|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | | |  |
| **鲲鹏基座软件** | | |  |
| **搭建指南** | | |
| **文档版本** | **01** | |
| **发布日期** | **2022-07-30** | |
| 华为网格系统---方案4-032.png | | | | |
|  | 华为技术有限公司 | | 附件1-16K |  |

|  |
| --- |
| 版权所有 © 华为技术有限公司2022。保留一切权利。  非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。  商标声明  附件3-版权声明页图和其他华为商标均为华为技术有限公司的商标。  本文档提及的其他所有商标或注册商标，由各自的所有人拥有。  注意  您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或默示的声明或保证。  由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。 |

|  |  |
| --- | --- |
| 华为技术有限公司 | |
| 地址： | 深圳市龙岗区坂田华为总部办公楼邮编：518129 |
| 网址： | <https://www.huawei.com> |
| 客户服务邮箱： | [support@huawei.com](mailto:support@huawei.com) |
| 客户服务电话： | 4008302118 |

目录

[1 介绍 1](#_Toc109230819)

[2 环境要求 2](#_Toc109230820)

[3 移植规划数据 4](#_Toc109230821)

[4 安装毕昇编译器 5](#_Toc109230822)

[4.1 yum源安装方式 6](#_Toc109230823)

[4.2 软件包安装方式 6](#_Toc109230824)

[5 安装Hyper-MPI 9](#_Toc109230825)

[5.1 毕昇编译hucx软件包 10](#_Toc109230826)

[5.2 毕昇编译Hyper-MPI软件包 11](#_Toc109230827)

[6 修订记录 13](#_Toc109230828)

# 介绍

鲲鹏HPC基座软件包括毕昇编译器、Hyper-MPI高性能通信库等，其以鲲鹏服务器为硬件基础，结合芯片特点，构建了各场景解决方案计算底层生态。

以Hyper-MPI为例，目前业界领先的MPI库，均采用闭源发布策略，且不支持鲲鹏生态；Hyper MPI则不仅可以在x86服务器上使能，而且可以在以鲲鹏芯片为核心的服务器和集群上发挥作用。且相比于x86服务器，鲲鹏服务器具有单节点核数多、指令集简单、耗能低、价格实惠等特点，因此这有利于构建以鲲鹏芯片为硬件核心的计算生态。

关于鲲鹏生态的更多信息请访问鲲鹏官网。

# 环境要求

硬件要求

硬件要求如表2-1所示。

硬件要求

| 项目 | 说明 |
| --- | --- |
| CPU | Kunpeng 920 |

软件要求

软件要求如表2-2所示。

软件要求

| 项目 | 版本 | 下载地址 |
| --- | --- | --- |
| 毕昇编译器 | 2.1.0 | https://www.hikunpeng.com/zh/developer/devkit/compiler/bisheng |
| Hyper-MPI | 1.1.1 | <https://github.com/kunpengcompute/hmpi/archive/refs/tags/v1.1.1-huawei.tar.gz> |
| HUCX | 1.1.1 | <https://github.com/kunpengcompute/hucx/archive/refs/tags/v1.1.1-huawei.tar.gz> |
| XUCG | 1.1.1 | <https://github.com/kunpengcompute/xucg/archive/refs/tags/v1.1.1-huawei.tar.gz> |

操作系统要求

操作系统要求如表2-3所示。

操作系统要求

| 项目 | 版本 | 下载地址 |
| --- | --- | --- |
| openEuler | openEuler 20.03 SP3 | https://repo.openeuler.org/openEuler-20.03-LTS-SP3/ |
| Kernel | 4.19.90 | https://gitee.com/openeuler/kernel |

# 移植规划数据

本章节给出各基座软件在移植过程中涉及到的相关软件安装规划路径的用途及详细说明。

移植规划数据

| 序号 | 软件安装规划路径 | 用途 | 说明 |
| --- | --- | --- | --- |
| 1 | */usr/local/bisheng* | 毕昇编译器的安装规划路径。 | 这里的安装规划路径只是一个举例说明，建议部署在共享路径中。需要根据实际情况调整，**后续章节凡是遇到安装路径的命令，都以现网实际规划的安装路径为准进行替换，不再单独说明。** |
| 2 | */usr/local/hmpi* | Hyper-MPI的安装规划路径。 |

# 安装毕昇编译器

本节介绍毕昇编译器的安装步骤，以下操作均使用root用户执行。毕昇编译器已经融入openEuler源，在openEuler操作系统中，可以使用yum源方式安装毕昇编译器；在非openEuler操作系统中，可以通过软件包方式安装毕昇编译器。

