ISSN: 2321-7782 (Online)
Volume 3, Issue 3, March 2015
International Journal of Advance Research in Computer Science and Management Studies
Research Article / Survey Paper / Case Study
Available online at: www.ijarcsms.com

Cloud Compiler with Green Strategies with Multi Language Compiling Facility

A.Yaganteeswarudu[1]
Asst. Prof, Computer science and engineering
Sreenidhi institute of science and technology (Autonomous)
Hyderabad, Andhra Pradesh, India

Y.Vishnuvardhan[2]
Asst. Prof Computer science and engineering
Dr. MVSIT
Mangalore, India

Abstract: ***Energy conservation has become a critical issue in modern system electronic devices. Energy efficiency in electronic devices depends on hardware and software components of device. In computer systems, software drives hardware, and decisions taken during software development and design have significant impact on energy consumption of a computing system. Compilers are used to run programs and convert them from a text format to executable format. A compiler that is to be installed manually on every system physically requires a lot of space and also configuring of it if not installed using default parameters. Also once a program is compiled it becomes platform dependent. It is also not easy to carry the same program code to multiple systems if situation doesn't permit the usage of a single system. Another drawback is that we would need to install a different complier on each language on which we wish to work.***

Energy conservation in computer system by applying different techniques at software and hardware level is the area addresses by Green Computing. Energy efficient compiler is one of the software level green computing techniques. Compilers contains bundle of information related to software structure and execution. Therefore, focus of this paper is identification and implementation of various green aspects for energy conservation at compiler level. A Green Cloud Compiler (GCC) is presented in this work that is hardware independent and create an online compiler which helps to reduce the problems of portability of storage and space by making use of the concept of cloud computing. The ability to use different compilers allows the programmer to pick up the fastest or the most convenient tool to compile the code and remove the errors. Moreover a web based application can be used remotely through any network connection which is platform independent. The errors/Output of the compiled program can be stored in a more convenient way. Also the trouble of installing a compiler on each computer is avoided. Thus these advantages make this application ideal for conducting online examinations. It distributes source code of software over a network, reshapes binary code by applying green strategies during code transformation at compile time and gives green suggestion to software programmer for energy conservation.

Keywords: compiler, green cloud compiler, cloud computing, Green computing , Green Strategies

I. INTRODUCTION

Utilizing resources effectively, improving environmental performance and defending global warming are in priority on the list of global challenges that must be addressed urgently. Governments and business associations have introduced a range of programmes and initiatives to address environmental challenges, particularly global warming and energy use. Business associations have mainly developed initiatives to reduce energy costs and to demonstrate corporate social responsibility. Green computing is a large and increasing area. The need for saving energy has become a top priority in almost all segments of the IT market. The need for power efficiency has become a critical factor in the design of high performance computing. The information and communications technology (ICT) industry needs to further improve its environmental performance, and ICT applications have very large potential to enhance performance across the economy and society. Energy conservation can be made at hardware and software levels. Software level energy conservation can be achieved by implementing various Green

scheduling techniques in the operating system. Energy can also be saved during various stages of software development life cycle such as software analysis and design, by applying different green approaches. Data centers are found in all economic sectors, as they provide computational infrastructure for a wide range of applications. The most valuable possession of companies is information. It is expected that data centers should always be available with its secured data. A loss of information or lack of availability may result in large economical loss. The need for power efficiency has become a critical factor in the design of data centers also. This paper focuses on the various energy conservation methods in the software and hardware levels of computing systems. The paper also lists out the best practices to be followed in data centers for improving energy efficiency.

Our implementation is a private cloud on which the application would be hosted. The software would be provided to the end user using a SAAS cloud. The software would contain a system that has a text editor and a terminal. The user would be given an option to select the language in which he wants to compile the program. The software will compile the program and return the output to the user. Additional functionalities such as monitoring of the system, user usage, user forums, and collaborative development can be added as needed.

In section II, the green techniques that can be applied at the compiler are explained. The green strategies, which can be followed by programmers during software development, are discussed in section III. Extension over traditional compilers are given in the section IV. The best practices to be followed in energy efficient data center design are described in section V. In section VI Final implementation, VII future work section, VIII the paper is concluded and section.

