

## **Experience with Linux Cluster: Preliminary Report [Apr-2004]**

**J.Hamilton, Met Éireann, Glasnevin, Dublin, Ireland.**

### **Introduction**

Met Éireann has been running various versions of Hirlam for the last several years. Originally we ran on an SGI shared memory system [with 6 processors] but, 3 years ago, we bought an IBM RS/6000SP MPI system with 9 nodes each with 4 processors sharing 2 Gigabytes of memory [*i.e.* a total of 36 CPU's and 18 Gigabytes of memory]. We are currently running version 5.0.1 of the Hirlam forecast model, along with the Hirlam 3DVAR analysis scheme, both of which support MPI.

There have been a number of experiments, within the Hirlam community, involving running Hirlam on a Linux cluster using MPI. In particular, the operational model at SMHI now runs on a custom-built Linux cluster [designed by the Swedish National Supercomputing Centre, NSC, at Linköping]. Consequently, we decided to purchase a small cluster to investigate some of the options available and to gain experience of managing such a configuration.

Our investigation of clusters involved discussions with various manufacturers, attendance at two cluster conferences in Linköping and very helpful discussions with Lars Meuller of SMHI, Norrköping, and Niclas Andersson and Torgny Faxen both at NSC, Linköping. We are grateful for their help and support.

We issued a tender document in Sep-2003 and had a number of responses. After evaluating all the proposals we decided to purchase the system proposed by Dell.

### **Details of Hardware / Software**

The Dell cluster we purchased consists of 7 rack mounted nodes (*i.e.* 1 master node and 6 compute nodes). All nodes are Dell Powerededge 1750 nodes. The master node has dual 2.8GHz Xeon processors; each compute node has dual 3.2GHz Xeon processors. The master node has 4 Gigabytes of ECC DDR RAM; each compute node has 2 Gigabytes.

The compute nodes are connected as a two-dimensional torus via Dolphin SCI cards. Each compute node has a 4 Port Dolphin SCI HBA card for compute node interconnect.

Software consists of:

- Operating system: Redhat ES 3.0 on the master node, WS 3.0 on the compute nodes;
- Networking software: Scali MPI connect, Scali TCP connect, Scali Manage;
- Compilers: PGI cluster development kit and Intel Fortran compiler.

A full description of the system is given in Appendix I.

## **Installation of Cluster: Initial Experiences**

The cluster arrived in mid December 2003 and the Dolphin network cards, Redhat operating systems and Scali software were installed by Scali. There were some incompatibility problems between the, then current, version of the Scali software and the just released version of the Redhat software. Also, there were delays in delivery of the PGI and Intel compilers. Because of these various problems, the system was configured with a beta version of the Scali software and with temporary software licences for both the Scali software and the PGI compilers. In fact, the final version of the Scali software was not installed until March 2004.

The use of temporary licences meant that (a) the Scali licence has to be reinstalled every few weeks and (b) the PGI licence not only had to be reinstalled every few weeks but also all executable programs and libraries had to be recompiled and rebuilt. This slowed down development and testing of Hirlam, but in fact, it did not have a major impact on progress. However, since March 2004, we have been running with permanent licences for all the software.

A 'stripped-down' version of Hirlam was installed in late December / early January and the system has gradually been upgraded to be almost identical to the operational system running on the IBM RS/6000SP system. The system has proved to be very reliable and has failed only once in nearly four months [there was a problem with the loss of NFS mounts on that occasion].

## **Installing the Hirlam Forecast Model**

Version 5.0.1 of the Hirlam forecast model runs on the IBM RS/6000SP. This version has been modified locally, at Met Éireann, to produce extra output and to perform extra post-processing [such as calculation of the freezing level]. This fully modified version has now been installed on the cluster and has been running for about two months using MPI and all the compute nodes. It has been tested using both the Gigabit Ethernet and the Dolphin SCI interconnects. The system has been compiled using the PGI compilers; experiments have already started with the Intel compilers. Appendix II gives details of the compiler flags *etc.* used with the PGI compilers.

## **Hirlam Forecast Model: Comparative Timings**

The following are some results for the operational Hirlam model on the IBM RS/6000SP system [with 9 nodes, each with four 375 Mhz Power3-II CPU's]. The timings are for a 48-hour forecast. The grid is 438 x 284 with 31 levels and a timestep of 300 seconds.

