Web enabling RDB datas
(migrating from interactive to web access)
History

VMS introduction in 1988

- Interactive connections (no TP monitor...)
- RDB
- All the security based on VMS security (username/passwords, identifiers/acl, same in RDB)
- Form management first done with FMS, tried DECforms... then (1990) built our own (SGE) on top of SMG$ (need to have a single interface for different terminals (VT and minitel)). All the functionalities of FMS + some more specific needs (menus, editor, etc...)
Where are we today

More VMS systems... :
- Larger network (>170 POP in Europe)
- Less VTs (750) :-(
- More PCs as user side platform (telnet emulator + web browser) (>3000)
- Web servers (intranet & Internet) on VMS and Linux
The goal

- Going from a pure VT / interactive access
- To a browser access for both Internet & intranet
Web Access

First need in 1997 for 2 Internet applications
- Spare parts availability/purchase (aircraft maintenance)
- Freight tracing with display of the Proof of Delivery

Our choice: run the web server on VMS
- More secure (no FW at that time)
- Well known environment
Purveyor

Was (almost) the only choice at that time (1997)

- Internet application => the authentication is done through the application (a single generic user on VMS)

- Performance issues:
  - CGI: process/image activation + DB attachment: SLOW
  - DLL (ie sharable image): better perf. but update issues
    - Developed a solution (called CSW) with a generic «connector» DLL and a application jacket that can run on any other system
After purveyor...

1999... No more support on purveyor...
Need another product...

■ Netscape ?
  ❑ No equivalence to DLL on the early VMS ports, then product phased out...

■ Apache ?
  ❑ Looks great (easy to sell...)
  ❑ But not ready on VMS
  ❑ Interesting enough to look at porting CSW on Apache/Linux to wait for the VMS version (3 months for a student Q2 2000)

...
After purveyor...

■ WASD?
- Developed for VMS. Fully integrated into VMS.
- All the functionalities of a standard web server (SSL, URL re-mapping, virtual servers, ISAPI, ...)
- Very good performance (caching, «CGIplus», ...)
  - Benchmarked against Apache/VMS: x3 to x100 !!
- CGIplus
  - «Resident application servers» (multiple instances if needed / load balancing...). Can run under any VMS user profile.
  - Much easier to use than CSW...
- First test: development of the «back office» (logs, db updates, etc...) for our alarm monitoring system (sydal) (Q3 2000)
New web server

Apache and WASD seem OK to setup Internet or extranet web servers (applications dedicated to the «outside world»)

But can we go further? (integrate the intranet needs)
Interactive access: need to evolve

Users want more up to date interface than VTs...
- More and more PCs with telnet emulation

Why not using the web interface as the new standard interface on the intranet? (1997...)

10
Web interface, but...

We didn't want to have a lower security level than we already had

- Need to have VMS authentication AND each user must run within its own environment (persona)
- All current security access (applications & datas) based on ACLs/Identifiers. Don't want to loose that or to duplicate these information (coherency pb)

Did not seem to be easy in a «connection-less» environment

- Evolve to a java interface to our SGE applications
SGEjava

Same approach as DECforms web connector but
■ Runs entirely on VMS (no need for another web server elsewhere)
■ Linked to a Java telnet emulator which is the key to keep the VMS security
  □ The user is authenticated through this telnet connection
  □ When the user starts a SGE application, SGEjava is launched to display the application in a specific window with the same user persona
■ 1 year man of development (started mid-1998)
■ Add new functionalities: images, ...
SGEjava

Telnet window

SGEjava window
SGEjava
A step to convergence

Can we replace the telnet session with a web browser session?

- Need VMS authentication
- The user applications have to run within the user's environment (identifiers to access the applications & datas)
- Performance is an issue (don't want to setup the all user's environment each time the user clicks on a URL... Process creation/database attachment...)

WASD + Python

WASD

- Authentication with VMS username/password on a http(s) session

- CGIplus = permanent application server launched by WASD under any VMS user environment.
  - Can be a « generic » username for an Internet application (and use the CGIplus in the same way as a TP monitor queue)
  - Or can be the authenticated user environment (one application server launched per connected user - many processes but not more than in an interactive environment)
WASD + Python

Python

- OO scripting language
- Full access to VMS runtime (sys$, lib$, etc...)
- Full access to RDB dynamic SQL
- Full access to any sharable image (as easy as writing a jacket function to transform VMS descriptors to C/C++ like parameters...)
WASD + Python

The integration:

■ Run the Python interpreter in CGIplus
■ get/process requests (python functions calls) from the browser without the penalty of image activation/DB attachment
■ This also overcome the penalty of interpreted lang./CGIplus vs compiled lang./interactive session
■ Specific developments: python library to help programmers in specific tasks:
  □ Remanent datas
  □ Web page processing (separate page design from app.)
WASD + Python architecture

Web server (WASD) + Application server

Database server
WASD + Python

Limitation:

- Must run the web server and the CGIplus on the same system.
  - This is the case on all the web servers...
  - We have chosen to run the application server within the CGIplus but we could also have built a CGIplus using any middleware to connect to a « somewhere else » application server...

- Next version will allow to split them within a cluster (should get really good performance with a FAST intra cluster connection => Galaxy)
WASD + Python
next version...
WASD perennial ?

- Directly created for/on VMS => no «cross platform noise» in it...
- Very responsive news group (=> good support)
- Evolving product : many enhancements in the next release (functional and performance)
  - Enhancement around load balancing/queues management for the CGIplus
  - $IPC for communication between the web server and CGIplus => can spread the different parts on different nodes of a cluster (takes advantage of Galaxy...)
Is it less perennial than other vendors' proprietary solutions?

- How many web products/middleware/vendors have appeared/disappeared since 1994?

We keep our engineers because they really master a wide part of the system.
First application (internal need) to validate the solution:

- Access to the CTI interface (yes, we have CTI on VMS also ;-) ) to get the log of all the calls to/from the logged in user and place calls from the web browser... (Q1 2001)
  - VMS authentication through sysuaf
  - CGIplus running in the VMS user context
First Internet application:

- Pickup orders (freight)
  - Direct access to the RDB database
  - Processing of web pages designed externally to integrate the actual data
First « commercial » application: Freight tracking/tracing

- Complete application designed from scratch
- Intranet (company employees) with authenticated/secured access
- Internet access for company customers (security based on the application)

(specifications started in march '01)
The global architecture

All the servers are running VMS...
Usefull links

WASD

Python
- http://www.python.org

Python on VMS
- http://www.decus.de/~zessin/python/

Sydal (a system to concentrate/display alarms (server on VMS&RDB/display in java))
- http://www.sydal.com