The Progress Electronic Magazine

In this issue:

Did you sign up to receive this E-Zine? Send email to sauge@amduus.com to subscribe or fill out the forms at http://www.amduus.com/online/dev/ezine/EZineHome.html ! It's free! (Though donations are certainly welcome – whatever you feel is fair!)

Publisher's Statement:

This is a special edition of the Progress E-Zine. It focuses on using the Linux OS as a platform for your Progress applications. There seems to be a change of opinion in the business world regarding the use of open source applications and operating systems. Especially during a recession in the economy, alternate methods of belt tightening become far more interesting.

But, we have heard a lot of hype, and little in terms of testimonials of people who have taken the plunge. It is the purpose of this E-Zine to relate what I know and have experienced using Linux as a platform for Progress Apps.

I have been quite satisfied and I believe that other's would be too if given the opportunity.

I would like to alert everyone that I am looking for work. If any of you have tips about potential projects, I would appreciate hearing from you at sauge@amduus.com or scott_auge@yahoo.com. My resume is available here: http://www.amduus.com/Resumes/ScottAuge.html

To your success,

Scott Auge
Founder, Amduus Information Works, Inc.
sauge@amduus.com

Have you sent in a donation to keep the E-Zine available?

This E-Zine is published by Scott Auge on his own time. If you feel that the E-Zine is helping you with your skills or is valuable to your work - please send in a donation of $10.00. There is no appreciation like monetary appreciation!

Please send your donation to:

Scott Auge
925 South Wolfe Road Suite 26
Sunnyvale, CA 94086-8843

THANK YOU!

Special Article: Experiences with Progress RDBMS on Linux OS.

Written by Scott Auge sauge@amduus.com

Why did I choose Linux?

First off, let me say this is not a Linux versus Microsoft issue, and that is the last you will see that company's name in this issue. I have done development on Windows NT, however I am more in tune with the UNIX approach to creating software and that is my preferred approach.

Where I come from in terms of experiences and operating systems? I have used Progress on SCO OpenServer & UNIXWare, as well as NCR MP-RAS, HP-UX, Windows NT, and IBM's AIX. Now you know my biases.

In the beginning I started with SCO OpenServer for a character based application.

At another company (in fact, Progress Professional Services), I used Progress on a SCO UNIXWare machine. This is because UNIXWare was the only operating system for Intel processors that Webspeed supported. One of the main grief's with UNIXWare was the user licensing – if the license was for five users, only five telnets could be made to the machine. It was expensive for a product, that in my opinion, did less than Linux at the time. The user licensing issue tended to become a nuisance and an expense that Linux does not have.

I had also done work on AIX machines, and while nice, they definitely were out of my personal price range!

So, in order to work in and continue to be proficient in a UNIX environment, I settled on Linux. I chose the UNIX environment because most high paying jobs are with UNIX platforms. After all, the machines that generate and track millions of dollars of revenue generally are UNIX based and that kind of money gets spread around to the developers! Also, becoming functional on Linux means one becomes functional on HP-UX, AIX, Solaris, and a wide range of other UNIX flavored operating systems. The skills are easily transferable between all these operating systems.

I believe there definitely is a future with Linux. Many high profile companies are switching over to use of the Linux operating system. Companies such as IBM, Dell, and Compaq are making Linux based machines. Such companies as Amazon.com are moving their operations onto such

machines at a savings of millions of dollars and simplified license management – and google has become one of the top web search engines based on Linux machines. I believe Linux is here to stay.

There is talk about no one to support the operating system, no one vendor to point the finger at when trouble looms in the data center. This is incorrect, as there are many companies who will support Linux machines. In fact, the main revenue base for Red Hat is their support mechanisms – to handle questions about how to accomplish something, or to receive updates on the software. Add to this line of commercial companies a sea of programmers coming out of colleges familiar with the Linux operating system, as well many programmers from varied parts of the world. Most hard-core programmers have always known about Linux and have dabbled with it. Personally, I have been aware of Linux and working with it since 1991. (Believe me – a LOT has changed and improved since those days!)

