# Predix机器SDK概述

Predix机器SDK包括两个SDK：一个带有插件的Eclipse SDK，用于想要使用Predix机器开发应用程序的开发人员，以及一个用多种语言编写应用程序的边缘SDK，用于通过数据总线与Predix机器进行通信。

## Eclipse SDK

Predix机器软件开发套件（SDK）用于想要使用Predix机器开发应用程序的开发人员。Predix机器SDK允许您通过选择包括特性(feature)所需的所有捆绑包(bundle)的特定特性组来生成您自己的Predix机器运行时容器。您也可以通过选择个别捆绑包生成容器。特性的分组是根据特定的相关性，以及相关特性。

所有特性均包括以下功能：

* Predix机器运行时容器
* Felix相关性管理器
* 声明式服务支持
* Felix HTTP捆绑包
* Jersey（jaxrs）捆绑包
* 元类型-配置管理
* 日志记录
* 安全管理员服务：SSL认证管理
* 用户管理服务（账号管理）

下表显示了每个Predix机器特性组中包括的Predix机器捆绑包。

|  |  |  |
| --- | --- | --- |
| **特性** | **描述** | **相关特性** |
| Predix机器代理（容器化） | * 允许Docker容器互相通信。

注：通过边缘SDK进行开发时需要这一特性。这是运行在边缘上的其中一个Docker容器。 | * Predix机器数据总线
* Predix网页工具
* 命令框架
* Predix云网关
* Predix机器配置支持
 |
| Predix机器应用程序服务 | * Git存储库管理服务
 | * Predix网页工具
 |
| Predix机器云网关 | * Predix云身份管理
* 向云端更新设备细节
 | * Predix网页工具
* Predix机器设备细节
* 命令框架
* 存储与转发
 |
| Predix机器命令框架 | * 向运行在一台设备上的应用程序发送命令。
 |   |
| Predix机器连接管理 | * 打开VPN支持
 | * Predix机器配置支持
* Predix网页工具
 |
| Predix机器设备信息 | * 提供设备的动态和静态信息
 |  |
| Predix机器设备管理  | * 允许设备从云上接收命令和更新。
 | * Predix网页工具
* Predix云网关
* 命令框架
* Predix机器配置支持
 |
| Predix机器机器网关 | * OPC-UA适配器
* Modbus适配器
* 健康状况监视器适配器
* Hoover
 | * Predix网页工具
* Predix云网关
* 存储与转发
* HTTP River
 |
| Predix机器HTTP River  | * 从启用Predix机器的边缘设备向使用HTTPS的Predix云传输数据。
 | * Predix网页工具
* Predix云网关
 |
| Predix机器HTTP隧道 | * 通过HTTP/HTTPS简化不同网络协议的通信
 | * Predix网页工具
* Predix云网关
 |
| Predix机器数据总线 | * 在容器之间进行数据和控制管道(control plane)信息通信。
 |   |
| Predix机器MQTT支持 | * 向代理发布消息或订阅一个主题来接收消息。
	+ MQTT River
	+ MQTT客户端
 | * Predix网页工具
* Predix云网关
 |
| Predix机器OPC-UA服务器 | * 允许启用Predix机器的应用程序通过OPC-UA协议公开数据，该协议是一个机器到机器的协议。
 | * Predix网页工具
 |
| Predix机器配置支持  | * 远程设备管理
* 使用脚本（PROV）生成容器时提供配置支持
* 处理ZIP支持的捆绑包

注：更多信息请参见 EdgeManager 。  | * Predix网页工具
* Predix云网关
* 命令框架
 |
| Predix机器存储与转发 | * 将数据转发到云上并连续存储数据，以防数据丢失。
 |   |
| Predix机器技术人员控制台 | * Predix技术人员控制台
 | * Predix网页工具
* 命令框架
* Predix云网关
* 配置支持
 |
| Predix机器网页控制台（只能用于调试） | * Predix机器网页控制台和捆绑包更新

注：只能用于调试 | * Predix网页工具
 |
| Predix机器网页工具 | * 通过JAX-RS提供JSON支持
* 用于REST支持的HTTP服务
 |   |
| Predix机器网络套接字River | * 网络套接字River
 | * Predix网页工具
* Predix云网关
* Predix网络套接字
 |
| Predix机器网络套接字 | * 网络套接字服务器
* 网络套接字客户端服务
 |   |

## Predix机器边缘SDK

通过边缘SDK，您可以采用多种语言编写应用程序，以便通过数据总线与Predix机器通信。

关于使用边缘SDK的说明，请参见READMe.txt文件。

# 要求

## Java开发套件

您必须使用Java开发套件（JDK）而不是Java运行时环境（JRE）。

## 环境变量

必须使用Predix机器脚本设置以下环境变量：

JAVA\_HOME：与Predix机器一起使用的Java虚拟机的位置。

注：请勿使用反斜杠结尾或为 JAVA\_HOME 环境变量添加引用。

## 软件包

使用Predix机器SDK需要以下软件包。

* 带有插件开发环境（PDE）的Eclipse（Mars或Neon 64位版），例如用于Java EE开发人员的Eclipse IDE。

注：下载Eclipse [Mars](http://www.eclipse.org/downloads/packages/release/Mars/R)或[Neon](http://www.eclipse.org/neon/)。

如果您的开发环境正在使用不同的版本，您可以安装并使用适合Predix机器SDK的其他Eclipse版本。

* Predix机器SDK。参见[下载Predix机器SDK](https://www.predix.io/docs/?r=62889#dIlIc5DH)。
* Maven 3.1或更高版本。如果要验证您的Maven安装和版本，可以在命令行中输入： mvn -version。您可以从[此处](https://maven.apache.org/download.cgi)下载Maven。

