

# 2021低代码市场研究报告

旭日初升 臻于至善

Copyright reserved to EqualOcean Intelligence, July 2021

```
mirror_mod = modifier_ob.modifiers.new("MIRROR_X")
mirror_mod.mirror_object = mirror_ob

class MIRROR_X:
    mirror_mod.use_x = True
    mirror_mod.use_y = False
    mirror_mod.use_z = False

class MIRROR_Y:
    mirror_mod.use_x = False
    mirror_mod.use_y = True
    mirror_mod.use_z = False

class MIRROR_Z:
    mirror_mod.use_x = False
    mirror_mod.use_y = False
    mirror_mod.use_z = True

Selection at the end -add back to scene
mirror_ob.select= 1
mirror_ob.select=1
context.scene.objects.active = mirror_ob
```

# 前言

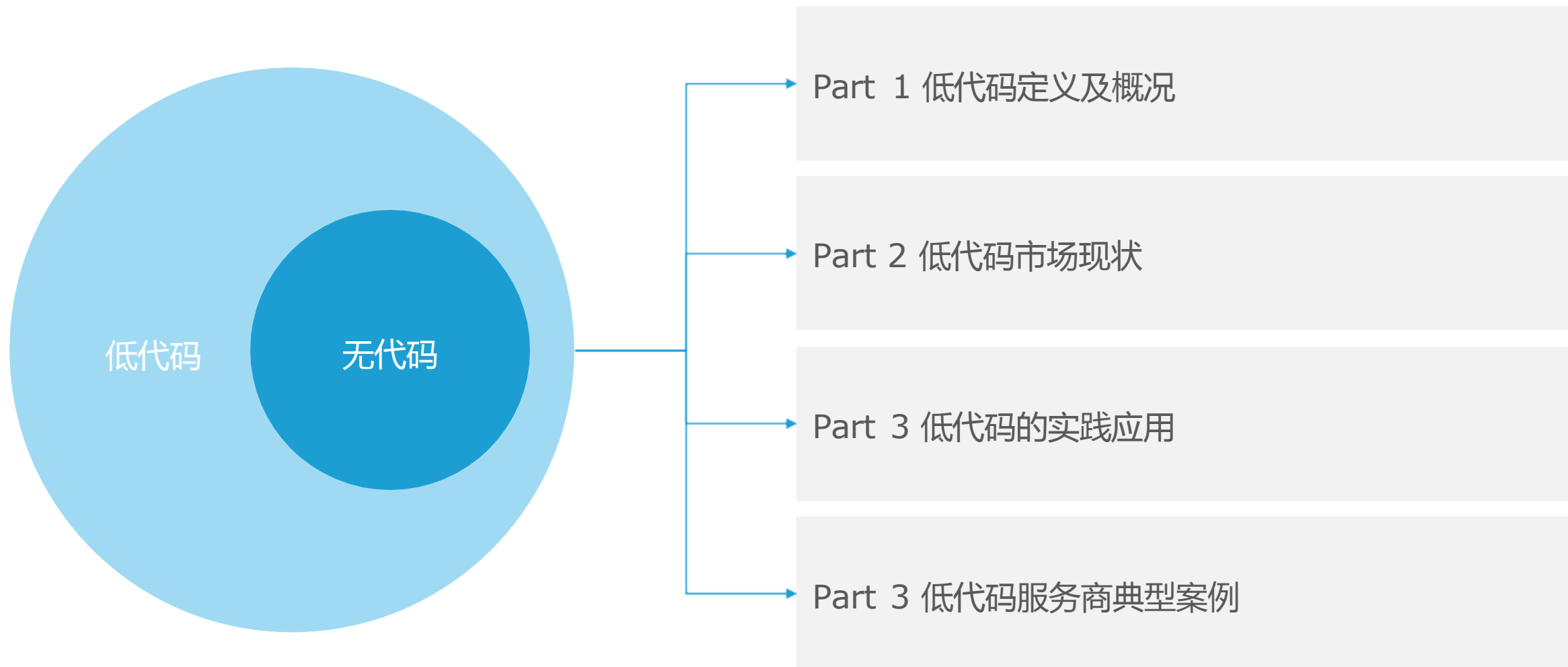
早在1999年，Salesforce便打出“软件终结者”的名号面市软件市场。随后，2014年，Gartner正式定义低代码平台。2018年，outsystem 融资以及西门子收购Mendix两件大事件将低代码平台正式带入大家视野。

疫情加快各方数字化转型的步伐，而低代码平台则起到了改变软件作业模式，实现企业数字化转型的作用。然而，**从低代码平台供给方分析，市场仍处于非常早期的阶段，低代码服务商还需要不断地探索以抢占未来市场份额。**

本篇报告将就低代码的概念及定义、低代码市场现状、低代码的实践应用以及低代码服务商典型案例四个方面阐述低代码市场的现状以及实践应用效果，并且对低代码服务商进行详尽分类及特点分析。

# 报告研究范围界定及主要结构

◆本篇报告研究的低代码市场包含低代码（需少量代码编写的快速开发平台）以及无代码（无需代码编写的快速开发平台）市场。研究结构包含低代码概念及定义、市场现状、应用实践及典型企业研究。



```
mirror_mod = modifier_ob.modifiers.new("Mirror")
# Add mirror object to mirror_ob
mirror_mod.mirror_object = mirror_ob

# MIRROR_X:
mirror_mod.use_x = True
mirror_mod.use_y = False
mirror_mod.use_z = False
# MIRROR_Y:
mirror_mod.use_x = False
mirror_mod.use_y = True
mirror_mod.use_z = False
# MIRROR_Z:
mirror_mod.use_x = False
mirror_mod.use_y = False
mirror_mod.use_z = True

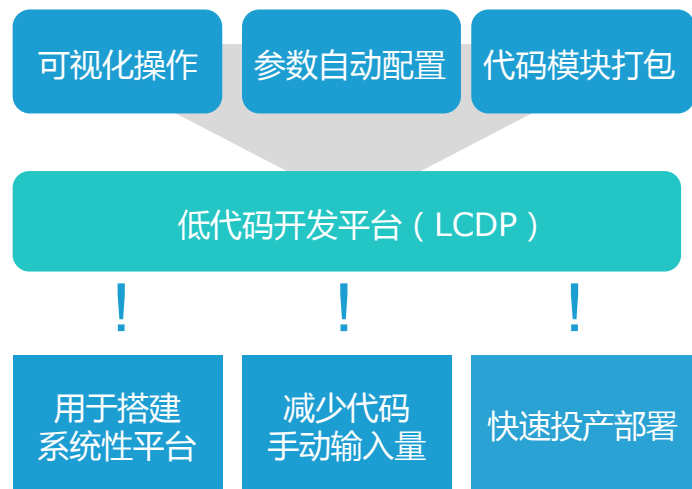
# Selection at the end of the test
mirror_ob.select= 1
mirror_ob.select=1
context.scene.objects.active = modifier_ob
```

