



The Hybrid Integration Landscape for Enterprise

Integration Scenarios, Functionality and Personas

Cloud Elements Point of View

The Hybrid Integration Landscape for Enterprise

Staid for years, software vendors offering tooling to IT teams and systems integrators (SIs) are suddenly working hard to be HIP.

You'll forgive us if that made you roll your eyes—we say that tongue in cheek. HIP stands for Hybrid Integration Platform, a term the analyst firm Gartner coined several years ago and has recently caught fire amongst some vendors. Or at least their marketing teams. Joking aside, at Cloud Elements we find this a welcome shift because it acknowledges two key truths:

1. After years of talking about 'digital transformation,' the relevant decision-makers in the IT function are no longer talking about overnight change and quickly replacing legacy information systems.
2. While change might be slower, it's not glacial; thousands of applications power the modern enterprise and many, even most, are cloud applications and IoT devices that communicate with each other and with on-prem systems via APIs (application programming interfaces). Today, APIs are novel only to Geoffrey Moore's 'laggards.'

APIs are
novel only to
Geoffrey Moore's
'laggards.'

The Cloud Elements Perspective

What we find interesting, however, is the shift to positioning HIP as a product. Not as an integration strategy or a conceptual framework to make sense of ongoing modernization work from the office of the chief architect.

When Gartner analyst Massimo Pezzini first started talking about HIP in 2014, it was described as a strategy made real by intelligently piecing together the right set of integration tools to cover a variety of integration scenarios. With this in mind, it's easier to see why some vendors, several of whom whitelabel Cloud Elements' platform, now talk about HIP as a product: they offer enough integration tooling, covering enough integration scenarios, to say to customers 'we have everything an enterprise needs in one package.'

That vendors offer a comprehensive suite of integration tools is not to say that enterprise customers use the full suite of tools. Most still piece together integration tooling from multiple vendors: EDI platforms to integrate with trading partners, ESBs to connect proprietary on-premises applications, iPaaS for tactical cloud point-to-point integration, and API management to publish the digital fabric across and beyond the enterprise. However, when we look closer at this patchwork infrastructure, it often appears held together with bandaids.

We believe this is a moment where additional clarity is needed: What is a HIP? How can leaders simplify and standardize patterns for hybrid integration challenges? And, of course, where does Cloud Elements' platform fit within this broader landscape in which APIs have become the central building blocks in connecting the enterprise?

To expand, greater clarity is needed more than ever as the number of applications continues to proliferate and the number of users within an organization that expect integrated systems and data continues to expand. Specifically, software continues to eat the world, though without effective integration each application hoards and silos data at a moment when employees and customers expect more deeply integrated processes and intelligent automation.

Second, with more and more applications needing to sync and share data in order to manage day-to-day business processes like marketing and selling customers or verifying and paying suppliers' invoices, business users expect to be able to configure integrations among many applications themselves without writing code.

Finally, enterprises refuse to be left behind: they are launching their own digital products (new revenue streams) and mobile apps. In a world where every company is a software company, as is sometimes quipped, the winners will be those that best integrate products and legacy systems to aggregate, contextualize, and synthesize unique data.

Cloud Elements Enables a HIP

To bring clarity to the landscape, we lean heavily on Gartner's conceptualization of a Hybrid Integration Platform, but we've tried to simplify some. First though, we believe it helps to anchor integration scenarios and functionality in the business scenarios that drive the need for integrations. For example:

Retail e-commerce integrations:

- Integrating product and order information between cloud marketplaces and applications, like Amazon Marketplace, Shopify, and Magento and on-prem or hosted enterprise resource planning (ERP) applications like SAP ECC, SAP S/4HANA, and NetSuite to fulfill and track customer orders.

IT or field service ticket sync and management:

- Integrating work requests, parts requisitions, sensory array data, and work status between service applications like ServiceNow, ServiceMax, and Salesforce Field Service Lightning with on-prem or cloud CRM and ERP applications like Salesforce, SAP ECC, or Oracle J.D. Edwards to maintain systems and equipment for users or customers.

Hyperscaler deployment for analytics and document management:

- Periodically duplicating system-of-record data or organizational knowledge in repositories like Google Cloud Storage, Azure Blob, Sharepoint, Amazon S3, and Google Drive and making that data available to BI and AI/ML tools like Tableau and SAP Analytics Cloud.

