



Tory: A Computer Inventory Script for HPC

M. Abdalla, K. Johnston, J. Walker



Objective:

To develop a script that gathers basic system information about a Linux based cluster and output results in various formats.

```
$ python tory -h
usage: PROG [OPTION] CMD...

optional arguments:
optional arguments:
-h, --help            show this help message and exit
-m MAX_RECORDS, --max-records MAX_RECORDS
                        Maximum number of records to show
-j, --json            convert to json
-r, --read            convert to human readable
-d, --database        send to database
-s, --show_database  show contents in database

commands:
CMD
disks                Get info on disks
network              Get info on network
cpu                  Get info on CPU
ram                  Get info on RAM
user                 Get info on users
simple                Get info on simple
package              Get list of packages or search for
one package
```

Why this is Cool?

On large systems, such as Sequoia, parts are continuously being replaced and software is constantly being added or updated due to the needs of users. This project will produce a fast and easy way to monitor these changes. Tory will give us an actual count of hardware or software.

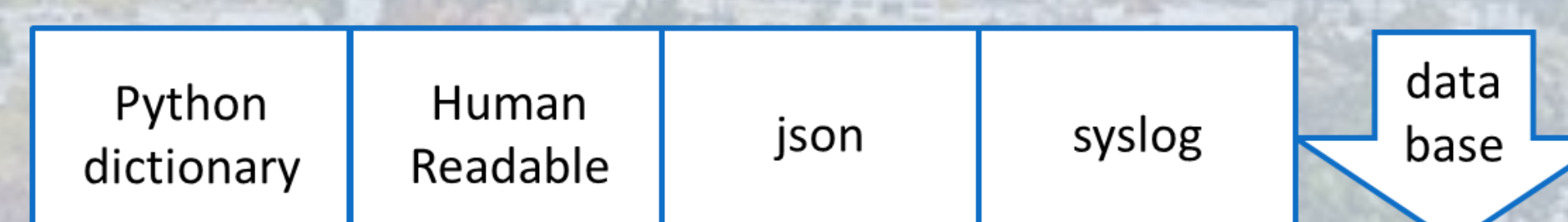
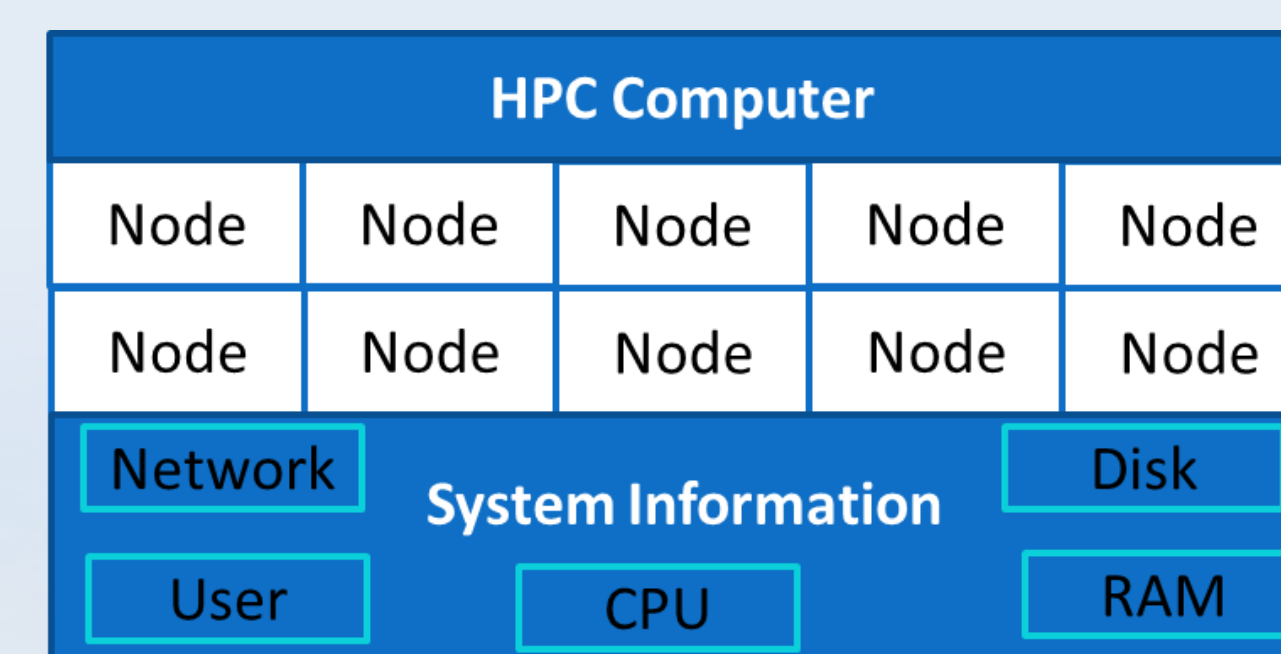


Method:

We used python modules, such as psutil, and Linux system calls to gather system information, such as network, CPU, and RAM. We then stored or displayed results in human readable format, json, and others. We wrapped up our project using the “Package Manager by Python”, pip, to package Tory and make it available for download on PyPi and the open source channel GitHub. We made Tory an interactive search tool, where users can query a specific target or request a list of items.

