



西安电子科技大学
XIDIAN UNIVERSITY

软件工程概论

软件部署

任课教师 宋娟 王徐华





内容

1. 软件及其环境

- ✓ 软件运行环境及其变化

2. 何为软件部署

- ✓ 软件部署的概念和原则

3. 软件部署的方式和方法

- ✓ 单机和分布式部署、多种软件部署方法

4. 软件部署的CASE工具

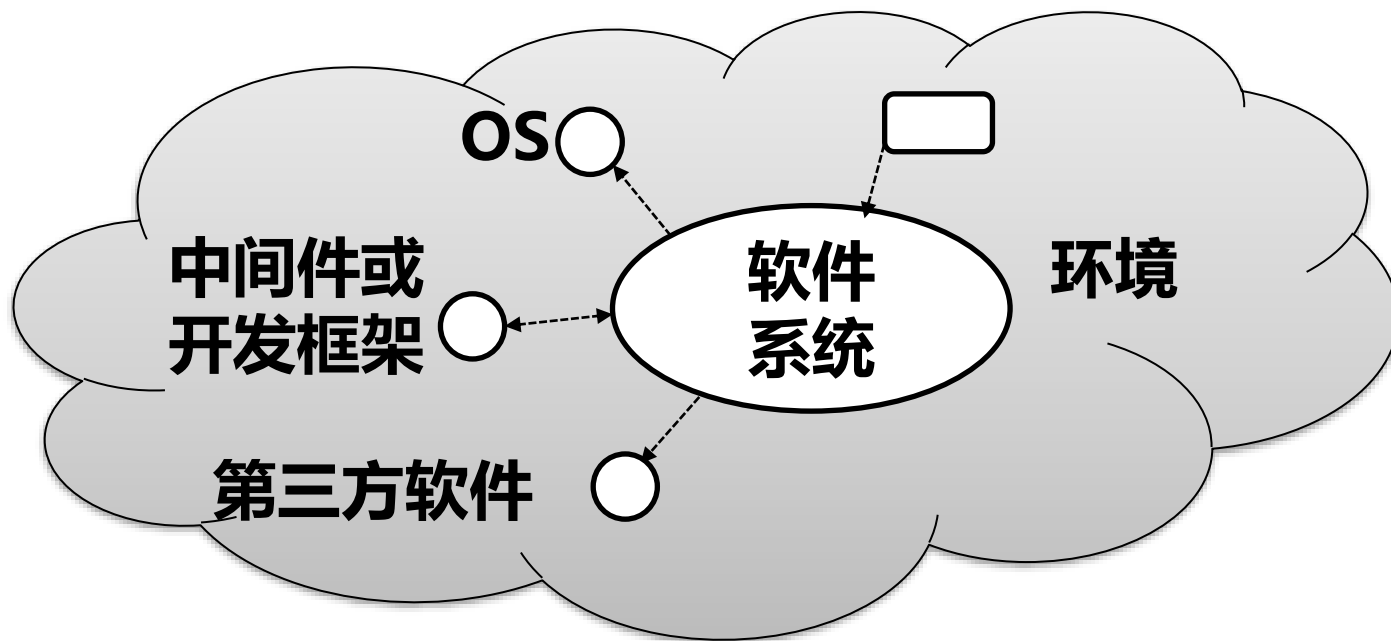
- ✓ Fat jar、Installer Projects、Jenkins等





1.1 软件运行环境

□ 软件运行所依赖的**上下文**，它为软件系统的运行提供必要的**基础服务和功能**、必须的数据和基本的**计算能力**





软件与其运行环境

□环境是软件赖以生存的**场所**

- ✓环境为软件运行提供各种要素，包括数据、计算、服务等等

