==========

==========

==========

Data Mining Initial Servers

2019-08-20

==========

==========

==========

Background:

This project is to address the Dark Data Mining needs of Technology. Data is made up of three elements: BUSINESS CRITICAL DATA, REDUNDANT OBSOLETE AND TRIVIAL (ROT) DATA, and DARK DATA:

- Gartner defines Data Mining as the process of discovering meaningful correlations, patterns and trends by sifting through large amounts of data stored in repositories.

- Gartner defines Dark Data as the information assets organizations collect, process and store during regular business activities, but generally fail to use for other purposes.

Objectives:

- Install High-Performance Analytics Software Stacks: Julia, KNIME, JanusGraph, and R plus all supporting software needed for mining data stored in Technology's Centralized Logging System

- Ability to create new use cases and gather requirements with Stakeholders for the use of Data Mining in Technology

- Create New Technology Data Asset Classes:

    - Technology Data Ontologies Assets: Common Information Models, Data Dictionaries, Taxonomies, and Ontologies

- Technology Data Cleaning Assets: Labeled Data, Conforming to ETL, Schemas, Structures, Parsing Rules

- Technology Data Exploration Assets: Shape of Data, Outliers, per Stakeholder Use Case

- Technology Data Modeling Assets: Reuse of models, algorithms, features will grow over time

- Technology Data Prediction Assets: Lifecycle of prediction models on new data

- Reuse of Technology Data Assets: Scope is Hundreds of Thousands of Technology Data Explorations, Hypothesis Testing Results, and Models can be reused

Benefits:

- Benefit 1: VALUABLE ASSETS
    - Data
        - Sets
        - Labels
        - Features
    - Code
        - Reusable ML code
        - APIs
    - Models
        - Machine Learning
        - Statistics
        - Neural Nets
    - Dictionaries
        - Ontologies
        - Common Information Models
    - Staff
        - Machine Learning Know-how
        - Statistics Know-how
        - Streaming Know-how
        - Graph Analysis Know-how

- Neural Nets Know-how

- Backend/Frontend Know-how

- Platform

- Data Science Toolkits

- Cutting-edge platforms leveraging Tensors which run on CPUs and can be scaled out to run on GPUs without changing much code

- APIs

- Benefit 2: ABILITIES

- Data Extraction—Analyze and crawl your data in a pattern to help you organize.

- Data Visualization—Interactive viz used by Data Mining experts to explore data.

- Linked Data Management—This is Graph Analysis; Provides a set of guidelines to organize data that is connected.

- Statistical Analysis—Uses basic descriptive statistics to explore and model data and its distribution shapes. Useful for exploration, description, probabilities, likelihood, causality. Vital.

- Machine Learning—Uses statistics and ML algorithms for prediction from patterns; Extracts useful patterns from your data. Vital.

- Neural Network Learning—Uses Low-Dimensional Neural Networks run on CPUs. Useful to throw into the mix when comparing against classic ML algorithms for accuracy, recall, precision, and scores. Vital.

- Deep Learning—Uses High-Dimensional Neural Networks run on GPUs. These discover latent structures within unlabeled, unstructured data. Vital.

- Semantic Search—This allows the context search engine to distinguish between different entities (people, places, and things) and interpret intent instead of simply finding literal matches.

- Text Mining—Derive high-quality information from your text databases.

- Benefit 3: INNOVATION

- Dark Data Discovery

- Tensor Designs

- Classifications

- Clusters

- Prediction

- Pattern Identification

- Benefit 4: IMPACT

    - Transparency—START HERE

    - Understanding—Explainable AI (XAI) aids with understanding

    - Collaboration—Builds from understanding each other

    - Trust—Builds from collaboration

    - Insight—Builds on trust

    - Influence—These are believable insights as Data Mining begins to be trusted and insights are acted upon

    - Leadership—Group of stakeholder champions who have benefitted from believable and insightful Data Mining, and this group carries significant influence to drive more data collection within the org and more use cases to a wider network of stakeholders

- Data Mining Types:

    - Anomaly Detection—Useful for fraud detection

    - Association—Identifies recommendations of data frequently found near each other

    - Clustering—Involves finding groups with similar characteristics

    - Classification—Sorts items into categories using the output of clustering or with categories premade by a data scientist

    - Natural Language Processing—Semantic search, context search, word counts, sentiment analysis, topics, themes, ELO, scoring, survey analysis, NPS

    - Regression—Advanced statistics common to predictive analytics

    - Semantic Search—Uses GANs, Genetic Algorithms (GAs) for optimization, and Deep Learning techniques on CPUs and GPUs to provide contextual search

- Components:

    - Advanced Analysis—Data Mining uses advanced analysis techniques found in Scientific Methods, Statistical Analysis, Time Series Analysis, Longitudinal Studies, Graph Theory, Machine Learning, Neural Network Theory, and Deep Learning

    - AI—Data Mining is a subset of AI and Data Science

    - AutoML—Automatic Machine Learning (AutoML) steps are used in both Data Mining and Machine Learning for rapid prototyping, running through multiple algorithms, and for optimizing feature selection

- Future Focused—Data Mining is focused on finding patterns and predicting the future

- Knowledge Graph—Knowledge Graph Databases are used to represent and understand linked data

- Machine Learning—Machine Learning uses the results of Data Mining to learn something new about the data

- Patterns—Data Mining finds patterns in the data

- Tensors—Data Mining uses scientific software, language libraries and frameworks (Julia, PyTorch etc), APIs, models, and data structures (Tensors for example) compatible with both CPU and GPU for applied computation paradigms specific to the world of AI

- Visualizations—Visualizations are used in the exploration phase to shape, understand behavior and map linked data characteristics to perform hypothesis testing, investigation and contextual search by a data scientist or analyst

-

Data Mining is not BI

==========

==========

==========

Local Setup

==========

==========

==========

Local Admin on Windows

Local Guest OS: Hyper-V arch linux

- i3

- yay

- tmux

Local script: ~/bin/mounthost.sh

#! /bin/bash

sudo mount -t cifs -o username=jfolkers,uid=$(id -u),guid=$(id -g)
//ASTH577YT2TECH/C$/Users/jfolkers/data /home/jf/data

tmux

tmux new -s local

Session is local

Rename pane to [arch] <control-b ,>

Create new pane <control-b c>

Name it infrdm01d <control-b ,>

ssh to astof-infrdm01d

tmux new -s dev

Session is dev

Rename pane to [rhel]

Create new pane <control-b c>

Name it infrdm01s <control-b ,>

ssh to astdc-infrdm01s

tmux new -s staging

Session is staging

Rename pane to [rhel]

infrdm01d

Create new pane <control-b control-b c>

Name it edex5 <control-b control-b ,>

Split horizontal <control-b control-b ">

Split horizontal again <control-b control-b ">

In bottom window put iostat

iostat -xhm sdb 10

In next up put dstat

dstat -lvrn 10

Split top pane vertically <control-b control-b %>

On Right: Clock, iftop, ns-estab.sh

On left: top nmon

infrdm01s

Create new pane <control-b control-b c>

Name it edex5 <control-b control-b ,>

Split horizontal <control-b control-b ">

Split horizontal again <control-b control-b ">

In bottom window put iostat

iostat -xhm sdb 10

In next up put dstat

dstat -lvrn 10

Split top pane vertically <control-b control-b %>

On Right: Clock, iftop, ns-estab.sh

On left: top nmon

==========

==========

==========

Weekly Milestones

==========

==========

==========

2019-08-19—Week 1 Setup

Verification of Server Builds (DONE)

Servers Setup Part 1 Tools and Utilities (DONE)

2019-08-26—Week 2 Software Install

Servers Setup—Part 2 Benchmarks (DONE)

- Dev

- Staging

Servers Software Install—Part 1 (Data Science Libraries)

- Pandas (AVAILABLE FOR PYTHON 2; NOT DONE YET FOR PYTHON3)

- Numpy (DONE)

- SciKitLearn (NOT DONE)

Servers Software Install—Part 2 (Languages)

- C#

- JuliaPro (DONE)

- Python3 (DONE)

- Perl (DONE)

- Java (LOCAL: DONE) (SERVERS: NOT DONE)

Servers Software Install—Part 3 (Frameworks and Platforms) (...IN PROGRESS...)

- Frameworks:

- DRILL

- .NET (mono)

- Keras

- Platforms:

- KNIME+H2O

- NLP

- Orange

- R+H2O (DONE)

- Spark

- SQL

- TensorFlow

Servers Software Install—Part 4 (Graph Technology) (...PENDING...)

- Graph Databases: Janus Graph

- Graph Languages: Gremlin

- Knowledge Graph: GRAKN

- Stacks: Tinkerpop

Servers Software Install—Part 5 (...PENDING...)

- IDE: JupyterLab

- IDE: GhostVIM (DONE)

Servers Software Install—Part 6 (Viz) (...PENDING...)

