

升级到2026

部署升级包

升级说明

升级包部署

将升级包解压缩，将升级包中包含的plm目录拷贝到2026的服务端安装目录下，覆盖2026安装目录下的plm目录。

备注:升级包中仅包含了升级相关的文件，不影响2026其它文件和功能。

升级前准备

支持的版本

目前仅支持2024版本升级到2026，仅支持目前2026包括的模块数据的升级。

低于2024版本的要升级到2026，需要先升级到2024版本。

缺省只支持没有定制的标准版升级。

数据库

1.仅支持相同类型的数据库升级。2026对数据库有版本要求，如果当前使用的数据库版本不符合要求，需要部署符合2026要求的数据库版本。

2.升级时，需要2024版本的plm数据库和2026版本的数据库在一个实例上。如果为2026安装了新的数据库，需要将2024版本的数据库备份并恢复到新安装的数据库上。

文仓

1.主服务对应一个2026的文仓服务。安装时2026的文仓服务的实例名称要和2024主服务的名称一致(2024缺省是mainserver)。

2.如果2024部署了多文仓，2026也需要部署多文仓，2024一个文仓服务对应一个2026的文仓服务，安装时2026的文仓服务的实例名称要和2024文仓服务的名称一致。

3.安装2026文仓服务时，需要指定文仓的存储路径。需要复制一份对应的2024的文仓服务的文仓，将2026的文仓服务路径指向复制的24文仓路径。(这部分相当于升级文仓文件)

例如，2024的文仓路径是c:\vault,复制一份到d:\vault2026,安装2026时，需要建立一个虚拟盘，虚拟盘路径指向d:\vault2026

4.当2024的一个文仓下面有多个路径时(此处指文仓服务配置界面有多个文仓路径)，每个路径对应一个2026的文仓下面的一个虚拟盘。

升级前，也可以根据需要将多个路径，来减少虚拟盘数量。

文档出库状态不升级

文档出库状态不升级，出库的文档升级后，均为非出库状态。

请在升级前，处理好出库文档，将出库文档入库或者取消出库。

文档图片签名

2026支持的签名图片格式和2024不同，2026格式主要为png图片，并且有大小限制，存储在数据库中；2024的签名图片格式支持bmp等格式，签名图片存储在文仓中。2024的签名图片无法升级到2026。升级前需要准备好人员对应的png签名图片。升级后，配置人员的签名图片，并勾选“用此次签名信息修改同步更新库中现有签名信息”。

工艺BOM

2026版本将工艺零件对象合并到零件对象中，升级过程会通过关键属性映射机制将工艺零件数据迁移至零件表。升级前需根据2024版本工艺零件的匹配规则手动调整配置文件：

```
install\upgrade\supported_version\2024\config_process_part.json
```

```
{
  "source_types": {
    "000000000001000000001020": { // 2024版本生产类型索引
      "internal_name": "manufacture", // 2026版本对应生产类型内部名称
      "fileds": [ // 关键属性映射列表
        {
          "from": "IdentityNo", // 2024版本工艺零件对象字段名
          "to": "id_code" // 2026版本零件对象字段名
        }
      ]
    },
    "000000000002000000001020": {
      "internal_name": "purchased",
      "fileds": [
        {
          "from": "IdentityNo",
          "to": "id_code"
        }
      ]
    },
    "000000000003000000001020": {
      "internal_name": "cooperation",
      "fileds": [
        {
          "from": "IdentityNo",
          "to": "id_code"
        }
      ]
    },
    "000000000004000000001020": {
      "internal_name": "standard",
      "fileds": [
        {
          "from": "IdentityNo",
          "to": "id_code"
        },
        {
          "from": "name",
          "to": "name"
        }
      ]
    }
  ]
}
```

工艺BOM的升级流程将基于上述配置执行：系统读取2024版本中的工艺零件数据，通过关键属性匹配零件表中是否存在相同关键字的对象。若存在匹配项，则工艺BOM中对应条目将替换为匹配到的零件（匹配到多个零件对象会使用第一个）；若不存在，则自动在零件表中创建新的零件对象。

