Progress®

The Progress OpenEdge Application Evolution Methodology

EBOOK

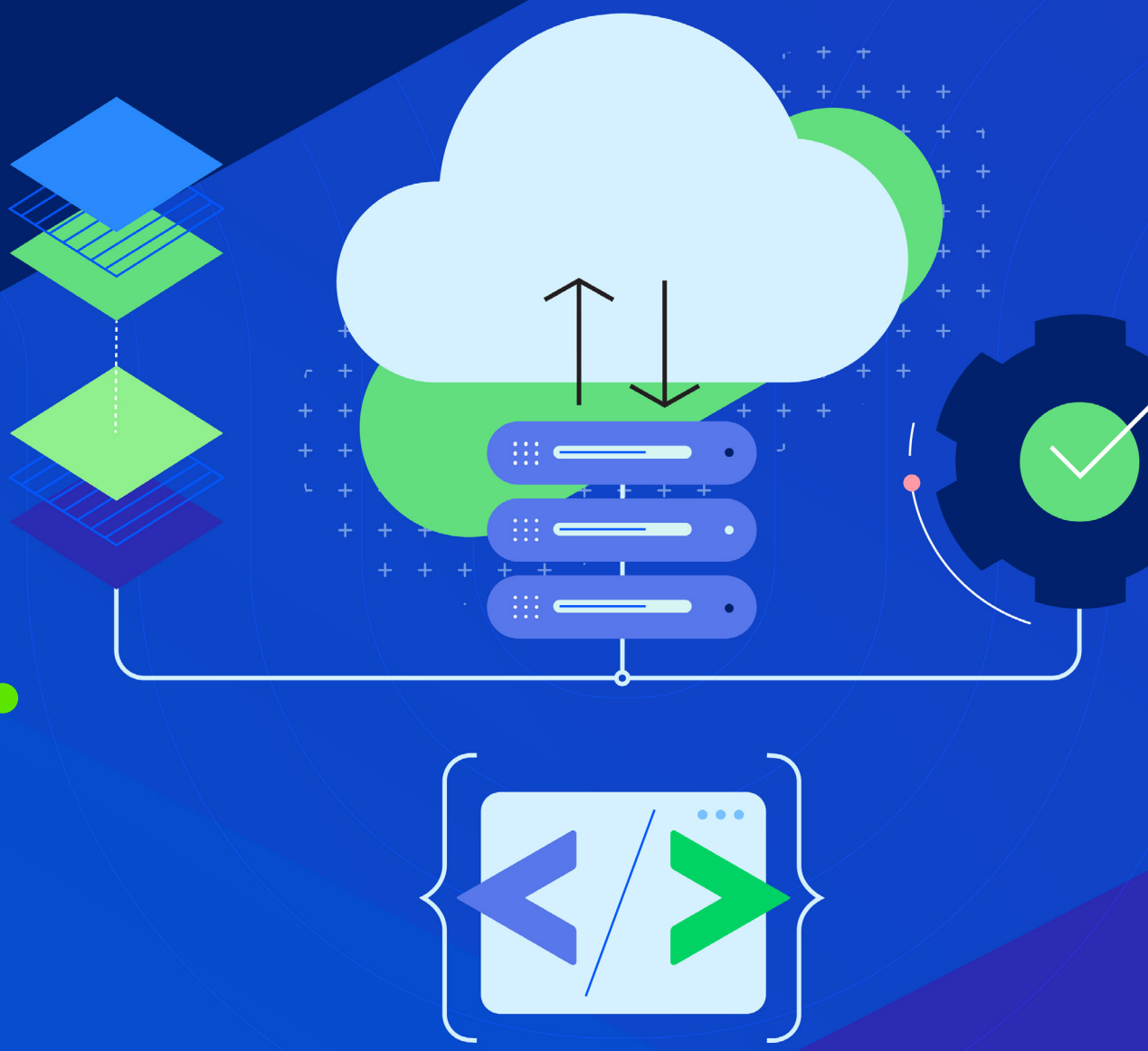


Table of Contents

Application Evolution with Progress Professional Services

Every organization looks for ways to accelerate time to market, grow market share and compete more effectively to protect that share. To increase operational efficiencies and decrease costs, it is necessary to keep pace with the industry standards and trends for their business applications while also looking for ways to outpace their competition.