- TensorBoard, Shiny, D3.js, three.js

==========

==========

==========

Issues

==========

==========

==========

2019-08-19—Week 1 Setup

ISSUE: Installation—Python science libs (matplotlib, etc)

FIXED: Add repos, install with yum, rpms, and pip

2019-08-26—Week 2 Software Install

ISSUE: KNIME Analytics Platform requires Remote in to Linux

FIXED: Marcus suggests No Machine

2019-09-04 Week 3 Compile R

ISSUE: ldd R shows libs not linking to openblas

FIXED: Compile both R and BLAS and link libraries with symlinks

Tested using R-Benchmark-25.R for up to 100x faster improvement with OpenBLAS over stock BLAS

https://www.r-bloggers.com/for-faster-r-use-openblas-instead-better-than-atlas-trivial-to-switch-to-on-ubuntu/

2019-09-12 Week 4 Compile R Packages

ISSUE: tidymodels not compiling due to stan and stanarm dependencies requiring gcc++14 standards

SUB-ISSUE: RedHat comes with gcc++ version 3.8.5 with most gcc++14 standards implemented but not all

SUB-ISSUE: Upgrading compilers can be explodey on servers and not recommended for most of the time

FIXED: stan compiled using -std c++1y

FIXED: stanarm compiled using another linux system with a newer gcc++ version 9.1 that implements gcc++14 fully

FIXED: also installed gcc-gfortran using another linux to compile stanarm

==========

==========

==========

Server Build Verification

==========

==========

==========

Verified server specs:

- astof-infrdm01d DNS: 10.27.32.45 4 cores 32G ram /data 500G /opt 40G /usr 15G

- astdc-infrdm01s DNS: 10.22.32.22 8 cores 64G ram /data 1T /opt 40G /usr 15G

Red Hat Enterprise Linux Server release 7.7 (Maipo)

Minimum kernel needed for modern container support (LXC, Docker, etc) is 3.10

Current kernel verified: 3.10

==========

==========

==========

Ports

==========

==========

==========

22 - ssh

80 - web

445 - CIFS (server reaches out to remote port 445 on laptop)

2181 - zookeeper

3306 - mysql

3838 - Shiny

4000 - NX No Machine

4040 - Spark

4698 - NX No Machine-UDP

5044 - Logstash

5601 - Kibana

6000 - X11 Server

6006 - TensorBoard

8080 - Monitorix

8081 - McAfee agent

8787 - RStudio Server

8888 - JupyterLab

9092 - Kafka

9200 - Elastic

64440 - KNIME Web

any	10.27.32.45	22	ssh	Development	Permit
any	10.27.32.45	80	web	Development	Permit
10.27.32.45	any	445	CIFS	Development	Permit
10.27.32.45	any	2181	Kafka—ZooKeeper	Development	Permit
10.27.32.45	any	2304	Spark—Thift Server	Development	Permit
10.27.32.45	any	3306	mysql	Development	Permit
10.27.32.45	any	3838	R—Shiny	Development	Permit
10.27.32.45	any	4000	NX NoMachine	Development	Permit
10.27.32.45	any	4040	Spark	Development	Permit
10.27.32.45	any	4567	GRAKN	Development	Permit
10.27.32.45	any	4698	NX NoMachine	Development	Permit
10.27.32.45	any	5044	Logstash	Development	Permit
10.27.32.45	any	5181	Spark—ZooKeeper	Development	Permit
10.27.32.45	any	5601	Kibana	Development	Permit
10.27.32.45	any	5660	Spark—MapR Filesystem Server	Development	Permit
10.27.32.45	any	5692	Spark—MapR Filesystem Server	Development	Permit
10.27.32.45	any	6006	TensorBoard	Development	Permit

10.27.32.45	any	7000	JanusGraph—CassandraDevelopment	Permit	
10.27.32.45	any	7001	JanusGraph—CassandraDevelopment	Permit	
10.27.32.45	any	7077	Spark—Standalone Master (RPC)	Development	Permit
10.27.32.45	any	7199	JanusGraph—CassandraDevelopment	Permit	
10.27.32.45	any	7222	Spark—CLDB	Development	Permit
10.27.32.45	any	7337	Spark—External Shuffle Service	Development	Permit
10.27.32.45	any	8032	Spark—Resource Manager	Development	Permit
10.27.32.45	any	8080	Monitorix	Development	Permit
10.27.32.45	any	8081	McAfee agent	Development	Permit
10.27.32.45	any	8580	Spark—Standalone Master (Web UI)	Development	Permit
10.27.32.45	any	8581	Spark—Standalone Worker	Development	Permit
10.27.32.45	any	8787	RStudio Server	Development	Permit
10.27.32.45	any	8888	JupyterLab	Development	Permit
10.27.32.45	any	8980	Spark—Standalone Master (Web UI)	Development	Permit
10.27.32.45	any	8981	Spark—Standalone Worker	Development	Permit
10.27.32.45	any	9042	JanusGraph	Development	Permit
10.27.32.45	any	9092	Kafka	Development	Permit
10.27.32.45	any	9160	JanusGraph—Thrift Server	Development	Permit
10.27.32.45	any	9200	Elastic	Development	Permit
10.27.32.45	any	9300	Elastic	Development	Permit
10.27.32.45	any	18080	Spark—History Server	Development	Permit
10.27.32.45	any	18480	Spark—History Server	Development	Permit
10.27.32.45	any	48555	GRAKN	Development	Permit
10.27.32.45	any	64440	KNIME	Development	Permit

any	10.22.32.22	22	ssh	Stage	Permit

any	10.22.32.22	80	web	Stage	Permit
10.22.32.22	any	445	CIFS	Stage	Permit
10.22.32.22	any	2181	Kafka—ZooKeeper	Stage	Permit
10.22.32.22	any	2304	Spark—Thift Server	Stage	Permit
10.22.32.22	any	3306	mysql	Stage	Permit
10.22.32.22	any	3838	R—Shiny	Stage	Permit
10.22.32.22	any	4000	NX NoMachine	Stage	Permit
10.22.32.22	any	4040	Spark	Stage	Permit
10.22.32.22	any	4567	GRAKN	Stage	Permit
10.22.32.22	any	4698	NX NoMachine	Stage	Permit
10.22.32.22	any	5044	Logstash	Stage	Permit
10.22.32.22	any	5181	Spark—ZooKeeper	Stage	Permit
10.22.32.22	any	5601	Kibana	Stage	Permit
10.22.32.22	any	5660	Spark—MapR Filesystem Server	Stage	Permit
10.22.32.22	any	5692	Spark—MapR Filesystem Server	Stage	Permit
10.22.32.22	any	6006	TensorBoard	Stage	Permit
10.22.32.22	any	7000	JanusGraph—Cassandra	Stage	Permit
10.22.32.22	any	7001	JanusGraph—Cassandra	Stage	Permit
10.22.32.22	any	7077	Spark—Standalone Master (RPC)	Stage	Permit
10.22.32.22	any	7199	JanusGraph—Cassandra	Stage	Permit
10.22.32.22	any	7222	Spark—CLDB	Stage	Permit
10.22.32.22	any	7337	Spark—External Shuffle Service	Stage	Permit
10.22.32.22	any	8032	Spark—Resource Manager	Stage	Permit
10.22.32.22	any	8080	Monitorix	Stage	Permit
10.22.32.22	any	8081	McAfee agent	Stage	Permit
10.22.32.22	any	8580	Spark—Standalone Master (Web UI)	Stage	Permit
10.22.32.22	any	8581	Spark—Standalone Worker	Stage	Permit
10.22.32.22	any	8787	RStudio Server	Stage	Permit
10.22.32.22	any	8888	JupyterLab	Stage	Permit

10.22.32.22 any 8980 Spark—Standalone Master (Web UI) Stage Permit

10.22.32.22 any 8981 Spark—Standalone Worker Stage Permit

10.22.32.22 any 9042 JanusGraph Stage Permit

10.22.32.22 any 9092 Kafka Stage Permit

10.22.32.22 any 9160 JanusGraph—Thrift Server Stage Permit

10.22.32.22 any 9200 Elastic Stage Permit

10.22.32.22 any 9300 Elastic Stage Permit

10.22.32.22 any 18080 Spark—History Server Stage Permit

10.22.32.22 any 18480 Spark—History Server Stage Permit

10.22.32.22 any 48555 GRAKN Stage Permit

10.22.32.22 any 64440 KNIME Stage Permit

==========

==========

==========

BEGIN HERE

==========

==========

==========

Verify jfolkers has sudoers (DONE by Systems) (approved by Marcus 2019-08-20)

sudo -i

AD Group—Add jfolkers to linuxuser (DONE by Systems) (approved by Marcus 2019-08-20)

Technology Systems does this

Allows access to all RedHat7 systems

SELinux—Change from enforced to permissive (DONE) (approved by Marcus 2019-08-20)

sudo vi /etc/selinux/config

Verify (DONE)

getenforce

Firewall—Disable (DONE) (approved by Marcus 2019-08-20)

sudo service firewalld stop

sudo systemctl disable firewalld

Reboot (DONE)

sudo reboot

==========

==========

==========

Shell

==========

==========

==========

CLI-vi (DONE)

vi .bashrc

set -o vi

==========

==========

==========

Repositories

==========

==========

==========

RHEL 7.x

EPEL

wget https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm

sudo yum install epel-release-latest-7.noarch.rpm

RPMFUSION

sudo yum localinstall --nogpgcheck https://download1.rpmfusion.org/free/el/rpmfusion-free-release-7.noarch.rpm