《编码升级工具》

一、编码升级前的注意事项：

1.补充升级配置文件内容（如果没有新增编码类型，请忽略）

前提：如果除了物料编码，产品编码，文件编码，图号编码，通知单编码之外，还有其它编码类型需要升级，则需要修改升级配置文件，以24的物料编码为例，方法如下：

1. 添加升级策略配置

(1) 路径：

install\upgrade\supported_version\2024\codemanagement\upgrade_strategy.json

(2) 添加25类名到文件名的映射

```
{
  "class_name": "item_code",
  "file_name": [
    {
      "old_file": "物料编码",
      "new_file": "item_code"
    }
  ]
}
```

(3) 说明：

1> class_name: 25编码内部名

2> file_name: xml文件名

3> old_file: 24编码的xml文件原始名

4> new_file: 升级时转换后的24编码的xml文件名，同时也是升级属性映射的配置名（可自取，但需为英文名）

(4) 使用：若还有一个新的24编码需升级到25的物料编码，则只需在对应的file_name里添加一个json对象：

```
{
  "old_file": "新编码名称",
  "new_file": "xx_code_to_item_code"
}
```

2. 添加属性映射配置

(1) 路径：

E:\server\install\upgrade\supported_version\2024\codemanagement\upgrade_strategy\item_code.json

(2) 配置说明：

1> item_code.json: 属性映射配置名称，需要与 升级策略配置（upgrade_strategy.json）中对应的 new_file 名 保持一致！

2> from_class: 24类名，通常也是24表名

3> to_class: 25类名

4> class_id: 24classId

5> properties: 属性映射列表

6> from_property: 24xml文件中的属性字段

7> to_property: 25编码类中的属性内部名（如果24的某个属性不需要被升级，则可以不配置 to_property）

8> conversion_method: 属性转换方法

9> lookup_map: 如果转换方法为lookup，则需要配置lookup_map

二、编码升级映射详情

1. 物料编码

- 24classId: 0
- 24编码表: code2_code
- 客户端XML文件 (原): 物料编码.xml
- 客户端XML文件 (转换后): item_code.xml
- 升级属性映射配置: item_code.json
- 中间表: u_item_code
- 升级到25的编码类型为: 物料编码

2. 产品编码

- 24classId: 1
- 24编码表: code2_product_code
- 客户端XML文件 (原): 产品编码.xml
- 客户端XML文件 (转换后): product_code.xml
- 升级属性映射配置: product_code.json
- 中间表: u_product_code
- 升级到25的编码类型为: 产品编码

3. 文件编码

- 24classId: 2
- 24编码表: code2_code2
- 客户端XML文件 (原): 文件编号.xml
- 客户端XML文件 (转换后): file_code_to_generic_code.xml
- 升级属性映射配置: file_code_to_generic_code.json
- 中间表: u_file_code_to_generic_code
- 升级到25的编码类型为: 通用编码

4. 图号编码

- 24classId: 3
- 24编码表: code2_code2
- 客户端XML文件 (原): 图号.xml
- 客户端XML文件 (转换后): drawing_code.xml
- 升级属性映射配置: drawing_code.json
- 升级中间表: u_drawing_code
- 升级到25的编码类型为: 图号编码

5. 通知单编码

- 24classId: 4
- 24编码表: code2_code2
- 客户端XML文件 (原): 通知单号.xml
- 客户端XML文件 (转换后): notice_number_to_generic_code.xml
- 升级属性映射配置: notice_number_to_generic_code.json
- 中间表: u_notice_number_to_generic_code
- 升级到25的编码类型为: 通用编码

注意

- 如果想要升级编码状态，需要在对应的编码的客户端原XML文件中添加编码状态db_status这一属性

三、编码规则升级说明

升级后的编码规则名称由24classId+upgrade_code_rule组成，例如：24文件编码升级到25通用编码的规则名称为2_upgrade_code_rule。

