cb8e6bb6a0255fc8c5df554e1e1dcf8dc6111dd3

Gitlab qsq372233

qishanqing qsq372233

gitlab\_admin Gitlab\_Admin

[system@indemind.cn](mailto:system@indemind.cn)

sudo gpasswd -a indemind docker

newgrp docker

upload-to-ftp -d $file.zip ftp://www:0lHtrr@192.168.0.51/

upload-to-ftp -d daxiao.sh <ftp://ftpuser:ftpuser@indemind@192.168.50.191/>

sudo swapon /swapfile

sudo curlftpfs 192.168.50.191 /mnt/ftp/ -o user=ftpuser:ftpuser@indemind -o allow\_other

/etc/fuse.conf

sudo mkfs -t ext4 /dev/sdb

sudo umount -l ~/ftpdir

apt-get --download-only --reinstall install libc6=2.23-0ubuntu11.2

ll /var/cache/apt/archives/

strings /lib/aarch64-linux-gnu/libc.so.6 |grep GLIBC

java -jar agent.jar -jnlpUrl http://jenkins.indemind.com/computer/my/slave-agent.jnlp -secret 379a236ae1660366e95d86467e2a5b63ad181ce1bbc8758d2360a57caf392651 -workDir "/home/indemind/workspace/code/jenkins" &

docker run -d --name confluence -p 8090:8090 --restart always --link mysql:mysql --user root:root cptactionhank/atlassian-confluence

dokcer run -d --name mysql-jira -p 8720:3306 --restart always --privileged=true -v /docker/mysqletc/mysql:/etc/mysql -v /docker/mysqletc/data:/var/lib/mysql -e MYSQL\_USER="jira" -e MYSQL\_PASSWORD="jira" -e MYSQL\_ROOT\_PASSWORD="qsq" mysql:5.7

sudo docker run --detach --restart always --link mysql-jira:mysql --publish 8721:8080 cptactionhank/atlassian-jira-software

sudo docker run --detach --restart always --link mysql-jira:mysql --publish 8722:8080 mscharley/atlassian-jira-software

docker run --name nginx -d -p 80:80 --restart always -v /usr/share/nginx/html:/u sr/share/nginx/html:ro -v /etc/nginx/conf.d:/etc/nginx/conf.d:ro -v /var/log/nginx:/var/log/nginx -v /etc/nginx/nginx.conf:/etc/nginx/nginx.conf nginx

docker run --name nginx -d -p 80:80 --restart always nginx

docker run -dit --name arm --restart always -v /root/code:/root/code --privileged=true arm:2.0

docker run --name dns -d -p 53:53/udp -p 8081:8080 -v /docker/dns/dnsmasq.conf:/etc/dnsmasq.conf --log-opt "max-size=100m" -e "HTTP\_USER=admin" -e "HTTP\_PASS=admin" --restart always jpillora/dnsmasq

mou

docker run -itd -p 9090:8080 -p 50000:50000 --name jenkins --privileged=true  -v /home/jenkins:/var/jenkins\_home jenkins:latest

docker run --name bind -d --restart=always --publish 53:53/tcp --publish 53:53/udp --publish 10000:10000/tcp --volume /docker/dns/bind:/data sameersbn/bind:latest

sudo update-alternatives --config java

mvn install -Denforcer.skip=true

Xray

?useSSL=false

find . -name \*manager-pl\* |grep jira

sudo docker cp mysqld.cnf mysql:/etc/mysql/mysql/mysql.conf.d/mysqld.cnf

/usr/bin/db\_load -T -t hash -f /etc/vsftpd/virtual\_users.txt /etc/vsftpd/virtual\_users.db

sudo docker cp default.conf nginx:/etc/nginx/conf.d/

sudo docker cp atlassian-extras-3.1.2.jar 7509371edd48:/opt/atlassian/jira/atlassian-jira/WEB-INF/lib/

GRANT ALL PRIVILEGES ON \*.\* TO 'root'@'%' IDENTIFIED BY 'root' WITH GRANT OPTION;

show variables like 'char%';

#SET character\_set\_database='utf8';