II. GREEN STRATEGIES FOR COMPLILER

Software programs are analysed at run time using energy aware compilers and software source code are reshaped by applying several green aspects during code transformation. The green techniques that can be applied at local, global or interprocedural level to make program energy aware are given below.

2.1 Cache skipping

Loops are very useful in programming and it increases performance, but causes high energy consumption due to repetition of the same thing. A good method is skipping of cache operations during unnecessary replication. The study presents an efficient method to solve cache-skipping problem by modification in compiler. In this technique, compiler needs to separate the blocks that has less chance for execution. It is found from study that in some cases there is no use of cache and hence this technique results in reduced power consumption.

2.2 Instruction clustering

The study shows that instruction clustering can conserve energy from 26% to 47%. A compiler with special type of architecture, can execute a cluster of instructions in one cycle. For example in signal processing applications, a cluster of related or similar signals can be compiled in one run. It will reduce the running time of program and leads to energy conservation.

2.3 Instruction reordering and memory addressing

Sometimes the order of instructions and memory addressing is not in the favorable order which supports energy saving mode. Energy consumption can be reduced by changing the order of instruction in such a way to suit the power-safe mode. A method is proposed using Gray Code and Cold Scheduling. Gray code is used to reference consecutive memory location. Using Gray code reduces the energy consumption by 36% as compared to binary representation of memory. Cold scheduling algorithm for instruction scheduling uses gray code that reduces 20% to 30% instruction switching.

2.4 Optimized energy cost tree

Energy aware compilers can use energy cost database for each transaction/instruction. This database can be used in code parsing and parse tree generation algorithms. During the first run of code processing, all possible parse trees are generated and their respective energy cost is stored in energy cost database. In subsequent run, tree with minimum cost will be selected for further compilation.

2.5 Loop optimization

Loop optimization methods are used to increase energy efficiency, which checks nested loops across dependency graph. Dependency graph are prepared for loop body in which nodes represent statements and edge corresponds to data dependency. If there is no cycle in the graph, compiler will create loop for each statement and run them in parallel using interleaved processing.

2.6 Dynamic power management

Power consumption in Complementary Metal - Oxide Semi conductors (CMOS) is classified into static and dynamic. If circuit is in operating state and there is no power leakage happens then power consumption is dynamic. Whereas power consumption is static, if circuit will not be in running form but it is still powered. Dynamic power management system sets the power of its hardware as per demand to decrease probable wastage of power.

2.7 Resource hibernation

Hibernation is the process of using low power mode. Idle resource can be kept in hibernate state but changing to and from this state can be wastage of precious resources and time. A compiler algorithm reshapes a program behaviour using source level transformation in such a way that idleness threshold of a resource can be extended, and it can be changed to hibernation mode with less switching. A compiler needs to call OS directives for activeness and inactiveness of specific resource.

2.8 Cloud aware task mapping

Services provided by different clouds can be used for cloud aware task mapping. A compilation technique uses cloud services at host level for possible computation by parallel processing and keeping records. A machine independent compiler can also use all services from remote clouds and can be in hibernation mode during progression of these services. There are limitations for this technique, one of them is the cost of virtual machine and migration cost of host machines. Second problem is failure of network machines causing delay in compilation or other service utilization.

2.9 Eliminate recursion

Recursive procedures using stack executed by compiler takes a lot of space and time causing reduced performance as well as additional energy consumption. Using compiler which can converts recursion into iteration may save time and energy in some cases.

III. GREEN STRATEGIES FOR SOFTWARE DEVELOPMENT

During various stages of software development life cycle such as software analysis, design and implementation, energy can be conserved. At design level, energy can be saved by making energy efficient software structure. Programmers can use following strategies during software development.

3.1 Use of green compiler

Several energy aware or green compilers are available, which can be used for energy conservation. For example Green Hill compiler can be used for C and C++, encc can be used for C++.

3.2 Use of readymade computer resources

There are number of readymade computer resources available as a service, for example currency converter, calculators etc. Making use of these readymade resources in program will be beneficial in terms of energy, cost and time.

