Energy Efficient Software Development Life Cycle Model

A Thesis Presented to the Faculty of

The Department of Computer Science

Jaypee University of Information Technology

In Partial Fulfilment

Of the Requirements for the Degree of

Master of Technology

In
Computer Science

By

Sunil Kumar Sharma

2012-2014

Under the Direction of

Dr. Pradeep Kumar Gupta

CERTIFICATE

This is to certify that the work titled **"An Approach: Energy Efficient Software Development Life Cycle"** submitted by **"Sunil Kumar Sharma"** in partial fulfillment for the award of degree M.Tech in Computer Science and Engineering of Jaypee University of Information Technology, Wakanghat has been carried out under my supervision. This work has not been submitted partially or wholly to any other university or institute for the award of this or any other degree or diploma.

Signature of Supervisor .....

Name of Supervisor: **Dr. Pradeep Kumar Gupta**

Designation: **Assistant Professor**

Department: **Department of CSE, Jaypee University of Information Technology, Wakanghat.**

Acknowledgement

This may seem long but the task of my thesis work both theoretically and practically may not have been completed without the help guidance and mental support of the following persons.

Firstly, I would like to thanks my guide Assistant Professor, Department of CSE Jaypee University of Information technology Wakanghat, **Dr. Pradeep Kumar Gupta** sir who provided me the idea and related material for the project proposal. He indeed guided me to do the task for my thesis in such a way that it seems to be research work. His continuous monitoring to support me and my research work encouraged me a lot for doing my thesis in very smooth manner.

Secondly, I would like to thanks **my Parents** who have always been with me for inspiring me and thirdly, I would like to thanks **God** for keeping me energetic, healthy and enthusiastic

Once again thanks a ton to all mentioned people in my life

Signature of the student ..  .....

Name of Student: Sunil Kumar Sharma

Date 26-05-2014

Table of Content

CERTIFICATE

This is to certify that the work titled **"An Approach*:* Energy Efficient Software Development Life Cycle"** submitted by **"Sunil Kumar Sharma"** in partial fulfillment for the award of degree M.Tech in Computer Science and Engineering of Jaypee University of Information Technology, Wakanghat has been carried out under my supervision. This work has not been submitted partially or wholly to any other university or institute for the award of this or any other degree or diploma.

Signature of Supervisor ……..

Name of Supervisor: **Dr. Pradeep Kumar Gupta**

Designation: **Assistant Professor**

Department: **Department of CSE, Jaypee University of Information Technology, Wakanghat.**

Acknowledgement

This may seem long but the task of my thesis work both theoretically and practically may not have been completed without the help guidance and mental support of the following persons.

Firstly, I would like to thanks my guide Assistant Professor, Department of CSE Jaypee University of Information technology Wakanghat, **Dr. Pradeep Kumar Gupta** sir who provided me the idea and related material for the project proposal. He indeed guided me to do the task for my thesis in such a way that it seems to be research work. His continuous monitoring to support me and my research work encouraged me a lot for doing my thesis in very smooth manner.

Secondly, I would like to thanks **my Parents** who have always been with me for inspiring me and thirdly, I would like to thanks **God** for keeping me energetic, healthy and enthusiastic

Once again thanks a ton to all mentioned people in my life

Signature of the student ……..

Name of Student: Sunil Kumar Sharma

Date ……..

Abstract

We think that computers are non-polluting and consume a little energy. But at this time we are wrong, because according to a survey $ 250 billion per year spent on powering computers worldwide only about 50% of that power is spent computing – the rest is wasted idling. [1]

In this thesis we are proposing our own software development life cycle modal which is composed by six stages. These stages are Green requirement analysis, Green design, Green implementation, Green testing, Green maintenance and green analysis. The last stage green analysis is very important because all the life cycle stages are similarly connected to the green analysis phase. The reason behind it When one stage is completed, then green analysis phase check that, this stage follow the green computing rules or not.

In green analysis phase, we will use the different important concept like green data centre, virtualization, cloud computing, power optimization, grid computing.

Our green modal follows the iterative approach instead of the sequential approach, because if we want to change in any phase then we can do easily. But this is the disadvantage of the linear sequential life cycle modal. So we will refer to iterative life cycle modal.