□软件需要与环境进行持续的**交互**

- ✓软件通过环境获得基础服务和计算能力，环境通过软件获得相应的运行进程和数据等

□软件系统的运行环境可以表现为**多种形式**

- ✓既可以是物理和硬件设备（如计算机、服务器、机器人等），也可以不同抽象层次的软件系统

□软件系统的运行环境不仅包括**纵向层次**的基础软件及平台，还包括**横向层次**上与其运行相关的其他软件系统

1.2 软件间的关系

□纵向的层次性

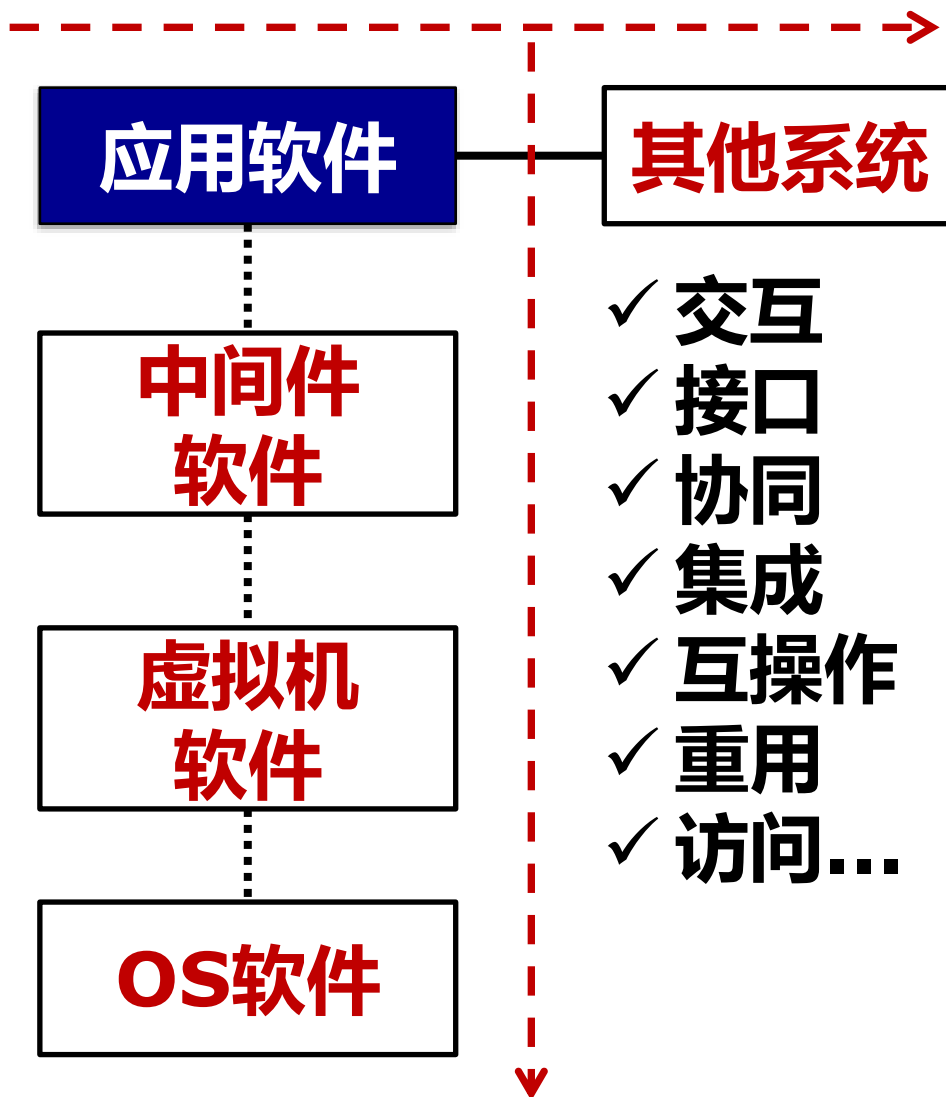
- ✓ 依赖于支撑其运行的基础软件系统

□横向的相关性

- ✓ 需要与其他的软件系统进行交互

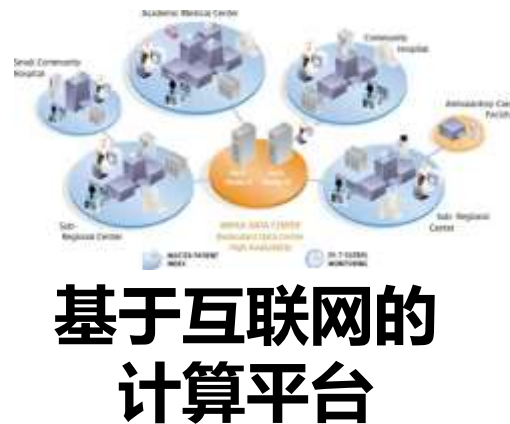
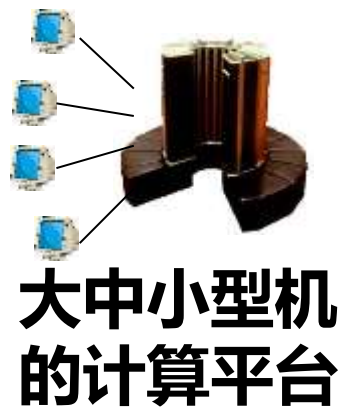
□软件部署需要考虑到

