

# Integration with Liferay DXP

How to Choose the Right Strategy  
for Your Digital Business

# Table of Contents

Overview . . . . .	1	Cross-Domain Concerns . . . . .	13
The Need for Application Integration . . . . .	2	Mobile Integration . . . . .	14
The Era of Industry 4.0 . . . . .	2	Mobile Website vs. Native vs. Hybrid . . . . .	14
Integration Lays the Foundation for		Avoiding Additional Silos . . . . .	14
Continuous Transformation . . . . .	3	Headless and Decoupled DXP . . . . .	15
Integration Improves Business Efficiency and		Headless vs. Decoupled . . . . .	15
Customer Experiences . . . . .	4	API Support . . . . .	16
Integration Means Connected Experiences . . . . .	5	Open API . . . . .	16
Liferay DXP: Designed and Architected for		REST Builder . . . . .	16
Integration . . . . .	5	Key Use Cases for Decoupled	
Headless and Decoupled . . . . .	5	DXP Integration . . . . .	17
OpenAPI . . . . .	6	Best Practices . . . . .	19
OSGi Standards . . . . .	6	IoT Integration . . . . .	19
UI/UX Frameworks Support . . . . .	6	Edge Devices . . . . .	21
Development Tools and Platforms . . . . .	6	Dashboards . . . . .	21
Identity Management . . . . .	7	Liferay is the Flexible Choice for IoT . . . . .	21
Authentication and Authorization . . . . .	7	Best Practices . . . . .	21
Permissions and Roles Based Access Control . . . . .	7	Native Liferay App and Widget Integration . . . . .	23
Ready-to-Deploy Applications . . . . .	7	Liferay App Development . . . . .	23
Liferay Application Integration Strategies . . . . .	8	Summary of Development Tools	
Levels of Application Integration . . . . .	8	Within the DXP Platform . . . . .	24
Liferay DXP Integration Options . . . . .	9	Front End Development for UI/UX . . . . .	24
IFrame Integration . . . . .	9	Examples and sample projects for	
IFrame Authentication . . . . .	10	native apps and widgets that integrate . . . . .	25
Basic vs. Form-Based Authentication . . . . .	11	Advanced Integration Techniques . . . . .	25
Automatic vs. Configured Username		Summary . . . . .	26
and Password . . . . .	12	Resources . . . . .	26
Hidden Variable . . . . .	13	Moving Forward . . . . .	27

# Overview

Integrations are the cornerstone of successful digital experience platform deployments. After all, digital experience platforms (DXPs) do not stand alone or exist within a vacuum by themselves. They serve a purpose of bringing data and functionality from disparate sources into one cohesive system and suite of applications. Liferay DXP makes it possible to achieve different levels of integration, each with their own benefits and varying levels of effort, allowing you to choose the right strategy for your business needs. Laying out a realistic and pragmatic integration strategy is critical and Liferay DXP allows you to transform long-standing legacy systems into modern, streamlined solutions. Every effective integration will have its pros and cons, and strategic compromises must be well-planned at key points in your organization's overall digital transformation journey in order to realize the long-term vision of your business. Liferay DXP allows you to pursue digital transformation in a way that is best for your organization.

Liferay DXP is flexible enough to integrate within a variety of scenarios, utilizing existing corporate and IT investments and assets. Common scenarios include:

- Integrating Liferay DXP with an existing SSO authentication or authorization policy engine, which may involve development of a custom SSO connector to the service or the re-use of an out-of-the-box Liferay authentication connector.
- Existing ERP or CRM workflows need to be managed and executed through your Liferay DXP instance. This can range from a simple presentation-level IFrame to a native DXP widget that calls open APIs and headless engines.
- Multiple legacy web-based applications that need to be unified for a consistent user experience.
- Integrating with popular social networking platforms so that your business has impactful presence or improve the tracking and segmentation of anonymous users that connect to your social media assets in your campaigns.

These kinds of integration and much more are possible with Liferay DXP. In this paper, we explain how to integrate existing enterprise web applications into Liferay DXP in order to provide a seamless user experience, while capitalizing on corporate legacy IT assets. Additionally, we'll also cover how to share from the core Liferay DXP asset repository and allow assets to be consumed by existing systems and new digital touchpoints outside of the conventional web browser. In this way, you can extend the value of your current systems, leverage valuable data and allow those and newer systems to consume DXP assets.

# The Need for Application Integration

## The Era of Industry 4.0

We are living in the era of Industry 4.0 - the Fourth Industrial Revolution. Since the beginning of industrialization, technological leaps have led to paradigm shifts which today are ex-post named “industrial revolutions”: in the field of mechanization (the First Industrial Revolution), of the intensive use of electrical energy (the Second Industrial Revolution) and of widespread digitalization (the Third Industrial Revolution). The combination of internet technologies and future-oriented technologies in the field of “smart” objects (machines and products) seems to result in a new fundamental paradigm shift in industrial production.<sup>1</sup>

The Fourth Industrial Revolution describes the exponential changes to the way we live, work and relate to one another due to the adoption of cyber-physical systems, the Internet of Things and the Internet of Systems. As we implement smart technologies in our factories and workplaces, connected machines will interact, visualize the entire production chain and make decisions autonomously. This revolution is expected to impact all disciplines, industries and economies. The Fourth Industrial Revolution is disrupting almost every industry in every country and creating massive change in a non-linear way at unprecedented speed.<sup>2</sup> From the First Industrial Revolution (mechanization through water and steam power) to the mass production and assembly lines using electricity in the Second, the Fourth Industrial Revolution will take what was started in the third with the adoption of computers and automation and enhance it with smart and autonomous systems fueled by data and machine learning.<sup>3</sup>

---

1 Adaptive Integrated Digital Architecture Framework with Risk Management for Global Enterprise

2 What is Industry 4.0?

3 What is Industry 4.0?

# Integration Lays the Foundation for Continuous Transformation

Legacy technology is the greatest barrier to digital transformation for 50 percent of companies, according to a recent survey of senior stakeholders.<sup>4</sup> But research also shows that if enterprises were to take a combinatorial approach to data and information systems, they could significantly boost both savings per employee and market capitalization. A combinatorial approach is also key to value creation as an Industry 4.0 business: a smart, connected, living and learning enterprise that embraces constant technological change, and drives profitable growth from it.<sup>5</sup>

Every company across every industry is now compelled to reconsider their traditional ways of doing business to keep pace with rapidly changing technology and consumer expectations.<sup>6</sup> As Industry 4.0 disrupts and transforms information technology, enterprises can be prepared to take advantage of new trends among consumers, employees, partners, students, staff and how they realize the ROI of their digital technology stacks.