==========

==========

==========

Scripts

==========

==========

==========

bin (DONE)

mkdir ~/bin

cpu1 (DONE)

vi ~/bin/cpu1.sh

#! /bin/bash

ps -eo pid,ppid,cmd,%mem,%cpu --sort=-%cpu | head

cpu2 (DONE)

vi ~/bin/cpu2.sh

#1 /bin/bash

top -b -n1 | head -n 12 | tail -n 6

custom.sh (DONE)

vi /etc/profile.d/custom.sh

```
#! /bin/bash

echo
echo
echo 'Technology Infrastructure - Data Mining Team'
echo
echo

/usr/local/bin/dataminerlogo

echo
echo
echo "Data Miner Server - Dev'
echo
echo

/usr/bin/inxi

echo

# Aliases and functions

set -o vi

export TERM="screen-256color"

# R libs
export LD_LIBRARY_PATH=/usr/local/lib64/:$LD_LIBRARY_PATH
```

```
export R_LIBS_USER="/opt/R/library"
export R_HOME=/opt/R/3.6.1/lib64/R

# R ALL USERS - Installed Packages
export R_LIBS_USER="/opt/R/library"

PATH=/opt/R/3.6.1_openblas/bin/:$PATH:/usr/local/bin
export PATH

powerline-daemon -q
POWERLINE_BASH_CONTINUATION=1
POWERLINE_BASH_SELECT=1
. /usr/local/lib/python3.6/site-packages/powerline/bindings/bash/powerline.sh




# dataminerlogo (DONE)
# compile with cc and put in /usr/local/bin/dataminerlogo
#include <stdio.h>

void red () {

  printf("\x1b[38;5;198m");

}



void green () {
```

```
  printf("\x1b[38;5;82m");

}



void bright () {

  printf("\x1b[38;5;87m");

}

void reset () {

  printf("\033[0m");

}



int main () {

  red();

  printf("  ^   ^   ^   ^       ^   ^   ^   ^   ^ \n");

  reset();
```

```
  printf(" /D\\  /A\\  /T\\  /A\\       ");

  bright();

  printf("/M\\  /I\\  /N\\  /E\\  /R\\ \n");

  green();

  printf("<___><___><___><___>     <___><___><___><___><___>\n");

  reset();

  return 0;

}
```

```
# netstat (DONE)
vi ~/bin/ns-estab.sh
#! /bin/bash
netstat -antp 2>/dev/null | grep -v 'tcp6' | grep 'ESTAB' | cut -c 21-61 | sed 's/    //'
```

```
vi ~/bin/ns-listen.sh (DONE)
#! /bin/bash
netstat -antp 2>/dev/null | grep -v 'tcp6' | grep 'LISTEN' | cut -c 21-61 | sed 's/    //'
```

top-specific.sh (DONE)

```
#! /bin/bash
top -p `pgrep -d ',' "rserver|knime"`
```

==========

==========

==========

Tools—Part 1

==========

==========

==========

asciinema (DONE)

```
sudo yum -y install asciinema
```

cifs-utils (DONE)

```
sudo yum -y install cifs-utils
```

dstat (DONE)

```
sudo yum -y install dstat
```

fuse already installed

git (DONE)

sudo yum -y install git

hwinfo (DONE)

sudo yum -y install hwinfo

inxi (DONE)

sudo yum -y install inxi

libappindicator-gtk3 (dependencies for dissenter) <<<<<<<<<<<<<<<<<

sudo yum install libappindicator-gtk3

liberation-fonts (dependencies for dissenter) <<<<<<<<<<<<<<<<

sudo yum install 'liberation-fonts'

libXScrnSaver (dependencies for dissenter) <<<<<<<<<<<<<<<<<<<<<

sudo yum install 'libXss.so.1()(64bit)'

lsof (DONE)

sudo yum -y install lsof

net-tools already installed

openssh already installed

openssl already installed

pcre already installed

redhat-lsb-core (dependencies for dissenter) <<<<<<<<<<<<<<<<<<<<<<<<<<

sudo yum install redhat-lsb-core

tmux RH HAS OLD TMUX 1.8—DO NOT INSTALL

install tmux 2.x for mouse scroll ability (see below in Tools—Part 4)

time (DONE)

sudo yum -y install time

tex (DONE)

sudo yum -y install tex

sudo yum -y install texinfo

samba (DONE)

sudo yum -y install samba

snappy already installed

vim (DONE)

sudo yum -y install vim

yum-utils already installed

==========

==========

==========

Tools—Part 2

Extra Packages for Enterprise Linux (EPEL)

==========

==========

==========

Add epel repository

https://access.redhat.com/discussions/3140721

sudo rpm -ivh https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm

sudo yum update

iftop (DONE)

sudo yum -y install iftop

iptraf-ng (DONE)

sudo yum -y install iptraf

jq (DONE)

sudo yum -y install jq

ncdu (DONE)

sudo yum -y install ncdu

nmon (DONE)

sudo yum -y install nmon

==========

==========

==========

Tools—Part 3

Containers

==========

==========

==========

lxc (SKIP UNTIL NEEDED)

https://www.linuxjournal.com/content/everything-you-need-know-about-linux-containers-part-i-linux-control-groups-and-process

yum install libcgroup libcgroup-tools

https://www.youtube.com/watch?v=sK5i-N34im8&feature=youtu.be @ 41:48

Make sure mount points are private; make sure mount points inside containers do not bleed out to the host

However, https://bugs.debian.org/cgi-bin/bugreport.cgi?bug=739593

makes the opposite case. Just leave it shared at / and opt out of mount namespace for your control group

mount --make rprivate /

Create directories for images and containers

mkdir -p images containers

Create image

btrfs subvol create images/alpine

get alpine image from docker

CID=$(docker run -d alpine true)

echo $CID

```
# cbdb549b8b0dd644...
# docker generates a tarball... extract it into alpine directory
docker export $CID | tar -C images/alpine/ -xf-
ls images/alpine/
bin dev etc home lib linuxrc media mnt proc root run sbin sys tmp usr var
# make a snapshot
btrfs subvol snapshot images/alpine/ containers/tupperware
# keep track
touch containers/tupperware/THIS_IS_TUPPERWAAARE
ls containers/tupperware/
bin dev etc home lib linuxrc media mnt proc root run sbin sys THIS_IS_TUPPERWAAAARE tmp usr var
# change
chroot containers/tupperware/ sh
ls
THIS_IS_TUPPERWAAARE bin dev etc home lib linuxrc media mnt proc root run sbin sys tmp usr var
exit
# use namespaces... give me all the namespaces except user namespace
unshare --mount --utc --ipc --net --pid --fork bash
hostname tupperware
exec bash
# yes in container
# not really a container yet, but we do have namespaces
ps
# not pid1 yet...
pidof unshare
6902
kill 6902
# no such process... because I'm in the namespace and that PID doesn't exist in this namespace
# but if I mount proc as a test... now I only see processes inside...(I see bash has PID 1... etc)
```

```
mount -t proc none /proc
ps
ps faux
umount /proc/
# now i want to get into the filesystem of my container
cd /btrfs/containers/tupperware
mkdir oldroot
pivot_root . oldroot/
cd /
mount --bind //btrfs/containers/upperware/ /btrfs/containers/tupperware/
mount --move /btrfs/containers/tupperware/ /btrfs/
cd /btrfs/
ls
pivot_root . oldroot/
cd /
ls
# I'm in my tupperware container
mount -t proc none /proc
mount
# still tons of mounts from host
umount -a
mont -t proc none /proc
mount
# still have old root
umount -l /oldroot/
mount
# now I have just my container
# no networking yet...
ping 4.2.2.1
```

Docker (DONE)

Don't use docker. Requires a Red Hat subscription for container-selinux dependency.

Add docker repository

https://www.unixarena.com/2018/06/how-to-install-docker-on-redhat-linux-rhel-centos.html/

Docker provides the community edition for Centos which also can be installed on RHEL.

sudo -i

yum-config-manager --add-repo https://download.docker.com/linux/centos/docker-ce.repo

sudo yum install docker-ce

Error: Package: containerd.io-1.2.6-3.3.el7.x86_64 (docker-ce-stable)

Requires: container-selinux >= 2:2.74

Error: Package: 3:docker-ce-19.03.1-3.el7.x86_64 (docker-ce-stable)

Requires: container-selinux >= 2:2.74

Redhat Linux (RHEL 7) requires extra rpms. Enable the extras RHEL repository. This ensures access to the container-selinux package which is required by docker-ce. It requires a Red Hat subscription.

yum-config-manager --enable rhel-7-server-extras-rpms

This system has no repositories available through subscriptions.

subscription-manager repos --enable=rhel-7-server-extras-rpms

This system has no repositories available through subscriptions.

https://stackoverflow.com/questions/45272827/docker-ce-on-rhel-requires-container-selinux-2-9

Installing the Selinux from the Centos repository worked for me:

1. Go to http://mirror.centos.org/centos/7/extras/x86_64/Packages/

2. Find the latest version for container-selinux i.e. container-selinux-2.21-1.el7.noarch.rpm

3. Run the following command on your terminal: $ sudo yum install -y http://mirror.centos.org/centos/7/extras/x86_64/Packages/**Add_current_container-selinux_package_here**

4. The command should looks like the following $ sudo yum install -y http://mirror.centos.org/centos/7/extras/x86_64/Packages/container-selinux-2.21-1.el7.noarch.rpm