Connecting the front-office (sales, marketing) with the back-office (finance & accounting):

- Syncing product, inventory, and promotional data in on-prem or hosted ERP systems like SAP S/4HANA and NetSuite with CRM systems like Salesforce and MS Dynamics as well as product data and content management applications like InRiver and Bynder so sellers have accurate product and inventory information.

Human capital application integration throughout the full employee lifecycle from applicant to employee through payroll, benefits, employee performance and satisfaction, and finally on to retirement:

- Integrating applicant tracking and onboarding systems like Greenhouse or Taleo with core HRIS applications like PeopleSoft, SuccessFactors, and Workday as well as with payroll and benefits administrators like ADP or Ultipro for a hiring experience that helps attract and retain top talent.

Integration Scenarios & Functionality

More than
50%

There's clearly more work to be done in the 'Gateway' space despite significant investments to date. In the most recent [State of API Integration report](#), more than half of all survey respondents said that the primary ask from their API consumer is "Customized APIs that fit a specific business need" (55% of respondents vs 19% for the second-highest response).

These business needs are, of course, only a small fraction of enterprise needs. But already we can see patterns emerge. In terms of integration scenarios, the key question becomes 'What environments do our target databases/applications/devices live in: on-premise or in the cloud?' The second question then becomes, 'if connecting on-premise to the cloud, what's the gateway?' With these two questions we can outline the following integration scenarios:

On-Premises (on-prem):

- **Direct-to-Database:** data-centric integration between databases and/or a database and an application (owned or third-party hosted). Database integrations typically relate to either very old IT systems or very modern data and analytics infrastructure.
- **Application-to-Application (A2A):** orchestrated integrations (data + processes) between multiple standalone applications that may be deployed on owned infrastructure or in hybrid-cloud or virtual private cloud environments but are managed and maintained by the enterprise.
- **Business-to-Business (B2B):** the file exchange mechanisms that allow two companies, typically in a vendor-buyer relationship in a supply chain, to exchange data. We include these in 'on-prem' because EDI platforms are often owned and managed on-prem, providing just a narrow gateway for trading partners.

Gateway: the platform that allows the enterprise to write, publish, manage and/or monetize, and ultimately retire APIs.

- **Ground-to-Cloud:** orchestrated integrations between cloud applications and any on-prem or privately hosted application or database.