Well-planned enterprise architecture visualizes the current corporate IT environment and business landscape to promote a desirable future IT model, offering benefits to companies, such as coordination between business and IT, enhancement in organizational communication and reduction in the complexity of IT.<sup>7</sup> **When we are able to integrate and unite millions of data points across systems, equipment and processes, we are able to derive actionable insights across the entire value chain—everything from product design to operations to customer support.** Data insight from an integrated system reveals patterns and answers questions across the business about users, equipment, locations, streaming sensor data and more.<sup>8</sup>

---

4 Legacy technology is biggest barrier to digital transformation

5 Combine and Conquer: Industry X.0 and the Digital Reinvention of Industry

6 The Fourth Industrial Revolution

7 Adaptive Integrated Digital Architecture Framework with Risk Management for Global Enterprise

8 Be Ready for Industry 4.0 with Cognitive Manufacturing

# Integration Improves Business Efficiency and Customer Experiences

Enterprises have complicated application ecosystems, especially with a large number of different apps that are necessary to run the business at any given time. Disruption or gaps in production can cost enterprises millions in revenue and countless hours of lost productivity from workers. Enterprise integration not only mitigates loss of production from apps, but also allows disparate systems to come together and provide a seamless experience to end users and audiences through various touch points. Without integration, the quality of the customer experience diminishes, along with the collective efficacy of the enterprise. To thrive in the era of Industry 4.0, it is critical to unlock the potential of legacy, real-time and unstructured data. It is a tremendous challenge to make that data accessible and actionable, with most enterprises running over a hundred software applications.<sup>9</sup>

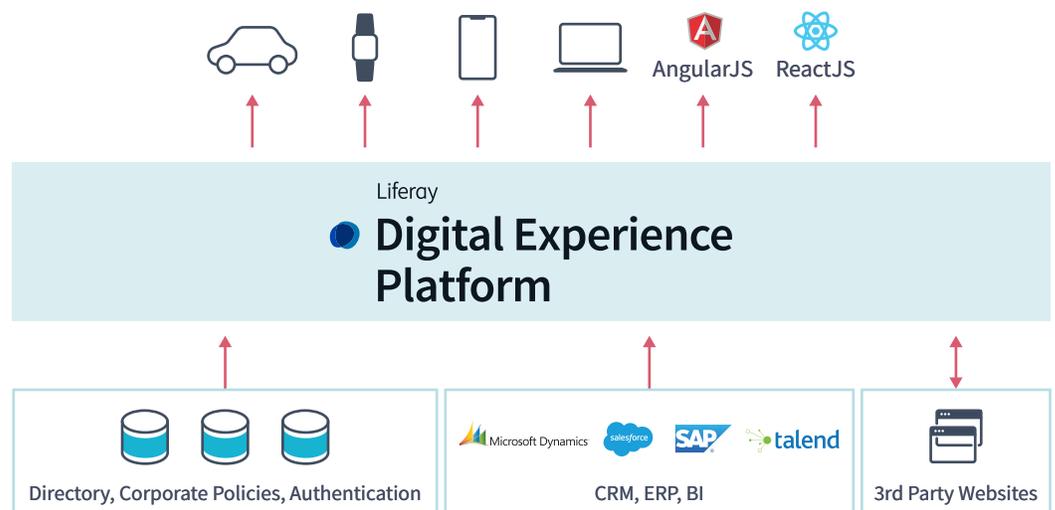


Figure 1: Examples of supported Liferay DXP integration points - Omnichannel, front-end, back-end, 3rd party enterprise software, 3rd party websites

<sup>9</sup> Be Ready for Industry 4.0 with Cognitive Manufacturing

## Integration Means Connected Experiences

In the era of Industry 4.0, enterprise application integration has extended beyond the walls of a building or the context of a web browser. Smart devices, mobile computing and embedded machines extend their reach to consumers, allowing workers to do their job and be productive out in the field, in transport and at any remote work location. Traditionally, integration has always improved the UI/UX of conventional web applications to be more consistent visually, extending across modern devices and new form factors. However, Industry 4.0 demands that the collective experience be cohesive and thoughtfully connected through different media, channels of communication and IoT devices and embedded machines.

## Liferay DXP: Designed and Architected for Integration

Liferay DXP has a framework that has been designed to help with integration in a flexible architecture with open standards for the modern digital enterprise.

## Headless and Decoupled

Headless has been gaining popularity lately and exclusively headless CMS players have been positioning themselves as a better alternative to the traditional CMS approach. The key benefit of using a headless CMS is that it allows front-end developers to access core services while using the tools they are most comfortable with and the front-end technologies they prefer. This benefit has grown in importance since these technologies have improved greatly in recent years, making the front-ends that come out of the box with the CMS obsolete (or close).

The strategy we are following for Liferay DXP is to provide a decoupled CMS (or better a decoupled DXP), which gives our customers the choice of when to use a headless approach and when not to.

## OpenAPI

The new Headless REST APIs leverage OpenAPI, an API specification backed by the main players in the API space (such as Google, Microsoft, IBM) and formerly known as Swagger. By supporting OpenAPI, a widely-used standard, we ease the learning curve to consume and develop new APIs in Liferay DXP and our customers benefit from the thousands of client generators, validators, parsers and hundreds of tools with support for OpenAPI. The OpenAPI specification encourages a contract-first approach to API design, defining your paths and entities (schemas) first, then writing the code to implement them.

## OSGi Standards

Liferay DXP leverages the OSGi framework to provide a development environment for modular applications. There are many OSGi best practices that Liferay DXP aligns with, to provide an easy-to-develop-for platform for creating OSGi bundles (modules) for integration.

## UI/UX Frameworks Support

Liferay DXP can run in a headless or decoupled fashion and this architecture supports the implementation of mobile development and front end frameworks, such as React or Angular, to call Liferay APIs and web services.

## Development Tools and Platforms

Liferay DXP is a platform that is flexible and developer-friendly. Liferay DXP provides a robust development environment, along with plugins, sample projects and wizards to streamline the process to help you get your job done faster. Writing integration applications on Liferay DXP's standards-based platform makes your life easier. You can create headless services for clients to access, full-blown web applications with beautiful UIs or anything in between. Liferay's framework embraces your existing tools and build environments like Maven, Gradle, Eclipse and IntelliJ. You can work with the standard technologies you know and leverage Liferay DXP's APIs for documents, permissions, search or content when you need them.