List of Figures

Figures **Page No.**

List of Tables

CHAPTER 1

Introduction

Today computer play an important role in our personal and business life. According to Lamb J. "Life without information technology will become paralyzed". So different organization are doing work in the information technology for their performance growth and value. If organizations do not invest in the technology then they will lose their market place [1]. Furthermore *IT* has increases the environmental issue and problem from e-waste disposal, energy requirements and production [1]. So we can say that increasing information technology has side effects on the environment too. The awareness of these effects has attracted some renowned organization towards the environmental friendly computing and practices known as "Green Computing".

Green computing is the environmentally responsible use of computers and its related resources. Green computing includes the implementation of the energy efficient software, hardware, peripherals, servers, as well as reduces the resource consumption and e-waste. There are different types of programmes are running for green computing in different countries. The United States was the voluntary labelling program known as "Energy Star". It was organized by the Environmental Promote Agency (*EPA*) in 1992.This programme promote energy efficiency in all types of hardware.

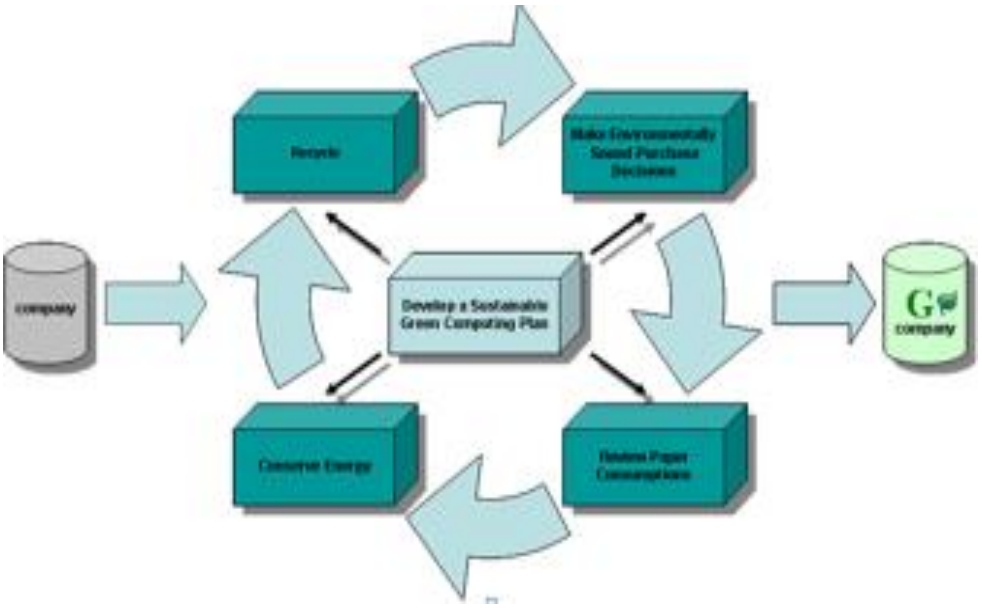


Figure 1 Industries work in green computing

As population has increased energy use is also increased. The prevalent use of technology sceptically computers the requirements of energy are increasing day by day. It means computers play an important role of increasing demand of energy in the word. So computer Power consumption is more important topic electricity prices climb. To make the computer eco-friendly there are four type of approach is given:

Figure 2 Green computing main parameters

1. Green Use: In this approach use such type of software, hardware, peripherals in which required less energy and all these are eco-friendly.
2. Green Disposal: In this approach Re-making existing computer. It means reduce the hazards materials. Use such types of materials which are easily recycled and easy to dispose.
3. Green Design: In this approach designing energy-efficient computer, server, printers, projectors and other digital devices.
4. Green Manufacturing: In this approach manufacture such type of computers and Sub-system which minimize the waste and reduce environment impact.

1.1 History of Green Computing

In 1992, the U.S. Environmental Protection Agency launched Energy Star, a voluntary labelling program which is specially designed to endorse and to identify energy-

efficiency in monitors, environment control equipment, and other technologies. This resulted in the well-known acceptance of sleep mode among consumer electronics. The word "green computing" was probably coined shortly after the Energy Star program began; there are a number of USENET posts dating back to 1992 which use the term in this manner. simultaneously, the Swedish organization TCO Development launched the TCO_Certification program to encourage low magnetic and electrical emissions from CRT-based computer displays; this program was later expanded to include criteria on energy consumption, ergonomics, and the use of hazardous materials in construction. When it comes to PC dumping, it is compulsory to know everything there is to know in order to be involved in green computing. Basically, the all green aspect came about quite a few years back when the news that the environment was not a renewable resource really hit home and people started realizing that they had to do their part to protect the environment.

