

BSC Tools Hands-On

Lau Mercadal Melià (tools@bsc.es) Barcelona Supercomputing Center





Getting a trace with Extrae



Extrae features

- Platforms
 - Intel, Cray, BlueGene, Intel MIC, ARM, Android, Fujitsu Sparc, IBM POWER..
- Parallel programming model
 - MPI, OpenMP, pthreads, OmpSs, CUDA, OpenCL, Java, Python...
- Performance Counters
 - Using PAPI interface
- Link to source code
 - Callstack at MPI routines
 - OpenMP outlined routines
 - Selected user functions
- Periodic samples
- User events (Extrae API)

No need to recompile or relink!

Extrae overheads

	Avg. values	MN4	CTE-POWER
Event	150 – 200ns	171ns	173ns
Event + PAPI	750 – 1000ns	755ns	1.4µs
Event + callstack (1 level)	1µs	1µs	5.3µs
Event + callstack (6 levels)	2µs	2.5µs	12µs

How does Extrae work?

Symbol substitution through LD_PRELOAD

- Specific libraries for each combination of runtimes
 - MPI
 - OpenMP
 - OpenMP+MPI
 - CUDA
 - ...
- Dynamic instrumentation
 - Based on DynInst (developed by U.Wisconsin/U.Maryland)
 - Instrumentation in memory
 - Binary rewriting

• Function instrumentation through "-finstrument-functions" compiler option

Static link (i.e., PMPI, Extrae API)



Using Extrae in 3 steps

- **1. Adapt** your job submission script
- 2. Configure what to trace
 - XML configuration file
 - Example configurations in \$EXTRAE_HOME/share/example
- 3. Run it!
- For further reference check the Extrae User Guide:
 - <u>https://tools.bsc.es/doc/html/extrae</u>
 - Also distributed with Extrae in \$EXTRAE_HOME/share/doc

V VIRTUAL INSTITUTE - HIGH PRODUCTIVITY SUPERCOMPUTING

Login to MareNostrum4

```
laptop> ssh -Y <USER>@mn1.bsc.es
login1> ls $HOME/tools-material
   ... apps/
   ... clustering/
   ... extrae_cte-power/
   ... extrae_mn4/
                             Here you have
   ... slides/ 🔶
                           a copy of the slides
   ... src/
   ... traces/
```

Step 1: Adapt the job script to load Extrae with LD_PRELOAD

login1> vi \$HOME/tools-material/extrae_mn4/job.sh



VIRTUAL INSTITUTE -- HIGH PRODUCTIVITY SUPERCOMPUTING

Step 1: Adapt the job script to load Extrae with LD_PRELOAD

login1> vi \$HOME/tools-material/extrae_mn4/job.sh



V VIRTUAL INSTITUTE - HIGH PRODUCTIVITY SUPERCOMPUTING

Step 1: Adapt the job script to load Extrae with LD_PRELOAD



Step 1: Which tracing library?

Choose depending on the application type

Library	Serial	MPI	OpenMP	pthread	CUDA
libseqtrace	\checkmark				
libmpitrace[f] ¹		\checkmark			
libomptrace			\checkmark		
libpttrace				\checkmark	
libcudatrace					\checkmark
libompitrace[f] ¹		\checkmark	\checkmark		
libptmpitrace[f] ¹		\checkmark		\checkmark	
libcudampitrace[f] ¹		\checkmark			\checkmark

¹ include suffix "f" in Fortran codes

Step 3: Run it!

Submit your job

login1> cd \$HOME/tools-material/extrae_mn4

login1> sbatch job.sh

- Once finished the trace will be in the same folder: lulesh_mn4_27p.{pcf,prv,row}
 - Check the status of your job with:

login1> squeue

Any issue?

Already generated at \$HOME/tools-material/traces

Step 2: Extrae XML configuration





Step 2: Extrae XML configuration (II)

login1> vi \$HOME/tools-material/extrae_mn4/extrae.xml

```
<counters enabled="yes">
 <cpu enabled="yes" starting-set-distribution="1">
    <set enabled="yes" domain="all" changeat-time="500000us">
     PAPI_TOT_INS, PAPI_TOT_CYC, PAPI_L1_DCM, PAPI_L2_DCM, PAPI_L3_TCM, PAPI_BR_INS, PAPI_BR_MSP, RESOURCE_STALLS
   </set>
    <set enabled="yes" domain="all" changeat-time="500000us">
     PAPI_TOT_INS, PAPI_TOT_CYC, PAPI_SR_INS, PAPI_LD_INS
    </set>
    <set enabled="yes" domain="all" changeat-time="500000us">
     PAPI_TOT_INS, PAPI_TOT_CYC, PAPI_VEC_SP
    </set>
 </cpu>
 <network enabled="no" />
 <resource-usage enabled="no" />
                                                                                                   Select which HW counters
 <memory-usage enabled="no" />
                                                                                                           are measured
</counters>
```

(How's the machine doing?)

