



Application Modernization Guide

Support For Your Business Case



Table of Contents

Introduction

Is It Time for a Refresh?

Is It Time to Rehost?

Is It Time to Rebuild or Replace?

Conclusion



Introduction

Software applications are at the heart of every business. They help to improve efficiency, connect business processes, communicate seamlessly, analyse business trends and respond to market demands.

Not only are they an asset to your business, in the case of customer-facing applications they can also become a source of revenue and a competitive differentiator. Nowadays, everyone wants secure, quick and reliable access to vast amounts of information, right at their fingertips, and from any device they choose. With end-of-life software and hardware deadlines looming, and more companies undertaking digital transformation projects, how we use tech is quickly and continually evolving.

In this dynamic environment, software and applications are subject to a defined lifecycle and require on-going checks and maintenance to ensure they continue to work as effectively and securely as they did when they were put into place. Any application that's not working optimally could also cause you to lose customers or top talent who become frustrated by lagging, outdated systems.

How to Know Your Application Needs Attention

Digital transformation is often cited as one of the main drivers for application modernization projects and it's true that to take advantage of some of the innovations of the data-centric era, modern hardware and software platforms are a must.

“ ... legacy systems can lack the flexibility and scalability to keep up — especially given today's tech-savvy end consumers. They are often handcuffing the business with what we call legacy technical debt. It's a real problem. Modern concepts like microservices, the cloud, and mobile computing are simply beyond many legacy technologies. - Deloitte



But even if a major transformation undertaking is not in your immediate plans, you need to pay attention to how your legacy applications are performing and what needs to be done today to ensure optimal performance. Lagging performance and declining productivity, security concerns, increasing maintenance time and costs or users complaining about the applications are signs it's time to take action.

Application modernization doesn't have to be a massive transformation project. Project scope and scale can vary greatly depending on your business goals, application age and any new requirements.

Application Modernization Action Plan

Every application goes through different stages in its life. How do you know if you need a simple tune up or a complete overhaul? Our survey will help you identify some of the issues that impact applications to help you understand where yours might be in its lifecycle.

Our Application Health Check is broken into three sections – [Refresh](#), [Rehost](#) and [Rebuild/Replace](#) – but it is possible some applications might need more than one of these strategies (like a refresh and a rehost) to provide users and companies with the solution they really need.

While you might think you know exactly what stage you're in, we recommend reviewing all three sections for a full picture of the current state of your application and steps you could start taking to modernize it.



[Let's Get Started.](#)



Is It Time for a Refresh?

Application refresh is the most common and frequent update to an application.

A refresh can refer to a wide range of changes but modernizing the customer-facing user interface (UI) is one of the most common refresh projects. Other examples include updating the design, branding and icons, improving navigation, light feature improvements or addressing accessibility needs.

There are many signs applications need a refresh. Some of them come on gradually (like slowing performance or negative comments from users) and others seem to develop rapidly. Regular refresh cycles will help keep your application looking and performing at its peak.

But don't think an application refresh is something simple to be taken lightly: Any time you change an application, care must be taken to plan it out right so you don't create new problems while you're solving others.



Common Refresh Signals:

Here are some issues that could be a sign your application is ready for a refresh:

Usage Drops

Whether it's an external software solution or an internal tool, when people stop using it, that's a sure sign there's a problem. You need to assess why usage is dropping and correct the problem.

Complaints are Increasing

Complaints could be esthetic, functional and technical. Keep track of what the complaints are about and prioritize addressing them.

You Need to Comply with Accessibility Legislation

Ensuring your site or application can be read or viewed by those with visual impairments is required in many provinces.



“ Outdated enterprise systems not only slow digital strategies but weigh them down through digital debt — debt from the exorbitant amounts of money spent keeping these old applications running and from the inflexibility that hinders the business. - Forrester

Take Our Survey:

Yes No

Does your user interface look out-of-date?

Does your application lack a responsive design?

Was your application created or last updated prior to Provincial accessibility standards (such as AODA in Ontario)?

Do your customers/employees complain about the layout or design elements?