#SET character\_set\_server='utf8

create database jira default character set utf8 collate utf8\_bin;

create database confluence character set utf8;

alter database confluence default collate utf8\_bin;

SET GLOBAL tx\_isolation='READ-COMMITTED';

flush privileges;

**修改/var/atlassian/application-data/confluence/confluence.cfg.xml**

**SELECT CONCAT('ALTER TABLE ',TABLE\_SCHEMA,'.', TABLE\_NAME, ' COLLATE=utf8\_bin;') 'SQL'**

**FROM information\_schema.TABLES**

**WHERE TABLE\_COLLATION <>'utf8\_bin' and TABLE\_SCHEMA ='jira';**

**SELECT CONCAT('ALTER TABLE ', TABLE\_SCHEMA,'.',TABLE\_NAME, ' MODIFY COLUMN ',COLUMN\_NAME,' ',COLUMN\_TYPE,' CHARACTER SET utf8 COLLATE utf8\_bin;') 'SQL'**

**FROM information\_schema.COLUMNS**

**WHERE DATA\_TYPE='varchar' AND COLLATION\_NAME <> 'utf8\_bin' and TABLE\_SCHEMA ='jira';**

java -jar atlassian-agent.jar -p jira -m aaa@bbb.com -n my\_name -o https://zhile.io -s ABCD-1234-EFGH-5678 # 生成的许可证，直接复制于jira页面提交即可

dpkg -i --force-overwrite

project = ABBY AND issuetype in (bug, 任务) AND resolution = Unresolved AND due <= "0" ORDER BY priority DESC, updated DESC

due <= "0" AND resolution = Unresolved

AAABKg0ODAoPeNptUMtqwzAQvOsrBL20BwVJxCYJGJrYPgT8onaaSy+q2LQCR44lOTR/XyVuaPo4L

Cw7O7Mze5d3GufihNkMc74I6ILOcJw0mFNOUQJWGnVwqtNR3OldO4CWgO9rMEcwDy8LnB5FO4jzA

ooNXJpEOIjOdEIDwubIE52QrhB7P54GPOSM8ce+n8huj6RXnXhUHSFyZoBxUDthHJhoJ1oLV4E0F

6r9q/Dt4EYgUxK0heZ0gMvduMzz9CleLzPUjtAzGHvmcORltQMtfLL046DM6SZASNgMleZNaGXHG

73t0Rh/nUSrpgrJZllXpNwWK0KbaovqtIh8kYwF03BOpxx9mfH72Tr5CV3MFsP+FUy521jvKSLsS

vjfTTUY+S4s/H7zJ2AokcIwLQIVAJNiMSGPODecXhyGK85de84WQiNwAhReyVkAgnpw4fDpkUuVJ

s1wmIMmHA==X02f3

atler database jira7 character set utf8 collate utf8\_bin Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'atler database jira7 character set utf8 collate utf8\_bin' at line 1 0,00068 sec

<http://192.168.50.191:8722/issues/?filter=10200>

AAABdA0ODAoPeNp9UUtvgkAQvvMrSHppD0tgRdOakNTCpqEVNaBNmvSyxVG3QcDZxdZ/3xU01fo4z

u58z7kZLyoz4hvTcU273bUfui1q+sHYpDa1jTkC5IuiLAGtvkghl8CmQoki99hgzOJRHCbMGFTLT

8DhbCIBpUccwy9yxVM14EvwqNumHeo49HG1stJiaXwJ5NYJZFRhuuASAq40RmsTu00c19ipjjcl1

HT+MIpY7Ie9/v6L/ZQCNwe4DnFaewss4iI79ZAArgHDwHuKn1/J/fuIEToZTQjrxVFjsMRiWqXK2

g5EFjP1zREszSjW4CmsoFm7XMqZ6s6F0P5yBTnP0wtBrrg5KXGno3P1wyBhA9J32q5737JdQ0/e8

csV4kRxVIDejGcSjCHOeS4krxOu5MrwEerh/62yRv5Nu9mu0qMOQMfEEoXc1ReATFGUNemLVjeTn

bp521zn7qNrsjXPqlqrsXup/3PNHoof4v44m/kXkSUDzTAsAhQCb5hpwzuLPAL/Xt7qoMuVRXU7r

wIUQaGaOaWlqRt9gamF8DPeOYcPtLI=X02i6

apt-get install libboost-atomic1.58-dev libboost1.58-dev

cd sensor/

Mkdir build

cmake ..