Add to that, the Linux operating system is extremely stable. The rack mounted server shown below has been online for over 110 days without a reboot. I have had other servers running previous versions of Linux run for 300 days or so before reboot (and that is because I was packing them up for the next job move.)

Many database vendors have ported their DBs to Linux, including IBM with DB2 product and Oracle with their RDBMS product. In keeping with the competition (and I suspect a lot of "suggestions" from users), Progress ported their RDBMS and development tools to the Linux OS.

So in short, it comes down to this. Very large companies are pulling for Linux in terms of hardware, very large companies are available for support of Linux, as well a multitude of programmers, and many large database vendors feel Linux is safe enough for them to port their wares (and reputations) on that OS.

But is it really safe to count on this OS?

The machines available

Linux machines have reached past the patched and bail-wire contraptions of hobbyists to professional grade equipment. Just recently IBM has announced Linux distributions for their range of computing equipment – right up to the S/390 mainframe. There are many outfits building Linux based machines that are of professional quality (such as Dell, Inc., IBM, as well as lesser known outfits like Penguin Computing, Inc.)

Rack Mounted Linux Server running Red Hat Distribution

Pictured are the server and workstation that I have used over time which have given me the experiences to write about them in this article. These machines are designed specifically for Linux and are made out of components able to withstand hours and hours of use.

Designed and warranted by computer manufacturers, Linux hardware is certainly ready for prime time use, and I wanted to make use of such a server. It was comforting to know that at least in terms of hardware, they were as good as any other Intel based server.

Linux Work Station running Red Hat Distribution

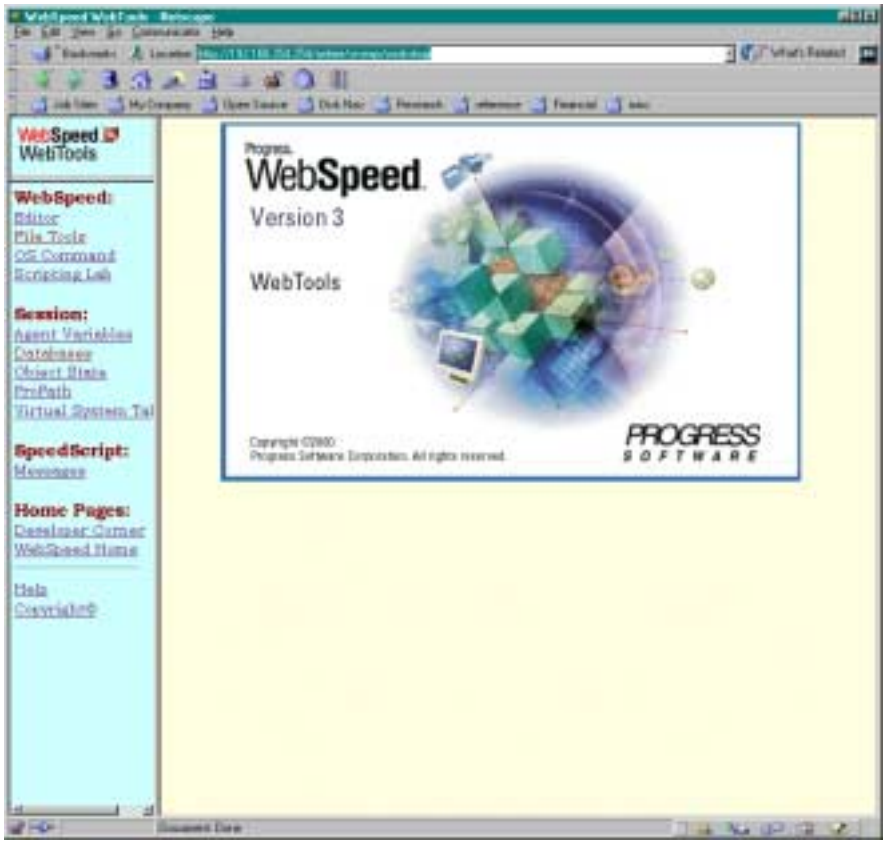
Progress under Linux machines

Once I decided that the hardware and operating system were as risky as any other OS, the question becomes how risky is it to use Progress on Linux? Would it be different to use? Would there be performance considerations? Would it fall over and leave my data in bits and bytes spread randomly around the disk? Would installing it be a b***h (to be ever so blunt)?

