

# Identify and Evaluate Your Next Low-Code Development Technologies

Published 13 April 2021 - ID G00742070 - 10 min read

By Analysts [Paul Vincent](#), [Jason Wong](#), [Saikat Ray](#), [Akash Jain](#), [Keith Guttridge](#), [Kimihiro Iijima](#), [Adrian Leow](#), [Yefim Natis](#)

Initiatives: [Software Engineering Technologies](#); [Software Engineering Strategies](#)

Low-code development tools promise success in application democratization, workflow automation and composite development. Software engineering leaders should use this research to understand the low-code development tool landscape, and evaluate solutions that meet their strategic and tactical needs.

## Overview

### Key Findings

- The low-code paradigm applies to many automation and application capabilities, which can cause software engineering leaders problems in identifying the specific technologies and automation types they need.
- Software engineering leaders are faced with complex overlapping offerings across multiple low-code development technology markets when considering technology purchases.
- Procurement decisions for low-code development technologies can result in multiple expensive platform subscriptions, technical debt and inefficient platform use leading to early and disruptive platform replacement cycles.

### Recommendations

As a software engineering leader looking to evaluate and select a low-code application development technology, you should:

- Identify the specific low-code technologies that support your productivity and time-to-market needs, by analyzing the specific application, automation, integration and workflow requirements.
- Select your desired low-code platform types and markets by cross-referencing the Gartner low-code technologies and platform classification against these technology needs.
- Evaluate candidate low-code platform offerings to meet your strategic and tactical platform requirements by applying Gartner's business, technology, developer and ecosystem criteria.

## Introduction

Low-code development is an increasingly proven approach for increasing application developer productivity, reducing development times and enabling composable business strategies. Low-code development tools also enable democratized and self-service application development for business and citizen developers, as well as help scale business automation and hyperautomation across enterprises.

These potential benefits have led to a boom in low-code tooling and platforms across all aspects of application development and automation, with Gartner tracking over 400 low-code vendors. Many hundreds of vendors, ranging from cloud hyperscalers to vertical market specialists, are competing for your business.

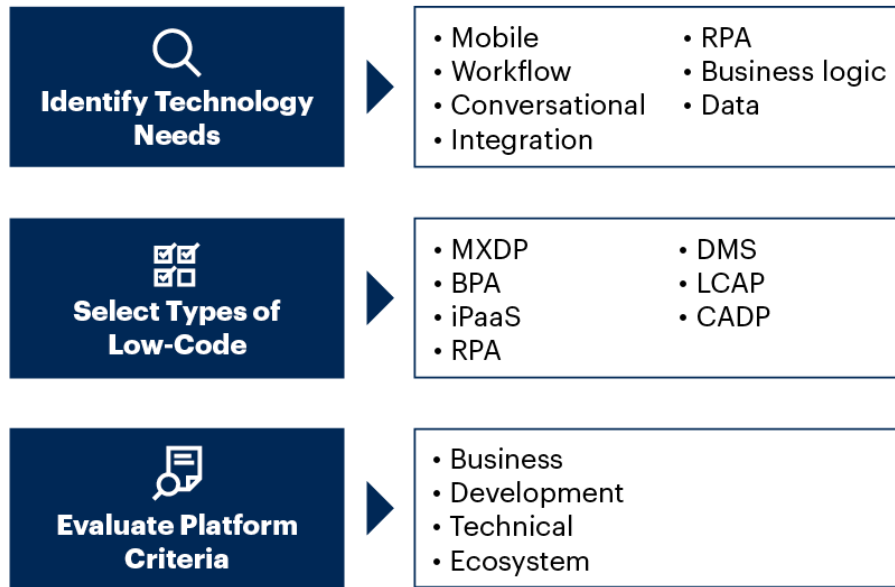
The increasing availability and scope of low-code development technology means that enterprises need to identify and select from a proliferating low-code platform landscape. These platforms need to be managed and used for the correct uses cases in order to maximize utilization and minimize subscription costs.