make

cd ../slam

Mkdir build

sudo apt-get install libglew-dev libboost-dev libboost-thread-dev libboost-filesystem-dev libboost-all-dev

安装pangolin

sudo apt-get install libx11-dev libxmu-dev libglu1-mesa-dev libgl2ps-dev libxi-dev g++ libzip-dev libpng12-dev libcurl4-gnutls-dev libfontconfig1-dev libsqlite3-dev libglew\*-dev libssl-dev

git clone https://github.com/stevenlovegrove/Pangolin.git

cd Pangolin

mkdir build && cd build

cmake -DCMAKE\_INSTALL\_PREFIX:PATH=$HOME -DCPP11\_NO\_BOOST= 1 ..

make -j4

sudo make install

安装opencv

sudo apt-get install build-essential libgtk2.0-dev libvtk5-dev libjpeg-dev libtiff5-dev libjasper-dev libopenexr-dev libtbb-dev

git clone <https://github.com/opencv/opencv>

cd opencv/

git checkout b38c50b3d0c31e82294315ec44b54b7ef559ef12

mkdir build && cd build

cmake -DCMAKE\_INSTALL\_PREFIX:PATH=$HOME -D CMAKE\_BUILD\_TYPE=Release ..

make -j4

sudo make install

安装eigen

sudo apt-get install libxmu-dev

sudo apt-get install libeigen3-dev

安装GTSAM

**sudo apt-get install libc6-dev-i386**

git clone https://github.com/borglab/gtsam -b develop

安装gflags

git clone <https://github.com/gflags/gflags.git>

mkdir build && cd build

cmake .. -DCMAKE\_INSTALL\_PREFIX:PATH=$HOME -DGFLAGS\_NAMESPACE=google -DCMAKE\_CXX\_FLAGS=-fPIC ..

make -j4

sudo make install

安装glog

sudo apt-get install autoconf automake libtool

git clone https://github.com/google/glog

./autogen.sh

./configure CPPFLAGS="-I/usr/local/include -fPIC" LDFLAGS="-L/usr/local/lib"

make -j4

sudo make install

if( ${ANDROID\_ABI} MATCHES "armeabi-v7a" )

# set(CMAKE\_CXX\_FLAGS "${CMAKE\_CXX\_FLAGS} -g -mfloat-abi=softfp -mfpu=neon -march=armv7-a -mtune=cortex-a8")

-march=i686

-march=prescott

-march=armv8-a

-march=pentium4

可以禁止编译32位版本，在configure时带上参数--disable-multilib

make DESTDIR=/install/directory install

set(CMAKE\_CXX\_STANDARD 11)

set(CMAKE\_CXX\_FLAGS "-std=c++11 -march=armv8-a -mfloat-abi=softfp -mfpu=neon-vfpv4 ${CMAKE\_CXX\_FLAGS} -ltbb")

echo nameserver 8.8.8.8 > /etc/resolv.conf

source /etc/resolv.conf

apt update

apt install make cmake

apt-get install g++-arm-linux-gnueabihf

apt install libglew-dev libboost-dev libboost-thread-dev libboost-filesystem-dev -y

apt-get install build-essential libgtk2.0-dev libvtk5-dev libjpeg-dev libtiff5-dev libjasper-dev libopenexr-dev libtbb-dev -y

set(BOOST\_ROOT /usr/local/boost)

SET(CMAKE\_SYSTEM\_PROCESSOR arm)

set(YOUR\_TARGET\_ARCH armv8-a)

set(CMAKE\_CXX\_FLAGS "-std=c++11 -march=armv8-a -mfloat-abi=softfp -mfpu=neon-vfpv4 ${CMAKE\_CXX\_FLAGS} -ltbb")

