COST-ET: COSTing and Exploring TCO for data centers

Documentation

Marios Kleanthous, Damien Hardy, Isidoros Sideris, Yiannakis Sazeides
Computer Architecture Research Group
University of Cyprus

The research leading to this tool is supported by the European Commission FP7 project "Energy-conscious 3D Server-on-Chip for Green Cloud Services (Project No:247779 "EuroCloud")".

Tool: http://www.cs.ucy.ac.cy/carch/xi/cost-et.php
Project: http://www.eurocloudserver.com/
European Commission FP7: http://cordis.europa.eu/fp7/home_en.html

Revision Control

Ver 1.0	10/03/2011	Isidoros Sideris
Ver 1.1	02/12/2011	Damien Hardy
Ver 1.2	17/04/2012	Damien Hardy
Ver 1.3	18/05/2012	Damien Hardy
Ver 1.4	18/04/2013	Marios Kleanthous

COST-ET

This tool models the total cost of ownership of a data center populated by different resources (computing nodes, storage nodes…). This tool is divided in two parts: the kernel part for the estimation and the wrapper part for the exploration.

Change Log:

Version 1.4:

- Configurations for validating the tool with published results were added (input_examples/barossoA and input_examples/Hamilton)
- More accurate configurations for High performance and Low power servers based on real products were added
- Option to include/exclude the network and maintenance cost in the TCO was added
- Option to avoid having cold spares for the maintenance model was added
- Option to have loan interest only for the servers' cost was added
- Output of Peak and Idle power per server module was added
- Option to have extra area for non-data center equipment (eg. Power generators, cooling compressors, etc.) was added

Basic Operation

The kernel takes as an input a data center configuration and at least one resource configuration (computing nodes, storage nodes…) and computes the TCO of that configuration, some information about the data center and the environmental impact of the data center. The computation starts with an estimation of the number Hot and Cold spares required to reach the number of server request during the server depreciation lifetime. This computation used the MTTF of the server components and the performance vulnerability factor (PVF) of the chip. The tool, after performing this step, computes the server acquisition cost, the network acquisition cost, the power cost (consumed by servers, network equipment and cooling equipment), the maintenance cost and the cost of acquisition of the building (space) and cooling and power delivery equipment infrastructure. After these steps, the resulting TCO and its different component are print as the environmental impact and some information about the data center.

The tool is defined to explore trade-offs between different configurations. Some exploration experiments show how to define your own experiment based on the kernel tool (see Section Wrappers).

Download & Install

Requirements:
- Perl, gnuplot

Download the tool from http://www.cs.ucy.ac.cy/carch/xi/eetco.php
The tool is written completely in Perl scripts. There is no need to install it.

Note:
- gnuplot is recommended (but not strongly needed) only for the exploration experiments (see Section Wrappers).

Use

Run the tool by writing:
./cost-et.pl input_example/dc.params input_example/r.params
dc.params is a data center configuration file and *r.params* is a data center configuration file (see Section Input).
The output of the tool shows extensive information about costs and other estimations concerning the data center. For details about the output see Section Output.

Structure

The kernel is written in perl scripts organized in 3 files:
+ *cost-et*.pl
+ packages/dc.pm
+ packages/resource.pm

The **cost-et.pl** script is the main tool script, which uses a data center analytical model to estimate and output the TCO and the environmental impact, along with other intermediate results (see Section Output). **dc.pm** and **resource.pm** are packages used by cost-et.pl script to parse configuration files, to store the inputs information, to print the configuration, to estimate TCO and intermediate results.

In the main directory, there are three subdirectories:

+ doc	which contains this document
+ input_example	which contains configuration file examples See Section Input for more details.
+ packages	which contains dc.pm and resource.pm
+ wrappers	which contains different exploration experiments. See Section Wrappers for more details.

Input

COST-ET takes as an input a data center configuration file and at least one resource configuration file where their structures are detailed bellow. In these files, character # is used for comments and there is no specific order for the parameters. In the example files, parameters are grouped by categories.

For the data center:

- Basic data center parameters
- Non server modules requirements & parameters

For the resource:

- Basic workload requirements
- Rack cabinet parameters
- Network parameters
- Server module configuration & characteristics
- Chip configuration & characteristics