VIRTUAL INSTITUTE -> HIGH PRODUCTIVITY SUPERCOMPUTING

Step 2: Extrae XML configuration (III)

login1> vi \$HOME/tools-material/extrae_mn4/extrae.xml <buffer enabled="yes"> <size enabled="yes">5000000</size> Trace buffer size <circular enabled="no" /> (Flush/memory trade-off) </buffer> <sampling enabled="no" type="default" period="50m" variability="10m" /> **Enable sampling** <merge enabled="yes" (Want more details?) synchronization="default" tree-fan-out="16" **Automatic** max-memory="512" post-processing joint-states="yes" to generate the keep-mpits="yes" **Paraver trace** sort-addresses="yes" overwrite="yes"> \$TRACE NAME\$ </merge>



Installing Paraver & First analysis steps



VIRTUAL INSTITUTE - HIGH PRODUCTIVITY SUPERCOMPUTING

Install Paraver in your laptop



Install Paraver (II)

\$HOME/tools-packages

- Download tutorials:
 - Documentation -> Tutorial guidelines



laptop> scp mn1:\$HOME/tools-packages/paraver-tutorials-20150526.tar.gz \$HOME

VIRTUAL/INSTITUTE -->HIGH PRODUCTIVITY SUPERCOMPUTING

Uncompress, rename & move

Paraver

laptop> tar xf wxparaver-4.8.1-linux-x86_64.tar.gz

laptop> mv wxparaver-4.8.1-linux-x86_64 paraver

Tutorials

laptop> tar xf paraver-tutorials-20150526.tar.gz

laptop> mv paraver-tutorials-20150526 paraver/tutorials

Check that everything works

Start Paraver

laptop> \$HOME/paraver/bin/wxparaver &

Check that tutorials are available



Remotely available in MareNostrum





First steps of analysis

Copy the trace to your laptop (All 3 files: *.prv, *.pcf, *.row)

laptop> scp <USER>@mn1.bsc.es:\$HOME/tools-material/extrae_mn4/lulesh* ./

Load the trace



Click on File → Load Trace → Browse to the *.prv file

Follow Tutorial #3



Measure the parallel efficiency

Click on "mpi_stats.cfg"

		••••••••••••••••••••••••••••••••••••••	•• 2 • -					
	THREAD 1.16.1	92.23 %	0.05 %	0.03 %	0.06 %	0.38 %	0.02 %	0.00 %
1115	THREAD 1.17.1	91.11 %	0.07 %	0.05 %	0.05 %	0.91%	0.02 %	0.50 %
ne first question to answer when analyzing a parallel code is "how efficient does it n?" The efficiency of a parallel program can be defined based on two aspects: the	THREAD 1.18.1	88.95 %	0.05 %	0.03 %	0.05 %	0.96 %	0.03 %	0.00 %
arallelization efficiency and the efficiency obtained in the execution of the serial	THREAD 1.19.1	91.58 %	0.03 %	0.02 %	0.09 %	0.58 %	0.03 %	0.00 %
	THREAD 1.20.1	88.29 %	0.05 %	0.03 %	0.04 %	0.54 %	0.02 %	0.00 %
 To measure the parallel efficiency load the configuration file cfgs/mpi/mpi_stats.cfg This c infiguration pops up a table with %time that 	THREAD 1.21.1	86.69 %	0.04 %	0.02 %	0.17 %	0.66 %	0.03 %	0.94 %
every thread spends in every MPI c II. Look at the global statistics at the bottom of	THREAD 1.22.1	87.61 %	0.05 %	0.03 %	0.07 %	0.63 %	0.03 %	0.00 %
efficiency, entry Avg/Max represents the global load balance and entry Maximum represents the communication efficiency. If any of those values are lower than	THREAD 1.23.1	95.95 %	0.07 %	0.04 %	0.04 %	0.83 %	0.02 %	0.00 %
85% is recommended to look at the corresponding metric in detail. Open the control window to identify the phases and iterations of the code.	THREAD 1.24.1	93.78 %	0.05 %	0.05 %	0.05 %	0.71%	0.03 %	0.00 %
To measure the computation time distribution load the configuration file	THREAD 1.25.1	93.89 %	0.04 %	0.03 %	0.05 %	0.79 %	0.03 %	0.00 %
cfgs/general/2dh usefulduration cfg This configuration pops up a	THREAD 1.26.1	92.85 %	0.06 %	0.03 %	0.16 %	0.68 %	0.02 %	0.00 %
are delimited by the exit from an MPI call and the entry to the next call. If the	THREAD 1.27.1	91.41%	0.05 %	0.02 %	0.05 %	1.22 %	0.03 %	0.00 %
not balanced. Open the control window to look at the time distribution and visually								
correlate both views.	Total	2,467.65 %	1.36 %	0.97 %	2.97 %	17.74 %	0.51%	2.99 %
• To measure the compu- configuration file of gs/r configuration pops up a The computation regions		91.39 %	0.05 %	0.04 %	0.11%	0.66 %	0.02 %	0.11%
	Maxim	98.04 %	0.09 %	0.08 %	0.36 %	1.22 %	0.03 %	0.94 %
to the next call. If the histogram account on a vertice management of the instruction of the instruction	mimum	86.69 %	0.02 %	0.02 %	0.03 %	0.24 %	0.00 %	0.00 %
look at the time distribut	StDev	3.09 %	0.01%	0.01%	0.08 %	0.23 %	0.01 %	0.26 %
• To measure the serial COMM ETHCIENCY	Avg/M-	0.93	0.53	0.46	0.30	0.54	0.69	0.12
Lood balance								
	1							

