



THE DEFINITIVE

APPLICATION RATIONALIZATION GUIDE

Introduction:

How many applications does the average enterprise use? Take a guess before continuing. 50? 100? Maybe even as many as 500?

The actual figure is 1516.

And this is not just an enterprise level problem either, with research from Okta finding that the average across all large firms has grown by 68% over the last 4 years, up to 129.

Given the continued developments in technology and growth of new business models, it is perhaps unsurprising that the number of applications a firm uses has ballooned. Nonetheless, this rapid growth will impose heavy costs upon firms, not just from a financial point of view, but from an efficiency standpoint as well. Every application in a firm's portfolio needs to deliver business value. It's no surprise that Application Rationalization has become one of the most important initiatives for CIOs and Enterprise Architects today.





What is Application Rationalization?

Application Rationalization (sometimes referred to as Application Portfolio Rationalization) is the act of reducing the size of an organization's application portfolio. In other words, using fewer applications than previously, typically done with the aim of cutting costs and improving efficiency.

When discussing costs, the typical term used is Total Cost of Ownership (TCO), which is a calculation of all costs involved over time with IT equipment. Gartner highlights the following as areas that might contribute to TCO:

For IT, TCO includes hardware and software acquisition, management and support, communications, end-user expenses and the opportunity cost of downtime, training and other productivity losses.

Typically led by the CIO of an enterprise, Application Rationalization is of vital importance to enterprise architects, with Orbus's own research indicating it is the main use case for enterprise architecture departments.

The scale of Application Rationalization can vary, with industry leaders likely to take a continuous, careful approach to application portfolio management that minimizes the need for major change, while less active companies may need huge, enterprise wide projects that need large investments of time and money.

Why Rationalize?

For those who guessed incorrectly in the Introduction, this is probably self-explanatory – companies are using far, far more applications than seems necessary, and there is ample space to save money by removing redundant software. Gartner estimates that firms which follow best practices for application rationalization can reduce total cost of ownership by 30%. Given that application costs are estimated to make up 80% of the entire IT budget, that can be a significant saving.

It is not merely financial cost that rationalization can impact upon either. Opportunity cost is another possibility. Given the nature of technological development, applications can cause opportunity costs in two ways. First, clinging to older versions of certain software will prevent an organization from making use of new features and efficiencies as they advance. This is true even if a firm actively updates and invests in their applications. For example, a firm which uses Office 365 may still have certain locations or teams making use of Office 2007 due to simple inertia. The second opportunity cost arises due to simple budgeting. IT departments have limited budgets

like every other function of a business, but face unique pressures for continual investment. Any firm which wants to exist on the cutting edge of technology will by necessity need to keep acquiring new applications, and indeed this likely explains why application portfolios have become so bloated. Nonetheless, there will come a time when an IT department has no choice but to delay implementation of technologies or applications for budgetary reasons. Consider the example of a retailer in the year 2005 that chose not to invest in e-commerce solutions; the opportunity cost there could be massive.

Opportunity Cost ties into another benefit, which is increased efficiency. It almost goes without saying, but an IT department has to manage hundreds of different applications and this complexity can slow everything down. Removing applications from the portfolio reduces the burden of maintenance, as well as not having to train staff to use dozens of unnecessary applications. More generally, application rationalization can be the first step towards better application portfolio management. Maintaining a centrally managed application portfolio requires effort at the best of times, but when a business can't even keep track of all the applications that they use it is practically impossible.





Barriers to Application Rationalization

The Application Portfolio

It's a tautology, but if you're looking to undergo application rationalization, then your application portfolio is likely to be complex and unmanageable. If your portfolio is sufficiently cumbersome, then the very act of identifying applications in need of rationalization can be difficult. If you can't even keep track of every application that is in use or paid for within a firm, it will be impossible to make decisions on redundancy or cost.

Even if a firm is confident that it has records of every application within the portfolio, the information itself may prove troublesome to extract. Application data can be buried in ancient Excel spreadsheets or Word documents, which can also contribute to collaboration difficulties if information is not easily passed between teams and team members.