# 低代码定义及概况

# 低代码平台是一种新型开发平台，起到提升开发生产力的效果

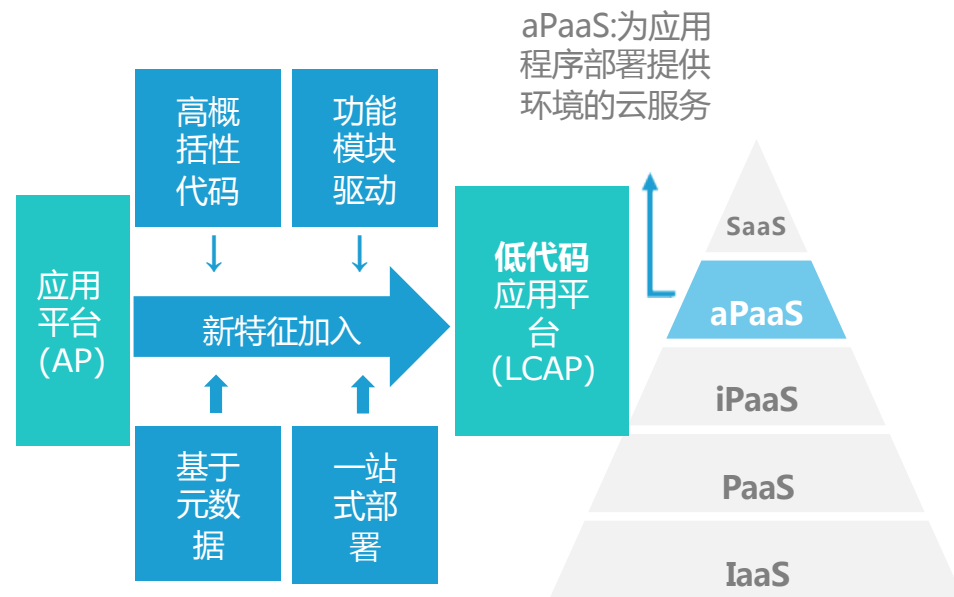
◆低代码平台这一概念最早由Forrester Research于2014正式提出，Gartner随后用基于aPaaS的高生产力平台（hpaPaaS）来命名这一品类，Microsoft、Mendix等深耕低代码头部企业也基于自身产品特点与技术路线解释何为低代码平台。

## 01. Forrester Research



Forrester：低代码开发平台指可通过最少的手工编程就能快速交付应用程序，并能快速设置和部署用于参与系统的平台。

## 02. Gartner



Gartner：低代码应用平台是一个应用程序平台，它使用声明性的高级抽象程序以及一步式部署来支持快速的应用程序开发，部署，执行和管理；hpaPaaS是aPaaS的一种形式，提供了平台支持云应用程序快速高效开发，部署和执行。

## 03. Microsoft

◆低代码开发的目的：  
人人皆可上手；帮助开发者加速开发投用

◆低代码开发的特点：  
直觉的、可视化的、几乎不需要编程技能

◆低代码平台应具备的功能：  
可视化界面、连接后端、  
内置应用生命周期管理功能

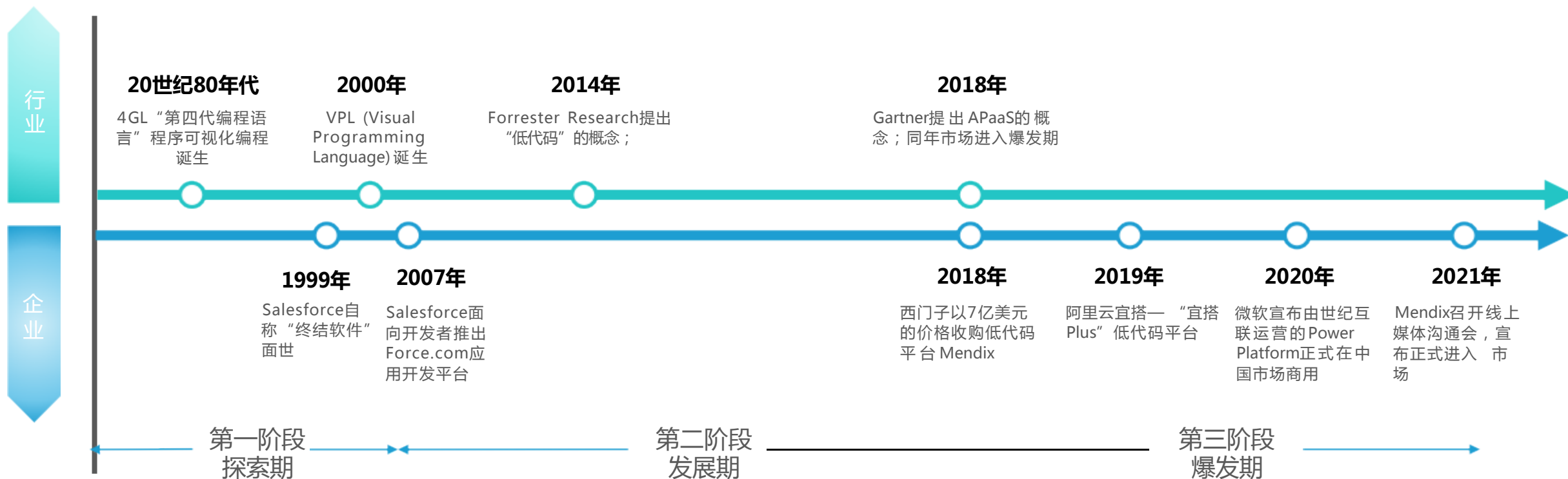
## 04. Mendix

低代码开发是一种可视化应用开发方法。通过低代码开发，不同经验水平的开发人员能够通过图形用户界面，使用拖放式组件和模型驱动逻辑来创建 Web 和移动应用。

# 低代码平台发展多年，近年成为市场热点

◆低代码开发平台起始于20世纪80年代4GL“第四代编程语言”。

◆2014年，Forrester research 提出低代码平台的概念，低代码市场进入发展期；2018年，西门子收购低代码企业Mendix、美国低代码独角兽企业Outsystem获得1.5亿美元的融资，此次收购事件以及融资事件的发生将低代码市场带入资本方的视野，因此，低代码市场进入市场快速发展期。



# 低代码平台与无代码平台皆能降低开发门槛、快速响应业务需求，但两种开发平台性能和面向人员有所差别

- ◆ 亿欧智库认为低代码平台是指通过模型驱动的技术路径和可视化的用户界面，实现基于封装组建的通用模块，编写少量代码进行调用即可完成应用程序搭建的开发平台；无代码平台则是无需任何代码便可实现应用程序创建的开发平台。
- ◆ 低代码平台仅需编写少量代码便可搭建各种应用系统，而无代码平台则无需编写代码，便可快速构建业务应用系统；因此低代码平台与无代码平台分别面向开发人员以及非开发人员。

