Progress®



Learn about OpenEdge

OpenEdge®

Copyright

© 2021 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

These materials and all Progress® software products are copyrighted and all rights are reserved by Progress Software Corporation. The information in these materials is subject to change without notice, and Progress Software Corporation assumes no responsibility for any errors that may appear therein. The references in these materials to specific platforms supported are subject to change.

DataDirect Cloud, DataDirect Connect, DataDirect Connect64, DataDirect XML Converters, DataDirect XQuery, DataRPM, Defrag This, Deliver More Than Expected, DevReach (and design), Icenium, Inspec, Ipswitch, iMacros, Kendo UI, Kinvey, MessageWay, MOVEit, NativeChat, NativeScript, OpenEdge, Powered by Chef, Powered by Progress, Progress, Progress Software Developers Network, SequeLink, Sitefinity (and Design), Sitefinity, Sitefinity (and design), SpeedScript, Stylus Studio, Stylized Design (Arrow/3D Box logo), Styleized Design (C Chef logo), Stylized Design of Samurai, TeamPulse, Telerik, Telerik (and design), Test Studio, WebSpeed, WhatsConfigured, WhatsConnected, WhatsUp, and WS_FTP are registered trademarks of Progress Software Corporation or one of its affiliates or subsidiaries in the U.S. and/or other countries.

Analytics360, AppServer, BusinessEdge, Chef Automate, Chef Compliance, Chef Desktop, Chef Habitat, Chef WorkStation, Corticon.js, Corticon Rules, Data Access, DataDirect Autonomous REST Connector, DataDirect Spy, DevCraft, Fiddler, Fiddler Everywhere, FiddlerCap, FiddlerCore, FiddlerScript, Hybrid Data Pipeline, iMail, JustAssembly, JustDecompile, JustMock, KendoReact, NativeScript Sidekick, OpenAccess, PASOE, Pro2, ProDataSet, Progress Results, Progress Software, ProVision, PSE Pro, Push Jobs, SafeSpaceVR, Sitefinity Cloud, Sitefinity CMS, Sitefinity Digital Experience Cloud, Sitefinity Feather, Sitefinity Insight, Sitefinity Thunder, SmartBrowser, SmartComponent, SmartDataBrowser, SmartDataObjects, SmartDataView, SmartDialog, SmartFolder, SmartFrame, SmartObjects, SmartPanel, SmartQuery, SmartViewer, SmartWindow, Supermarket, SupportLink, Unite UX, and WebClient are trademarks or service marks of Progress Software Corporation and/or its subsidiaries or affiliates in the U.S. and other countries. Java is a registered trademark of Oracle and/or its affiliates. Any other marks contained herein may be trademarks of their respective owners.

July 2021

Product version: Progress OpenEdge 12.4

Updated: 2021/06/29

Table of Contents

Identify your business application architecture.....9

Build your application architecture.....11

Develop your applications.....11

Provide reliable data.....12

Code with a powerful language.....13

Use integrated tools.....14

Supply a standard data format.....14

Design a custom user experience.....15

Use external business rules.....16

Enforce process management.....18

Deploy your applications.....18

Manage your applications.....20

Integrate your applications with external systems.....23

Access external data.....24

Expose data to external applications.....24

Call external REST services to access logic and data.....25

Communicate with JMS messages.....25

Preface

Purpose

The purpose of this content is to introduce OpenEdge products and architecture to new developers and application architects.

Audience

The audience is those new to the OpenEdge tools and technology. The content provides an overview of a modern OpenEdge application architecture and provides links to videos and guided user journeys for different developers,administrators and DevOps engineers to explore.

Organization

The organization of this content is to provide a high level view of an OpenEdge application architecture and provide links to additional resources where developers, administrators and DevOps engineers can explore concepts in more detail.

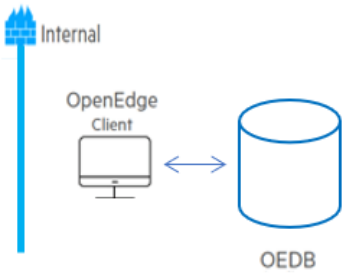
Documentation Conventions

See Documentation Conventions for an explanation of the terminology, format, and typographical conventions used throughout the OpenEdge content.

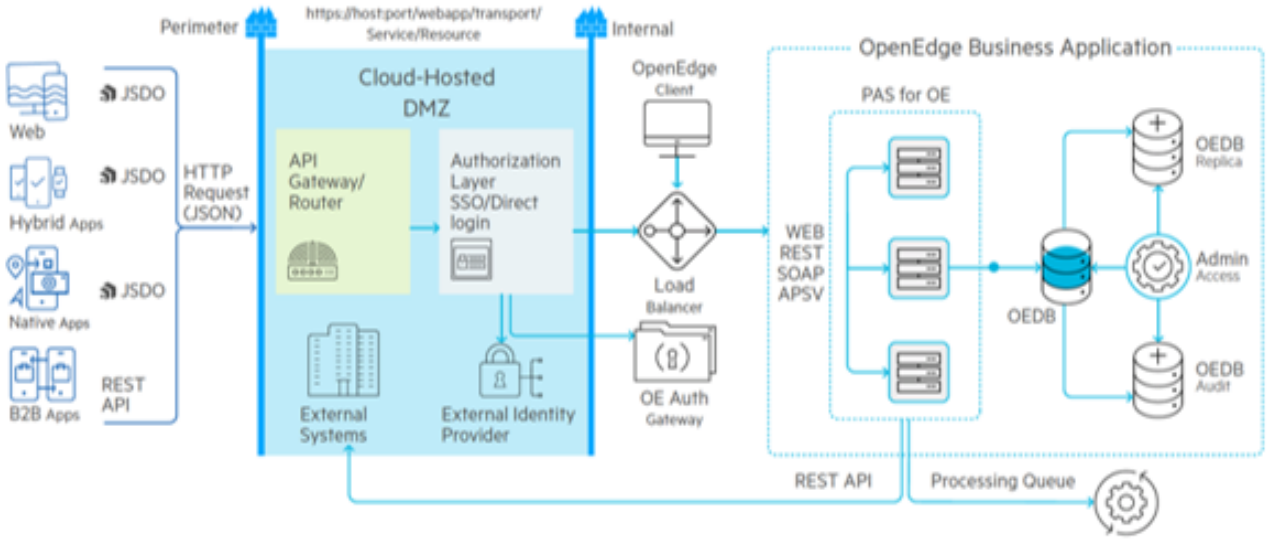
1

Identify your business application architecture

Early business application architecture was simple. Customers directly connected to a database behind the safety of the corporate firewall. Customers didn't have to worry about exposing their business applications to security threats outside of the office walls. There were very few user interface choices and the load on the system was limited to the employees and customers with on-site access.



