Xavier de Poorter

DEVOPS Architecte Admin Developer

+ Technical Achievements and Skills

TRAINING

I have always trained the people around me (family, friends, collaborators) : office tools, multimedia creation tools (image, 3D, video, sound processing, music, ...) Linux server administration and network.

LINUX KERNEL

Configuration, compilation and addition of modules and patches (Vserver, grsecurity, ...).

Boot system without hard drive through the network to create light X clients.

BASH FRAMEWORK

I made a Bash FrameWork that simplifies the writing and management of BASH scripts: TurbinoBash.

It is very handy, because it is enough to put a file in a module so that it can be as early accessible for completion and execution:

Example :

```
tb module script/path
```

Example with self-completed parameters and options :

```
tb module script/path param0 param1 --option -thing=value
```

Example to activate the firewall of a VM:

```
tb pf sudo/on
```

Example to create an application and / or web space :

```
tb app sudo/create crypter-v3
```

I MADE MY OWN DOCKER 7 YEARS AGO

To solve low level problems, I created modules :

- for creating Debian packages for deploying modules, applications, and other resources.
- Backup of VMs, Applications, Users ...

Construction according to a precise recipe :

- of a chroot (useful to have the universes of compilation according to the versions of the distributions to be maintained)
- a bootable system fromscratch (physical or virtual machine) server or desktop
- to be able to apply a script on several machines at the same time (global change of a configuration file, addition / deletion of user, ...)
- for creating WEB space / application
- deployment of human users (SSH keys, SSH and Mysql login and password, Git configuration, ...)
- Mysql management (users, dumps, SQL queries, ...)

NGINX & PHP

I coded my own version of PHP-FPM (well before it came out) using TubinoBash and spawn-fcgi (but without dynamically controlling the number of PHP processes).

Again thanks to TB, I can compile all versions of PHP from 5.2 to 7.1 on UBUNTU so many version of PHP can cohabit together on the same system

I have made working URLs of the type

```
https://VM_NAME.ndd.tld/application/version/
```

thanks to the regular expression of NGINX in a single configuration file.

I think that NGINX is the WEB server to use so it is efficient, easy to configure and easy to compile to add new modules.

I use the push-stream module to manage the push to an application because it creates WEBSOCKETs super simply.

PHP PRO FRAMEWORK

I created a FrameWork PHP for an AMF / FLEX development context and automatic processing of XML and CSV streams.

It allows a distributed computing architecture thanks to a convention system and a task manager.

Templates can easily be found on different database and inheritance object templates have been implemented to resemble Postgres / Oracle behavior

The focus was on DEBUG and error management in the face of the lack of true IDE development at the time of its design.

It is intended to go only to the essential as opposed to FrameWorks to the Zend / Cake / Symphony which loads the whole universe just to lift a little finger ...

I extracted from the Zend FrameWork the code allowing me to dialogue in AMF with FLEX.

ROBUST TASK MANAGER

The system schedules PHP tasks to accomplish:

There is **n** bots :

1 by CORE on the processing VM + 1 CORE on the database

PHP + TB

A bot launches a PHP script that runs in the context of an application :

```
Class::Method(Array)
```

The manager handles subtleties such as priority, delayed execution, single channel (mandatory tasks one after the other), restart if error of the remote service provider HS ...

The task manager is also handy to avoid holding the leg to the WEB server during heavy demands of the interface. A POLLING or PUSH system warns it as soon as a processing is finished and there causes the necessary update of the interface.

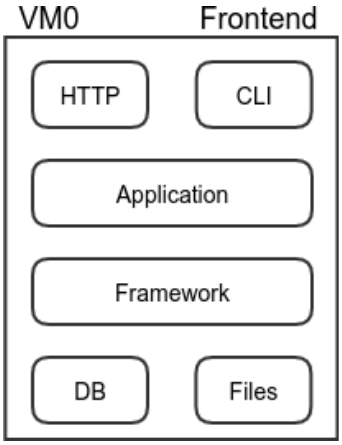
Status of a task :

- A task is inserted with the status TO_DO
- Status is set to IS_PROCESSING during processing
- The status is set to DONE if all went well during processing, ERROR otherwise
- The status can be changed to TO_CANCEL if the task is scheduled to be able to stop in progress and then switched to CANCELED once the household is done.

Currently, four large applications are working wonders with this system of tasks.