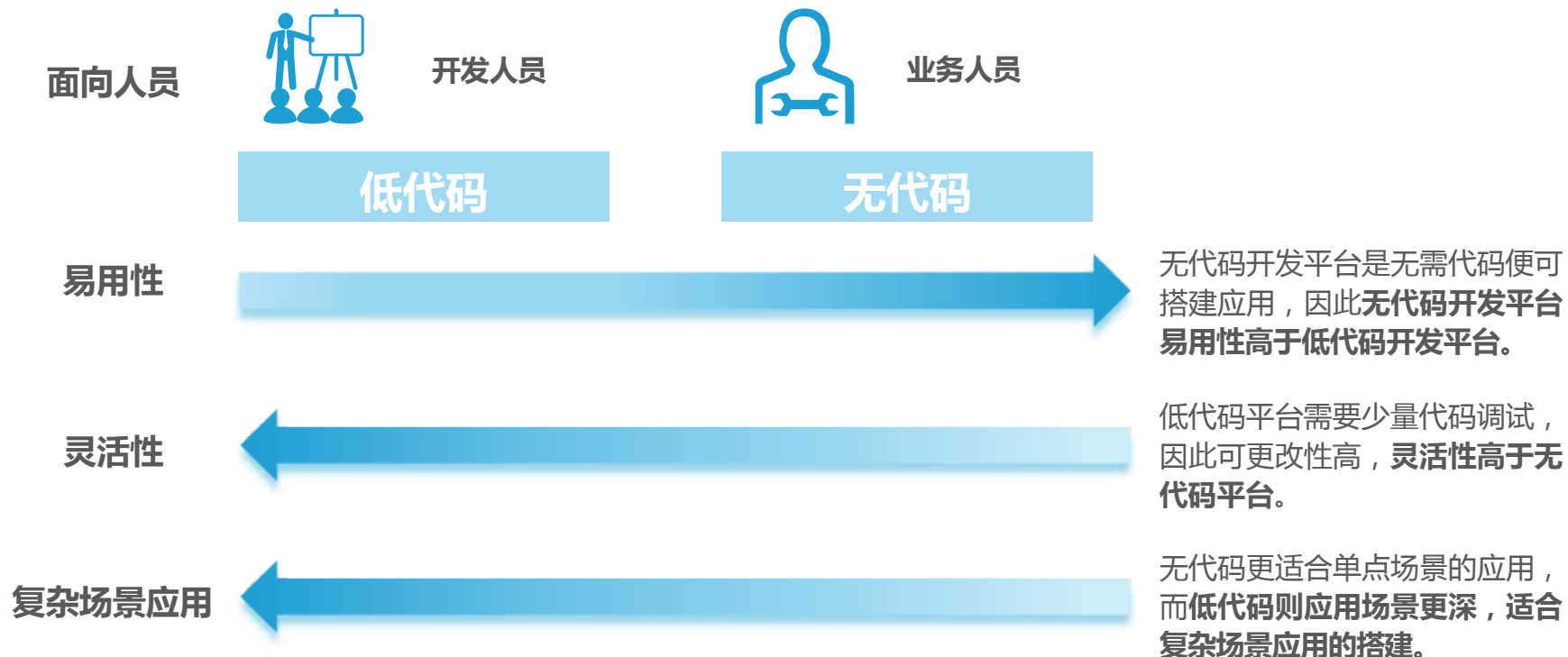
亿欧智库：低代码与无代码相同点

01 开箱即用

02 可视用户界面

03 提升开发效率

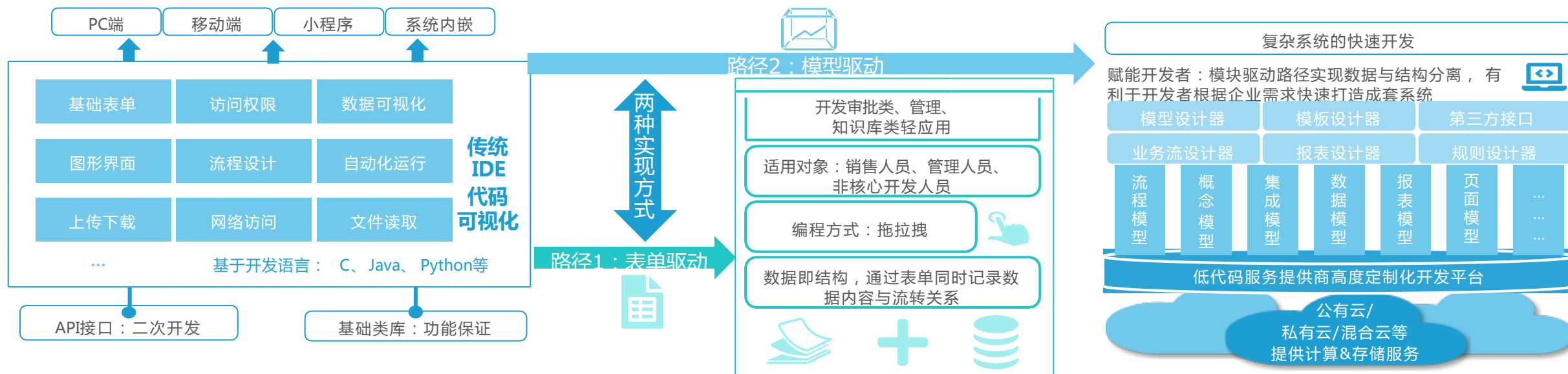
亿欧智库：低代码与无代码不同点



# 模型与表单驱动是低代码平台两种主要技术路径

- ◆低代码平台源自于集成开发环境（Integrated Development Environment，IDE）的可视化、模块化与集成化，避免大量代码输入带来的重复劳动，最终实现部分或全部去IDE化。在具体实现上，目前形成了两种技术路径，一是将数据与储存结构合一的**表单驱动**，二是数据与逻辑完全分离、各自独立的**模型驱动**。
- ◆表单驱动以数据为核心，通过拖拽构建数据表方式展开业务分析设计。适合人事行政、资料归档、OA审批、客户管理等简单任务，对完全零基础的销售、管理人员友好，可以做到完全去IDE化，像搭积木一样按流程构建程序逻辑，**更贴近“无代码”开发**。模型驱动对软件所涉及到的功能进行建模，以应用开发平台为核心，承载各种开发工具和复杂逻辑，并将其可视化，辅以少量代码。**适合作为开发者技术中台核心**，快速按照企业需求产出一整套系统。模型驱动与表单驱动面向不同人群与不同任务，二者并无优劣之分。

