Decommissioning Oracle Applications: A Comprehensive Guide on Steps to Retire Oracle EBS

Rajalakshmi Thiruthuraipondi Natarajan

rajalan11@gmail.com

Abstract

Oracle Applications EBS decommissioning is an end-state activity where the application is deemed no longer useful to the organization and is removed from daily operations, eventually sunset and the underlying hardware repurposed. The retirement scope can be as small as a single module or the entire suite along with or without the underlying Database. Depending upon the architecture of the application, there needs to be several steps taken while preparing, during and post decommissioning the application. It is imperative that every part of the application like hardware, data, set-ups, etc must be examined closely and carefully plan for its disposal. Since application once retired can no longer by accessed and is gone for good, all functionalities that their provided during its tenure should be examined and determined if it is no longer needed or replicated in the new system that is replacing it, all while taking the security into consideration.

Keywords: Oracle EBS decommissioning, Application Retirement, Oracle Database sunset, Oracle EBS end-of-life

I. INTRODUCTION

A life cycle of any product start with installation and ends with its retirement. For someone involved with the application, this might not be a pleasant topic, yet inevitable. All technology sooner or later would become obsolete and be replaced by a newer, better product, leading to its retirement. There are a few old school products in the market, which are still being actively used even after several decades, at the same time there are some products that lost its relevance within few years of its service. Either way, all products will have to go through retirement someday.

Though Oracle Applications is one of most complete ERP products in the market, which can serve pretty much any operational need, there is always a possibility that an organization might feel that it is no longer a right fit for its IT goals and can be marked for retirement. Though Oracle decommissioning might not be as complex and resource intensive as implementation, there still is a critical activity, which needs to be carefully planned and executed so that the application is safely and securely disposed. Prior to decommissioning, there needs precautions taken and checklists defined based on the complete scope of decommissioning, the future plan for the hardware, if any, and the executed in a proper sequence. With data becoming one of the most valuable commodities these days and the hackers getting more innovative and resourceful, it is extremely important that the retired application is in no way vulnerable to these threats and a source for such malignant attacks.

II. PREPARING ORACLE E-BUSINESS SUITE FOR END-STATE

Preparation is the key to success, and it perfectly applies to a product retirement too. When implementing Oracle EBS, or any product for that matter, seldom does the team plan for its end-state. In fact, a roadmap for strategic decommissioning is even initiated only when the decision is made to retire the application. While implementing any new product brings in new ability and features to enable user and business to improve productivity, the retirement effectively takes away an existing ability away from the user, which might or might not be available in the replacing solution. Hence it is all the more important to carefully plan and execute the retirement and guarantee that pros outweigh the cons.

A. Scope of Decommission

Just as requirements gathering, which is the first step of implementation, understanding what is getting retired is the first step of retirement. With Oracle Applications being as large as it is, the extent of retirement can range anywhere from a single module to total scrapping of the application and its underlying systems. By having this scope finalized, the team will have a clear perspective of the complexity and build the roadmap for retirement.

A single module retirement essentially means that a section of Oracle application is either getting moved to a different product or is no longer user used, however, other modules remain untouched and are actively used. For example, a company can decide to move Payment activity to a centralized solution but decide to retain Payables Invoicing and General Ledger. On the other hand, the retirement might mean that the entire application is ported to a different solution and the application along with its features is no longer needed.

The next scope is the database decommissioning, with in most cases will be along with the application. It is extremely rate that an application is retained, but the database is retired. This usually happens if the company adopts a new database solution, such as cloud IaaS or a new and improved Database and retaining the old database might not be cost effective for the company. Though primary used by Oracle EBS, it is possible that the database is shared with other systems for their operations or for reference, in which case proper accommodations needs to be made so that the dependent systems are not impacted by the database decommissioning.

Or, the entire application and its underlying database and hardware gets decommissioned and the server get repurposed for a different application. This type of retirement is the ultimate form, where the retirement plan needs to consider every piece and have the loop closed. In addition to the database and application considerations mentioned above, the hardware needs to be evaluated, and the end-state needs to be determined. While it is quite possible that the hardware might get repurposed, it is always possible that it might too old or out of sync that retaining it might be a costly affair than scrapping it, since every server, irrespective of its value might have overhead or maintaining it, in terms of support team contract or location rental, or even as simple as holding up real estate. One must also take into account that even scrapping the server would cost them since it needs to done safely and securely to avoid any dumpster diving hacker from getting hold of any company related data.