Visualization

Assuming that you can access your application data, the next challenge to overcome relates to making sense of the data. When a firm has upwards of 900 applications, it is no small feat to pick out targets for rationalization. A spreadsheet might be able to do the job for smaller organizations, but chances are you will need particular tools and techniques to be effective.

Poor visualization won't just impact on the rationalization process either. A key part of application rationalization is presenting to and gaining the buy in of stakeholders, which becomes challenging without user friendly presentation options. Rationalization decisions can have impacts on a wide range of business functions, and so even with the support of the CIO, push back could derail attempts to discontinue legacy applications.

Barriers to Effective Enterprise Architecture

Judgment

You have managed to gather all your necessary application data and have organized it in order to begin culling applications. What now? How does an enterprise architecture team go about deciding which applications are unnecessary?

While there will undoubtedly be some obvious choices in every rationalization project, only targeting low hanging fruit will not realize large gains. Utilizing industry best practices and methods can achieve 30+% cost savings, but these methods will often be opaque and unknown to inexperienced firms.

Collaboration

Stakeholder engagement is vital for successful application rationalization. This barrier arguably impacts on several of the others: for example, you would not be able to compile an application portfolio without application owners and users supplying relevant data.

Gartner suggests that application rationalization initiatives that do not have the support of business leaders rarely get off the ground, as rationalization initiatives often require change in business processes, which cannot be driven by the IT department.



The Steps to Successful Application Rationalization

Common focuses for a rationalization initiative include:

Functional Duplication – identify applications which provide similar functionality to others in the portfolio.

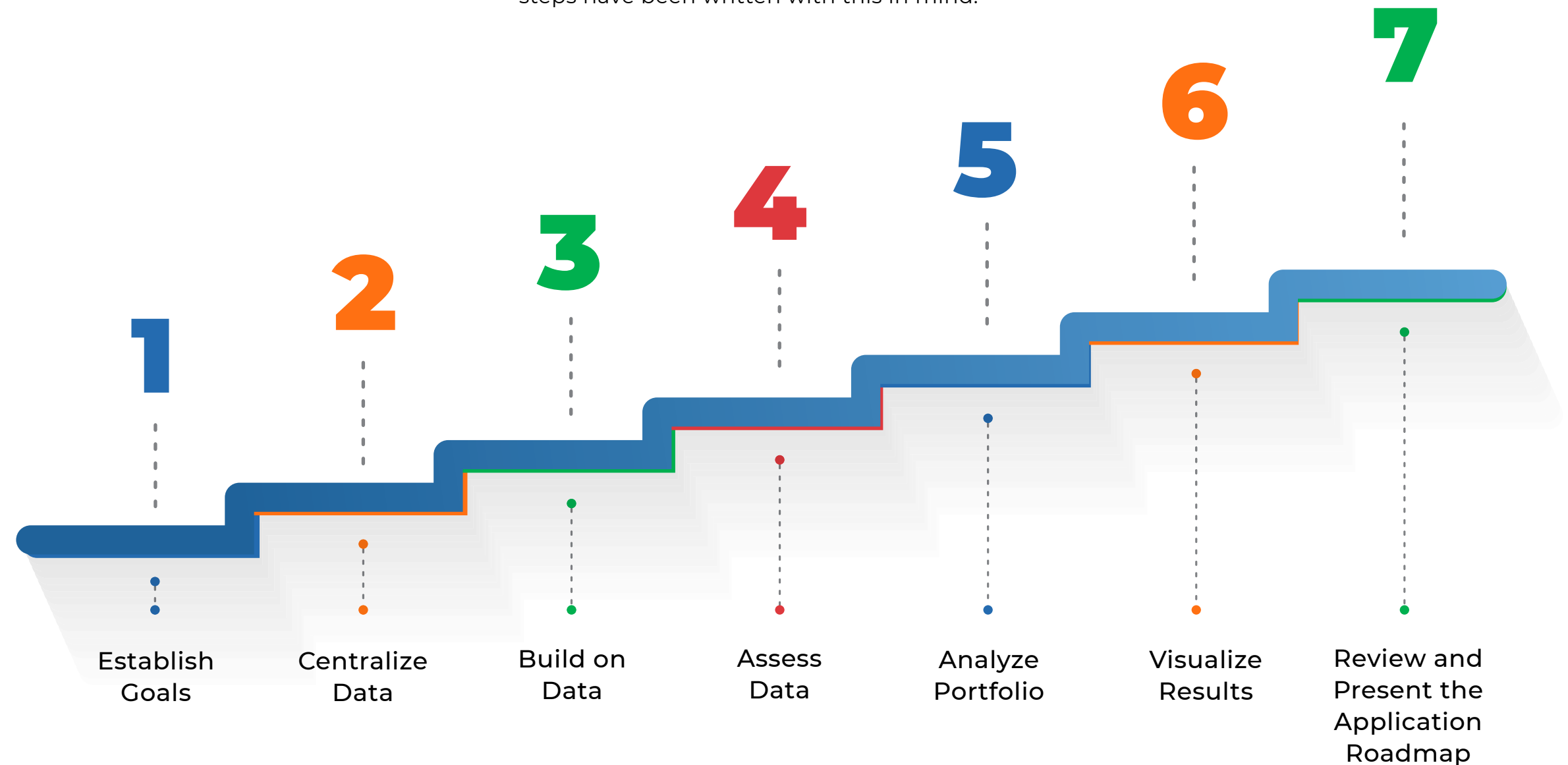
Vendor Consolidation – lower the number of application ‘suppliers’ that an organization deals with.