3.3 Using iteration

More energy consumption happens in the case of recursion due to longer execution time. Therefore, a better approach is to use iteration and avoid recursion as much as possible in software development.

3.4 Data structures and algorithms

Using algorithms with less time complexity and energy efficient data structures in software development can be helpful for saving energy.

Hardware Energy Saving Methods

Research is going on to find new materials that can be used to do computations and to operate transistors at lower voltage levels. It is assumed that near threshold computing can reduce energy requirements by 10 to 100 times in future systems. Many digital design techniques are described in the book “Power reduction techniques for microprocessor systems” by V.Venkatachalam, M. Franz. These are:- making smaller transistors, reordering transistors in a circuit, logic gate restructuring , technology mapping where the components are selected from a library to meet energy constraints and the use of low power flip flops.

4.1 Dynamic Voltage and Frequency Scaling.

DVFS requires special hardware components, but is controlled by software. It is used to reduce the supply voltage of a processor when the work load is too less that the processor can reduce its speed and still have a performance , that is sufficient enough to meet the system requirements. Frequency reduction makes it viable to reduce supply voltage since gates can take longer time for switching. This will reduce the dynamic power consumption.

4.2 Sleep mode

Energy can be saved, if the system is put into sleep mode, after finishing the execution.

4.3 Power management techniques at the OS level

Many power management techniques control the amount of parallelism dynamically. FDT(Feedback driven threading) is a framework that dynamically controls the number of threads using run time information. FDT can be used to implement synchronization aware threading SAT. BAT, Bandwidth Aware Threading ,predicts how many threads can be executed before the off chip bus get saturated. Both these techniques can reduce execution time and power consumption upto 70%.

4.4 Energy Efficient Algorithms

A survey of algorithmic techniques to solve energy management is given in the research paper by S albers.The aim of these algorithms to reduce energy consumption without compromising on performance.

IV. EXTENSIONS OVER TRADITIONAL COMPILERS

Cloud based compiler mainly deals with providing a platform to compile and execute programs that is not dependent on any platform related restriction or complication. The cloud compiler can be used by any user who has subscribed to the compiler and can use it for a specific period of time. The functionalities that are provided by the cloud are:-

» **Compile on the Go:** The cloud will have an IDE where the source code can be written. The code can be implemented on the cloud itself and the output can be viewed at same time.

» **File Management:** The source code can be saved in a text format or in .java format in the space that is allocated to each user. The user can extended his storage space by buying a bigger storage space.

» **Forums:** The user can log on into a forum where he can discuss the various problems or the solution to problems he is facing.

» **Security**: We provide by security by implementing a sandbox that is a security mechanism for separating running programs. It is often used to execute untested code, or untrusted programs from unverified third-parties, suppliers, untrusted users and untrusted websites. Also we use signature testing mechanism i.e. a type of antivirus program that searches a system for virus signatures that have attached to executable programs and applications such as e-mail clients. A virus scanner can either search all executable codes when a system is booted or scan a file only when a change is made to the file as viruses will change the data in a file.

The various advantage and disadvantage of the systems are:-

ADVANTAGES

» No need to download the SDK of any compiler.

» No issues of setting path variables.

» Elegant and simple to use GUI for better coding.

» Support from other users in forums.

DISADVANTAGE

» Not cost Effective.

» Requirement of an Internet connections Also we use signature testing mechanism i.e. a type of antivirus program that searches a system for virus signatures that have attached to executable programs and applications such as e-mail clients. A virus scanner can either search all executable codes when a system is booted or scan a file only when a change is made to the file as viruses will change the data in a file.

V. ENERGY EFFICIENT DATA CENTER DESIGN

The power consumption of data centers are 100 to 200 times that of standard office spaces. With such large power consumption, they are prime targets for energy-efficient design measures that can save money and reduce electricity use. The following practices are recommended for energy efficient data center design.

5.1 Efficient servers

Servers represent the largest portion of the IT energy load in a typical data center and drive entire operation. Most of the energy wastage also happens through these servers. Even the percentage of utilization of most of the servers is 20% or below, it draws full power during the process. Recently vast improvements in the internal cooling systems and processor devices of servers have been made to minimize this wasted energy. Using variable speed fans make it possible to delive sufficient cooling with less energy consumption while running slower. The Energy Star program helps consumers by recognizing high-efficiency servers. Servers that meet Energy Star efficiency requirements will, on average, be 30% more efficient than standard servers.

