

# 项目解决方案的作用

课程主讲：常慧勇

# 项目组成和解决方案

- 项目

- ▮ 有广义的说法。比如你说我正在开发一个项目。我们通常把开发的一个应用项目，叫做一个解决方案。
- ▮ 狭义上的说法，就是指解决方案中的一个部分（模块）

- 解决方案

- ▮ 通常一个解决方案会包括若干狭义上的项目。
- ▮ 这些狭义上的项目，都是直接或间接有联系的。通过引用添加这些项目之间建立的关系。
- ▮ 好处：当我们运行程序的时候，解决方案中的所有关联着的项目都会统一编译。生成exe或dll文件。
- ▮ 生成和重新生成解决方案一般情况下没什么区别。都是把解决方案中所有的子项目重新编译一次生成新的文件
- ▮ 生成解决方案和Debug启动区别就是，Debug启动会把项目编译后启动起来。生成解决方案只是重新编译各个项目而已。
- ▮ 清理解决方案：就是我们在调试过程中，有时候因为错误或经常调试，导致生成的问题可能不会正常的替换。
- ▮ 这时候我们可以清理一下解决方案。如果自动清理失效了，我们也可以手动清理。



# Debug和Release有什么区别

课程主讲：常慧勇

# Debug和Release的对比

## ● 概念

- Debug：调试。也就是我们尝试着运行一下程序。在运行中，可以通过断点观察程序的执行过程。
- Release：发布。也就是我们的程序开发和调试都成功了。这个时候我们可以选择发布程序。

## ● 断点

- Debug：在任何情况下，都可以打断点观察变量。
- Release：在很多情况下，无法直接断点，并且观察数据可能存在不正确的情况。

## ● 调试

- F5：从开始到第一个断点停止，如果继续，会执行到下一个断点停止。如果下一个没有断点，则当前全部执行完毕。
- F11：单步运行，也就是一行代码一行代码的执行。

## ● 生成

- Debug：所生成的文件，包含很多调试信息，生成的文件比较大。
- Release：不包含调试信息，生成的文件较小。

## ● 优化与选择

- Debug：不做任何的优化代码任务。开发中经常使用Debug调试。
- Release：有时候需要做性能优化，以便提升运行速度。项目结束最后使用Release发布。