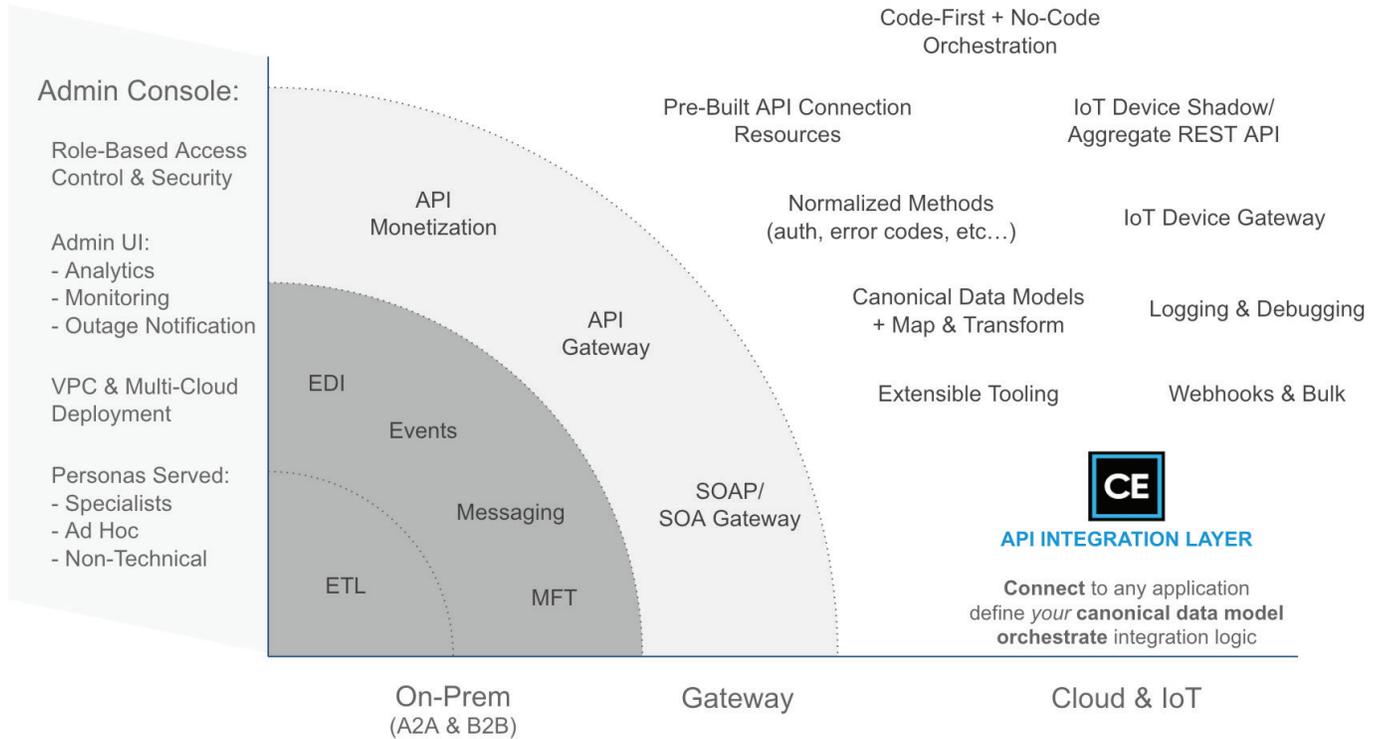
Note: The API gateway not only creates the gateway for on-prem-to-cloud integrations, but also allows the enterprise to publish reusable artifacts that other business units or product lines across the enterprise can use. These artifacts include common object models, reusable integration workflows, and robust connectors that can be leveraged by developers and IT users anywhere in the enterprise.

Cloud: Note that many platform services from major cloud providers (AWS, Azure, and Google) fall into this category, though self-managed private cloud environments may look and feel more like on-prem environments:

- **Cloud-to-Cloud:** orchestrated integrations between two cloud applications hosted and managed by the vendor—software-as-a-service. While cloud-to-cloud scenarios are more relevant for enterprise IT every day, seldom do mission-critical applications fall into cloud-to-cloud scenarios.
- **Devices (or Internet of Things—IoT):** integrating the millions of connected sensors and devices in the world that need to talk to one another and to control systems. We include these as cloud because the devices typically roll up to a control system API that looks like any other cloud application.

Integration Scenarios & Functionality

Depending on the integration scenario, IT teams and SIs have a range of tools to choose from. Figure 1 presents a more detailed view of the integration tooling landscape. However, given the explosion of best-of-breed cloud applications and recognizing that digital transformation often means slow and steady modernization, we feel confident that the main challenge facing enterprise architects today is integrating legacy on-prem applications with modern cloud tools.