B. Data after Application Retirement

Data after Application Retirement might be the most critical piece that the entire retirement plan revolves around it. Irrespective of the scope, an application or its underlying components, once retired, is removed from the organization’s operations and can no longer be used and is under no obligation that any data maintained by the application be available to anyone including IT teams and admins. Please note that the application is not considered retired, even it is used for read-only purpose, in which case it would fall under the archival category.

A detailed knowledge of the data collected and maintained by an application is mandatory to take an

educated decision on what to do with the data. The most important question in this section is

- Is the data in the application needed by the organization, either for their operations or for legal purpose?
- Who might need this data and what purpose?
- What is the impact if the data is not available if needed?

Based on the answers for the questions above, the team needs to strategize the data handling, wither to archive the data or delete them.

While it is very rare and a risky decision, the company might decide to completely discard the data as this is no longer needed. This usually happens, when an application remains in a dormant state for several years and there is a clean-up project imitated as part of a software audit, in which case, the plan needs to be out in place to delete the data in a secure manner. This could be anything ranging from the master and transactional data stored in the database or the application configurations or user accounts and their accesses to the flat files such as the inbound/outbound data file on the server, even log file for that matter. Any or all of these data might have something significant that in wrong hands might damage the company.

C. Application Functionalities – Retain or Retire

As mentioned earlier, retiring an application means that the ability and the functionality that the application provides to assist in the company's operations goes away along with it. Hence, the user of the retiring application needs to aware of this fact and should be fine with it. All divisions that use this retiring application need to do a reality check and understand the risk of not having the application anymore.

As part of the preparation for retirement, each and every functionality, reports and the tools need to be evaluated to assess the damage it might case if the functionality is no longer present. If deemed necessary, then they should identify if the new solution has this intended ability in some form and should the tool or the functionality be transferred to the new application.

This is an extremely sensitive activity, since even with the organization, there could be different division using the same application in different capacity and retiring a functionality might impact at a different level for each of these divisions. While this might not be the responsibility of the retirement team to ensure that the users have an alternative, it is important to have this documented and accounted, as every team needs to sign-off the application retirement and this might be a heated topic while having different teams in the room to gauge the impact.

D. Application Dependencies

Almost every system in an organization's IT landscape will be talking to one system or the other. It might either be receiving some data or feeding to a downstream application for further processing. Hence when an application gets decommissioned, it is imperative to know the impact to these dependent systems. The obvious solution is that the replacing application or applications would replace the retiring system, but nevertheless, an impact analysis is mandatory to see its effect on the bigger scale.

Not only the target systems, but the middleware, which acts as a transport medium to interact between different systems needs to be investigated. These middleware needs to be checked for any connectors and subscriptions that are currently running to feed or retrieve the data from the retiring application and should be gracefully terminated, failing which might lead to data leaks or unnecessary load on these middleware systems or continuous invalid failure notifications that might take up the resource and time.

These dependencies can be at any level, and each should be individually addressed and aligned. For instance, an external application might be using Oracle Application for authentication to access their application, or Oracle Database might be used by a reporting system to pull data and generate reports for financial teams or the server on which the application is installed might be used by various other application as a common pool to store and retrieve data files. In any case, an appropriate solution or workaround needs

to be put in place before the application be retired.

III. RETIRING ORACLE APPLICATIONS

After the preparation activities are complete, the next logical step is not initiate the decommissioning process. Based on the preparation steps and the information gathered, all necessary remediation steps need to be considered and complete for smooth transition and decommissioning. One essential check is to make sure that necessary archival is done so that in the unforeseen situation where the data from retired Oracle application is needed, there is some way that it can be retrieved.

It goes with out saying that the step 0, is to make sure are all systems are a go and the application users, both internal and external, and by extension, entire organization, is aware of the retirement process and necessary sign-off from every involved party is received and recorded before proceeding

A. Terminate User Accesses

The first step is terminating all functional users' access and retain only minimal IT and admin accesses. This is to ensure that no new activities are triggered that might impede with the retirement process.

At the application level, all users access except admin users like *SYSADMIN*, or any other organization specific admin users, needs to be disabled, even for Inquiry access, since eve inquiry access can create access logs in the system.

The same applies for the data base and application level. DBA related accesses such as *APPLSYS* or *SYSTEM* at the DB level and *root* or *applmgr* at the server level are some of the only users who should be active in the system.