# Identity Management

## Authentication and Authorization

Liferay has a robust set of options for authentication in the form of supported identity management platforms. These include: LDAP, Kerberos, CAS, SiteMinder, OpenSSO/OpenAM, OpenID, Facebook, Google and SAML support. Mobile websites built on Liferay can leverage these identity management configurations out-of-the-box while native and hybrid solutions can implement login screenlets from Liferay Screens to simplify app login development and leverage existing authentication systems.

Similarly, the Liferay permissions system is integrated into the web experience and content management system in Liferay. Once logged in, users will only see what they are provisioned to see. For native and hybrid apps, Liferay screenlets are available to surface existing content and assets, simplifying development as the screenlets will manage access to each end user according to their assigned Liferay roles and permissions.

## Permissions and Roles Based Access Control

Liferay has a fine-grained permissions system and can aggregate and group specific permissions into Liferay Roles. These roles give user and user groups access to content, pages, fragments, widgets and applications that may be integrated into Liferay DXP and its UI/UX. Whether the assets are accessed through a conventional browser, on a mobile device, or as an IoT service, Liferay DXP will only allow access to what each user has been provisioned for.

## Ready-to-Deploy Applications

For a look at integrations that are ready to go in Marketplace or out-of-the-box, please visit [marketplace.liferay.com](https://marketplace.liferay.com). Apps that are designed for Liferay DXP and produced by the Liferay team can be found under the category labeled Liferay Subscription Apps. In addition, the Liferay community contributes greatly to the library of DXP integration capability through components for communication, productivity, security, utility and themes.

# Liferay Application Integration Strategies

The Liferay DXP environment provides several facilities designed to help integrate external applications into a single, cohesive user experience. Depending on the needs of your digital business, these facilities range from simple presentation layer integrations to deeper and more comprehensive application integrations.

## Levels of Application Integration

The table (Table 1) explains how DXPs and portals integrate applications at different levels, identifying full native portal-level integration as the most seamless and powerful, but also the most costly and complex.

	<b>Presentation Level</b>	<b>APIs/Headless</b>	<b>Full/Native</b>
<b>Complexity</b>	Low	Medium	High
<b>Cost</b>	Low	Medium	High
<b>Conflict Potential</b>	None	Medium	None
<b>Control Over User Experience</b>	None	High	High
<b>Portal Artifact Access</b>	None	Medium/High	High

Table 1: Relative strengths and weaknesses of various integration levels

# Liferay DXP Integration Options

The following table (Table 2) summarizes a few of the integration options available within Liferay DXP and their key differences. These options can be used in combination or on their own to achieve the various levels of integration referenced in Table 1.

	<b>IFrame</b>	<b>Mobile App</b>	<b>APIs/ Headless</b>	<b>Native Widget</b>
<b>Difficulty</b>	Easy	Medium	Medium/ Complex	Complex
<b>Authentication Options</b>	HTTP Basic, Form	Custom / Screens	Custom	Native
<b>Authorization Options</b>	None	Native/ Custom	Custom	Native
<b>Customizations</b>	None	Flexible	Flexible	Easy
<b>Resource Usage</b>	Low	High	Low	Low
<b>Access to Liferay Artifacts</b>	None	High	Flexible	High

Table 2: Key differences between some of Liferay’s integration options

The following sections more comprehensively cover these and more integration options in Liferay DXP and associated strategies to help you better understand and plan the effort involved in integration for your digital business.

## IFrame Integration

An IFrame is a very inexpensive and simple way to integrate applications at the presentation level. It is a popular choice for a “phase 1” type of integration, where you want to get content into Liferay, but plan to do a fuller and tighter integration later on. In Liferay, this is implemented using an IFrame portlet. The portlet makes it possible to embed another HTML page inside the current page. Furthermore, the user can navigate through the embedded page without losing context of the portal page.

The IFrame portlet uses the HTML IFrame tag that is part of HTML4 and is supported in all major browsers. The IFrame portlet will adjust to the size of the HTML page if that page is hosted in the same server. The browser will add scrollbars if the embedded page does not fit within the size of the IFrame.

To get started with the IFrame portlet, you can add an instance of it to any page using Liferay's built-in applications. Once added to a page, click the configure icon to configure the address of the page you wish to embed. For example, type in <http://www.google.com> and save the changes. Navigate back to the page on which the IFrame portlet was added and it will now show Google's search page inside the portlet.

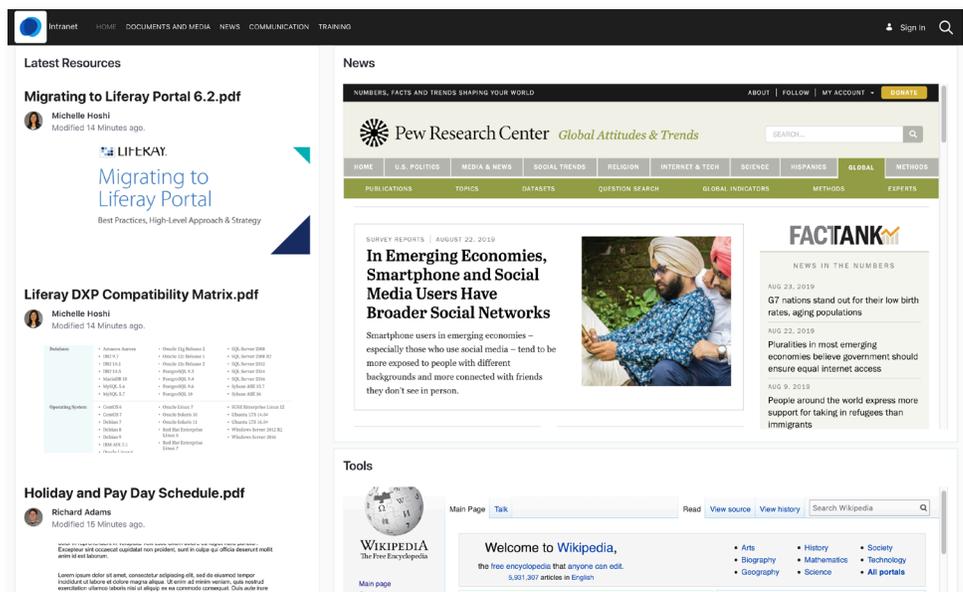


Figure 2

You can interact with this page as though it was the only page you were viewing. The window acts as a miniature browser for that page alone.