#list\_append\_cache(GTSAM\_COMPILE\_OPTIONS\_PUBLIC "-march=armv8-a")

docker.image('arm:2.0').inside{

mkdir /etc/apt/preferences.d

apt-get install apt-transport-https

cpp-5{a} g++-5 gcc-5 gcc-5-base{a} libasan2{a} libgcc-5-dev{a} libisl15{a} libmpfr4{a} libstdc++-5-dev{a}

aptitude install gcc-5=5.4.0-6ubuntu1~16.04.12 g++-5=5.4.0-6ubuntu1~16.04.12

apt-get install libglew-dev libboost-dev libboost-thread-dev libboost-filesystem-dev libboost-all-dev libx11-dev libxmu-dev libglu1-mesa-dev libgl2ps-dev libxi-dev g++ libzip-dev libpng12-dev libcurl4-gnutls-dev libfontconfig1-dev libsqlite3-dev libglew\*-dev libssl-dev build-essential libgtk2.0-dev libvtk5-dev libjpeg-dev libtiff5-dev libjasper-dev libopenexr-dev libtbb-dev libeigen3-dev libc6-dev-i386 autoconf automake libtool gcc g++ -y

Cake -DCMAKE\_CXX\_COMPILER="/usr/local/gcc-linaro-5.4.1-2017.01-x86\_64\_aarch64-linux-gnu/bin/aarch64-linux-gnu-g++" -DCMAKE\_C\_COMPILER="/usr/local/gcc-linaro-5.4.1-2017.01-x86\_64\_aarch64-linux-gnu/bin/aarch64-linux-gnu-gcc" ..

cmake -DCMAKE\_CXX\_COMPILER="/root/code/workspace/code/gcc-linaro-5.4.1-2017.01-x86\_64\_aarch64-linux-gnu/bin/aarch64-linux-gnu-g++" -DCMAKE\_C\_COMPILER="/root/code/workspace/code/gcc-linaro-5.4.1-2017.01-x86\_64\_aarch64-linux-gnu/bin/aarch64-linux-gnu-gcc" ..

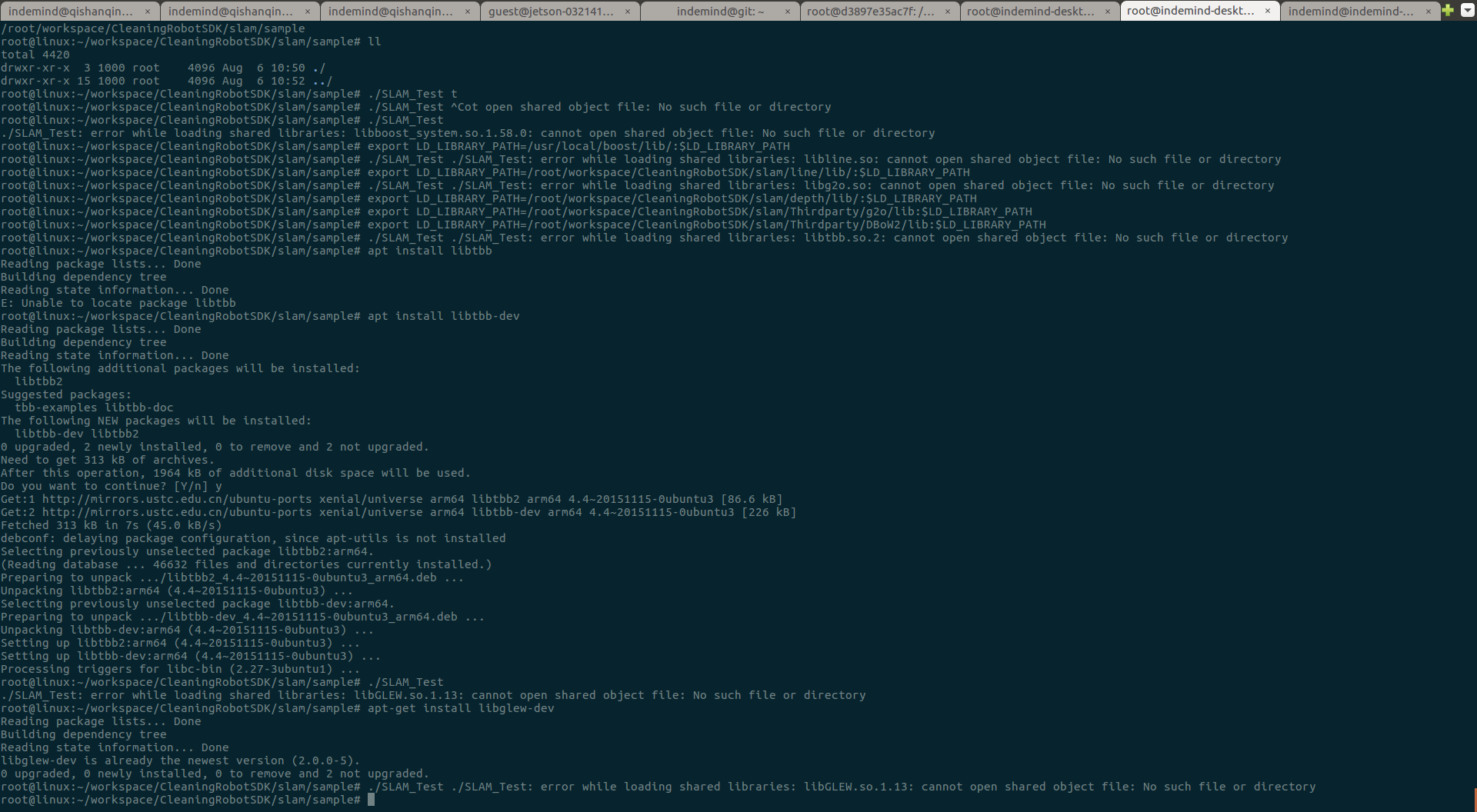
cmake -DCMAKE\_CXX\_COMPILER="~/workspace/gcc-linaro-5.4.1-2017.01-x86\_64\_aarch64-linux-gnu/bin/aarch64-linux-gnu-g++" -DCMAKE\_C\_COMPILER="~/workspace/gcc-linaro-5.4.1-2017.01-x86\_64\_aarch64-linux-gnu/bin/aarch64-linux-gnu-gcc" ..