## 需要的Linux实用程序

由于某些设备可能运行多种操作系统的最低或有限版本，以多种脚本启动和运行Predix机器容器的某些可执行实用程序可能未安装。在Linux环境中运行Predix机器需要以下可执行实用程序。

| **实用程序** | **描述** | **实用程序** | **描述** |
| --- | --- | --- | --- |
| **awk** | 对数据执行模式匹配操作 | **kill** | 将一个信号发送到一个进程 |
| **cat** | 链接 | **nohup** | 如果您退出，防止命令中止 |
| **chmod** | 更改文件或目录的许可 | **printf** | 打印一个格式化字符串 |
| **command** | 验证一个命令是否存在 | **ps** | 显示进程状态 |
| **date** | 打印或更改时间和日期 | **readlink** | 打印符号链接或文件名的值 |
| **dirname** | 去掉文件名的最后部分 | **sed** | 筛选并转换文本 |
| **echo** | 重复输入的发送至外围设备的文本 | **sleep** | 延迟或暂停 |
| **find** | 在一个文件中搜索文本 | **sudo** | 允许您以另一名用户的身份执行命令 |
| **getopts** | 解析命令行自变量 | **systemctl** | 控制systemd系统和服务管理器的systemd实用程序。 |
| **grep** | 一行一行地处理文本并打印与特定模式匹配的行 | **tr** | 停止处理器等待进一步的指令 |
| **head** | 输出文件的第一部分 | **trap** | 翻译字符集 |
| **jar** | 操作Java归档（JAR）文件 | **uname** | 打印系统信息 |
| **java** | 启动一个Java应用程序 | **unzip** | 用于提取压缩文件 |
| **keytool** | 创建私有密钥 |   |   |

## 内存要求

Predix机器不为Java虚拟机（JVM）提供与内存管理相关的指令，允许JVM中的Ergonomics特性进行智能选择，因此可以动态进行调节。JVM根据Predix机器所在服务器的类别进行选择，而服务器类别由总可用内存、CPU的数量和平台架构（32位或64位）决定。

如果没有指定内存分配的明确命令行参数，JVM在起动时确定最小和最大堆内存尺寸，并确保在此限值范围内使用，根据需要增加和收缩确认的堆内存分配。例如，对于具有两个或更多CPU以及2 GB或更大RAM的32位系统，Java 7和8将最小堆内存尺寸设置为可用物理内存的1/64，最大堆内存尺寸设置为可用物理内存的1/4至1 GB。在具有128 GB或更大RAM的64位系统上，默认的最大堆内存尺寸可达到32 GB。

因此，在具有64 GB RAM和16个CPU的64位Linux服务器上，Predix机器（或未明确指定堆内存参数的任意Java进程）将被赋予最小1GB最大16 GB的堆内存尺寸。另一方面，运行在具有434 MB RAM的Raspberry PI等更小设备上的Predix机器将被赋予最小7 MB最大110 MB的堆内存尺寸。经证明Predix机器在多个系统上使用默认设置运行良好，包括Raspberry PI。

除非情况另外需求，最佳做法是由JVM进行堆内存配置和调节。但是，运行在Predix机器下的应用程序捆绑包的工作特性可能需要比默认分配更多的堆内存空间。堆内存空间还取决于您在Predix机器容器中选择的特性。另外，由于系统上运行的其他应用程序，可能需要减少堆内存的使用。

## Predix机器特性的内存占用

下表显示了每个Predix机器特性的内存占用。

| **特性** | **容器运行时使用的内存****（MB）** |
| --- | --- |
| Predix其他服务 | 49 |
| Predix应用程序服务 | 72.8 |
| Predix云网关 | 57 |
| Predix命令框架 | 17.2 |
| Predix连接管理 | 41 |
| Predix设备信息 | 67 |
| Predix设备管理  | 61.8 |
| Predix HTTP River | 54 |
| Predix HTTP隧道 | 59 |
| Predix机器网关 | 36.5 |
| Predix MQTT支持 | 34.5 |
| Predix OPC-UA服务器 | 188 |
| Predix配置  | 70.3 |
| Predix存储与转发 | 106.2 |
| Predix技术人员控制台 | 73.5 |
| Predix网页控制台 | 72.5 |
| Predix网页工具 | 58.1 |
| Predix网络套接字River | 63.4 |
| Predix网络套接字 | 59.2 |

## Docker容器的内存占用

下表显示了Ubuntu 14.04上每个Predix机器Docker容器的内存占用。

| **容器** | **容器运行时使用的内存** | **镜像尺寸** |
| --- | --- | --- |
| Predix机器监督程序 | ~300MB | 201.50MB |
| 启动引导加载程序 | ~1MB | 13.63MB |
| Predix机器 | ~200MB | 186.90MB |
| Mosquitto | ~20MB | 15.42MB |
| C++ SDK样例订阅服务器 | ~1MB | 10.12MB |
| C++ SDK样例订阅服务器 | ~1MB | 10.12MB |