## IFrame Authentication

If the embedded web page requires authentication the IFrame portlet can be configured to provide authentication information. This section describes the different options offered to the user.

## Basic vs. Form-Based Authentication

There are several ways to provide the authentication information to the embedded page. The IFrame portlet supports two of them:

- **Basic authentication:** This method is described in the HTTP specification. The username and passwords are usually provided as HTTP headers but it's also possible to embed them in the URL with the syntax `http://username:password@www.myurl.com`. This method is simple but it's not supported in many websites.
- **Form authentication:** Is the name given to websites that authenticate the user by making him fill in a form with his username and password. It's the one used in most web sites. There are two variants of this: GET and POST. The first one sets the username and password in the URL and the second one in the body of the request. Usually they make little difference and both will probably work with any website.

When form-based authentication is used, the IFrame portlet will first make a hidden request to a portal page that recreates a form and then submits it to the configured URL.

Be aware that neither of these methods are secure unless they are sent through an HTTPS connection. If you want to make sure that the user and password information cannot be seen by third parties only use authentication with sites that support HTTPS and use only URLs in the IFrame that start with `https://`.

Figure 3

## Automatic vs. Configured Username and Password

Liferay provides two ways to retrieve the username and passwords that will be used to authenticate against the embedded web page:

- **Automatic:** The username and password of the current user that is viewing the page with the IFrame portlet will be used. It's only supported by the 'basic authentication' method described in the previous section.
- **Configured:** The administrator will configure a username and password and they will be used for all users.

To use the automatic way just leave the ‘username’ and ‘passwords’ fields blank. Otherwise fill them with the values that will be used for authentication. You may use @email\_address@, @screen\_name@, or @user\_id@ as tokens that represent the currently logged-in user information. Liferay will replace the tokens with the information from the currently logged-in user when the IFrame portlet is displayed.

## Hidden Variable

When using form-based authentication, the IFrame portlet allows the administrator to configure a set of additional parameters that will be sent with the form. These parameters must be set in the *Hidden Variables* field and its format must be:

```
xxx=blah; abc=formSubmit1
```

## Cross-Domain Concerns

There is often a need for dynamic communication between applications when integrating them together at the presentation level using IFrames. For example, when a user clicks on the name of a sales contact from their list of contacts in one window, a separate window should update to show the geolocation of that contact on a map. Another common situation is resizing the IFrame to snugly fit around the content. However, when applications are being hosted and provided through separate hosts, modern browsers will prohibit direct communication or execution of code between the two sites hosted in the IFrames. They will also prevent properties (such as the content size) from being read at all, thereby making resizing the surrounding frame impossible.

There are some cases where this can be bypassed, but requires that you have the ability to modify the application being embedded, in order to send messages which can be read by code in Liferay. There are many off-the-shelf open source solutions available for this, most of which rely on the `window.postMessage` JavaScript API. However, the complexity and sustainability of this kind of implementation further illustrates that presentation-level integration may not always be the best solution.

## Mobile Integration

Liferay enables integration for mobile platforms, devices and form factors. Mobile integration may encompass many techniques, and what is available from Liferay DXP can allow your enterprise to have a strong mobile presence and to implement a mobile-first strategy that reduces content silos and leverages existing assets and SEO.

### Mobile Website vs. Native vs. Hybrid

Techniques for mobile presence typically fall into three conventional categories. One can implement a mobile website, create native apps for iOS and Android or create a hybrid app. Table 3 compares some of the advantages of each implementation.

	<b>Mobile Website</b>	<b>Native App</b>	<b>Hybrid App</b>
<b>Difficulty</b>	Medium	Complex	Medium/ Complex
<b>Authentication Options</b>	Conventional Web	Custom / Liferay Screens	Custom / Liferay Screens
<b>Authorization Options</b>	Liferay Permissions	Liferay Permissions / Liferay Screens	Liferay Permissions / Liferay Screens
<b>Customizations</b>	Flexible, Dynamic	Flexible, Restricted	Flexible, Dynamic
<b>Resource Usage</b>	Low	High	Medium to High
<b>Access to Liferay Artifacts</b>	Easy	Liferay Screens	Easy, Liferay Screens

Table 3: Key Differences Between Mobile Strategies

### Avoiding Additional Silos

Liferay content and assets are leveraged across devices and form factors, avoiding redundant repositories by using responsive design and screenlets to deliver content to a wide array of touch points. As media providers must

consider differences between devices form factors when delivering content, they must also consider bandwidth and processing capabilities. The preinstalled Adaptive Media app allows administrators to control image quality and dynamically adjusts uploaded media to best fit the screen being used. Adaptive Media integrates with Documents and Media, Blogs and Web Content. It generates a set of images for use on various screens. When the content is accessed, Adaptive Media checks the screen type and resolution and selects the appropriate image.

## Headless and Decoupled DXP

The Liferay platform can be run in a headless or decoupled mode to develop web apps or mobile apps with any front end technology of your choice (e.g., Angular, React, Backbone, Cocoa, Android’s Material Design components, Apache Cordova). It can also be used as a web integration layer, leveraging technologies such as portlets to allow several applications to coexist on the same web page.

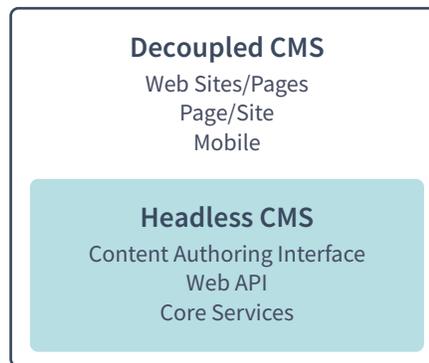


Figure 4 - Headless/Decoupled Architecture Diagram

## Headless vs. Decoupled

Though closely related to each other, headless or decoupled modes run in subtly different ways, serving some use cases better than others (Figure 6). A pure headless CMS option is optimal when there is a need for a custom frontend and there is a strong front-end development team. However, headless may not be a good choice in the following scenarios:

1. When the company doesn't have a strong front-end development team.
2. The value added of a custom frontend is lower than the investment necessary to build it.
3. When it's desired to provide more control to business users and content authors over the end result, such as allowing them to preview it before publishing.

In all of these cases a good alternate to consider is a decoupled CMS, which offers the option to mix a headless approach along with a full UI, allowing companies to choose for each project and need what's the best fit.