Lets examine the use of Progress on a Linux computer….

Webspeed for Linux

Using Webspeed on a Linux computer is no different from using Webspeed on a UNIXWare or AIX computer. (I have not used Webspeed on the other OS's supported by it, but I would be confident that it is no different.)



Webspeed on a Linux computer accessed by Windows Browser

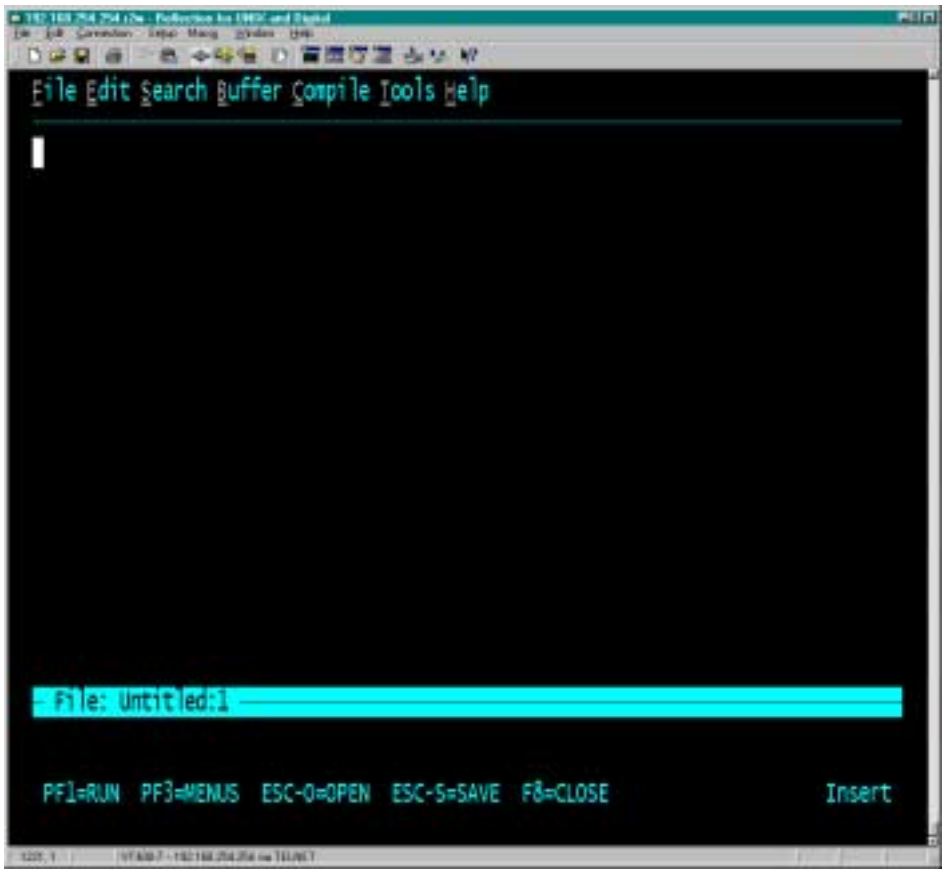
As can be seen from above, accessing Webspeed on a Linux computer from a windows computer is no different than that of the other operating systems. It still has the same functionality and look and feel.

Webspeed on a Linux computer accessed by Linux Browser

Using Webspeed on the native interface for Linux works just like the Windows interface because in the end, browser based applications really are OS independent.