### IBM RS/6000SP

```
-rw-r--r-- 1 hirlam res 32580000 Apr 12 02:18 fc200404120000
-rw-r--r-- 1 hirlam res 33588000 Apr 12 03:20 fc200404120048
IBM Forecast took ... 62 mins
-rw-r--r-- 1 hirlam res 32568000 Apr 12 08:21 fc200404120600
-rw-r--r-- 1 hirlam res 33444000 Apr 12 09:24 fc200404120648
IBM Forecast took ... 63 mins
-rw-r--r-- 1 hirlam res 32760000 Apr 12 14:19 fc200404121200
-rw-r--r-- 1 hirlam res 33420000 Apr 12 15:23 fc200404121248
IBM Forecast took ... 64 mins
-rw-r--r-- 1 hirlam res 32532000 Apr 12 20:20 fc200404121800
-rw-r--r-- 1 hirlam res 33480000 Apr 12 21:23 fc200404121848
IBM Forecast took ... 63 mins
-rw-r--r-- 1 hirlam res 32652000 Apr 13 02:19 fc200404130000
-rw-r--r-- 1 hirlam res 33504000 Apr 13 03:23 fc200404130048
IBM Forecast took ... 64 mins
-rw-r--r-- 1 hirlam res 32700000 Apr 13 08:19 fc200404130600
-rw-r--r-- 1 hirlam res 33576000 Apr 13 09:21 fc200404130648
IBM Forecast took ... 64 mins
```

The following results are for the same version of the model on the same grid running on the Dell Linux cluster with 6 nodes each with two 3.2Ghz Xeon CPU's.

### Dell Linux Cluster

```
-rw-r--r-- 1 hirlam hirlam 32676000 Apr 12 02:32 fc200404120000
-rw-r--r-- 1 hirlam hirlam 33552000 Apr 12 03:54 fc200404120048
Cluster Forecast took ... 82 mins
-rw-r--r-- 1 hirlam hirlam 32568000 Apr 12 08:34 fc200404120600
-rw-r--r-- 1 hirlam hirlam 33420000 Apr 12 09:54 fc200404120648
Cluster Forecast took ... 80 mins
-rw-r--r-- 1 hirlam hirlam 32664000 Apr 12 14:34 fc200404121200
-rw-r--r-- 1 hirlam hirlam 33444000 Apr 12 15:53 fc200404121248
Cluster Forecast took ... 79 mins
-rw-r--r-- 1 hirlam hirlam 32604000 Apr 12 20:32 fc200404121800
-rw-r--r-- 1 hirlam hirlam 33516000 Apr 12 21:51 fc200404121848
Cluster Forecast took ... 79 mins
-rw-r--r-- 1 hirlam hirlam 32688000 Apr 13 02:29 fc200404130000
-rw-r--r-- 1 hirlam hirlam 33552000 Apr 13 03:48 fc200404130048
Cluster Forecast took ... 79 mins
-rw-r--r-- 1 hirlam hirlam 32652000 Apr 13 08:28 fc200404130600
-rw-r--r-- 1 hirlam hirlam 33576000 Apr 13 09:46 fc200404130648
Cluster Forecast took ... 78 mins
```

Comparing these 6 runs we see that the IBM takes between 62 and 64 mins per run; the cluster takes between 78 and 82 mins. Alternatively, the mean time for the IBM runs is

63.3 mins, for the cluster 79.5. Using these latter figures gives the result  $(79.5/63.3) = 1.26$  or  $(63.3/79.5) = 0.80$  meaning the IBM system is 1.26 times as fast as the cluster or, alternatively, the cluster is 0.80 times as fast as the IBM system.

Comparing individual processors [36 on the IBM, 12 on the cluster] we note that  $((63.3*36)/(79.5*12)) = 2.4$  or  $((79.5*12)/(63.3*36)) = 0.4$  meaning that a cluster processor is 2.4 times as fast as an IBM processor, or alternatively, an IBM processor is 0.4 times as fast as a cluster processor.

Note that these results are only for this particular configuration of the forecast model with the particular hardware and software system used. Details of the compiler flags are given In Appendix II.

### **Installing the Hirlam 3DVAR Analysis Scheme**

The Hirlam OI [Optimal Interpolation] analysis scheme does not work with MPI, so to get the system running on all the nodes, we installed the 3DVAR version of the analysis which is MPI compatible. The operational version, running on the IBM RS/6000SP system, has been optimised in various ways to make use of special IBM software libraries: these optimisations had to be removed [*via* flags in the makefile] for compiling with the cluster version. Also, a more strict version of MPI must be used [controlled *via* the flag MPI\_STRICT]. See Appendix II for full details.

