Effective Fail-Over Without Redundant Servers

This document outlines the reasons why redundant license servers should be replaced by the more elegant and straightforward approach used by the Reprise License Manager (RLM™).

Why Redundant License Servers?

The concept of redundant license servers dates back to the early days of software licensing software circa 1988. At the time, networks and servers were unreliable. Fixing them was a very time-consuming process, often taking days or even weeks. So, when a server failed causing a license server to go down, customers wanted a way for their licenses to be available during the outage. Vendors addressed this need by designing a scheme in which redundant servers operated together with exact replicas of server software and license files on a trio of servers. There was always only one “master” server, but in the event that that server went down, license service remained intact as long as the other two servers could communicate with each other and client applications could connect to the new master. The primary perceived benefit of a properly configured redundant triad was that a customer’s complete inventory of licenses would remain available to at least part of the user community in the event of a single server failure.

But even with redundant servers, there was always an assumption that the licensing universe consisted of a single “master server.” Applications were designed to “lock-on” to a single license server once it was found. This caused poor queuing behavior and forced application developers to program around server “lock-on” constraints. Redundant servers have survived because of the lack of a better approach – until now.

A redundant server sounds like a good idea, but is the added administration complexity and overhead worth the trouble in today’s more reliable computing environment? Are redundant servers still the best approach? If vendors and their users could rely on a more robust solution, would they still prefer to use redundant servers?

Let’s examine the downside of redundant servers.
Server Redundancy Problems

Poor server configuration:

In order for a redundant server trio to operate properly, each server must have a complete and independent copy of the license server software and license files. Often this setup is not achieved. Instead shortcuts are taken to make the administrator’s job easier, such as placing a single copy of the server software on a master server and then mounting those file systems to the other servers. This creates a single point of failure, defeating the primary purpose of the redundant setup. If the master server goes down, then there is no way for the other two servers to continue to serve licenses because the software and license files that they need are available.

Extra administration coordination:

If each license server in the triad is managed by a different team of system administrators, then special coordination is needed between them whenever any change is made to any of the license files.
Poor scaling characteristics:

Since all licenses that belong to a redundant server group must be commingled by design, there is no way to improve licensing performance unless you upgrade all three of the server computers that serve the license pool. And even if you upgrade, there are still operating system limits that cap the number of clients than can be served by a single server (whether it is a redundant server or not), forcing multiple servers to be deployed. In the case of redundant servers, this must be multiple sets of redundant servers.

Version incompatibility between server releases:

Whenever a new license is introduced into the common redundant license pool, administrators have to check the version of the license server software associated with that license. If it is later than the current version, then it needs to be copied to all three redundant servers simultaneously to keep the redundant trio viable.

Some advanced features may be unavailable:

Often end-users are unaware that they are making functionality tradeoffs by selecting redundant servers. With redundant servers, access to advanced license management features, such as license borrowing, may be awkward or simply unsupported.

Network segmentation problems:

Since redundant servers operate only when at least two servers can communicate with each other, a network failure can take one of the servers offline. If you are a user who is on the “wrong side” of a network outage, then you cannot access any of the licenses in the managed pool, even though the local server that was part of the triad is functioning normally. In this case, the entire license pool is available only to the users who are lucky enough to be on the “right side” of the break. With multiple independent RLM license servers, client licenses are available on any part of a network that can access any server.

Report log consolidation problems:

Whenever a redundant license server fails a new master server must be selected. If that server had been writing to a report log, then it must later be combined with the original report log in order to obtain a complete and accurate picture of license usage. If the clocks are not completely in sync, then the report log may be difficult to accurately reconstruct.
Reprise License Manager – A Better Approach

What’s different? Decentralized and Independent License Servers.

The Reprise License Manager does not support redundant servers. Instead, it supports advanced technology that allows end users to benefit from the freedom of deploying multiple independent license servers. When vendors use RLM, they take advantage of this new approach without writing a single new line of code.

In older license managers, support for redundant servers pre-dated the ability for licensed applications to communicate reliably with multiple independent servers. As a result, many of the features of these products simply worked better with redundant servers than with independent servers. From day one, RLM was designed with multiple independent servers in mind. As a result, Reprise Software has been able to make multiple servers operate in a much more transparent and efficient manner.

![Diagram of Virtual License Pool]

RLM_LICENSE=5053@winter;5053@spring;5053@summer;5053@fall
**Transparent reconnection to dispersed license servers:**

RLM has re-engineered the basic technique with which licenses are checked out at runtime. The notion of a single “master” server is gone. In its place is the concept of multiple independent license servers, each with equal ability to serve licenses to the licensed application.

**Virtual License Pooling:**

Conceptually, you can think of the decentralized, independent RLM server approach as a Virtual License Pool. The pool consists of all the licenses managed by servers that are listed in a user's license environment variable (RLM_LICENSE or isvname_LICENSE). Instead of aggregating redundant licenses into a single license file, and then replicating it across all three redundant hosts, RLM aggregates the licenses virtually, leaving the licenses on servers where they are most conveniently installed and managed. RLM transparently connects, queues, and reconnects to the entire list of decentralized servers giving each user access to his own virtual pool of licenses. The completely independent RLM servers do not have to be configured to be aware of each other at all.

**Intelligent Queuing:**

Additionally, the above redesign supports full license queuing across multiple license pools. With older license managers it was impossible to queue at any but the first pool of licenses within a license server. That means that there could have been pools of licenses that were available yet inaccessible to your application. In contrast, RLM transparently queues on each server (and on each license pool within the server) that could satisfy the request. Once the application acknowledges the license grant, all secondary queued licenses are automatically dequeued. This happens completely transparently to the application code. End user site can set or unset queuing to suit its own local conventions.

**Better Scaling:**

With RLM, expanding license service can be accomplished by simply adding server machines into the existing network.

**Simplified Administration:**

With RLM, license servers are installed where they are most needed. You install it in one place, period. You don't have to worry if there are correct license files, binaries, etc. on each of the three machines. Each license server has its own report log. Your report log is in one place, not dispersed across multiple servers depending on whether servers were running continuously.
About Reprise Software

Reprise Software was founded in 2006 by the creators of FLEXlm. We develop software licensing products, and software licensing products only. Benefit from our 30+ years of experience in the field of license management to have an ‘it just works’ experience. At Reprise, we continuously focus on making our products easier to install and use, so you can have licensing as a benefit - not as an obstacle – to your daily work. Reprise Software is driven to make RLM the best software licensing system available while helping lower the total cost of ownership of licensed applications for both software vendors and their customers.

13388 Old Airline Highway
Paicines, CA 95043
United States
Phone: +1 408-907-6756
Fax: +1 408-404-0890

info@reprisesoftware.com

terprisesoftware.com

Rev 2.1 April 2018