## API Support

Liferay DXP includes a set of REST APIs that strengthens support as a headless CMS and as a decoupled CMS. They are available for Liferay DXP 7.1+ and the new REST APIs have been carefully designed to address common business use cases. The APIs are not just simple CRUD representation of Liferay's entities but empower developers with the functionality that covers the requirements driven by two main use cases:

- Development of SPAs and other custom frontends
- Development of omnichannel experiences on multiple devices

Because of the focus on these use cases, most APIs fall in the category of delivery APIs. They support strong filtering, searching and sorting capabilities, which are the most common needs for these two use cases, especially for web content and documents.

## Open API

Liferay DXP supports OpenAPI. OpenAPI is the most widely adopted API modelling and documentation standard. All of our APIs are publicly available at <https://app.swaggerhub.com/organizations/liferayinc> where users are free to experiment and try them out.

## REST Builder

Liferay DXP further support REST with a tool called REST Builder. It facilitates creating REST APIs from an Open API definition. In addition to Open API, REST Builder can also generate a GraphQL endpoint and import/export batch support. Using REST Builder is optional for developers, who can also choose to use JAX-RS directly or other OpenAPI generators.

## Key Use Cases for Decoupled DXP Integration

Though it does not cover the gamut of use cases, Liferay focuses on customer and community feedback and research to guide prioritization of API development. The following are the key scenarios for which we recommended using Liferay DXP for decoupled integration.

1. **DXP as the Back End of a Custom Front End** - Enable your developers to create experiences with the technologies of their choice.

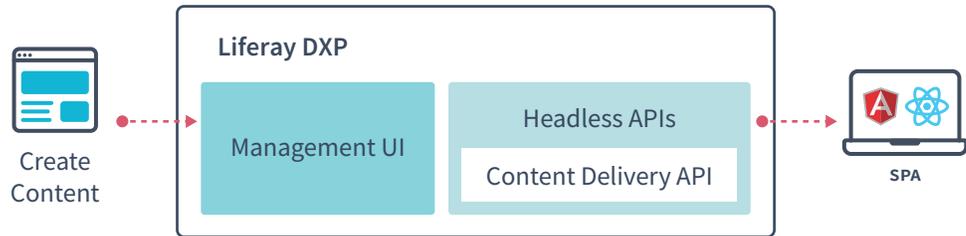


Figure 5 - Example of using alternative UI frameworks such as React or Angular

2. **Enabler of Omnichannel Experiences** - Provide your users with a continuous experience from a single source of data.

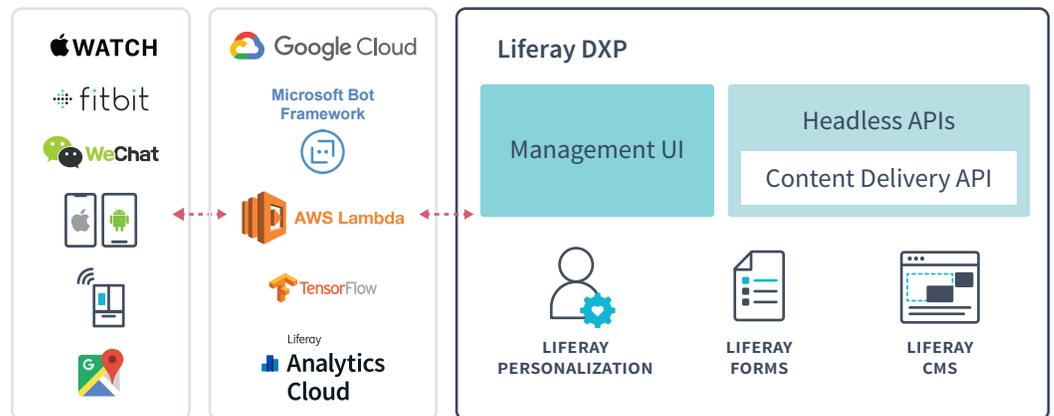


Figure 6 - Examples of different devices, touchpoints, form factors, and services for creating omnichannel experiences with Liferay DXP

3. **Data Integration and Platform Administration** - Automate integration with other systems thanks to OpenAPI support



Figure 7 - Example of back end system integration with other enterprise systems

**DXP Services that are Ready Out-of-the-Box**

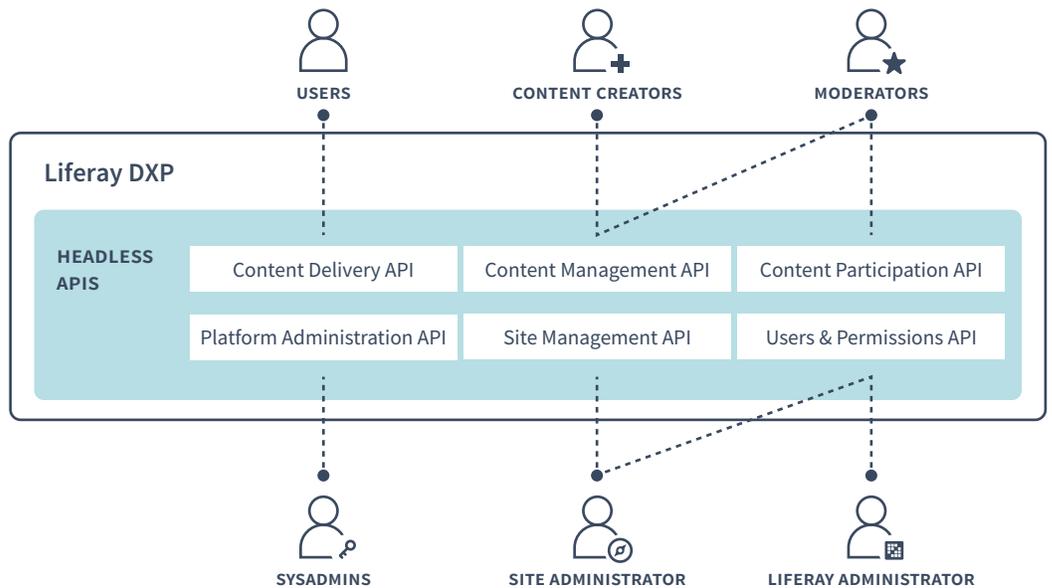


Figure 8

As Liferay DXP evolves and release Service Packs, more and new APIs will become available. With DXP 7.2 GA1, Liferay has taken a research-based approach to focus on the most common use cases to strengthen adoption and be relevant to our community and customers.

## Best Practices

Liferay encourages using DXP in a headless or decoupled mode, especially if there is a requirement that aligns with the three key scenarios described above and depicted in Figure 9.

