

What You Need to Know About Application Integration



No modern enterprise operates in a vacuum. Business processes are constantly evolving, and so are the software applications companies rely on for critical tasks such as enterprise resource planning (ERP), supply chain management, and financial analysis and reporting.

Consequently, application integration has become increasingly important to anyone seeking to connect older enterprise applications to newer cloud applications, ensure access to shared data for both mobile and on-premises users, and navigate the digital transformation with minimal disruption to (and, hopefully, great improvement upon) productivity and profitability.

An informed approach to enterprise application integration is essential to creating an integration solution that harnesses big data and makes all of your individual applications work together smoothly.

What Is Application Integration?

From cutting-edge mobile apps to hoary old legacy systems, your business likely depends upon a mixed bag of individual applications that need to communicate

smoothly—and often don't.

Application integration is the process of connecting discrete enterprise applications in order to cut overhead costs—especially in information technology (IT)—while boosting both scalability and organization-wide productivity and efficiency.

Application integration has four standard levels: *presentation-level integration*; *business process integration*; *data integration*; and *communications-level integration*.

Rather than a hierarchy, the four levels are overlapping technologies that, used together, create a complete application integration solution connecting new applications with existing ones.

Presentation-Level Integration ties together a mishmash of different applications into a single application.

This type of application integration is one of the oldest, and is also known as “screen scraping” for the practice of using transitional software (aka *middleware technologies*) to collect and organize information.

By creating a virtual middleman to collect the data and distribute it to each of the separate applications, presentation-level integration provides an improvised but suboptimal method for data integration and access.

Business Process Integration connects essential business processes to local and cloud servers.

It leverages the power of automation and artificial intelligence to connect individual applications, organize and prioritize workflows, and improve efficiency while reducing errors and eliminating roadblocks.

Data Integration is the full and transparent sharing of relevant information between applications.

Without it, integration isn't complete, and efficiency, accuracy, and efficacy all suffer.

Most application integration solutions rely on either specialized code written to

ensure all applications can “speak” to each other in mutually intelligible language, or, more commonly, by using a translation method to convert each application’s output to a data format readily accessible to the others.

These concepts—often referred to as data transformation—provide the core of any successful enterprise application integration solution.

Communications-Level Integration is the specific method used to achieve data integration and business process integration.

If data integration is basic human communication, then communications-level integration is choosing whether to write, speak, sing, or dance.

Special protocols known as Application Programming Interfaces (APIs) are used to create these go-betweens, and ensure every application is receiving information it can understand while transmitting information that’s useful and clear to the others.

Communications-level integration can use different methods of middleware to facilitate data transformation within the integration platform, including *point-to-point*, *hub-and-spoke*, or *Enterprise Service Bus* (ESB).

- **Point-to-Point** allows applications to send and receive requests and replies directly. It suffers from potential chokepoints due to the need for each application to handle requests and replies directly.
- **Hub-and-Spoke** uses centralized middleware to handle data transformation and route information appropriately.
- **ESB** converts each “spoke” in the hub-and-spoke model into a processor for each requesting application, leaving the hub free to handle lighter routing tasks. This method vastly improves scalability and speed compared to the other two approaches.

Application integration is the process of connecting discrete enterprise applications in order to cut overhead costs—especially in information technology (IT)—while boosting both scalability and organization-wide productivity and efficiency.

Synchronous vs. Asynchronous Exchange

In addition to defining a mutually-intelligible *lingua franca* your applications can use to communicate, application integration solutions also use a mix of two specific types of messaging to transmit and receive relevant information between data sources and applications: *synchronous exchange* and *asynchronous exchange*.

Synchronous Exchange

Using this method (also called *request-response*), an application sends a request to another application and cannot continue with other functions until it receives a response.

The connection formed by this exchange method remains open for the duration of the data exchange and relies on middleware to prevent needless delays due to errors and timeouts.

For example, a Web server delivering site content to online shoppers using a browser in real time is a synchronous exchange using the Hypertext Transfer Protocol (HTTP) API.

Another example would be a customer service rep clicking “call” for a specific contact within a customer resource management (CRM) system, and the customer’s contact information being served to the communications suite from for automatic dialling.

Synchronous exchange is ideal for sequential processing, but its dependency on confirmed responses can slow your workflow if the middleware on the back end is suboptimal and requests frequently timeout, or fail due to lack of response.

Asynchronous Exchange

Unlike its synchronous sibling, this exchange method doesn’t rely on a confirmed reply to continue operations.

Connections are closed as soon as requests are sent, and operations can be performed in parallel and in any order.

API management is more robust, and as a result, this method is more scalable, swift, and versatile than synchronous exchange.

Asynchronous exchange integration methods use a few different types of messaging:

- **Request-Callback** is an asynchronous variation on Request-Reply. The application sending information requests doesn't "hold the line," so to speak; instead, it sets up a specific action called a callback to process whatever response is sent and continue with other operations as needed. This speeds and streamlines automated tasks, such as a single application pulling information from multiple other applications, collating and analyzing the information received (sorting it based on dependencies), and then generating a report once it receives all the information required.
- **Fire and Forget** is a *one-way* method of asynchronous exchange. The requesting application sends one (or more likely, many) requests to other applications, and proceeds without receiving confirmation. If your sales staff uses a CRM solution to enter new information about a customer, for example, the CRM might update your ERP system automatically with the new contact data automatically, and vice-versa.
- **Message Routing** is a variation of Fire and Forget that integrates middleware to coordinate requests from multiple applications served to a single receiving application for processing. Routing and splitting rules are created to ensure data is processed in the proper order to complete the next step in each process (e.g., report generation, or creating invoices for customers using information pulled from multiple distinct systems).
- **Publish and Subscribe** is an inversion of the Message Routing method. Receiving applications create rules for the types of information they want to receive, and uses middleware to pull the necessary information from diverse sources as required. For example, a megacorporation using a centralized ERP solution could feed that system from CRM, billing software, financial and big data analysis suites, etc. This method allows for different applications to serve as both senders and receivers, creating an intricate but powerful network of cross-connected applications serving data to a single user interface with the highest possible data quality and speed.

Connect, Collaborate, and Coordinate with Ease

Many hands make light work, as the saying goes—but too many chefs spoil the broth.

Application integration is the key to unlocking all of your applications' potential and ensuring they, and the people who depend upon them, communicate effectively in support of your company's goals.

Making application integration a central part of your IT infrastructure is an important step to building a data-driven integration architecture designed to help your company compete, grow, and innovate.

What's your goal today?

1. Use PLANERGY to manage purchasing and accounts payable

We've helped save billions of dollars for our clients through better spend management, process automation in purchasing and finance, and reducing financial risks. To discover how we can help grow your business:

- Read our case studies, client success stories, and testimonials.
- Visit our "Solutions" page to see the areas of your business we can help improve to see if we're a good fit for each other.
- Learn about us, and our long history of helping companies just like yours.

[Book a Live Demo](#)

2. Download our guide "Preparing Your AP Department For The Future"

Download a free copy of our guide to future proofing your accounts payable department. You'll also be subscribed to our email newsletter and notified about new articles or if have something interesting to share.

download a free copy of our guide

3. Learn best practices for purchasing, finance, and more

Browse hundreds of articles, containing an amazing number of useful tools, techniques, and best practices. Many readers tell us they would have paid consultants for the advice in these articles.

Related Posts