Just install selinux latest version to fix it:

sudo yum install -y http://mirror.centos.org/centos/7/extras/x86_64/Packages/container-selinux-2.95-2.el7_6.noarch.rpm

sudo yum install docker-ce

https://docs.docker.com/install/linux/linux-postinstall/

sudo groupadd docker

sudo usermod -aG docker $USER

newgrp docker

Change docker's image and data path to point to /opt/docker

This avoids running out of space on the /var mount (/var/lib/docker)

Edit /etc/docker/daemon.json

{

"data-root": "/opt/docker"

}

Restart daemon

sudo systemctl stop docker

sudo systemctl start docker

sudo systemctl enable docker

docker run hello-world

Docker Compose (DONE)

sudo yum install docker-compose

Python Virtual Environment (DONE)

https://developers.redhat.com/blog/2018/08/13/install-python3-rhel/

```
# Create py36-venv and activate it
python3 -m venv py36-venv
source py36-venv/bin/activate
# python3 -m pip install ...some modules...
```

```
================================================
================================================
================================================
            Tools—Part 4
           Manual Install
================================================
================================================
================================================
```

```
# Diskplorer Requirement: fio (DONE)
git clone https://github.com/axboe/fio
./configure
make
sudo make install
```

```
# Diskplorer Requirement: libaio (DONE)
sudo yum install libaio libaio-devel
```

Diskplorer (DONE)

git clone https://github.com/avikivity/diskplorer

./configure

make

sudo make install

neofetch (NOT DONE)

powerline (DONE)

https://www.tecmint.com/powerline-adds-powerful-statuslines-and-prompts-to-vim-and-bash/

sudo pip3 install git+git://github.com/Lokaltog/powerline

wget https://github.com/powerline/powerline/raw/develop/font/PowerlineSymbols.otf

wget https://github.com/powerline/powerline/raw/develop/font/10-powerline-symbols.conf

sudo mv PowerlineSymbols.otf /usr/share/fonts/

sudo fc-cache -vf /usr/share/fonts/

sudo mv 10-powerline-symbols.conf /etc/fonts/conf.d/

get location of powerline

pip3 show powerline-status

put powerline scripts in .bashrc

vi .bashrc

export TERM=”screen-256color”

powerline-daemon -q

POWERLINE_BASH_CONTINUATION=1

POWERLINE_BASH_SELECT=1

. /usr/local/lib/python3.6/site-packages/powerline/bindings/bash/powerline.sh

put powerline scripts in .vimrc

vi .vimrc

set rtp+=/usr/local/lib/python3.6/site-packages/powerline/bindings/vim/

set laststatus=2

set t_Co=256

put powerline scripts in .tmux.conf

vi .tmux.conf

source /usr/local/lib/python3.6/site-packages/powerline/bindings/tmux/powerline.conf

set -g mouse on

setw mode-keys vi

BUG: current version of powerline script does not load proper symbols in tmux due to tmux -V returning 'tmux next-3.1'

powerline-config tmux setup: shows __init__.py stops with ValueError: invalid literal for int() with base 10: 'next-3'

FIX: edit /usr/local/lib/python3.6/site-packages/powerline/bindings/tmux/__init__.py

in get_tmux_version, add line above suffix: major = "3"

SpaceVIM (DONE)

https://spacevim.org/quick-start-guide/

curl -sLf https://spacevim.org/install.sh | bash

(TODO) NEXT STEPS: VERIFY PYTHON IDE AND JAVA IDE WORKS

tmux (DONE)

```
# Need tmux>=2.x to get mouse scroll ability

# Remove RedHat's old tmux 1.8 version

sudo yum remove tmux

# tmux depends on ncurses-devel

sudo yum install ncurses-devel

# tmux depends on libevent 2.x but is inaccessible because RedHat has put libevent-devel in Server Optional channel

# compile libevent 2.x and store in /usr/local/lib/

wget https://github.com/libevent/libevent/archive/release-2.1.11-stable.tar.gz

tar xvfz release-2.1.11-stable.tar.gz

cd libevent-release-2.1.11-stable/

sh autogen.sh

./configure && make

sudo make install

# compile tmux 3.1

git clone https://github.com/tmux/tmux

cd tmux/

sh autogen.sh

export DIR="/usr/local"

export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$DIR/lib

./configure --prefix=$DIR CFLAGS="-I$DIR/include" LDFLAGS="-L$DIR/lib"

make

sudo make install

# Add /usr/local/lib and /usr/local/lib64 to shared library path on RHEL

# https://access.redhat.com/solutions/3020411

# https://access.redhat.com/solutions/2432

# newly compiled tmux won't find libevent-2.1.so.7 in /usr/local/lib unless you

# tell RedHat to update the ld.so.cache with /usr/local/lib and /usr/local/lib64

touch /etc/ld.so.conf.d/usrlocal.conf
```

```
echo "/usr/local/lib" > /etc/ld.so.conf.d/usrlocal.conf

echo "/usr/local/lib64" >> /etc/ld.so.conf.d/usrlocal.conf

sudo /sbin/ldconfig -v | grep -i libevent



# vim 8.1 compiled (SKIP) (USE SPACEVIM)

# https://github.com/Valloric/YouCompleteMe/wiki/Building-Vim-from-source

sudo yum install -y ruby ruby-devel lua lua-devel luajit luajit-devel ctags git python python-devel python3 python3-devel tcl-devel perl perl-devel perl-ExtUtils-ParseXS perl-ExtUtils-XSpp perl-ExtUtils-CBuilder perl-ExtUtils-Embed

# No ruby-devel

# https://stackoverflow.com/questions/30665912/no-ruby-devel-in-rhel7

# https://www.rpmfind.net/linux/rpm2html/search.php?query=ruby-devel

# RHEL7.5, this package can be installed with:

# rpm -ivh ruby-devel-2.0.0.648-33.el7_4.x86_64.rpm

# https://www.rpmfind.net/linux/centos/7.6.1810/updates/x86_64/Packages/ruby-devel-2.0.0.648-35.el7_6.x86_64.rpm

# Latest 2.0.0.648 version is "ruby-devel-2.0.0.648-35.el7_6.x86_64.rpm" but 36 does not exist except only in RedHat

# Therefore, do it all in a container because ultimately python3-devel will be needed and same problem.

# No lua-devel

# No python3-devel

# https://github.com/vim/vim

# +cscope

# C/C++ IDE

# PHP IDE

# Ruby IDE

# Perl IDE

# JavaScript IDE
```

```
# CoffeeScript IDE

# Lua IDE

# Go IDE

# Python IDE

# Java IDE

# https://stackoverflow.com/questions/30444890/vim-use-python3-interpreter-in-python-mode

# let g:pymode_python = 'python3'

# :help python-mode

cd vim

./configure \

--enable-perlinterp \

--enable-python3interp \

--enable-rubyinterp \

--enable-cscope \

--enable-gui=auto \

--enable-gtk2-check \

--enable-gnome-check \

--with-features=huge \

--enable-multibyte \

--with-x \

--with-compiledby="xorpd" \

--with-python3-config-dir=/usr/lib/python3.4/config-3.4m-x86_64-linux-gnu \

--prefix=/opt/vim74




# vim plugins (SKIP) (USE SPACEVIM)

# https://opensource.com/article/19/1/vim-plugins-developers

# (gruvbox and molokai colorthemes, auto-pairs, supertab, nerd completer, nerdtree, tagbar)
```

vim plugin: auto-pairs

https://www.vim.org/scripts/script.php?script_id=3599

git clone git://github.com/jiangmiao/auto-pairs.git ~/.vim/bundle/auto-pairs

mkdir -f .vim/plugin

cp .vim/bundle/auto-pairs/plugin/auto-pairs.vim .vim/plugin

vim plugin: nerd commenter

==========

==========

==========

Fonts

==========

==========

==========

...

==========

==========

==========

Groups

==========

==========

==========

List

sudo yum group list hidden

Scientific support (DONE)

sudo yum -y groupinstall "Scientific support" --setopt=group_package_types=mandatory,default,optional

Development tools (DONE)

sudo yum -y groupinstall "Development tools" --setopt=group_package_types=mandatory,default,optional

Infrastructure server (DONE)

sudo yum -y groupinstall "Infrastructure server"

==========

==========

==========

Languages

==========

==========

==========

go (DONE)

sudo yum -y install go

others (DONE)

sudo yum install -y ruby ruby-devel lua lua-devel luajit luajit-devel ctags git python python-devel python3 python3-devel tcl-devel perl perl-devel perl-ExtUtils-ParseXS perl-ExtUtils-XSpp perl-ExtUtils-CBuilder perl-ExtUtils-Embed

note: lua-devel python-devel and ruby-devel did not install (NOT AVAILABLE)