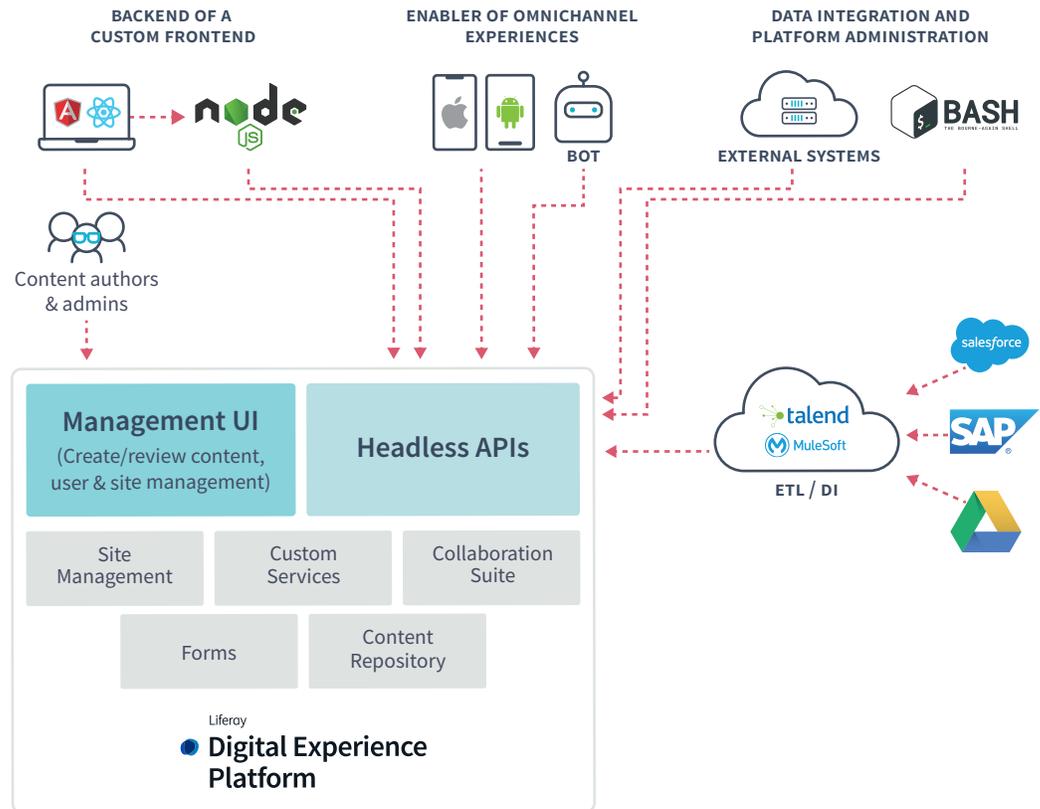


Figure 9 - Examples of implementing Liferay DXP and its APIs in the context of three use cases

As Liferay evolves, support for these key scenarios will grow to include more APIs. It will also make it easier to expand into new scenarios in the future because of the development foundation that has been laid down, bringing expertise in the reuse of Liferay APIs and OpenAPI.

## IoT Integration

As Industry 4.0 continues to disrupt modern enterprises and industries, especially in manufacturing, businesses have to consider how to integrate Internet of Things, that has now evolved beyond home automation devices. Liferay DXP API support does enable integration for IoT and edge devices, but no discussion of IoT integration would be complete without mention of

Kafka and MQTT (Message Queuing Telemetry Transport). Kafka and MQTT are two complementary technologies that together allow enterprises to build IoT end-to-end integration from the edge to data centers running on Liferay, on-premise or in the cloud.<sup>10</sup> MQTT is a lightweight messaging protocol for IoT. Beyond MQTT, if you require more scalability, greater storage and simpler integration to legacy systems, Apache Kafka is an option. Kafka is a highly scalable distributed streaming platform, which ingests, stores, processes and forwards high volumes of data from thousands of IoT devices.<sup>11</sup>

