

Invenio System Practices

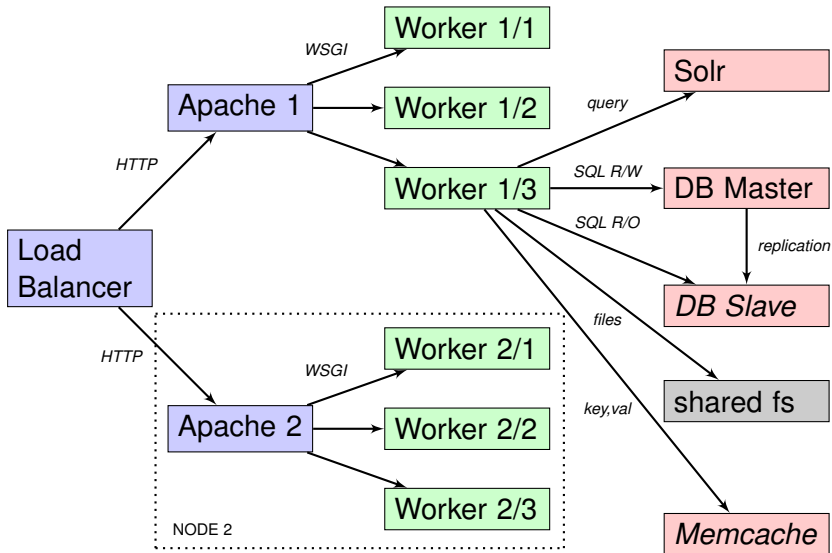
Tibor Šimko

`<tibor.simko@cern.ch>`

Department of Information Technology
CERN

Invenio User Group Workshop 2012
CERN, May 7–9 2012

Scalable Architecture



Multi-Node Architecture

- load balancer:
 - Apache mod_proxy_balancer
 - HAProxy
 - Heartbeat
- code deployment:
 - initial node replication via rsync
 - invenio-create-deploy-recipe
 - Fabric
- ...demo

■ prerequisite packages

- easy one-liner (`aptitude install`) on package-rich systems such as Debian GNU/Linux
- compiling and providing RPMs for popular server systems such as RHEL
- pip/virtualenv anyone?

■ OS tuning

- SELinux permissive mode
- TCP stack tuning

```
net.ipv4.tcp_max_syn_backlog = 4096
net.core.somaxconn = 1024
net.ipv4.tcp_fin_timeout = 30
net.ipv4.tcp_keepalive_time = 300
net.ipv4.tcp_tw_recycle = 1
net.ipv4.tcp_tw_reuse = 1
net.ipv4.ip_local_port_range = 1024 65535
```

- MPM: prefork vs worker
 - MaxClients, MaxClients, ThreadsPerChild and friends
- KeepAlive on vs off
- mod_xsendfile
- /server-status
- WSGI processes, timeouts, reloads

```
WSGIDaemonProcess processes=5 threads=1  
inactivity-timeout=3600 maximum-requests=10000
```

- ...demo

- use dedicated in-memory tmpfs for MySQL tmpdir
 - good for various JOIN and GROUP queries for tables with TEXT columns; e.g. parallel bibindexing
- use `mysqltuner` to observe and set up server variables; example

```
key_buffer = 4G
max_allowed_packet = 1G
open_files_limit = 4096
table_cache = 1024
query_cache_size = 256M
query_cache_limit = 3M
max_heap_table_size = 256M
tmp_table_size = 256M
max_allowed_packet = 1G
log-slow-queries
```

- memory and `max_connections` concerns
- ...demo

WSGI Application Profiling

- how to obtain profiles: log in as admin and use profile URL argument; e.g. /search?p=ellis&profile=t
- how to read the profile information: busy times example

```
16383 function calls in 3.069 CPU seconds
```

```
Ordered by: cumulative time
```

```
ncalls  tottime  percall  cumtime  percall  filename:lineno(function)
     1   0.000   0.000   3.069   3.069  webinterface_handler.py:362(_handler)
     1   0.000   0.000   3.067   3.067  webinterface_handler.py:184(_traverse)
     1   0.000   0.000   3.067   3.067  websearch_webinterface.py:1166(__call__)
     1   0.001   0.001   3.060   3.060  search_engine.py:4735(perform_request_search)
   435   0.006   0.000   3.022   0.007  dbquery.py:152(run_sql)
   435   0.009   0.000   3.002   0.007  cursors.py:129(execute)
   435   0.002   0.000   2.986   0.007  cursors.py:311(_query)
   435   2.968   0.007   2.980   0.007  cursors.py:273(_do_query)
     1   0.000   0.000   2.708   2.708  search_engine.py:3903(print_records)
    25   0.001   0.000   2.703   0.108  search_engine.py:4304(print_record)
    26   0.001   0.000   1.810   0.070  access_control_engine.py:39(acc_authorize_action)
    78   0.001   0.000   1.724   0.022  access_control_admin.py:1211(acc_is_user_in_role)
    75   0.002   0.000   0.622   0.008  search_engine_utils.py:26(get_fieldvalues)
    50   0.001   0.000   0.561   0.011  search_engine.py:3299(record_exists)
    25   0.000   0.000   0.271   0.011  search_engine.py:4664(call_bibformat)
    25   0.000   0.000   0.268   0.011  bibformat.py:61(format_record)
```

- ...demo

Garbage Collection, Backup, Monitoring

- garbage collection:
 - `inveniogc`
- DB backups:
 - `dbdump`
- monitoring:
 - Lemon
 - Nagios
- SMS alerts:
 - `CFG_SITE_EMERGENCY_EMAIL_ADDRESSES`
- file system:
 - AFS
 - NAS