential to be about 90 Million Euros in value and in yearly operative cost it would mean saving potential of 4,5 Million Euros in the year one with assumption of 10% saving. So no doubt this is worth of investigating.

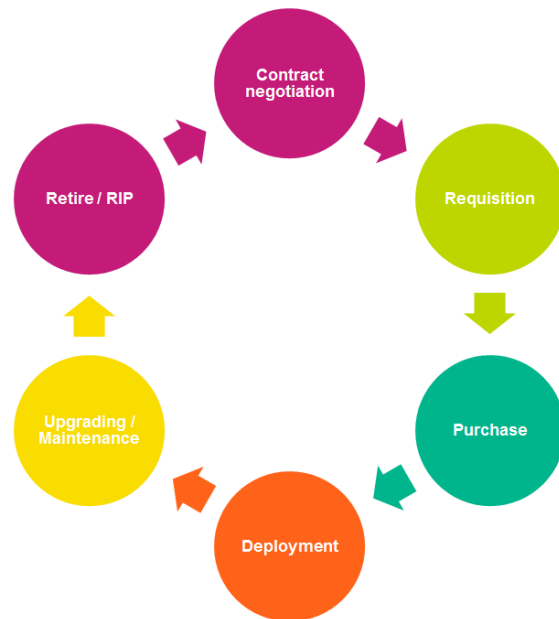
4 Pre-study about present state

When the project was planned by the project team we quickly realized that SAM task were fragmented into several units or we have so many places in the organization where SAM tasks were done. So we decided that we had to make some kind of mapping how tasks where done in different parts of the company. After considerations we decided to have workshops in all those several areas we felt had impact on SAM.

We ended up by having a five separate workshops and prepared questions to participants forehand so that they could understand better what angle we wanted to have on SAM. Since responsibility for SAM is not managed centrally we did not have any common way of working with SAM on corporate level. We divided the workshops according to the model that shows the life cycle of license in our own organization. This model can be taken as general model that shows the license lifecycle in the organization. This is illustrated in picture1 below.

License Life-cycle

- Contract negotiation
- Requisition
- Purchase
- Deployment
- Upgrading / Maintenance
- Retire / RIP



Picture 1 License life-cycle shows the steps in the SAM

First step in life cycle is Contract negotiation; this is step where the organization makes agreement with vendor. Agreement covers for example volumes to be purchased, prices and other license terms that are valid between organization and vendor.

In the requisition phase organization notices that they have a need for additional license purchases. After checking if there is no existing licenses the next step is the purchasing phase. In the case licenses exist the next step is deployment, which is explained later in this chapter.

In purchase step one does all the needed tasks to ensure that the organizations procurement process is followed and that all necessary information is recorded in all the needed systems. This means financial systems and SAM systems at a minimum level.

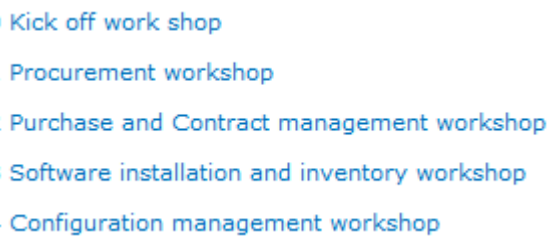
In Deployment step the actual software itself is deployed and all the required information is recorded in all the needed databases in order to make production work smoothly like a well-oiled machine.

In upgrading / Maintenance step there is checkpoint that at least every year all installation base of any particular software is assessed and then mirrored against purchases.

Intention is also to renew needed supports for software's and terminate those which are not needed anymore by the organization.

Finals step in the lifecycle of a license is the retirement of license. In this step it is made sure that all installations of the software in question are removed. Contracts against vendors for maintenance also must be terminated. Also important task is to remove these assets from financial systems.

From these lifecycle steps we identified a need for five separate workshops in order to have some kind of manageability on topics we could cover with reasonable amount of people and at the same time keep the focus strictly on the issues that should have the focus in the workshop. The workshops are shown in picture 2. There was a kick off workshop where we invited all the relevant people in order to share our view of things to all parties in effective manner. The remaining four workshops are presented along with their results in next chapter.



- Kick off work shop
- Procurement workshop
- Purchase and Contract management workshop
- Software installation and inventory workshop
- Configuration management workshop

Picture 2 workshop structure

For the workshops we identified the key persons who had the best possible knowledge of current situation from their point of view. So we had persons participating workshops who actually knew best what the reality in the company is. We decided to record all the meetings so we could listen recordings if we disagreed what the participants told us in the workshops. This proved to be good convention, we found out that already with in the research core team we sometime interpreted something that was said in different way. Recordings helped us to clear the right understanding to all team members. We also did separate observations notes during workshops trying to also capture the non-verbal messages that participants might have. Also after each work shop the core group gathered and we made conclusions about each workshop and adjusted next workshop to be better that previous one. Next we will go through all the workshops and their results.

4.1 Procurement workshop

The first workshop handled IT procurement, where there were two participants from IT procurement and one person from a financial IT system department. In the workshop IT procurement presented what role and responsibilities procurement has in Software Assets Management. Procurement is the function that negotiates all the corporate level frame agreements with vendors. Their strategy is to have global contracts negotiated in order to reduce price and also to get better terms and conditions by using economies of scale as a bargain power producing item.

We saw immediately that the negotiation phase is in a rather good shape and it creates competitive advantage to the whole group. From financial side we found out that there is place for improvements that were confirmed in the next workshop that we held and we are going to dive in to those issues in the next chapter.

4.2 Purchase and Contract management workshop

In the second workshop we had five participants. One of the participants was from the financial IT department. Three of them from production responsible for the license support. Also there was one participant with financial controller role.