Today's applications exist in a more diverse ecosystem. A modern application architecture is more complex like the one shown here.



Customers want the freedom to connect with any device. User are often authenticated through external identity providers. Authorization is required to access confidential data. Load balancers distribute thousands of client requests routed to large scale servers that connect to the back-end business services using an API-first strategy where the same business logic is available to all client types using a standard call signature. Database management systems have significantly evolved to perform under these new demands through table partitioning and multi-tenancy keeping systems running 24 x 7 with replication, online maintenance, and online backups. In an era of greater than ever compliance requirements, auditing helps to meet those requirements. This content introduces new users to the Progress products and tools that support development in this evolving application architecture.

2

Build your application architecture

The key factors for application success are ***high availability, performance,security, extensibility, and a modern user experience***. As always, Progress continues to evolve and to provide a development platform and tools to build applications for this changing ecosystem. Our customers are our partners. Together we deliver applications that work with flexible user interface choices that securely connect to easy-to-code business services accessing secure, reliable data. In other words, with Progress you build applications that continue to "just work".

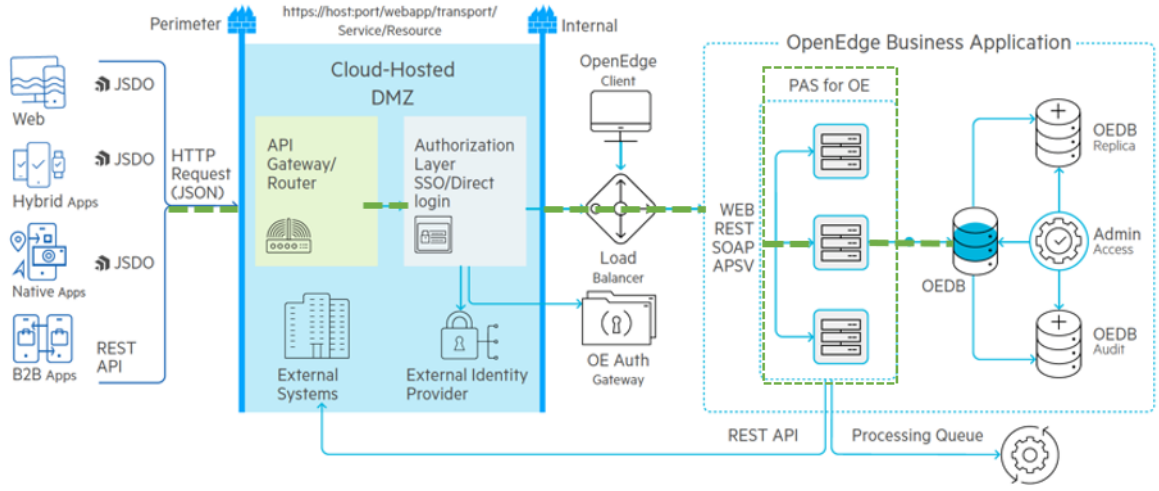
Progress has the tools you need to build applications that thrive in this ever-changing world. At the heart of all applications is the business logic interacting with data. The following topics explore how Progress helps you develop, deploy, and manage applications.

For details, see the following topics:

- Develop your applications
- Deploy your applications
- Manage your applications

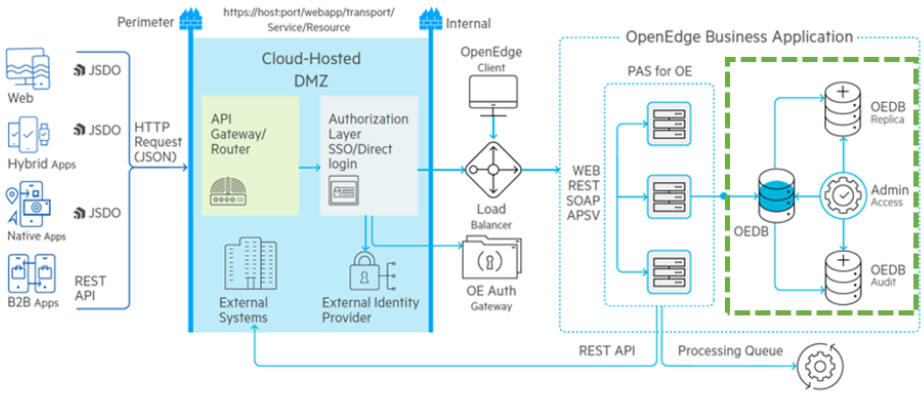
Develop your applications

The architecture for a cloud business application is broken into three major components: the user interface, the DMZ and the OpenEdge servers as shown in this diagram:



Provide reliable data

Reliable data is central to business applications. The OpenEdge database is a Relational Database Management System (RDBMS) which is capable of handling thousands of users and high-traffic loads with sub-second response times. It ensures users are productive and customers are satisfied while protecting the data security and reliability.



Key benefits of the OpenEdge RDBMS include cost-effectiveness, easy to administer, scalable, and easy to manage. Applications may take advantage or one or all the following database features.

- ***Security*** – To authenticate and authorize users of the application. Database Administrators can lock the database schema and data forcing all request to provide authentication and authorization credentials using the OpenEdge Authentication Gateway that can be integrated with external identity providers to offer single-sign on for application users.
- ***Transparent Data Encryption (TDE)*** – To secure access to the data. OpenEdge TDE balances both security and performance needs in a complete out-of-the-box solution, using standard encryption libraries and encryption key management for secure, encrypted data. Protect as much as you want, down to a single table, index or field. Execute at full speed with less than 2% degradation while encrypting/decrypting. TDE includes both policy tools and a secure encryption keystore.
- ***Replication*** – To improve uptime and protect against data loss due to a breach or disaster. Replication is the process of duplicating a source database to one or more target databases that are running on a separate

machine. This duplication keeps an identical copy while also providing a hot standby in case a database fails. If a database does fail, a replica becomes active, ensuring that mission-critical data is available to your users 24 hours a day, seven days a week.