Basically, the well-organized use of computers and computing is what green computing is all about. The triple bottom line is what is important when it comes to anything green and the same goes for green computing. This considers social responsibility, economic viability and the impact on the environment. Many businesses simply focus on a bottom line, rather than a green triple bottom line, of economic viability when it comes to computers. The idea is to make the whole process surrounding computers friendlier to the environment, economy, and society. This means manufacturers construct computers in a way that reflects the triple bottom line positively. Once computers are sold businesses or people use them in a energy efficient way by reducing power usage and disposing of e-wastage properly or recycling them. The purpose is to make computers a green product.

1.2 Regulations and Industry Initiative

1.2.1 From the Government

Many government agencies are working to implement standards and regulations that promote green computing. The Energy Star program was launched in October 2006 to include stricter efficiency requirements for computer equipment The European Union's directives 2002/95/EC (RoHS), on the reduction of hazardous substances, and 2002/96/EC (WEEE) on waste electrical and electronic equipment required the replacement of heavy metals and flame retardants like PBBs and PBDEs in all

electronic equipment put on the market starting on July 1, 2006. The directives placed responsibility on manufacturers for the gathering and recycling of old equipment (the Producer Responsibility model).

1.2.2. From the Industry

- **Climate Savers Computing Initiative :** CSCI is an attempt to reduce the electric power consumption of computer in working and inactive states. The CSCI provides a list of green products from its member organizations, and information for reducing Computer power consumption. It was started on 2007-06-12.
- **Green Computing Impact Organization, Inc.** GCIO is a non-profit organization dedicated to supporting the end-users of computing products in being environmentally responsible. This assignment is accomplished through educational programme, cooperative events and subsidized auditing services. The heart of the group is based on the GCIO Cooperative, a community of environmentally concerned IT leaders who pool their time, resources, and buying power to educate, broaden the use, and improve the efficiency of, green computing products and services
- **Green Electronics Council :** The GEC offers the Electronic Products Environmental Assessment Tool (EPEAT) to assist in the purchase of "green" computing systems. By the help of 28 criteria Council evaluates computing equipment's. These criteria measure a product's efficiency and sustainability attributes. On 2007-01-24, President George W. Bush issued Executive Order 13423, which requires all United States Federal agencies to use EPEAT when purchasing computer systems.
- **The Green Grid :** This is a global association dedicated to advancing energy efficiency in data centres and business computing eco-systems. It was founded in February 2007 by a number of key companies in the industry – AMD, APC, Dell, HP, IBM, Intel, Microsoft, Rack able Systems, Spray Cool, Sun Microsystems and VMware. The Green Grid has since grown to hundreds of members, including end users and government organizations, all motive of improving data centre efficiency.

1.3 Approaches to Green Computing

1.3.1 Virtualization

Computer virtualization refers to the abstraction of computer related resources, for example the process of running two or more logical computer systems on one set of physical hardware. The concept originated with the IBM mainframe operating systems of the 1960s, but was commercialized for x86-compatible computers only in the 1990s. By the help of virtualization, a system administrator could combine several physical systems into virtual machines on one single, powerful system, thereby unplugging the original hardware and reducing power and cooling consumption. Many commercial companies and open-source projects now offer software packages to enable a transition to virtual computing. Intel Corporation and AMD have also built proprietary virtualization enhancements to the x86 instruction set into each of their CPU product lines, in order to facilitate virtualized computing.

In case of server consolidation, many small physical servers are replaced by one larger physical server, to proper utilization of expensive hardware resources such as CPU. Although hardware is consolidated, typically Operating System is not. Instead, each Operating System running on a physical server becomes converted to a distinct OS running inside a virtual machine. The large server can "host" many such "guest" virtual machines. This is called Physical-to-Virtual (P2V) transformation. Virtual machine can be easily controlled and inspected from outside than a physical one; the configuration of VM is also more flexible than physical. This is very helpful in kernel development and for teaching operating system courses.