==========

==========

==========

Scientific Distributions

==========

==========

==========

MUST USE SEPARATE PYTHON ENV (or sandbox container)

www.anaconda.com

www.enthought.com/products/canopy

www.activestate.com/activepythong/downloads

==========

==========

==========

Libraries

==========

==========

==========

matplotlib (DONE) (needed by diskplorer above)

git clone https://github.com/matplotlib/matplotlib.git

sudo yum install freetype-devel

sudo python3 -m pip install -U pip

sudo python3 -m pip install -U matplotlib

numpy (DONE)

sudo yum install python36-numpy

pandas (DONE)

sudo python3 -m pip install -U pandas

scipy (DONE)

sudo python3 -m pip install -U scipy

beautifulsoup (DONE)

sudo yum install python36-beautifulsoup4

sympy (DONE)

sudo python3 -m pip install -U sympy

ipython (DONE)

sudo python3 -m pip install -U ipython

==========

==========

==========

Hive Data Storage

==========

==========

==========

Instructions

https://www.elastic.co/guide/en/elasticsearch/hadoop/current/hive.html

yum clean packages

yum clean metadata

https://github.com/big-data-europe/docker-hadoop

git clone https://github.com/big-data-europe/docker-hadoop

cd docker-hadoop

docker-compose up -d

yum clean all

https://github.com/big-data-europe/docker-hive

```
git clone https://github.com/big-data-europe/docker-hive

docker-compose up -d

ncdu /var

# out of space on /var/lib/docker/overlay2/

# https://stackoverflow.com/questions/30604846/docker-error-no-space-left-on-device

docker volume ls -qf dangling=true

docker volume rm $(docker volume ls -qf dangling=true)

docker system prune

docker-compose down

# Remove all images

docker rm $(docker ps -aq)

# Remove all images

docker rmi $(docker images -q)
```

==========

==========

==========

Platforms

==========

==========

==========

JuliaPro (DONE)

Version: 1.1.1.1 https://juliacomputing.com/products/juliapro.html

#

Download

https://pkg.juliacomputing.com/jpro_auth/juliapro/1041/JuliaPro-1.0.4.1_build-35.sh

sudo mkdir /opt/julia

cd /opt/julia

./JuliaPro-1.0.4.1_build-35.sh /opt/julia/

R Requirements Comments (DONE)

Version: 3.6.1 (2019-07-05)

REQUIREMENT: Compile R from source

REQUIREMENT: Use a native optimum build for R

REQUIREMENT: Build with --enable-R-shlib so R-Studio works

REQUIREMENT: Build with --enable-BLAS-shlib to enable linking to faster OpenBLAS

REQUIREMENT: Compile OpenBLAS from source with a native optimum build

REQUIREMENT: For OpenBLAS, USE_THREAD=1 and USE_OPENMP=1 options are needed

REQUIREMENT: Link R to use shared libraries: OpenBLAS and R

REQUIREMENT: Verify all of the above

Although -O1 is sufficient for this optimization with some compilers

Always use at least -O2 for overall performance, preferably -O3 -march=native -flto -fno-plt and also profile-guided optimization.

To see how the compiler sees the hardware:

gcc -march=native -mtune=native -Q --help=target

Dependencies List (DONE)

R Dependencies from YUM (DONE)

yum install libcurl-devel openssl-devel libxml2-devel

yum install libjpeg-turbo-devel readline-devel libSM-devel libICE-devel libXt-devel libXmu-devel cairo-devel

yum install libtiff-devel pango-devel

yum install pcre2

R Dependencies from RPM (DONE)

cd /opt/repo/packages/

wget mirror.centos.org/centos/7/os/x86_64/Packages/pcre2-devel-10.23-2.el7.x86_64.rpm

sudo rpm -ivh pcre2-devel-10.23-2.el7.x86_64

wget https://mirror.centos.org/centos/7/os/x86_64/Packages/pcre2-utf32-10.23-2.el7.x86_64.rpm

sudo rpm -ivh rpms/pcre2-utf32-10.23-2.el7.x86_64.rpm

wget https://rpmfind.net/linux/centos/7.6.1810/os/x86_64/Packages/texinfo-tex-5.1-5.el7.x86_64.rpm

sudo rpm -ivh texinfo-tex-5.1-5.el7.x86_64.rpm

wget mirror.centos.org/centos/7/os/x86_64/Packages/texlive-epsf-svn21461.2.7.4-43.el7.noarch.rpm

sudo rpm -ivh texlive-epsf-svn21461.2.7.4-43.el7.noarch.rpm

R Dependencies from Source (DONE)

cd /opt/repo/sourcebuilds/

git clone git://sourceware.org/git/valgrind.git

cd valgrind

./autogen.sh

./configure

make

sudo make install

R Requirement: OpenBLAS (DONE)

cd /opt/repo/sourcebuilds/

```
git clone https://github.com/xianyi/OpenBLAS.git

# https://groups.google.com/forum/#!topic/openblas-users/W6ehBvPsKTw

cd /opt/repo/sourcebuilds/OpenBLAS/

make CFLAGS='-g -O3 -march=native -flto -pipe' USE_THREAD=1 USE_OPENMP=1 2>&1 | tee
makeout.txt

# Install: Default path is /opt/OpenBLAS/

make install

# Linked Libs

sudo ln -s /opt/OpenBLAS/lib/libopenblas.so /usr/local/lib64/libRblas.so

sudo ldconfig


# R (DONE)

# https://stackoverflow.com/questions/21001388/fma3-in-gcc-how-to-enable

# https://www.avrahamadler.com/2014/04/20/r-3-1-0-openblas-speed-comparisons/

# https://stackoverflow.com/questions/5470257/how-to-see-which-flags-march-native-will-activate

cd /opt/repo/packages/

wget https://cran.r-project.org/src/base/R-3/R-3.6.1.tar.gz

cd /opt/repo/sourcebuilds/

tar xvgz /opt/repo/packages/R-3.6.1.tar.gz

cd R-3.6.1

# -march=corei7-avx -O3 -std=gnu++0x --param l1-cache-line-size=64 --param l1-cache-size=64 --param
l2-cache-size=256.

./configure CFLAGS='-g -O3 -march=native -mtune=native -flto -pipe -Wno-unused-variable -Wno-
unused-function -fPIC' CXXFLAGS='-g -O3 -march=native -mtune=native -flto -pipe -Wno-unused-variable
-Wno-unused-function -fPIC' CXX98FLAGS='-g -O3 -march=native -mtune=native -flto -pipe -Wno-
unused-variable -Wno-unused-function -fPIC' CXX11FLAGS='-g -O3 -march=native -mtune=native -flto -
pipe -Wno-unused-variable -Wno-unused-function -fPIC' CXX14FLAGS='-g -O3 -march=native -
mtune=native -flto -pipe -Wno-unused-variable -Wno-unused-function -fPIC' OBJCFLAGS='-g -O3 -
march=native -mtune=native -flto -pipe -Wno-unused-variable -Wno-unused-function -fPIC' FCFLAGS='-g
-O3 -march=native -mtune=native -flto -pipe -Wno-unused-variable -Wno-unused-function -fPIC' --
prefix=/opt/R/3.6.1 --enable-R-shlib --enable-BLAS-shlib --enable-memory-profiling --with-cairo --with-
jpeglib --with-libpng --with-libtiff --with-system-tre 2>&1 | tee configure.txt

make -j16 2>&1 | tee make.txt
```

make check

make info

make pdf

make install

make install-info

make install-pdf

R Linked Libs (DONE)

ln -s /usr/local/lib64/libopenblas.so /opt/R/3.6.1/lib64/R/lib/libRblas.so

ln -s /opt/R/3.6.1/lib64/R/lib/libR.so /usr/local/lib64/libR.so

sudo ldconfig

ldd /opt/R/3.6.1/lib64/R/bin/exec/R

linux-vdso.so.1 => (0x00007ffec5fb0000)

libR.so => /usr/local/lib64/libR.so (0x00007fcd855c2000) <<<<<<<<<<<<<<<<<<<<<<<<<<<<
Correct /usr/local/lib64

libRblas.so => /usr/local/lib64/libRblas.so (0x00007fcd84685000) <<<<<<<<<<<<<<<<<<<<<<<<<<<<
Correct /usr/local/lib64

libgomp.so.1 => /lib64/libgomp.so.1 (0x00007fcd8445f000)

libpthread.so.0 => /lib64/libpthread.so.0 (0x00007fcd84243000)

libc.so.6 => /lib64/libc.so.6 (0x00007fcd83e75000)

libgfortran.so.3 => /lib64/libgfortran.so.3 (0x00007fcd83b53000)

libm.so.6 => /lib64/libm.so.6 (0x00007fcd83851000)

libquadmath.so.0 => /lib64/libquadmath.so.0 (0x00007fcd83615000)

libreadline.so.6 => /lib64/libreadline.so.6 (0x00007fcd833cf000)

libtre.so.5 => /lib64/libtre.so.5 (0x00007fcd831bf000)

libpcre2-8.so.0 => /lib64/libpcre2-8.so.0 (0x00007fcd82f48000)

libpcre.so.1 => /lib64/libpcre.so.1 (0x00007fcd82ce6000)

```
        liblzma.so.5 => /lib64/liblzma.so.5 (0x00007fcd82ac0000)
        libbz2.so.1 => /lib64/libbz2.so.1 (0x00007fcd828b0000)
        libz.so.1 => /lib64/libz.so.1 (0x00007fcd8269a000)
        librt.so.1 => /lib64/librt.so.1 (0x00007fcd82492000)
        libdl.so.2 => /lib64/libdl.so.2 (0x00007fcd8228e000)
        libicuuc.so.50 => /lib64/libicuuc.so.50 (0x00007fcd81f15000)
        libicui18n.so.50 => /lib64/libicui18n.so.50 (0x00007fcd81b16000)
        /lib64/ld-linux-x86-64.so.2 (0x00007fcd85c93000)
        libgcc_s.so.1 => /lib64/libgcc_s.so.1 (0x00007fcd81900000)
        libtinfo.so.5 => /lib64/libtinfo.so.5 (0x00007fcd816d6000)
        libicudata.so.50 => /lib64/libicudata.so.50 (0x00007fcd80103000)
        libstdc++.so.6 => /lib64/libstdc++.so.6 (0x00007fcd7fdfc000)
LD_DEBUG=libs ldd /opt/R/3.6.1/lib64/R/bin/exec/R
# At this point, R is properly linked to OpenBLAS
# VERIFY
# Start R
/opt/R/3.6.1/bin/R
# Get R's PID
ps aux | grep R
# Verify actual linked lib (CORRECT)
lsof -p 32106 | egrep "blas"
lsof: WARNING: can't stat() fuse file system /root/.cache/doc
    Output information may be incomplete.
R       32106 jfolkers  mem    REG  253,4  29291360 121786588 /opt/OpenBLAS/lib/libopenblas_haswellp-r0.3.8.dev.so  <<<<< ONLY
# Summary Notes for R Install (DONE)
```