So, how do you identify the gaps in your low-code development portfolio, and the potential low-code offerings that can fill those gaps?

Gartner's analysis identifies the low-code development technologies available, the platform markets that have evolved to service these and the evaluation criteria to consider. These technologies, markets and evaluation criteria continue to evolve (see Figure 1) and satisfy a growing number of enterprise use cases.

Figure 1: Evaluation Process for Low-Code Development Technologies

## Evaluation Process for Low-Code Development Technologies



Source: Gartner  
742070\_C

Gartner®

## Analysis

### What Low-Code Technologies Do You Need?

Low-code development platforms support a variety of application technologies. Choosing a platform requires software engineering leaders to understand their application requirements against low-code technology components. The low-code approach reduces skill sets for development and related requirements across a wide range of application development technologies through higher-code abstraction levels, the use of visual metaphors and guided developer tools.

Common low-code technology components (mapped in Figure 2), some or all of which you may require, include:

- User interface capabilities — The development tooling can include support ranging from fixed application user interfaces, communication mechanisms such as email and messaging, simple web and mobile forms, to multiexperience including artificial intelligence (AI)-driven chatbots, augmented reality and wearables. Low-code platform types vary in their creation of end-user interfaces, ranging from none in integration platforms as a service (iPaaS) and limited in many business process automation (BPA) tools, to multiexperience development platforms (MXDPs) that specialize in user experience (UX) capabilities.

- Data capabilities — Development technologies will usually include data modelling, database or data access technology or master data management features. Other types of data, such as documents and other content, may also be provided. Prebuilt data schemas may be provided for domain or SaaS compatibility. Many (low-code) platforms only embed a database for their cloud-based offerings, while others require the use of existing data sources.
- Process and workflow capabilities — Various types of “processes” may be supported through orchestration or workflow tooling, these may include:
  - Business process automation of workflows
  - Processing data and information flows across different users for business transactions
  - Managing the flow of communications in a call center as workforce workflow management
  - Managing the extension of business processes into controlled “knowledge work” as a part of case management

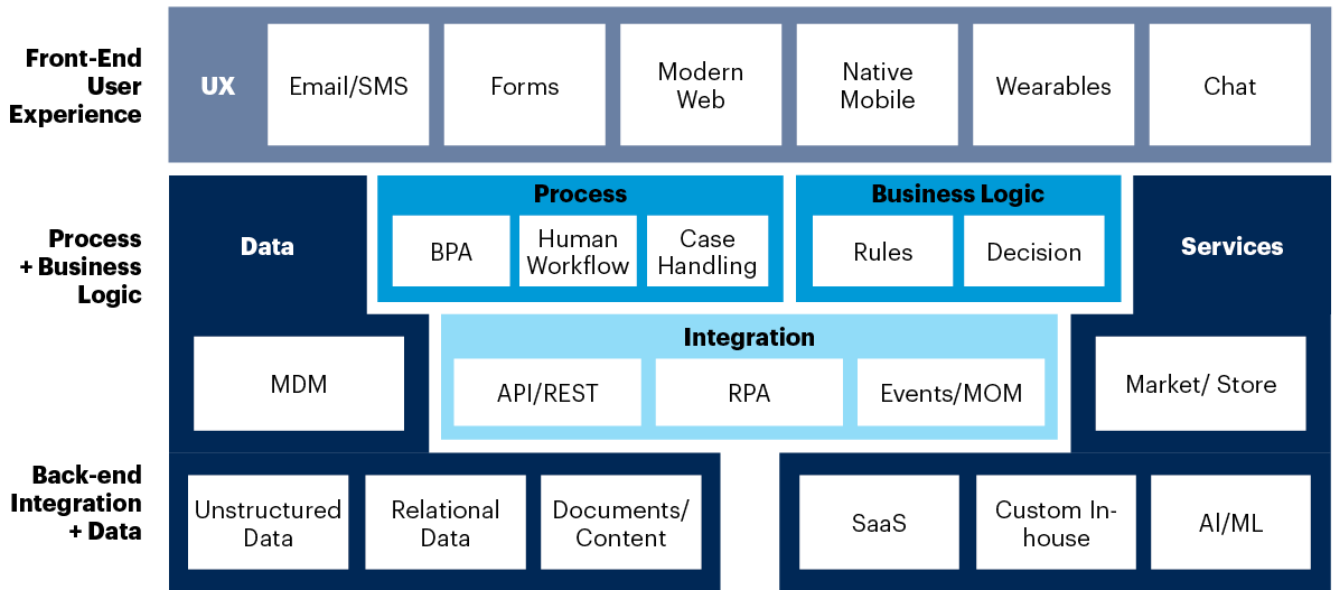
Orchestrated processes may also control the UX, the flow of data across a business process, or the flow of control across an robotic process automation (RPA) bot or integration transaction.