A new virtual machine can be provisioned as needed without the need for up-front hardware purchase. Also, virtual machine can be easily re-located from one physical machine to another as needed. For example, a sales person going to a customer can copy a virtual machine with the demonstration software to its laptop, without the need to transport the physical computer. At the same time and error inside a virtual machine does not harm a host system, so there is no risk of breaking down the OS in said laptop.

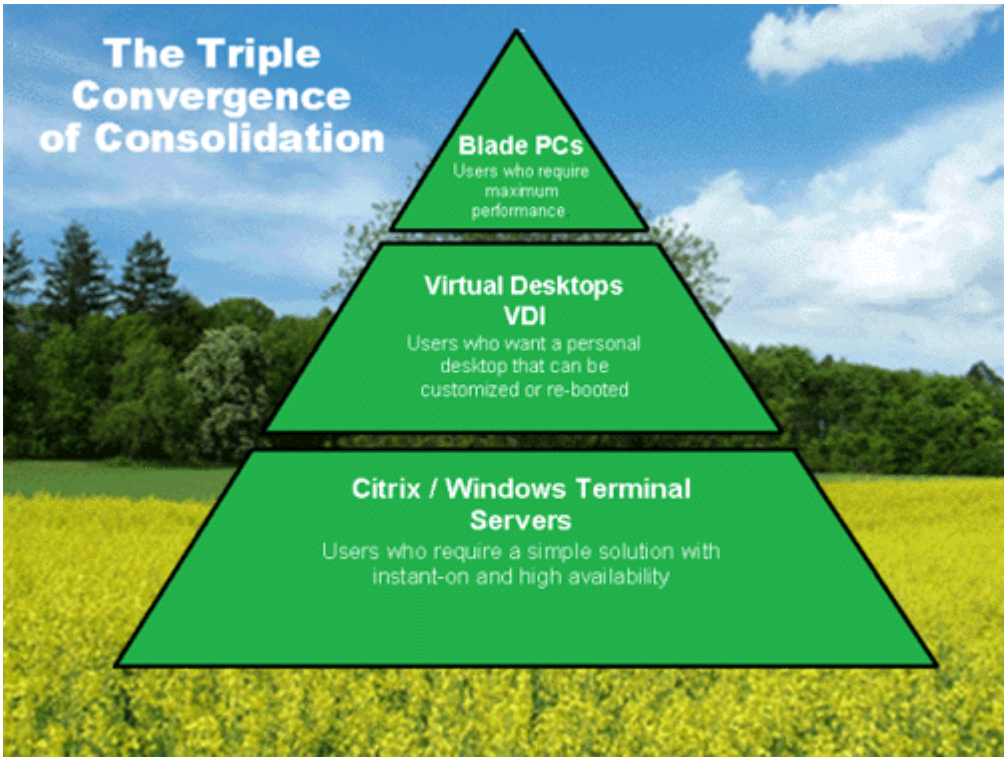


Figure 3 The triple convergence of consolidation

1.3.2 Material management

- **RoHS**

In February 2003, the European Union adopted the Restriction of Hazardous Substances_Directive (RoHS). The legislation restricts the use of six hazardous materials in the manufacture of various types of electronic and electrical equipment. The directive is closely linked with the Waste Electrical and Electronic Equipment Directive (WEEE), which sets collection, recycling, and recovery targets for electrical goods and is part of a legislative initiative that aims to reduce the huge amounts of toxic e-waste**.** Driven by these directives, VIA implemented a set of internal regulations in order to develop products that are compliant with these accepted policies, including the use of non-hazardous materials in its production of chipsets, processors, and companion chips. In 2001, they focused on lead-free manufacturing, introducing the Enhanced Ball Grid Array (EBGA) package for power efficient VIA processors and the Heat Sink Ball Grid Array (HSBGA) package for their chipsets. In traditional manufacturing processes, lead is used to attach the silicon core to the inside of the package and to facilitate integration onto the motherboard through tiny solder balls on the underside of the package. VIA's

lead-free manufacturing technologies do not require a lead bead, and the solder balls now consist of a tin, silver, and copper composite.