Strategic Importance – identify and visualize applications that are poorly aligned to strategy and are of low criticality.

Fit Analysis – scoring applications for technical and business fit; see below for more detail on these terms.

Application Lifecycle or Usage – review basic characteristics of applications to find quick wins, such as applications that can be retired early, or applications with minimal usage.

It may seem strange, but the best path towards succeeding with application rationalization is to use the right application. An enterprise architecture tool with a central repository, such as iServer, will drastically improve the ability to execute on rationalization. The following steps have been written with this in mind.



1. Establish Goals

This is an oft overlooked step, but one which is vital to any business initiative. What do you aim to achieve with application rationalization? While it is all very well hoping to get all the benefits outlined earlier in this paper, an unfocused approach is likely to lead to a slow, troubled process. The steps required to gather and analyze data are the most time consuming and having to account for a wide variety of different aims can all contribute to additional complexity. There are a broad set of angles from which to approach a rationalization initiative, though organizations can attempt some combination of the above if a larger scope is needed.

2. Centralize Data

With iServer's central repository, gathering all your necessary data in one place is not too challenging. Much of the work for this step will be manual, relying on accurate lists of application ownership or an up to date application catalog.

3. Build on Data

Once your application catalog is loaded into the central repository, you will then most likely need to augment the data with manually gathered information. Application ownership and lifecycles are particularly important, and of course the usage of each application is necessary. You can also link applications in the iServer repository to other key enterprise concepts, such as business capabilities or processes, to start to build a comprehensive picture of impacts.

4. Assess Data

Depending on the key criteria established in step 1, you will need to score applications to see which are in need of rationalization. There are three measures that are commonly used to score applications: TCO, Technical Fit and Business Fit (or Value). Business and Technical Fit are typically qualitative measures that determine how useful an application is to the business, and the technological 'health' of an application. iServer has predefined scoring methods for all of these measures, so you won't need to build your own, though interested readers can head to our posts on **Business Fit** and **Technical Fit** to find out more.



5. Analyze Portfolio

With scoring complete, you will now need to analyze and group your application portfolio according to your desired criteria, as established in step 1. A common sorting is TIME, which stands for Tolerate, Invest, Migrate or Eliminate, and was designed by Gartner.

Applications in Tolerate have high technical fit but low business fit. These applications don't need much attention, but should be monitored in case they degrade any further.

