

超算极简入门

仲殷旻

zhongyinmin@pku.edu.cn



目录

- 理论知识基础篇
- 动手实践基础篇
 - 服务器的登陆 (ssh, public/private key)
 - 命令行环境 (shell, zsh, 环境变量)
 - 文件的传输 (scp)
 - 文件的编辑 (vim)

什么是超算？

Wiki百科：超级计算机（英语：Supercomputer），指能够执行一般个人电脑无法处理的高速运算的计算机，规格与性能比个人计算机强大许多。



服务器 (node)



超算集群 (cluster)

天河二号

- 16000个节点组成
- 每个节点有2颗Xeon E5 2692处理器和3个Xeon Phi
- Xeon Phi最多72个处理器核心，每个核心拥有4个超线程，
最多288个线程
- 总计有312万个计算核心

超算带来了什么？

- 无与伦比的并行处理能力

- 海量的计算以及存储资源

随之而来的问题？

- 涉及到计算机系统的几乎一切

软



Pthread
Pytorch/Tensorflow
Cuda
Openmp
Openmpi

上层编程抽象

编译器

GCC

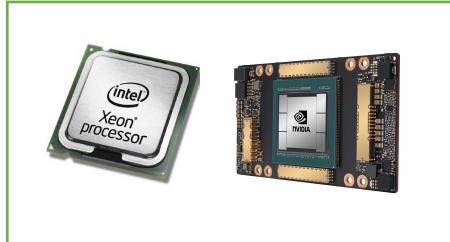
LLVM
COMPILER INFRASTRUCTURE



CLANG

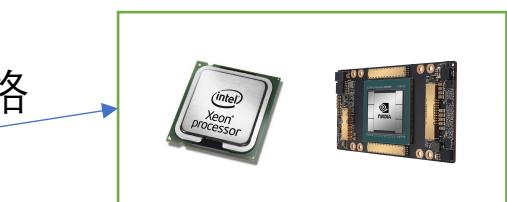
硬

体系结构



集中式

网络



分布式

分布式系统

Cluster

从何学起？



<https://csdiy.wiki/>

- 体系结构 >
- 操作系统 >
- 并行与分布式系统 >
- 计算机系统安全 >
- 计算机网络 >
- 数据库系统 >
- 编译原理 >

都学完才算准备好吗？

在动手实践中学以致用，加深理解

你想象当中的超算：



你现实当中的超算：



目录

- 理论知识基础篇

- 动手实践基础篇

- 服务器的登陆 (ssh, public/private key)
- 命令行环境 (shell, zsh, 环境变量)
- 文件的传输 (scp, vscode)
- 文件的编辑 (vim, vscode)



此部分所有内容均在[MIT Missing Semester](#)中有所覆盖

如何连接服务器 Secure Shell (ssh)

太长了，记不住？

登陆：

ssh -p xxx username@hostname

e.g., ssh flyingpig@scc.pkusc.org

e.g., ssh -p 2222 flyingpig@103.79.27.154



配置文件：~/.ssh/config

再根据提示输入密码，你将远程连接到目标机器的shell

```
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.4.0-91-generic x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

Welcome to Alibaba Cloud Elastic Compute Service !

Last login: Fri Jan  7 16:26:55 2022 from 124.205.77.106
root@iZ2ze2xy1cd004ixin6kd7Z:~# ls
root@iZ2ze2xy1cd004ixin6kd7Z:~#
```

Host scc

HostName scc.pkusc.org

User flyingpig

Host sccc

HostName 103.79.27.154

Port 2222

User flyingpig

e.g., ssh scc

e.g., ssh sccc

如何连接服务器 Secure Shell (ssh)

每次登陆都需要输入密码 → 记不住 → 123456

public-key cryptography



ssh-keygen -f ~/.ssh/flyingpig_id1

~/.ssh/flyingpig_id1

~/.ssh/flyingpig_id1.pub

Copy into

~/ssh/authorized_keys

Option 1:
ssh-copy-id -i ~/.ssh/flyingpig_id1.pub flyingpig@scc.pkusc.org

Option 2:
cat ~/.ssh/flyingpig_id1.pub | ssh flyingpig@scc.pkusc.org 'cat >> ~/ssh/authorized_keys'

