|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | | |  |
| platanus  1.2.4 | | |  |
| **移植指南** | | |
| **文档版本** | **01** | |
| **发布日期** | **2022-3-29** | |
|  | | | | |
|  | 华为技术有限公司 | |  |  |

|  |
| --- |
| 版权所有 © 华为技术有限公司2020。 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  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://e.huawei.com> |

目 录

[1 介绍 1](#_Toc14670)

[2 环境要求 2](#_Toc3912)

[3 移植规划数据 4](#_Toc9208)

[4 配置编译环境 5](#_Toc31422)

[5 获取源码 6](#_Toc28124)

[6 编译和安装 7](#_Toc19550)

[7 运行验证 9](#_Toc28345)

[8 修订记录 11](#_Toc23991)

# 介绍

Platanus是一款用来组装核苷酸序列的应用。

关于Platanus的更多信息请访问官网：<http://platanus.bio.titech.ac.jp/>。

语言：C++

一句话描述：基因组装。

建议的版本

建议使用版本为“Platanus 1.2.4”。

# 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://mirrors.huaweicloud.com/kunpeng/archive/compiler/bisheng_compiler/bisheng-compiler-2.1.0-aarch64-linux.tar.gz> |
| Platanus | 1.2.4 | <http://platanus.bio.titech.ac.jp/?ddownload=150> |
| 算例 | - | <http://platanus.bio.titech.ac.jp/?ddownload=325> |

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 |
| --- | --- | --- |
| kylin | V10 |  |

# 移植规划数据

本章节给出Platanus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 | - | 基础环境搭建中的各安装包安装路径。 | 参考《[HPC解决方案 基础环境搭建指南](https://support.huaweicloud.com/instg-kunpenghpcs/kunpenghpcs_instg_0001.html" \o " )》中“安装规划数据”章节。 |
| 2 | /path/to/PLATANUS | Platanus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。现网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。 |
| 8 | /path/to/PLATANUS/CASE | 算例放置路径。 |

# 配置编译环境

前提条件

使用SFTP工具将各安装包上传至服务器对应目录下。

配置流程

配置流程

| 序号 | 配置项 | 说明 |
| --- | --- | --- |
| 1 | 基础环境搭建 | 参考[《HPC解决方案 基础环境搭建指导书》](https://support.huaweicloud.com/instg-kunpenghpcs/kunpenghpcs_instg_0001.html" \o " )中“单机场景环境搭建”章节。 |
| 2 | 安装毕昇编译器 | 参考《[安装毕昇编译器](https://support.huaweicloud.com/ug-bisheng-kunpengdevps/kunpengbisheng_06_0005.html)》 |

# 获取源码

操作步骤

下载Platanus安装包“Platanus\_v1.2.4.tar.gz”。

下载地址：<http://platanus.bio.titech.ac.jp/?ddownload=150>

使用SFTP工具将Platanus安装包上传至服务器*“*/path/to/PLATANUS*”*目录。

----结束

# 编译和安装

操作步骤

执行以下命令，进入安装目录并解压软件包。

**cd** /path/to/PLATANUS **&& tar -zxf Platanus\_v1.2.4.tar.gz && cd Platanus\_v1.2.4**

修改Makefile文件第1行、第2行。

修改前：

1 CXX = g++

2 CXXFLAGS = -std=c++0x -O3 -funroll-loops -Wall -fopenmp -finline-limit-50000 -lm -Dnullptr=0

修改后：

1 CXX = clang++

2 CXXFLAGS = -std=c++0x -O3 -funroll-loops -Wall -fopenmp -lm -Dnullptr=0

修改counter.h文件第426行、第455行、第684行。

修改前：

426 countKmerOrWriteTemporary(loopFlag, key, tmpOccurrenceTable, unmappedFP, lock, kmer);

455 inline void Counter<KMER>::countKmerOrWriteTemporary(bool &loopFlag, const typename KMER::keyType &key, DoubleHash<typename KMER::keyType, unsigned short> tmpOccurrenceTable[], FILE \*unmappedFP, omp\_lock\_t lock[], const KMER &kmer, const unsigned iterateTimes=32)

684 inline void Counter<KMER>::countKmerOrWriteTemporary(bool &loopFlag, const typename KMER::keyType &key, FILE \*unmappedFP, omp\_lock\_t lock[], const KMER &kmer, const unsigned iterateTimes=32)

修改后：

426 countKmerOrWriteTemporary(loopFlag, key, tmpOccurrenceTable, unmappedFP, lock, kmer, 32);

455 inline void Counter<KMER>::countKmerOrWriteTemporary(bool &loopFlag, const typename KMER::keyType &key, DoubleHash<typename KMER::keyType, unsigned short> tmpOccurrenceTable[], FILE \*unmappedFP, omp\_lock\_t lock[], const KMER &kmer, const unsigned iterateTimes)

684 inline void Counter<KMER>::countKmerOrWriteTemporary(bool &loopFlag, const typename KMER::keyType &key, FILE \*unmappedFP, omp\_lock\_t lock[], const KMER &kmer, const unsigned iterateTimes)

修改graph.h文件第1764行。

修改前：

1764 void BruijnGraph<KMER>::cutBranchIterative(const unsigned long long numThread=1)

修改后：

1764 void BruijnGraph<KMER>::cutBranchIterative(const unsigned long long numThread)

修改scaffoldGraph.cpp文件第2280行。

修改前：

2280 if (abs(insertLength - averageInsSize) > tolerence) continue;

修改后：

2280 if (std::fabs(insertLength - averageInsSize) > tolerence) continue;

修改baseCommand.cpp文件第175行。

修改前：

175 for (;moveArgs != '\0'; ++moveArgs) {

修改后：

175 for (;\*moveArgs != '\0'; ++moveArgs) {

编译安装Platanus

**make**

----结束

# 运行验证

操作步骤

使用PuTTY工具，以root用户登录服务器。

执行以下命令，设置环境变量。

**export PATH=***/path/to/PLATANUS***/Platanus\_v1.2.4:$PATH**

将下载好的算例文件上传至“*/path/to/PLATANUS/CASE*”目录并解压。

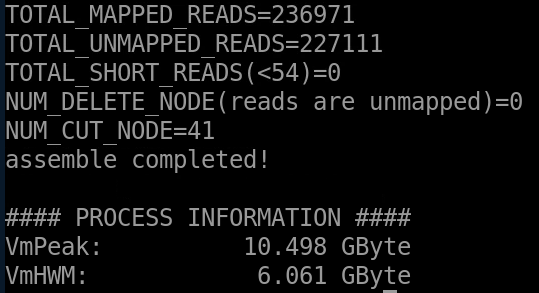
执行以下命令，运行算例。

**cd** */path/to/PLATANUS/CASE/***Platanus\_allee\_test\_data\_180619/test\_run**

**platanus assemble -t 64 -tmp ./ -m 8 -o out -f ../reads/Illumina\_PE500/R\*.fq 2>&1 |tee out.assemble.log**

结果样例如图7-1所示：

结果样例



----结束

# 修订记录

| 发布日期 | 修订记录 |
| --- | --- |
| 2022-3-29 | 第一次正式发布。 |