- ✓ 依赖的其他软件系统
- ✓ 相关的其他软件系统



1.3 软件运行环境的变化

从前端的PC终端、可穿戴设备、智能手机到后端的云中心、高性能计算中心



从孤立、独立、局域和可控的计算环境到分布、开放、动态、难控、无处不在计算环境



内容

1. 软件及其环境

✓ 软件运行环境及其变化

2. 何为软件部署

✓ 软件部署的概念和原则

3. 软件部署的方式和方法

✓ 单机和分布式部署、多种软件部署方法

4. 软件部署的CASE工具

✓ Fat jar、Installer Projects、Jenkins等





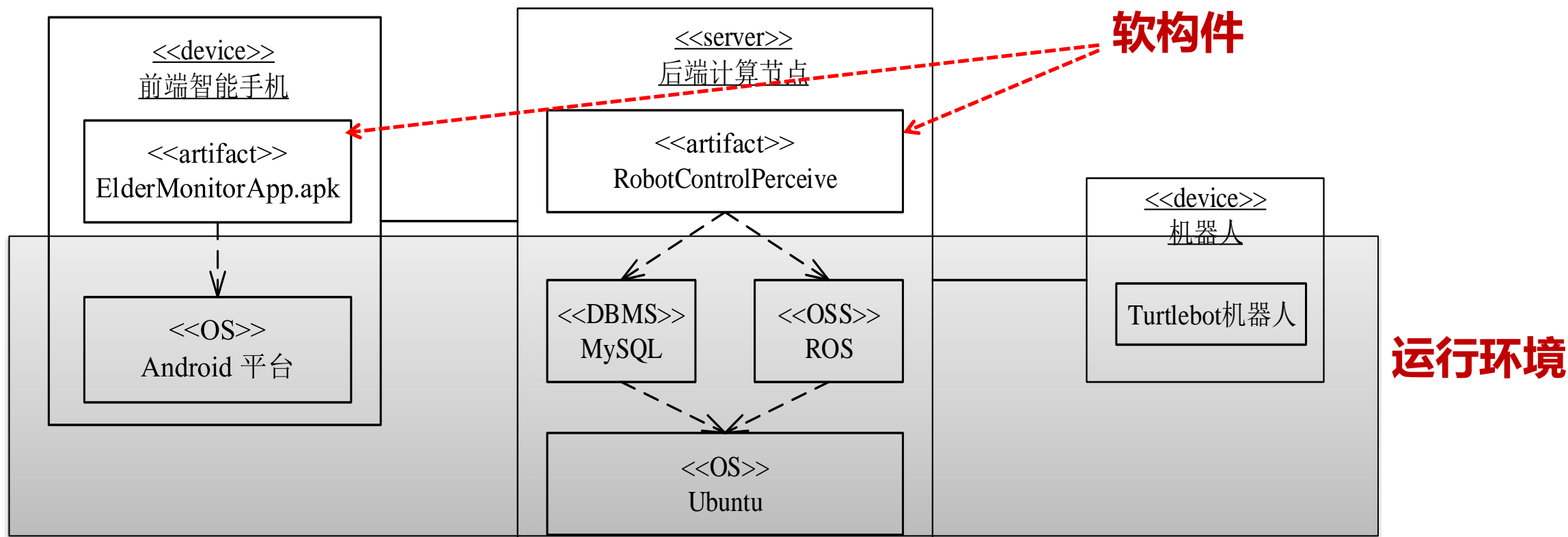
2.1 何为软件部署

- 软件部署是指将目标软件系统（包括软构件、配置文件、用户手册、帮助文档等）进行**收集、打包、安装、配置和发布到运行环境的过程**
- 涉及二方面的工作
 - ✓ 安装和配置运行环境
 - ✓ 安装和配置软件系统



2.1.1 安装和配置运行环境

- 运行环境是目标软件系统运行赖以生存的**上下文**
- 在将软件系统部署到运行环境之前，软件开发工程师首先需要安装和配置好运行环境，包括**构成运行环境的各类软硬件系统以及它们之间的相关性**

