



Recommended methodology for managing Software Licenses using LAMLM

Engineering software Licenses are very expensive, should be more carefully managed. This requires access to a lot of information associated with Licenses and the Vendors that offer them. We should first define "License". A license is the "right to use" a certain software application or tool. "Right to use" can be perpetual or term based (Subscription). "Right to use" also comes with some restriction- LAN, WAN, node-locked, User name,.... There are costs, the PO, expiration dates, and there is almost always a License Agreement, signed by an officer of your company, which details terms and conditions for the "right to use". We should also define "license key", which is typically a file containing a list of the 'features' which enable the products that you buy. The license file will also contain a Vendor daemon, the authorized license server, and port number. Each 'feature' on the license key will have a quantity, an "expiration date" and a version number. A combination of features makes up a product. The application package knows what features are needed to run that application. The User typically has no idea, or even cares what features are being used. The User however, will not be able to access or use that feature/product beyond the quantity purchased, or the expiration date on the license key. It is important to note that you buy "Products", but monitor usage of "features". Sometimes the feature name can be easily associated with the product, but many times it has no clear association at all. And usage data of many features is totally irrelevant and unnecessary.

TeamEDA has the opinion that it is not necessary to monitor every feature's usage. That can lead to confusion and information overload. We recommend you monitor only closely associated features to the product name, and that may only be one or two features per product.

It is important now to understand the distinction between asset management and usage metrics. Asset management is tracking License inventory at the product level, how many, where they are located, how much did it cost, what is the expiration date, etc. Usage metrics is tracking utilization (of typically shared licenses), at the feature level. The TeamEDA solution is called LAMLM. With LAM you can track asset inventory at the product level, and all related Vendor and License information. With LM you track usage at the feature level, monitor license servers, daemons, etc. The two components are integrated. That is, you can load product License information into LAM, point to the appropriate license file (or port@host), select the features you want to monitor, and that information is sent to LM. LM then monitors that server, daemon, and feature usage and sends the information back to LAM. An important decision to be made is whether you want to start with License inventory (LAM) and let LAM configure LM, or setup LM independent of LAM and track every feature on every license file. It is possible to use LAM without concern of license files, servers, daemons, or feature usage. It is also possible to use LM independently for monitoring License servers, daemons and all features usage.

TeamEDA recommends using LAM to track all product information and point to the appropriate license key for each License asset. A line item in LAM could be a specific product or a "bundle of products". In either case, once the associated license key is attached, pick a few features in LAM. LM will then monitor the usage for those features and report usage back to LAM at the product level.

Some clarification in terminology is needed:

An Application may also be referred to as a Tool or Product. A License is the right to use that tool/product. A license can also be called a "Seat". It is common to say "we own 5 licenses of product x" or "we own 5 seats of product x".

A Bundle is a package of Applications sold by the Vendor. A Bundle can be very specific or very broad. A Bundle can be 2 of Product A, 4 of Product B, etc. Or a Bundle can be based on "token" use. With "tokens", each product in the Bundle requires a certain number of "tokens". When "tokens" are all used up, no further product can be checked-out.

A Vendor is usually the one who created the software, and usually the one you buy from. However, the Vendor may be a Reseller, VAR, or Distributor, one who represents the maker of the software. Legal usage obligation is to the creator of the software. Legal purchase obligation is to the organization that you bought the product from. It is important to distinguish between the two and track related agreements.

Shared licenses are also called networked licenses, or concurrent-use licenses, or FLEXlm licenses,.... And are licensed as either LAN (Local Area Network) or WAN (Wide Area Network). LAN typically refers to a specific "site". WAN typically is available across sites, sometimes worldwide. Both require network access to the license server.

A Vendor Daemon is used to monitor check-out/check-in of the shared license. It is a small software program provided by the Vendor which runs on the License server, at the assigned port number. Typically features are associated with a daemon. Most usage reporting tools report features by daemon, or port@host ('host' meaning License server). To the manager or casual User, this means nothing. Port number? License server name? Daemon? These names can be cryptic. The same daemon name could be running on a few different license servers, so understanding which features of which product are being monitored can sometimes be very confusing.

A "Tag" name is a unique identifier used to report feature utilization, by Tool name, by Site, by Vendor, or some combination thereof. So instead of reporting usage of feature X at 3550@emu, one could report usage of feature X at Vendor_Site, or Vendor_WAN, or Tool_Site, making the report more useful. Of course nomenclature would have to be decided up front.

TeamEDA recommends the use of unique Tag names to easily identify specific Tool usage by Site and/or Vendor name and/or Restriction. Examples: MATLAB_SITE A, or MathWorks_WAN, or whatever you want. Tracking usage by daemon name or port@host is not recommended.