环境依赖

* 内存：8GB以上
* 操作系统：openEuler21.03、openEuler 20.03 (LTS)、CentOS 7.6、Ubuntu 18.04、Ubuntu 20、麒麟V10、UOS 20
* 架构：AArch64
* GCC版本：4.8.5以上
* glibc版本：2.17以上
* libatomic版本：1.2及以上

## yum源安装方式

**步骤1**在/etc/yum.repos.d/目录下增加配置文件bisheng-compiler.repo，运行如下命令：

**cat> /etc/yum.repos.d/bisheng-compiler.repo<< EOF**

**[bisheng-compiler]**

**name=bisheng-compiler**

**baseurl=https://repo.oepkgs.net/bisheng/aarch64/**

**enabled=1**

**gpgcheck=0**

**priority=100**

**EOF**

**步骤2**从yum源下载和安装毕昇编译器rpm包

**yum update**

**yum install bisheng-compiler**

**步骤3**（可选）清空当前窗口的hash表

如果系统中有其他版本的 LLVM 编译器，请在安装毕昇编译器之后立即运行如下命令

**hash -r**

防止clang命令被hash捕获，出现毕昇编译器或开源LLVM编译器无法使用的问题。

**步骤4**验证安装是否成功

安装完毕后执行如下命令验证毕昇编译器版本：

**clang -v**

若返回结果已包含bisheng compiler版本信息，说明安装成功。

## 软件包安装方式

**步骤1**准备工作

在毕昇编译器产品页选择“毕昇编译器软件包下载”获取毕昇编译器软件包。且已上传至服务器上。

https://www.hikunpeng.com/zh/developer/devkit/compiler/bisheng

**步骤2**设置安装目录

1. 创建毕昇编译器安装目录

**mkdir -p /usr/local/bisheng**

注：**/usr/local/***为举例说明路径，请根据用例存放实际路径修改。*

2. 将毕昇编译器压缩包拷贝到安装目录下，并解压：

**cp bisheng-compiler-2.1.0-aarch64-linux.tar.gz /usr/local/bisheng**

**cd /usr/local/bisheng**

**tar -zxvf bisheng-compiler-2.1.0-aarch64-linux.tar.gz**

解压完成后在当前目录下出现名为**bisheng-compiler-2.1.0-aarch64-linux**的目录。

**步骤3**配置毕昇编译器的环境变量

* 安装**environment-modules工具**

**yum install environment-modules -y**

**source /etc/profile**

* 执行以下命令，创建环境变量配置文件

vi /**usr/local/bisheng/bisheng\_modulefiles**

**新增如下内容：**

**#%Module1.0**

**conflict bisheng**

**prepend-path PATH /usr/local/bisheng/bisheng-compiler-2.1.0-aarch64-linux/bin**

**prepend-path LD\_LIBRARY\_PATH /usr/local/bisheng/bisheng-compiler-2.1.0-aarch64-linux/lib**

* **执行以下命令，在当前shell中加载环境变量**

**module use** /**usr/local/bisheng/**

**module load** /**usr/local/bisheng/bisheng\_modulefiles**

* **若要避免每打开一个shell，导入一次变量。可写入到系统配置文件中。**

**vi /etc/profile**

**新增如下内容：**

**module use** /**usr/local/bisheng/**

**module load** /**usr/local/bisheng/bisheng\_modulefiles**

**步骤4**（可选）清空当前窗口的hash表

如果系统中有其他版本的 LLVM 编译器，请在安装毕昇编译器之后立即运行如下命令

**hash -r**

防止clang命令被hash捕获，出现毕昇编译器或开源LLVM编译器无法使用的问题。

**步骤5**验证安装是否成功

安装完毕后执行如下命令验证毕昇编译器版本：

**clang -v**

若返回结果已包含bisheng compiler版本信息，说明安装成功。

# 安装Hyper-MPI

Hyper MPI包含Hyper MPI和HUCX两个软件层，其中Hyper MPI的安装依赖于 HUCX，编译时应先编译HUCX。

前提条件

已完成以下依赖包的安装。

* **perl-Data-Dumper**
* **autoconf**
* **automake**
* **libtool 2.4.2**
* **毕昇编译器2.1.0**
* **numactl**
* **binutils**
* **systemd-devel**
* **valgrind**
* **flex**

## 毕昇编译hucx软件包

**步骤1** 下载HUCX源码包“hucx-1.1.1-huawei.tar.gz”。

下载地址：<https://github.com/kunpengcompute/hucx/archive/refs/tags/v1.1.1-huawei.tar.gz>