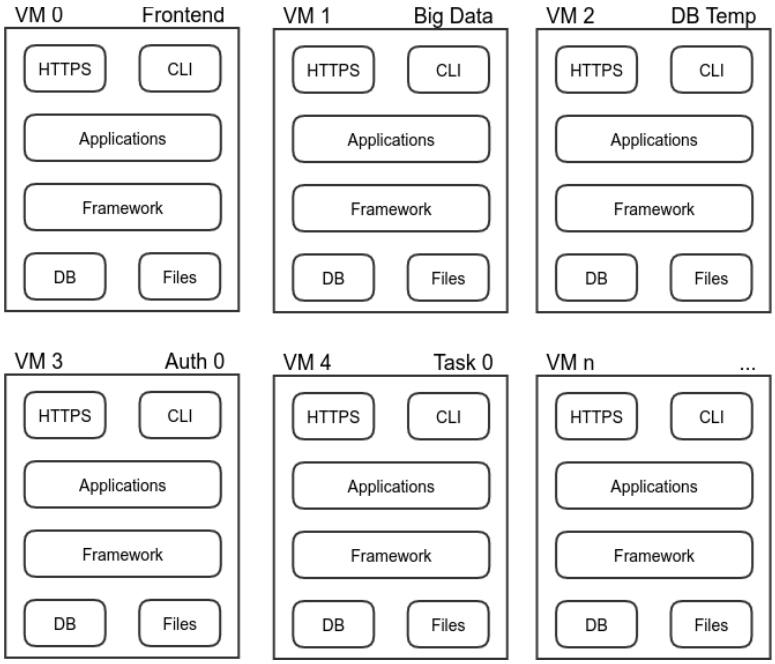
VM ORGANISATION

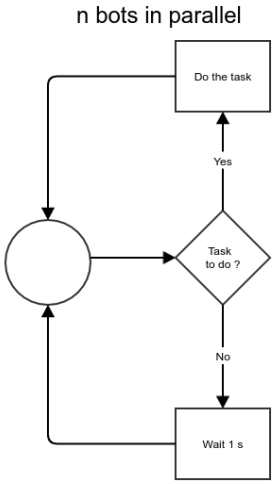
Each virtual machine is organized according to precise conventions that make an application easily duplicable from one VM to another (dev to prod) or exploded for a distributed computing model.

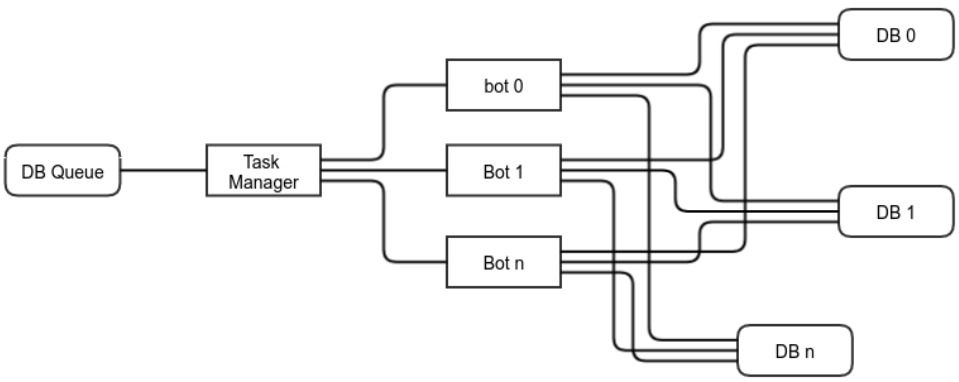


POWERFULL DISTRIBUTED CALCULATION SYSTEM

This system allows a good scalability in resources without being dependent on a cluster of databases.







LAGUAGES AND CONSIDERATIONS

It should be noted that all this logic that I developed in PHP could very well have been programmed in another language.

If I had to port these concepts, I would produce them in CSHRP, NODEJS or in RUBY depending on the constraints of performance or productivity.

DEDICATED CLOUD OVH

I maintained about sixty VMs running on these architectural models within a VMWare OVH cloud: from dev to prod through large applications. I have a reference VM that I duplicate at will.

Before the OVH cloud, I managed the virtualization with several servers and the Linux VServer + context isolator with my TURBINOBASH framework for automatic generation of VMs, moving VMs between servers, managing resources and backups.

PHP CUSTOM FRAMEWORK

I have produced a Custom FrameWork which has inherited the experience of the PRO one above but oriented for WEB sites. It goes even further to the basics and uses more a page object model and widget than a pure MVC model.

It is based on a convention system and gets all the files it needs to generate a page according to a tree inheritance: if a method is not found in the definition of a class then the engine will look for it in the paths declared in the AUTOLOAD to load only a PHP file: the object is instantiated and the PHP file is used as a method : it behaves as if it were in the object : it avoids having classes to extend and we have specialized files: it is a totally unusual approach but it works very well.