- ***Multi-tenancy*** – To ensure privacy between tenant data. Multi-tenancy is a database architecture principle where a single instance of database runs on a server that serves multiple client organizations (tenants). Database administrators centralize database management tasks on a single database rather than separately for multiple tenants.
- ***Table partitioning*** – To ease maintenance and increase performance. By partitioning database tables into smaller, more manageable units, database administrators use a "divide and conquer" approach to data management and improve access times.
- ***Auditing*** – To support compliance regulations, auditing provides nonrepudiation of audit data. You can see who did what, where, when, and how. Auditing is a means of securely tracking and recording a trail of events that occur during execution of an application.
- ***Change Data Capture*** – To ensure data accuracy across the enterprise CDC allows you to define change data capture policies that are created and activated at the table and field level to scale with your business needs. The process is driven by CDC triggers which execute on Create, Update and Delete (CUD) operations.
- ***OpenEdge Pro2*** - To provide real-time data replication for reporting and analytics purposes, Pro2 provides easy, fast data replication from an OpenEdge database into a separate OpenEdge, SQL Server or Oracle database. Gain quick and easy access to mission-critical data from your OpenEdge system, without disrupting normal business operations or risking transactional database stability.

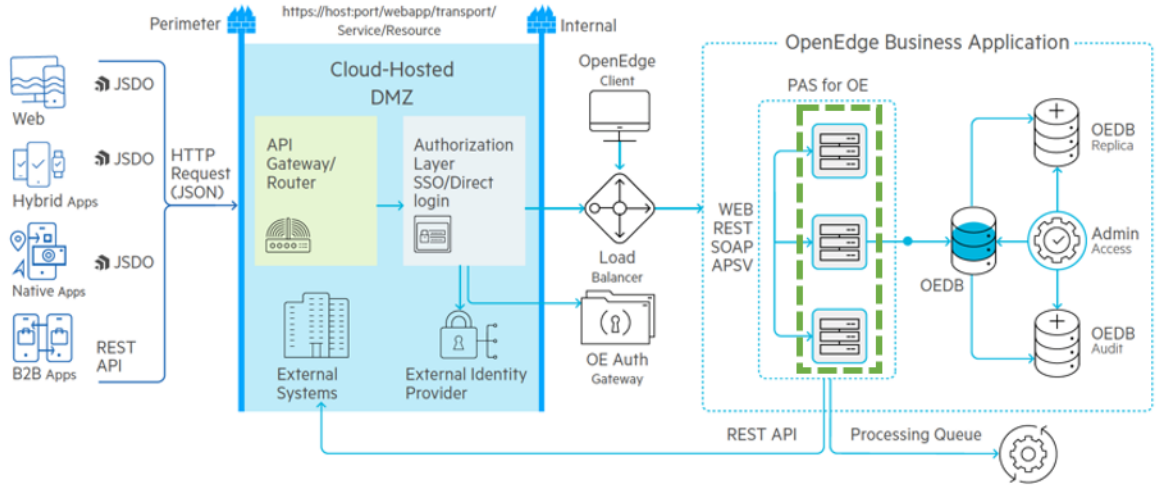
Explore the database

Database administrators are the primary architects, implementors, and managers of the OpenEdge RDBMS. The following role-based learning track offers more details on the database.

- Database Administrator - Explore the content specific to the Database Administrator role.

Code with a powerful language

ABL (Advanced Business Language) is an advanced programming language that is interpreted, object-oriented and built on robust transactional data semantics. It supports both procedural and object-oriented programming paradigms that includes statements and keywords which are specialized for building enterprise business applications and working directly with the underlying database.



The language is well designed for an Application Programmers Interface (API) first strategy. Exposing business functionality across the internet through Representational State Transfer (REST) calls using a uniform and predefined set of stateless operations provides the greatest flexibility of client types.

The key benefits of ABL are easy to code, efficiently manages data, built-in error handling and secure commits optimized to work with the OpenEdge database. Use the language to develop services, event procedures and implement service-level authorization for secure applications.

Explore the ABL

ABL developers design and deliver business applications that are extensible, secure, performant, scalable, and available. Each of the following role-based learning tracks offers more details on the ABL.

- Basic ABL Developer – Explore the basics of ABL.
- Object-oriented ABL Developer – Explore the object-oriented ABL syntax and examples.
- Services Developer – Explore how to create ABL services, event procedures, and implement service-level authorization to be deployed on PAS for OpenEdge.

Use integrated tools

Progress Developer Studio for OpenEdge consists of a set of editors, views, dialogs, and wizards that run in the Eclipse framework. Because Eclipse is an open-source, plug-in based framework, you can integrate a wide variety of software tools into the Eclipse installation that supports Progress Developer Studio for OpenEdge. Use Developer studio to:

- Quickly build user-interface and business logic
- Incorporate and manage application data sources
- Run, debug, and deploy code

Developer Studio includes a developer license for the Progress Application Server for OpenEdge to deploy, test and debug API services before securing and exposing those services in a production instance.

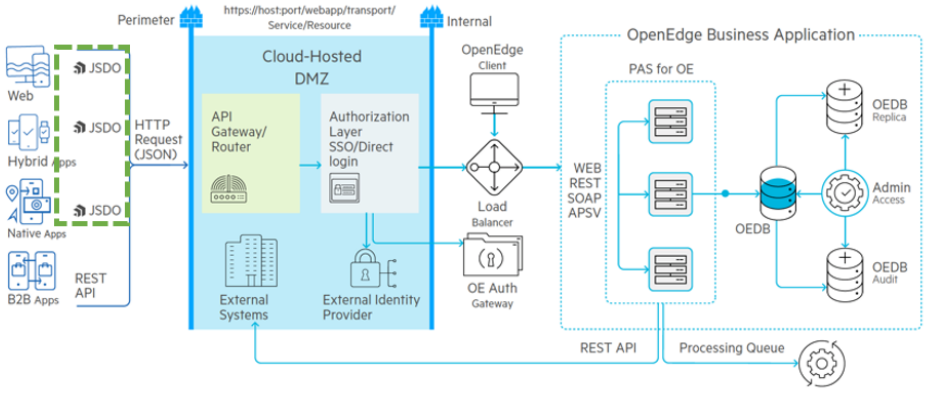
Explore Progress Developer Studio

In the following videos, learn about the integrated development environment that provides syntax checking, direct connections to a database for schema access and a development version of the Progress Application Server for OpenEdge for deploying and testing application code.