Is your application difficult for people to navigate?

Is poor or outdated design affecting efficiency and productivity?

Have you, or are you planning to, update your logo or branding?

Does your application have performance issues on different browsers and/or operating systems?

Is your business growing?

Are you targeting new markets (industry or geography)?

Total Your Results

Assign 1 point for each Yes



Your Results:

There are no hard and fast rules for an application refresh. Rebranding and accessibility might be your only two issues, but the urgency for these updates could escalate your refresh timeframe.

As a rule of thumb, the more challenges you're seeing, the more pressing a refresh becomes.

1-3 – Monitor Issues. There are a few items you want to address but they're not causing major disruptions yet. Plan to include them in your next refresh cycle in the next 12 months.

4-6 – You have some time but there's a growing number of issues you will want to address within the next 6-12 months.

6-10 – You're Overdue. The problems are starting to add up and it could be having serious consequences on your business. It's time to start planning a refresh.

Let's Chat.

Next Steps:

Application refresh is something that most companies who are using mission critical software (including off-the-shelf solutions, solutions that have been customized and custom software) should be performing on a regular schedule.

Here are some tips for planning your next refresh project.

1. Conduct an Application Review

Host a user focus group to review UI, UX, functionality and identify gaps



Compile and categorize user feedback

Run performance and UX tests

Review accessibility and privacy compliance

Review design, navigation and branding elements

Identify any process changes that need to be reflected in the application

Identify new features and functionality needed or old ones that can be removed

2. Develop a Plan of Action

List and prioritize design and feature changes and additions

Keep in mind how any connected applications will be affected

Think about hardware and end-of-life deadlines for any integrated solutions

Set a timeline and deadlines for build, testing and deployment

Consider phasing in updates in order of priority for continuous improvement and less disruption

3. Establish a Budget

Meet with internal decision makers to present your plans and budget needed

Clearly outline the negative financial impact of your legacy applications and ROI of a refresh

Consider a plan that staggers updates to stay within budget constraints



4. Allocate Resources

Bring in internal and external expertise as needed to help ensure the project is executed correctly and securely

Assign a project manager to keep you on track

Ensure everyone is aware of how much of their time is required at different points in the project

5. Kick Off Project

Define and stick to a development process such as agile or waterfall

Use collaborative tracking tools such as DevOps

Schedule regular check-ins to track progress

Be flexible and anticipate scope changes

6. Build and Test

Ensure new features are performing as expected

Double check that data is flowing as it should within the app and between apps

Do another round of user testing to see if challenges have been addressed

7. Train

Host a training session for system managers and end-users

Allow some time for users to get familiar with the refreshed app in a contained environment before fully deploying



8. Deploy

Choose the best time to go live, like on a weekend when usage is minimal

Consider releasing updates in stages to minimize disruption

9. Review

Continue to perform regular checks and reviews

Refresh again as needed

[Back to Introduction](#)[Take the Rehost Survey](#)[Take the Rebuild/Replace Survey](#)

Is It Time to Rehost?

When businesses think about hosting, it's not uncommon to start talking about cloud services.

Cloud solutions have become increasingly popular for a wide range of reasons, but when looking at the hosting requirements of an application or solution, it's important to consider all the options including:

1. **Public or Private Cloud**
2. **On Premise**
3. **Hybrid Solutions**



“ By the end of 2020, 83% of the company workload will be stored on the cloud. - Forbes

There are pros to each solution, but which option is best for you will depend on your specific business needs, processes and operations. There's no denying that more and more companies are flocking to the cloud for easier management, the security benefits, and to transfer expenses from capital to operating budgets.

At the same time as companies are moving large parts of their workloads to the cloud, everything as a Service (XaaS) is gaining momentum. Businesses are looking to outsource IT services, software and even hardware and infrastructure so they can focus their efforts on customers, efficiency, productivity and business growth.

In a world of increasing pressure on IT departments to deliver business value, outsourcing services also allows companies to access the expertise they need as they need it, without the cost of an in-house team.