Calls to database and road management are very simple.

Also it is easy to make a GATEWAY to send and return XML, SOAP, JSON or any other format.

Calls to the GATEWAY are of type class, method, optional array of parameters for the method and optional array of parameters for the instantiation of an object of the class

I still have to produce the cache management (disk, MEMCACHED, ...)

FRAMEWORKS

I practiced CAKEPHP a lot when it came out.

I have practiced ZEND FrameWork 1 a lot and I use it mostly in support as the Mail class that I can easily integrate into my FrameWorks

I looked at YII and LARAVEL: I like them but I did not practice.

I received a training SYMFONY 2: this training allowed us to see that this framework was not adapted to our problems. It is ok for pure WEB, not necessarily powerful and especially unsuitable for big treatment.

So we opted for C # / .Net for the backend and AngularJS + Material for the frontend in place of ADOBE / FLEX

I received a training AngularJS: it's great!

MANAGEMENT TOOL ALIASES GROUPS POSTFIX / GOOGLE APPS

Thanks to the architecture presented above, I realized the engine of a business application allows my collaborators to manage the assignment of people to traditional mail aliases and to groups belonging to our domain APPS

WEBMARKETING DECISION HELP TOOL

Thanks to the architecture presented above, I realized the engine of a business application in the field of WebMarketing and Affiliation

Manage ADWORDS keyword campaigns by balancing Google / Bing costs with third-party trust earnings.

The application handles an average of thirty APIs to retrieve and mix statistics (Google, Bing, Trade Doubler, Commission junction, NetAffliliation, ...)

TAGs DEBUG

I have gained experience in site debugging that implements different tags (Analytics, Adwords, Tag Manager, ...) but that does not go up the information.

PRODUCT CATALOG EXPORT TOOL

With the architecture presented above, I have co-realized the engine of a business application in the field of catalog export products to comparators and marketplaces.

We declined this application for tour operators and we can process catalogs that go up to forty million offers: it is a question of processing files of several gigas!

To do this, I can set up servers optimized for the best possible performance: SSD in RAID 0, RAM file system for intermediate computational BDD tables, RAM file system for XML and CSV generation and for their compression, blocking of the CPUs to the maximum of their frequency (no cores in energy saving because of the awakening latency), ..

REAL TIME WEB THUMBNAIL MAKER

Long ago, he asked me to realize a THUMBNAILS generator of real-time website for a LANDING PAGE of affiliation: it was necessary to generate on the fly the images to accompany the text of partners websites.

I PLAY THE LEGO WITH OPEN SOURCE

I had taken a eight hearts server to put 8 sessions VNCs simultaneously to be able to launch in RUBY the engine Gecko of Firefox + FLASH to make the snapshot of the rendering of a Web site and to shrink it with Image Magick all orchestrated by a rudimentary task manager in PHP.

Now I would do it more simply with PHANTOMJS or directly with the WEBKIT engine in C ++.

ONLINE SHOP BLOG

I also have experience in the installation and configuration of WORDPRESS, PRESTASHOP and MAGENTO: I realized two XML export modules of the product catalog: one for PRESTASHOP and the other for MAGENTO.

OPERATING SYSTEMS

I am comfortable on Windows as well as Mac OSX or Linux to work but I have a preference to work under Linux (or else a Windows machine with a Linux VM)

Otherwise, I never wanted to put the finger in the Windows server licensing gear and I always managed to do without.

I have never had a security flaw in everything I have produced so far: I have solved the problems of automatic robots and I have only endured a few frontal attacks. I will not feel safe from a Windows server.

LANGUAGES

I can adapt a program that has been coded in any language as long as I have access to the sources

Example: I had to cohabit the patch GRSECURITY of the kernel LINUX with the functionality SUEXEC of APACHE which conflicted about the UID and GID which could become a user since root: I therefore adapted the code C of the patch and SUEXEC.

However, I avoid the functional languages (Lisp, OCaml, Haskell, ...)

COMPANY NETWORK

At first we were few and only needed an Internet BOX to work.

After a while and several collaborators in addition I set up a router LINUX with 3 BOXes Internet to be able to work serenely.

Finally, until there were up to 7 Internet BOXes and a BOX SDSL for IP telephony.

For the occasion I have mounted a gigabit bay with several SWITCHs, two NAS, video surveillance, eight BOXs, several VLANs, a guest network, a corporate network and isolated networks for the offices rented beside.

I realized this with an Ubuntu LTS, my framework BASH and IPTABLES.

Now JVWEB have FIBER and my system remains in backup in case

CONCLUSION

I am an absolutely creative and versatile architect who produces open structures on the future while sticking to the reality of the present.