- Business logic capabilities — Beyond simple scripting of business rules, development tools may explicitly support specific types of business rules and rulesets and their execution, or as decision-based organizations for business logic. Decision automation is often extended to include AI and analytics services generating business rules or predictive models (see the [Magic Quadrant for Analytics and Business Intelligence Platforms](#) for details).
- Integration capabilities — Many information sources required in applications exist in external components that need to be accessed via APIs. Other APIs may provide access to event-based APIs or event brokers. For some legacy applications, access through APIs may not be available or accessible and instead require the use of RPA bots to provide access. Integration capabilities are key to the support of composability across packaged business capabilities, packaged integration processes (see [Accelerate Your Integration Delivery By Using Packaged Integration Processes](#)), SaaS APIs including embedded integration (see [Quick Answer: When to Use \(or Not to Use\) Embedded Integration Features Provided by Your SaaS Vendor](#)), and any custom services supporting your business.
- Services — Low-code development technologies may include various prebuilt services specific to a business or application domain. These services may be blackbox, or include customizable definitions and models. They may also be provided as cloud services that abstract low-level services such as AI/machine learning (ML) for specific use cases. Catalogs of services or components may be available as an application or app store. These may also be monetized and provide a way for your own components and services to be shared internally or sold externally.

The two main dimensions of low-code technologies are front-end UX complexity, and back-end composability (see Figure 2).

Figure 2: Low-Code Technology Capabilities Map

### Low-Code Technology Capabilities Map



Source: Gartner  
742070\_C



### Select Low-Code Product Types of Interest

The various types of low-code development tools provide different levels of support for the different capabilities defined above, primarily around general capabilities (UX, process orchestration, data support and API integration) but also with some specialist capabilities (multiexperience, decisions/rules and RPA integration via bots). The most common differentiators for low-code development tools (see Figure 1) are around developer profile (see Evaluation Criteria section below) and the requirement for front-end (UX) versus back-end (process/service) support.

While the definitions of these low-code development platform types map on to specific capabilities (see Figure 3), vendors often support multiple capabilities across platform types. Such multifunction platforms provide rich support across more use cases in exchange for higher costs.

Figure 3: Feature Map by Type of Low-Code Development Platform

### Feature Map by Type of Low-Code Development Platform

● Required ● Expected ● Optional

Platform Types		MXDP	LCAP	CADP	BPA	DMS	RPA	iPaaS
Front-End User Interface	Advanced User Experience	●	●	●	●	●	●	●
	Basic User Experience (Web and Mobile Forms)	●	●	●	●	●	●	●
Process + Business Logic	Process Orchestration	●	●	●	●	●	●	●
	Decisions/Rules	●	●	●	●	●	●	●
Back-end Integration + Data	Data (User database)	●	●	●	●	●	●	●
	Integration via Bots	●	●	●	●	●	●	●
	Integration to APIs Process + Connectors	●	●	●	●	●	●	●

Source: Gartner

Note: Multi-experience Development (MXDP); Low-Code Application Platform (LCAP); Citizen Automation & Development (CADP); Business Process /Workflow Automation; Decision Management Suites (OMS); Robotic Process Automation (RPA); Integration Platform-as-a- Service (iPaaS)