There were improvement tasks that were already raised in the previous workshop and those were confirmed in this one. Issues were for instance that the lack of Software catalogue in the company, meaning that people were ordering wide variety of same software parallel not able to check already existing licenses making efficient SAM very challenging. One other issue was that the free text information to Purchase Orders (PO). There were not enough mandatory predefined fields to make the data accurate. While handling the information the purchase process produces has a high risk for errors. Using this information for any inventory usage would have to be manual work. In an enterprise situation this is very challenging since number of PO's may be in thousands.

From the workshops came three conclusions. First was that there is a clear need to use catalogue based purchase with the most used licenses in order to have data quality that in that level could be utilized in SAM. Second conclusion was that centralizing

the purchases to one unit would give a lot of benefit the company. The third conclusion was that the cost reporting from licenses could be much more accurate than it is.

4.3 Software installation and inventory workshop

In the third workshop there were largest group of participants. Reason for this is that most important tasks for software license management are done at this phase. There were five persons who are license managers. Two persons were involved from configuration management database function and three application operations persons who are responsible for installing the software in to the production.

There were four main conclusions from this workshop. First the licenses outside Group It Infra were not centralized and therefore not in control that means that they have higher risk for noncompliance. Second that sufficient inventory information is getting better due to the system development that is already in progress. Third conclusion is that the definition of software installation identification from discovery data to actual license asset information in Configuration management database must be done for each software product. Fourth conclusion was License management will prevent illegal installations in future via software discovery.

4.4 Configuration management workshop

In the final workshop there were three participants, all of them were in management role for configuration management database. In configuration management area there is actually few different databases up and running due to the historical facts and working methods that are quite different depending on who is in driver seat so to say. As a conclusion from this workshop we found out that Contract, Software's and License assets are not considered as Configuration Items in The configuration management database.

4.5 Conclusions from workshops

As a conclusion from the workshops we found out following things:

1. There is a lot of data available in different IT systems related to the software license management. But the data is not integrated between the systems. Partly the same data is updated manually and automated in different systems.
2. Information model on top level is missing e.g. what can be found from where.
3. There are functionalities (modules) in the systems which are not implemented into use.
4. The data is not structured, a lot of free text is used and several data items are optional for the end user to fill in. This cause a loss of a benefit of the possibility to build up comprehensive information view of the data in systems.
5. The overall provocative observation is, that we have not used the IT systems on the way they are meant to be used and lack of discipline in the working methods have caused the present situation. The Business Requirements for the focus area have never required detailed, accurate and integrated information.

To our disappointment we realized that we did not have as fast track to set up system as we had thought earlier. These results prohibited us to gain the quick win that we originally thought that we could use existing system as building block for future. There was too much things missing. After carefully going the situation trough in project steering group we decided to go on with Development project that we will go through next.

5 Development model

After we had built up the picture of present stage in the company we decided that we should aim to develop a common way of working in the company including processes and tools to support the way of working. Due to the strict time limit for the project we took a practical route and started with defining all use cases for toll that we would use as SAM tool in the company. This was done in order to communicate the need that tool had to fulfil in order to tackle SAM in most efficient way. After use cases were done we could then map if any of our existing tool could solve the issues that we had and we could make estimates of the costs that this kind of exercise would create in order to calculate the business case later on.

We had two possible candidate models how we could illustrate the task at hand. They were Rummler-Brache diagram also called swim lane diagram or RACI matrix. Next we

will look at both of those models and reasoning why we ended up with swim lane diagram. Both models are used frequently in the case company and because they are most familiar for employees in the company limited the models that we chose from to these two models.

5.1 RACI matrix

RACI matrix is quite common tool in project management to describe and track responsibilities on tasks that needs to be executed in the project, process or in company in general the name of this matrix come from Responsible, Accountable, Consulted and Informed. Responsible is the one who will do the work itself. Accountable is the responsible in case anything goes wrong in other words the person whose head will roll if things end up failing. Consulted is someone who has more knowledge on task at hand and can give advises if any problems occur. Informed party is someone who is depended for example on result or progress of task at hand.

In RACI matrix all tasks are listed in excel shape format and for all of them there must be at least Responsible and Accountable party. The benefit of this model is that it clearly shows who is responsible of doing what. This prevents quite efficiently the hazard of assumption, that the task was someone else problem and it makes sure that all tasks have someone who is responsible and someone who is accountable for task. Below is example of RACI matrix just to illustrate how matrix looks like.

Task	Responsible	Accountable	Consulted	Informed
Plan project	Project Manager	Sponsor	Subject matter experts	Stakeholders
Plan requirements	Business analyst	Project manager	Subject matter experts	Sponsor
Analyse requirements	Business analyst	Business analyst	Subject matter experts	Project manager
Testing	Quality Assurance	Project manager	Business analyst, Developers	Sponsor, Team

Table 2 example of RACI matrix

RACI matrix itself is not that important when it is finished the making of it is more significant because the building of matrix actually requires a lot of talk and once matrix is ready all parties that have tasks in the current project has clear vision what is expected from them enabling all parties to aim towards common goal. (Best Project Management Software Reviews 2014, accessed 13.11.2014)

Weakness of this model is that it is not very illustrative and this might possibly create delay to development. Therefore we picked also other model we wanted to compare and choose to use for description. That model is introduced next.