亿欧智库：低代码与无代码技术路径

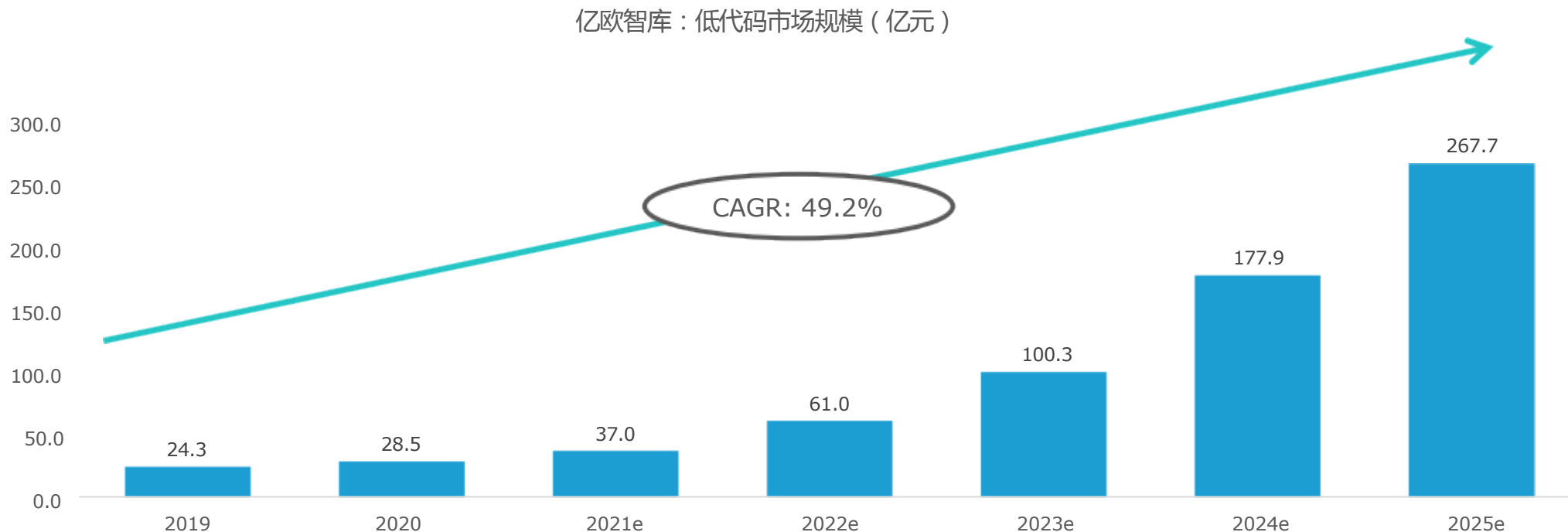




# 低代码市场发展初期行业渗透较低，但市场潜力巨大，复合增长率将高达49.2%

## 2020年，低代码市场规模是28.5亿元，将在2025年达到267.7亿元

- ◆ 亿欧智库认为目前低代码平台在发展初期无法服务嵌入性系统和信息安全产品，因此低代码市场将替代软件研发的部分市场份额。通过亿欧智库测算，2020年低代码市场规模已达到28.5亿元的水平。
- ◆ 由于低代码市场仍处于发展早期，市场渗透率不足1%，因此目前市场规模依旧较小。但低代码平台能够助力企业释放真正的能力和价值，因此未来渗透率将高速增长，通过亿欧智库测算，低代码市场复合增长率将达到49.2%，2025年市场规模将达到267.7亿元。



```
mirror_mod = modifier_ob.modifiers.new("Mirror")
# Add mirror object to mirror_ob
mirror_mod.mirror_object = mirror_ob

# MIRROR_X
mirror_mod.use_x = True
mirror_mod.use_y = False
mirror_mod.use_z = False

# MIRROR_Y
mirror_mod.use_x = False
mirror_mod.use_y = True
mirror_mod.use_z = False

# MIRROR_Z
mirror_mod.use_x = False
mirror_mod.use_y = False
mirror_mod.use_z = True

# Selection at the end - add back the text
# mirror_mod.select = True
# mirror_mod.scene.objects.active = modifier
```

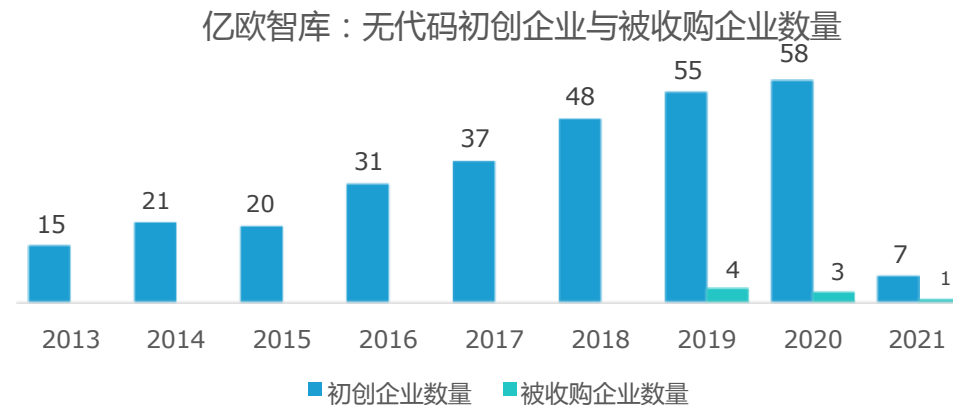
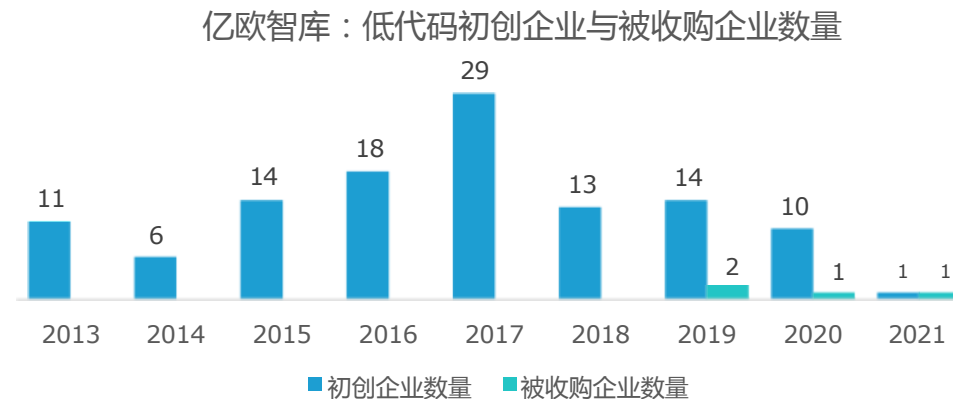
# 低代码/无代码市场现状

# 全球低代码市场发展相对较早，集中度逐渐凸显，无代码市场尚未形成市场格局

◆全球大型企业以收购低代码初创企业或自建低代码平台的方式切入市场；低代码市场在全球的多年发展使得市场已经出现Outsystems等独角兽企业；低代码初创企业在2017年达到顶峰，然后出现下降，然而无代码初创企业仍热度不减。2018年，西门子收购低代码企业Mendix将低代码市场推向热潮，因此，从2019年起，全球低代码和无代码初创企业都出现并购整合趋势。

◆因而，低代码市场集中度逐渐凸显，大型企业逐渐抢占市场份额；而无代码市场还处于早期阶段，市场格局尚未形成。

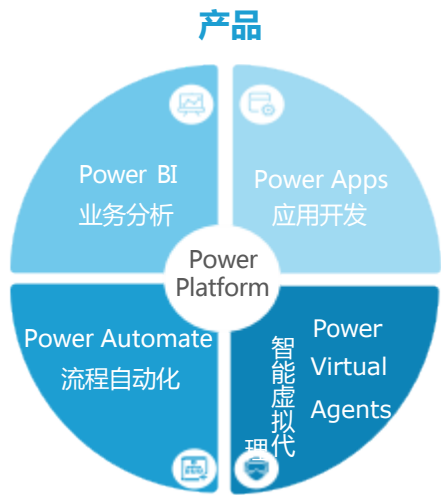
2013	<b>微软</b> <ul style="list-style-type: none"><li>微软推出第一个低代码工具Power BI，用于微软Office的Excel表格中。后续微软推出使用Power BI以及Power App等为核心组件的Power Platform无代码平台。</li></ul>
2018	<b>西门子</b> <ul style="list-style-type: none"><li>2018年8月，西门子以7亿美元的价格收购低代码企业Mendix，切入低代码市场。</li></ul> <b>Outsystems</b> <ul style="list-style-type: none"><li>Outsystems在2018年、2019年以及2021年连续获得融资，成为低代码独角兽企业。</li></ul>
2020	<b>Appain</b> <ul style="list-style-type: none"><li>Appain是成立于1999年的低代码企业，目前在低代码市场已经有一席之地。2020年，Appain收购西班牙RPA开发商Novayre以加强在低代码市场的布局。</li></ul>
2021	<b>SAP</b> <ul style="list-style-type: none"><li>2021年1月，SAP收购芬兰无代码先锋企业AppGyver，加速布局低代码市场。</li></ul>