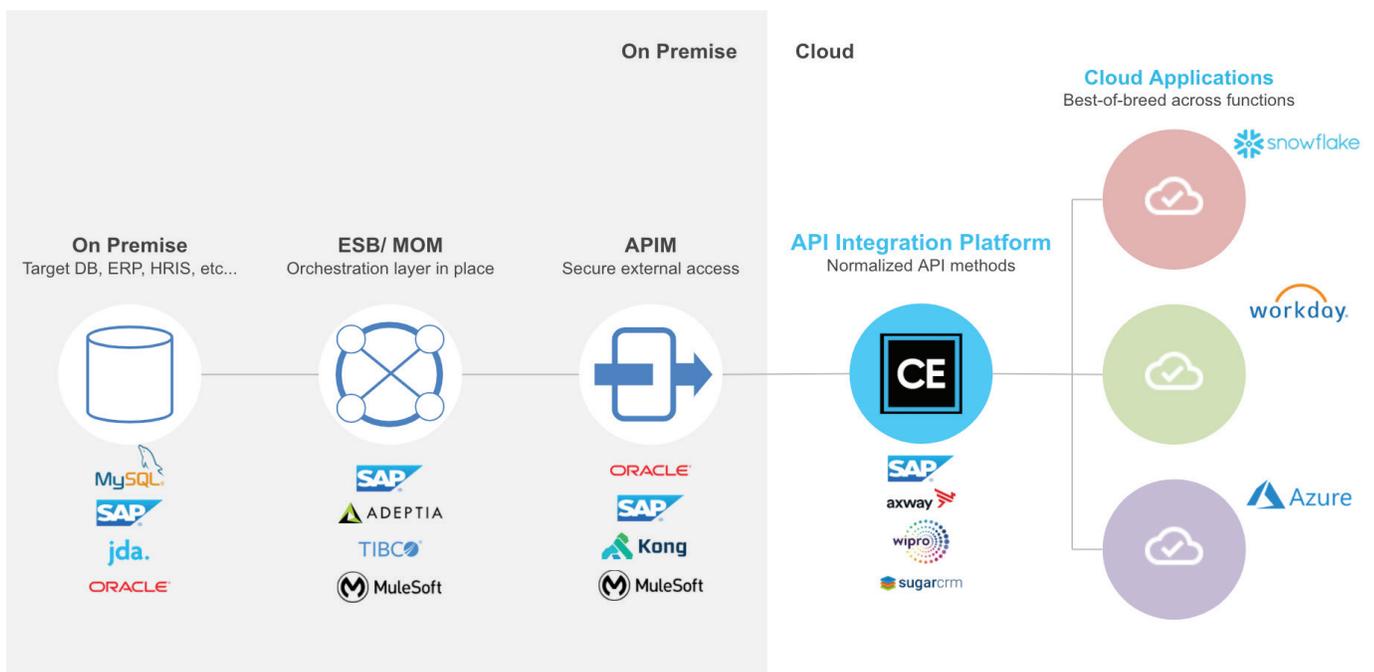


Start With 'Hybrid'

Legacy System-to-Cloud Integration (and Data Virtualization)

Legacy systems are not going away and yet cloud applications proliferate. More customers, users, and employees, conditioned by the modern UIs of the consumer apps we use daily, all demand new and modern best-of-breed software tools.

Figure 2 below represents the most common pattern within ground-to-cloud scenarios: on-prem systems and the ESB integration tooling already in place, plus an APIM platform to expose a stable API, which is connected to the cloud application via an API integration platform.



Why the API integration platform? Integration specialists can write and host integration code in multiple places, but leaders from SAP to Wipro increasingly prefer to let integration specialists take advantage of pre-built API connection resources and other purpose-built functionality to connect with cloud APIs as well as map and transform the data. Process orchestration can live in the on-prem or the API integration tooling depending on the team's preference; we recommend an API integration platform for a more modern developer experience.

Start With 'Hybrid'

How does an API management layer like Apigee, Kong, etc. relate to an API integration platform? At the most basic level, API management tools help your organization publish and maintain APIs (and support the developer experience). APIM makes on-premise applications and data available for third parties to consume. The API integration layer, on the other hand, helps drive the consumption of APIs, whether by your teams or third parties.

Is the API management layer necessary? In our opinion, it's not necessary but is strongly recommended. Philosophically we believe APIs should be thought of as products with their own lifecycles, security concerns, documentation, and maintenance needs. API management supports all of this elegantly. So while the minimum requirement is to expose an API, APIM is usually the smartest way to do that at scale.

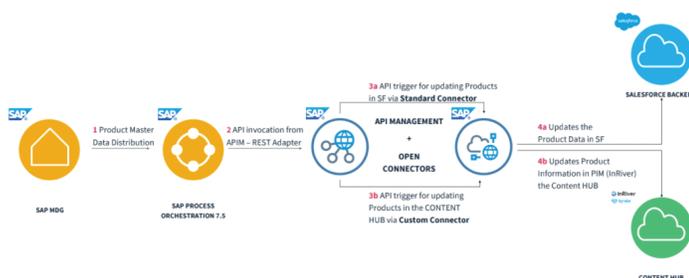
Does that mean enterprises need to buy API management? Not necessarily. In the examples below, two SAP customers, Bacardi and Harrods, found they already had access to API management via SAP's Cloud Platform (which includes SAP Open Connectors, the white-label of Cloud Elements that [SAP makes available to customers](#)).