R and OpenBLAS are now both compiled to the native hardware

R has gcc O3 production grade optimization, and OpenBLAS is running threaded OpenMP and gcc O3 production optimization

This is the best you can get with R in 2019

However, Threads will still show 1

cat /proc/`ps aux | grep R | grep '3.6.1' | awk '{print $2}'`/status | grep Threads

R REFERENCE (DONE)

https://software.intel.com/en-us/forums/intel-math-kernel-library/topic/491299

https://michaellindon.github.io/lindonslog/linux-unix/compile-r-openblas-source-guide/index.html

https://support.rstudio.com/hc/en-us/articles/218004217-Building-R-from-source

https://www.animalgenome.org/bioinfo/resources/manuals/R/R-admin.html

https://cran.r-project.org/doc/manuals/r-patched/R-admin.html

https://stat.ethz.ch/R-manual/R-devel/library/base/html/capabilities.html

https://gist.github.com/cheuerde/8fb9fd0dc8c0eca17c16

https://colinfay.me/r-installation-administration/installing-r-under-unix-alikes.html

https://community.oracle.com/blogs/machinelearning/2016/11/08/building-r-with-intel-mkl-blas-on-linux

https://www.r-bloggers.com/compile-r-and-openblas-from-source-guide/

https://www.r-statistics.com/2012/04/speed-up-your-r-code-using-a-just-in-time-jit-compiler/

https://www.r-bloggers.com/speeding-up-r-computations-pt-ii-compiling/

http://www.onthelambda.com/2015/05/31/lessons-learned-in-high-performance-r/

https://www.openmp.org/wp-content/uploads/omp-hands-on-SC08.pdf

https://codeyarns.com/2017/11/02/how-shared-library-locations-are-found-at-runtime/

https://stackoverflow.com/questions/21730547/openblas-routine-used-from-r-rcpp-runs-only-on-a-single-core-in-linux

https://stackoverflow.com/questions/34578526/how-to-make-openblas-work-with-openmp

https://codeyarns.com/2014/01/14/how-to-fix-shared-object-file-error/

https://lonesysadmin.net/2013/02/22/error-while-loading-shared-libraries-cannot-open-shared-object-file/

BENCHMARK-25.R (DONE)

https://mac.r-project.org/benchmarks/R-benchmark-25.R

source("/home/jfolkers/RScripts/R2.R")

10 to 100 times faster over stock R

https://mac.r-project.org/benchmarks/

R Studio Server (DONE) (NOT USED DUE TO LOCAL USER ONLY—AVAILABLE IF NEEDED IN A PINCH)

wget https://download2.rstudio.org/server/centos6/x86_64/rstudio-server-rhel-1.2.1335-x86_64.rpm

sudo rpm -ivh install rstudio-server-rhel-1.2.1335-x86_64.rpm

Updating / installing...

1:rstudio-server-1.2.1335-1   ################################## [100%]

useradd: user 'rstudio-server' already exists

groupadd: group 'rstudio-server' already exists

Created symlink from /etc/systemd/system/multi-user.target.wants/rstudio-server.service to /etc/systemd/system/rstudio-server.service.

.libPaths("/opt/R/3.6.1/lib64/R/library")

R Studio (DONE)

https://www.rstudio.com/products/rstudio/download-server/

wget https://download1.rstudio.org/desktop/centos7/x86_64/rstudio-1.2.1335-x86_64.rpm

sudo rpm -ivh rstudio-1.2.1335-x86_64.rpm

R Makevars

http://r-pkgs.had.co.nz/src.html

Generally, R packages should avoid a custom Makefile. Instead, use Makevars.

Makevars is a make file that overrides the default make file generated by R

(which is located at file.path(R.home("etc"), "Makeconf")).

This allows you to take advantage of R's default behaviour

(it's over 150 lines, and battle-tested across many years and many systems, so you want to!)

while being able to set the flags you need.

The most commonly used flags are:

PKG_LIBS: Linker flags. A common use is PKG_LIBS = $(BLAS_LIBS). This allows you to use the same BLAS library as R.

PKG_CFLAGS & PKG_CXXFLAGS: C and C++ flags. Most commonly used to set define directives with -D.

PKG_CPPFLAGS: Pre-processor flags (not C++ flags!).

Most commonly used to set include directories with -I.

Any package listed in the LinkingTo field in the DESCRIPTION will be automatically included

you do not need to explicitly add it.

To set flags only on Windows, use Makevars.win. To build a Makevars with configure, use Makevars.in.

By default, R will use the system make, which is not always GNU compatible (i.e. on Solaris).

If you want to use GNU extensions (which are extremely common),

add SystemRequirements: GNU make to DESCRIPTION.

If you're not sure if you're using GNU extensions, play it safe and add it to the system requirement.

file.path(R.home("etc"), "Makeconf")

[1] "/opt/R/3.6.1/lib64/R/etc/Makeconf"

```
# R Packages (ERROR on tidymodels dependency rstanarm)

# ISSUE: rstanarm requires gcc++14 c standard to compile

# and RedHat's classic RHEL gcc 4.8.5 compiler only mostly supports gcc++14 with -std=c++1y


install.packages("devtools")

library(devtools)

install.packages("h2o")

install.packages("tidyverse")

# rstan requires gcc++14 standard

# workaround is use the prelimary gcc++14 standard called -std=c++1y that comes with the gcc version 3.8.5 on RedHat

# vi ~/.R/Makevars

# CXX_STD = CXX14

# CXX14STD = -std=c++1y

install.packages("rstan")

install.packages("bayesplot")

install.packages("colourpicker")

install.packages("dygraphs")

install.packages("gtools")

install.packages("igraph")

install.packages("lme4")

install.packages("packrat")

install.packages("rstantools")

install.packages("shinystan")

install.packages("shinythemes")

install.packages("threejs")

# rstanarm requires gcc++14 c standard

# RedHat's gcc 3.8.5 of does not fully implement gcc++14 and fails on this package install
```

```
# Therefore, rstanarm (dependency of tidymodels) must be compiled elsewhere first and copied in to /opt/R/3.6.1/lib64/R/library/

# install_github("stan-dev/rstanarm", build_vignettes = FALSE)

library(rstanarm)

install.packages("tidyposterior")

install.packages("tidymodels", repos='http://cran.us.r-project.org')


install.packages("lintr")

install.packages("plotly")

install.packages("lime")

install.packages("DataExplorer")

install.packages("parsnip")



library(devtools)

to_install <- c("arules", "catboost", "caTools", "data.table", "doParallel",

        "foreach", "forecast", "ggplot2", "h2o", "itertools",

        "lubridate", "magick", "Matrix", "monreg", "nortest","pROC", "RColorBrewer", "recommenderlab",

        "ROCR", "scatterplot3d", "stringr", "sde", "tm", "tsoutliers", "wordcloud", "xgboost", "zoo")

for (i in to_install) {

  message(paste("looking for ", i))

  if(i == "catboost" & !requireNamespace(i)) {

    devtools::install_github('catboost/catboost', subdir = 'catboost/R-package')

  } else if(i == "h2o" & !requireNamespace(i)) {

    if ("package:h2o" %in% search()) { detach("package:h2o", unload=TRUE) }

    if ("h2o" %in% rownames(installed.packages())) { remove.packages("h2o") }

    pkgs <- c("RCurl","jsonlite")

    for (pkg in pkgs) {
```

```
    if (! (pkg %in% rownames(installed.packages()))) { install.packages(pkg) }
  }
  install.packages("h2o")
 } else if (!requireNamespace(i)) {
  message(paste("     installing", i))
  install.packages(i)
 }
}
```

```
# ERROR: dependency ‘rstan’ is not available for package ‘shinystan’
# * removing ‘/opt/R/3.6.1/lib64/R/library/shinystan’
# Warning in install.packages :
#   installation of package ‘shinystan’ had non-zero exit status
# ERROR: dependencies ‘rstan’, ‘shinystan’ are not available for package ‘rstanarm’
# * removing ‘/opt/R/3.6.1/lib64/R/library/rstanarm’
# Warning in install.packages :
#   installation of package ‘rstanarm’ had non-zero exit status
# ERROR: dependency ‘rstanarm’ is not available for package ‘tidyposterior’
# * removing ‘/opt/R/3.6.1/lib64/R/library/tidyposterior’
# Warning in install.packages :
#   installation of package ‘tidyposterior’ had non-zero exit status
# ERROR: dependency ‘tidyposterior’ is not available for package ‘tidymodels’
# * removing ‘/opt/R/3.6.1/lib64/R/library/tidymodels’
# Warning in install.packages :
#   installation of package ‘tidymodels’ had non-zero exit status

install.packages("rstan")
```

* installing *source* package ‘rstan’ ...