742070\_C

The core types of low-code development technology platforms requested by Gartner clients has evolved to:

- Multiexperience development platforms (MXDP)
- Low-code application platforms (LCAP)
- Citizen automation and development platforms (CADP)
- Business process/workflow automation platforms (BPA)
- Decision management suites (DMS)
- Robotic process automation platforms (RPA)

- Integration platform as a service (iPaaS)

## Assess Tools using Low-Code Evaluation Criteria





Software engineering leaders need to consider any investment in low-code development platforms carefully to optimize the selection, cost and utilization of these platforms. This is especially true when one considers the potential life span for applications built on these platforms and the attendant technical debt if a platform change is required later; along with the lack of standards for interchange of applications or their components.

Evaluation criteria across all platform types can be considered in four areas (see Figure 4): business, technology, development and ecosystem. All of these should be considered, although often technology-fit, developer-fit and cost are often the highest priorities, with developer governance a common area of concern:

- Business criteria typically include whether the platform is to be used for a tactical use case or strategic usage, whether the vendor is already a supplier (such as a preexisting strategic SaaS vendor for the organization), vendor risk (especially if a new or smaller vendor), amount of lock-in (related standards support, if any), cost fit (such operating expenditure [opex] or capital expenditure [capex]), and pricing model fit (such as resource-based pricing for B2C use cases).
- Technology criteria include whether and what UX is required, ratio of front end to back end needed, amount of composability support from workflows to API integrations as well as API creation capabilities, architecture constraints (such as support for scalability or events consumption), and deployment models whether on-premises, cloud or both (or other specialist models like Internet of Things [IoT]/edge).
- Development criteria include the developer profile required (including fusion team support), developer governance needed (often related to the developer profile), application governance required (including security), predefined or prebuilt domain support such as industry processes or data models, and design support (such as user interface design systems).
- Ecosystem criteria include vendor support models and quality, customer ecosystem opportunities (such as shared development of standard components within the enterprise), partner ecosystem requirements (such as contractor availability or turnkey options), learning ecosystem support (such as training and certification services), and third-party marketplace support (whether monetized or source-provided models).

Figure 4: Evaluation Criteria for Low-Code Development Tools

### Evaluation Criteria for Low-Code Development Tools

 <b>Business</b>	 <b>Technology</b>	 <b>Development</b>	 <b>Ecosystem</b>
<ul style="list-style-type: none"> <li>• Planned Usage Type</li> <li>• Vendor Fit/Risk</li> <li>• Lock-In/Standards</li> <li>• Cost Fit</li> <li>• Pricing Model Fit</li> </ul>	<ul style="list-style-type: none"> <li>• User Experience</li> <li>• Front- versus Back-End</li> <li>• API &amp; Composability</li> <li>• Architecture</li> <li>• Deployment</li> </ul>	<ul style="list-style-type: none"> <li>• Developer Profile</li> <li>• Developer Governance</li> <li>• Application Governance</li> <li>• Domain Support</li> <li>• Design System</li> </ul>	<ul style="list-style-type: none"> <li>• Vendor Support</li> <li>• Customer Ecosystem</li> <li>• Partner Ecosystem</li> <li>• Learning Ecosystem</li> <li>• Component Marketplace</li> </ul>

Source: Gartner  
742070\_C

For a more detailed technology and development evaluation criteria refer to Gartner for Tech Professionals evaluation criteria for the [Solution Comparison for Low-Code Application Platforms](#) (summarised in Table 1).