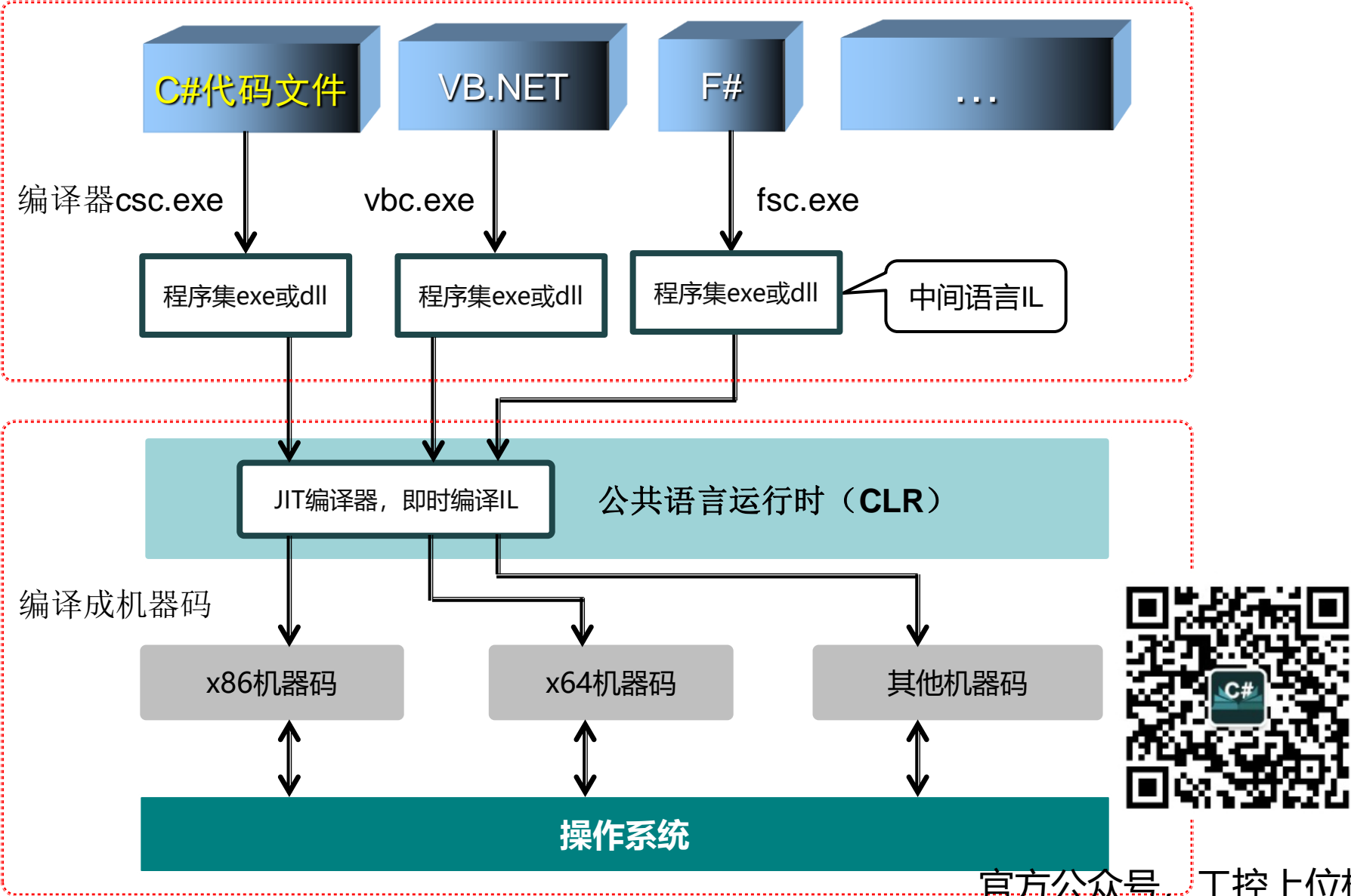
官方公众号，工控上位机学习专用

# 项目是如何生成和运行的

课程主讲：常慧勇



# 项目生成和运行过程分析



# 托管代码与CLR的理解

课程主讲：常慧勇

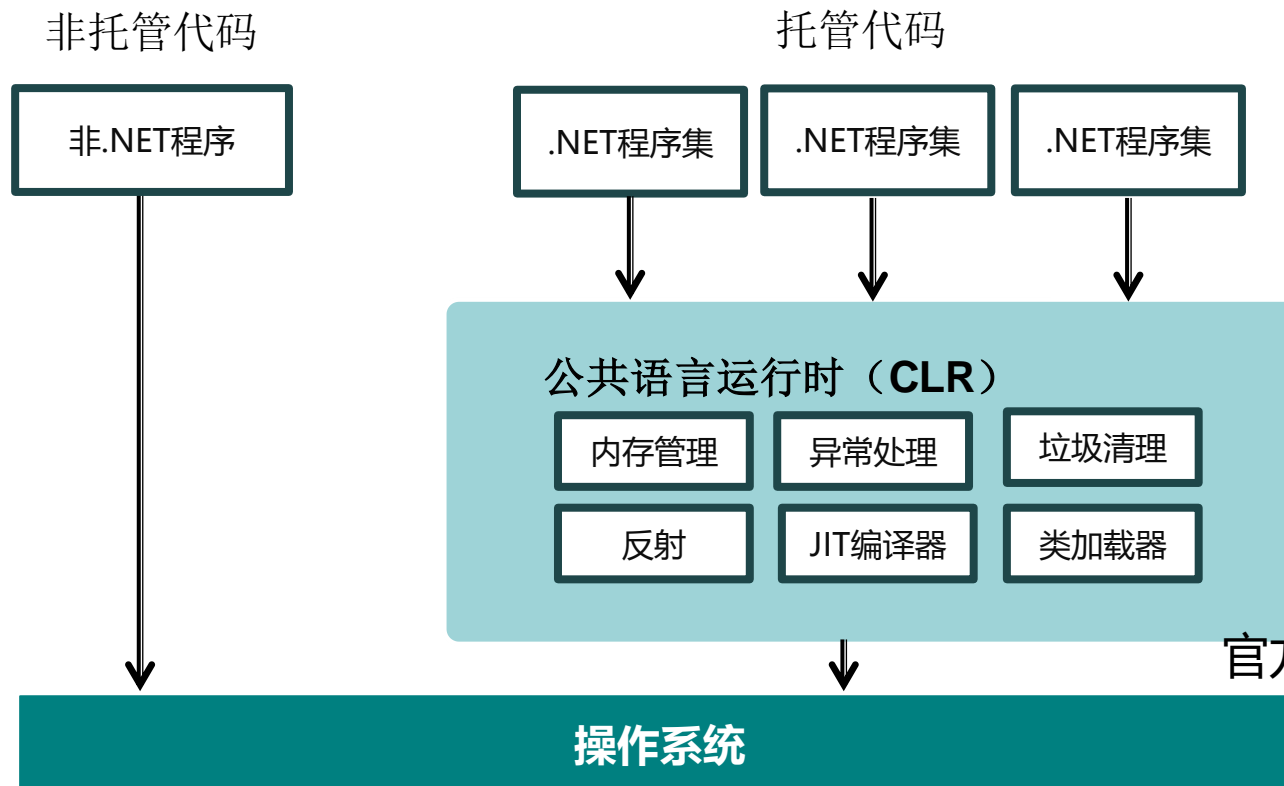
# 托管代码和非托管代码

- 概念区分

- ▮ 托管代码：在CLR管理之下运行的程序指令代码（MSIL指令）称为托管代码。
- ▮ 非托管代码：仅针对特定操作系统而生成的CPU机器指令，称为非托管代码。

- 公共语言运行库CLR（.NET中的虚拟机）

- ▮ CLS（Common Language Specification）公共语言规范。解决不同开发语言之间的语法规则。
- ▮ CTS（Common Type System）通用类型系统。解决不同开发语言之间的数据类型差异。



官方公众号，工控上位机学习专用



# .NET平台混合语言开发

- VB.NET和C#项目代码可以无缝集成
  - ▢ 两者都被编译成微软的中间代码（MSIL）



官方公众号，工控上位机学习专用