5.2 Storage devices

Power consumption depends on the number of storage modules used. Storage redundancy needs to be rationalized and right-sized to avoid rapid increase in size and power consumption. Consolidating storage drives into a Network Attached Storage or Storage Area Network are two options that take the data that does not need to be readily accessed and transports it

offline. Taking superfluous data offline reduces the amount of data in the production environment, as well as all the copies. Consequently, less storage and CPU requirements on the servers are needed, which directly corresponds to lower cooling and power needs in the data center.

5.3 Network equipment

There are active energy management measures that can be applied to reduce energy usage as network demand varies. Such measures include idle state logic, gate count optimization, memory access algorithms and input/output buffer reduction.

5.4 Power supplies

Most data center equipment uses internal or rack mounted alternating current/direct current (AC-DC) power supplies. Historically, a typical rack server's power supply converted AC power to DC power at efficiencies of around 60% to 70%. Today, through the use of higher-quality components and advanced engineering, it is possible to find power supplies with efficiencies up to 95%. Using higher efficiency power supplies will directly lower a data center's power bills and indirectly reduce cooling system cost and rack overheating issues.

5.5 Virtualization

Virtualization is a method of running multiple independent virtual operating systems on a single physical computer. It is a way of allowing the same amount of processing to occur on fewer servers by increasing server utilization. Instead of operating many servers at low CPU utilization, virtualization combines the processing power onto fewer servers that operate at higher utilization. Virtualization will drastically reduce the number of servers in a data center, reducing required server power and consequently the size of the necessary cooling equipment. Some overhead is required to implement virtualization, but this is minimal compared to the savings that can be achieved.

5.6 Air management

Air management for data centers entails all the design and configuration details that go into minimizing or liminating mixing between the cooling air supplied to equipment and the hot air rejected from the equipment. Effective air management implementation minimizes the bypass of cooling air around rack intakes and the recirculation of heat exhaust back into rack intakes. When designed correctly, an air management system can reduce operating costs, reduce first cost equipment investment, increase the data center's power density (Watts/ square foot), and reduce heat related processing interruptions or failures.

5.7 Electrical systems

Following are general guidelines for delivering electrical power in the most energy-efficient manner possible:

» Minimize the resistance by increasing the cross-sectional area of the distribution path and making it as short as possible.

» Maintain a higher voltage for as long as possible to minimize the current.

» Use switch-mode transistors for power conditioning.

» Locate all voltage regulators close to the load to minimize distribution losses at lower voltages.

VI. FINAL IMPLEMENTATION

The extensive support for deploying PHP based applications in Windows Azure triggered the idea of building the application using PHP. Also the web application to create an online compiler was developed. The code is a server side script which sets the path and gives variety options. The only dilemma lied in the deploying of compilers. The compilers are hosted on

virtual machines created in our windows Azure cloud account. The path is set to these compilers using environment variables on the virtual machine.

VII. Future Work

1) Provide more compilers: The application can be extended to provide compilers for Python, FORTRAN, COBOL, C# etc. Most compilers require authorizing their certain load libraries and System name parameters.

2) Implement a security mechanism: This is an issue that may question the survivability of the application. The user premises are usually secure by firewall protection. And data is secure in the cloud as Azure provides various data confidentiality and integrity mechanisms. Any malicious could be run on the compiler by an attacker to corrupt the entire system directed to the centralized compiler. User authentication could be maintained using passwords, pass Ticket authentication or FTP authentication. A complex system providing Sandbox Testing mechanism can be used to avoid malicious program attacks.

3) Provide efficiency meters and code optimizer add ons: They would provide the complexity of the code and help the user to improve writing better codes by providing a measure to understand their coding skills

4) Create Web API's: We could also provide the above project by using API's in the cloud. This helps to create a more interactive way of providing software as a service.