资料来源：Crunchbase，亿欧智库绘制

# 2020年起，国外成熟低代码企业进军市场

◆ 目前进入市场的国外低代码企业仅有微软和西门子旗下Mendix，皆通过与大型软件企业或互联网企业合作的方式入局低代码市场。



## 布局市场优势

- ◆ 微软Power Platform是集成Office 365、Azure、Dynamics 365和独立引用的统一低代码平台；
- ◆ 微软拥有丰富的服务大型企业经验。

## 布局市场策略

- ◆ 微软Power Platform通过与世纪互联合作在商用落地；
- ◆ 微软与中软国际形成战略合作共同赋能低代码业务发展；
- ◆ 基于 RPA 的 UI Flow 功能将在市场正式推出。



## 布局市场策略

- ◆ 2021年4月，西门子Mendix低代码平台正式登陆腾讯云；
- ◆ Mendix将以本地化方式，价值为导向定位市场，后续会推出高竞争性的价格。

## 布局市场优势

- ◆ Mendix是全球首个推出一站式低代码平台的企业；
- ◆ Mendix是西门子数字化工业软件旗下品牌，Mendix自身服务过多家大企业，同时西门子拥有市场经验。

## 机遇

低代码企业可以通过收购国外企业提升自身技术能力。

企业可以通过与入局市场的低代码大型企业合作提升自身品牌信任度和技术能力。

国外企业的入驻起到低代码平台的市场教育作用，无形中增大了企业的市场可能性。

国外大企业的入局给予初创公司资本力量，国外大企业将会通过投资优质企业实现本地化布局。

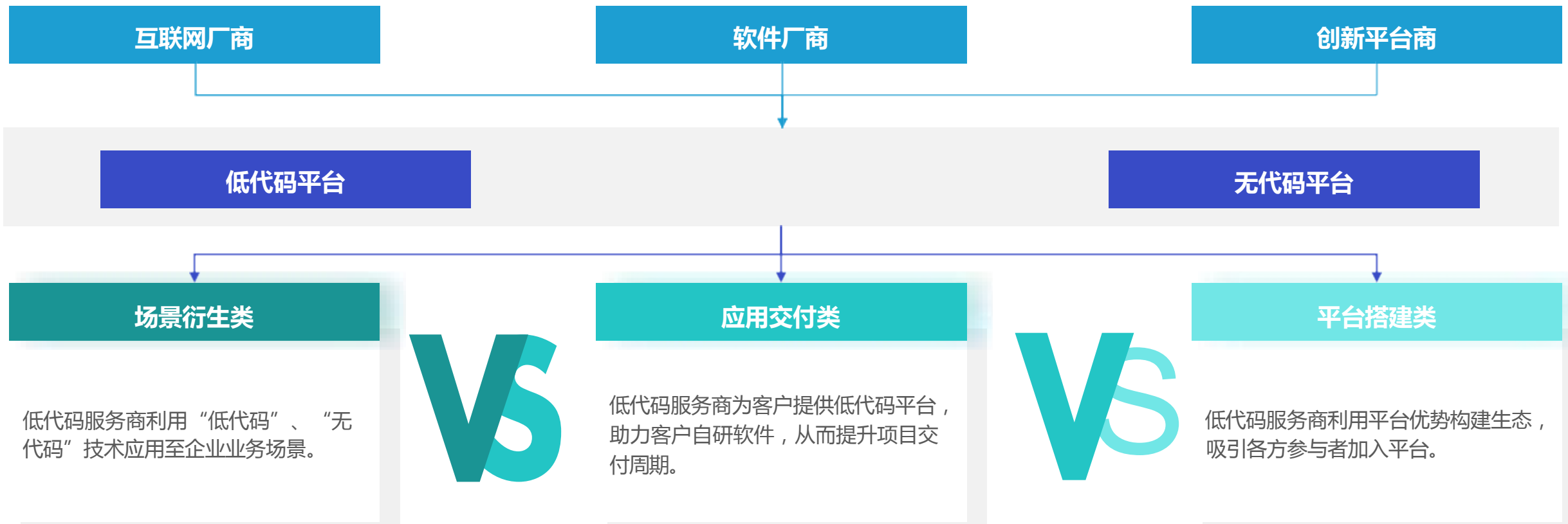
## 挑战

国外低代码大企业已经布局多年，不论在技术或服务经验方面都具有先发优势。

为布局市场，国外大企业可能会通过收购国内低代码厂商的方式增强起本地化能力，因此以更快速度抢占市场份额。

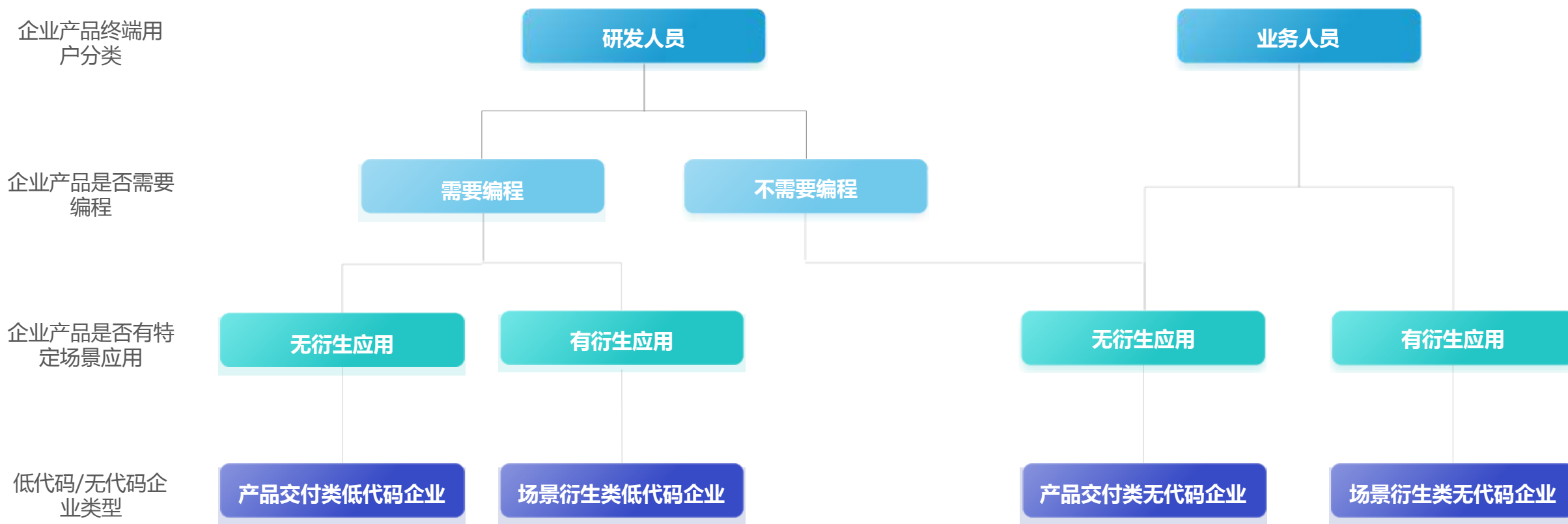
# 低代码市场仍处于蓝海市场，各大服务商入局较晚，市场集中度偏低

- ◆ 目前，共有互联网厂商、软件厂商以及创新平台商三种服务商，通过技术优势深耕复杂场景、业务优势扩展业务场景或平台优势构建生态等方式入局低代码市场。
- ◆ 由于低代码市场属于蓝海市场，行业内未出现占据较大市场份额的服务商，因此**低代码市场集中度皆较低，仍处于充分竞争的阶段。**