5.2 Rummler-Brache Diagram

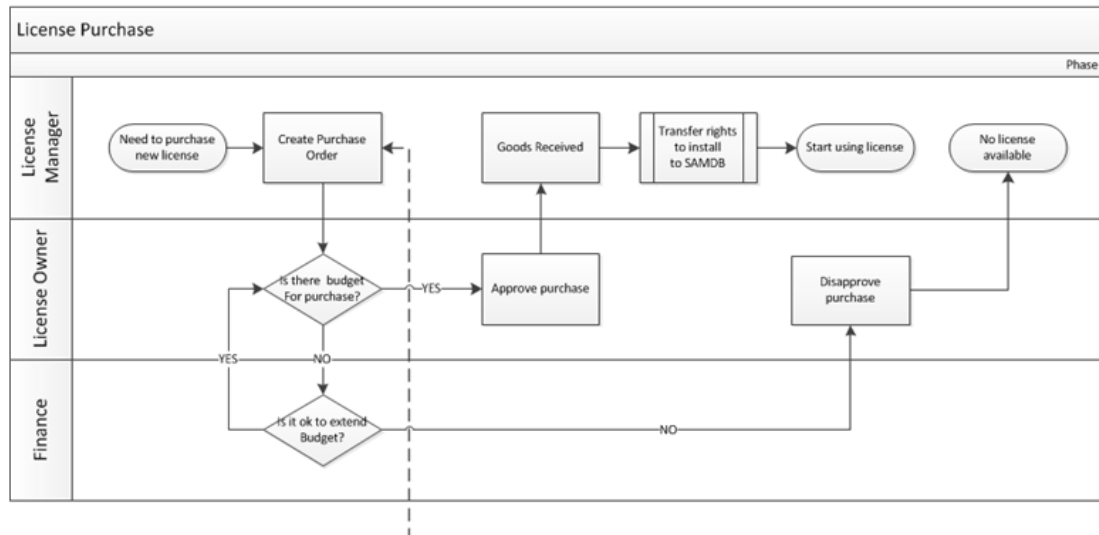
The second model that is used to describe the use case it is classical swim lane diagram also known as Rummler-Brache Diagram. It was proposed by them in their book *Improving Processes* (1990) it allows one to follow Process flow between teams, departments and even between different processes.

Like other process diagramming techniques, with the Rummler-Brache method, you map processes linearly as a series of tasks across the page. Lines and arrows between tasks represent the flow of information, goods or work-in-progress, and also represent changes in responsibility.

The difference and the genius part of Rummler-Brache is the use of "swim lanes", horizontal rows. Think of a swim meet where each competitor has own lane to swim: In this diagramming method, each "swim lane" may belong to an individual, a team, a department, or any other organizational unit you choose.

These Diagrams are great tool for describing not only process flow but also the responsibility behind them giving out the high risks between handovers to other teams and departments included in particular process flow.

As describing method it also is useful as you can use ready described process as sub process in the new task and just point to that existing process reduces the complexity and change of re-inventing the wheel situation. In this situation you try to resolve something that is already resolved, this is problem since a problem usually has multitude ways that it can be resolved. But in process driven situation key for success is that output is always in same format so that the next part in process flow always get same kind of input for same kind of problem. That makes sure that they can concentrate on solving the underlying cause and not try to guess what the input really means this time. Below in picture 3 you can see an example of a Swim lane diagram. (Mind Tools, accessed 13.11.2014)



Picture 3 Example of a swim lane diagram

6 Development Project

We ended up using swim lane model since we saw that we needed tool for future discussion for bringing people on board what we wanted to achieve as end result for users. The proof if something works is in the results that the model can deliver to people that needs to be convinced that this work is necessary for company and worthwhile doing.

We discovered 11 different use cases that we described in that way that they were applicable despite what tool we would later select. This would also ensure that we don't try to fix a problem by just introducing yet another tool.

Use cases are as listed below:

1. License enquiry
2. License purchase
3. Extend license support
4. License Support check
5. Use rights addition
6. License compliance check
7. Incompliance
8. Software decommission
9. Server decommission

10. License decommission

11. License optimization

We identified total of four roles that functions as swim lanes in our diagrams. We chose role as dividing factor because we wanted model be in depended from organizations so it will stand against renewal of organizations that are quite frequent all over. Also we saw that in this model is more guidelines how things should be handled opposed to building huge centralized “office” or other function. More beneficial for the organization would be that individuals who are working around SAM, would work with same kind of processes and procedures in order to grasp the global view on Software Assets in future.

Roles that were identified were; License User, License Manager, License Owner, Finance and CESAM. License User is role of person who is looking after system that is using licensed software. Usually these persons are engineers or other technical personnel in organization.

License manager is role for person who is looking after particular software or vendors licenses in the company. License manager is usually responsible for purchasing, allocating and retiring licenses. He or She also assists IT Procurement in the contract negotiations between organization and vendor.

The License Owner is role for the person who owns the budget for licenses. Usually this role goes hand in hand with role of ownership of IT systems that utilize software, but it is not mandatory demand.

The finance is role that takes care of receiving invoices and taking care that they are correct and put in to the appropriate cost center in order to keep track of financials in the organization. This is also important for profit calculations as licenses are also part of products cost structure.