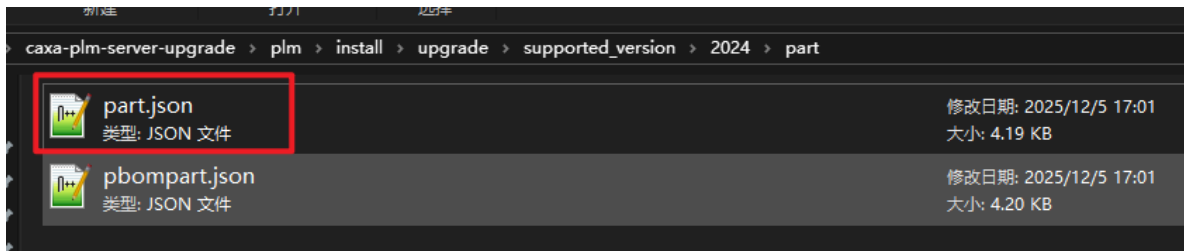
2026环境准备

- 1.升级前，需要部署2026，文仓服务需要按照上面文仓部分进行部署。
- 2.升级前请确保2026是空库，没有业务数据，否则会影响升级。

元模型定制

- 1.如果元模型有定制，需要先登录2026，定制2026的元模型并映射更新数据库。
- 2.如果有相关界面的修改，还需要定制相应的界面配置。
- 3.需要手动修改升级配置映射文件，来处理定制的属性升级。

以零部件为例，在如下配置文件中设置升级前后的字段之间的映射



2024配置文件准备

客户端配置文件

将C:\Users\Public\CAXA\CAXA EAP CLIENT\1.0\Cfg目录压缩为zip文件。

服务端配置文件

将服务端部署路径下的Server\Cfg目录压缩为zip文件。

workflow模板

查看 workflow服务设置界面，找到模板库配置路径
将模板库路径下的workflow目录压缩为zip文件。

升级失败的处理

升级过程中，如果数据升级失败，需要人为干预后再继续升级。
再次升级时，已升级的业务数据可能会影响重新升级的处理。尤其是 workflow模板升级错误。

没有元模型定制的升级失败后处理

如果没有元模型定制，升级失败，处理造成升级失败的原因后，可以删除2026的业务数据库，重新安装2026，再次升级。

有元模型定制的升级失败后处理

如果有元模型定制，升级前，需要定制元模型并映射数据库。

1.映射数据库后，在升级前，备份plm业务数据库，如果升级失败，处理造成升级失败的原因后，使用备份文件恢复plm业务数据库，再次升级。

特殊情况

测试过程中，部分机器因为上传配置文件后，防护软件扫描锁定解压后文件,造成升级程序无法访问解压后文件，造成升级出错。

暂停防护软件或者等待几分钟后(防护软件扫描完毕)，再次升级正常。

升级

windows下升级程序位置

[2026安装目录]\plm\bin\caxa_upgrade.exe

linux下升级程序位置

/opt/CAXA/plm/bin/caxa_upgrade

升级到2026

- 1.运行升级程序，默认会打开浏览器，显示升级界面
- 2.输入2024数据库信息
- 3.上传配置
工作流模板上传后，需要指定模板对应的分类。对不需要分类的模板指定 常规->其他 分类。
- 4.勾选要升级的数据项
- 5.执行升级

不可升级数据

- 1.编码相关流程模板和流程不支持升级

升级后注意事项

人员密码

2026默认开启密码安全，对密码有强度要求。密码需要包含大写字母，小写字母，数字，特殊字符 (!, @, #, %, *, &)四类，长度8-16位。

可以使用2026缺省配置来设置用户密码或者在配置系统中修改密码安全配置。

配置入口在配置系统中 系统配置管理-》安全设置-》密码安全

人员可用应用

当用户的许可中包含超过一种应用时，升级后，需要在配置系统中设置人员的可用应用。如果不设置，登录业务系统会报错。

文档图片签名

2026支持的图片签名格式主要为png图片，且有大小限制。2024如果使用图片签名，升级后，需要配置人员的签名图片，并勾选 "用此次签名信息修改同步更新库中现有签名信息"。

签名配置

升级后的签名配置，需要在配置系统中《文档管理-》签名-》基础设置-》签名关联的流程模板文件中选择对应的流程模板后显示。

权限

2026的权限和2024不是一一对应的。升级后，请检查人员角色的授权，确保权限符合所需。
2026基于工作组管理对象权限，与2024有较大差异，需要熟悉2026的对象权限管理。
2024的对象参与者权限不升级，如果有对象权限管理需求，需要按2026的权限管理进行设置。

配置

2026和配置和2024的配置格式也不尽相同，升级后，请检查关键配置，比如匹配规则等，以免影响后续新数据入库。

工艺配置

工艺配置升级时仅处理工艺类型和工艺模板映射配置的升级。工艺显示配置会基于升级后的规程创建一个默认的配置，需基于默认配置重新配置；工艺推荐、拆分、校验、合并配置需升级后需重新配置。