5) Mobile Applications: Mobile Applications can be developed so that the users can create and execute their applications using their mobiles with greater ease. This would provide them with a better ease of use as they would not have to go through the hassle of logging into a Desktop optimized site through their mobiles.

6) Collaborative Editing: Collaborative editing features can be added so that large project groups can work on the project online and with ease.

VIII. Conclusion

The main reason for creating the project is to provide a centralized compiling scheme. Also, it will act as a centralized repository for all the codes written. The other major advantage that this system will have over the others is that it will make the users system lightweight i.e. there will be no need to maintain separate compilers at the client side. Also, the process of maintenance and distribution of dynamic usernames and passwords will be greatly simplified. Also, authentication and personalized task distribution will be made possible. A compiler, which is the heart of any computing system, transforms source code from a higher level language to a lower, machine level language. This is mainly done in order to create executable files which can then be run in order to execute the program and its instructions. As compared to the current scenario where each compilers required to be installed on each machine separately this would eliminate the need to install compilers separately. So we can check our code at the centralized server. Another advantage of such project is that whenever the compiler package is to be upgraded it can be done easily without again installing it on each and every machine.

References

1. A.Rabiyathul and Basariya k.Tamil Selvi, "Centralized C# Compiler Using Cloud Computing", International Journal of Communications and Engineering Volume 06– No.6,Issue:02 March2012.
2. "Challenges in deploying SaaS applications", Imaginea Inc. white paper i. Kremer U, Department of Computer Science Rutgers University (2002) Low Power/Energy Compiler Optimizations. International conference of Book power aware computing. CRC Press, 2004, ch. 35 ii. Bellas N, Hajj IN, Polychronopoulos CD, Stamoulis G (2000) Architectural and Compiler Techniques for Energy Reduction in High- Performance Microprocessors. IEEE Trans Large Scale Integration (Vlsi) Syst 8(3):317–326
3. Naik K (2010) A Survey of Software Based Energy Saving Methodologies for Handheld Wireless Communication Devices. Tech. Report No. 2010-13. Dept. of ECE, University of Waterloo viii. V. Venkatachalam, M. Franz. "Power reduction techniques for microprocessor systems". ACM Computer Survey, Sempember 2005.
4. U S Department of Energy March 2011 "Best Practices Guide for Energy Efficient Data Center Design" by National Renewable Energy Laboratory.
5. "Future of cloud computing", www.roseindia.net
6. Donovan Kretsman, "SaaS | don't let the Cloud rain on your parade", www.focalscope.com's blog on SaaS

7. "Advantages of SaaS", www.cloudtweaks.com
8. M. Tim Jones, "Anatomy of a cloud storage infrastructure", www.ibm.com
9. Delaluz V, Kandemir M, Vijaykrishnan N, Irwin MJ (2000) Energy- Oriented Compiler Optimizations for Partitioned Memory Architectures.CASES '00 Proceedings of the 2000 international conference on Compilers, architecture, and synthesis for embedded systems.
10. Mehtal H, Owens RM, Irwin MJ, Chen R, Ghosh D (1997) Techniques for Low Energy Software. In: Proceedings of the 1997 international symposium on Low power electronics and design (ISLPED '97). ACM, New York, NY, USA, pp 72–75.
11. B.Zhai et al. "Energy efficient near-threshold chip multiprocessing".Proceedings of the 2007 Int'l Symp on Low Power Electronics and Design Series, ISLPED2007.
12. "Quickly build and deploy Software as a Service applications", Ironspeed Inc. white paper
13. "Tutorials Windows Azure" http://www.windowsazure.com/en-us/develop/net/tutorials/getstarted
14. Online multiple platform provider: www.compilr.com
15. Online compiler: www.cloudcompiling.com

AUTHOR(S) PROFILE

A.Yaganteeswarudu Working as assistant professor in CSE department in Sreenidhi institute of science and technology (Autonomous), Ghatkesar, and Hyderabad. Having 5 Years of experience in teaching. Completed M.Tech in SJCET, Kurnool in 2012.

Y.Vishnuvardhan working as assistant professor in CSE department in Dr.MVSIT, Mangalore. Completed M.Tech in NIT suratkal, Bangalore.