# 验证的平台

Predix机器经验证运行在以下平台上。

|  |  |  |
| --- | --- | --- |
| **系统/品牌** | **操作系统** | **规格示例** |
| 戴尔笔记本 | Windows 7 | * Intel ® Core TM i7-2720QM CPU @ 2.20GHz
* 8核
* 16GB RAM
 |
| 苹果MacBook | MacOS X 10.9.4 | * Intel Core i7 2.3GHz
* 16GB RAM
 |
| Linux Server | GNU/Linux 3.8.13-35.1.2.el6uek.x86\_64 | * Intel Xeon® CPU X5650 @ 2.67 GHz
* 12GB RAM
* 6核
 |
| Linux VM | GNU/Linux 2.6.32-400.29.2.el6uek.x86\_64 | * Intel Xeon CPU X7550 @ 2.00GHz
* 8GB RAM
 |
| Intel Galileo |   | * 400MHz 32-bit Intel Pentium ISA兼容
* 512MB片上SRAM
 |
| Raspberry Pi | Linux ARM6, 3.10.25 Raspbian | * ARMv6兼容处理器版本7（v6I）
* 448180KB RAM
 |
| Intel Kontron M2M | GNU/Linux 2.6.34.10-WR4.3.0.0\_standard | * Intel Atom® CPU E640 @ 1.00 GHz
* 1 GB RAM
 |
| Beagle Bone |   | * AM335x 720MHz ARM Cortex-A8
* 256MB DDR2 内存
 |
| HP ProLiant DL320 Server | Centos 6.7和JDK 1.7 | * Intel Xeon 3.60 GHz
* 32 GB RAM
 |
| Qualcomm DragonBoard 410 | 基于Ubuntu的Linux，Oracle JDK 1.7 | * ARM Cortex A53 1.2 GHz
* 1 GB RAM
 |
| GEIP MFA | Pengutronix Linux, Oracle Embedded JDK 1.7 | * ARM 600 MHz
* 512MB RAM
 |
| GEIP PAX Rxi | Win7 Pro, Oracle JDK 1.7 | * VIA Eden X2 U4200 1.2 GHz
* 4 GB RAM
 |
| Intel MoonIsland on Advantech UTX 3115 | Wind River Linux, JDK 1.7 | * Intel Atom Dual Core E3826 1.46 GHz
* 2 GB RAM
 |
| Cisco829 | VM Running on Linux, Oracle JDK/Open JDK 1.7 | * Intel Atom Dual-Core Rangeley CPU 1250 MHz
* 2GB RAM
 |
| PACSystems RXi ICRXIBN7E000A | Windows 7 Professional, SP1 | RXi Box IPC，带嵌入式32GB mSATA SSD |
| PACSystems RXi ICRXIBN7M000A | Windows 7 Professional, SP1 | RXi Box IPC，带250GB Magnetic SATA硬盘 |

# 下载Predix机器SDK

如果要访问下载网址，您必须拥有一个predio.io账号。

如果要开始使用SDK，您必须首先下载SDK包。

1. 访问Predix机器SDK下载页面 https://artifactory.predix.io/artifactory/PREDIX-EXT/predix-machine-package/predixmachinesdk/16.4.0/predixmachinesdk-16.4.0.zip。
2. 下载PredixMachineSDK-16.4.0.zip文件。
3. 解压并提取ZIP文件中的所有文件。

## 发放许可

您下载Predix机器SDK时，您会接收到一个非生产许可密钥。如果要获取生产许可密钥，请联系Predix机器产品管理人员。

下表列出了Predix机器SDK版本非生产许可密钥的过期日期。

| **Predix机器SDK版本** | **过期日期** |
| --- | --- |
| 16.3 | 2018-01-01 |
| 16.2 | 2017-01-01 |
| 16.1 | 2017-01-01 |

# Predix机器SDK目录结构

解压后，下载的SDK文件创建以下目录结构：

| **目录** | **描述** |
| --- | --- |
| docs | 包含SDK文件和带有Javadoc API的apidocs.zip文件。 |
| eclipse-plugins | 表示您将Eclipse安装指向的位置。包括以下文件夹：* 特性
* 插件
 |
| edgesdk | 包含一个多语言SDK，您可以用它编写应用程序，以便通过数据总线与设备进行通信。  |
| license | 包含许可证文件。 |
| samples | 包含用于样例应用程序的sample-apps.zip和sample-cloud-apps.zip文件。 |
| utilities | 包含生成容器所用的脚本。 |
| InstallationGuide.pdf | Predix机器软件开发套件安装指南。 |

# 安装Predix机器SDK

要安装Predix机器SDK，遵循以下步骤。

1. 打开Eclipse。

出现欢迎来到Eclipse的页面。

1. 在帮助菜单上，选择安装新软件。

出现可用软件窗口。

注：如果您刚刚下载了最新版的Eclipse，您应该取消勾选在安装过程中联系所有更新站点以查找所需软件复选框，以防止Eclipse为所有安装包搜索更新。如果您的Eclipse版本比较旧，您可以选中该复选框执行此程序，但是可能会花费较长时间。

1. 点击添加按钮。

出现添加存储库对话框。


1. 点击本地按钮然后浏览您将Predix机器SDK文件解压的安装位置，选择eclipse-plugins文件夹，并点击确定。在添加存储库对话框中点击确定。

现在，PredixSDK和PredixSDK Maven Support支持选项出现在可用软件的列表中。


1. 选择以下安装选项：
	* Predix SDK
	* PredixSDK MavenSupport

***注：如果您已经安装了Maven集成（m2eclipse）插件，仅选择PredixSDK MavenSupport即可。如果未安装M2E，在您安装M2E或用于Java JEE开发人员的Eclipse IDE之前，安装无法继续。这可以在Eclipse市场中找到。***

1. 在详细信息部分，选择您想要对您的安装进行定制的选项。您可以使用默认选择。点击下一步按钮。

出现安装详细信息窗口。


1. 在安装详细信息窗口上，点击下一步按钮。

出现审核许可证窗口。

1. 审核许可协议的条款然后选择我接受许可协议的条款选项并点击完成按钮。

出现安装软件进度条。


***注：如果出现安全警告，请点击确定继续。***

1. 软件安装完成时，点击是按钮重启Eclipse。

如果您还没有为Eclipse创建工作区，将打开选择一个工作区对话框。


请选择工作区位置并点击确定。

## 启动Eclipse

当您在安装Predix机器SDK之后启动Eclipse时，Eclipse将配置您的环境以使用该SDK。启动过程中，将提示您切换到Eclipse插件开发环境（PDE）。默认的PDE配置使用OSGi运行时组件。如果您没有切换，可以稍后切换。