### **Hirlam 3DVAR Analysis: Comparative Timings**

The following results are for the operational system on the IBM RS/6000SP system using version 5.0.0 of 3DVAR. [Note that feedback files are generated].

#### **IBM RS/6000SP**

```
-rw-r--r-- 1 hirlam res      4479032 Apr 12 02:03 ob2004041200
-rw-r--r-- 1 hirlam res      32580000 Apr 12 02:18 fc200404120000
IBM Analysis took ... 15 mins
-rw-r--r-- 1 hirlam res      1835832 Apr 12 08:04 ob2004041206
-rw-r--r-- 1 hirlam res      32568000 Apr 12 08:21 fc200404120600
IBM Analysis took ... 15 mins
-rw-r--r-- 1 hirlam res      3742756 Apr 12 14:03 ob2004041212
-rw-r--r-- 1 hirlam res      32760000 Apr 12 14:19 fc200404121200
IBM Analysis took ... 17 mins
-rw-r--r-- 1 hirlam res      4898768 Apr 12 20:03 ob2004041218
-rw-r--r-- 1 hirlam res      32532000 Apr 12 20:20 fc200404121800
IBM Analysis took ... 17 mins
-rw-r--r-- 1 hirlam res      4795100 Apr 13 02:03 ob2004041300
-rw-r--r-- 1 hirlam res      32652000 Apr 13 02:19 fc200404130000
IBM Analysis took ... 16 mins
```

```

-rw-r--r-- 1 hirlam res      3435856 Apr 13 08:03 ob2004041306
-rw-r--r-- 1 hirlam res      32700000 Apr 13 08:19 fc200404130600
IBM Analysis took ... 16 mins

```

The following results are for the cluster, with the PGI compiler and the compiler flags listed in Appendix II.[Note that feedback files are not generated].

#### Dell Linux Cluster

```

-rw-r--r-- 1 hirlam hirlam  4169936 Apr 12 02:02 ob2004041200
-rw-r--r-- 1 hirlam hirlam  32676000 Apr 12 02:32 fc200404120000
Cluster analysis took ... 30 mins
-rw-r--r-- 1 hirlam hirlam  1798088 Apr 12 08:02 ob2004041206
-rw-r--r-- 1 hirlam hirlam  32568000 Apr 12 08:34 fc200404120600
Cluster Analysis took ... 32 mins
-rw-r--r-- 1 hirlam hirlam  3260512 Apr 12 14:02 ob2004041212
-rw-r--r-- 1 hirlam hirlam  32664000 Apr 12 14:34 fc200404121200
Cluster Analysis took ... 32 mins
-rw-r--r-- 1 hirlam hirlam  4386440 Apr 12 20:02 ob2004041218
-rw-r--r-- 1 hirlam hirlam  32604000 Apr 12 20:32 fc200404121800
Cluster Analysis took ... 30 mins
-rw-r--r-- 1 hirlam hirlam  4311096 Apr 13 02:02 ob2004041300
-rw-r--r-- 1 hirlam hirlam  32688000 Apr 13 02:29 fc200404130000
Cluster Analysis took ... 27 mins
-rw-r--r-- 1 hirlam hirlam  3146940 Apr 13 08:02 ob2004041306
-rw-r--r-- 1 hirlam hirlam  32652000 Apr 13 08:28 fc200404130600
Cluster Analysis took ... 26 mins

```

Comparing these 6 runs we see that the IBM takes between 15 and 17 mins per run; the cluster takes between 26 and 32 mins. [There is a larger variation in timings, compared with the forecast model, because the number of observations can vary from run to run]. Alternatively, the mean time for the IBM runs is 16 mins, for the cluster 29.5. Using these latter figures gives the result  $(29.5/16) = 1.84$  or  $(16/29.5) = 0.54$  meaning the IBM system is 1.84 times as fast as the cluster or, alternatively, the cluster is 0.54 times as fast as the IBM system.

Comparing individual processors [36 on the IBM, 12 on the cluster] we note that  $((16*36)/(29.5*12)) = 1.6$  or  $((29.5*12)/(16*36)) = 0.6$  meaning that a cluster processor is 1.6 times as fast as an IBM processor, or alternatively, an IBM processor is 0.6 times as fast as a cluster processor.

Note that these results are only for this particular configuration of 3DVAR with the particular hardware and software system used. Details of the compiler flags are given in Appendix II. The fact that the cluster version runs without generating feedback files is not expected to have a significant effect on the timings.