Character client for Linux

Running a character client on Linux achieves the same look and functionality as on other UNIX systems. Running a VT220 terminal emulator (Reflection from www.wrq.com) had me up and running right out of the box. Sitting on top of the Linux OS, one would little know that it was any different from HP-UX, UNIXWare, MP-RAS or AIX.



Character client on a Linux computer accessed by Windows Telnet

The installation

Seeing that Progress on Linux is like Progress on any other UNIX OS in the preceding section, the question becomes – does it install as easily?

One does receive a package of media and installation instructions from Progress for the Linux version of Progress. It comes with a couple of books as well as release notes. These documents are as clear as ever for installation of the product onto the machine. If you have never done it before, you will likely need to take a couple of test runs at it.

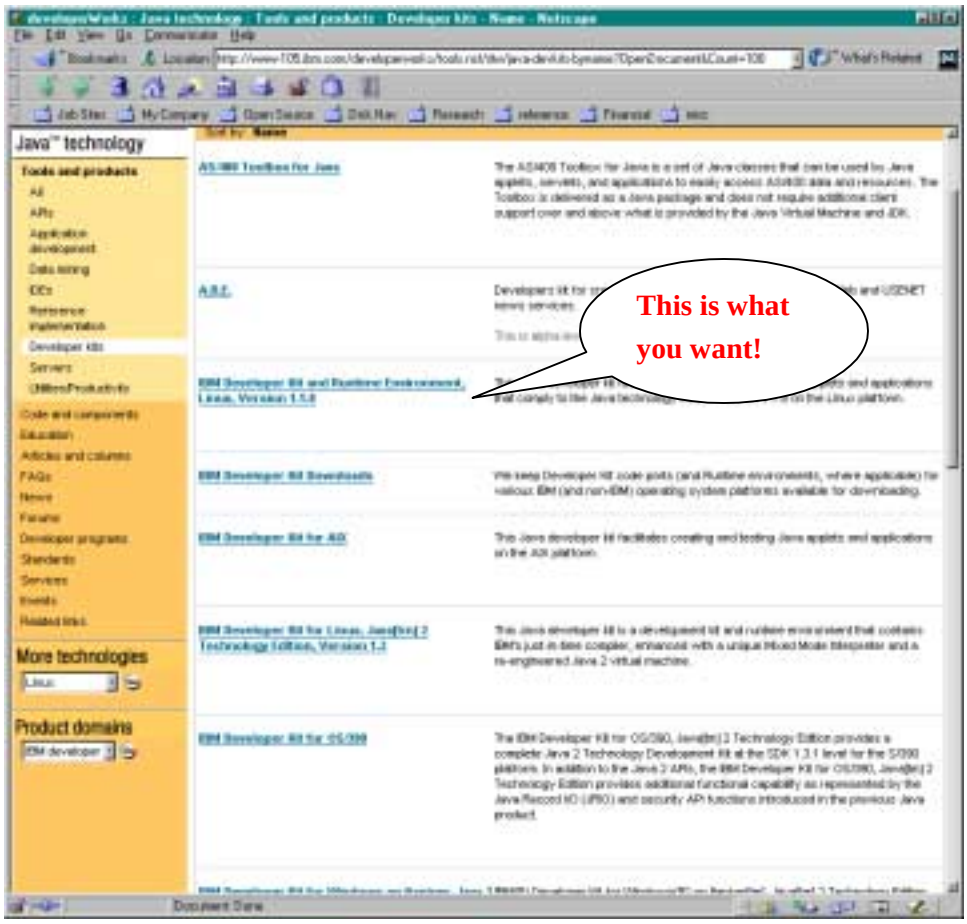
One does need at least Version 6.2 of the Red Hat distribution of Linux. (I have heard of others getting it to work on other distributions with little to no difference in steps.)

Installation media for Linux computers

Does the tools install with the same simplicity as other OS's?

Well, there are a couple of extra steps….

First off, Progress has become reliant on java. There are plenty of java JDK and JRE's available, but it is best to go with the version noted by Progress. Progress recommends the version from IBM that is freely available by download. Off the top of one's head, this might seem like a risk – what happens if IBM decides to stop working on Linux? The short answer is "Believe me, this is not going to happen! IBM has found something to stick it to Microsoft and they are going to go all the way with it."