1. 启动Eclipse。

出现默认目标平台对话框。


1. 点击切换。

## **捆绑包移植**

当升级Predix机器版本时，捆绑包可能会被重命名、重新确定因数或移除。为了简化升级，您可以移除您镜像中的所有捆绑包然后全部用新的捆绑包进行替换。您可以通过在添加捆绑包对话框中选择相应的捆绑包类型为特定特性添加新的捆绑包。说明请参见[使用Eclipse生成一个Predix机器容器](https://www.predix.io/docs/?r=62886#yh8q25xt)。

# 生成Predix机器容器

您可以通过使用Eclipse中的Predix机器SDK或命令行脚本生成Predix机器运行时容器。您也可以创建Predix机器用作一个Docker镜像 或将一个现有的Predix机器容器转换为一个Docker镜像。

## **Predix机器运行时容器类型**

可以生成以下类型的Predix机器运行时容器：

* Predix机器默认容器：如果您不指定容器类型，将创建默认容器
* Predix机器代理：为Docker支持提供带有代理特性的预定义Predix机器。
* Predix机器代理调试：为Docker支持提供带有代理特性的调试Predix机器。
* Predix机器调试：提供Predix机器网页控制台
* Predix机器配置：提供配置支持
* Predix机器定制：使用您在Eclipse中创建的定制镜像来定制Predix机器容器
* Predix机器技术人员控制台镜像
* Predix机器连接：提供连接支持

如果您使用Eclipse中的Predix机器SDK或使用命令行脚本，您指定容器类型的方法不同。

* 如果您使用Eclipse中的Predix机器 SDK，您在创建新的镜像描述并选择工作区镜像时指定容器类型。

*注：参见*[*使用Eclipse生成一个Predix机器容器*](https://www.predix.io/docs/?r=62885#yh8q25xt)

* 如果您正在使用脚本生成容器，您可以使用以下命令指定容器类型：
	+ AGENT：带有代理支持的Predix机器
	+ AGENT\_DEBUG：用于Docker支持带有代理特性的Predix机器调试。
	+ PROV：Predix机器配置（仅包括支持配置的JAR捆绑包）
	+ DEBUG：带有Predix机器网页控制台的Predix机器调试
	+ TECH：技术人员控制台镜像
	+ CONN：Predix机器连接
	+ CUSTOM <镜像文件路径>：使用您在Eclipse中创建的定制镜像的一个Predix机器容器
	+ [未指定]：Predix机器默认容器

## 使用命令行脚本生成一个Predix机器运行时容器

## 在开始之前

* 从<https://artifactory.predix.io/artifactory/PREDIX-EXT/predix-machine-package/predixmachinesdk/16.4.0/predixmachinesdk-16.4.0.zip>下载Predix机器。

*提示：对于Windows，将您的SDK下载放在您的计算机中一个较高的目录中。例如，避免c:\\<directory>\<directory>\<directory>\Predix Machine SDK而使用类似于c:\\PredixMachineSDK来代替。如果您嵌套了多个目录中的下载，当您尝试运行容器时路径可能会过长。*

* 下载带有PDE运行时插件的Eclipse；例如用于Java EE开发人员的Eclipse IDE。该下载应仍保留.zip或tar.gz格式。
* 确保您已经安装Maven。在一个命令行界面上，输入mvn -version。您的版本应该为3.1或更高。

在命令行中使用以下命令生成一个Predix机器运行时容器：

* -e <ECLIPSE\_PATH>：下载的Eclipse ZIP/TAR文件的路径。
* -c <CONTAINER\_TYPE>：要创建的Predix机器容器的类型。

*注：关于生成Predix机器用作一个Docker镜像的信息，请参见*[*使用命令行脚本生成一个Predix机器用作一个Docker镜像*](https://www.predix.io/docs/?r=62883#EXl0CrR)*。*

您可以生成以下容器类型：

* + AGENT：用于Docker支持带有代理特性的Predix机器。在Docker化的模型中运行时需要该容器，以便与边缘SDK一起使用。
	+ AGENT\_DEBUG：用于Docker支持带有代理特性的Predix机器调试。
	+ PROV：Predix机器配置（仅包括支持配置的JAR捆绑包）
	+ DEBUG：带有Predix机器网页控制台的Predix机器调试
	+ TECH：技术人员控制台镜像
	+ CONN：Predix机器连接镜像
	+ CUSTOM <镜像文件完整路径>：使用您在Eclipse中创建的定制镜像的一个Predix机器容器
	+ [未指定]：Predix机器默认容器

*注：这其中的每个容器类型映射至Eclipse中Predix SDK中的镜像描述对话框中的工作区镜像选择。*

步骤

1. 打开一个终端窗口。
2. 在命令行中，导航至<SDK download location>/predixmachinesdk-16.4.0/utilities/containers文件夹。
3. 运行以下其中一个命令：
	1. Windows：

GenerateContainers.bat -e <full path and name of downloaded Eclipse.zip file> -c <type of container>

* 1. UNIX和Linux：

GenerateContainers.sh -e <full path and name of downloaded Eclipse.tar.gz file> -c <type of container>

1. 例如，在Windows中：

GenerateContainers.bat -e D:\users\16.4.0\SDK\eclipse-jee-mars-SR2-win32-x86-64.zip -c PROV该脚本在<SDK download location>/predixmachinesdk-16.4.0/utilities/containers文件夹中创建Predix机器运行时容器。

注：如果您接收到以下控制台错误，您可以忽略： java.lang.ClassCastException: org.eclipse.osgi.internal.framework.EquinoxConfiguration$1 cannot be cast to java.lang.String at org.eclipse.m2e.logback.configuration.LogHelper.logJavaProperties(LogHelper.java:26) at org.eclipse.m2e.logback.configuration.LogPlugin.loadConfiguration(LogPlugin.java:189) at org.eclipse.m2e.logback.configuration.LogPlugin.configureLogback(LogPlugin.java:144) at org.eclipse.m2e.logback.configuration.LogPlugin.access$2(LogPlugin.java:107) at org.eclipse.m2e.logback.configuration.LogPlugin$1.run(LogPlugin.java:62) at java.util.TimerThread.mainLoop(Timer.java:555) at java.util.TimerThread.run(Timer.java:505)