** package ‘rstan’ successfully unpacked and MD5 sums checked

** using staged installation

** libs

Error in .shlib_internal(args) :

C++14 standard requested but CXX14 is not defined <<<<<<<<<<<<<<<< See Above. Add custom Makevars file.

* removing ‘/opt/R/3.6.1/lib64/R/library/rstan’

Warning in install.packages :

installation of package ‘rstan’ had non-zero exit status

FIX ATTEMPT #1

mkdir ~/.R

vi ~/.R/Makevars

CXX14 = g++ # or clang++ if you have that

CXX14FLAGS = -O3

install.packages("rstan")

ERROR: compilation failed for package ‘rstan’

FIX ATTEMPT #2

vi ~/.R/Makevars

CXX14 = clang++

CXX14FLAGS = -O3

install.packages("rstan")

In file included from /opt/R/3.6.1/lib64/R/library/RcppEigen/include/Eigen/Eigenvalues:58:

/opt/R/3.6.1/lib64/R/library/RcppEigen/include/Eigen/src/Core/util/ReenableStupidWarnings.h:10:30: warning: pragma diagnostic pop could not pop, no matching push [-Wunknown-pragmas]

#pragma clang diagnostic pop

In file included from chains.cpp:19:

/opt/R/3.6.1/lib64/R/library/StanHeaders/include/stan/math/prim/mat/fun/mean.hpp:24:75: error: expected '>'

Eigen::Map<const Eigen::Matrix<T, Eigen::Dynamic, 1>> m(&v[0], v.size());

^

/opt/R/3.6.1/lib64/R/library/StanHeaders/include/stan/math/prim/mat/fun/mean.hpp:25:10: error: use of undeclared identifier 'm'

return m.mean();

^

In file included from chains.cpp:20:

In file included from /opt/R/3.6.1/lib64/R/library/StanHeaders/include/stan/math/prim/mat/fun/sum.hpp:5:

/opt/R/3.6.1/lib64/R/library/StanHeaders/include/stan/math/prim/arr/fun/sum.hpp:20:49: error: expected '(' for function-style cast or type construction

return std::accumulate(xs.begin(), xs.end(), T{0});

~^

12 warnings and 3 errors generated.

make: *** [chains.o] Error 1

ERROR: compilation failed for package ‘rstan’

FIX ATTEMPT #3

vi ~/.R/Makevars

CXX_STD = CXX14

CXX14STD = -std=c++1y

FIXED: rstan installs (DONE)

https://community.rstudio.com/t/error-in-shlib-internal-args-c-14-standard-requested-but-cxx14-is-not-defined/16819

vi ~/.R/Makevars

CFLAGS = -g -O3 -march=native -mtune=native -flto -pipe -Wno-unused-variable -Wno-unused-function -fPIC -fno-plt $(LTO)

CXXFLAGS = -g -O3 -march=native -mtune=native -flto -pipe -Wno-unused-variable -Wno-unused-function -fPIC -fno-plt $(LTO)

CXX98FLAGS = -g -O3 -march=native -mtune=native -flto -pipe -Wno-unused-variable -Wno-unused-function -fPIC -fno-plt $(LTO)

CXX11FLAGS = -g -O3 -march=native -mtune=native -flto -pipe -Wno-unused-variable -Wno-unused-function -fPIC -fno-plt $(LTO)

CXX14 = g++ -std=c++1y <<<<<<<<<<<<<<<< fixes rstan but not rstanarm

CXX14FLAGS = -g -O3 -march=native -mtune=native -flto -pipe -Wno-unused-variable -Wno-unused-function -fPIC -fno-plt $(LTO)

NOTE: /tmp uses 500MB for the stan build alone with these compiler options above so make sure you have enough space

NOTE: Also make sure your home directory has enough space. Watch for .ccache sizes if you have a small home or multiple compiles.

rstanarm fails

https://unix.stackexchange.com/questions/265668/g-doesnt-recognize-standard-14-std-c14

gcc++ --version

g++ (GCC) 4.8.5 20150623 (Red Hat 4.8.5-39)

gcc 4.8.5 does not fully implement the c++14 standard and calls it by c++1y

UPGRADE NEEDED: gcc 4.9.1 is the answer because the c++14 standard is not fully implemented in c++1y

https://superuser.com/questions/1088612/upgrade-gcc-to-gcc-4-9-in-rhel-7

yum install devtoolset-3-gcc-c++

No package available

https://marksallee.wordpress.com/2016/05/03/gcc-4-9-2-install-rpm-or-source/

https://centos.pkgs.org/7/centos-sclo-rh-x86_64/devtoolset-3-gcc-4.9.2-6.el7.x86_64.rpm.html

404 Not Found

Options at this point...

1. Try downloading the binary for rstanarm and seeing if the rest of tidymodel will install...

2. Try finding the RPM sourceball for gcc4.9.2

3. Try finding the RPM sourceball for higher than 4.9.2

4. Try binary of gcc 4.9.2

5. Skipping tidymodels all together (NOT recommended)

FIX ATTEMPT #4 (SUCCESS)

Try installing rstanarm via R on linux laptop (DONE)

Compile with x86-64 architecture and generic tune

make: gfortran: Command not found <<<< igraph needs older than /usr/lib/libgfortran.so.5 that comes with gcc

ERROR: compilation failed for package 'igraph'

ERROR: dependency 'igraph' is not available for package 'threejs'

ERROR: dependency 'threejs' is not available for package 'shinystan'

ERROR: dependency 'lme4', 'shinystan' are not available for package 'rstanarm'

~/.R/Makevars

CFLAGS = -g -O3 -march=x86-64 -mtune=generic -flto -pipe -Wno-unused-variable -Wno-unused-function -fPIC -fno-plt $(LTO)

CXXFLAGS = -g -O3 -march=x86-64 -mtune=generic -flto -pipe -Wno-unused-variable -Wno-unused-function -fPIC -fno-plt $(LTO)

CXX98FLAGS = -g -O3 -march=x86-64 -mtune=generic -flto -pipe -Wno-unused-variable -Wno-unused-function -fPIC -fno-plt $(LTO)

CXX11FLAGS = -g -O3 -march=x86-64 -mtune=generic -flto -pipe -Wno-unused-variable -Wno-unused-function -fPIC -fno-plt $(LTO)

CXX14FLAGS = -g -O3 -march=x86-64 -mtune=generic -flto -pipe -Wno-unused-variable -Wno-unused-function -fPIC -fno-plt $(LTO)

sudo pacman -S gcc-fortran

R

install.packages("rstanarm")

COPY OVER RSTANARM and any dependencies (DONE)

Compare R base vs rstan vs rstanarm (DONE)

List R base packages (DONE)

base

boot

class

cluster

codetools

compiler

datasets

foreign

graphics

grDevices

grid

KernSmooth

lattice

MASS

Matrix

methods

mgcv

nlme

nnet

parallel

rpart

spatial

splines

stats

stats4

survival

tcltk

tools

translations

utils

List rstan and its dependencies (DONE)

checkmate

gridExtra

inline

loo

matrixStats

RcppEigen

rstan

StanHeaders

List rstanarm and its dependencies (in addition to the above rstan list) (DONE)

base64enc

bayesplot

colourpicker

dplyr

dygraphs

ggridges

gtools

igraph

lme4

miniUI

minqa

nloptr

packrat

plogr

rstanarm

rstantools

shinyjs

shinystan

shinythemes

threejs

tidyselect

xfun

xts

zoo

PASTE MAKEVARS FROM LAPTOP FOR GENERIC BUILD OF RSTAN AND RSTANARM (DONE)

PASTE SCREENSHOT FOR COMPARISON TO GENERATE LIST OF RSTANARM INSTALLED PACKAGES (DONE)

PORT OVER RSTANARM AND RELATED PACKAGES TO SERVER (DONE)

library(rstanarm) (ERROR)

Good news is R recognizes it as a package

even though it was compiled on another linux

Bad news is it is expecting GLIBC_2.29 on the server and it's not there

Loading required package: Rcpp

Registered S3 method overwritten by 'xts':

method   from

as.zoo.xts zoo

Error: package or namespace load failed for ‘rstanarm’ in dyn.load(file, DLLpath = DLLpath, ...):

unable to load shared object '/opt/R/3.6.1/lib64/R/library/rstanarm/libs/rstanarm.so':

/lib64/libm.so.6: version `GLIBC_2.29' not found (required by /opt/R/3.6.1/lib64/R/library/rstanarm/libs/rstanarm.so)

ISSUE: On server, libm.so.6 points to GLIBC_2.17

ls -ltrah /lib64/libm.so.6

lrwxrwxrwx. 1 root root 12 Aug 19 15:02 /lib64/libm.so.6 -> libm-2.17.so

On linux that compiled rstanarm package, libm.so.6 points to GLIBC_2.29

ls -ltrah /lib64/libm.so.6

lrwxrwxrwx. 1 root root 12 Jul 10 00:14 /lib64/libm.so.6 -> libm-2.29.so

2019-09-12 I DON'T KNOW HOW TO FIX THIS (above)

Reference:

https://unix.stackexchange.com/questions/522076/glibc-2-29-can-not-be-found-for-avrdude-even-after-downloading-it

Per those instructions, I think I have to recompile the package using a different path hardcoded into the binary

and then copy it over to the server along with the GLIBC_2.29 so file, place it in its custom dedicated path

and then it should work... R should not only recognize the package but it will find GLIBC_2.29 just for rstanarm.