◆低代码企业的特点是其产品需要终端用户（研发人员）有一定的编程能力，此外，场景衍生类低代码企业的产品仅可以在某个特定场景落地应用；然而无代码企业的产品则无需编程，因此业务人员是其主要终端用户，另外，场景衍生类无代码企业产品的落地场景也较为多元化，无限制场景。

亿欧智库：低代码/无代码企业分类标准



注：产品交付类低代码企业必须落地于有限的特定业务；场景衍生类无代码企业无限制落地业务

# 市场低代码服务商图谱

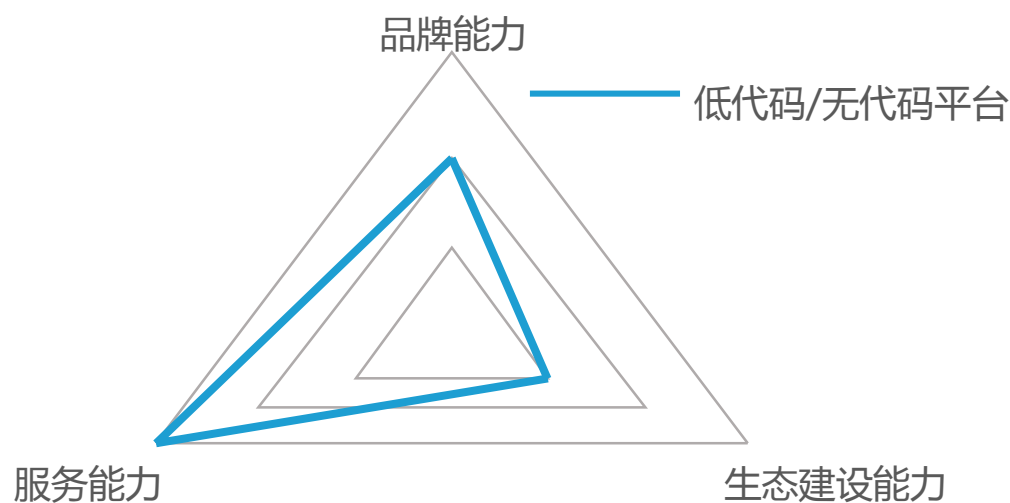


注：图谱排序无排名

# 低代码服务商各有不同的选型标准

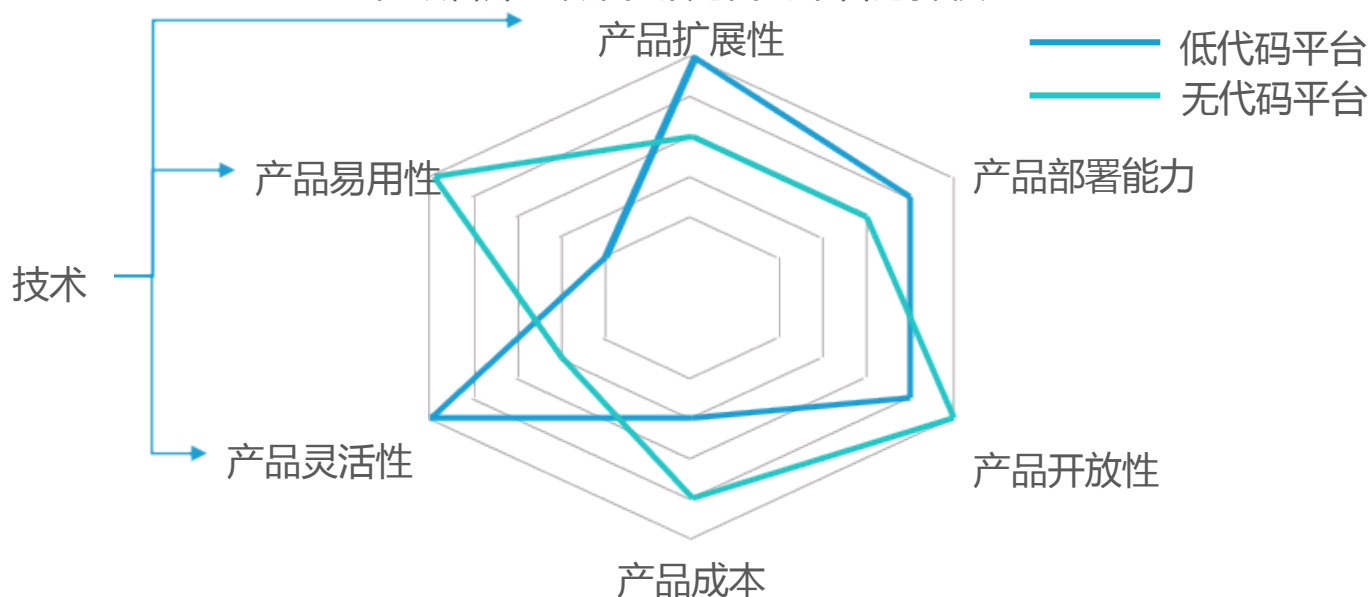
- ◆ 亿欧智库根据访谈、调研以及公开资料整理搭建企业选型指标。低代码企业选型主要分为企业自身能力以及企业产品能力两个维度。
- ◆ 从企业选型出发，低代码与无代码的选择并无差异，主要分为品牌知名度、服务能力以及生态能力三个维度；产品维度分为产品扩展性、易用性、灵活性、开放性、产品价格以及部署能力，根据无代码、低代码平台特性，选型维度有所差异。

亿欧智库：低代码/无代码企业选择模型



品牌能力	服务能力	生态建设能力
<ul style="list-style-type: none"> <li>◆ 企业规模</li> <li>◆ 企业知名度</li> </ul>	<ul style="list-style-type: none"> <li>◆ 过往服务案例</li> <li>◆ 内部人员结构</li> </ul>	<ul style="list-style-type: none"> <li>◆ 渠道拓展能力</li> <li>◆ 生态合作伙伴</li> </ul>

亿欧智库：低代码/无代码平台选择模型



产品扩展性	产品灵活性	产品易用性	产品开放性	产品部署能力	产品成本
<ul style="list-style-type: none"> <li>◆ 数据库、WebAPI集成能力</li> <li>◆ 可扩展组件能力</li> </ul>	<ul style="list-style-type: none"> <li>◆ 产品是否易于维护以及修改</li> </ul>	<ul style="list-style-type: none"> <li>◆ 产品是否易于上手使用</li> </ul>	<ul style="list-style-type: none"> <li>◆ 数据连接能力</li> <li>◆ 与其他产品连接能力</li> </ul>	<ul style="list-style-type: none"> <li>◆ 是否支持私有化部署</li> <li>◆ 部署的安全能力</li> </ul>	<ul style="list-style-type: none"> <li>◆ 产品性价比</li> </ul>