## 使用Eclipse生成一个Predix机器运行时容器

遵循以下步骤使用Eclipse生成一个Predix机器运行时容器。

1. 在Eclipse中，选择文件 > 新建 > 项目创建一个项目，在其中创建文件和文件夹。
	1. 在导航器面板中，右击并选择新建 > 项目。

出现新建项目对话框。


* 1. 选择项目向导并点击下一步。

注：如果项目仅包含容器镜像，请选择常规 > 项目。

在新建项目向导中出现项目页面。



* 1. 在新建项目向导的项目页面中，为您的项目分配一个项目名称并点击完成。

您的项目出现在导航器窗格中。

1. 创建一个Predix机器SDK镜像文件。
	1. 在导航器面板中右击，并选择新建 > 镜像描述。

出现新建镜像描述窗口。


* 1. 在文件名文本框中，为文件输入一个名称。
	2. 在从镜像继承部分，为工作区镜像选择以下其中一个选项并点击完成。
		+ Predix机器默认容器：如果您不指定容器类型，将创建默认容器
		+ Predix机器代理：为Docker支持提供带有代理特性的预定义Predix机器。
		+ Predix机器代理调试：为Docker支持提供带有代理特性的调试Predix机器。
		+ Predix机器调试：提供Predix机器网页控制台
		+ Predix机器配置：提供配置支持
		+ Predix机器技术人员控制台镜像

镜像被创建。


1. 为您将构建的容器添加捆绑包。
	1. 在捆绑包部分，点击添加按钮。

出现添加捆绑包窗口。

* 1. 在捆绑包类型列表中，选择Predix特性组为各种特性添加必要的捆绑包。

出现每个Predix特性的列表。



参见[Predix机器SDK概述](https://www.predix.io/docs/?r=62882#MgHwD2pM)查看Predix机器捆绑包的列表，它是每个Predix机器特性的一部分。

注：当您打开镜像文件时Predix机器SDK将提示您更新镜像编辑器文件。在更新过程中，旧版本将更新为新版本。

* 1. 选择在容器中包括的特性并点击确定。

注：如果您正在为镜像添加一个Maven捆绑包，Eclipse将创建一个用于该镜像的绝对路径。同样，编辑pom.xml以便将绝对路径更改为一个相对路径。例如：

<element type="com.prosyst.tools.builder.osgiBundle" location=maven:rel:../../../../gut/PredixMachine/master/machine-ge/ge-httpclient/pom.xml

1. 生成容器。
	1. 在测试部分，点击导出镜像链接。

出现OSGI镜像制作器导出向导。


* 1. 在工作区镜像列表中，选择您在第2步中创建的镜像。
	2. 在目标文本框中，浏览至容器的目标位置并点击完成。

向导构建容器并将其存储在目标位置。

注：您也可以使用运行、调试和分析选项代替导出。这样您就可以测试您想要添加的服务。

分析会话限制为5到10分钟。

在Linux上使用Eclipse时，分析器无法正确工作，除非在启动Eclipse之前设置 LD\_LIBRARY\_PATH 。必须将其设置为包括<workspace root>/.metadata/.plugins/com.prosyst.tools.mbsemulator/images/<image name>/osgi/lib/mbprofiler-agent/runtimes/linux-x86\_64-generic其中<workspace root>是所用Eclipse工作区的位置，而<image name>是您创建的镜像文件的名称。

1. 启动生成的容器。
	1. 在您的计算机上，导航至<exported location>//bin/文件夹。
	2. 对于Windows来说，双击start\_predixmachine.bat文件，对于Linux/Mac来说双击start\_predixmachine，或运行bash脚本 start\_predixmachine。

## Predix机器目录结构

生成的Predix机器容器具有以下目录结构：

| **目录** | **描述** |
| --- | --- |
| appdata | 作为最佳实践，服务只能拥有此文件夹的读写权限。包含用户通过Predix机器网页控制台/技术人员控制台上传的文件。包含以下子目录： * GitRepositories

包含用于创建和复制Git存储库的Git捆绑包。所有Git存储库应存储在该目录中。  |
| bin | 包含用于启动和停止Predix机器的start\_predixmachine和stop\_predixmachine脚本。 |
| configuration | 用于捆绑包的配置文件。包含以下子目录：* machine

Predix机器捆绑包和任意定制捆绑包的配置文件 |
| installations | 监控该文件夹的Yeti进程，用于要安装的软件。软件必须是一个包括install.bat/install.sh文件在内的ZIP文件。 |
| license | 终端用户许可协议和开源软件的列表。  |
| logs | 运行时日志。包含以下子目录：* machine

Predix机器的运行时日志* installations

Yeti安装日志 |
| machine | 包括以下子目录的容器目录： * bin/predix/

启动容器和停止容器脚本* bin/vms

solution.ini和所有其他ini文件* bundles

样例应复制到此处。* configs

捆绑包com.prosyst.mbs.osgi.metatyp.bundle中包含的OSGi XML文件* install

安装脚本* lib/framework/\*.jar

包含运行时库。 |
| security | 安全策略和密钥存储。在部署环境中，应仅限具有管理权限的用户访问此文件夹。  |
| yeti注：仅限配置容器。 | 包含将Predix机器用作服务启动和安装脚本的service\_installation文件夹。还包括将Yeti用作看门狗的文件和脚本。 重要提示：请勿移动此文件夹中的文件。 |

# 用户许可和角色

如果要使用Predix机器，您必须符合要求部分所述的技术要求。您还必须拥有访问Predix机器SDK下载网址的Predix.io账号。

您下载Predix机器SDK后，将推荐使用特定用户/角色和许可，以保证创建和运行Predix机器运行时容器进程的安全，并通过EdgeManager将设备配置到云上。