Companies must adapt their technology to improve collaboration and connectivity and boost user adoption and brand loyalty. Meeting these goals requires an agile approach to application modernization. For users of Progress OpenEdge, the transition to the latest version can be aided by the Professional Services team.

Seven Application Modernization Essentials

Taking a character, GUI or early web application and migrating to a modern web-based platform is the most common example of OpenEdge application modernization. There are several ways to achieve this goal. While rip and replace is an option, most users choose to extend their application instead of completely replacing it or deliberately replacing it in phases based on business priorities.

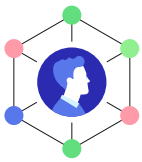
The latter options allow companies to retain the value created by and fine-tuned within their existing application. This way, the business rules differentiating a business from the competition remain in place and past work can be built upon rather than entirely recreated. These approaches lessen business risk and can be done in a way that minimally disrupts business operations.

When modernizing an application, there are seven considerations that will keep an application relevant, improve efficiency and increase customer satisfaction. A successful modernization will address all the following aspects.

Progress®

Extensibility and Integration

The ability to extend information and capabilities across many connections, devices and data sources.



Scalability

The ability to grow and scale as the number of users and connections increases without affecting performance.

Performance

The measurement of response time to act on a user's input or finish an amount of work under normal and peak loads.

High Availability

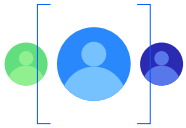
The measurement against a goal of continuous uptime and avoidance of data loss due to a disaster or breach.

Security

The ability of the organization to protect the system from unauthorized users while staying in compliance with regulatory mandates.

Organizational Agility

The organization's agile culture and ability to develop and deploy new application functionality quickly and productively.



Engaging Usability

The presentation layer's ability to provide intuitive and exciting experiences for users.

The Progress Professional Services Blueprint

When OpenEdge users are ready to modernize their application, they reach out to the experts on the Progress Professional Services team for guidance. We offer an approach to modernization that follows a three-part process but is customized to the needs of the individual client.

Modernization Assessment
First, we determine how Progress can facilitate the activities needed to modernize an application to achieve the desired business goals. Based on the needs of our clients, our approach is flexible enough to satisfy the specific needs of large and small customers, partners or direct end users.

Proof of Concept
Next, we develop a Proof of Concept (POC). The POC demonstrates that the prescribed approach will work and gives clients a representation of the final result. The POC Includes functionality that is well known within the business and utilizes real data.

Modernization Project
In the final step, Progress either takes responsibility for or assists with the analysis, design, development, testing and deployment of the functionality defined in the overall modernization project. The approach is determined by business drivers, priorities and budget.

Modernization Assessment

The Progress approach begins with an assessment. We include a suggested reference architecture for a modernized application in the cloud or on-premises and identify the organization's high-priority application essentials and business goals. We work with the client to identify an entry point into the recommended architecture and then define a roadmap to achieve goals specific to the environment and organization.

Needs and expectations for an application will change over time, so our approach to application evolution is incremental and iterative. Before beginning the three-step process, we walk our customers and partners through the various entry points to discover the best approach for their most pressing business and application goals. After creating the initial plan and establishing the entry point, the Progress team can be involved as much or as little as desired.

The Modernization Assessment typically takes two to three weeks. We begin with the preparation and definition of an agenda to create a Modernization Charter document. This requires involvement from Progress, customer stakeholders and subject matter experts.

During the Charter document creation, activities include:

- **Assessing existing roles and skills of the development team. This may include:**
  - Organization structure
  - Roles and responsibilities
  - Skills and competencies

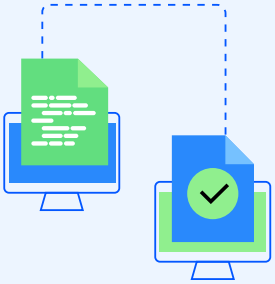
- **Conducting an inventory of current and future technology requirements such as:**
  - Data models
  - Entity relationships
  - Process/functional flows

- **Understanding the business model and assessing business requirements such as:**
  - Data models
  - Entity relationships
  - Process/functional flows

- **Determining the approach and scope of the modernization project. Areas covered include, but are not limited to:**
  - Security model
  - Business functionality
  - CI/CD practices
  - Architecture
  - Target platform
  - Data model
  - User interface (standards, design, types of users)
  - Integration with other systems
  - Business logic processing

Proof of Concept

A proof of concept (POC), also described as a pilot, is an early version of production software targeting a limited scope of the final solution and a specific user persona. It is used to validate the architectural design, user interaction and deployment strategy to provide early feedback into the final product. Work performed in a POC involves an in-depth study of a customer's platform and follows a recommended architectural design. A POC may include training, coaching and knowledge transfer. The outcome should reflect the full development lifecycle of the client's final solution.