# 低代码典型企业评价：各大服务商各有优势，市场尚未出现领头羊

◆ 目前，低代码企业多为初创新平台商和软件厂商，自身企业各有优势，并未抉择出独角兽企业。例如虽然微软和mendix作为全球性企业，技术和品牌服务能力略胜一筹，但是由于在市场的本地化部署较为早期，生态建设和产品开放性还有待提升。

	企业			产品		
	品牌能力	服务能力	生态建设服务能力	技术能力 (扩展性、灵活性、易用性)	开放性	部署能力 (是否支持私有云)
致远互联	★ ★ ★	★ ★ ★	★ ★ ★ ★	★ ★ ★ ★ ★	★ ★	✓
即构	★	★ ★	★ ★	★ ★ ★	★ ★ ★	✓
葡萄城	★ ★	★ ★ ★	★ ★	★ ★ ★ ★	★ ★ ★ ★	✓
宇能云企	★	★ ★ ★	★ ★	★ ★ ★ ★	★ ★ ★	✓
金蝶	★ ★ ★ ★ ★	★ ★ ★ ★ ★	★ ★ ★ ★ ★	★ ★ ★ ★ ★	★ ★ ★ ★ ★	✓
Linkflow	★ ★ ★	★ ★ ★	★ ★ ★	★ ★ ★ ★	★ ★ ★	✓
ClickPaaS	★	★ ★ ★	★ ★	★ ★	★ ★ ★ ★	✓
炎黄盈动	★ ★	★ ★ ★	★ ★	★ ★ ★ ★	★ ★ ★	✓
微软	★ ★ ★ ★ ★	★ ★ ★ ★ ★	★ ★	★ ★ ★ ★ ★	★	✓
Mendix	★ ★ ★ ★	★ ★ ★ ★ ★	★	★ ★ ★ ★ ★	★ ★	✓

# 无代码典型企业评价：无代码服务商多为初创平台，各大服务商并未有明显优势

◆无代码市场起步晚于低代码市场，不论头部互联网企业或是初创企业皆在商业模式的摸索过程中，属于初创平台。虽然各企业竞争优势尚未凸显，但轻流等无代码服务商已获得市场认可，发展潜力较大。

	企业			产品			
	品牌能力	服务能力	生态建设服务能力	技术能力 (扩展性、灵活性、易用性)	开放性	部署能力 (是否支持私有云)	成本
数睿数据	★★	★★★★	★★	★★★★	★★★	✓	★★
轻流	★★★	★★★★	★★★	★★★★	★★★★	✓	★★★★
ivx	★★	★★	★★★★	★★	★★★★★	✓	★★★★★
明道云	★★★★	★★★★★	★★★★	★★	★★★★	✓	★★★★★
简道云	★★★	★★★★	★★★	★	★★★	✗	★★
宜搭	★★★★★	★★★★★	★★★★★	★★★	★★	✓	★
奥哲·氚云	★★★	★★★★	★★★	★★★★★	★★	✗	★★★★

```
mirror_mod = modifier_ob.modifiers.new("Mirror")
# Add mirror object to mirror_ob
mirror_mod.mirror_object = mirror_ob

# MIRROR_X
mirror_mod.use_x = True
mirror_mod.use_y = False
mirror_mod.use_z = False

# MIRROR_Y
mirror_mod.use_x = False
mirror_mod.use_y = True
mirror_mod.use_z = False

# MIRROR_Z
mirror_mod.use_x = False
mirror_mod.use_y = False
mirror_mod.use_z = True

# Selection at the end - add back the mesh
obj.select()
obj.scene.objects.active = modifier
```