No rule to make target 'lib/libSGM.so', needed by '../lib/libINDEMIND\_SLAM.so'. Stop.

192.168.1.253 1.100 firefly

apt autoremove dh-python gir1.2-harfbuzz-0.0 ibverbs-providers icu-devtools libboost-all-dev libboost-atomic-dev libboost-atomic1.65-dev libboost-atomic1.65.1 libboost-chrono-dev libboost-chrono1.65-dev libboost-chrono1.65.1 libboost-container-dev libboost-container1.65-dev libboost-container1.65.1 libboost-context-dev libboost-context1.65-dev libboost-context1.65.1 libboost-coroutine-dev

export LD\_LIBRARY\_PATH=/root/workspace/CleaningRobotSDK/slam/lib/libINDEMIND\_SLAM.so:$LD\_LIBRARY\_PATH



ls -lrt| awk '{print $9}'| head -n 10 | xargs rm -rf

/var/atlassian/confluence/backups

/var/atlassian/jira/export

082c97e132e073647f00f77c8fa7dd48c9c3c157

少 resbase 多 简单

多 dev rebase 高通 少 复杂

export ALTERNATE\_EDITOR=""

export EDITOR=emacsclient

sdk

assignee = ruoyangxiong OR assignee = zhaoyanan OR assignee = mayong

sdk team task

Slam

assignee = fangwanyuan OR assignee = liuqiaojun OR assignee = qiuliru OR assignee = tangaoyang OR assignee = xuhaonan OR assignee = liusichao OR assignee = wangren

slam team task

Path\_plan

assignee = wangshuaishuai OR assignee = liuziwen OR assignee = zhangqiang OR assignee = sunyu

path\_planning team task

deep\_learning

assignee = hexin OR assignee = sunhao OR assignee = liuyongkang

深度学习任务

嵌入式

嵌入式组任务

assignee = lishaohui OR assignee = weibinbin

Hardware

硬件部门任务

assignee = hanchong OR assignee = tianwei OR assignee = liuyaru OR assignee = zhaojunchao

assignee = zhaotielong OR assignee = 马增炳 OR assignee = wangjinggen

测试组任务

test

assignee = hanzhiqiang

software

软件组任务