B. Stop all Interfaces

The next source of transactions are the interfaces that feed data into the system from external systems. These interfaces need to be shut down. Typically, there interfaces would either be a flat file loaded into the server which is loaded using SQL* Loader, or a direct JDBC connection or XML gateway, among other ways. There needs to be a clear cutover when the interfaces will be stopped so that there is no feed that gets lost during the retirement. Ideally, this needs to be done before the archival process if planned. However, if any delta archival is planned, it is important that any feed is stopped or redirected.

C. Suspend Non-Critical Processes

If possible, all concurrent programs, workflow processing, background process needs to be stopped before retirement. If not, except for the most critical jobs, all other jobs need to be terminated. And background process that is active might interfere with the retirement process. This is similar to the message one might have faced while shutting down a laptop, where it says background process stopping from system to shut down. Any interruptions during decommissioning might leave the application exposed since there is a high chance that the security might be compromised.

D. Remove the application from the Network

All access to the network needs to be severed and the application needs to be made inaccessible with in the intranet and outside the company. The only possible access that should be retained is directly in the server, preferably with the server rooms. This is needed, since at the time of retirement, the security will be severely low and the data can be stolen right before it gets destroyed.

E. Data Scrambling and Removal.

Data scrambling is the process of intentionally corrupting the data so that it is unusable. There are

several professional scrambling software in the market, which can be leveraged to scramble the data make it useless for anyone accessing it, both within and outside the organization. Post scrambling, the data can be systematically deleted as needed keeping the dependencies and reference keys in mind, like child table deleting before deleting the parent table. This might be a bit of a overkill, yet it is important that the sensitive data not compromised. Also, this step might not be needed for all data. The compliance laws and the kind of data stored needs to be examined and decision needs to be taken case by case basis.

F. Retire & Power down the Application

This is the final step of the retirement. Depending upon the provision provided by the application, it needs to be properly uninstalled, like Oracle Installer can be leveraged to uninstall certain Oracle products. Once uninstalled, the entire server can be reset back to its original version, it possibly might reset the server back decades. However, it is needed to scrap clean. Once all is done, the final formality id power down the server and it is done. THE APPLICATION IS OFFICIALLY RETIRED!

IV. POST RETIREMENT

These are some of the optional post-retirement steps that can be done to add additional precaution and make the hardware ready to be repurposed. It is recommended that the server remain out of network for a brief period of time where it can be reanalysed and scrubbed of any residual data, before repurposing it. Once the teams are confidant that there is no issues the hardware can be patched up to with latest patches and bring it ready for reuse.

V. CONCLUSION

Application retirement is the last thing ant that any planning is made. However, it is quite important that it is done in a right way. While this article covers various steps and activities that lead up to a successful retirement, not all might be needed by every organization or application. It is the job of the project team to chart down what is needed and come up with a custom solution, which can safely and securely retire the application

REFERENCES

[1]“How to Retire Oracle E-Business Suite | Inoapps Cloud Services”. https://www.inoapps.com/insights/news/how-to-retire-oracle-e-business-suite/, Nov-2021

[2]“Reducing the Oracle E-Business Suite Data Footprint (Doc ID 752322.1)”, https://support.oracle.com/epmos/faces/SearchDocDisplay?_adf.ctrl-state=158gjnql7y_4&_afrLoop=333139257028155#LISTPROG, May-2022.

[3]“Application Decommissioning: Benefits, Challenges, Checklist and Best Practices”, https://avendata.com/blog/simplifying-it-application-decommissioning

[4] Lucas Jellema, “Use it or lose it. On Application and Feature decommissioning”, Technology Blog AIMS, Jan-2023

[5]Adrian Neumeyer, “The Missing Checklist for Application Server Decommissioning”, https://www.tacticalprojectmanager.com/system-decommissioning-checklist/

[6]Feroze Arif , Divyesh Vaidya, “Legacy Application Retirement Guide”, https://cdn-prod.scdn6.secure.raxcdn.com/static/media/a75a3ae4-5443-45bb-97c4-ccd88aac7cc4.pdf, Nov-2017

[7] K Kumar Guduru, “Database decommission process”, https://www.slideshare.net/slideshow/database-decommission-process/69254201, Nov-2016

[8]“Deinstallation Examples for Oracle Database”, https://docs.oracle.com/en/database/oracle/oracle-database/23/ladbi/deinstallation-examples-for-oracle-database.html,