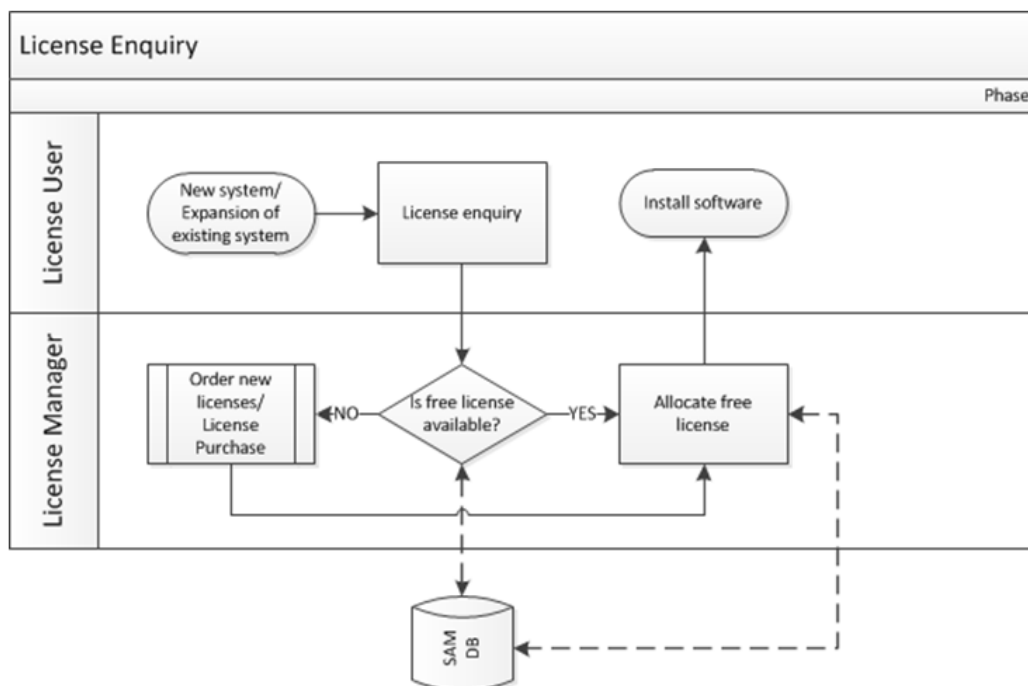
CESAM or Central Software Asset Management is Corporate Group function that owns and develops Organizations SAM model. Their role is also to function as party that will handle software audit requests that vendors may have toward organization.

6.1 Use case License enquiry

First use case is License enquiry shown in picture 3 below. This is the most simple use case in SAM. In this case there are two roles involved License User and License Manager.

It describes the situation that License User identifies need to use software that needs license. It can be triggered by either new system or expansion of existing system. In both cases License User creates request to License Manager who checks from SAM tool if there is any available Use-rights that can be allocated to this particular need. If there are available use-rights then the answer is yes and the needed licenses are allocated to this new or expanded system.

Otherwise it will trigger sub-process called order new license / License Purchase. And after Purchase of licenses is complete the allocation of license is done. After these steps License User can install and start to use software that he needed to have.



Picture 4 License Enquiry use case

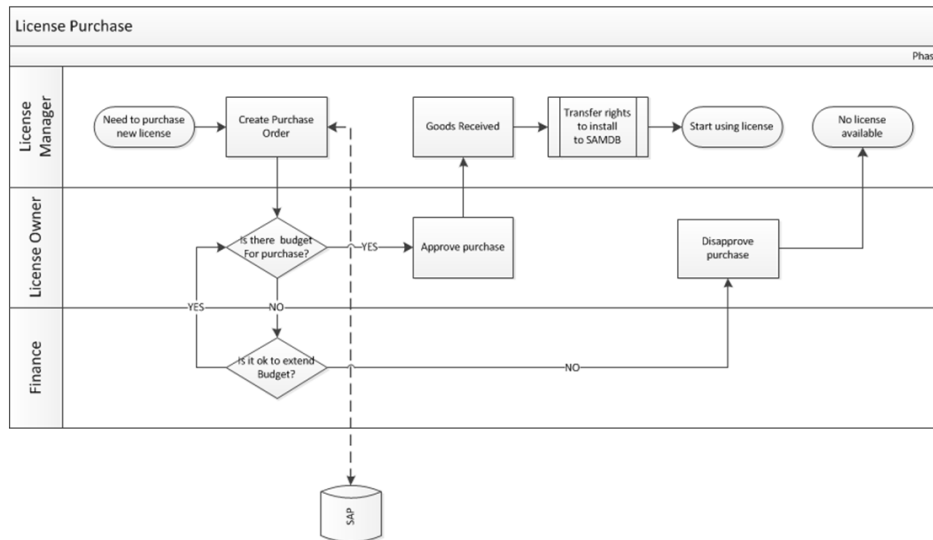
The other use case we will go through is the sub process introduced in License enquiry process flow Order New license / License Purchase.

6.2 Use Case License Purchase

This Use case is triggered by need for purchasing new licenses to be used by the organization. Purchases are done against pre-negotiated contracts with vendors. License manager does the Purchase order in to the financial system with all the needed information in order to have good knowledge on what was purchased.

After order is done License Owner checks if He or She has budget that can be used to purchase this license or licenses. If the budget is there then License Owner will simply approve the Purchase Order. In the case that the budget is lacking funds for this purchase it is his/hers responsibility to extend the budget. If the budget cannot be extended the License Owner have to disapprove the Purchase Order. Disapproving means that needed license cannot be delivered. This means that License manager have to give really clear message to the License User that they should not in any case install the software. This would cause the organization breach the agreement with vendor. The extension of a budget should be really rare thing to happen as organization should have good forecasting in place that can predict this kind of need.

When License Owner approves the Purchase Order financial systems send the order to vendor. Vendor acknowledges the order and sends the goods ordered. Usually nothing tangible is today sent to customer besides the invoice. License Manager confirms the goods received in the financial system. This will activate the investment in financial system and in book keeping. Then License Manager will record obtained License or Licenses in to the SAM tool to License pool that represents this particular software. After all this it can be allocated for any particular use.