MPI call profile @ lulesh mn4 27p.prv.gz

Computation time and work distribution

■ Click on "2dh_usefulduration.cfg" (2nd link) → Shows time computing





Computation time and work distribution

• ... and "2dh_useful_instructions.cfg" (3rd link) > Shows amount of work

2dh usefu	l instructions @ lulesh_mn4_27p.prv.gz 3D [🔍 🔍 [🔳 🛏 🛏 💵 🏧 🌫 🏂
	7
/Ork	
alance	1.1.1 [719,061,017.51723,315,810.21) = 0 us
-zag)	

Where does this happen?

Go from the table to the timeline

Where does this happen?

Slow & Fast at the same time -> Imbalance

Where does this happen?

• Hints \rightarrow Callers \rightarrow Caller function

VIRTUAL INSTITUTE - HIGH PRODUCTIVITY SUPERCOMPUTING

Save CFG's (2 methods)

VIRTUAL/INSTITUTE --- HIGH-PRODUCTIVITY SUPERCOMPUTING

Save CFG's (2 methods)

VIRTUAL INSTITUTE - HIGH PRODUCTIVITY SUPERCOMPUTING

CFG's distribution

Paraver comes with many more included CFG's

Paraver				
File Hints Help Load Irace Previous Traces Unload Traces	Load Configura	ation		
Load Configuration Previous Configurations Save Configuration	Look in:	cfgs	💽 🔬 🖾	
Load Session CTRL+I Save Session CTRL+S Preferences vith	burst_mode	🛅 Java 🛅 mpi	i sampling	+folding
Quit 2dh useful instructions 3 useful instructions 2DZoom range [1.60154e+0(3 MPI caller	CUDA	PI DoppSs DopenCL OpenMP	software spectral	_counters
	📄 General	in pthread		
 ⊕- ☐ lib ▲ ☐ libbsctools ⊕- ☐ share ⊕- ☐ src 	File <u>n</u> ame:		•	<u>O</u> pen
telegram teleg	Files of type:	Paraver configuration f	file (*.cfg)	Cancel
Paraver files				

Hints: a good place to start!

Paraver suggests CFG's based on the information present in the trace

Cluster-based analysis

Use clustering analysis

Run clustering

login1>	module load clustering_suite/2.6.8
login1>	cd \$HOME/tools-material/clustering
login1>	<pre>BurstClustering -d cluster.xml -i/extrae_mn4/lulesh_mn4_27p.prv \ -o lulesh_mn4_27p_clustered.prv</pre>

If you didn't get your own trace, use a prepared one from:

login1> ls \$HOME/tools-material/traces/lulesh_mn4_27p.prv

Copy the results to your computer

laptop> scp <USER>@mn1:\$HOME/tools-material/clustering/* ./

Cluster-based analysis

Check the resulting scatter plot

laptop> gnuplot lulesh_mn4_27p_clustered.IPC.PAPI_TOT_INS.gnuplot

- Identify main computing trends
- Work (Y) vs. Speed (X)
- Look at the clusters shape
 - Variability in both axes indicate potential imbalances

Correlating scatter plot and time distribution

• Open the clustered trace with Paraver and look at it

laptop> \$HOME/paraver/bin/wxparaver lulesh_mn4_27p_clustered.prv

- Display the distribution of clusters over time
 - File → Load configuration → \$HOME/paraver/cfgs/clustering/clusterID_window.cfg

BSC Tools Hands-On

Lau Mercadal Melià (tools@bsc.es) Barcelona Supercomputing Center