So that then I can finish installing tidymodels package for R and be done.

I just don't understand the instructions to compile with the compiler path options as listed in the URL above.

I'm stumped... at the moment...

#

URL notes...

glibc consists of many pieces (200+ shared libraries) which all must match. One of the pieces is ld-linux.so.2,

and it must match libc.so.6, or you'll see the errors you are seeing.

To build an executable that will work with the new glibc, do this:

g++ main.o -o myapp ... \

-Wl,--rpath=/path/to/newglibc \

```
#    -Wl,--dynamic-linker=/path/to/newglibc/ld-linux.so.2
# The -rpath linker option will make the runtime loader search for libraries in /path/to/newglibc
# (so you wouldn't have to set LD_LIBRARY_PATH before running it),
# and the -dynamic-linker option will "bake" path to correct ld-linux.so.2 into the application.
#
# On server, I don't have a ld-linux.so.2, but I do have a /usr/lib64/ld-2.17.so
# On guest linux VM, I don't have a ld-linux.so.2, but I do have a /usr/lib/ld-2.29.so
#
# So next steps, on guest linux VM rebuild the rstanarm package with the two linker options at an invented new path...
# then copy over the binaries for GLIBC_2.29 and place them into the new path on the server
# and see if it works.



# KNIME (DONE)
# https://www.knime.com/download-installer/6/64bit
# https://download.knime.org/analytics-platform/linux/knime_4.0.1.linux.gtk.x86_64.tar.gz
cd /opt
tar xvfz knime_4.0.1.linux.gtk.x86_64.tar.gz
mv knime_4.0.1 knime
mkdir knime_workspace



# X11 (DONE)
sudo yum groupinstall "X window system"
```

sudo yum install xorg-x11-xinit (groupinstall does this above)

startx fail. gbm: failed to open any driver (search paths /usr/lib64/dri)

gdm xfce4 (DONE)

Fixes issue by populating /usr/lib64/dri/ on error message gbm: failed to open any driver

May not need this due to groupinstall "X window system" may be sufficient

sudo yum groupinstall Xfce

GNOME tools (DONE)

gnome tools

gnome-tweaks

sudo yum install gnome-tweak-tool

gconf-editor

sudo yum install gconf-editor

NoMachine (DONE)

Download

https://www.nomachine.com/AR06N00891

Edit /usr/NX/etc/node.cfg

Change from

DefaultDesktopCommand "/etc/X11/Xsession default"

Change to

DefaultDesktopCommand "/etc/gdm/Xsession 'gnome-session --session=gnome'"

/usr/NX/bin/nxserver --restart

R Studio Desktop Icon (DONE)

[Desktop Entry]

Exec=/usr/lib/rstudio/bin/rstudio %F

Icon=/opt/repo/icons/rstudio.png

Type=Application

Terminal=false

Name=RStudio

Categories=Developm6ent;IDE;

MimeType=text/x-r-source;text/x-r;text/x-R;text/x-r-doc;text/x-r-sweave;text/x-r-markdown;text/x-r-html;text/x-r-presentation;application/x-r-data;application/x-r-project;text/x-r-history;text/x-r-profile;text/x-tex;text/x-markdown;text/html;text/css;text/javascript;text/x-chdr;text/x-csrc;text/x-c++hdr;text/x-c++src;

KNIME Desktop Icon (DONE)

Edit /usr/share/applications/knime.desktop

Alternatively, you can put it here ~/.local/share/applications/knime.desktop

Icon path must NOT have quotes or dashes

[Desktop Entry]

Encoding=UTF-8

Type=Application

Comment=KNIME

Exec="/opt/knime/knime"

Icon=/opt/repo/icons/knime.png

Name=KNIME

Categories=Science;

i3 (optional)

Requirements

https://centos.pkgs.org/7/centos-x86_64/perl-Task-Weaken-1.04-6.el7.noarch.rpm.html

sudo rpm -ivh perl-Task-Weaken-1.04-6.el7.noarch.rpm

sudo yum install i3

startx

sudo startx /usr/bin/i3 -- :0

==========

==========

==========

Live Video Streaming

FFMPEG

and

Matroska Server Mk2

for

Presentations

==========

==========

==========

FFMPEG (DONE)

Reference (DONE)

https://unix.stackexchange.com/questions/104290/html-client-for-x11-ssh-forwarding

http://blog.devinrkennedy.com/2009/10/live-screencasting-using-ffmpeg.html

2018-01-06 ffserver dropped, try alternative:

https://github.com/klaxa/mkvserver_mk2

Dependencies for FFMPEG (DONE)

sudo yum install autoconf automake bzip2 bzip2-devel cmake freetype-devel gcc gcc-c++ git libtool make mercurial pkgconfig zlib-devel

x11grab is deprecated use libxcb (already installed from above steps)

RPMFUSION (SKIP)

These come from RPMFUSION but do not fulfill everything so use COMPILE STEPS

sudo yum install x264 x264-devel x264-libs x265 x265-devel x265-libs

sudo yum install lame lame-devel lame-libs

sudo yum install libvpx

sudo yum install libass libass-devel

sudo yum install vorbis

sudo yum install libxcb libxcb-devel

COMPILE STEPS (/OPT)

nasm (DONE)

curl -O -L https://www.nasm.us/pub/nasm/releasebuilds/2.14.02/nasm-2.14.02.tar.bz2

tar xjvf nasm-2.14.02.tar.bz2

cd nasm-2.14.02

./autogen.sh

./configure

make

sudo make install

yasm (DONE)

curl -O -L https://www.tortall.net/projects/yasm/releases/yasm-1.3.0.tar.gz

tar xzvf yasm-1.3.0.tar.gz

cd yasm-1.3.0

./configure

make

sudo make install

libx264 (DONE)

```
git clone --depth 1 https://code.videolan.org/videolan/x264.git
cd x264
./configure --enable-static
make
sudo make install
```

libx265 (DONE)

```
hg clone https://bitbucket.org/multicoreware/x265
cd x265/build/linux
cmake -G "Unix Makefiles" -DENABLE_SHARED:bool=off ../../source
make
sudo make install
```

aac (DONE)

```
git clone --depth 1 https://github.com/mstorsjo/fdk-aac
cd fdk-aac
autoreconf -fiv
./configure --disable-shared
make
sudo make install
```

mp3lame (DONE)

```
curl -O -L https://downloads.sourceforge.net/project/lame/lame/3.100/lame-3.100.tar.gz
tar xzvf lame-3.100.tar.gz
cd lame-3.100
./configure --disable-shared --enable-nasm
make
sudo make install
```

opus (DONE)

```
curl -O -L https://archive.mozilla.org/pub/opus/opus-1.3.1.tar.gz
tar xzvf opus-1.3.1.tar.gz
cd opus-1.3.1
./configure --disable-shared
make
sudo make install
```

vpx (DONE)

```
git clone --depth 1 https://chromium.googlesource.com/webm/libvpx.git
cd libvpx
./configure --disable-examples --disable-unit-tests --enable-vp9-highbitdepth --as=yasm
make
sudo make install
```

libass (DONE)

sudo yum install libass libass-devel

ogg (DONE)

git clone https://github.com/xiph/ogg

./autogen.sh

./configure

make

sudo make install

vorbis (DONE)

wget https://ftp.osuosl.org/pub/xiph/releases/vorbis/libvorbis-1.3.6.tar.gz

./configure

make

sudo make install

theora (DONE)

git clone https://github.com/xiph/theora

./autogen.sh

./configure

make

```
sudo make install



# ffmpeg (DONE)
# https://trac.ffmpeg.org/wiki/CompilationGuide/Centos
git clone https://github.com/ffmpeg/ffmpeg

PKG_CONFIG_PATH="/usr/local/lib/pkgconfig" ./configure \
  --prefix="$HOME/ffmpeg_build" \
  --pkg-config-flags="--static" \
  --extra-cflags="-I$HOME/ffmpeg_build/include" \
  --extra-ldflags="-L$HOME/ffmpeg_build/lib" \
  --extra-libs=-lpthread \
  --extra-libs=-lm \
  --bindir="$HOME/bin" \
  --enable-gpl \
  --enable-libass \
  --enable-libfdk_aac \
  --enable-libfreetype \
  --enable-libmp3lame \
  --enable-libopus \
  --enable-libtheora \
  --enable-libvpx \
  --enable-libx264 \
  --enable-libx265 \
  --enable-nonfree \
  --enable-libxcb
make
```

```
sudo make install
hash -d ffmpeg



# automake 1.14 (DONE)
# required by gstreamer
wget http://ftp.gnu.org/gnu/automake/automake-1.14.tar.gz
./configure
make
sudo make install



# gstreamer (DONE)
git clone https://github.com/gstreamer/gstreamer
./autogen.sh
./configure
make
sudo make install
```