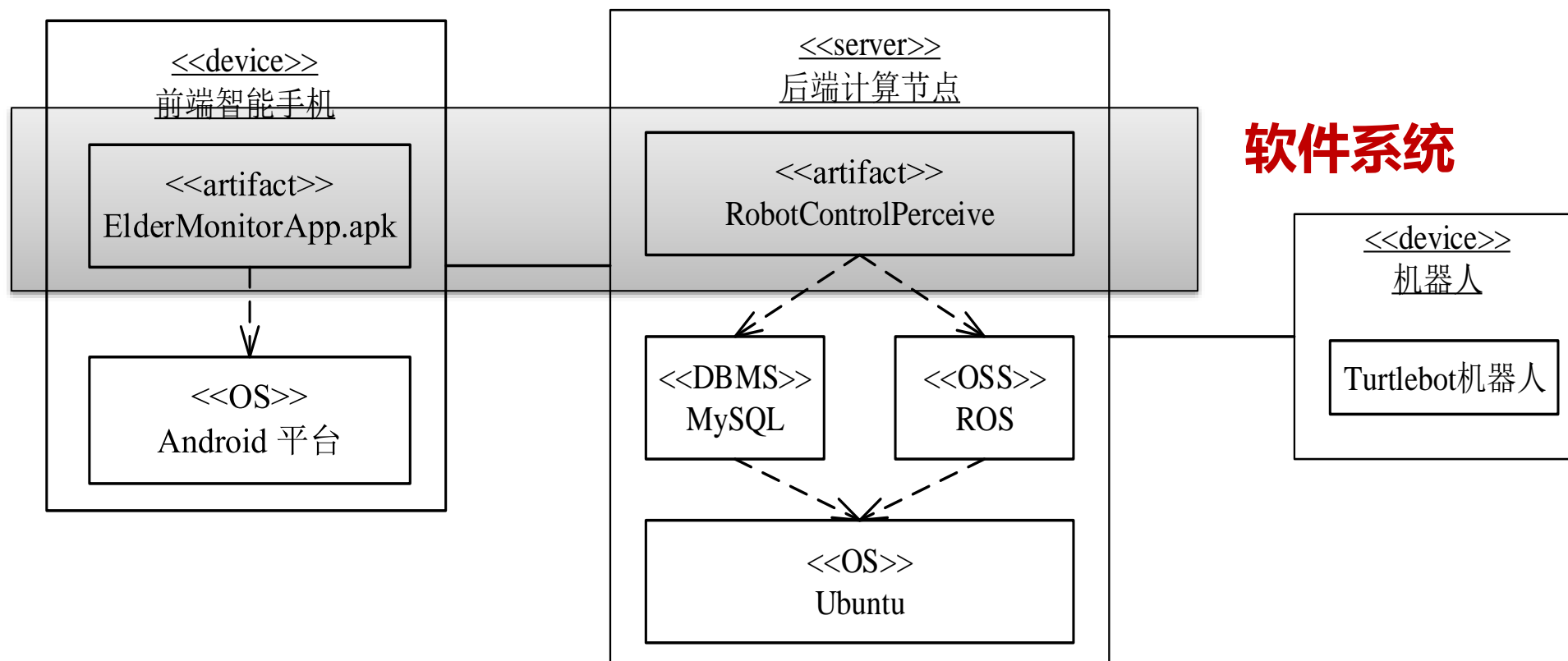




2.1.2 安装和配置软件系统

□收集和打包目标软件系统中需要安装的**软件要素**，然后将这些软件要素**安装到目标计算平台中**，并进行必要**配置**

✓包括软构件、所依赖软件包、软件文档和必要的**数据**



2.2 软件部署的原则

□ 最小化原则

- ✓ 只需安装、部署和配置支撑软件运行和服务提供的最少软硬件要素，以提高软件系统和运行环境的精简性，提升目标软件系统的运行效率，减低运行和维护成本

□ 相关性原则

- ✓ 部署的运行环境和软件系统要素均与系统建设相关联，剔除不相关的软硬件要素，防止将无关的软件要素部署到计算平台之中，以简化软件系统的部署和配置，降低软件运行和维护的复杂度

□ 适应性原则

- ✓ 当软件系统的运行环境发生变化时，目标软件系统的部署也要随之发生变化，以确保目标软件系统部署的灵活性，提高目标软件系统的健壮性，提升软件部署和运维的自动化程度



内容

1. 软件及其环境

✓ 软件运行环境及其变化

2. 何为软件部署

✓ 软件部署的概念和原则

3. 软件部署的方式和方法

✓ 单机和分布式部署、多种软件部署方法

4. 软件部署的CASE工具

✓ Fat jar、Installer Projects、Jenkins等





3.1 单机部署方式

□将软件的各个要素（如可运行软构件、数据、文档等）集中部署到某个**单一的**计算设备上

- ✓软件的运行环境只依赖于单一的计算设施
- ✓不同软构件之间不存在网络通讯
- ✓计算设施不仅仅是指各种计算机，如个人计算机、笔记本电脑或服务器等，还包括智能手机、智能手环等嵌入式计算设施

□典型示例

- ✓小米便签、闹钟和时钟、光盘刻录软件、扫雷游戏软件



3.2 分布式部署

□ 将软件的各个要素（如可运行的软构件、数据和文档等）
分散部署在**多个计算设备上的部署方式**

- ✓ 基于**C/S**的部署方式
- ✓ 基于**客户端-应用服务器-数据库服务器**的部署方式
- ✓ 基于**互联网**的软件部署方式

□ **典型示例**

- ✓ 空巢老人看护、Google搜索引擎、淘宝和中国铁路“12306”