**Table 1: Expanded Evaluation Criteria for Tech Professionals**

Enlarged table in Appendix

Technology	Development
<b>End-user experience</b> UI design and frameworks	<b>Governance</b> Application life cycle management Application approvals Integration controls Application and platform metrics Security and access controls Application and model versioning Use-case licensing Developer access and onboarding
<b>Integration</b> Data access Application connectors API integration Event-driven capabilities	<b>Developer and operations experience</b> Developer personas Data validation Developer and operations tools DevOps methodologies Sandbox environments
<b>Architectural considerations</b> Platform scalability Hosting model Exposing applications as APIs Portability and extensibility Disaster recovery Offline capabilities Application testing	

Source: Gartner (April 2021)

## Evidence

Gartner surveys vendors in low-code development technologies annually to appraise the state of the low-code market which, in 2021, includes not only LCAP, MXDP and iPMS/BPA, but also RPA and iPaaS as these are increasingly used in a low-code way. The 2020 vendor survey was completed by 120 vendors including: Agile Labs, AgilePoint, AINS, Alemba, Allwork.Space, Alphinat, AmpleLogic, AnyData Solutions, AppGyver, Appian, AppPoint, Appspotr, Appway, Arise, Arrayworks, ASG, atfinity, AuraQuantic, Avalant, Aware IM, Axway, Barium, Betty Blocks, BP Logix, Bryter, Bubble.io, Bubblz.net, Cflow, ClaySys, ClearString, Convertigo, Cotham Technologies, Creatio, datb, Dell Boomi, Discus Solutions, DronaHQ, FAT FINGER, Faveod, Fliplet, Flowfactory, ForeSoft, Genesis, GeneXus, Graphite GTC, Grexx, GW Apps, Hyland, i-exceed, Infinite Blue, Innoveo, Intellileap, Intense Technologies, Ionic, Ivanti (Cherwell Software), Jadu, JobRouter, Kintone, Lianja, LogicNets, Mendix, MicroPact, Microsoft, MobileFrame, Mobilous, monday.com, Neptune Software, Netcall, Newgen, Nintex, Novacura, NTT Data Intramart, Open as App, OpenDraft, OpenText, Oracle Netsuite, Oracle APEX, Oracle VBCS, OrangeScape, OutsyStems, Pega,

Plasmacomp, PMG, PNMSOft, ProcessMaker, Progress, ProntoForms, QuickBase, ReadINow, Rintagi, Salesforce, SAP, Sapho, Scopeland, SER Group, Servoy, SIB Visions, Simplicité, Simplifier, Sikiwis, Skyve, SmartApps, Smartflow, Smartsheet, Snappi, Softengi, Softools, SugarCRM, Synchronoss, Temenos, Thinkwise, TipoTapp, TOTVS, TrackVia, Tulip, Ultimius, Vantiq, WebRatio, Webcon, WorkFlowWise, IngageAPP (Xiaoshouyi), XOne, Zoho, Zudy.

## Acronym Key and Glossary Terms

AI	artificial intelligence
API	application programming interface
AR	augmented reality
B2C	business to consumer
BPA	business process automation
CADP	citizen automation and development platform
capex	capital expenditure
DMS	decision management suite
iBPMS	intelligent business process management suite
IoT	Internet of Things
iPaaS	integration platform as a service
LCAP	low-code application platform
ML	machine learning
MXDP	multiexperience development platform
opex	operating expenditure
RAD	rapid application development
RPA	robotic process automation
SaaS	software as a service
UX	(end) user experience

## Note 1. Example Vendors for Low-Code Technologies

See Table 2 for sample vendors across the different low-code development platform types, including some of the more specialist types.

**Table 2: Example Vendors for Low-Code Technologies**

Enlarged table in Appendix

Acronym	Platform Role	Example Vendors
MXDP	Multiexperience development	GeneXus, HCL, Neptune, Progress, Oracle, SAP, XOne
LCAP	Low-code application development	Mendix, Oracle, OutSystems, Salesforce, ServiceNow
CADP	Citizen automation and development	Betty Blocks, Microsoft Power Apps, Process Street, QuickBase
BPA	Business process automation	Appian, Bizagi, Bonitasoft, Camunda, Nintex, Pegasystems
DMS	Decision automation	Actico, Experian, FICO, Sapiens, OpenRules, Red Hat, Trisotech
iPaaS	Integration automation	Boomi, Informatica, Microsoft Azure Logic Apps, Mulesoft, Workato
RPA	Robotic process automation	Automation Anywhere, Blue Prism, Kofax, UiPath, WorkFusion
	Content process automation	Hyland OnBase, Newgen, OpenText
	Document processing automation	Hypatos, SuccessData
	Workforce workflow management	IBM, SoftwareAG, TIBCO
	Case management	AINS, Eccentex, MicroPact
	Rapid mobile application development	Alpha Software, Skuid
	RAD code generators	Graphite GTC, Scopeland