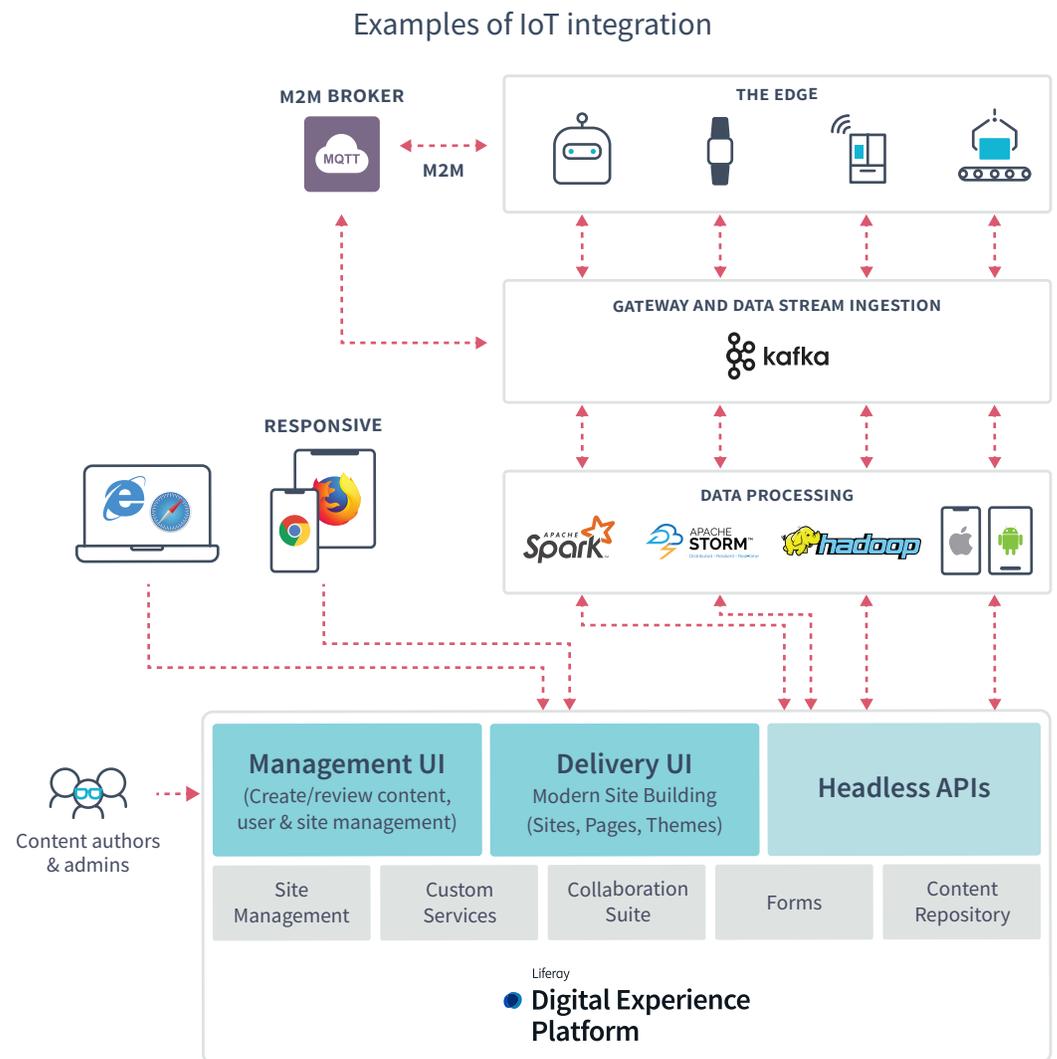


Figure 10 - Example of IoT Architecture

<sup>10</sup> Apache Kafka + MQTT = End-to-End IoT Integration  
<sup>11</sup> Apache Kafka + MQTT = End-to-End IoT Integration

## **EDGE DEVICES**

Edge devices in the IoT world encompass smart devices, sensors, actuators and even routers and gateways. When integrating IoT and the edge with Liferay DXP, it is important to address the ability to process the very large amounts of data and messages that eventually will be aggregated into Liferay. Specifically, adding components to the architecture (See Figure 12) that hasten response times, reduce latency and conserve network resources is essential for reliability and high availability.

## **DASHBOARDS**

As data from the edge and IoT devices is processed and ingested into Liferay DXP, it is made available for display through the Liferay UI. A popular form of integration is to create dashboards that show end users functional collections of data sets that may be used for analysis and monitoring. Liferay DXP exposes its headless APIs so that web content and other assets can be created from ingested IoT data, allowing convenient page and dashboard configuration using Liferay's native web content and web experience widgets. Liferay DXP also allows developers to create custom widgets for unique dashboard UI designs to get and display the aggregated data from the edge.

## **Liferay is the Flexible Choice for IoT**

The IoT landscape is ever-evolving and the array of technology used in an IoT solution can require a change in your enterprise architecture in a matter of months. With Liferay DXP's flexible architecture, headless and decoupled API support and developer-friendly tools, you gain the flexibility to make incremental and modular changes to your technology stack and evolve with the bleeding edge.

## **Best Practices**

In order to successfully integrate IoT devices, enterprises must be prepared to handle the large scale of data collected from the growing amount of IoT devices and its end users. IoT devices have data points that send high-frequency messages and enormous amounts of data and, in aggregate, will create stress on queueing, brokering and processing that data into Liferay DXP.

Real-time data goes through the hot path if it is important to track metrics like temperature, humidity and air quality in real-time to take corrective action. Hot path data points that require real-time processing should utilize Apache Storm and Apache Spark. The cold path is more appropriate for data that is collected over a period of time. These data points are aggregated and analyzed and typically done in batch processes, where Hadoop is in the pipeline for

periodic data analysis. Irrespective of the path that the data points take, they need to be ingested into Liferay DXP.<sup>12</sup> Data that goes through the hot path, such as real-time analytics, can feed into Liferay DXP to be displayed in a dashboard page or site that displays monitoring, system health status, alert notifications and other dynamic information that gets updated frequently. Data that goes through the cold path can be processed into Liferay DXP for reports that are delivered through web content, generated PDF documents and activity feeds that display milestones and checkpoints. Cold path data can also be batch processed to be analyzed by AI/ML services to aid in Liferay DXP personalization and content suggestions.

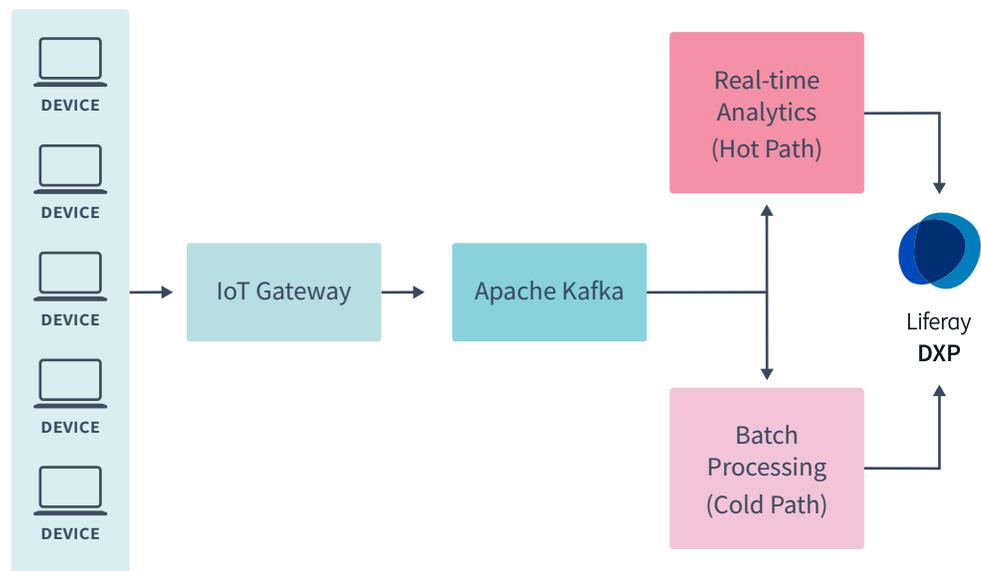


Figure 11 - Hot vs. Cold Paths

<sup>12</sup> Apache Kafka: The Cornerstone of an Internet-of-Things Data Platform

# Native Liferay App and Widget Integration

## Liferay App Development

Liferay DXP is based on the Java platform and can be extended by adding new applications, customizing existing applications, modifying its behavior or creating new themes. You can do this with any programming language the JVM supports, such as Java itself, Scala, jRuby, Jython, Groovy and others. Liferay DXP is lightweight, can be deployed to a variety of enterprise Java containers and app servers and supports a variety of databases. Because of its ability to be customized, you can add support for more app servers or databases without modifying its source code; just develop and deploy a module with the features you need. Here are some of the most common ways of expanding or customizing Liferay DXP features:

- **Developing a new full-blown web application.** The most common way to develop web applications for Liferay DXP is with widgets and portlets, because they integrate well with other existing applications. You are not, however, limited to widgets and portlets if you don't need to integrate your apps with others.
- **Customizing an existing web application or feature.** Liferay DXP is designed to be extended. Many extension points can be leveraged to modify existing behavior and most of these can be developed through a single Java class with some annotations.
- **Creating a new web service** for an external system, a mobile app, an IoT device or anything else.
- **Developing a mobile app** that leverages Liferay as its back-end, which you can write in a fraction of the normal time thanks to Liferay Screens and Liferay Mobile SDK.
- **Developing a custom theme** that adapts the look and feel of the platform to the visual needs of your project. You can leverage your brand and branding assets with ease.