命令行环境 (shell)

- 基础命令 个人建议：别太纠结花里胡哨的配置，最好形成一套简洁方便的流程，能让自己在一台新机器上几分钟内就能搭建出一个自己习惯的命令行环境
- Shell脚本与环境变量

- Bourne shell(bash) vs Z shell (zsh)

zsh 是 bash的升级版，提供了很多友好的特性

同时zsh 拥有一个庞大的插件社区生态以及众多开箱即用的Framework

e.g., 个人喜欢的[oh-my-zsh](#)

文件的传输

- 单个小文件：

```
cat localfile | ssh remote_server tee serverfile
```

p.s. tee 会把STDIN读入的数据写进指定文件

- 大文件或者文件夹

```
scp path/to/local_file remote_host:path/to/remote_file
```

```
scp -r path/to/local_dir remote_host:path/to/remote_dir
```

- scp的上位替代[rsync](#)

自动检测远程和本地的同内容文件从而避免重复复制

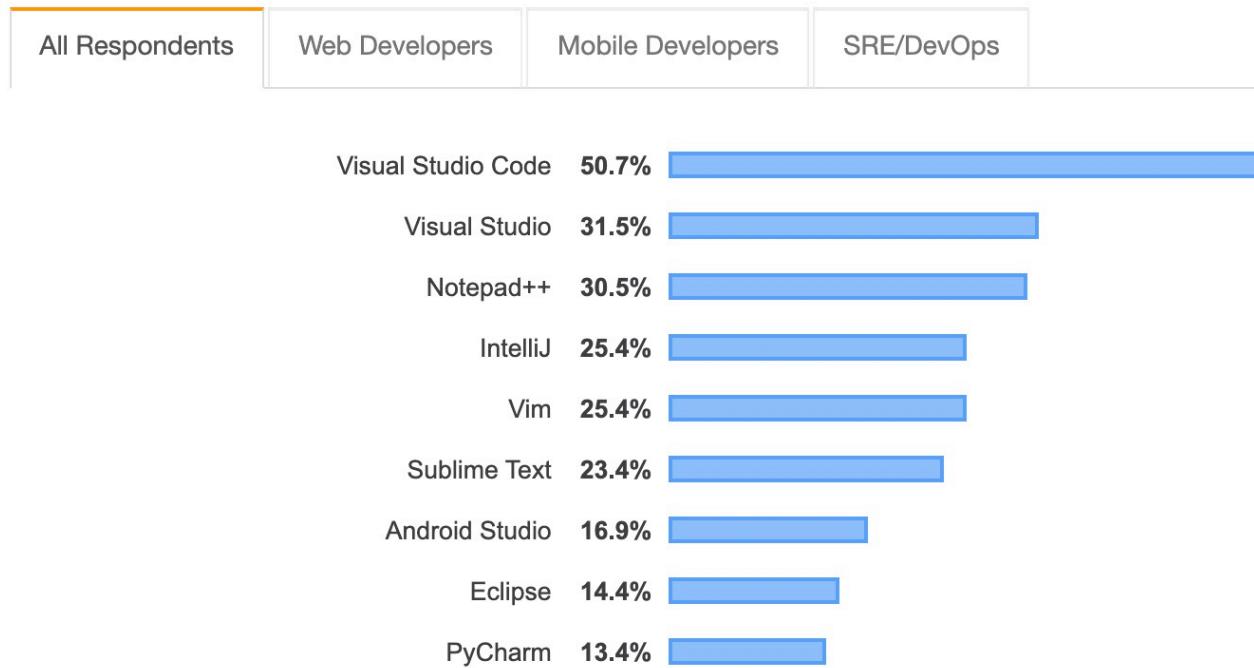
支持传输断线重联

Rsync

- 文件增量同步工具
- 以原生rsync传输协议或者透过远程shell如RSH或者SSH提供文件
- 服务端口：873
- rsync -[options] source destination
- 功能：
 - 增量同步
 - 保留MetaData
 - sync –av source destination