Source: Gartner (April 2021)

## Recommended by the Authors

[Magic Quadrant for Enterprise Low-Code Application Platforms](#)

[Magic Quadrant for Enterprise Integration Platform as a Service](#)

[Magic Quadrant for Robotic Process Automation](#)

[Magic Quadrant for Multiexperience Development Platforms](#)

[Market Guide for Intelligent Business Process Management Suites](#)

[How to Choose Your Best-Fit Decision Management Suite Vendor](#)

[Quick Answer: What Is the Difference Between No-Code and Low-Code Development Tools?](#)

[Top Strategic Technology Trends for 2021: Hyperautomation](#)

[Solution Comparison for Low-Code Application Platforms](#)

## Decision Point for Process Automation Platforms

© 2021 Gartner, Inc. and/or its affiliates. All rights reserved. Gartner is a registered trademark of Gartner, Inc. and its affiliates. This publication may not be reproduced or distributed in any form without Gartner's prior written permission. It consists of the opinions of Gartner's research organization, which should not be construed as statements of fact. While the information contained in this publication has been obtained from sources believed to be reliable, Gartner disclaims all warranties as to the accuracy, completeness or adequacy of such information. Although Gartner research may address legal and financial issues, Gartner does not provide legal or investment advice and its research should not be construed or used as such. Your access and use of this publication are governed by [Gartner's Usage Policy](#). Gartner prides itself on its reputation for independence and objectivity. Its research is produced independently by its research organization without input or influence from any third party. For further information, see "[Guiding Principles on Independence and Objectivity](#)."

Table 1: Expanded Evaluation Criteria for Tech Professionals

Technology	Development
<b>End-user experience</b> UI design and frameworks	<b>Governance</b> Application life cycle management Application approvals Integration controls Application and platform metrics Security and access controls Application and model versioning Use-case licensing Developer access and onboarding
<b>Integration</b> Data access Application connectors API integration Event-driven capabilities	<b>Developer and operations experience</b> Developer personas Data validation Developer and operations tools DevOps methodologies Sandbox environments
<b>Architectural considerations</b> Platform scalability Hosting model Exposing applications as APIs Portability and extensibility Disaster recovery Offline capabilities Application testing	

Source: Gartner (April 2021)

Table 2: Example Vendors for Low-Code Technologies

Acronym	Platform Role	Example Vendors
MXDP	Multiexperience development	GeneXus, HCL, Neptune, Progress, Oracle, SAP, XOne
LCAP	Low-code application development	Mendix, Oracle, OutSystems, Salesforce, ServiceNow
CADP	Citizen automation and development	Betty Blocks, Microsoft Power Apps, Process Street, QuickBase
BPA	Business process automation	Appian, Bizagi, Bonitasoft, Camunda, Nintex, Pegasystems
DMS	Decision automation	Actico, Experian, FICO, Sapiens, OpenRules, Red Hat, Trisotech
iPaaS	Integration automation	Boomi, Informatica, Microsoft Azure Logic Apps, Mulesoft, Workato
RPA	Robotic process automation	Automation Anywhere, Blue Prism, Kofax, UiPath, WorkFusion
	Content process automation	Hyland OnBase, Newgen, OpenText
	Document processing automation	Hypatos, SuccessData
	Workforce workflow management	IBM, SoftwareAG, TIBCO
	Case management	AINS, Eccentex, MicroPact
	Rapid mobile application development	Alpha Software, Skuid
	RAD code generators	Graphite GTC, Scopeland

Source: Gartner (April 2021)