There is a dramatic difference in performance between the forecast model and the 3DVAR analysis *viz.* the cluster performance is 80% of the IBM for the model but just 50% for the analysis. As already noted, the IBM version of 3DVAR is extensively optimised using special IBM libraries for calculating maths functions and for FFT's. Also the cluster uses a more strict [*i.e.* slower] implementation of MPI. However, it is hoped to address some of these issues by (a) switching to the Intel compiler and its specialised maths libraries [which I have been told should be faster than the equivalent PGI system] and (b) looking in detail at the behaviour of the MPI routines, using the Scali diagnostic options, with a view to optimising them.

### **Summary and Discussion of Results**

The results for the forecast model and 3DVAR differ - when running the forecast model the cluster is about 80% as fast as the IBM RS/6000SP but with 3DVAR it is only 50% as fast. There are a number of reasons for the poor performance of 3DVAR including (a) the use of special optimised maths libraries on the IBM [including special FFT routines] which are not available on the cluster and (b) the use of a more strict version of MPI on the cluster. With careful optimisation, it is felt that the results for 3DVAR could be improved.

Generally, the experience with the cluster has been very good. It has proved reliable and its performance is comparable with the much more expensive IBM system. The addition of a few more nodes would be an inexpensive upgrade and would improve its performance further.

### **Plans for the Future**

The current system implements the 3DVAR analysis, the forecast model, the routines for generating climatological files, various post-processing programs, the single-processor version of the WAM model and various programs for generating products for customers. However, there is still work to do *viz.* (a) modify the boundary processing programs to run on more than one processor, (b) write a script to check which processors are available, and if not all are available, modify the run as appropriate to use the reduced set of processors, (c) implement the MPI version of WAM. It is also proposed to experiment with the Intel compilers and maths libraries to see if they will produce a faster run.

### **Acknowledgements**

Thanks to Lars Muller of SMHI, Norkopping, and Niclas Andersson and Torgny Faxen both at NSC, Linkopping for very helpful discussions. Also, thanks to Aarne Mannick, Tartu University, Gerald Cats, KNMI and Per Uden, SMHI, for helpful e-mails. Finally thanks to David Hutton, Scali and Greg Moore, Dell.

## Appendix I: Detailed Description of Cluster

The cluster consists of 7 rack-mounted nodes (1 master and 6 x compute nodes). The specification is:

- 6 x Dell Poweredge 1750 Dual Xeon Processor Compute Nodes
- 1 x Dell Poweredge 1750 Dual Xeon Processor Master Nodes
- 1 x Dell PowerConnect Cluster Communication Switch.
- PowerVault Tape System PV112T VS80 Rack Base 1U Single 40/80GB
- Dell Poweredge 4210 Rack.
- RedHat ES 3.0 on Master Node; Redhat WS 3.0 on compute nodes.
- Scali Manage , Scali MPI Connect and Scali TCP Connect.
- PGI Cluster Development Kit ,
- Intel Fortran , C and Math kernel Libraries.

### Master Node: Dell PowerEdge 1750

- Dual Intel Xeon processors at 2.8GHz with 512kb cache, 533 Mhz Front Side Bus
- 4GB ECC DDR RAM
- 3 x 146 GB 10k rpm Ultra320 SCSI disk drive, Dell PERC4/Di U320 Raid controller.
- On-Board dual PCI-X 10/100/1000 BaseT ethernet port
- ERA port for remote management
- Redundant Power Supply.

### Compute Node: Dell PowerEdge 1750

- Dual Intel Xeon processors at 3.2 GHz with 1Mb cache, 533 MHz Front Side Bus
- 2GB ECC DDR RAM
- 1 x 36 GB 10k rpm Ultra320 SCSI disk drive
- On-Board dual PCI-X 10/100/1000 BaseT ethernet port
- ERA port for remote management
- 4 Port Dolphin SCI HBA for compute node interconnect.

## Appendix II: Compiler Flags for PGI

### Forecast Model

The forecast model and associated programs [such as those used for the generation of climatological files] have been compiled with the PGI compilers. The following is a list of the flags used. Note that a new option 'CS=PGI' has been defined in Env\_system. The following sections show changes to the Env\_system, Makefile and Makefile\_x files.

