

Paper 284-28

Early Experiences with SAS® Release 9 on an OS390 Linux Platform

ir. Henri Theuwissen, SOLID Partners, Belgium

Petri LAVANDER, SOLID Partners, Luxembourg

ABSTRACT

Developing a SAS Business Intelligence application on an IBM zSeries mainframe running the Linux/390 operating system in 2002 is only possible through the version 9 of SAS.

A strong cooperation between SAS Institute, IBM and SOLID Partners enabled an early adopter implementation of Version 9.

For the fact that a new version of SAS and a new platform for the Linux operating system were used a proof of concept was required showing the data warehousing capabilities of SAS and the possibilities for a dynamic multidimensional reporting using SAS Web reporting tools.

Prerequisites or constraints for the solution were:

- Operational data in DB2 on IBM MVS.
- All required SAS modules and Web services running on Linux/390.
- Implementation before January 2003.

INTRODUCTION

The paper describes the ups and downs during the project: features, restrictions and SAS support during the project.

The project was developed using Base SAS®, SAS/AF®, SAS MDDDB Server®, SAS/EIS®, SAS/IntrNet®, SAS/Share®, SAS/Connect® and AppDev Studio™

Attendees should have a solid experience with SAS.

BUSINESS CASE**BUSINESS DESCRIPTION**

Charity companies like Amnesty International, the International Red Cross, UNICEF, Child Focus and many other help people in need. The only way for them to survive and to be able to help is by fundraising campaigns. But like within the traditional business world, all these companies are each other's competitors, in the fundraising activities.

An international charity company, with fundraising campaigns in different countries wants to streamline the fundraising activities. They want to build a Business Intelligence application allowing them to do:

- RFM analysis (recency, frequency; monetary).
- Donator segmentation, based on demographic data.
- Donator segmentation, based on history.
- Campaign response analysis.
- Calculation of the total cost of ownership of a fundraising campaign.
- Prospect analysis

In the first phase this Donator Relationship Management system will help them to better know their donators and prospects.

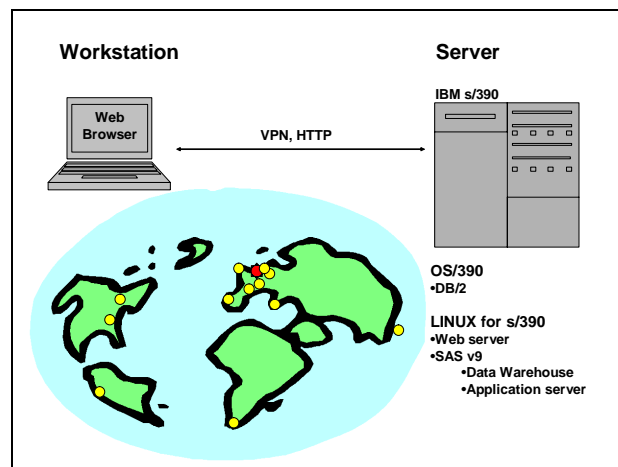
In the future this information will be used for targeted fundraising campaigns with higher return of investment.

TECHNICAL DESCRIPTION

Along with the business needs some technical requirements were set.

- The solution should allow multidimensional exploitation of data in a data warehouse.
- Users would only have a Web browser installed on their PC.
- Source data is residing in DB/2 on a MVS platform
- The complete ETL process is created with SAS tools.
- The data warehouse is stored in SAS tables.
- The whole solution must run on a Linux/s390 platform

The request of having a SAS Data Warehouse on a Linux/390 platform got us to turn towards SAS Institute and an early adopter implementation of SAS version 9. Support of IBM was already confirmed when the decision to use the Linux/390 platform was done.



For the web exploitation SAS/Intrnet, SAS/AF and AppDev Studio tools were selected.

SOFTWARE INSTALLATION

The full software installation process was a sequence of following tools:

- Linux/390
- SAS version 9
- Apache web server
- Java SDK
- Tomcat servlet container

LINUX/390

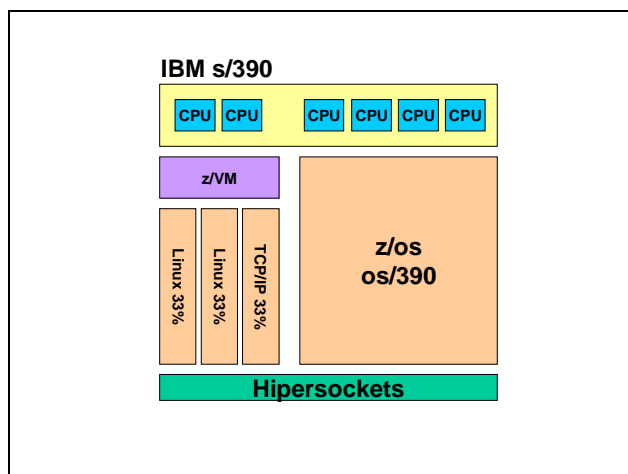
Installation

SuSe Linux was selected instead of other Linux distributions because of its better support. When the choice was done IBM RedBook covered only the installation of the SuSe Linux.