- Progress Developer Studio: Get Started video - Part 1
- Progress Developer Studio: Get Started video - Part 2

Supply a standard data format

Progress Data Objects define resources in ABL that can be used both within an OpenEdge application and from external clients using standard web technologies. This allows a client application running on the Internet to access data resources from a well-defined API using Data Object Services made available via web servers. Data Object Services employ web technologies to support a broad range of data access options, depending on the resource, from read/write access of simple sets of data elements to direct invocation of business logic on the server that can control complex transactions over that data. The supported clients include both OpenEdge client UIs built in ABL and mobile and web apps built using JavaScript, HTML5, and jQuery-based UI libraries, depending on the tools you use to build these apps.



Progress Data Objects can be used in desktop, mobile and web apps.

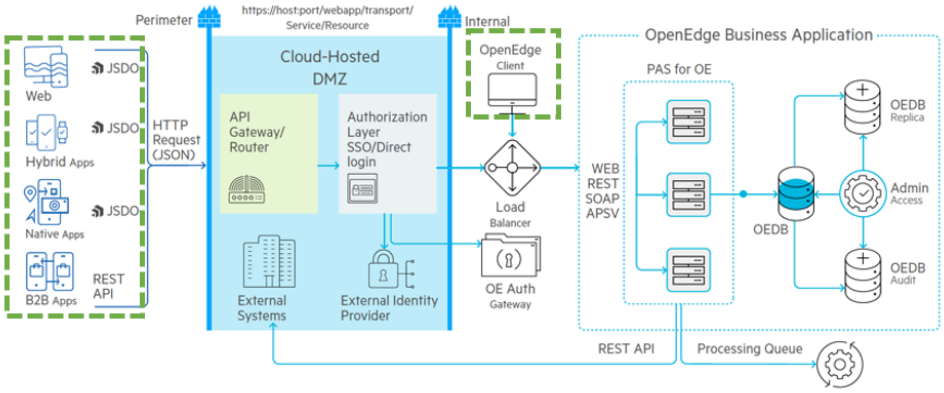
Explore JSDO

Developing with the JSDO offers more freedom to choose the user-experience that best matches your application needs. The following role-based learning track offers more details on using JSDO.

- Web-UI Developer Explore how the Progress Data Objects represents the Progress-supported implementation of the Cloud Data Object, which is an open source project on GitHub.

Design a custom user experience

OpenEdge customer experiences are varied as their industries. Traditional OpenEdge clients exist behind the fire wall. Outside the firewall Web, Hybrid Apps, Native Apps and B2B Apps use Javascript Data Object (JSDO) and Representational State Transfer (REST) APIs to access the business logic and reliable data on the other side of the firewall.



Explore Web Applications

Progress provides the following web UI options for building customer user experiences that interact with the back-end business logic and data.

- NativeScript and Progress Kinvey Explore native cross/platform mobile and web apps with beautiful UX faster and with less code reuse.
- Progress Services UX Design Services Consult the expertise you need to craft the ideal user experience work with experts.

- Progress Sitefinity Explore delivering rich, personalized user experiences across all digital channels

Additionally, any web UI technology that can work with REST services is supported.