"Denials" occur when a shared license User attempts to get a tool license, but the license server daemon refuses, most often because all licenses are currently checked out. As an example, suppose you purchased 4 shared licenses of a certain tool, 2 licenses on server A, and 2 licenses on server B. Say that all server A licenses are checked, but one is available on server B. The User may go to server A and get denied, but does get a license from server B. Server A records a "denial". This is what is called a 'false denial'. Other situations may report false denials, so a good denial reporting system must filter out false denials.

“Portals of interest” is simply a unique URL to specific page in feature reporting, saved as a ‘Favorite’. For example, let’s say an engineer or manager wants to view “MATLAB Current Check Status” at any time. The URL to that page becomes a “Favorite” portal to view current MATLAB status. The Favorite name can be changed to something easy to recognize.

Standalone (node-locked licenses) are also called User-locked, or individual licenses, typically tied to a single host computer, and to that IP, or sometimes secured with a hardware key plugged into a serial or USB port. Node-locked licenses may be inexpensive but can add up if there are a lot of them. Should they be tracked as assets? Some companies think so. Some companies just buy them and forget about them.

TeamEDA recommends tracking all standalone licenses that cost > \$2000 each.

In License pricing, node-locked is the cheapest. LAN is typically 3-4x node-locked. WAN is typically 10-12x node-locked. So the optimal combination of WAN, LAN and node-locked is important in minimizing overall cost, while providing acceptable tool access. 100% guaranteed Tool access at any time is unrealistic and would be very expensive. Some level of wait or license denial may be perfectly acceptable. Most companies strive for 80-90% tool access probability.

Let’s look at the kind of information about that can be tracked in LAM.

- Application/Tool name and quantity
- Where the license resides
 - o If shared: license server, server “site”, daemon, port
 - o If node-locked: host ID, User name, “site”
- What the application/Tool does (Discipline)
- Vendor who offers it (or Reseller who sold it to you)
- How much did it cost
- License key expiration date*
- Who approved the purchase*
- What PO was used to buy it*
- What did the Vendor give away on that PO (discount, free options, free support,...)
- Type of License, i.e. how was it purchased (Perpetual, Subscription, Tokens,...)*
- What Requisition was used to support that PO
- What license server and port number is the License authorized to run on
- How is its use restricted (Node, User, LAN, WAN,...)*
- What is the expiration date of “features”
- What is the version of “features”
- Is there an approved production version
- Renewal date of the Contract*
- Where is the Vendor located
- Who are the contacts for each Vendor (phone #, email address, etc)
- What Contracts and Agreements were signed with that Vendor*

Appendix A details the proper for entering data.

(* needed for Vendor or internal “audit”).

Let's look at the kind of information available in LM

- How many licenses of specific feature are currently in use, who has them checked out, for how long, and what host is running the application
- Who are my power users
- Are there any "denials" for a given feature
- What are historical usage patterns by product, User, site, discipline, etc
- Patterns of usage over a typical week
- Trends in usage
- License checkout time exceeds threshold alert
- License type approaching checkout saturation alert

What kind of company are you?

Some Companies are very concerned about managing license inventory, desire an effective approach to controlling these costs, and are willing to invest resources to get there. Some companies don't mind winging it, considering engineering software costs just a part of doing business. Some companies are very concerned about being legal, being prepared for audits, and will want to track all information relating to "compliance".

Prior to LAM, most companies kept spreadsheets of some license information. There could be 2-3 different people keeping 2-3 different spreadsheets, and not sharing any information. In the past, license management was decentralized. Some companies have been happy just tracking feature usage, hundreds, if not thousands of features, not concerned about mapping feature-product. Experienced Administrators may know the features to look for, and seeing usage of that feature is enough to decide on "more or less" product licenses. If you aren't experienced, you'll need to find a feature-product map.

Some License Administrators don't care about Vendor contracts but do track Vendor contacts in Outlook or business cards in the drawer. Some companies actually do want License agreements and contract accessible to the License Administrators. Companies concerned about audit will probably want Contracts and Agreements centralized in LAM, to reference terms and conditions as needed. Some companies simply don't worry about contracts and agreements. Some companies will want to use LAM as a centralized Contact Management system to keep track of Vendor Contacts, and share that information. Some companies are fine keeping multiple Excel spreadsheets by multiple people on various pieces of Vendor or License information. So what kind of Company are you?