Invest, predictably, is for applications with both high technical fit and high business fit. As the name implies, companies should aim to invest and evolve these applications. For applications in migrate, they will have high business fit but low technical fit. These applications are valuable business assets, but will likely be in need of modernization to preserve their business value.

Finally, Eliminate refers to applications with both low technical fit and business fit. These applications should be decommissioned.

TIME analysis is a good starting point, there are additional factors that can be considered. Not every application assigned for elimination should necessarily be removed. Further factors to consider include:

Complexity

How heavily integrated is the application? Investigate the number of interfaces/integrations the application has, and discuss these with the application owner.

Data Implications

Is the application associated with any sensitive data? The nature of the data processed by the application may pose additional implications for retiring it early.

Replacement / Business Impact

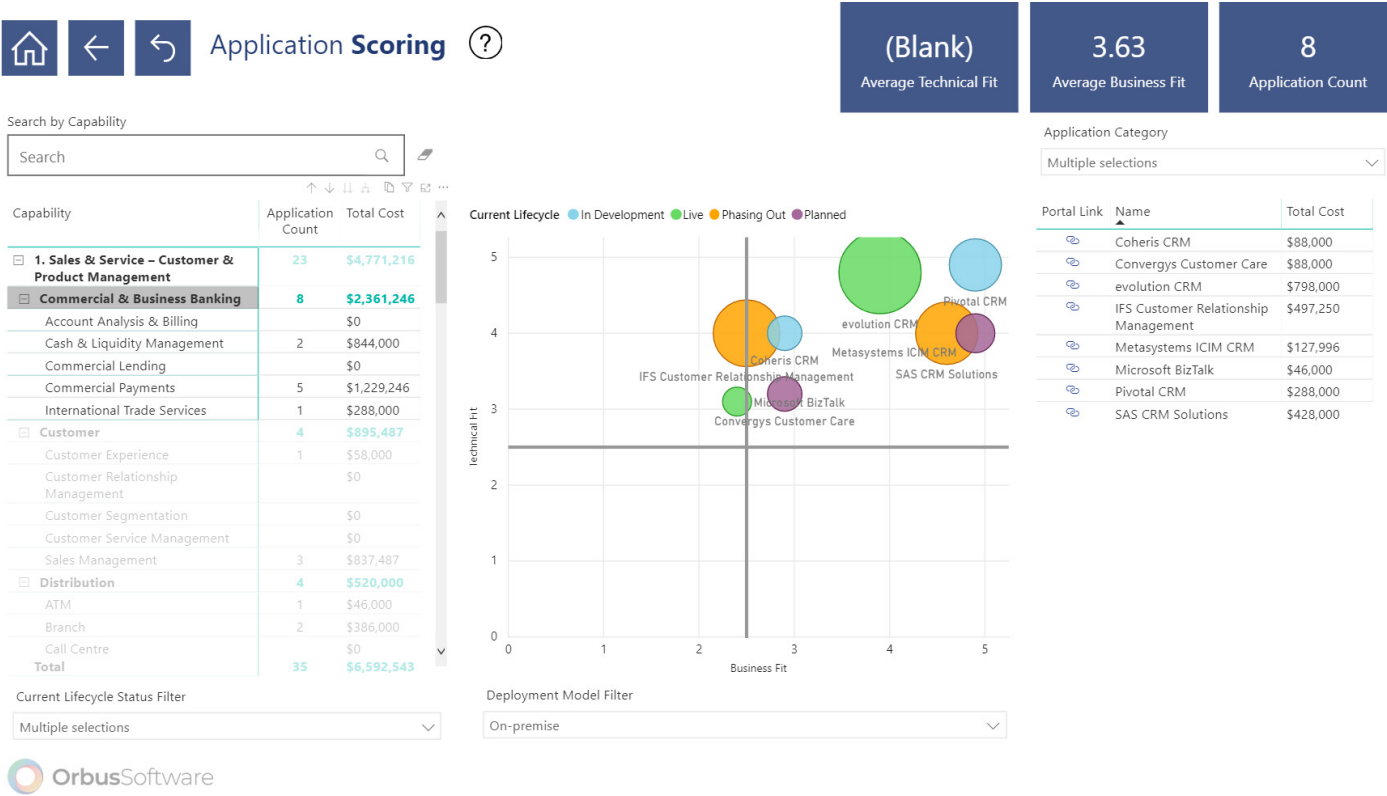
What would the impact of retiring this application be to users in the business? Interviews should assess real business impact from removing the application.