3.3 软件部署方法

基于软件中间件、开发框架和容器部署的软件

基于操作系统部署的软件

软件中间件

软件开发框架

软件容器

操作系统层（如Windows、Linux、MacOS等计算机操作系统，及Android、IOS、鸿蒙等移动智能终端操作系统）

软件部署的目标平台（如计算机、服务器、移动智能终端等）



3.3.1 基于操作系统的部署

□ 软件系统直接部署在目标计算设施的操作系统之上运行

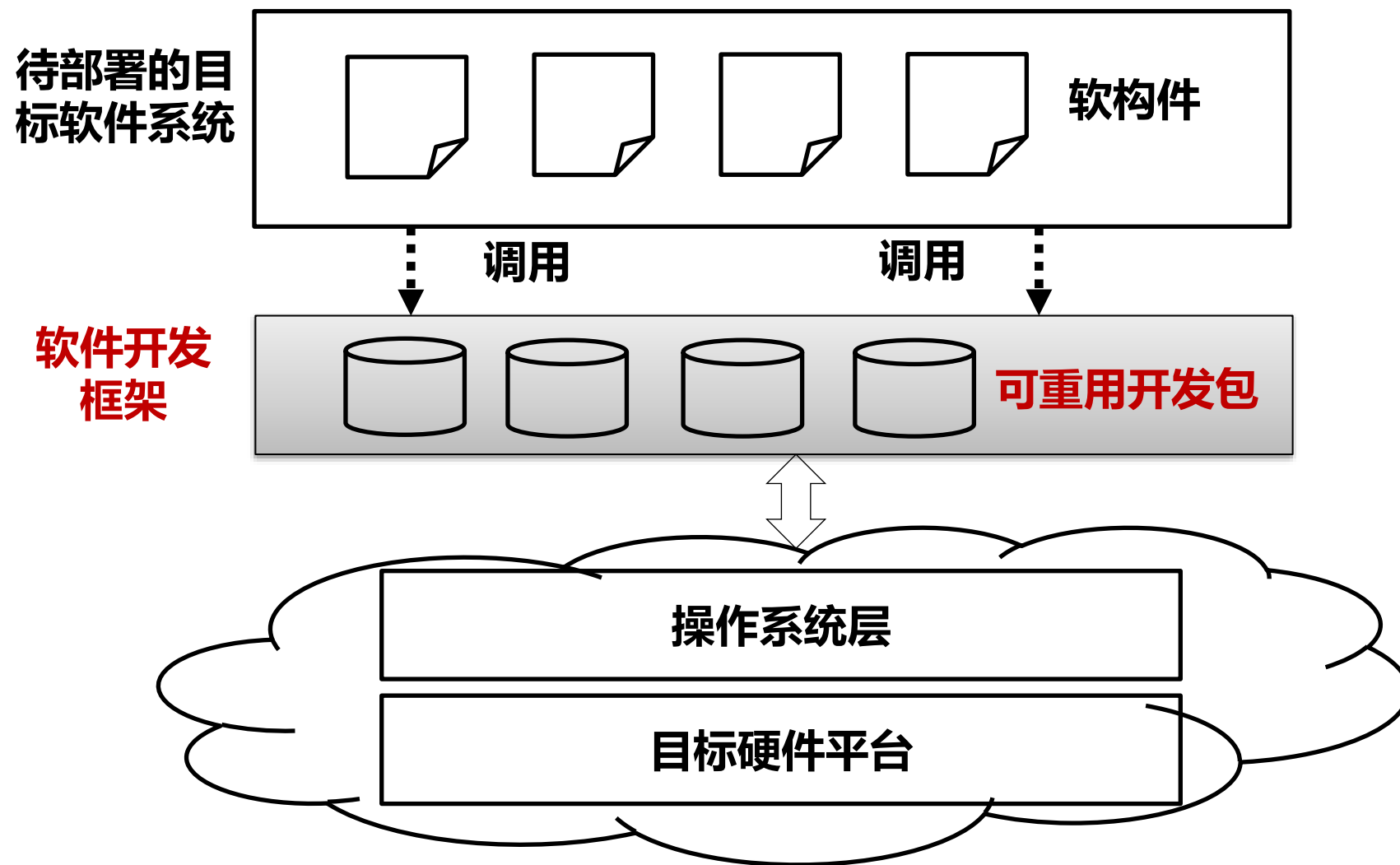
- ✓ 软件的运行**仅依赖于计算设施上的操作系统及其提供的基础服务** (如文件操作和管理、窗口界面的创建和操作等)
- ✓ 操作系统构成了软件运行的上下文环境