Picture 5 Licence Purchase use case

6.3 Extend License Support

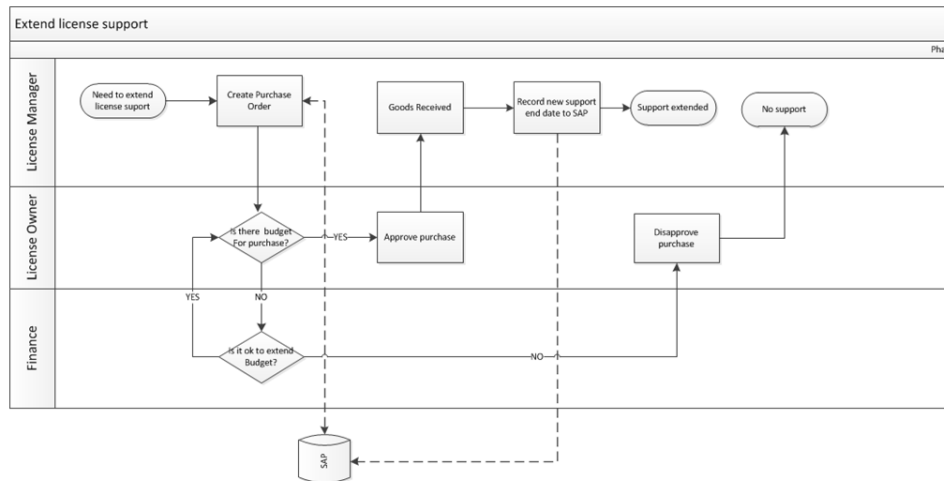
This use case is triggered once a year when it is time to make decision if License support will be extended. This decision has financial impact on company; usually the fees are around 20% of purchase price of license so it has an impact of Total cost of ownership of Licenses. One does not want to pay support on licenses that are not in use or will not be used in future.

Use case starts with need to renew the support for licenses in question. The license manager will create Purchase order in to the financial system for right licenses and with proper amount of money. After that License owner makes sure that he or she has money reserved for this support in the budget. If that allocation does not exist then it is the responsibility of license owner to secure that it will be added to budget.

When in most cases the money exists already in budgets the license owner simply approves the purchase order and then License Manager Makes good received for this support extension. Then License manager also updates the new end date for this renewed license support in order it to be triggered again when this contract term is about to be end.

If owner cannot for some reason secure the money then purchase order will be rejected and the licenses end up without support. This might mean different things depend-

ing on contract terms. In some cases it means that company will not get technical support if something is wrong with software. In other cases it might mean that right to use for software might be removed. For this reason in case support is removed it must be gone through very carefully what this decisions impact will be.

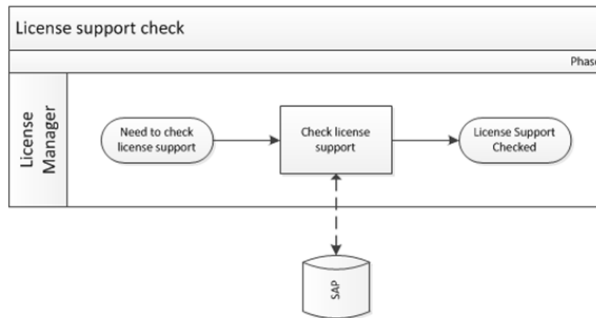


Picture 6 Extend License support use case

6.4 License support check

Next use case is very simple but it is needed by License manager in order to check where his/hers particular license stands regarding licence support status. In normal situation all licenses should have support enabled. But in some cases support is removed in order to optimize costs, one example is when system will be removed in near future and the usage of system is very low and the support cost would be too high taking in to the count the risk to the business if there would be something wrong in the system in question.

The use case is triggered by License Manager. He or She will check if the licenses in question have support agreement and that it has been paid according the agreement. This can be verified from financial system because it is kept up to date by License support extend use case. As a result from this use case License Manager has knowledge whether licenses have or don't have support.



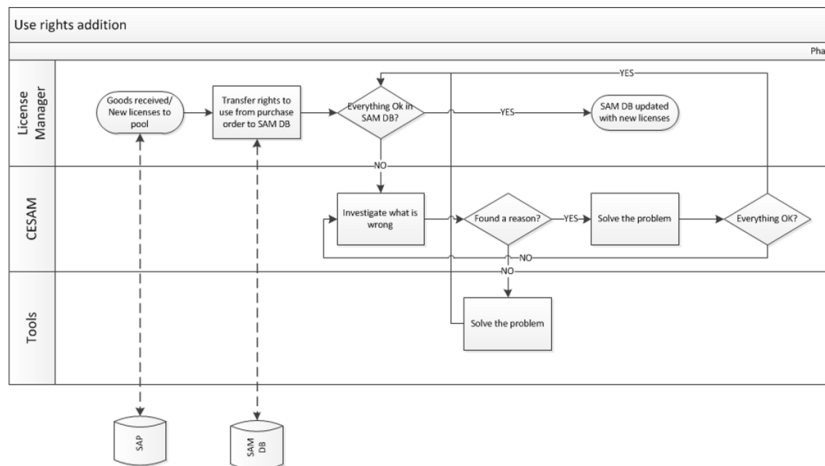
Picture 7 Licensee support check use case

6.5 Use rights addition

As licenses produces right to use for software it is needed that when licenses are purchased for the company. This use case is triggered as sub use case by License purchase use case in most of cases, we identified that also that use rights addition might be used without purchase for example in merger or acquisitions so we ended up describing it as its own use case.