Here is our recommended methodology:

- a. Hire TeamEDA consulting to help initialize the project
 - Conduct a tutorial session on "best practices"
 - Perform survey on current tools, license servers, licenses and methodology
 - Assist on nomenclature development
 - Perform LAM User training
 - Help obtain and load initial data
 - Initialize the usage monitoring components
 - Perform basic LM training- Current checkouts and historical usage reporting
- b. Track expensive (>\$2k) node-locked licenses- Host ID, Site, and User name.
- c. Track all Shared licenses, and decide if you want to track at Product level or Bundle level

- If tracking at Bundle level, monitor all features in the Bundle
- If tracking "Products", monitor usage on 1-2 features each.
- d. Allow "browsing" to license files
- e. Use a License file revision control system (ala CVS) for license key updates
- f. Set up unique Tag names for daemons, and spend time up front deciding on nomenclature
- g. Setup unique Restriction types, and spend time up front deciding on nomenclature
- h. Setup unique Purchase types, and spend time up front deciding on nomenclature
- i. Setup unique License types, and spend time up front deciding on nomenclature
- j. Incorporate your company logo on asset/inventory reports (LAM pages)
- k. Setup "Admin" Users of LAM
 - Attend all LAM training sessions
- l. Setup "Manager" Users of LAM- provide them URL to log-in page
 - Attend an abbreviated LAM training session
- m. Set up "Admin" Users of LM
 - Attend all Basic and Advanced LM training sessions
- n. Setup "Casual Users" of LM and provide URL to "portals of interest"
- o. Track Vendors and Contacts in LAM
- p. Track PO's if audit is a concern
- q. Allow "Purchasing" access to LAM to get contact info and PO history.
- r. Track Requisitions only if the acquisition thread needs to include Req's for each PO.
- s. If renewal cost budgeting is necessary, track cost/line item in PO
- t. Determine periodic inventory reports needed..create batch or portals
- u. Determine periodic usage reports needed..create batch or portals

Let's say you want to implement LAM to manage Licenses, centralize and share Vendor, Contact, Tool, Server, daemon and License information. How do you get started? Here's one scenario:

- Choose a host server for LAM
 - o Linux
 - o Windows Server 2003/2008
- Assign an IT person to work with TeamEDA to download, install, configure, and test the software
 - o The IT person will then not be needed except for periodic "upgrades"
- Assign a Software License Administrator (full time recommended)
- Attend training sessions provided by TeamEDA
 - o Methodology
 - o Data import sequencing
 - o Excel importing
 - o GUI
 - o Report generation
 - o Email alerts
- Obtain needed Vendor information
 - o Create a Excel spreadsheet
 - o Enter/import into LAM
- Create list of Tools/Bundles to be used
 - o Create a Excel spreadsheet
 - o Enter/import into LAM
- Enter License server, daemon and Tag names
- Obtain License information

- Create a Excel spreadsheet
- Enter/import into LAM
- Point (browse) to license file or select port@host
- Select 'features' from parsed license file data.
- Attend training session provided by TeamEDA on LM Basic use
 - Current and Historical Tabs
 - Create "portals of interest" for certain engineers and managers
- Attend training session provided by TeamEDA on LM Advanced use
 - Threshold alerts
 - License file management
 - Daemon management
 - Batch reports

Maintaining data

- Add a new Vendor- manually update LAM
- Add a new Tool- manually update LAM
- Add many new Vendors- use Import Spreadsheet
- Add many new Tools- use Import Spreadsheet
- Add new License Server or daemon,
 - Manually update LAM, or
 - Point to new License file for that license, or
 - If license file already mounted, click Update LAM from LM (from Servers page)
- Vendor Contact info changes- manually update LAM
- Subscription or Maintenance renewals,
 - Manually update quantity, PO#, Expiration date,....., or
 - If you want to save history (save the old line item in LAM),
 - Clone the old license information, update quantity, PO#, Expiration date,.....
- New License file mounts-
 - LAM License will require a Reread, which will update feature info
 - User LM for revision control (LM Admin)
- Add a new License- manually update LAM
- Add many new Licenses- use Import Spreadsheet

Reports – Asset inventory

- Inventory by Restriction type
- Inventory by Purchase type
- Inventory by Discipline
- Inventory by expiration date
- Inventory by cost
- Inventory by Vendor
- Inventory by Site
- Perpetual Inventory with Maintenance
- Perpetual Inventory without Maintenance

Reports – Renewals

- Expected renewal cost by Vendor for any time period
- Expected renewal cost by Application/Tool

- Expected renewal cost for all Vendors for any time period

Reports – Current license features checkouts

- Checkouts by Tag name
- Checkouts by feature name
- Total Checkout time by User for any specific feature
- All Checkouts by a specific User