□ 典型示例

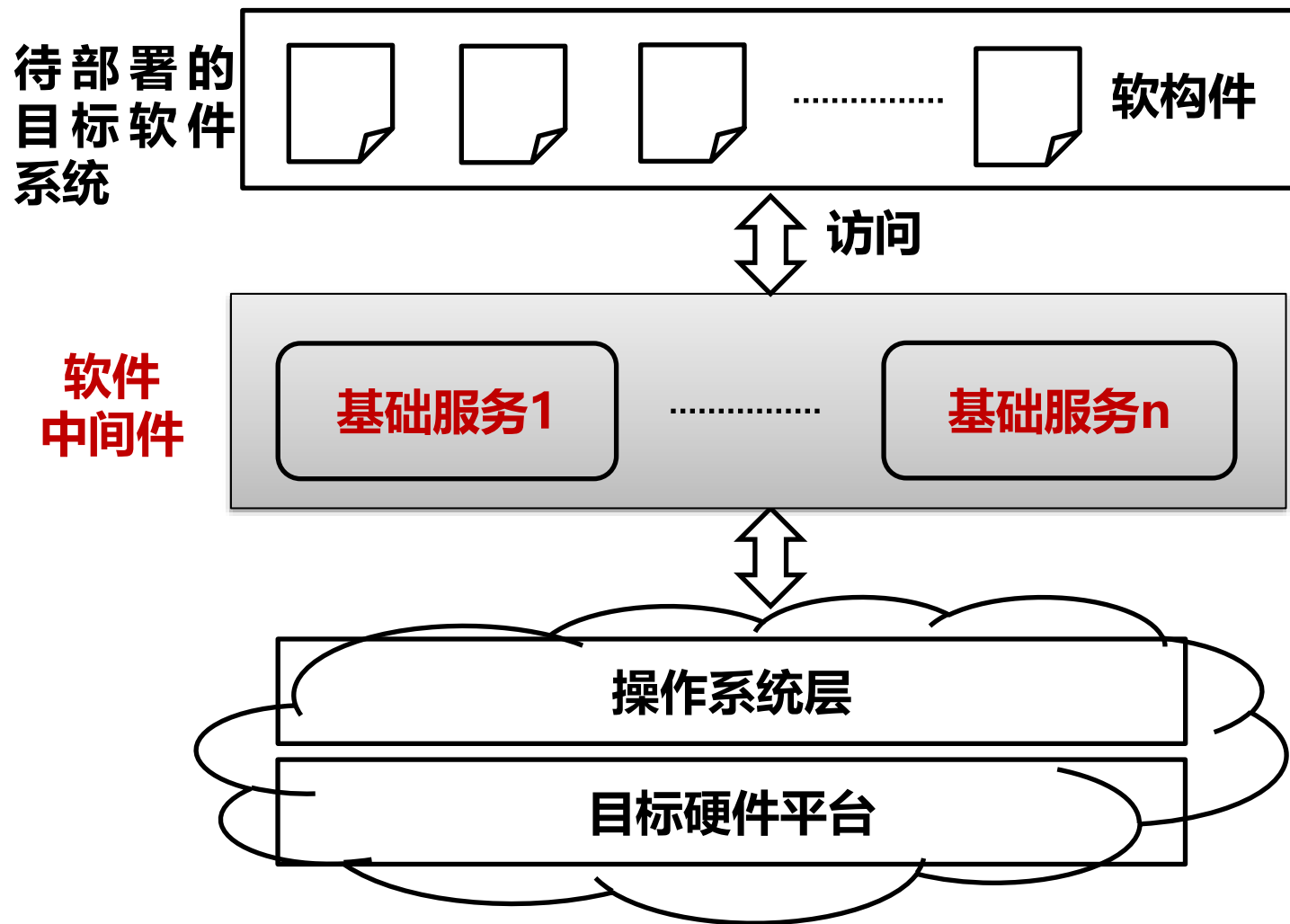
- ✓ **移动端APP软件部署**在移动端的操作系统之上，如智能手机、平板电脑等中的Android、IOS等
- ✓ **桌面端软件主要部署**在计算机上，常见的操作系统有Windows、Linux、MacOS等



3.3.2 基于软件开发框架的部署



3.3.3 基于软件中间件的部署





3.3.4 基于容器和镜像的部署

□ **容器**是一个视图隔离、资源可限制、具有独立文件系统的**进程集合**

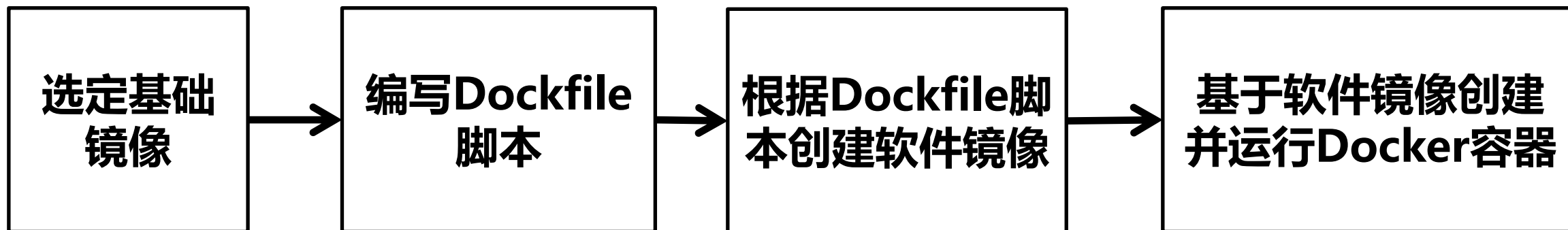
- ✓ 容器作为软件运行的上下文
- ✓ 容器具有独立文件系统，包括二进制文件、配置文件以及依赖

□ **容器运行时所需要的所有文件集合称之为容器镜像，又叫做rootfs**

- ✓ 对于容器镜像而言，它打包的不仅仅是应用代码，而且还包括应用运行所需要的所有依赖文件
- ✓ 容器镜像就是容器的文件系统，有了容器镜像，就可以构建该镜像的多个容器实例