A POC is typically completed within eight to ten weeks. We begin with target functionality that is highly visible but not overly aggressive to keep changes incremental. When harvesting code, the legacy application must be analyzed to determine what, if any, business logic must be re-packaged and the best process to complete the harvesting. Candidates for code harvesting are key business logic components that differentiate the business. This logic may have been developed over many years and there may be a strong desire to use the same algorithms.

Once the code is identified, it must be re-packaged in such a way that it can be called from the modernized application. This usually involves removing the user interface dependencies in such that the code module (method) can deliver and consume temp tables and ProDataSets. A project manager and architect create a project plan that identifies tasks, owners, milestones, effort and dates.

During code review and assessment, we:

- **Identify code that pertains to the POC**
- **Evaluate business logic within the applicable procedures to identify patterns in code where harvesting is desired**
- **Identify and document any gaps between existing logic and logic desired during the iteration**
- **Identify procedures that are no longer needed because of framework functionality, such as CRUD, Security or Managers**
- **Establish patterns for posting and packaging result sets**

The next step of the POC is to configure the REST adaptor and modernization framework. Eventually, this may need to be done in multiple environments: development, test and production. Completing all three environments during this phase is not in scope. Once the modernized OpenEdge environment is installed, testing is conducted to ensure that all the architectural components are working correctly. Depending on the customer's requirements, IT staff may be trained in the components of the architecture at this time.

In some cases, the client may decide to take ownership of delivering and packaging the server-side business logic. In these scenarios, there will be some mentoring and coaching to define a best practice approach so that all resources are working together as efficiently as possible.

In the User Interface/User Experience (UI/UX) phase, we develop all UI artifacts described by the high and low fidelity wireframes. In many cases, clients have already chosen a specific JS Tool Kit as a company standard. As in previous steps, based on patterns that are identifiable during the UI/UX discovery, user interface templates can be created and reused throughout the application. Additionally, there are tools available that can build web and mobile UIs to facilitate the creation of modernized experiences.

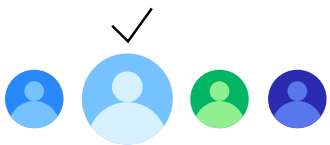
Application Programming Interface (API) Development begins with an understanding of the API requirements as defined by the UI/UX phase. During this step, API Developers and UI Developers must agree upon API contracts and URL routes. Developers must also agree upon API payloads and supporting structure. Also considered would be API to support identity/domain assertion as well as authorization. Documenting APIs and contracts is critical. When done correctly, client-side and server-side developers can work independently of each other.

The construction phase commences based on the code assessment, customer skills and business need. We use the Progress API Viewer to enable client-side developers and server-side developers to work independently of each other. This catalog ensures that the appropriate method calls are made with the appropriate signature, and all team members define and understand the required results.

After this step, we identify, harvest and refactor existing ABL code and develop any new or extended logic that is needed to support the new UI. For example, based upon patterns that have been identified, we can create automated processes for CRUD operations. In other cases, one could automate the transformation of legacy codebase to a more modernized object-oriented structure. These practices and tools enable increased agility and efficiency of development.

The POC phase ends with the delivery of a Readiness Plan. This is a written modernization plan, which includes:

- **An architectural design of the proposed solution and recommended deployment architecture**
- **Documented process and patterns that can be used for the rest of the evolution effort**
- **A comprehensive curriculum of on-demand and instructor-led training courses for knowledge enhancement**



Our clients' success is our goal, and the best outcome is knowing the client has access to new application possibilities such as an omni-channel user experience, a successful shift to the cloud or new integrations and functionality.

Modernization Project

Some clients choose to engage in a knowledge transfer with the Progress Professional Services team at the end of the POC step so that they can continue the modernization project on their own. The amount, type and timeliness of knowledge transfer are contingent on the skills of the client's staff and their long-term development goals. For these clients, a detailed plan for knowledge transfer is crucial so the IT staff can continue future iterations interdependently and with confidence.