Reports – Historical license feature checkouts

- Usage (# checkouts) of a specific feature over any time period
- Usage (# checkouts) of a combination of features over any time period
- Heatmap (pattern of checkouts over a typical week)
- Trends in usage

Reports – Historical “denials”

- Denials by feature for any time period
- Denials by Tag name for any time period
- Denials by specific User
- Denials Heatmap (patterns of denials over a typical week)

Expiration Alerts

- License expiration for features (per license key)
- License expiration by Vendor

Threshold violation alerts

- License server down
- License daemon down
- Exceed Max allowed checkout time by feature
- Checkout saturation reached for any feature

In summary, there are a lot of different ways to approach License Management. How much you choose to manage is dependent on importance and resources available. In order of importance and ease to implement,

1. Track inventory of Shared license assets by Product or Bundle
2. Track Vendor information including Contacts
3. Track license expiration dates
4. Track license costs and POs
5. Track all Contracts and Agreements
6. Track feature checkouts of FLEXIm shared license managers
7. Track feature checkouts other license managers (LUM, Sentinel, Elan, Reprise, etc)
8. Track inventory of Shared and Standalone (node-locked) license assets
9. Track “Executable” of Standalone licenses

To fully implement LAMLM the following personnel should be made available:

- **IT person** to install the LAMLM software, setup email alert system, and install periodic upgrades (occasional job)
- **License Asset Manager**- person responsible for keeping track of license inventory of both shared and standalone license assets, contracts, POs, manage Vendor and contact information... (part time job)
- **License Administrator**- person who will manage license files using a revision control system (CVS), manage daemons, interface with IT, and create current checkout portals, historical utilization and denial reports as needed. (full time job if greater than 1000 Users)

Finally, if you choose to use TeamEDA consulting, here what the Implementation Consulting Statement of Work might look like:

The engagement is expected to be 5 days but can be terminated earlier by mutual agreement. Tasks associated with the project are:

- 1) Review and approve project goals and consulting plan at onset of project
- 2) Provide a tutorial on "best practices" methodology for engineering license management, reference existing LAM customers as needed
- 3) Conduct of survey of company license usage environment and methodology
- 4) Identify license information needed to achieve goals
- 5) Assist in collection of needed information
- 6) Assist in selection of appropriate company personnel for full implementation
- 7) Train User personnel on use of LAM (asset management component)
 - a. Train on use of excel import template for LAM
- 8) Train User personnel on basic use of LM (usage metrics component)
 - a. Basic reports- Current and History Tabs
- 9) Help company select appropriate naming conventions
 - a. License line item and types
 - b. Restriction types
 - c. Purchase types
 - d. Tag names
- 10) Populate LAM import spreadsheets as needed
- 11) Assist in the setup of email alerts
- 12) Identify "threshold alerts" needed
- 13) Assist in import of initial data into LAM (import spreadsheets)
- 14) Insure LM is producing accurate usage reports as needed
- 15) Create Current Check-out portals as needed.
- 16) Review progress of consulting effort after 2 days.
- 17) Review results of consulting effort after 5 days
- 18) Review project implementation status after 30 days

The consulting effort is estimated at 5 days, which could be a combination of on-site and remote based, with company permission. On-site work hours will be same as company work hours.

This consulting will commence after LAM LM is installed and tested. SOW is subject to change by mutual agreement.

For more information please contact TeamEDA 603-656-5200, or info@teameda.com

Appendix A: Recommended sequence for entering data into LAM

1. LAM Users, title, phone,...
 - email (if to receive expiration alerts)
2. Vendors (import spreadsheet?)
3. Tools and Bundles (import spreadsheet?)
4. Shared License Servers
 - Site
5. Shared License Daemons
 - Unique Tag names (if desired)
6. Shared Licenses (import spreadsheet?)
 - Vendor
 - Tool or Bundle name
 - Quantity
 - Expiration date
 - License key location (browse to “file” or select daemon@host)
 - Associated PO # and item cost (if known)
7. Parse License file and select key Features to be monitored
8. Add additional Vendor info
 - Contacts
 - Contracts/Agreements
 - E-copies of documents
9. Add additional PO information
 - Charge center
 - E-copies of documents
 - Requisition data
10. Prepare “Type” lists (in Settings)
11. Add all LAM User types (Managers, Responsible Engineers, IT support people,...)
12. Add additional Shared License information
 - PO and cost
 - Responsible Engineer
 - Restriction type
 - Tool type
 - Purchase type
13. Add Standalone/node-locked Licenses (if desired)