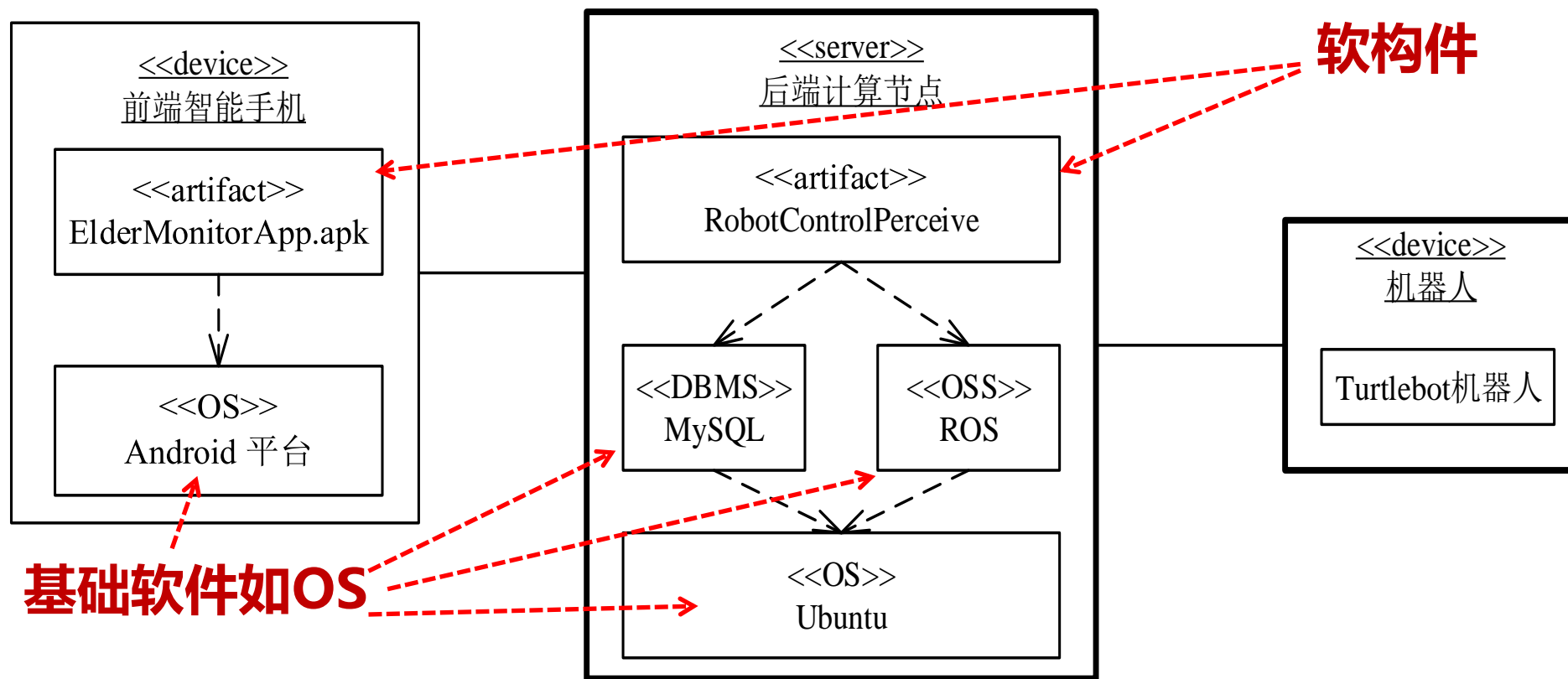
容器部署流程



3.4 基于部署图来搭建环境和部署软件

□设计阶段的部署图

✓计算节点，通信方式，节点的运行环境，所要部署的软构件





内容

1. 软件及其环境

- ✓ 软件运行环境及其变化

2. 何为软件部署

- ✓ 软件部署的概念和原则

3. 软件部署的方式和方法

- ✓ 单机和分布式部署、多种软件部署方法

4. 软件部署的**CASE工具**

- ✓ Fat jar、Installer Projects、Jenkins等





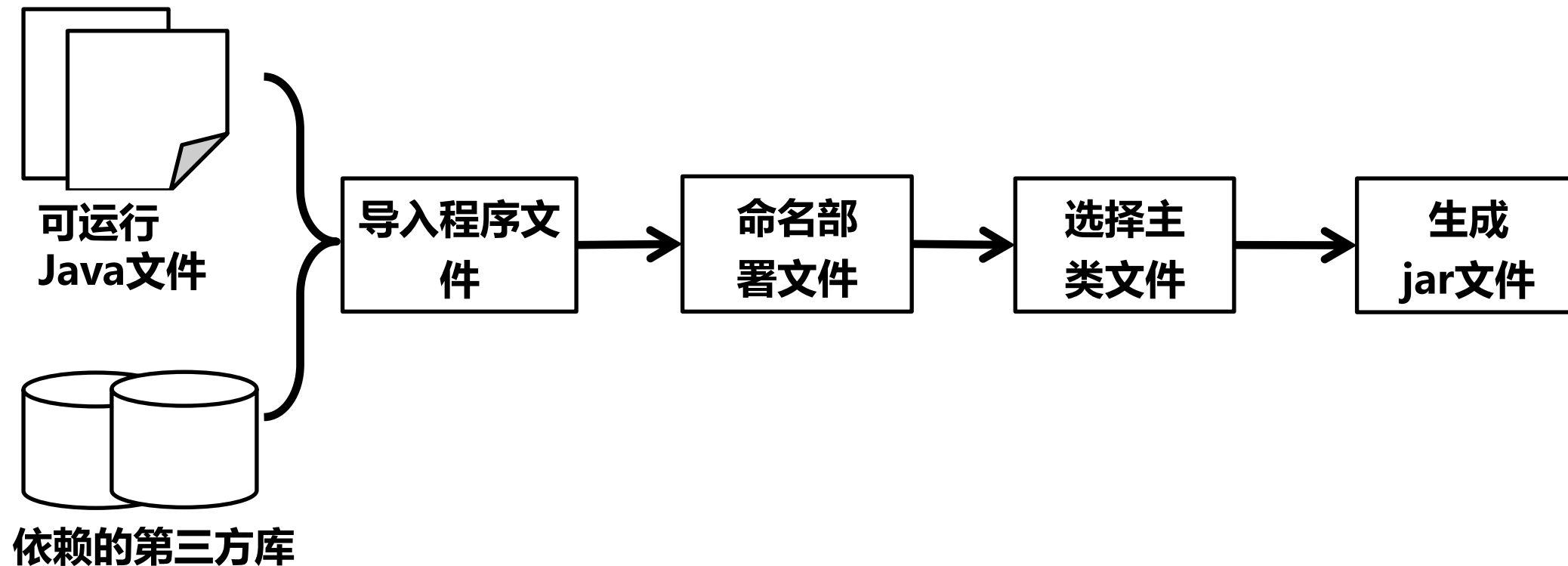
4.1 Fat jar部署工具

□支持Java程序代码的打包、安装和部署

- ✓将Java程序的所有资源（包括源代码、调用的第三方库、图片、项目配置文件等）进行打包，生成可直接安装部署的jar文件
- ✓能脱离Eclipse环境灵活部署到目标计算平台上
- ✓是一个开源软件，托管在SourceForge开源社区中
- ✓Eclipse集成开发环境的一个组件



Fat jar软件工具生成部署文件的过程





4.2 Installer Projects部署工具

□ 用于创建 **Microsoft Windows** 应用安装程序和安装包

- ✓ 设置软件图标和快捷方式、设置软件安装路径、增加或删除安装包中的文件等，采用图形化的方式对软件进行打包