Links:

GitHub: <https://github.com/tbenz9/tory>
PyPi: <https://pypi.python.org/pypi/tory/0.1>



```
$ python tory -r simple
Number of CPUs: 8
Total RAM: 16.66 GB
IP Address: 192.168.80.32
Total HDD: 22.61GB
Hostname: academy-vm2
```

```
$ python tory simple
{'num_cpus': 8,
'total_ram': '16.66 GB',
'ip_address': 192.168.80.32,
'total_hdd': '22.61GB',
'hostname': 'academy-vm2'}
```

```
$ python tory -j simple
{"num_cpus": 8,
"total_ram": "16.66 GB",
"ip_address": 192.168.80.32,
"total_hdd": "22.61GB",
"hostname": "academy-vm2"}
```

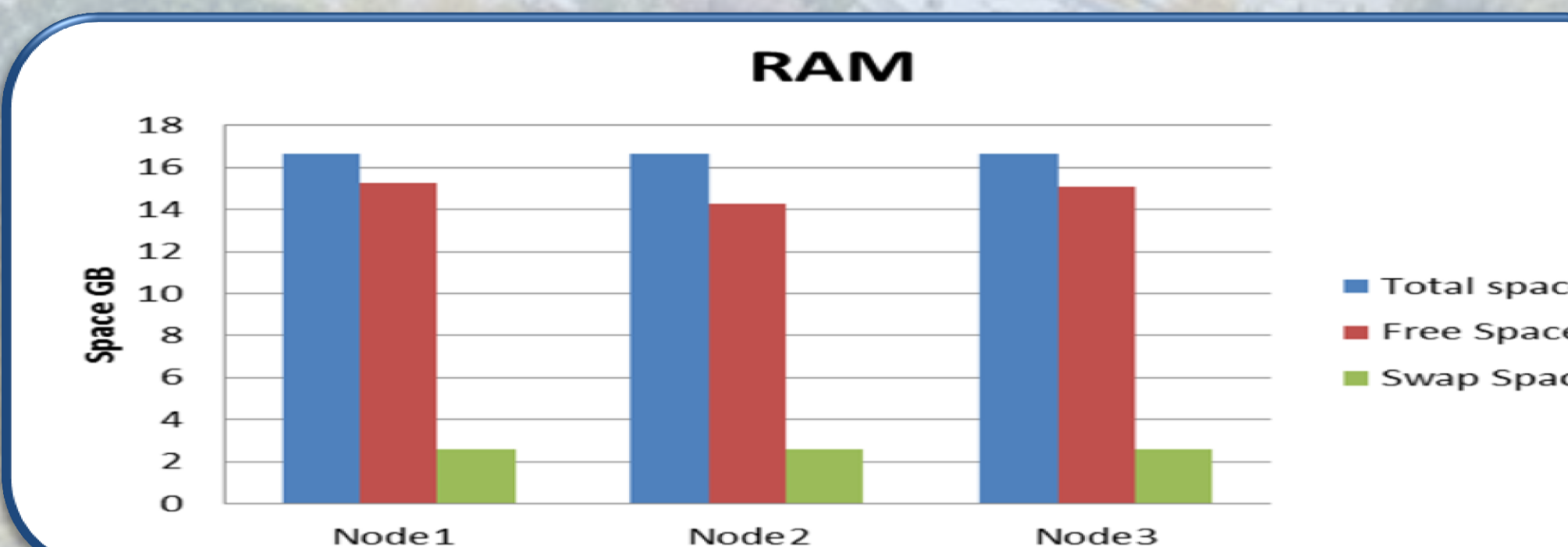
Tools:



Expected deliverable:

There is still much work for Tory, such as:

- Graphic displays of information.
- Alerts for mapping inventory scripts.
- Deploying Tory on a large scale, HPC computers.
- Make a robust Tory, that can handle any unexpected situation.
- Insure that our code follows Python Pep 8 style guide.
- Create documentation and man page.



Example of visual output for RAM information