The following steps are required to successfully build Linux systems as VM guests:

- Format and label DASD (Direct Access Storage Device)
- Create user definitions
- Set up TCP/IP definitions
- Build the first Linux instance
- Create additional Linux instances
- Tailor Linux instances

The Linux installation under VM is described in details in the RedBook of IBM. This RedBook and many others related to Linux/390 are available through www.ibm.com.



In the first step the Linux IPL material has to be uploaded to a minidisk. Using FTP the Linux kernel, ramdisk and parm line file are uploaded from the distribution CD to the minidisk. After the files are copied, Linux can be booted up from the VM reader. The first Linux instance is then built using the Linux installation program YaST.

We followed the installation instructions and built the first instance through FTP. Everything seemed to work properly. Installation went through without errors and we were able to boot Linux.

Later the Samba protocol was installed to make the network devices available for Linux. As in every case the first installation of the Linux was found to be incomplete. The second try was done directly from the distribution CD. The CD was mounted through Samba and it was therefore accessible from YaST. With this new installation we noticed that the first installation wasn't complete at all. A lot more programs were automatically installed when accessing the CD directly through Samba.

Several Linux instances can run independently under VM. All the files don't need to be copied to all of the instances. It is possible to create a disk image and clone it to other Linux instances. This disk cloning is extremely fast as it is physically just a link. The only limitation is that cloned disks are in read only mode. However we installed part of the operating system and the whole SAS system on a disk image. This helped us to set up rapidly development, test and production environments.

When there is a need to store large files (bigger than 2Gb) to Linux/390 system you have to choose the Ext2 file system. After trial and error we found out that the Reiser file system doesn't support large files on Linux/390. Also the performance was found to be better with Ext2.

Linux access

Four different ways were used to access Linux/390 instances:

- Telnet session
- Remote connection from SAS version 8 running on Windows
- Xwindows with KDE
- Xwindows with VNC Tight

Telnet was found to be good for most of the Linux maintenance. However SAS cannot be used interactively through Telnet as it doesn't provide graphical interface.

At the beginning SAS was used through remote connection from SAS version 8 on Windows. This however doesn't fulfill the technical requirement that everything runs on Linux/390. Also some tasks like the SAS repository set up cannot be done through remote connection.

For the graphical interface, two different solutions were examined: Xwindows KDE and VNC tight.

Xwindows with KDE is very heavy on network and CPU resource utilization. In the first tests we made 3 simultaneous Xwindows sessions. They took 100% of the CPU of one processor. Due to the CPU load the response times were slowly as well.

VNC Tight is based on VNC server running on the Linux and a client that runs on a Windows workstation. VNC Tight is developed to have a graphical interface through a modem connection. It does not provide many features but it is a lot lighter and it was able to run SAS version 9. There were no difficulties to decide to go for VNC Tight as our graphical interface.

TCP/IP STACK

TCP/IP Stack runs under z/VM, on the same level as Linux. In general the mainframe CPU is very slow in intensive calculations. When a file transfer from MVS to Linux was ongoing the TCP/IP stack only took 30% of the available CPU. This CPU time was mainly used for the software packet checking and sizing. Implementation of the hardware checking of packets in the router and increasing in packet size radically reduced the amount of

CPU needed. After tuning efforts the CPU usage with the similar load was reduced to 10%.

SAS VERSION 9

General

A first problem showed up by analyzing the system requirements of SAS version 9:

- A Netscape web browser does not exist for Linux/390. The Konqueror browser that comes with the SuSe distribution CD was used instead.
- SAS requires PDF reader software. We were unable to find any software that would run on Linux/390.
- No postscript reader was installed.

Installation

An important difference between version 8 and version 9 of SAS, at installation time, is the fact that the installation script only shows the list of licensed products, whereas in version 8 (and before) one could install all products, even if they were not licensed.

The installation process itself is straightforward. Paths for the source and target are prompted as well as the path for the license information file. Products that will be installed are selected from the list of licensed products. We selected a minimum amount of post installation from the script. The installation of the Open Metadata Server and all SAS/Access products was skipped.