Cost Savings

How much would we actually save by retiring this application? Perhaps the most important consideration; an application should not be retired early if doing so will not result in a meaningful cost saving.

6. Visualize Results

After your analysis, predefined visualizations and reports can illustrate the results with ease. If using the TIME methodology, visualizing applications in a matrix will demonstrate this clearly.



7. Review and Present the Application Roadmap

To complete the project, enterprise architects will need to present their results to stakeholders, showing potential savings identified and other positive impacts, such as elimination of redundancy, as well as the selection of applications that will be rationalized. iServer offers predefined reports for this stage.





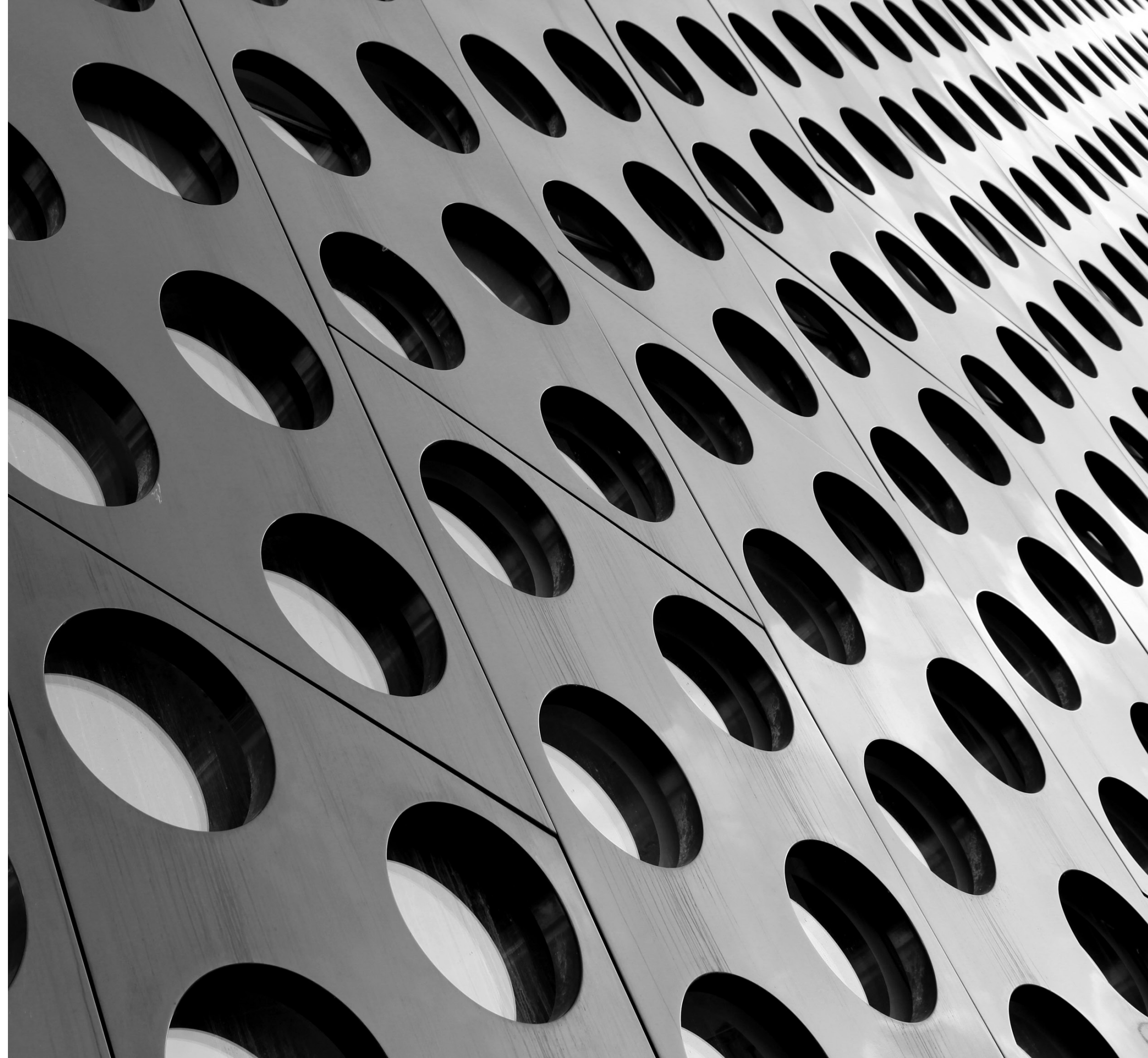
Maintaining an Efficient Application Portfolio