## Summary of Development Tools Within the DXP Platform

You can write code for Liferay using any standard toolset. Liferay is tool-agnostic, rather than pigeonholing you into something specific. This frees you to work with whatever you're already productive using. Liferay has also created its own tools that streamline Liferay DXP development. These tools integrate with popular build environments (e.g., Gradle, Maven and NodeJS). They include:

- [Blade CLI](#): a command line interface used to build and manage Liferay Workspaces and Liferay Portal projects. This CLI is intended for Gradle or Maven development.
- [Liferay Workspace](#): a generated Gradle/Maven environment built to hold and manage Liferay Portal projects.
- [Liferay Dev Studio](#): an Eclipse-based IDE supporting development for Liferay Portal.
- [Liferay IntelliJ Plugin](#): a plugin providing support for Liferay Portal development with IntelliJ IDEA.
- [Liferay Theme Generator](#): a generator that creates themes, layouts templates and themelets for Liferay Portal development.
- [Liferay JS Generator](#): a generator that creates JavaScript portlets with JavaScript tooling.

Liferay also provides a plethora of [Gradle](#) and [Maven plugins](#) you can apply to your projects. Many of these are already built into tools such as Liferay Workspace.

If you're a newbie looking for the best development tool for Liferay Portal, or even a seasoned veteran looking for a tool you may like more than your current setup, this section answers your tooling questions.

## Front End Development for UI/UX

You have complete [front-end development](#) freedom. You can use Liferay DXP front-end frameworks, along with the front-end technologies you love the most:

- EcmaScript ES2015+
- React, Angular, Vue, etc.
- [Metal.js](#) (developed by Liferay)
- [AlloyUI](#) (developed by Liferay)
- jQuery (included)
- Lodash (included, but disabled by default)

To load modules, you must know when they are needed, where they are at build time, whether they should be bundled together or loaded independently and you must assemble them at runtime. Liferay's Loaders (YUI/AUI, AMD, and npm in AMD format) handle loading for you.

The Liferay JS Bundle Toolkit ([JS Portlet Extender](#), [Liferay Bundle Generator](#), [liferay-npm-bundler](#)) has the tools you need to create and develop JavaScript portlets with pure JavaScript tooling. You can use the [liferay-npm-bundler](#) to bundle npm packages in your applications. It even has several presets for common module types (AMD, React, Angular, etc.) to save you time. It creates an OSGi bundle for you, extracts all npm dependencies and transpiles your code for the Liferay AMD Loader.

While developing JavaScript applications, you may need to access Liferay DXP-specific information or web services. The Liferay global JavaScript Object [exposes this information for you](#) to use in your JavaScript applications.

## Examples and sample projects for native apps and widgets that integrate

Want samples or predefined project templates? Liferay has you covered with 30+ [project templates](#) and many more [project samples](#).

## Advanced Integration Techniques

Liferay provides even more integration options when the basics don't work. For example, many enterprise data systems are integrated using some form of a messaging bus, such as an ESB, or even in conjunction with Liferay's own message bus, web services or headless and decoupled architectures. These messaging solutions deal with network fault tolerance, differing application data interchange formats and the inevitable change that comes with a long-lived enterprise system. Liferay can integrate with these systems and proxy enterprise data into your Liferay environment for further processing.

# Summary

Through acquisition, organizational changes and environmental factors, today's enterprises are a heterogeneous mixture of legacy applications, enterprise data warehouses, emerging internet touchpoints, edge devices and IT staff. Coordinating these resources and creating a cohesive user experience is necessary to realize the full value of organizational assets.

All but the most basic of enterprise use cases require some form of integration to be truly useful. Integration cost, complexity and effectiveness should be evaluated and measured against goals before deciding how to proceed with an integration. In some cases, a simple presentation-oriented integration will cover 80% of the use cases, and be "good enough" to not justify the cost of a more complex undertaking. In other cases, a deeper integration of products, portals and DXPs is necessary to achieve the desired experience, or to take advantage of features already offered by the products being integrated.

Liferay DXP is in a unique position to offer a single point of access and administration for organizations' data, content and information, whether originating from Liferay itself, from existing in-house applications (ERP, CRM, etc), or from the edge. Liferay's flexible and modern architecture exposes integration points at many levels, including presentation, web services (SOAP and JSON) and APIs in a headless or decoupled architecture. As long as your existing applications are open and provide support through some service layer or API, Liferay can integrate with those applications. There is a range of options for application integration depending on your needs, from Liferay Marketplace components that are ready to deploy, conventional web services and a Liferay IFrame portlet for lighter integration scenarios, to headless and decoupled APIs for unique UI frameworks and edge devices, or native application widgets for more complex integration scenarios.

## Resources

For additional information on Liferay integration options, please [access Liferay DXP documentation here](#).

## Moving Forward

Liferay offers enterprise application integration and architecture assistance for your system. Since integrations vary from system to system, Liferay recommends a thorough evaluation of your environment and business requirements from our consulting team, Liferay Global Services, before applying any of the techniques mentioned in this paper.

Learn more about Liferay Global Services at [liferay.com/consulting](https://liferay.com/consulting).

Request a customized demo on how Liferay DXP can help grow your digital business at [liferay.com/request-a-demo](https://liferay.com/request-a-demo) or contact [sales@liferay.com](mailto:sales@liferay.com).



Liferay makes software that helps companies create digital experiences on web, mobile and connected devices. Our platform is open source, which makes it more reliable, innovative and secure. We try to leave a positive mark on the world through business and technology. Hundreds of organizations in financial services, healthcare, government, insurance, retail, manufacturing and multiple other industries use Liferay. Visit us at [liferay.com](https://www.liferay.com).

© 2021 Liferay, Inc. All rights reserved.