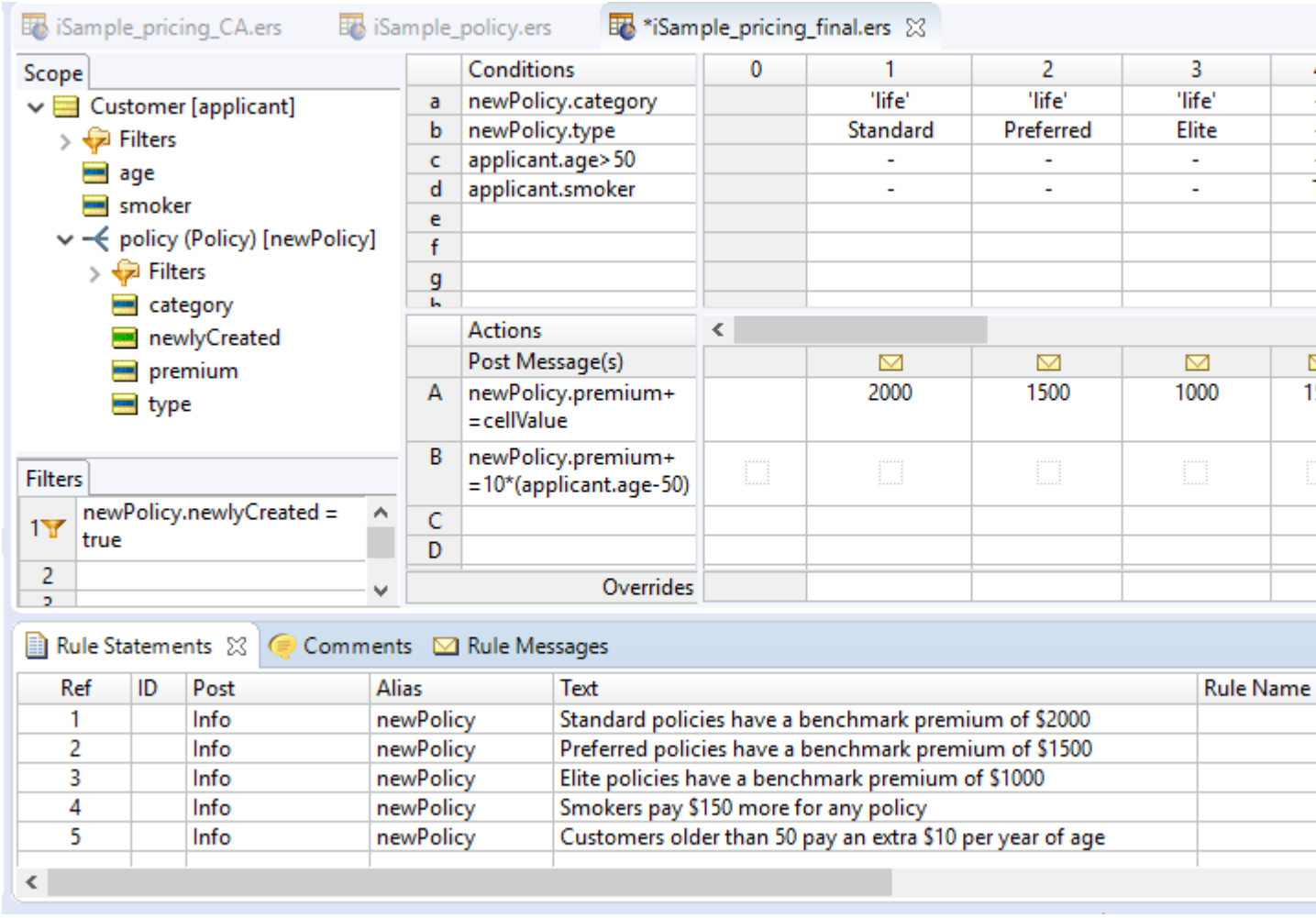
Explore OpenEdge clients

OpenEdge client applications continue to be integral to the success of our customers and partners and include

- **ABL Client** – Provides default client user interface in Windows (GUI) or in Linux/Unix (Character) representation of user interface widgets.
- **GUI for .NET** - Provides extensions to the OpenEdge GUI client, to support the ability to access Microsoft .NET objects from ABL as if they were native ABL objects.
- **Batch clients** - Runs reports and/or performs administrative tasks in the background. This type of client has no user interaction.
- **Open Client** – Runs in a client process that is not an AVM. Open Clients can be written in C#, VB.NET, or Java, and they can present a variety of user interfaces to the end-user. They also provide access to ABL business logic to other applications written in the supported languages that do not have a user interface.
- **REST and SOAP Web service clients** - Clients written in a variety of languages (ABL, Java, C#, VB.NET, HTML) can access an ABL application's services if the application has been built to provide Web services. An ABL Web services client runs in its own AVM. Other Web services clients run in client processes that run in a DLL or JVM.
- **WebSpeed client** - A WebSpeed client runs in a Web browser and is written in HTML and/or JavaScript. It accesses services by using HTTP or HTTPS to connect to OpenEdge applications.

Use external business rules

Corticon automates decision making logic with agility and business control, helping organizations make better, smarter and faster decisions. The rules engine is used to separate business logic from application code in order to accommodate faster, and business driven, change.



Corticon provides an intuitive, spreadsheet-like interface to define and manage the decision logic as business rules (rather than programming code), and automates the logic with performance and scale. It can also compress your initial development of business logic by as much as 10x and maintenance cycles by 25x. And, it provides higher quality business logic by catching errors in the design and requirements phase.

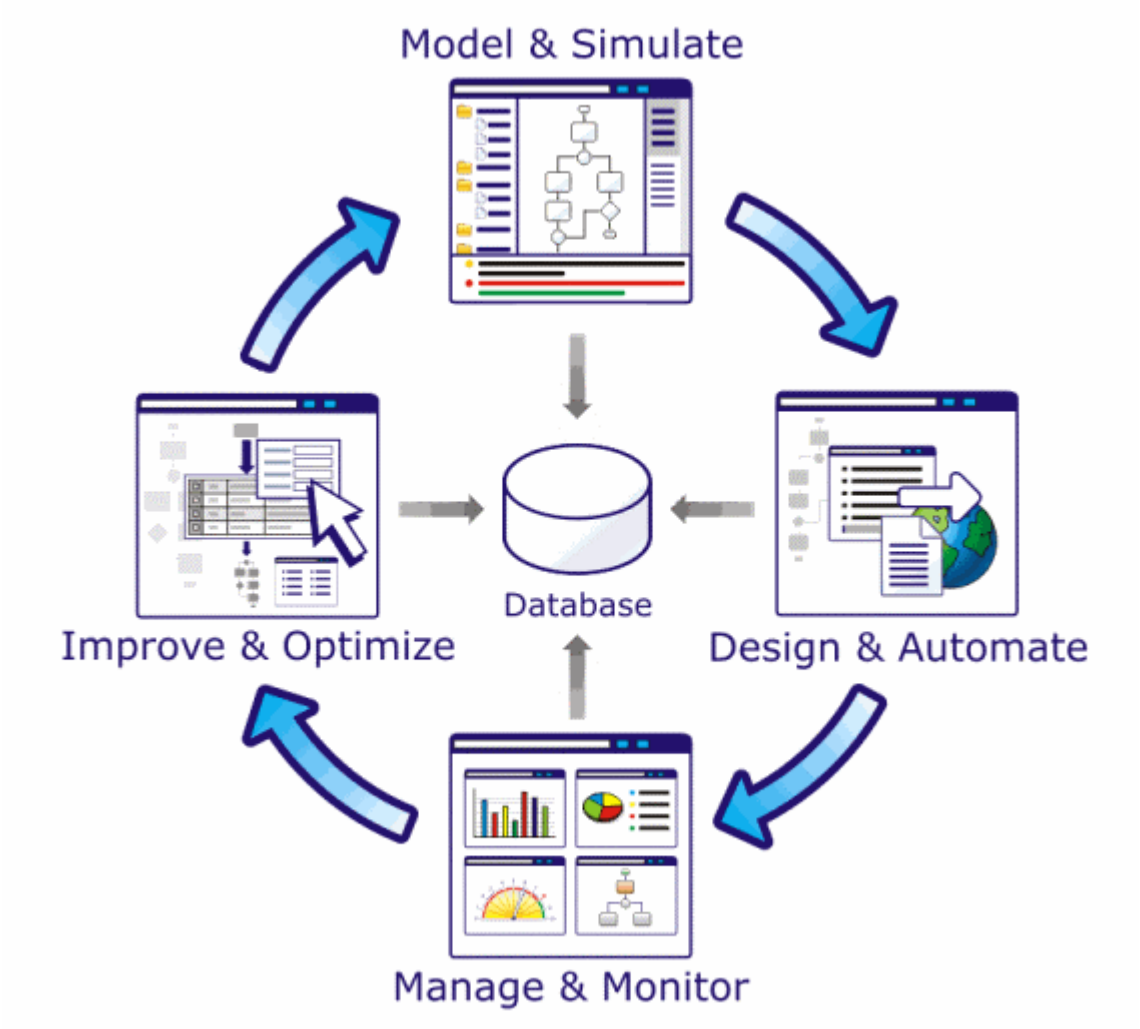
Explore Corticon

The following content offers more information on how to implement rules with Corticon:

- Watch the following videos:
  - Learn about rule modeling
  - Learn about Corticon.js.
- Try it.
  - Explore rule modeling concepts, build a vocabulary, define rules, test rules, enhance rules, and combine rules into ruleflows with the Get started with Rule modeling developer content.
  - Get hands-on experience with the basic and advanced tutorials.