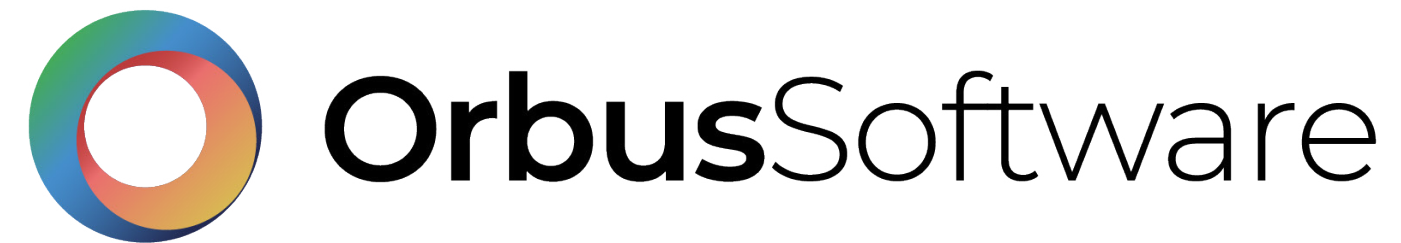
By following this guide, enterprises can drastically cut the size of their application portfolio and reduce their software costs. However, if an organization was able to accrue hundreds of applications, what is preventing this buildup from occurring again? For a truly successful approach to application rationalization, firms need not only succeed in the initial rationalization, but to complement this with effective, continuous application portfolio management.

iServer comes with a number of application portfolio management features built in, so for users the transition from an application rationalization project to continual portfolio management is smooth and simple. Prebuilt dashboards enable users to instantly view application scoring at any time and keep track of application lifecycles through a dynamic Gantt chart.

Conclusion

The need for businesses to maintain a technological edge has never been greater, but many are crippling their abilities by failing to keep control of application growth. Application Rationalization is not an exciting or easy initiative, but it is arguably one of the most important methods in the enterprise architecture toolbox, delivering outsized gains in a variety of areas. Ironically, the biggest reason to undertake application rationalization today is applications, with modern tools such as iServer rendering many tedious tasks far easier. Those driving rationalization will still need to be aware of the need to bring stakeholders on board and communicate the end result effectively, but the opportunity to cut software costs, reduce the burden on the IT department and start application portfolio management should convince even the most recalcitrant stakeholder.





© Copyright 2020 Orbus Software. All rights reserved.

No part of this publication may be reproduced, resold, stored in a retrieval system, or distributed in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior permission of the copyright owner.

Such requests for permission or any other comments relating to the material contained in this document may be submitted to: marketing@orbussoftware.com

Orbus Software

Floor 4, 60 Buckingham Palace Road, SW1W 0RR

+44 (0) 20 3824 2907

enquiries@orbussoftware.com

www.orbussoftware.com