In the example below, Bacardi describes its work to create a better experience for sellers and distributors by syncing product data and inventory (ERP) with the tools sellers use daily (Salesforce + Content Hub). The on-prem systems (SAP ECC, not pictured—and master data management – MDG, pictured) connect via the ESB layer already in place (SAP PIPO), through the API management layer (SAP API Management), and then via the API integration platform (SAP Open Connectors) to the cloud applications (Salesforce, InRiver, and Bynder).

The Bacardi team was able to deliver this production-ready, end-to-end integration, including three disparate cloud applications, in under a month. Without the tooling to expose the API to MDG and the normalized connectors of the API integration platform, this integration would have taken four to six months according to SAP application owner Sergio Morales.

Enterprise On-Prem - Cloud Integration: Bacardi

On-Prem + API Management + Cloud Elements to Sync Product Data for Sales



Using SAP Open Connectors (Cloud Elements), Bacardi **improves the experience for sellers and distributors in weeks**.

- Added resources to **pre-built connectors in minutes**
- Built, tested, and **deployed custom connectors in days**
- **Deployed production integration in weeks** with little disruption to the business

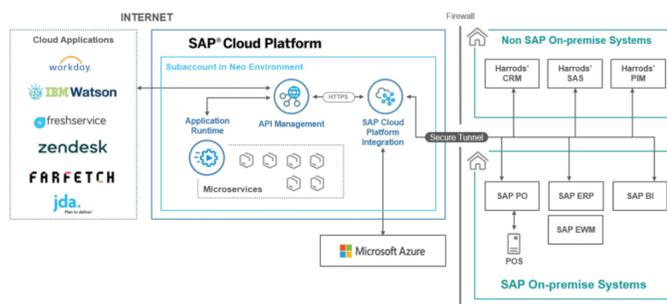
Start With 'Hybrid'

As a second example (below), UK retailer Harrods, whose flagship store is the third-most popular tourist destination in London, uses a similar architecture to connect multiple business-critical, on-prem systems (including those that process transactions and manage inventory) with best-of-breed cloud applications.

The on-prem ESB layer (SAP PIPO, not pictured) orchestrates processes and calls behind the firewall. Via a secure tunnel, they connect with SAP's Cloud Platform and use SAP API Management to both manage a series of microservices and their Azure data lake environment. Finally, using SAP Open Connectors (not pictured) they connect their on-prem and owned cloud environment to best-of-breed cloud applications and AI/ML tools like Workday and IBM Watson.

Enterprise On-Prem - Cloud Integration: Harrods

On-Prem + API Management + Cloud Elements to Modernize Customer & Employee Experience



Using SAP Open Connectors (Cloud Elements), Harrods adopts an integration strategy **that gives customers & employees best-of-breed applications.**

- Reduce reliance on non-redundant ESB layer
- Decouple integration strategy from business-critical applications
- Uses pre-built connector + extensible tooling to unlock analytics & leverage cloud apps

Philosophically we believe APIs should be thought of as products with their own lifecycles, security concerns, documentation, and maintenance needs.

It's worth asking what Harrods is doing in the Azure cloud environment. They're very likely replicating or virtualizing the data from their on-premise systems-of-record in what Gartner calls a 'high-performance data store' within a **digital integration hub architecture**. This architecture replicates some or all of the data in the system(s)-of-record in a modern environment that can handle the load of front-end applications and uses modern integration methods that legacy systems do not support.

Data virtualization architectures like this are novel ways of building a hybrid landscape, not just integrating one, and come with complexity like syncing and maintaining state between the virtualized data store and the system of record. However, the data store functions like a new digital product, which means organizations like Harrods can take advantage of Cloud Elements' Virtual Data Resources (VDRs), which allow users to go beyond mapping and transformation to create one-to-many canonical models of data objects used across multiple applications. In the world of enterprise integration platforms, only Cloud Elements VDRs offer a data governance option that breaks the one-to-one reliance on the data models of pillar applications in the stack.

Who needs to be empowered, and how?

Multiple Interfaces For Multiple Personas

The spectacular growth of cloud applications has had an unintended consequence: shadow IT purchases. With low-cost cloud apps, business users are less dependent on IT's backlog than in the past. This, however, creates new vectors for attack and new challenges for data governance. Some organizations, no doubt, have tried to simply say no to these purchases. Not out of principle, but merely out of control. Others aim to empower people outside the IT function with tools and opportunities appropriate for their level of technical knowledge to solve integration problems themselves. Gartner recommends the latter, along with smart policies and guidelines, and we tend to agree.

