Linux disk space consumption analysis.

Desktop distro’s have wonderful graphical disk space analysis programs such as Baobab (KDirStat), QDirStat, xdiskusage, duc, JDiskReport and with your desktop distro being connected to the internet, even if you dont already have them installed, installing them from your repositories is easy. You can quickly drill down using these treemapper programs and find the culprit for filling your disk up.

In the datacentre, things are never so easy. You have no internet access, and no local repository configured, and even if you did, you have no change control to install it on a live system, and even if you did, no GUI to view it. All you have is a production problem, a stressed out ops manager and a flashing cursor winking at you -oh and native tools.

Sure, you can use the find command to go looking for files over a certain size,

```
find ./ -type f -size +1000000M -exec ls -al {} \;
```

removing a zero and re-running as required until it starts finding something, but you’ll fight with the find command syntax for 15 minutes trying to get it to work, only to be unconvinced of the results. As good as *find* is, it’s not exactly easy trying to put together a command that does something that should be simple.

Here is a much simpler solution. Just use *du*. In particular…

```
du -h –max-depth=1
```

This will summarize the size of the top level sub-directories underneath your present working directory. You then *cd* in to the biggest one, run it again and repeat until you basically end up digging down and arriving at the largest file on disk – in my case a 32GB mysql database in */var/lib/mysql/zabbix.*

So there you go. Have a play with it and you’ll see what I mean. It’s my favourite way of finding out what’s eating all my disk space.