下表列出了使用的多种用户/角色和设置，而且包括特定角色的默认或推荐用户名。

| **用户/角色** | **用户名（如果适用）** | **描述/功能** | **相关文档** |
| --- | --- | --- | --- |
| Predix.io/Cloud Foundry用户 | （您的Predix.io登录） | 允许用户下载Predix机器SDK。 | [注册Predix平台用户账号](https://www.predix.io/docs/?r=62881#hRtX6xs6)[下载Predix机器SDK](https://www.predix.io/docs/?r=62880#dIlIc5DH)[配置用户代理和存储库设定值](https://www.predix.io/docs/?r=62879#nzHqVhxN) |
| Predix机器管理员 | 默认是 predix。首次登录时必须更改密码。 | 创建用户，配置网页控制台插件，更改Predix机器网页控制台登录。创建 predixuser 保护安全文件夹的安全用户。 | [管理用户](https://www.predix.io/docs/?r=62878#JvCwOksc)[配置Predix机器网页控制台插件](https://www.predix.io/docs/?r=62877#zjf8c64s)[管理服务（用户）](https://www.predix.io/docs/?r=62876#Wao9tNn) |
| 安全用户 | predixuser  | 不是管理员或根用户的用户。确保安全文件夹安全。仅限有权读/写安全文件夹的用户。 运行Predix机器容器。重要提示：作为最佳实践，具有有限管理权限的该用户应运行容器。根/管理员不得运行Predix机器。  | [确保Predix机器安全](https://www.predix.io/docs/?r=62875#ebgcpVl) |
| EdgeManager管理员 |   | 管理访问和许可。创建用户组，导入设备，创建用户和为用户分配角色，以及更改用户密码。 | [创建用户和分配用户角色](https://www.predix.io/docs/?r=62874#P5BzBjBT)[用户组概述](https://www.predix.io/docs/?r=62873#qTLtG6GH) |
| EdgeManager操作员 | （由EdgeManager管理员分配） | 访问EdgeManager中的设备管理器和存储库。  | [设备管理器概述](https://www.predix.io/docs/?r=62872#aHbMSNyS)[将软件上传到存储库](https://www.predix.io/docs/?r=62871#zTbBNKUC) |
| EdgeManager技术人员 | （由EdgeManager管理员分配） | 仅查看EdgeManager中的警报和设备软件。在Predix机器网页控制台/技术人员控制台中拥有云登记许可 | [登记一台启用Predix机器的设备](https://www.predix.io/docs/?r=62870#XDY6RsjG-ICpcbL20) |

# 启动Predix机器

您可以启动Predix机器，这样它就可以启动您已经安装的运行时容器和其他Predix机器服务，例如Yeti。

注：如果要将Predix机器作为服务运行，请参见[将Predix机器作为服务运行](https://www.predix.io/docs/?r=62869#eqHPCWXR)。

1. 导航至<Predix Machine runtime container location>/bin.
2. 使用以下其中一条命令启动Predix机器：

start\_predixmachine.bat（Windows）

start\_predixmachine.sh（Linux和Mac）

# 将Predix机器作为服务运行

您可以在Windows和Linux环境中作为服务安装Predix机器，这样您无需手动启动Predix机器配置容器，它可以在用户交互最少的情况下运行。服务监控容器并在更新推送到Predix机器时安装更新。

注：只有生成一个配置（-PROV）容器类型时，您才能将Predix机器作为服务运行。

## 安装Predix机器Windows服务

您可以在Windows环境中安装Predix机器服务。服务安装设置文件在<Predix Machine runtime container location>bin/service\_installation目录中。

Windows服务的安装使用了.Net Framework的工具软件InstallUtil.exe。这样既可以在命令行中安装也可以无人守值安装。

重要提示：因为服务在启动和停止脚本时使用绝对路径，安装完服务后切勿移动容器位置。如果你移动了容器，服务就不能启动Predix机器。

## 在您开始前

* 创建一个称为 predixmachineuser 具有有限权限的用户账号，以运行服务。您不得以系统管理员身份运行服务。
* 您必须生成一个配置（-PROV）容器类型，以使用该服务。
* 您必须已经安装.NET Framework 4.5.2或更高版本。

## 过程

1. 打开一个命令行窗口，导航至<Predix Machine runtime container location>bin/service\_installation目录。
2. 在命令行中输入setup.bat启动安装脚本。
3. 输入用户名 predixmachineuser，包括域名，例如domain\predixmachineuser
4. 输入密码，安装过程就结束了。
5. 为了执行无人值守安装：
	1. 使用管理员权限打开一个命令行窗口，导航至<Predix Machine runtime container location>bin/service\_installation目录。
	2. 运行如下命令：

InstallUtil.exe /username=yourdomain\yourusername /password=yourpassword /unattended PredixMachineService.exe

安装过程就结束了。默认情况下，服务设置为自动启动，所以开机后服务就启动了。

安装之后，您可以通过具有 predixmachineuser 权限的管理员账号启动服务

注：如果启动服务时您遇到Error Code 3，说明Machine容器的结构有问题。请确保PredixMachineService.exe 位于 bin\service\_installation 目录。如果启动服务时您遇到Error Code 5 拒绝访问，通常意味着在安装时指定的用户对Predix机器容器目录没有完全控制权。如果是这样，请参考步骤6.