The installation can be done from the SAS distribution CDs if the Samba protocol is available. If there is enough disk space available it is also possible to upload the contents of the SAS distribution CDs and make the installation from disk. Both ways were tested and they work without errors.

Post installation

As we skipped some of the installations in the script we had to install some products later.

The spawner installation to allow non-scripted SIGNON appeared to be rather difficult. Even by following the installation instructions very carefully the spawner didn't work. By accident it was found out that spawner works if it is started with VERBOSE ON. After this was found out, the connection from SAS version 8 on Windows to SAS version 9 on Linux/390 was working well.

The SAS/Intrnet Application Dispatcher installation caused some problems as well. The dispatcher itself could not be executed at all. Most probably the compilation of the executable was done for Linux/Intel. After getting the correct compilation the installation and configuration was like in the SAS version 8.

Early Adopter 'Features'.

Within an early adopter version, you know in advance that there will be some surprises. We encountered the following items:

- Some Linux/Intel executables were found in the tests. After mailing to the support team in Cary, they were replaced by Linux/390 versions.

- The SAS repository was unreachable, due to a missing executable, called SASXBS. A customized version of this executable was delivered by the support team.
- The spawner can handle non-scripted SIGNON only when the option VERBOSE is set to ON.
- All the SAS executables had to be recompiled to allow support for large files.

WEB COMPONENTS AND SDK

Apache web server and Tomcat servlet container come with the SuSe distribution CD. Their installation appeared to be fairly simple. However we were running Tomcat as a standalone HTTP server. It would have been more complicated to link Apache and Tomcat together.

SDK version 1.4.1 can be downloaded from the IBM website. Also its installation was done without problems.

THE APPLICATION

DATA WAREHOUSE

The development of the ETL process was already started using SAS version 8 on MVS platform. These SAS programs were simply transferred to Linux/390 and all the links to external files and catalogs were changed to comply Unix naming conventions. When testing the programs with sample data all the transferred code ran without problems in SAS version 9 on Linux/390.

Fast Access to MVS Data

Data from DB/2 on MVS was read to SAS tables via flat files. The size of the complete flat file exceeded 2 Gb. Three different ways were tested to read this flat file into a SAS table

- FTP file transfer from MVS to Linux and then read to SAS.
- Read file directly from MVS using the Samba protocol.
- Read file directly from MVS using the SAS FTP filename.

The SAS FTP filename appeared to be clearly the best solution. The read process from the FTP filename was as fast as reading from the file that was already on Linux and there was no need for an extra temporary disk space.

SAS repository

For the multidimensional reports MDDBs were created to match with the data model descriptions. Repository registration caused lots of problems as one of the executables used to access the SAS repository was completely missing. After receiving the missing component the repository could be used as in version 8. MDDBs were registered and used later in the exploitation through a SAS HOLAP data provider.

As SAS will be the only tool accessing the metadata we didn't set up the Open Metadata Server. Although it is referenced in many SAS version 9 documents it is not obligatory to install or use it.

USER INTERFACE

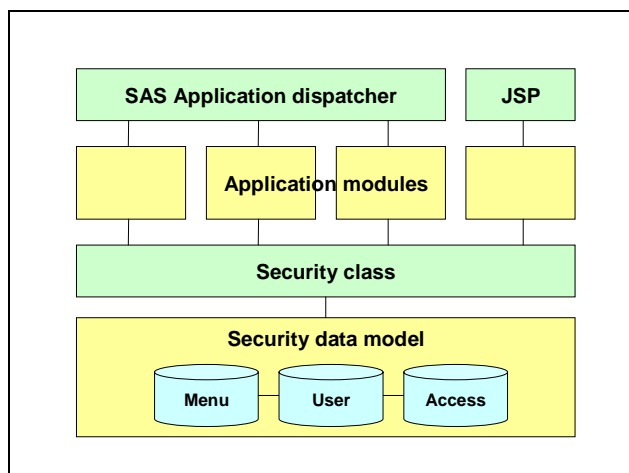
Security Model

Our solution for the user interface is built on top of the security model. Security data is stored in the set of SAS tables. It contains data about

- Users and groups
- Dynamic user specific menu definitions
- User specific data access restrictions

This data is always accessed through a security class. Application modules will use the methods of the security class to get the security data that they need.

The security class was originally developed in SAS version 8 using the Object Oriented features of SAS/AF. No modifications were required for version 9. The application catalog was just uploaded to SAS version 9 on Linux s/390 and recompiled and it was ready for use.