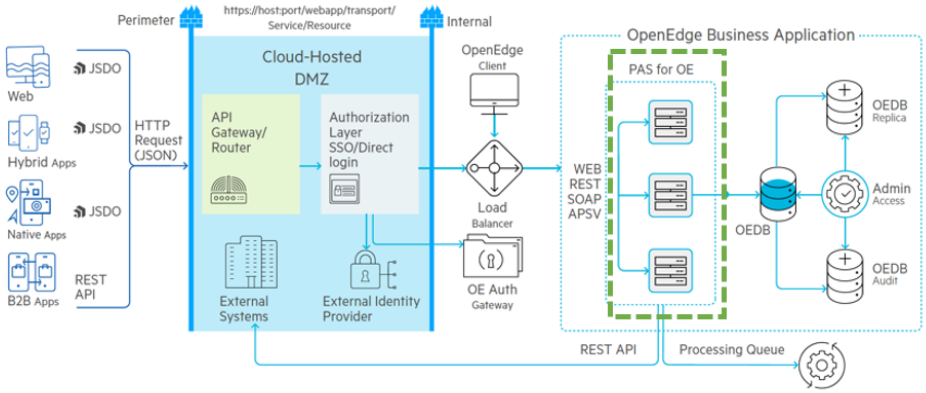
Enforce process management

OpenEdge Business Process Management's, Business Process Server addresses every stage in the business life cycle: define, integrate, publish, monitor, analyze, improve, and control. By adopting an end-to-end approach, Business Process Server incorporates all the key elements required to meet the ever-changing demands of e-business while ensuring e-business success. By providing integrated management tools, Business Process Server lets you monitor operations proactively, modifying automated processes dynamically based on changing external operations online. An overview of a typical automated business process management solutions is shown here.



Deploy your applications

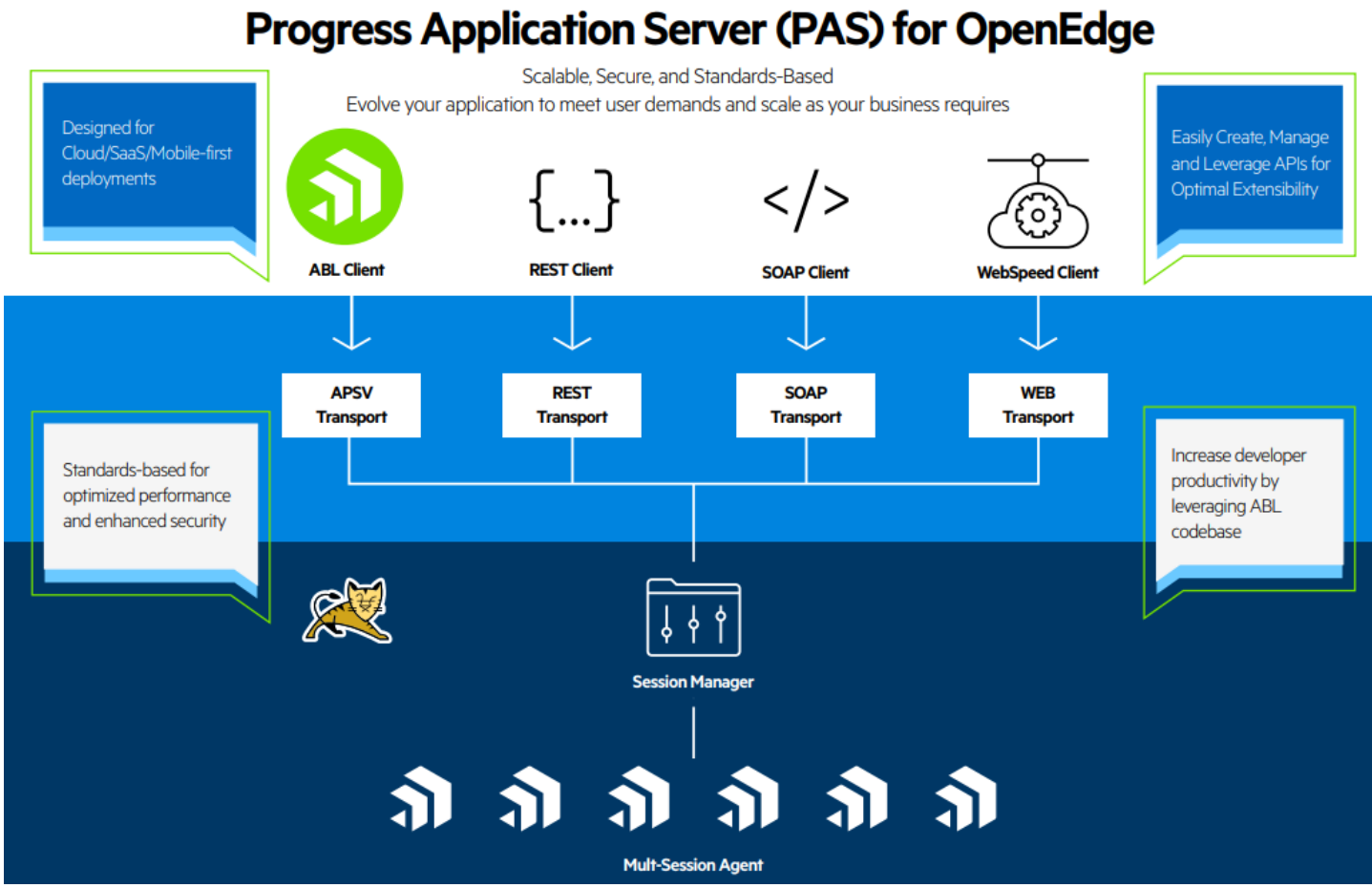
The Progress Application Server (PAS) for OpenEdge is a high performance web application server based on Apache Tomcat. It is tailored specifically to support OpenEdge applications.