Data center Configuration file

Each parameter of a data center configuration are described in the following tabular

c_buildingpersqm	Cost of land acquisition/building deployment ($/m^2)
a_coolingarea	A simplified factor accounting for x% more space for the cooling equipment Example: 1.2 for 20%
a_other	Area required for other equipment outside the data center. Eg. Power generators, cooling compressors etc. (m^2)
dcdepr	Depreciation of data center (years)
c_coolpowerdeplperW	Cost of cooling and power distribution infrastructure deployment ($/W)
c_electperKWH	Cost of electricity per KWh
PUE	Power Usage Effectiveness
salary_perrackpermonth	Salary cost of data center staff per rack per month ($)
n_othermodules	Number of non server modules
p_othermodules	Total Power of non server modules
p_othermodulesidle	Total Power of non server modules when idle
n_othermodulesrepl	Number of replacements needed for non server

	modules throughout server depreciation time interval (depr_othermodules)
c_nonsrvmodule	Average cost of a non server module ($)
depr_othermodules	Depreciation of other modules (year)
u_othermodules	Average utilization other modules
a_othermodules	Total area used by other modules (m^2)
n_racks_othermodules	Number of racks used by other modules
report_resources	1 to report each resource details in the output 0 otherwise
cost_of_money	Optional parameter: Cost of money for the interests of loans x% interests fix per year over the depreciation of infrastructure, servers and network Example: 0.05 for 5%
cost_of_money_servers	Optional parameter: Cost of money for the interests of loans to buy only the servers x% interests fix per year over the depreciation of infrastructure, servers and network Example: 0.05 for 5% Warning: If this parameter is defined then the cost_of_money parameter will be ignored
c_elecpeak	Optional parameter: if the parameter is not present, the default value is 0. 1 if the peak power consumption has to be considered for the power cost contribution to TCO 0 otherwise, if the actual power consumption has to be considered
include_maintenance	Flag to include the maintenance cost in the TCO
include_network	Flag to include the network cost in the TCO

Resource Configuration file

Each parameter of a resource configuration are described in the following tabular

resource_name	Name of the resource
n_srvmodulesreq	Total number of server required for the peak workload requirements
u	Average server utilization (0<=u<=1)
coef_maintenance	Coefficient to reduce the cold spares' cost e.g.: 1 no reduction ; 0.5 half the price
k_bladesperrack	Number of blades per rack
k_modulesperblade	Number of (server) modules per blade
height	Rack height (m). 42RU->1.78m - not used
width	Rack width (m)
depth	Rack depth (m)
minimumdistance	Minimum distance between two racks (m) - not used
useddistance	Distance between two rack lanes in our data center (m)
c_networkperrack	Cost of network equipment per rack ($)
networkdepr	Depreciation of network equipment (years)
network_peak_powerperrack	Peak network power per rack (W)
srvdepr	Depreciation of server equipment (years)
mttr	Mean time to repair a faulty server module (hours)
nb_time_step	Number of steps used in spares estimation
SPUE	Server Power Usage Effectiveness IT power/SPUE= power to pure electronic components (no fans etc.)
k_chipspersrvmodule	Number of chips per server module
c_system	System cost ($)
p_system	System power (W)
p_system_idle	System power idle (W)

n_dram	Number of DRAM DIMM per server module
c_dram	DRAM cost (1 DIMM) ($)
p_dram	DRAM power (1 DIMM) (W)
p_dram_idle	DRAM power idle (1 DIMM) (W)
mttf_dram	DRAM MTTF (1 DIMM) (year)
n_disk	Number of disk element per server module
c_disk	Disk cost (1 element) ($)
p_disk	Disk power (1 element) (W)
p_disk_idle	Disk power idle (1 element) (W)
mttf_disk	Disk MTTF (1 element) (year)
n_switchport	Number of switchports per server module
c_switchport	Switchport cost (1 element) ($)
p_switchport	Switchport power (1 element) (W)
p_switchport_idle	Switchport power idle (1 element) (W)
mttf_switchport	Switchport MTTF (1 element) (year)
c_other	Cost other component ($)
p_other	Power other component (W)
p_other_idle 0	Power idle other component (W)
c_chip	Chip cost ($) (i.e. 1 processor cost)
pvf	Performance vulnerability factor (0<=pvf<1)
p_chip	Chip Power (W)
p_chip_idle	Chip Power idle (W)
chip_mttf	Chip Mean Time To Failure (years).
chip_perf	Performance of 1 chip
mttf_coef	Optional parameter: (between 0.5 and 1) To adjust the MTTF of all component in case the ambient temperature increases 1: no MTTF reduction 0.5: MTTF divided by 2

Output

The main output consists of an extensive costs report followed by data center information and the detailed for each resource like the one bellow:

```
----------------------
DC parameters
----------------------
…
----------------------
Resources parameters
----------------------
…
----------------------
DC_results
----------------------

tco:                    9120281.24 $/month
c_infrastructure:       964075.8 $/month 11%
c_server:               5222222.22 $/month 57%
c_power:                877231.56 $/month 10%
c_network:              248125 $/month 3%
c_maintenance:                  1808626.67 $/month 20%

n_racks:        1191
a_dc:           2058.05 m^2
a_other:        0 m^2
p_total_avg:    8388588 W
p_total_peak:   17405388 W

co2peryr:       40064963.32 kg
----------------------
Resources_results
----------------------
Resource:HPE

tco:                    9120281.24 $/month
c_infrastructure:       964075.8 $/month
c_server:               5222222.22 $/month
c_power:                877231.56 $/month
c_network:              248125 $/month
c_maintenance:                  1808626.67 $/month

n_racks:        1191
a_perrack:      1.44 m^2
p_peakperrack:  11246.4 W
a_dc:           2058.05 m^2
n_srvmodules:   50000
n_coldsp:       15036
c_srvmodule:    3760 $
availability:   0.999954230405454

p_total_avg:    8388588 W
p_total_peak:   17405388 W

Performance:             100000
PerformanceOverTCO:      10.96 x1000
PerformanceOverWatt:     5.75 x1000

----------------------
p_peakperserver: 259.2 W
p_idleperserver: 85.8 W
Peak/Idle Server: 3.02097902097902
Peak/Idle Total: 2.83734709966178
```

In the following tabular there is an explanation of the outputs concerning the data center and resources.