1. 为predixmachineuser 赋予Predix机器容器目录的完全控制权：
	1. 右击容器文件夹并选择属性。
	2. 点击安全选项卡并点击编辑按钮更改许可。
	3. 对于完全控制许可，选中允许复选框。

下一步：如果您想要卸载服务，运行以下命令：

<Predix Machine runtime container location>\bin\service\_installation \uninstall.bat

## 启动Predix机器Windows服务

安装服务之后，您可以手动启动Predix机器服务。否则，服务将在开机时启动。

1. 在您的Windows机器上，打开Windows任务管理器。
2. 点击服务选项卡。
3. 在服务选项卡的右下角，点击服务。
4. 找到Predix机器服务并点击启动。

## 安装Predix机器Linux服务

您可以在Linux环境中安装Predix机器服务。

## 在您开始前

您必须生成一个配置（-PROV）容器类型，以使用该服务。

服务安装设置文件在<Predix Machine runtime container location>bin/service\_installation目录中。

重要提示：因为服务在启动和停止脚本时使用绝对路径，安装完服务后切勿移动容器位置。如果你移动了容器，服务就不能启动Predix机器。

## 过程

1. 导航至<Predix Machine runtime container location>bin/service\_installation。
2. 调用setup.sh脚本启动安装程序。

下一步：如果您想要卸载服务，可以调用<Predix Machine runtime container location>bin/service\_installation/uninstall.sh脚本

## 启动Predix机器Linux服务

您可以使用systemd版本225在Linux系统上启动Predix机器服务。自从2015年起，在许多Linux发行版中，Systemd在安装时用作默认初始化系统。您也可以使用systemctl版本215在Raspberry Pi上运行服务。

1. （可选）使用以下命令重新扫描systemd查找新的或更改的装置：

systemctl –user daemon-reload

您指定这是一个用户服务而非系统服务。

1. 使用以下命令启动服务：

systemctl –user start predixmachine

您可以忽略服务配置文件名末尾的.service文件类型。

1. （可选）如果您想要在启动时重新启用服务，可以使用以下命令：

systemctl –user enable predixmachine

这会在.config/systemd/user目录中的服务与 .config/systemd/user/default.target.wants/predixmachine.service目录之间创建一个超链接。请勿自行创建该超链接。这是系统在开机时进行搜索以启动服务的位置。

1. （可选）如果您想要在启动时禁用服务，可以使用以下命令：

systemctl --user disable predixmachine

这会在启动时移除超链接并禁用服务。

# 构建并运行样例应用程序

提供三组样例应用程序，以展示如何使用Predix特性：一个用于Predix机器运行时容器，一个用于云应用程序，一个用于边缘应用程序。

如果要查看容器或云样例，请导航至<SDK installation location>/samples并从sample-apps.zip、sample-cloud-apps.zip中提取文件。

sample-apps.zip文件包括：

* sample-basicmachineadapter
* sample-configuration
* sample-container
* sample-custompolling
* sample-databus（OSGi样例）
* sample-gitrepository
* sample-healthmachineadapter
* sample-hoover
* sample-httpclient
* sample-httpriver
* sample-mqttclient
* sample-mqttmachineadapter
* sample-security
* sample-storeforwardclient
* sample-subscriptionmachineadapter
* sample-websocketclient
* sample-websocketriver
* sample-websocketserver

注：参见[从命令行构建样例](https://www.predix.io/docs/?r=62868#yO80iEfd)和[在Eclipse中运行样例](https://www.predix.io/docs/?r=62867#UcPgtwmh)或[使用Eclipse IDE构建样例](https://www.predix.io/docs/?r=62866#MN0PfR2c)。

sample-cloud-apps.zip文件包括：

* httpdata
* httptunnel-server

注：关于如何使用样例的说明，请阅读每个云样例的相关readme.txt文件。

如果要查看边缘样例，请导航至：<Predix Machine SDK download location>/edgesdk/predixmachine-edgesdk-java-16.4.0.zip或<Predix Machine SDK download location>/edgesdk/predixmachine-edgesdk-cpp-16.4.0.zip并解压这些文件。

导航到解压出的文件夹，在samples子文件夹里从sample-edge-apps.zip中提取文件。

sample-edge-apps.zip文件包括：

* edge-container
* sample-edge-databus（单机样例）

## 从命令行构建样例

1. 生成一个API密钥。
	1. 登录[https://artifactory.predix.io](https://artifactory.predix.io/)。

注：使用您的predix.io账户用户名和密码登录。

* 1. 在屏幕右上角，点击您的用户名。
	2. 输入您的密码。
	3. 点击解锁以生成API密钥字段。API密钥的值是隐藏的。
	4. 复制密钥。
1. 在<user directory>.m2/settings.xml文件中：
	1. 根据您自己网站的需求为Maven配置代理设定值。如果关于您的要求有问题，请询问您的网络管理员。
	2. 添加一个具有以下信息的服务器实体。（您将用到您在第1步复制的API密钥。）

 <server><id>artifactory.external</id><username>{your predix cloud login}</username><password>{encrypted password - API key}</password></server>

* 1. 您还可能必须为HTTPS协议设置代理设定值。

 <proxy><id>optional</id><active>true</active><protocol>http</protocol><username>proxyuser</username><password>proxypass</password><host>proxy.host.com</host><port>80</port><nonProxyHosts>\*.host.com|localhost</nonProxyHosts></proxy>

1. 如果要从命令行构建样例：
	1. 导航至<SDK installation location>/samples并解压sample-apps.zip或sample-cloud-apps.zip文件。
2. 如果要构建单独的样例：
	1. 导航至<SDK installation location>/sample/sample-apps/sample/<sample-name>。
	2. 运行以下命令：

mvn clean install

注：某些样例需要将项目推送至artifactory。（例如，MQTT客户端和云样例。）您还必须将一个第三方JAR文件推送至您的本机m2进行构建。.说明请参见readme.txt文件。

## 在Eclipse中运行样例

## 在您开始前

使用Eclipse创建一个新容器。参见[使用Eclipse生成一个Predix机器运行时容器](https://www.predix.io/docs/?r=62864#yh8q25xt)。