PAS for OpenEdge is available as two separate products:

- The ***Progress Development Application Server for OpenEdge*** — Included in Progress Developer Studio for OpenEdge license to allow you deploy, test, and debug business logic on the server.
- The ***Progress Production Application Server for OpenEdge*** — Designed for production staging test environments and full production systems. System administrators and DevOps engineers have the ability to deploy, monitor and manage applications using OpenEdge Management, command line utilities and/or APIs.

All client types send requests to your ABL server code using **one of four transports** shown here:



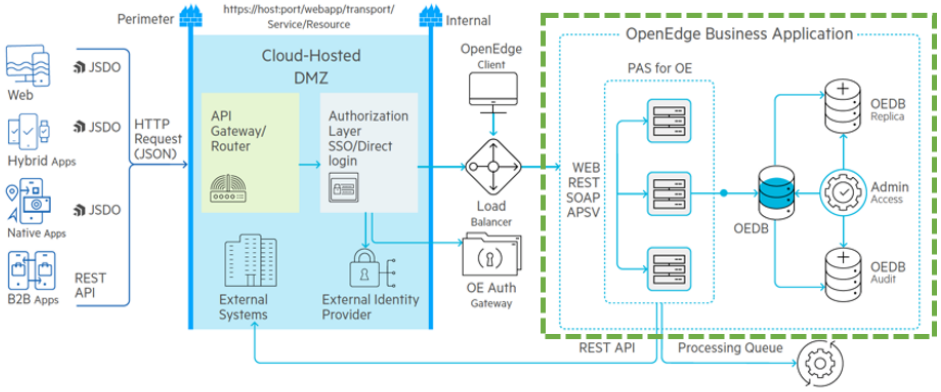
Explore PAS for OpenEdge

While all roles interact with PAS for OpenEdge. The following roles have the most interaction with PAS for OpenEdge. The following role-based learning track offers more details on using PAS for OpenEdge.

- Service Developer Explore how the service developer develops service to be deployed to PAS for OpenEdge.
- System Administrator Explore how the system administrators install, configure, manage and monitor production instances.
- DevOps Engineers Explore how the DevOps engineers interact with PAS for OpenEdge in supporting the CI/CD (Continuous Integration,/Continuous deployment) pipeline.

Manage your applications

OpenEdge Management is a system management center that provides visibility, analysis, and proactive monitoring of critical information assets. OpenEdge Management optimizes the availability and performance of OpenEdge applications through system monitoring, alerting, and automatic handling of corrective actions. OpenEdge Management empowers you to become more efficient, decrease the cost of managing the OpenEdge environment, and ensure high availability and performance.



OpenEdge Management (OEM)

The OpenEdge Management Console provides visibility into a broad range of configuration, performance and status information on resources. It offers a variety of capabilities to the administrator or user to manage and monitor the performance of OpenEdge resources (PAS for OpenEdge instances and databases). OpenEdge Management allows you to:

- Start and stop resources (databases and servers)
- Deploy and undeploy application resources to servers
- Proactively detect, prevent and troubleshoot failures to improve reliability
- Collect and store necessary data for reports and future reference
- Display resource information and data for the benefit of the system administrator

Explore OpenEdge Management

Interaction with OpenEdge Management is typically limited to those with administrative user privileges. Each of the following role-based learning tracks offers more details on the how these roles interact with OpenEdge Management.

- System Administrator Explore how the role of the system administrator installs, configures, manages and monitor production instances using OEM.
- DevOps Engineers Explore how the role of DevOps engineer plays in supporting the CI/CD (Continuous Integration/Continuous deployment) pipeline using OEM.
- Database Administrator - Explore how the database administrators manage and monitor their databases using OEM.

3

Integrate your applications with external systems

Seamless integration of your application with external systems is critical to the success of any business application.

- **Data integration:**
  - Your application may need to connect to external data sources. Progress provides a variety of connectivity solutions.
  - External applications may need to connect to your data using SQL, JDBC and/or ODBC access. Progress makes this easy to do with our SQL Engine and DataDirect Drivers.
- **Application integration:**
  - External application may need to integrate your business logic into the their applications. Using Progress REST services external applications can easily access your business logic.
  - External applications that use JMS messaging can opt to use JMS adapters to interact with your business logic.

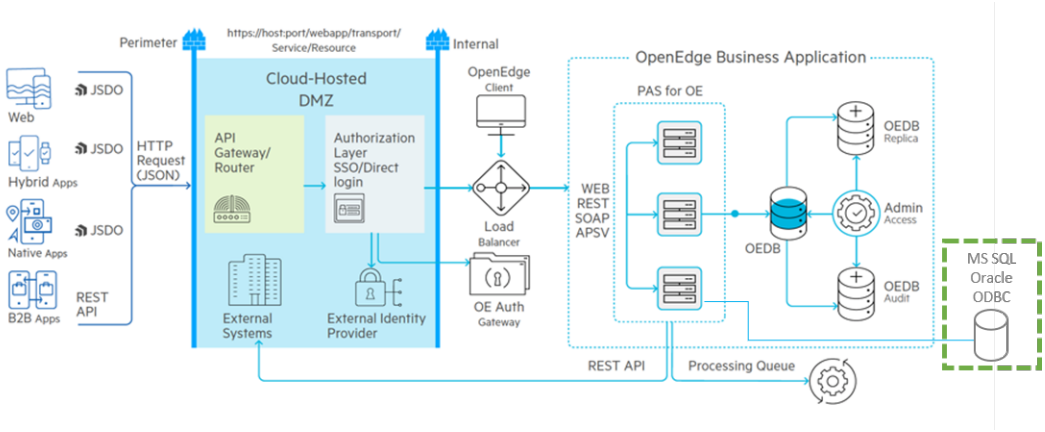
For details, see the following topics:

- Access external data
- Expose data to external applications
- Call external REST services to access logic and data
- Communicate with JMS messages

Access external data

Progress OpenEdge DataServers provide Advanced Business Language (ABL) applications with a flexible, transparent interface that enables simultaneous access to multiple data sources. This enables organizations to allow access to diverse data sources throughout the enterprise.