### Env\_system:

```
# Setting LAMHOME forces system to use mpirun
export LAMHOME LAMHOSTS # activate these lines if needed
LAMHOME=/opt/scali
PATH=/opt/scali:$PATH
LAMHOSTS=
#
# serial version -- switch off MPI options
# LAMHOME=
#
# Setting MPIINC switches on compilation with MPI options
export MPILIB MPIINC # activate these lines if needed
MPILIB=
MPIINC=/opt/scali/include
#
# serial version -- switch off MPI options
# MPIINC=
```

### Makefile:

```
ifdef MPIINC
# MPIINC will lead to compilation of gc_com code
SCALI_MPI_HOME = /opt/scali
SCALI_MPI_LDLIBS = -lfmpi -lmpi -lpthread
FFLAGS += -D_REENTRANT -I$(SCALI_MPI_HOME)/include
          -L$(SCALI_MPI_HOME)/lib $(SCALI_MPI_LDLIBS)
          $(map) -DGC $(map) -DMPI_SRC
endif
#
ifeq ($(cs),PGI)
CC = pgcc -Mchkfpstk -Mchkstk -Mbounds -g
CFLAGS += -DG77 $(PREC) $(DB) -D_REENTRANT
FC = pgf90
OPT = -O2 $(PREC) -Mchkfpstk -Mchkstk -g
db = -g
endif
```

### Makefile\_x:

```
ifeq ($(PGM),hlprog)
  ifdef LAMHOME
    COST = /opt/scali/bin/mpirun -np $(NCPUS_PROG) -v
    RUN_OK =
```

```

        RUN_FAILED = $(RUN_OK)
    endif
endif

```

### Analysis: 3DVAR

The 3DVAR analysis program is essentially an independent system and it is compiled separately. The tests reported here were carried out with version 5.0.0. The system is built with 'ARCH=linuxpgi' which means the system takes its configuration setting from the file './config/config.linuxpgi'. The following section shows the changes made to this file.

#### config.linuxpgi:

```

ARCH=linuxpgi
SCALI_MPI_HOME=/opt/scali
SCALI_MPI_LDLIBS=-lfmpi -lmpi -lpthread
OPT =-fast -Mchkfpstk -Mchkstk -g -D_REENTRANT
#
MACHINECPPGRIB=$(MACHINECPP) -DGRIB32 -DIRISHBUFR
#
CC = pgcc -D_REENTRANT -I$(SCALI_MPI_HOME)/include
        -L$(SCALI_MPI_HOME)/lib $(SCALI_MPI_LDLIBS)
CCFLAGS = -c -DFUJITSU -DPREC32
        -I$(ROOTDIR)/ifsaux/ioassign/
        -I$(ROOTDIR)/ifsaux/pcma/ -DHIRLAM -DIRISHBUFR
FC =pgf90 -D_REENTRANT -I$(SCALI_MPI_HOME)/include
        -L$(SCALI_MPI_HOME)/lib $(SCALI_MPI_LDLIBS)
FCFLAGS_CMA = -c $(OPT) -r8 -i4
FCFLAGS_CMA90 = -c $(OPT) -r8 -i4 -module
$(ROOTDIR)/$(ARCH)/cmamod
MODEXT=mod
FCFLAGS_VAR = -c $(OPT) -r8 -i4 -module
$(ROOTDIR)/$(ARCH)/varmod
FCFLAGS_VAR90 = -c $(OPT) -r8 -i4 -module
$(ROOTDIR)/$(ARCH)/varmod
FCFLAGS_GRIB = -c $(OPT) -r4 -i4
LD = pgf90
#
LDFLAGS_VAR = -L$(SCALI_MPI_HOME)/lib $(SCALI_MPI_LDLIBS) -o
LDFLAGS_CMA = -L$(SCALI_MPI_HOME)/lib $(SCALI_MPI_LDLIBS) -o
#
#PARAMS_MAX: include file for params_max.h
PARAMSMAX=params_max.ibmsp
#
# FFT: 1) FFTFUJ;
#       2) CRAYMPPFFT (fft991 with FUJITSU modification)
#       3) STANDARDFFT991 (fft991 original)
FFT=-DFFTFUJ
#
# PARALLELIZATION LIBRARY
# Options -DMPILIB,-DSHMEMLIB,-DMPILIB -DMPI32TO64, <none>
PARLIB=-DMPILIB -DMPILARGE -DMPI32TO64 -DMPI_STRICT
        -I$(SCALI_MPI_HOME)/include

```

**Analyse:**

In addition to the above changes to the configuration the following command is used within the Analyse script to run 3DVAR using MPI:

```
MPPCMD="/opt/scali/bin/mpirun -np $MP_PROCS -v"
```

J. Hamilton  
Met Éireann  
15-April-2004