It starts with need to extend or create license pool in the company with new licenses. As described earlier licenses have different mechanisms to produce right to use. This is the skills that license manager brings to the use case, License Manager creates rules what kind of right to use each license produces and then adds the proper amount of licenses in to the Software Asset Management system. After adding Licenses Manager checks that everything is ok in the system. If everything is ok then new licenses and their entitlement for usage is updated and everything is as it should be.

In the case that there is something wrong with the system License Manager inform Centralised Software Asset Management (CESAM) that there is something wrong in the system or in the way licenses have been added to the system. CESAM have more deep understanding about the SAM system so they will be the second line of support that have the needed resources to tackle harder cases that might occur in the SAM system and it is their responsibility to find the solution to the problem at hand and do the fix for it.

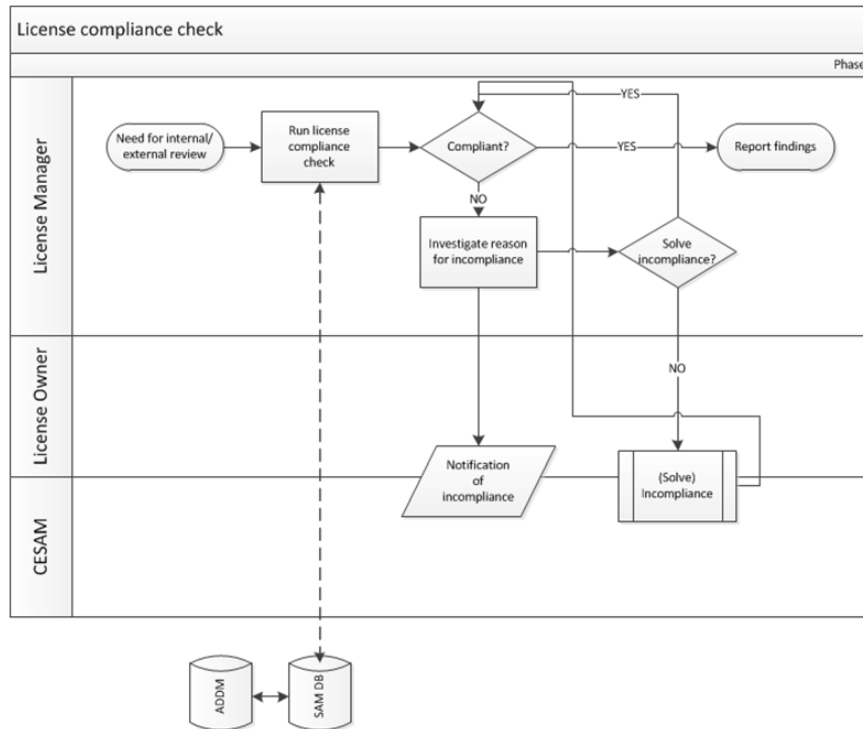


Picture 8 Use rights addition use case

6.6 License compliance check

License compliance use case is triggered by review need that everything is good with any particular license or vendor. This can be triggered by internal needs like periodical internal check-up that everything is as it should be or by external request like license review by vendor in question.

As inventory data of installations is automated in CMDB and the other use cases for license management ensure that the data is up to date it is quite easy to make this comparison. License compliance check is done and according the result is made decision by system that either company is compliant or not as the result of this check. In case company is compliant results are presented as such. If there is situation of incompliance it must be fixed that is presented as next use case. In the case of incompliance CESAM and License Owner are informed about the situation so they are aware of this situation and can resolve incompliance in that case that License Manager is not able to resolve it. But incompliance must be always resolved one way or other so that company is not in incompliance state. How this can be done is shown in next use case Incompliance.

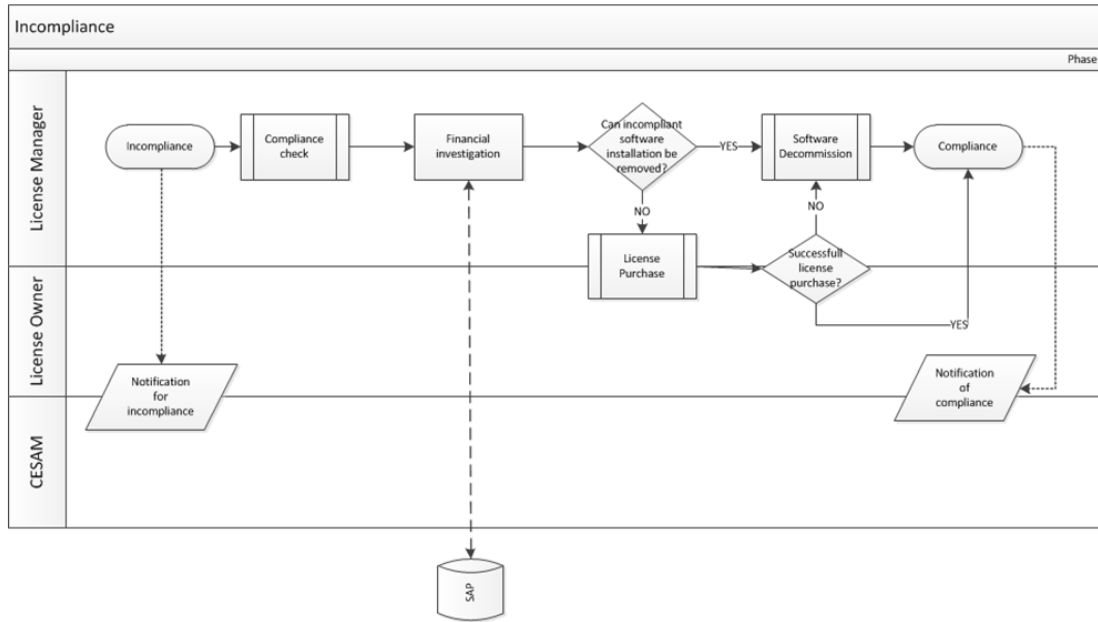


Picture 9 License compliance check use case

6.7 Incompliance

In the case on incompliance the situation is simple company has too many installations compared to the rights to use produced by licenses that company has. There is two ways to correct this situation, either purchase more licenses or get rid of installations that are not needed. When one or both of these correcting actions are done as result there should be always compliance. Software decommission use case is introduced later.