- ✓ 作为Microsoft Visual Studio的一个扩展组件

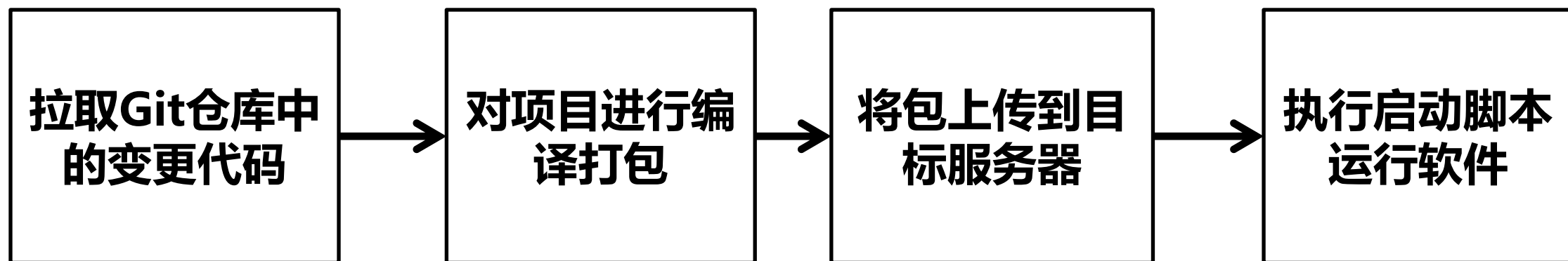
□ 基本步骤

- ✓ 将应用程序文件和图标导入到Microsoft Visual Studio项目；
- ✓ 在“解决方案”中使用“Setup Project”模板添加项目
- ✓ 设置程序作者信息、描述信息、版本号和默认安装路径等参数
- ✓ 将图标、程序依赖的动态链接库DLL文件、可执行程序及其快捷方式添加到“Application Folder”应用文件夹
- ✓ 设置应用图标和快捷方式
- ✓ 执行打包操作，分别生成以.msi和.exe为后缀名的安装包

4.3 Jenkins部署工具

□ 针对Java应用、实现持续集成和部署的软件工具

- ✓ 它允许其它的工具以插件的方式加入到Jenkins软件之中，为自动化的构建和部署提供多样化的服务





小结

□软件需要部署和运行在特定的环境中

- ✓环境包括硬件、基础软件、中间件和其它软件系统等

□软件部署是指将目标软件系统进行收集、打包、安装、配置和发布到运行环境的过程

- ✓单机部署和分布式部署

□软件部署的方法多样化，取决于软件的运行环境

- ✓基于操作、基于中间件、基于开发框架、基于容器

□支持软件部署的CASE工具

- ✓针对不同的编程语言和环境，提供不同的支持