Common Rehost Signals:

How do you know you need to review and possibly update your hosting arrangements? Here are a few of the top reasons:

Aging Hardware

If the hardware supporting an application is more than 4 years old, you need to start looking at updates because maintenance time and costs will start to rise and uptime can start to get impacted.

Applications Slowing Down

As data loads increase, companies grow or the number of users is going up, applications can start to slow down if the hardware can't scale on demand, leaving users waiting (and frustrated).

End-of-Life

This is a major one: Scheduled end-of-life for server or database operating systems can leave your business very vulnerable and requires immediate action to maintain security.



Take Our Survey:

Yes No

Do you currently have on-premise servers?

Are you facing end-of-life for operating systems or databases?

Are your servers operating above 50% capacity, or older than 5 years?

Would you like to automate software updates and improve security?

Have your users noticed a slow down in application responsiveness?

Would you like to increase application uptime for business continuity and to more effectively handle business growth?

Does your traffic have peaks and valleys requiring on-demand scalability?

Would you like to offer flexible, remote work options to staff?

Has your business expanded or added new locations?

Do your users travel or require greater mobile access?

Would you like to reduce internal IT resources and/or IT operational costs?

Would you like to move IT expenses from capital to operating expenses?

Total Your Results

Assign 1 point for each Yes



Your Results:

If you answered Yes to Questions 2 and 3, capacity and performance issues (regardless of any other issues) escalate your need to act so you'll jump to the "Cloud is Calling" level, even if you've only scored a total of 2 points.

For all other results, multiple accessibility, performance and security issues will make discussions about hosting options for your application more pressing, including decisions on whether a cloud-based solution is right for you.

1-4 – You're on the Cusp. Cloud may have advantages but it might not require a move today or offer enough major business benefits in the immediate future. Monitor and re-evaluate every 3-6 months.

5-8 – Option Evaluation Phase. You have applications that could benefit from the move to a cloud solution. Start mapping out strategy and establish a timetable to migrate those tools.

9-12 – Cloud is Calling. Get help to make the move quickly and effectively.

We can help. Let's Chat.

“ Moving an older system isn't just about the cost of the hardware and software ... It will be expensive, but when you look at what the legacy system is costing you 2-3-4 years down the road, we have seen most of these able to pay themselves off in a 2-to-3-year period. - IDC



Next Steps:

The decision to rehost an application most often comes around when a business is looking holistically at its application objectives, and concerns or issues related to the hardware and infrastructure come to light during the application evaluation discussion.

It's important to remember the cloud is not just one thing, and it's not an all or nothing option. Many businesses have some applications and services in the cloud and others on premise. And even in these situations, the types of hybrid configurations can be as diverse and varied as the companies using them.

Here are some tips for planning your next rehost project.

1. Review Current Hosting

Document where all your applications are currently hosted

Identify end-of-life concerns

Document issues related to accessibility, mobility, scalability and performance

Meet with internal stakeholders to discuss changing business requirements and how the applications need to support them

Evaluate and select the hosting solution that's best for your needs

2. Develop a Plan

Map out how migration will be structured including which applications will be moved, how many users to move at a time and when

Clearly articulate the strategy and workplan to all stakeholders

Set a timeline and deadlines for migration, testing and deployment



3. Set A Budget

Outline current costs including maintenance and downtime

Outline the cost of the rehosting project and ongoing expenses

Consider budget implications of on-prem versus cloud which moves expenses from capital to operating

4. Allocate Resources

Bring in internal and external expertise as needed to help ensure the project is executed correctly and securely

Assign a project manager to keep you on track

Ensure everyone is aware of how much of their time is required at different points in the project

5. Clean Up Your Data

Look for duplication

Evaluate how your content is used, how often, and if you need to access it regularly

Identify content that can be archived

Review your information architecture and content structure

6. Prepare for Change

Create new cloud protocols such as governance, collaboration and file structure

Map out and communicate the new processes

Dedicate resources to training so everyone understands how to make the most of the new hosting environment



7. Migration Time

Decide on how you're going to migrate data from simple cut and paste to using a migration tool

Configure the cloud environment and the applications, including customizing security features before you start moving any data

Test for performance and security vulnerabilities before completing the transition and opening up the application to all users

Select a convenient time to complete the migration such as a weekend when usage is low

Consider migrating your content in chunks to minimize disruption

[Back to Introduction](#)[Take the Rebuild/Replace Survey](#)[Take the Refresh Survey](#)

Is It Time to Rebuild or Replace?