您可以使用Predix机器SDK运行样例。

## 过程

1. 访问Eclipse并打开您的Predix机器镜像。
2. 在捆绑包部分，点击添加按钮。

出现添加捆绑包窗口。

1. 在捆绑包类型列表中，选择Predix样例组。

出现每个Predix样例的列表。

1. 选择您想要运行的样例并点击确定。
2. 点击运行。

## 从生成的容器运行样例

您可以从使用脚本生成的容器运行样例。

从使用脚本生成的容器运行任何样例之前，请确保项目已构建。

注：开始之前，确保solution.ini文件要么被复制到或者被创建于machine/bin/vms目录。

1. 从<SDK installation location>/samples/sample-apps/sample/<sample-name>将JAR文件复制到<Predix Machine installation location>/machine/bundles/。
2. 从<SDK installation location>/samples/sample-apps/sample/configuration/machine目录将样例需要的所有配置文件复制到<Predix Machine runtime container location>/configuration/machine目录。
3. 修改solution.ini文件，通过为每个样例添加一个 <bundle> 标签实现。例如：

<bundle><name>com.ge.dspmicro.{sample-name}-{version}.jar</name></bundle>

您也可以复制并粘贴用于所有样例的现有solution.ini文件并修改该文件。solution.ini文件位于<SDK installation location>/samples/sample-apps/sample/machine/bin/vms。

1. 如果要运行容器：
	1. 导航至<Predix Machine runtime container location>/bin。
	2. 运行以下命令：start\_predixmachine.sh （对于Linux）或start\_predixmachine.bat（对于Windows）。

## 使用Eclipse IDE构建样例

使用您将Predix机器SDK安装在其中的Eclipse IDE构建样例。

1. 生成一个API密钥。
	1. 登录[https://artifactory.predix.io](https://artifactory.predix.io/)。
	2. 点击您的用户名。
	3. 输入您的密码。
	4. 点击解锁以生成API密钥字段。
	5. 复制密钥。
2. 在<user directory>.m2/settings.xml文件中：
	1. 根据您自己网站的需求为Maven配置代理设定值。如果关于您的要求有问题，请询问您的网络管理员。
	2. 添加具有此信息的服务器实体。（您将用到您在第1步复制的API密钥。）

 <server><id>artifactory.external</id><username>predix cloud login</username><password>{encrypted password - API key}</password></server>

1. 启动Eclipse并创建您自己的工作区。
2. 通过选择：文件 > 导入 > Maven > 现有Maven项目导入样例。点击下一步。
3. 浏览并选择<SDK installation location>/samples/sample-cloud-apps/sample或<SDK installation location>/samples/sample-apps/sample作为根目录。
4. 点击完成将所有样例导入您的工作区。
5. 选择样例根目录然后选择运行 > 运行方式 > Maven安装。

# 访问Predix机器网页控制台

## 在您开始前

启动一个Predix机器调试容器，或者，如果您正在执行设备登记，启动一个Predix机器配置（PROV）容器。

您可以使用Predix机器网页控制台在您的Predix机器环境中访问并管理捆绑包。

注：您只能在Predix机器DEBUG、PROV或TECH容器类型中访问Predix机器网页控制台；当您从Predix SDK生成容器时必须指定这其中一种类型。

## 过程

1. 打开一个网页浏览器并导航至：

https://localhost:8443/system/console/

显示一个登录提示。

1. 在提供的字段中输入您的用户名和密码。Predix机器使用以下默认值：

用户名：predix

密码：predix2machine

注：首次登录时，您必须更改密码。新密码必须为：

* + 长度至少为8个字符
	+ 至少包含一个小写字母
	+ 至少包含两个数字
	+ 密码中不包含用户名
	+ 以字母开始和结束

Predix机器控制台打开，使用Apache Felix网页界面。关于使用Apache网页控制台的更多信息，请参见<http://felix.apache.org/documentation/subprojects/apache-felix-web-console.html>。

1. 设置HTTP客户端代理。
	* 在Predix机器网页控制台中，点击OSGi > 配置。
	* 点击名称列中的Apache HttpClient OSGI捆绑包。例如，CM\_GENERATED\_PID.0。
	* 选中proxy.enabled.name复选框。
	* 在proxy.host.name框中，输入代理主机。
	* 在proxy.port.name框中，输入代理端口号。
	* 填写proxy.password.name和proxy.exceptions.name框并点击保存。

## 配置Predix机器网页控制台插件

作为管理员，您可以使用org.apache.felixe.webconsole.internal.servlet.OsgiManager.cfg文件配置Predix机器网页控制台> > OSGI菜单中以菜单项出现的插件，以限制技术人员可以执行的操作。

重要提示：请勿使用Predix机器网页控制台中的Apache Felix OSGi管理控制台页面（Predix机器网页控制台> > OSGI > 配置 > Apache Felix OSGi管理控制台）配置插件/菜单选项。如果您更改该页面，菜单项可能无法正确显示。

1. 在Predix机器技术人员容器中，导航至<Predix Machine runtime container location>/configuration/machine。
2. 打开org.apache.felix.webconsole.internal.servlet.OsgiManager.cfg文件。
3. 使用以下任意值配置 plugins 属性。如果要配置的插件多于一个，请使用逗号分隔值。

| **插件** | **数值** |
| --- | --- |
| 配置管理器 | org.apache.felix.webconsole.internal.configuration.ConfigManager  |
| 日志服务 | org.apache.felix.webconsole.internal.compendium.LogServlet  |
| 捆绑包 | org.apache.felix.webconsole.internal.core.BundlesServlet  |
| 服务 | org.apache.felix.webconsole.internal.core.ServicesServlet  |
| 许可证 | org.apache.felix.webconsole.internal.misc.LicenseServlet  |
| VM Stat | org.apache.felix.webconsole.internal.system.VMStatPlugin  |

以下示例显示通过配置管理器和日志服务插件填写的 plugin 值。

plugins=org.apache.felix.webconsole.internal.configuration.ConfigManager,org.apache.felix.webconsole.internal.compendium.LogServlet