- Data center outputs

tco	Total Cost of Ownership per month
c_infrastructure	Cost of acquisition per month of the data center infrastructure (building, land, cooling and power delivery equipment)
c_server	Cost of server acquisition per month
c_power	Cost of power per month
c_network	Cost of network equipment acquisition per month
c_maintenance	Cost of maintenance per month. It includes staff salaries and replacements of server modules.
n_racks	Number of racks
a_dc	Total data center Area
a_other	Area required outside the data center
p_total_avg	Average power dissipated
p_total_peak	Peak power dissipated
co2peryr	Environmental impact of the data center in kg CO2 (per year)

- Resource outputs

tco	Total Cost of Ownership per month for the resource
c_infrastructure	Cost of acquisition per month of the data center infrastructure (building, land, cooling and power delivery equipment) for the resource
c_server	Cost of server acquisition per month for the resource
c_power	Cost of power per month for the resource
c_network	Cost of network equipment acquisition per month for the resource
c_maintenance	Cost of maintenance per month. It includes staff salaries and replacements of server modules for the resource

n_racks	Number of racks for the resource
a_perrack	Rack area
p_peakperrack	Peak power per rack
a_dc	Total data center Area for the resource
n_srvmodules	Number of server modules required for the peak workloads: n_srvmodulesreq (+hotspares if pvf>0)
n_coldsp	Number of replacements required throughout srvdepr (year)
c_srvmodule	Cost of a server module
availability	Availability of the resource
p_total_avg	Average power dissipated by the resource
p_total_peak	Peak power dissipated by the resource
Performance	Total performance of the resource
PerformanceOverTCO	Performance/TCO x1000
PerformanceOverWatt	Performance/p_total_peak x1000
p_peakperserver	Peak power per server
p_idleperserver	Idle power per server

Wrappers

So far, this document has focused on how to run the kernel with a given configuration. This tool can also be used to explore trade-offs between different configurations by defining wrapper scripts. Specific experiments are defined in the subdirectory wrappers:
- epc performance (comparison of different server performance)
- ambient_temperature impact

And more generic wrappers are defined to facilitate your explorations:
- breakdown: to show the TCO breakdown of different configurations
- 2D exploration: to perform a sweep on a given input parameter
- 3D exploration: to perform a sweep on two given input parameters
- iso: to get iso curve (keep constant one output) when a sweep is performed on a given input parameter
- sensitivity: to show the sensitivity of an output for all input parameters

Requirements: gnuplot* (*if you do not have gnuplot, you can see the results in an output text file.)

Each wrapper in the .pl file has its own explication the detail of how to call it and a command line example.

All wrappers are organized similarly in their directory:

- name.pl	which is the perl script to run this experiment
- output directory	empty directory used to store the output files generated during the execution of the script
+ output.txt	output file which contains the results of the experiment. This file is then used as an input of gnuplot.
+ output.eps	The resulting graph of the experiment

Also a template directory is included in the root directory of wrappers:

- template directory	
- dc_template.params	template of dc.params file.
- r_template.params	template of r.params file.

How to define a new wrapper

To define a new wrapper, you can look at the existing ones to get an idea of how there are defined. Basically, they are organized as follows:

- Initialisation phase (generic and specific)
- main loop exploration
- gnuplot call
- clean phase

Moreover a library is defined to help defining new wrappers in the directory wrappers/lib:

- helper.pm

Which contains the following functions:

- new: to instanciate the library
- parseDCFile: to get a pointer to a hash containing the DC's parameters of a file
- parseResourceFile: to get a pointer to a hash containing the resource's parameters of a file
- checkInputParameter: to check if an input parameter exist in the kernel part
- checkOutputValueName: to check if an output name exist in the kernel part
- createParameterFile: to create a parameter file (dc/resource)
- callKerner: to call the kernel
- parseResultKernel: to get a pointer to a hash containing the output of a kernel call
- deleteFile: to delete a file
- deleteFileInDir: to delete all the files contained in a directory
- makeGnuplotCurve2D: to call gnuplot and plot a 2D curve graph
- makeGnuplotCurve3D: to call gnuplot and plot a 3D curve graph
- makeGnuplotStackedHistogram: to call gnuplot and plot a stacked histogram
- makeGnuplotGroupCurve: to call gnuplot and plot a group curve graph