# 低代码/无代码平台的实践应用及应用价值

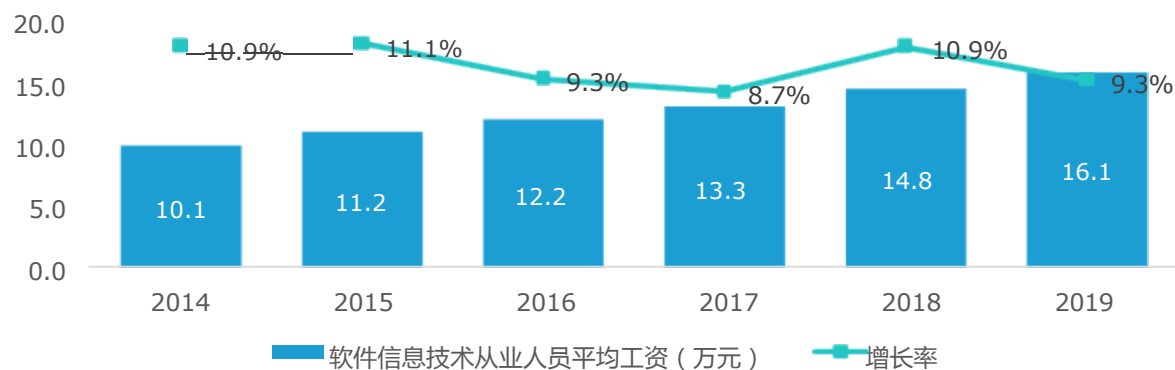
# 传统软件开发面临成本高、效率低等痛点

## 开发成本高昂：

随着对IT人才需求的增长，IT人才面临供需不平衡的难题，因此IT从业人员平均工资逐年上涨并成为平均工资最高的行业。

从占比分析，虽然软件业务收入呈增长趋势，但增长率低于IT就业人员工资增长比例，因此IT人力成本占比呈逐年上升趋势。

亿欧智库：IT就业人员平均工资及增速

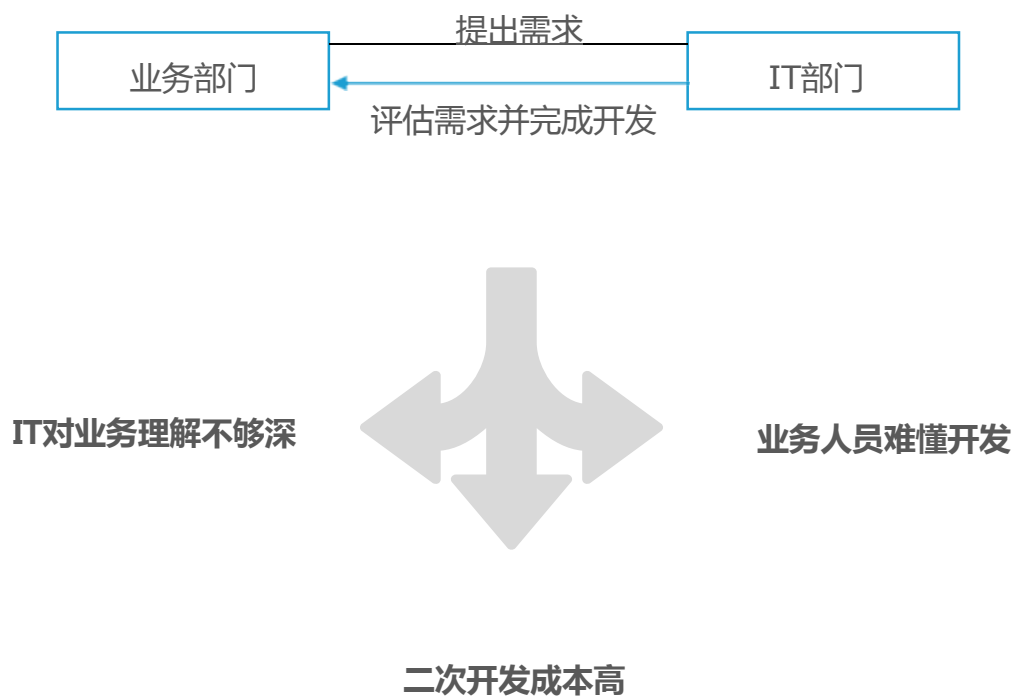


亿欧智库：IT就业人员平均工资总额、软件业务收入以及人工成本占比



## 业务人员与开发人员沟通存在鸿沟：

目前，由于需求和执行之间的差别，传统软件的开发人员和业务人员面临难以逾越的“沟通鸿沟”。



# 低代码开发解决传统开发痛点，重塑传统开发生产关系

◆低代码平台可以有效解决传统软件开发需求变动造成的业务与开发矛盾以及交付时间冗长等难题。

## 低代码开发

- 业务人员根据需求做简单分类，例如进销存等**轻需求可以使用无代码平台做开发，部分复杂场景需求可以使用低代码平台做开发**，从而解决大量需求堆积的难题。
- 业务人员可以通过无代码平台实现边缘化或简易需求的平台搭建，从而，**业务人员可以及时响应客户需求，自己改变搭建形成新应用**。
- 由于重复代码基本模块已经被封装。开发人员只需要完成少量代码的开发便可以搭建一个软件，这种开发模式大大**减少了开发员的开发成本、需求变化导致的二次开发成本以及维护成本**。
- 由于在低代码平台的“拖拉拽”的功能赋能下，企业搭建应用较为容易，这很大程度**缩减了项目交付时间，快则一天便可以完成部分应用的搭建**。

目前，低代码平台可以减少传统软件开发项目代码量

50%

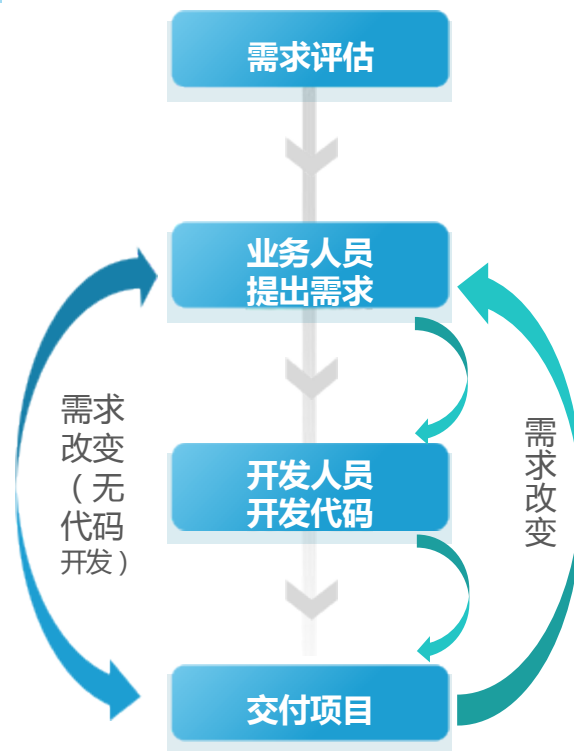


## 传统软件开发

- 对于企业来说，需求多而复杂，由于需求不能高效的被解决，导致**需求堆积严重**；其次**边缘性需求容易被搁置**，长时间不能得到有效解决方案。
- 在传统开发模式中，业务人员需将所有需求对接给开发人员，一旦需求进行更改，业务人员需要再次提交需求，**周而复始的需求对接将造成业务人员和开发人员互不解的矛盾**。
- 传统开发模式中，开发人员不仅需要重复性输入代码，软件的维护与二次开发成本也需要较高的人力成本；其次需求的改变常常意味着**大量时间、人力成本的注入**。
- 传统开发项目交付时间通常较长，**少则几个月，多则需要半年以上的时间才能完成交付**。

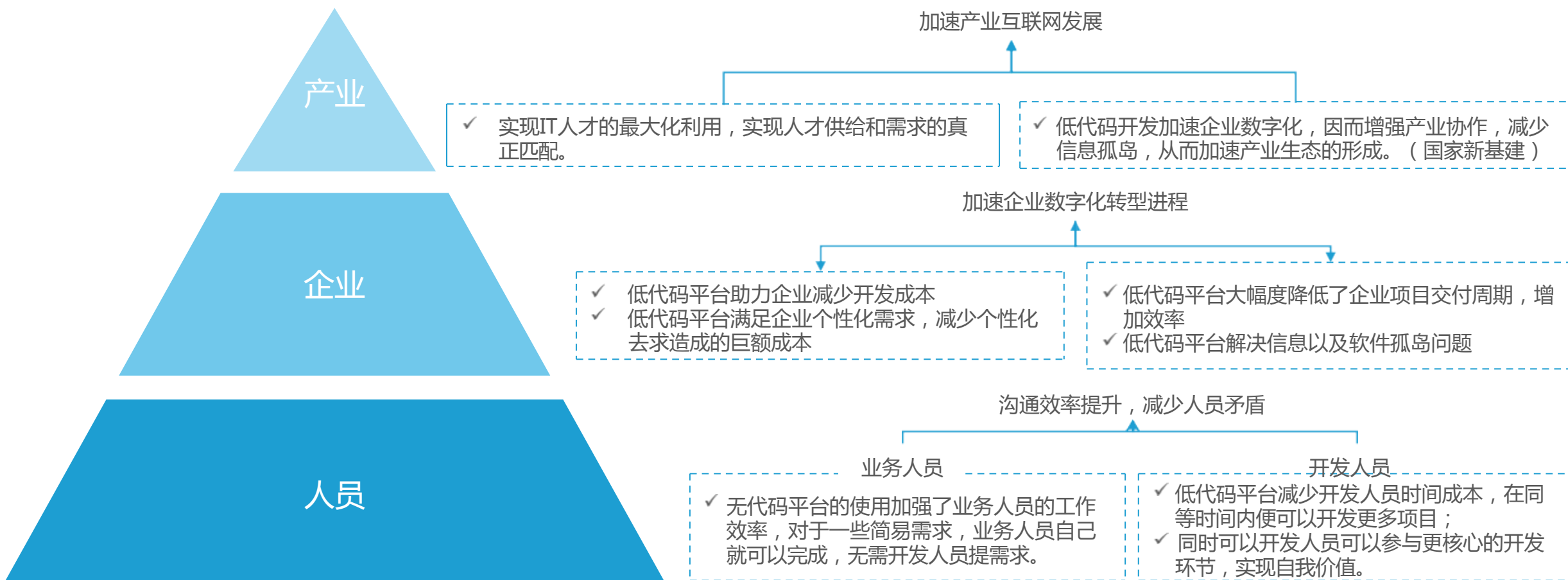
未来，低代码平台可以减少传统软件开发项目代码量

70-80%



## 低代码平台助力企业人员实现自我价值，加速数字化转型，最终实现产业互联网的增速发展

◆低代码平台的应用实现开发人员和业务人员的自我价值，从而减少沟通矛盾；从企业角度分析，低代码平台的应用助力企业降本增效，最终实现数字化转型；IT人才的最大化利用将改变现有产业工作模式，最终将助力产业形成新形态，加速产业互联网发展。



以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：  
<https://d.book118.com/637161061053010003>