

CLI语言列表

CLI语言是计算机编程语言，用于产生符合公共语言基础结构（CLI）规范的库和程序。除了一些值得注意的例外，大多数CLI语言都完全编译为通用中间语言（CIL），该中间语言可以使用由.NET Framework，.NET Core和Mono实现的通用语言运行时来执行。其中一些语言还需要动态语言运行时（DLR）。

在执行程序时，会将CIL代码及时地编译（并缓存）为适合该程序在其上运行的体系结构的机器代码。在较早的阶段使用Microsoft的ngen.exe和Mono的“-aot”选项等“提前”编译器进行缓存，可以手动省略此步骤。

内容

著名的CLI语言

当前语言

废弃或弃用的语言

相关

另请参见

参考

著名的CLI语言

当前语言

- **C#**：使用最广泛的CLI语言^[1]，与Java有很强的相似性，并且与Object Pascal（Delphi）和C++相似。.NET Framework，.NET Core和Mono提供的实现。
- **C++ / CLI**：A版C++，包括扩展用于使用通用语言运行时（CLR）对象。仅由.NET Framework提供的实现。产生混合模式代码，该混合模式代码产生C++对象的本机代码。编译器由Microsoft提供。
- **ClojureCLR**：Clojure到CLI的端口，属于Clojure项目的一部分。^[2]
- **眼镜蛇（Cobra）**：一种具有静态和动态类型的CLI语言，可通过合同进行设计并内置单元测试。
- **组件Pascal**：一种符合CLI的Oberon方言。它是Pascal和Modula-2的传统中的一种强类型语言，但具有强大的面向对象扩展。
- **埃菲尔（Eiffel）**：纯粹面向对象的语言，专注于软件质量，包括按合同和多重继承的集成设计。符合CLI。
- **F#**：一种多范式CLI语言，支持功能编程和命令式面向对象编程学科。ML的变体，与OCaml很大程度上兼容。.NET Framework，.NET Core和Mono提供的实现。
- **F***-基于F#的依存类型语言。
- **Fantom**-一种编译为.NET和JVM (<https://asna.com/us/products/visual-rpg>)的语言 (<https://asna.com/us/products/visual-rpg>)
- **IronPython**：Python的开源CLI实现，基于动态语言运行时（DLR）构建。
- **IronScheme**-基于DLR的R6RS兼容方案实施

- **JScript .NET**: ECMAScript版本3的CLI实现, 与 JScript兼容。包含静态类型的扩展名。不推荐使用 Managed JScript。
- **Limnor Studio**: 是一种通用的无代码可视化编程系统。目的是使用户无需编写文本编程语言即可创建计算机 软件。软件开发人员可以扩展它。
- **托管JScript**: 基于动态语言运行时 (DLR) 构建的 JScript的CLI实现。符合 ECMAScript版本3。
- **Nemerle**: 类似于C # , OCaml和Lisp的多范式语言。
- **Oxygene**: 一种基于对象Pascal的CLI语言。
- **PeachPie**: PHP到.NET和.NET Core的编译器。法伦格的继任者。
- **Phalanger**: 一种 PHP的实现, 带有针对 ASP.NET的扩展。PeachPie的前身。
- **PowerBuilder**: 自11.1版起可以定位CLI。
- **小型基础 (Small Basic)** : Microsoft创建的一种BASIC编程语言, 用于教授编程。支持的发行版针对 .NET Framework 3.5和4.5版本。
- **Silverfrost FTN95**: Fortran 95的实现。
- **Synergy DBL .NET**: 由Synergex生成的, 面向对象CLI的DBL和 DIBOL兼容实现。 [3]
- **Team Developer**: 自Team Developer 6.0起使用的SQLWindows应用程序语言 (SAL) 。
- **Visual Basic .NET (VB.NET)** : Visual Basic的重新设计方言。 .NET Framework , .NET Core和 Mono提供的实现。
- **Visual COBOL**: Micro Focus生产的COBOL的增强版本, 已移植到 .NET Framework和 JVM 。 [4]
- **PowerShell**: 面向对象的命令行 shell。PowerShell可以动态加载以任何CLI语言编写的.NET程序集。与其他基于C的语言类似, PowerShell本身使用独特的脚本语法并使用花括号。
- **XSharp**: X # 是基于xBASE语言的Microsoft .NET的开源开发语言。它具有不同的风格, 例如 Core , Visual Objects , Vulcan.NET , xBASE ++ , Harbour , Foxpro等。

弃用或弃用的语言

- **A #** : Ada的 CLI实现。
- **Axum**: 一种演员模型 并发编程语言。
- **Boo**: 一种受 Python启发的静态类型的 CLI语言。
- **GrGen .NET** -一个CLI语言图形改写
- **IronRuby**: Ruby的开源CLI实现, 基于动态语言运行时 (DLR) 构建。
- **J #** : Java的 CLI兼容实现。编译器由Microsoft提供。J # 已停产。最新版本随Visual Studio 2005一起提供, 并一直支持到2015年。
- **C ++的托管扩展**: 面向公共语言运行时 (CLR) 的 C ++版本。不推荐使用 C ++ / CLI。
- **Niecza** - Perl 6的 CLI实现。

相关

- **IKVM.NET**: 一个 Java虚拟机可被用于运行的 Java等JVM语言 (例如, JVM像 Groovy中, 斯卡拉) 在CLI实施。

另请参阅

- JVM语言列表

- .NET Framework，即将被弃用的原始实现。
 - 公共语言基础结构 (CLI)，.NET框架的运行时环境（虚拟机组件）的开放规范
 - 通用中间语言 (CIL)，一种可以使用CLI实现执行的中间语言
 - 公共语言运行库，是Microsoft .NET Framework实现的虚拟机组件。
 - .NET标准库
 - 基类库 (BCL)
- Mono是.NET Framework的开源实现，已在包括Unity和Xamarin在内的许多地方使用。
- .NET Core，.NET Framework的开源跨平台继承者。

参考

1. Priyadarshini, Manisha (2018年6月25日)。 “2018年10种最流行的编程语言：学习编码” (<https://fossbytes.com/most-popular-programming-languages/>)。Fossbytes。检索2018年6月25日。
 2. “ClojureCLR” (<https://github.com/clojure/clojure-clr>)。GitHub.com。检索2012-06-28。
 3. “Visual Studio和.Net的另一种语言：Synergex发布Synergy / DE” (<http://blogs.msdn.com/b/terryclancy/archive/2010/12/07/another-language-for-visual-studio-and-net-synergex-release-synergy-de.aspx>)。msdn.com。
 4. 灵便的亚历克斯 (2015年8月20日)。 “COBOL来到Visual Studio 2015-SD Times” (<http://sdtimes.com/cobol-comes-to-visual-studio-2015/>)。SD时代。检索2017-04-04。
-

取自 “https://en.wikipedia.org/w/index.php?title=List_of_CLI_languages&oldid=942479379”

该页面的最后编辑于2020年2月24日 (UTC) 22:48。

文本可在知识共享署名-相同方式许可下获得；其他条款可能适用。使用本网站，您同意使用条款和隐私政策。Wikipedia®是非营利组织Wikimedia Foundation, Inc.的注册商标。