One of the hardest decisions for a company is to rebuild or replace a core piece of software. You've invested in its development and upgrades over the years and don't want to see that investment lost.



First a couple of definitions:

- **Rebuild:** While more intensive than a refresh, a rebuild describes projects that change the structure of an application, deploy major new features or functionality, add new languages or currencies, integrate new or upgraded data sources and/or connect to new external solutions. A rebuild commonly includes updating the coding language to a more modern version.
- **Replace:** Building a new solution from the ground up to take new business requirements into account, adding significant new features, combining multiple applications, or providing more seamless integration with other corporate systems. The good news is the old solution can serve as a valuable roadmap and guide to help speed the development of the new application.



“ In 2023, 90% of current applications will still be in use with insufficient modernization. - Gartner

Common Rebuild/Replace Signals:

Business or Process Change

Anytime your business changes or the processes you use in your business change, your applications need to keep pace to remain relevant and effective.

Expansion

If you're growing, you might find you need additional language capabilities or geography-specific elements (like tracking dollars and euros).

Not Creating Value

When an application is no longer doing what it should, is cumbersome to use, usage plummets and it's more work to maintain than it's worth, that's a big signal the time has come to rebuild or replace the application.



Take Our Survey:

Yes No

Is the existing application no longer supporting your business needs/processes?

Are there business processes you've identified for automation?

Have the workflows (or business processes) mapped by the existing application changed significantly?

Was the application a custom solution?

Does your application use an outdated (10+ years) programming language/framework?

Does the application need to communicate with another application or database?

Has it been more than 5 years since the last major update?

Have you bolted on, or added new connections to the application that extend functionality beyond the original solution in the past 5 years?

Have you experienced increasing performance issues?

Are there new features and functionality you want to add?

Have you expanded locations and now require multi-lingual capabilities?

Is your application costing you more than the value it is generating (either through revenue or increased efficiencies)?

Have management costs (internal and external) been increasing?

Do you want to combine multiple applications into one to reduce redundancy or data management?

Total Your Results

Assign 1 point for each Yes



Your Results

At the heart of every application is a business process you're trying to automate to gain productivity efficiencies. Over time, applications can stop delivering the improvements you want or your business changes so it's time for the application to change too.

Every situation is different and you could be facing drivers that accelerate the need to rebuild or replace to take advantage of modern features and functions. It can be a difficult decision to replace an application but it can be the best decision for your business over the long term. It's not a case of throwing out the old for something new – any new application takes advantage of the lessons learned and best practices built into the original solution.

Under 5 - Start Thinking. You're seeing some performance issues so it's time to start thinking about modernization strategies. A refresh or rehost might be sufficient depending on the issues but consider if long-term a rebuild or replace will be needed and start building your case.

5-10 - Rebuild. Getting the performance and features you want might just require one of the application layers to be rebuilt. Start the discussions to identify the options and alternatives.

10+ - Replace. Sometimes starting fresh will allow you to incorporate all the elements you want, and let you ensure connectivity between applications is seamless. Overlaying band-aid after band-aid can make a solution unwieldy and unstable, which can cause even more problems down the road.

Let's Chat.



“ Up to 80% of a company's IT budget is spent on keeping legacy systems afloat, when it could be used for new projects and purchases. - Future of Work

Next Steps

We know a major system rebuild or replacement is very disruptive to a business in the short-term. But bolting on new functionality to meet emerging needs, or constantly adding new features on top of an old or outdated code could set your company up for spiraling costs or a disastrous system failure in the long-term.

To help you get started, here are some tips for planning the next rebuild or replace project.