Licence manager is responsible for making necessary Purchasing actions if needed according the purchase use case. CESAM and License Owner are kept in loop when incompliance cases are resolved because they have the need to be informed about these cases in the case that for example vendor wants to have review with company so there will not be any unpleasant surprises that might harm the image of company.

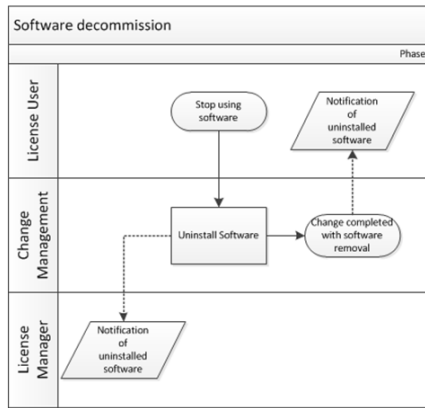


Picture 10 Incompliance use case

6.8 Software decommission

Software decommission will be triggered when need for software or system will end. For example new financial system would mean that eventually old existing system will be shut down and all licenses that it used to use should be removed and made sure that also all installations must be removed also so that usage is really ended. This makes it sure that there cannot be any inconvenient situation where vendor wants to have licenses for software that is not in use although it might be installed on servers.

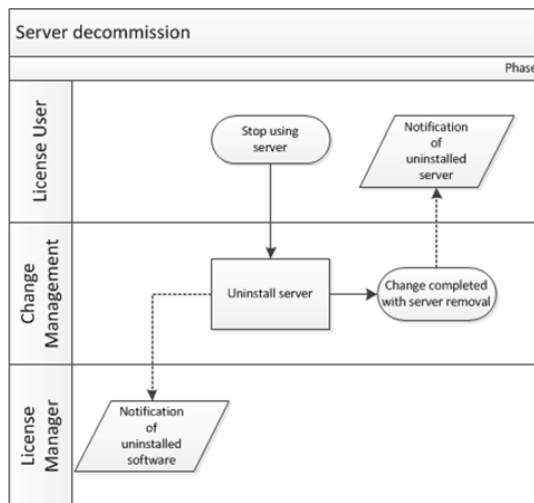
License user knows best when need for licence usage ends, therefore user makes the change request for change Manager that coordinates removal of software from server. Operational unit does the uninstallation for software and Change Manager will remove the installation from CMDB freeing up the license for other use. Change manager will also send notification to License Manager about the change so License Manager can follow what is happening inside that license pool.



Picture 11 Software decommission use case

6.9 Server decommission

Server decommission use case is very similar to software decommission use case. The only significant difference why we wanted to describe it as separate use case is that usually server decommissioning is handled and seen as separate process in the case company. Also in server decommissioning there is usually more than one software where the installation is removed, so it could be seen in that way that within server decommissioning use case the software decommission case is run multiple times.

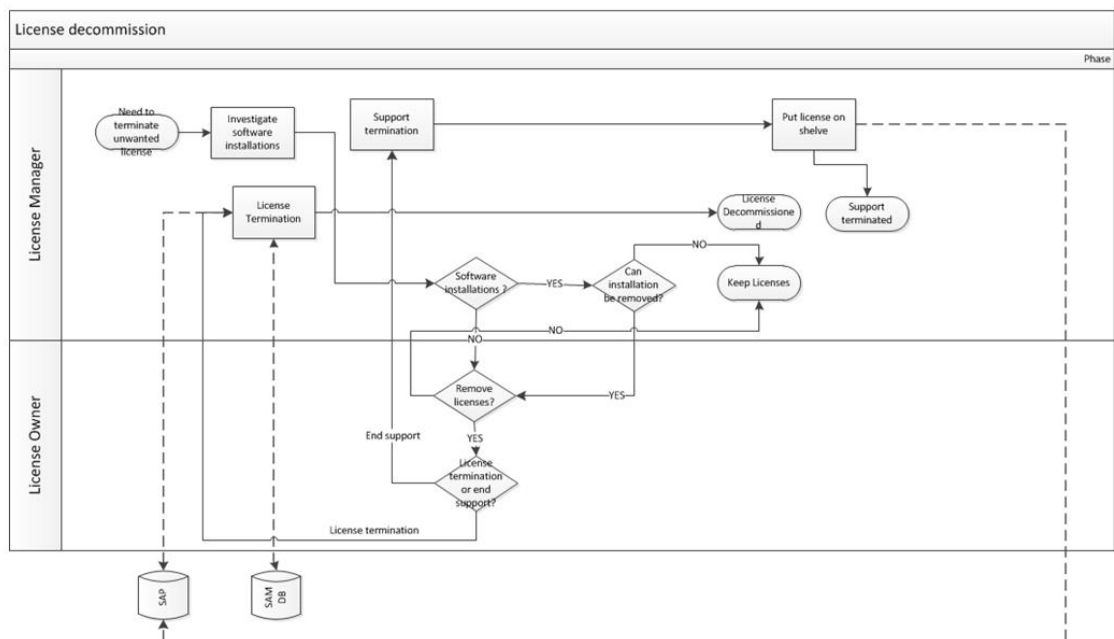


Picture 12 Server decommission use case

6.10 License decommission

License decommission use case is triggered for example if company decides that they will get rid of some software or system. In that case License Manager will investigate if there are any installations of software in question in the case company. If there is no installation then License owner can decide removing of licenses. If there is installations of software they must be removed before licenses can be decommissioned and if there is still real need for software the licence termination cannot be done because this would result in compliance state for company and that is not acceptable.