Finding the JDK needed for Java on Linux as expected by Progress' tools

People (including myself) have also had pleasant experiences with the BlackDown.org version of JDK 1.1.8. This can be found at www.blackdown.org which will lead you to some FTP sites where one can download the tools.

Blackdown.org web site where a 1.1.8 version can be downloaded also

Note that using the Blackdown.org version is outside the parameters recommended by Progress. I am only mentioning it, because it does work as far as I can tell (over 100 days of use.)

One should install the JDKs in `/usr/local/jdk1.1.8` or as instructed in the JDK's documentation.

Once the java JDK/JRE is installed, one can use the installation program to install Progress. The installation program will not allow you to install progress until the Java portion has been installed. So if you are hoping to skip over that step, don't waste your time.

If you wish to change your java environment in the future, you will want to modify the `/usr/dlc/bin/java_env` file to locate the directory. This can allow you to switch around JDK distributions if you are encountering troubles. (I did not have troubles, but did experiment.)

Installation of the product there after is documented and is the usual business of installing on a UNIX computer.

Performance considerations

So, once the questions of reliable hardware and reliable OS have been answered – the question becomes is Progress on Linux reliable?

The happy answer is yes. This is after over 100 days of continuous uptime all the utilities worked as expected (those utilities that are commonly used in a production environment – such as clients, servers, back up tools, analysis tools, etc.)

The largest DB I have run on the system was 87 MB and that was just to see how well it handled. Creation was a bit slow, but creating any 87MB file is going to take some time, reads and deletes were handled in the usual amount of time once the file was created.

Is it fast? I have not really run any standard benchmarks, but can offer this information:

Server:

128MB RAM
20GB Harddrive (Quantum Fireball ATA/66 connection)
1 Pentium III 833MHz

Scenerio 1:

Load of 1.04MB file divided into 42,000 records. Loaded in under 2 sec.

Scenerio 2:

More oriented towards benchmarking in a production environment.

Size of record: 256 bytes
Total number of records manipulated: 300,000

(Numbers are client server)