模板文件

2024中的批量导入模板等模板文件如果要继续在2026中使用，需要在2026配置系统的模板文件管理模块中将模板文件导入到对应的分类下。

《项目管理升级工具》

一 可升级数据

升级内容包括项目大类、项目模板大类，以及项目任务、项目模板的以下关联数据：

- 项目：基础属性、参考资料、WBS、团队角色、负责人、任务类别
- 项目任务：基础属性、报工、变更记录、交付定义、文档类交付物、负责人、依赖关系
- 项目模板：参考资料、WBS、团队角色、任务类别

二 不可升级数据

以下数据不支持升级，需要在新系统里设置或补录：

- 流程设置
- 企业日历
- 预警定义
- 虚拟交付目录
- 团队角色的权限

三 前置要求

1. 要求历史系统里同时存在图纸、文档和3D文档；缺少任何一种文档类型会影响参考资料关系数据无法正常升级；
2. 要求项目表是空表；

四 可能出现的问题

1. 升级完成后，人员权限管理里项目角色中新增两个组角色。对新系统无影响，保留即可；

《变更管理升级工具》

一、变更升级前的注意事项

1. 补充变更属性映射配置

1. 添加表单模板属性映射配置

(1) 路径: `install\upgrade\supported_version\2024\change\propertyMap.json`

配置示例:

```
{
  "ecn": {
    "test1": {
      "type": "string",
      "new_property": "test1"
    },
    "test2": {
      "type": "bool",
      "new_property": "test2"
    },
    "test3": {
      "type": "int",
      "new_property": "test3"
    }
  }
}
```

(2) 说明

1> **"ecn"**: 原变更表单模板的显示名, 如果如示例为英文, 需注意大小写。同时需要在新系统中在 **ECN** 或 **ECR** 类中增加分类, 并将分类名称同样设置为该显示名, 分类下再新建表单模板, 并将模板内部名也设置为该显示名。

2> **"test1"**: 原变更表单模板中设置的自定义属性内部名, **test1** 为示例, 不是实际字段名, 按需要自定义。

3> **"type"**: 原变更表单模板中设置的自定义属性的类型, 目前仅支持 **string**, **int** 与 **bool** 类型。

4> **"new_property"**: 映射过去的新的属性内部名, 可以选择与原模板中的属性内部名一致, 如示例所示, 也可以重新起名。新设置的属性需要在对应上述的分类下设置。

5> 注: 在设置新的变更模板时, 签名控件的标识应与升级前原变更模板中签名属性内部名一致, 如果签名属性内部名为中文则不支持升级该签名。

(3) 配置范例:

```
{
  "工程设计变更单": {
    "inventory_number": {
      "type": "int",
      "new_property": "inventory_number"
    },
    "change_description": {
      "type": "string",
      "new_property": "change_description"
    },
    "product_name": {
      "type": "string",
      "new_property": "product_info"
    }
  }
}
```

如上范例:

"工程设计变更单": 原表单模板显示名

"inventory_number": 原表单模板中配置的自定义属性内部名, 类型为 **int**, 并且能够映射到手动创建的内部名为 **"工程设计变更单"** 的分类下的 **"inventory_number"** 属性, 该属性可在内部名为 **"工程设计变更单"** 的表单模板视图内设置关联的控件。

二、可升级数据

- 1.ECN与ECR单据对象：基础属性（不包含 workflow 模板id与 workflow 模板实例id），表单签名。
- 2.ECN与ECR的相关关系：ECN/ECR与可变更类的关系，ECN与ECR的关系，表单与所属者的关系，表单与参考文档的关系。

三、不可升级数据

- 1.变更相关流程均无法升级。
- 2.原变更单模板中自定义的属性超出string,int或者bool类型外的其他类型数据无法升级。
- 3.原变更单模板无法升级，即原先设置的模板权限，流程，输出等设置均要在手动新建的新表单模板中重新设置。
- 4.变更工作组不支持升级。
- 5.pdf红批数据不支持升级，2026pdf红批文件是一个新格式文件，用于在web端浏览，与2024版本的文件格式不同，无法升级。
- 6.影响对象页签数据不支持升级。
- 7.变更对象页签中工艺相关的物料或文档对象不支持升级。

四、前置要求

- 1.尽量在旧系统中将正在执行的变更全部完成后再进行升级。如果存在正在运行的变更，升级过来后无法继续进行流程，只能重新发起。