The modular structure of the model gives a possibility to integrate the user management with a LDAP server later, without changing the application modules of the user interface.

User Interface

The user interface is started normally through the login prompt. A user specific dynamic menu is loaded after a successful login. It is split into a hierarchical folder panel and a list panel that shows the items in selected folder. Items contain links to reports and documents.



The menu is based on a Java applet. Instead of a HTML menu a Java applet was used for its better possibilities to handle different

mouse events efficiently. Instead of having just hypertext links the menu can also contain pop menus behind the right mouse click. Both the login screen and the menu are called through the SAS/Intrnet broker.

Reporting

Two kinds of reports have been defined for the system:

- Static reporting was used ones for the most commonly used reports.
- Dynamic multidimensional reports were created for the analysis tasks.

Static reports are created on-line, based on the users' access rights. These HTML tables are created using the TABULATE procedure and ODS running on aggregated data.

For the dynamic multidimensional web reporting SAS offers two solutions.

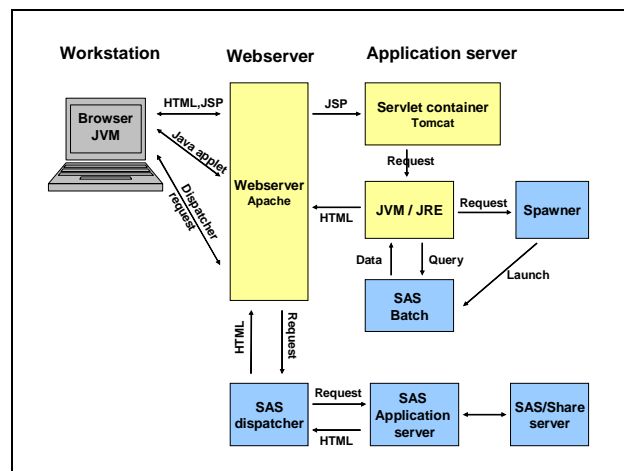
- The Multidimensional Report Viewer (MRV) that comes with SAS/Intrnet
- WebEIS tools, which are included in AppDev Studio.

Due to the business case's deadline, there was no extra time available that would have allowed some own development for the multidimensional reporting. The WebEIS solution was selected for it's easier customizability. Since Web/EIS was not found on the installation CD, a Web/EIS application was developed on PC. A multidimensional report was created with data connection that was pointing to the SAS repository on Linux/390. Then, this report was saved as JSP and just copied to the Linux, together with some AppDev Studio Java libraries.

No problems were encountered to run the JSP applications on the Linux platform. As expected there was even no need to recompile any of the Java programs.

Exploitation Architecture

The whole Exploitation Architecture is presented in the following chart. The Web server and the Application server run on Linux/390. The security model and the Data Warehouse are accessed through the SAS Application Dispatcher or through the SAS batch job running under JSP.



CONCLUSION

Although having a new version of SAS software running on a new platform we encountered less problems than we expected. All the problems that we had were solved and the proof of concept was delivered in time. For the future it looks like Linux will take itself a slice from the mainframe operating system market.

At the moment IBM mainframe processors are not the most powerful ones for the applications that require a lot of calculations. However for the applications with lots of users and 'light' queries the throughput of the mainframe architecture is still one of the best.

If your source data resides in the MVS mainframe it could be interesting to have your Data Warehouse and exploitation within the same computer in a Linux instance. The network traffic is reduced and the performance is better as the data transfer can be done internally in the mainframe. License costs can be lower as the software does not need to be bought / licensed for the whole mainframe but just for the MVS or Linux instances. Maintenance of the application servers is also easier as they can all run in the same computer. Instead of having several computers it is possible to clone Linux servers under the z/VM.

REFERENCES

IBM Redbook: Building Linux Systems Under IBM VM by Simon Williams

ACKNOWLEDGMENTS

We want to give our special thanks to Ed Connell, SAS Institute for all his support, help and hints during this project.

CONTACT INFORMATION

Contact the authors at:

ir. Henri THEUWISSEN
SOLID Partners NV
Minervastraat 14 Bis
B-1930 ZAVENTEM
Belgium
Work Phone: +32 495 54 52 53
Fax: +32 2 706 03 09
Email: Henri.Theuwissen@SOLIDPartners.be
Web: www.solidpartners.be

Petri LAVANDER
SOLID Partners Luxembourg SA
Cité Grand-Duc Jean 40
L-7233 BERELDANGE
Luxembourg
Work Phone: +352 091 42 63 42
Fax: +32 2 706 03 09
Email: Petri.Lavander@SOLIDPartners.be
Web: www.solidpartners.be

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.