文件的编辑

Most Popular Development Environments



几乎所有图形界面文件编辑器
或IDE都支持Vim key-binding

听别人讲10个小时也没法学会vim，唯有practice, practice, practice

文件的编辑——Vim学习路线

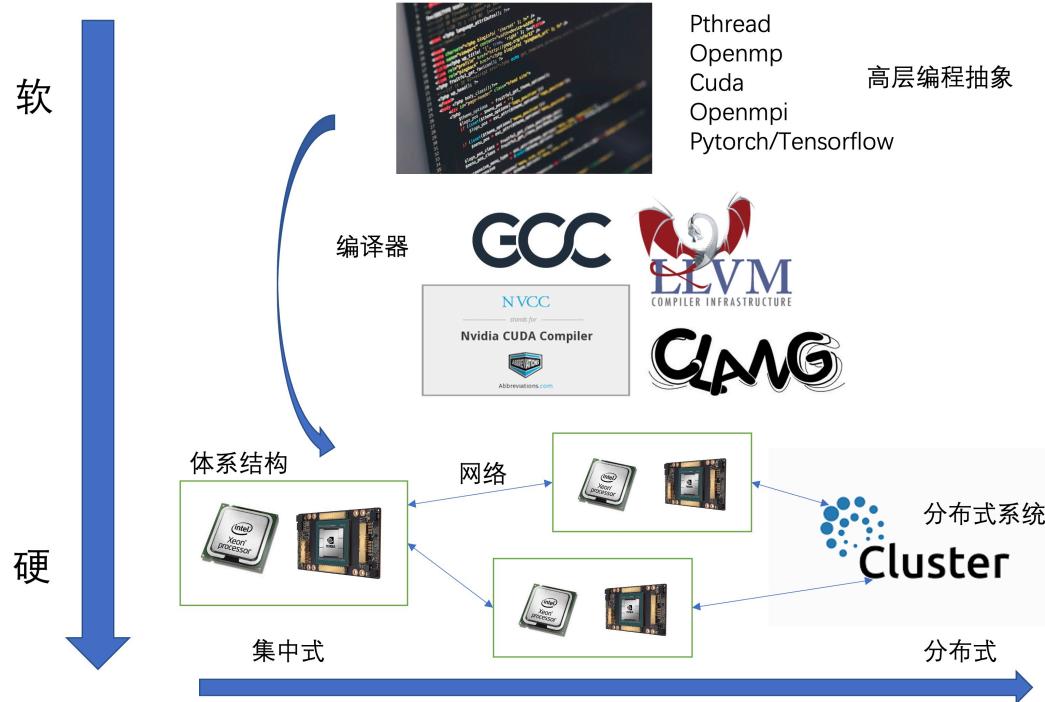
- 阅读这篇深入浅出的[Tutorial](#), 掌握vim的基本用法 (30 – 40 min)
- 用Vim自带的vimtutor进行练习, 安装完Vim之后直接在命令行里输入 vimtutor即可进入练习程序 (1hour)
- 强迫自己使用Vim进行开发, IDE里可以安装Vim插件 (2week)
- 花里胡哨的大门就此向你打开, 你可以按需配置自己的Vim (修改.vimrc文件), 网上有数不胜数的资源可以借鉴 (50 – 60 years)

当然, 如果你需要在服务器上编辑一个大型工程

个人建议使用 vscode + ssh插件

总结

至此，你已经初步掌握了与服务器交互的基本操作
这些trivial的技能将带领你逐步进入计算机系统的美妙世界 ~



希望大家可以充分利用超算队的
资源，在实践中锻炼和巩固自己
对计算机系统各个方面的理解，
为中国的系统领域添砖加瓦！