**步骤2** 下载XUCG源码包“xucg-1.1.1-huawei.tar.gz”。

下载地址：<https://github.com/kunpengcompute/xucg/archive/refs/tags/v1.1.1-huawei.tar.gz>

**步骤3** 将上述两个源码包上传至服务器HUCX源码包安装目录，如“/path/to/package/hucx”，执行以下命令，解压上述两个源码包。

**tar -zxvf hucx-1.1.1-huawei.tar.gz**

**tar -zxvf xucg-1.1.1-huawei.tar.gz**

**步骤4**执行以下命令，将XUCG源码包中的内容复制到HUCX源码包中的“src/ucg”目录下。

**cp -r xucg-1.1.1-huawei/\* hucx-1.1.1-huawei/src/ucg**

**步骤5**执行以下命令，进入“autogen.sh”脚本所在目录，并初始化git仓库。

**cd hucx-1.1.1-huawei/ && git init**

**步骤6**执行以下命令，进行源码包安装。

**./autogen.sh**

**./contrib/configure-opt -prefix=/usr/local/hmpi/hucx CC=clang CXX=clang++ FC=flang**

注：*/usr/local/为举例说明路径，请根据用例存放实际路径修改。*

注意

* 环境上若没有安装numactl插件，则需要执行以下命令，指定--disable-numa参数：
* **./contrib/configure-opt -prefix=/usr/local/hmpi/hucx --disable-numa CC=clang CXX=clang++ FC=flang**

**for file in `find . -name Makefile`;do sed -i "s/-Werror//g" $file;done**

**for file in `find . -name Makefile`;do sed -i "s/-implicit-function-declaration//g" $file;done**

**make -j$(nproc)**

**make -j$(nproc) install**

**步骤7**配置环境变量

**export PATH=/usr/local/hmpi/hucx/bin:$PATH**

**export LD\_LIBRARY\_PATH=/usr/local/hmpi/hucx/lib:$LD\_LIBRARY\_PATH**

**export INCLUDE=/usr/local/hmpi/hucx/include:$INCLUDE**

## 毕昇编译Hyper-MPI软件包

**步骤1**下载Hyper MPI源码包“hmpi-1.1.1-huawei.tar.gz”。

下载地址：https://github.com/kunpengcompute/hmpi/archive/refs/tags/v1.1.1-huawei.tar.gz

**步骤2**将源码包上传至服务器Hyper MPI源码包安装目录，如“/path/to/package/hmpi”；执行以下命令，解压软件包。

**tar -zxvf hmpi-1.1.1-huawei.tar.gz**

**步骤3**执行以下命令，进入“autogen.pl”脚本所在目录。

**cd hmpi-1.1.1-huawei/**

**步骤4**执行以下命令，进行Hyper MPI源码包安装。

**./autogen.pl CC=clang CXX=clang++ FC=flang**

**./configure -prefix=/usr/local/hmpi --with-platform=contrib/platform/mellanox/optimized --enable-mpi1-compatibility -with-ucx=/usr/local/hmpi/hucx CC=clang CXX=clang++ FC=flang**

注：**/usr/local/***为举例说明路径，请根据用例存放实际路径修改。*

**make -j$(nproc)**

**make -j$(nproc) install**

**步骤5配置环境变量**

* **执行以下命令，创建环境变量配置文件**

**vi /usr/local/hmpi/hmpi\_modulefiles**

**新增如下内容：**

**#%Module1.0**

**conflict hmpi**

**set OPAL\_PREFIX /usr/local/hmpi**

**setenv OPAL\_PREFIX /usr/local/hmpi**

**prepend-path PATH $OPAL\_PREFIX/bin:$OPAL\_PREFIX/hucx/bin**

**prepend-path LD\_LIBRARY\_PATH $OPAL\_PREFIX/lib:$OPAL\_PREFIX/hucx/lib**

**prepend-path INCLUDE $OPAL\_PREFIX/include:$OPAL\_PREFIX/hucx/include**

* **执行以下命令，在当前shell中加载环境变量**

**module use /usr/local/hmpi/**

**module load /usr/local/hmpi/hmpi\_modulefiles**

* **若要避免每打开一个shell，导入一次变量。可写入到系统配置文件中。**

**vi /etc/profile**

**新增如下内容：**

**module use /usr/local/hmpi/**

**module load /usr/local/hmpi/hmpi\_modulefiles**

**步骤6 安装后检查：**执行以下命令，进行安装后检查。

**which mpirun**

若返回如下安装路径示例，则证明安装成功。

**/usr/local/hmpi/bin/mpirun**

**----结束**

# 修订记录

| 发布日期 | 修订记录 |
| --- | --- |
| 2022-07-30 | 第一次正式发布。 |