For organizations that choose empowerment, the question becomes who exactly needs to be empowered, and how? [Gartner offers](#) several different personas of people who do integration work, and we find their definitions work well:

- **Integration specialists** – people for whom developing integrations is their primary or the majority of their daily work. This includes integration developers within SIs and many specialists who are accustomed to writing code in purpose-built Java environments.
- **Ad hoc integrators** – people for whom creating or maintaining integrations is a part of their daily work. This could include the operations lead in the marketing team who has enough technical knowledge to integrate Salesforce and Marketo without too much help from IT. It could also include the professional services or customer success team members who act on direct customer needs during onboarding for a new product or service or in order to expand usage of a digital product. Ad hoc integrators are likely familiar with the logic of orchestrated processes and data mapping, but may not have the time or knowledge to work with unique APIs.
- **Citizen integrators** – business users without technical skills and for whom integration may stand in the way of their daily work. Citizen integrators still need different applications to sync and share data but are often content with out-of-the-box integration functionality and cannot write code or deal with complex mappings. Several no-code tools have grown popular among citizen integrators, like IFTTT and Zapier, but more often the developers of digital products find they need to offer these users pre-built integrations with commonly requested applications that only require a username and password to configure.
- **Application developers** – developers for whom integration becomes an increasingly large challenge as the digital product and usage grows but acts as a feature, not the core value proposition of the application. Many application developers, while not as versed in the nuance of integration as the specialists described above, are increasingly familiar with embedding third-party services in their applications via API call. Maintenance of integrations is a particular concern for application developers because it represents time not spent advancing core features and functionality.

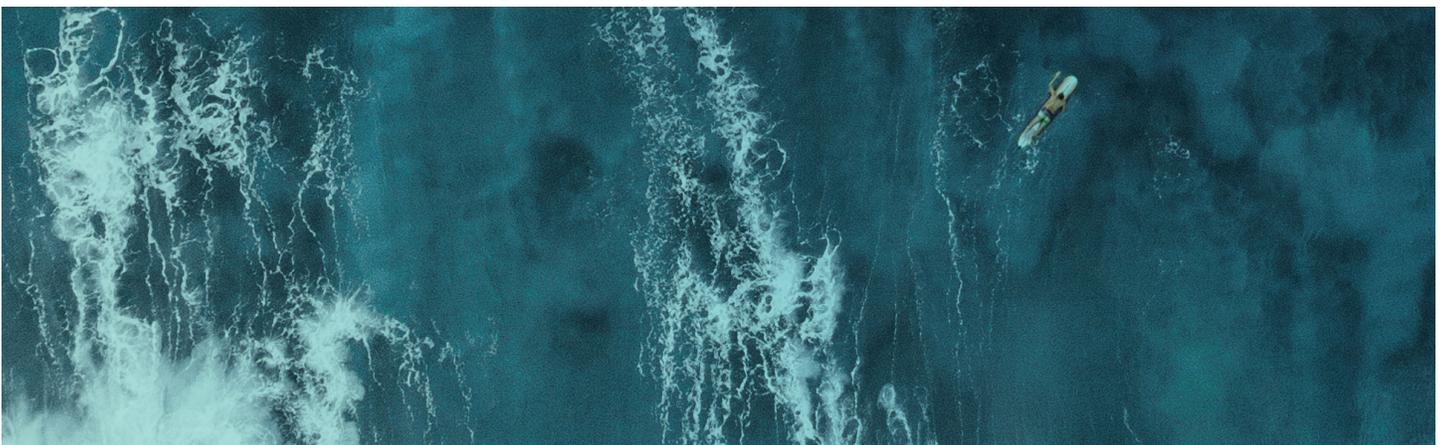
Code or No Code?

With so many disparate integration personas, not to mention the preferences of the systems integrator partners that support so many enterprises, it can be tempting to buy software that might support each integrator persona and hope for the best. A large organization likely already owns multiple different integration tools that serve these different user bases.