Created	Found	Deleted	Recs Manipulated/Sec

4	34	1	7692
1	33	0	8823
2	33	1	8333
2	33	0	8571

Here is the source code of the test, one can fiddle with it to get ideas of what happens.

```
/* Benchmark.p
 * Scott Auge
 */

/* RCS Stuff */

DEF VAR RCSVersion AS CHARACTER INIT "$Header:
/home/db/benchmark/src/RCS/benchmark.p,v 1.5 2001/01/17 09:31:25 root Exp sauge
$" NO-UNDO.

/* Benchmark my new VA Linux 420 WorkStation with 128MB RAM and 833 P-III */

DEF VAR MaxRecords  AS INTEGER NO-UNDO.
DEF VAR CurRecord   AS INTEGER NO-UNDO.

DEF VAR StartTimeInSec AS INTEGER NO-UNDO.
DEF VAR EndTimeInSec   AS INTEGER NO-UNDO.

DEF VAR TimeToCreate AS INTEGER NO-UNDO.
DEF VAR TimeToFind   AS INTEGER NO-UNDO.
DEF VAR TimeToDelete AS INTEGER NO-UNDO.
DEF VAR TimeTotal    AS INTEGER NO-UNDO.

DEF VAR i            AS INTEGER NO-UNDO.
DEF VAR RecordData   AS CHARACTER NO-UNDO.

/* Clean out the DB */

FOR EACH BenchMark EXCLUSIVE-LOCK:
  DELETE BenchMark.
END.

/* We need to make the record stored more like that in an application, so */
/* lets assume we need 256 bytes in a record.  This is a fair number of   */
/* bytes for a common application to fiddle with over an average.         */
```

```
ASSIGN RecordData = "".
DO i = 1 TO 1024:

  ASSIGN RecordData = RecordData + "A".

END.

/* First test - make a lot of records.  Using a pre-existing DB - hence the */
/* odd named tables.                                                        */

ASSIGN StartTimeInSec = TIME.

MESSAGE "Creating Records".
PAUSE 0.

ASSIGN
CurRecord = 1
MaxRecords = 100000
TimeToCreate = ETIME.

DO WHILE CurRecord < MaxRecords TRANSACTION:

  CREATE BenchMark.

  ASSIGN
  BenchMark.BenchMarkSeq = STRING(CurRecord)
  BenchMark.Data = RecordData.

  ASSIGN
  CurRecord = CurRecord + 1.

END.

ASSIGN
TimeToCreate = ETIME - TimeToCreate.

/* Lets randomly find the records for the number of records made.           */
/* That should simulate a pretty hefty load bouncing all over the DB of     */
/* users looking for different stuff on the DB.                             */

MESSAGE "Looking Up Records".
PAUSE 0.

ASSIGN
CurRecord = 0
TimeToFind = ETIME.
```

```
DO WHILE CurRecord < MaxRecords TRANSACTION:

  FIND BenchMark NO-LOCK
  WHERE BenchMark.BenchMarkSeq = STRING(RANDOM (1, MaxRecords))
  NO-ERROR.

  ASSIGN
  CurRecord = CurRecord + 1.

END.

ASSIGN
TimeToFind = ETIME - TimeToFind.

/* Now lets blow em all away and see how long that takes.                    */

MESSAGE "Deleting Records".
PAUSE 0.

ASSIGN
TimeToDelete = ETIME.

FOR EACH BenchMark:

  DELETE BenchMark.

END.

ASSIGN
TimeToDelete = ETIME - TimeToDelete.

ASSIGN EndTimeInSec = TIME.

BELL.

DISP "Time To Create: " TimeToCreate SKIP
     "Time To Delete: " TimeToDelete SKIP
     "Time To Find  : " TimeToFind SKIP
     "Time In Secs  : " EndTimeInSec - StartTimeInSec.
```

And the DF:

```
UPDATE DATABASE "?"

ADD TABLE "Benchmark"
```

```
  AREA "Schema Area"
  LABEL "BenchMark"
  DUMP-NAME "benchmar"

ADD FIELD "BenchMarkSeq" OF "Benchmark" AS character
  FORMAT "x(8)"
  INITIAL ""
  LABEL "BenchMarkSeq"
  POSITION 2
  SQL-WIDTH 16
  COLUMN-LABEL "BenchMarkSeqP"
  ORDER 10

ADD FIELD "Data" OF "Benchmark" AS character
  FORMAT "x(8)"
  INITIAL ""
  LABEL "Data"
  POSITION 3
  SQL-WIDTH 16
  COLUMN-LABEL "Data"
  ORDER 20

ADD FIELD "Count" OF "Benchmark" AS integer
  FORMAT "->,>>>,>>9"
  INITIAL "0"
  LABEL "Count"
  POSITION 4
  SQL-WIDTH 4
  COLUMN-LABEL "Count"
  ORDER 30

ADD INDEX "Count" ON "Benchmark"
  AREA "Schema Area"
  UNIQUE
  PRIMARY
  INDEX-FIELD "Count" ASCENDING

.
PSC
cpstream=ISO8859-1
.
0000000706
```

My one and only crash! (So far….)

Only once did the database come crashing down, and that was more my fault than Progress or the operating system. The scenario is this: while sending out this E-Zine to over 600 readers, my throttle for sending mail was to low, and the OS used up all it's file descriptors, thereby crashing progress. What is interesting, is that the OS did not go down, but quite a few programs attempting to start disk accesses did. Upon restart, the programs behaved themselves as expected, and the Progress DB made a complete recovery.

About the author: Scott Auge is the founder of Amduus Information Works, Inc. He has been programming in the Progress environment since 1994. His works have included E-Business initiatives and focuses on web applications on UNIX platforms.
sauge@amduus.com