OpenEdge DataServers combine the rapid application development benefits of the OpenEdge component based development environment with the high performance of today's powerful database engines, such as Oracle and Microsoft SQL Server.



Progress Autonomous REST Connector (ARC) is an intelligent, customizable data connectivity solution enabling businesses to build and deploy enterprise-grade data connectors in minutes without coding. Try it!!

Hybrid Data Pipeline - is a self-hostable hybrid connectivity solution that you can run in the cloud or on-premises. With HDP, you can access data in the cloud or on-premises behind a firewall. Connect through a standard interface—SQL (ODBC, JDBC) or REST (OData 2, OData 4). Try it!

Progress DataDirect OpenAccess helps mask data to various users. Develop a custom driver in ODBC, JDBC, ADO.NET or OLE DB Create quicker data access solutions to proprietary files or APIs . Quickly integrate data across any C, C++, Java, .NET, REST, ABL or C# applications. Try it!!

Custom data connections connect to external data using sockets or web services. This solution requires custom coding.

Expose data to external applications

OpenEdge offers an open, standards-based interface for SQL. The OpenEdge database and its SQL interface provide an environment that enables efficient integration with third-party tools, such as development software, reporting, and Online Analytical Processing (OLAP) tools.

Featuring a state-of-the-art, cost-based query optimizer, the OpenEdge SQL Engine incorporates APIs for ODBC and JDBC. OpenEdge SQL includes the following components:

- ***OpenEdge SQL Engine*** — The OpenEdge SQL Engine is installed as part of the OpenEdge Relational Database Management System. To support the OpenEdge Service Oriented Architecture, the SQL engine offers robust data type support, enables online schema changes, and provides for query optimization. These features are designed to maximize database scalability and performance.

- ***JDBC Driver for OpenEdge by DataDirect*** — The JDBC Driver for OpenEdge is provided by DataDirect Technologies. The JDBC driver enables Java-based applications to access the OpenEdge RDBMS. The JDBC driver is installed as part of the OpenEdge SQL Client Access product.
- ***ODBC Driver for OpenEdge by DataDirect*** — The ODBC Driver for OpenEdge is provided by DataDirect Technologies. The ODBC driver provides access to the OpenEdge database from applications that support the ODBC interface. The ODBC driver is installed as part of the OpenEdge SQL Client Access product.

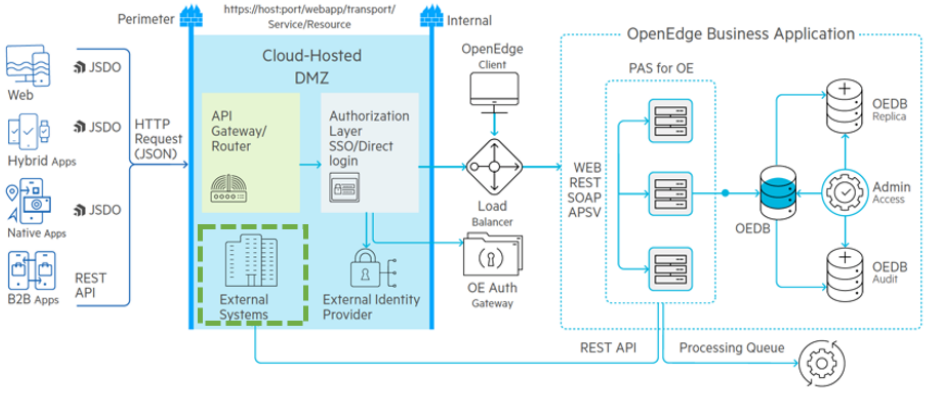
Explore SQL for data access to OpenEdge databases

SQL developers outside of the ABL developer community are the primary users SQL. The following content offers more details on SQL.

- An overview of OpenEdge SQL – For developers new to SQL, explore basic SQL concepts, manipulate data, work with database objects, maintain a database, extend the SQL engine and troubleshoot.
- OpenEdge SQL statements – For experienced SQL developers, explore the reference guide which lists all of the available statements and provides some sample code for getting started with OpenEdge SQL.

Call external REST services to access logic and data

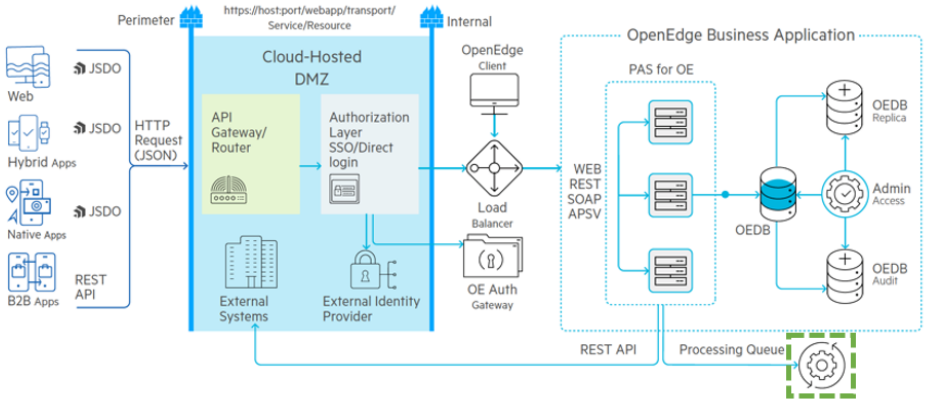
OpenEdge provides the ability to perform HTTP requests from ABL applications through the OpenEdge.Net.pl library. The HTTP Client feature is designed to have a simple, extensible programming interface, which is platform independent. It can be incorporated in most ABL clients, including GUI, TTY, and batch.



For more information see OpenEdge API Reference

Communicate with JMS messages

PAS for OpenEdge supports Java Message Service (JMS) messaging; an industry-standard messaging API that defines a full set of messaging capability. It allows diverse applications to communicate through a messaging queue in a distributed enterprise system. You may already have a JMS product in house. Progress fully supports Aurea's SonicMQ JMS implementation as it offers exceptional reliability, performance, scalability, availability, and security, as well as powerful XML-processing capabilities.



In addition to supporting standard JMS messaging, SonicMQ serves as the foundation for the powerful Aurea's Sonic ESB application integration framework. Highly sophisticated management and XML-editing tools, content-based routing, and an Orchestration Server make Sonic ESB an industry-leading integration framework for enterprise applications.