However, there are advantages to centralizing all of the integration work in one place and providing reusable resources to ad hoc integrators and application developers. In both cases, the myriad integrations between hundreds of applications are easier and less costly to maintain with a single source of truth and common patterns. But to serve multiple personas well, common resources and standardized patterns need to be accessible via multiple interfaces or modes.

In service of this vision, we have designed our API integration platform to give different user types the access they want to normalized methods and reusable integration resources:

- 'Code-first,' or headless, where application developers write to our APIs (every resource in our platform is accessible via API call). This is the preferred method for product and engineering teams that want to embed integration as a feature in new digital platforms, often serving a non-technical, citizen integrator end-user.
- What could be called 'code-lite' but is, in fact, a developer-focused UI giving an experienced integration specialist all the tooling necessary to develop new integrations and/or reusable resources that others in the organization can use. Writing integration code within this UI organizes the work intelligently, saving time during development, and makes it easier for other team members to understand existing integrations in order to maintain them.
- And a no-code, or graphical interface to configure integrations using pre-built resources (including Elements, VDRs, and process templates). This type of interface is designed for an ad hoc integrator that may be comfortable writing some javascript but can work more quickly using icons and components.



Every Business Is Now a Software Business

Many application developers, while not as versed in the nuance of integration as the integration specialists Gartner describes, are increasingly familiar with embedded third-party services in their applications via API call. Also called headless, these API-first services like Stripe for payments or Twilio for telephony offer developers and enterprises building digital products a clear build-vs-buy decision: they can build a service to effectively compete with Stripe or Twilio (both clear market leaders) or pay to embed those services and focus their developments on the core product.

This type of embedded integration is no longer the realm of startups. American Express and Western Union are examples of large enterprises that have recently launched new, digital products with integration embedded as a feature. HP and Xerox have also embedded integrations in their scanning and document management solutions.

In the case of American Express and Western Union, the new platforms need to integrate with their business customers' accounting and/or ERP systems in order to initiate payments or better manage current accounts across banks. However, accounting and ERP systems tend to be highly customized and difficult to integrate with. Embedding Cloud Elements has allowed both enterprises to "productize" self-serve integrations that their customers can configure themselves.

System integrators must also consider embedded integration strategies. Relentless pressure on IT budgets has caused many traditional services firms to turn to 'digital accelerators' that help them deliver client projects more reliably and more profitably. These accelerators often function similarly to digital products, or large pieces of them, though usually without the finished UI or architecture.

Whether new digital platforms or digital accelerators, we believe Cloud Elements' approach offers unique advantages. When looked at through the common lens of digital products, new offerings from established enterprises and accelerators from their SI partners benefit by embedding integration capabilities as a multi-tenanted and reusable feature, or feature set, rather than starting from scratch each time.

Embedded Integration + Accelerators

In the case of American Express and Western Union, the products mentioned are hardly the first, or the last, that these enterprises will offer to their customers. While each digital platform is unique, building integrations as features quickly becomes repetitive, undifferentiated work. Looking across a product portfolio, it makes sense for enterprises to standardize on common, reusable API integration resources (multi-tenanted connectors, canonical data models, and orchestration templates) to limit development time, bring products to market faster, and limit the downstream maintenance burden.

Similarly, systems integrators building digital accelerators can develop proprietary resources they can resell and redeploy, or at least build platform expertise using common methods and resources, in order to deliver reliably and profitably. This could take the form of proprietary Elements and VDRs to copy and reuse when implementing client projects. Multiple business models can meet a range of client requirements and preferences. But at the core is a normalized set of integration patterns and resources that, as mentioned previously, deliver production-ready integrations in a fraction of the time of other, legacy integration platforms.

In sum, whether your integration scenarios include heterogeneous process integrations between dozens of on-prem and cloud applications or you're pursuing the opportunity to create new revenue streams via digital platforms, Cloud Elements can help reign in the immense and expanding world of APIs and power your hybrid integration architecture to save costs and deliver user-facing innovation faster.

About Cloud Elements

Cloud Elements brings harmony to the world of APIs, allowing software providers to innovate faster and plug into digital ecosystems. The company's one-to-many virtualized API integration platform enables developers to unify thousands of APIs, build common data models for core business functions, and reduce the pain, cost and complexity of integration. Founded in 2012, Cloud Elements is headquartered in Denver, Colo., and serves customers worldwide. More information can be found at www.cloud-elements.com.