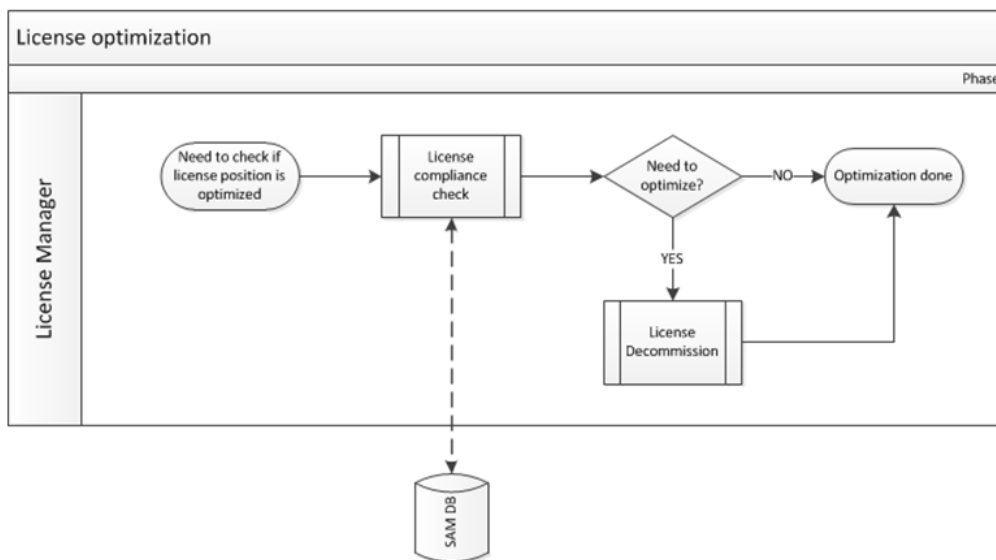
In some cases it makes sense to recycle the licenses inside the company for some other use then the licenses are kept for further use. If Licenses can be removed there is still decision to be made on will the licenses be terminated or should the licenses just be put on the shelf. When shelving licenses usually the support for license is terminated, when activating licenses vendor usually request an activation fee for license and also require that new support contract for licenses is ordered. Therefore sometimes make more sense to terminate whole license in order to avoid any future liabilities to that old license.



Picture 13 License decommission use case

6.11 License optimization

License optimization is most advanced use case from the maturity point of view for SAM. It is a proactive use case where License manager utilizes more advanced features of Licence Management tools offer today. It can recommend alternative licensing models for installations that company has in order to minimize the costs that the installations create to the company. License Manager can optimize the costs the license create even when the vendors change their licencing models as they time to time do in order to maximize their own profits.



Picture 14 License optimization use case

7 Conclusions

This paper illustrated project that was run in case organization. It aimed to get current state analysis on the Software Asset Management. Then based on learnings from this analysis the project made use cases that could be used for SAM tool. They were done in that way that they could be used regardless which tool would be chosen or used later by organization.

Project was successful by delivering the current state analysis and use cases for SAM tool. Use cases are done in a way that they are not depended on tool itself, so they can be also utilized in case that in future we will look at other tools than those we already have in the case company.

The model that was shown above was done as the result of workshops that showed us the current state of software asset management. Originally the model had only seven use cases, but after reviewing it with whole CESAM group we ended up extending its present 11 cases so that we could cover the whole lifecycle of licence management.

Model was also presented to all who participated in to the workshops in order to get their view on the model. Some minor adjustments were made as result of those reviews. Model was approved to be the blue print for future development by Director of CESAM.

As the most important finding from this project was the clear need for improving processes, roles and responsibilities in the case company. Even there was found plenty of good things and people still there is very good value in pursuing the savings and efficiency by improving SAM activities.

8 Recommendations

My personal recommendation is that there should be a larger SAM improvement action in the company. As the saving potential suggests that it is sensible thing to do as long as the investment costs are kept in control. And improvement in this field also could be seen as asset in negotiations of future software agreements. Because the case company is more trust worthy for vendors than without these improvements in internal processes and tools. Further development would also mean faster responses to License reviews saving time, money and efforts of all parties involved.

References

Snyder, William R. (2010), *Identifying the Three Key Reasons for Tracking IT Assets, and Why IT Asset Managers Should Care*, Gartner G00208090

Snyder, William R., (2012), *Gartner Survey Shows How to Save Money with ITAM*, Gartner G00225114

Barber V, et al, (2010) *Best Practice for Software Asset Management: Use the Purchase-to-Pay Process to Capture Entitlement Data*, Gartner G00206557

Barber Victoria, (2010), *Best Practice for software Asset Management: Identify and Locate Entitlement Data*, Gartner G00174730

Barber V, et al, (2010), *Best Practice for Software Asset Management: Software inventory; Locating and Identifying Installed Software*, Gartner G002877

Best Project Management Software Reviews 2014, Understanding Responsibility Assignment Matrix (RACI Matrix), and [WWW document]

<http://project-management.com/understanding-responsibility-assignment-matrix-raci-matrix/>

Mindtools Ltd, Swim Lane Diagrams, [WWW document]

http://www.mindtools.com/pages/article/newTMC_89.htm , 13.11.2014