For clients continuing the third step, the Modernization Project, in collaboration with Progress, we begin a process of iteration where the project team learns more and adjusts with each iteration. If mistakes are made along the way, it is necessary to learn from them to become more productive as a result. Each iteration has familiarity with the reusable user interface templates. Similarly, identifying code patterns on the server enables more productivity during business logic development.

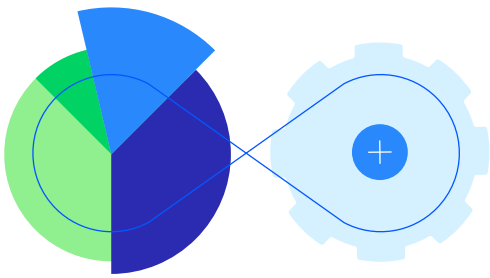
As with the POC, in early iterations, the preference is to target highly visual and not extremely complex functionality. We focus on creating graphically appealing interfaces to provide a compelling visual example of what the modernized application will look like. This has proven to generate excitement and ensure momentum with stakeholders early in the process. As subsequent iterations of the process take place, more patterns are identified and created which will be beneficial as more complex functionality is modernized.

It should be noted that major changes in workflow may impact the reusability of existing business logic. Evolving an application is a fantastic opportunity to make the user experience more productive and intuitive, but if time to market is important, workflow changes must be done thoughtfully.

Every iteration of the Modernization Project focuses on continuous integration and continuous delivery (CI/CD). CI is the practice of merging all developer working copies to a shared mainline several times a day, with the goal of integrating development teams' work early and often to avoid messy and painful "mega merges" later. We focus on the following best practices:

- **Maintain a code repository**
- **Automate the build**
- **Make the build self-testing**
- **Everyone commits to the baseline every day**
- **Every commit to baseline should be built**
- **Keep the build fast**
- **Test in a clone of the production environment**

CD and DevOps are both an approach and a set of practices intended to shorten the time between a change being made and that change being deployed into production. They're very similar, but CD focuses on the automation of the software delivery process and DevOps on the changes to the business/organizational processes. We focus on the definition of processes, quality gates and deployment infrastructure (e.g., containers, virtual machines, etc.) for production, staging, testing and development.



The Professional Services team collaborates with the client for as long as they desire, allowing the client to determine when the Modernization Project work is complete. Modernization is an ongoing process, but after the three phases clients feel equipped to continue the application modernization journey on their own. If assistance is needed later, we can continue to participate in the process with coaching, validation and resolution of any outstanding challenges. Our clients' success is our goal, and the best outcome is knowing the client has access to new application possibilities such as an omni-channel user experience, a successful shift to the cloud or new integrations and functionality.

Summary

Successful OpenEdge application modernization projects require an approach that allows the client to take steps that make each future iteration even easier than the last. By creating an API-first strategy, the Progress Application Evolution Methodology allows OpenEdge customers, partners and end users the ability to reuse existing application logic while building the architecture and technology that best suits the future objectives of the business.

A CI/CD approach to modernization allows users the opportunity to experience new features and enhancements that are more responsive, more available and more performant than before. Over time, as business and user requirements change, the application evolution process remains flexible enough to accommodate those changes and deliver a superior user experience.

When modernization projects are led by the Progress Professional Services team, clients have peace of mind because experts are there every step of the way. The Progress OpenEdge Application Evolution Methodology ensures that OpenEdge technology continues to be adaptable and provide high value to organizations and their users now and into the future.

To learn more about working with the Progress Professional Services team or improving business applications with the OpenEdge Application Modernization Methodology, contact us at https://www.progress.com/consulting.

Learn more about OpenEdge Application Modernization Methodology

About Progress

Dedicated to propelling business forward in a technology-driven world, Progress (NASDAQ: PRGS) helps businesses drive faster cycles of innovation, fuel momentum and accelerate their path to success. As the trusted provider of the best products to develop, deploy and manage high-impact applications, Progress enables customers to build the applications and experiences they need, deploy where and how they want and manage it all safely and securely. Hundreds of thousands of enterprises, including 1,700 software companies and 3.5 million developers, depend on Progress to achieve their goals—with confidence. Learn more at www.progress.com

Rev 2023/07 RITM0208068

Worldwide Headquarters

Progress Software Corporation
15 Wayside Rd, Suite 400, Burlington, MA01803, USA
Tel: +1-800-477-6473

facebook.com/progresssw
twitter.com/progresssw
youtube.com/progresssw
linkedin.com/company/progress-software
progress_sw_

Progress