Atiitt 程序语言vm与rt 虚拟机与运行时

目录

[1. 运行时 虚拟机的一种，一般指进程级别的虚拟机。 1](#_Toc29241)

[1.1. 线程模型 1](#_Toc21488)

[1.2. 堆栈机vs 寄存器 1](#_Toc24259)

[1.3. 存储模型 2](#_Toc11028)

[1.4. Io交互使用宿主机的 2](#_Toc18867)

[2. Vm软件系统 2](#_Toc17742)

[2.1. Jvm指令系统 2](#_Toc25990)

[2.2. Os v线程模型 2](#_Toc16242)

[3. 常见的vm 2](#_Toc26338)

[3.1. V8 - 开源，由Google开发，用C ++编写 3](#_Toc25132)

[3.2. Rhin- 由Mozilla基金会开源，完全用Java开发 3](#_Toc10228)

[3.3. Jvm clr 3](#_Toc19029)

# 运行时 虚拟机的一种，一般指进程级别的虚拟机。

[编辑](https://baike.baidu.com/item/%E8%BF%90%E8%A1%8C%E6%97%B6/javascript:;) [讨论](https://baike.baidu.com/planet/talk?lemmaId=3335184" \t "https://baike.baidu.com/item/%E8%BF%90%E8%A1%8C%E6%97%B6/_blank)

本词条缺少****概述图****，补充相关内容使词条更完整，还能快速升级，赶紧来编辑吧！

虚拟机的一种，一般指进程级别的虚拟机。

运行时是指一个程序在运行（或者在被执行）的状态。也就是说，当你打开一个程序使它在电脑上运行的时候，那个程序就是处于运行时刻。在一些编程语言中，把某些可以重用的程序或者实例打包或者重建成为“运行库"。这些实例可以在它们运行的时候被连接或者被任何程序调用

## 线程模型

## 堆栈机vs 寄存器

于堆栈的(Stack-based ) 和基于寄存器(Register-based) 的虚.htm

## 存储模型

Jvm使用Gc Heap 和 metaspace方法区loadheap方法表

对象分为大型对象、小型对象两类。如果对象大于或等于 85,000 字节，将被视为大型对象，**大型对象**通常是字符串，数组。

加载 CLR 时，将分配两个初始堆栈段（一个用于小型对象，另一个用于大型对象），我将它们分别称为小型对象堆 (SOH) 和大型对象堆 (LOH)。

然后，通过将托管对象置于任一托管堆栈段上来满足分配请求。如果对象小于 85,000 字节，则将其放在 SOH 段上；否则将其放在 LOH 段上。随着分配到各段上的对象越来越多，会以较小块的形式提交这些段。【2】【4】

## Io交互使用宿主机的

Atitit vm mem java jvm net clr 虚拟机内存模型对比

# Vm软件系统

## Jvm指令系统

## Os v线程模型

# 常见的vm

那么除了v8引擎，你还知道那些js引擎

## V8 - 开源，由Google开发，用C ++编写

## Rhin- 由Mozilla基金会开源，完全用Java开发

## Jvm clr

# ref

## (9+条消息)计算机的基本组成和工作原理 - sunshine\_hsm的博客 - CSDN博客.html

## atitit 虚拟机之道vm之道 runtime设计 运行时 .attilax著.docx

## atitit 虚拟机之道vm之道 runtime设计 运行时 .attilax著.docx

1. Atitit 虚拟机的层次 架构与常见的虚拟机 3

1.1. Os隔离了硬件的区别 4

1.2. 4

1.3. Jvm clr又隔离了os的区别 4

1.4. Web 隔离了 vm的区别，webkit vm环境。。 4

1.5. 框架又是一个小的vm环境，ioc容器。。 4

1.6. 4

1.7. Dockert容器 4

1.8. 4

1.9. 安卓虚拟机。。等，可以集成app 4

1.10. Game虚拟机。。 4

2. 4

3. 4

4. Vm的未来 4

4.1. 1.5.1　模块化 1.5.2　混合语言 1.5.3　多核并行 1.5.4　进一步丰富语法 1.5.5　64位虚拟机 5

5. 自动内存管理机制 5

6. 执行引擎工作原理：方法调用 25 6

7. 常量池解析 206 6

8. 类变量解析 280 8

9. Java栈帧 344 8

10. 类方法解析 440 9

11. 执行引擎 513 10

12. 类的生命周期 602 11

13. 垃圾收集器与内存分配策略 13

14. 第4章　虚拟机性能监控与故障处理工具 14

15. 第6章　类文件结构 15

16. 虚拟机类加载机制 16

17. 第8章　虚拟机字节码执行引擎 16

18. 第四部分　程序编译与代码优化 第10章　早期（编译期）优化 17

19. 晚期（运行期）优化 17

20. 第五部分　高效并发 第12章　Java内存模型与线程 18

21. 第13章　线程安全与锁优化 18

22. 两种计算模型 ，堆栈机和状态机（基于寄存器的虚拟机 19

22.1.1. 堆栈机 19

22.1.2. 状态机 20

23. 为什么状态机比堆栈机快呢？ 21

23.1. Stack based vm的指令 范例 23

24. 参考 23

25. 附录B　虚拟机字节码指令表 24

25.1. Mov系列指令 24

25.2. Push pop 24

25.3. 24

25.4. 算数逻辑移位 24

25.5. 24

25.6. 跳转指令 24

25.7. Oo指令 24

26. 附录C　HotSpot虚拟机主要参数表 24

27. 附录D　对象查询语言（OQL）简介 24

28. 参考资料 25

28.1. Atitit.php opcode虚拟机指令集 分类以及详细解释 25

28.2. 25

28.3. C:\Users\Administrator\Desktop\vm虚拟机设计之道资料包>dir /b 25

28.4. Atitit .jvm 虚拟机指令详细解释.docx 25

28.5. Atitit .jvm 虚拟机指令详细解释.docx.txt 25

28.6. Atitit runtime设计 运行时 vm设计 虚拟机设计.docx 25

28.7. Atitit runtime设计 运行时 vm设计 虚拟机设计.docx.E5A681566CA1E2DE28F6B1BB7072448A.20180208225522212.wps 25

28.8. Atitit vm虚拟机原理与概论book.docx 25

28.9. atitit 虚拟机之道vm之道.attilax著.docx 25

28.10. Atitit 虚拟机的层次 架构与常见的虚拟机.docx 25

28.11. Atitit.java 虚拟机 指令集合.docx 25

28.12. Atitit.java 虚拟机 指令集合.docx.txt 25

28.13. Atitit.php opcode虚拟机指令集 分类以及详细解释.docx 26

28.14. atitit. atitit.基于虚拟机的启动器设计 v2 q37 --java 启动器 java生成exe.docx 26

28.15. Atitit.基于寄存器的虚拟机设计.docx 26

28.16. Atitit.简化的基于堆栈的虚拟机指令 参考java jvm虚拟机指令.docx 26

28.17. Atitit.虚拟机与指令系统的设计.docx 26

28.18. Atitit。虚拟机指令集 的文本形式语法 二进制形式的规范.docx 26

28.19. Atitti.java android反编译解决方案-----虚拟机方案.docx 26

28.20. 《揭秘Java虚拟机：JVM设计原理与实现》(封亚飞)【简介\_书评\_在线阅读】 - 当当图书.mhtml 26