1. Review Existing Application

Host a focus group to see how users are currently utilizing the application

Document which features and functionalities they like and use, and which they don't

Make a list of what's missing based on current usage and changing business needs

Review all the layers of the application and identify critical structural issues

2. Define Requirements for Updated/New Application

What does the application need to do?

Identify and map the business processes involved

Identify who needs varying levels of access

Specify the technical requirements



3. Detail The UX/UI Needs

Core branding and design requirements

User journey and navigation

Backend admin or operational considerations

4. Define Integration Requirements

Other IT systems or software the application needs to communicate with

Consider if any applications can be combined for efficiency

Technical architecture at the application core

5. Review Hosting

Consider which solution is best for your needs: On-Prem, Cloud or a Hybrid

Consider remote access and scalability needs

Weigh security requirements

Outline maintenance resources and budgets

6. Present Your Case for a Rebuild or Replace

Define the key business objectives the application is supporting

Outline why a rebuild or replace is important to your business including ROI metrics

Be prepared for push back and prepare talking points to address core objections



7. Make a Plan Including Budget

Create a work back schedule and goal for final delivery

- Establish realistic timelines and deadlines
- Include adequate time for testing and training prior to deployment

Bring in internal and external expertise as needed to help ensure the project is executed correctly and securely

Assign a project manager to keep you on track

Ensure everyone is aware of how much of their time is required at different points in the project

Review development costs

Ensure you've allocated a contingency

Include costs for external resources, staff time and training

Consider the budget implications of your hosting decision (On-prem versus cloud solutions will move expenses from capital to operating)

8. Build and Test

Define and stick to a development process such as agile or waterfall

Use collaborative tracking tools such as DevOps

Schedule regular check-ins to track progress

Be flexible and anticipate scope changes



Ensure new features are performing as expected

Double check that data is flowing as it should within the app and between apps

Do another round of user testing to see if challenges have been addressed

9. Train

Host a training session for system managers and end-users

Allow some time for users to get familiar with the refreshed app in a contained environment before full deployment

10. Deploy

Choose the best time to go live, like on a weekend when usage is minimal

Consider releasing updates in stages to minimize disruption

[Back to Introduction](#)

[Take the Refresh Survey](#)

[Take the Rehost Survey](#)



Getting It Right – Conclusion



By now you should have a better idea of the current state of your application and the modernization strategy that might help address your issues.

But any major software project should begin with a comprehensive analysis and detailed action plan specific to your application and business needs.

It's critical to allocate time for a detailed discovery process and keep an open mind to what you might find. You may go into a project believing you need a quick refresh, but after diving through all the data and structures in your application it turns out it needs much more work. We've seen this before and it can come as a shock if you're not prepared.

For every software development project at Whitecap, we have a structured and very detailed process we call [e-RAD: Requirements Analysis and Design](#). This process helps articulate all the specifications and technical requirements to get the project right on track from the very start.

Regardless of the scope of the project, there are a few key steps every application modernization project needs to take:

1. Have a Plan

It might seem obvious but you need to make sure the executives, IT and business units are aligned and in agreement with the course of action, timing and budget. Whichever modernization strategy you propose, you need to back it up with measurable business outcomes.

2. Communicate Clearly & Often

Define expectations early to avoid miscommunication on requirements later in the project. Communicate throughout the project. Know who your key stakeholders are and involve them. Constant communication is key to understanding alignment and decision-making at every step of the effort.



3. Organize and Staff the Project

Whether you are implementing the changes in house or using an external expert, a strong project manager leading the project is critical to achieving a successful outcome. Making sure you have adequate resources and the right skillsets at your disposal can help the project reach a successful and timely completion. We've seen many projects go off the rails because they were improperly staffed, not enough time was allocated or they were prematurely pushed forward trying to hit deadlines.

Ready to modernize your business applications? Need help assessing your options?



We can help. Let's Chat.

About Whitecap

At Whitecap we have been designing and developing software that helps companies compete in the evolving digital world for over 20 years. We are a leading custom software, mobile app, Dynamics 365 CRM, SharePoint development and website design company in Toronto. We design and develop custom software solutions to help transform your business for the digital future. Learn more at whitecapcanada.com.