However, not everyone is satisfied with this new objective. Howard Johnson of the magazine says that the move toward lead-free devices is not only unhelpful but actually worse for the environment. "The additional tin mining required to produce high-purity tin alloys, plus the mining of other precious metals required to alloy with tin in substitution for lead, is a poor trade for the use of existing lead, much of which comes from recycled products," Johnson writes. He also believes that lead-free assembly is less reliable than lead-based assembly, partially due to the increased growth of tin whiskers — small, hair-like metallic growths that naturally emerge from the surface of solid tin. On lead-free tin surfaces, these whiskers can grow to a length sufficient to short an electronic circuit to another, leading to product failure.

- **Energy efficient Computing**

- Do not leave your computer running overnight and on weekends. Also, wait until you are ready to use it before you turn it on.
- A modest amount of turning on and off will not harm the computer or monitor. The life of a monitor is related to the amount of time it is in use, not the number of on and off cycles.
- Try to plan your computer-related activities so you can do them all at once, keeping the computer off at other times.
- Do not turn on the printer until you are ready to print. Printers consume energy even while they are idling.
- Do not print out copies of email unless necessary.
- If you spend a large amount of time at your computer, consider reducing the light level in your office. This may improve CRT (cathode ray tube) screen visibility as well as save energy.
- Most computer equipment now comes with power management features. If your computer has these features, make sure they are activated.
- The best screen saver is no screen saver at all - turn off your monitor when you are not using it. This option is second best only to turning off your computer all together.

- Use "paperless" methods of communication such as email and fax-modems.
- When typing documents, especially drafts, use a smaller font and decrease the spacing between lines, or reformat to keep your document to as few pages as possible, especially when typing drafts.
- Review your document on the screen instead of printing a draft. If you must print a draft, use the blank back side of used paper.
- Use a printer that can print double-sided documents. When making copies, use double-sided copying.
- Always buy and use recycled-content paper. Look for papers with 50-100% post-consumer waste and non-chlorine bleached. Also, recycle your paper when done.
- Buy a monitor only as large as you really need. Although a large monitor might seem more attractive, you should remember that a 17-inch monitor uses 40 percent more energy than a 14-inch monitor. Also, the higher the resolution, the more energy it needs.
- Ink-jet printers, though a little slower than laser printers, use 80 to 90 percent less energy.
- Request recycled / recyclable packaging from your computer vendor.
- Buy vegetable (or non-petroleum-based) inks. These printer inks are made from renewable resources; require fewer hazardous solvents; and in many cases produce brighter, cleaner colours.

➢ **Recycling**

Old-fashioned computers are an important source for secondary raw materials, if treated properly, however if not treated properly they are a major source of toxins and carcinogens. Rapid technology change, low initial cost and even planned obsolescence have resulted in a fast growing problem around the globe. Many materials used in the construction of computer hardware can be recovered in the recycling process for use in future production. Reuse of tin, silicon, iron, aluminium, and a variety of plastics – all present in bulk in computers – can reduce the costs of constructing new systems. In addition, components frequently contain copper, gold, and other materials valuable enough to

reclaim in their own right. Electronic devices, including audio-visual components (televisions, VCRs, stereo equipment), mobile phones and other hand-held devices, and computer components, contain valuable elements and substances suitable for reclamation, including lead, copper, and gold. They also contain a plethora of toxic substances, such as dioxins, PCBs, cadmium, chromium, radioactive, and mercury.

Whole computers and pieces of electronic equipment are shredded into smaller pieces to be more manageable and facilitate the separation of the constituent components. Leaded glass from cathode ray tubes is sold to foundries for use as a fluxing agent in the processing of raw lead ore. Other valuable metals, such as copper, gold, palladium, silver and tin are sold to smelters for metal recycling. The hazardous smoke and gases generated by these processes are captured, contained, and treated to ensure that they do not become a threat to the environment. These methods allow for the safe reclamation of all the valuable materials used in computer construction.

1.3.3 Telecommuting

Telecommuting, e-commuting, e-work, telework, working at home (WAH), or working from home (WFH) is a work arrangement in which employees enjoy flexibility in working location and hours. In other words, the daily commute to a central place of work is replaced by telecommunication links. Many work from home, while others, occasionally also referred to as nomad workers or web commuters utilize mobile telecommunications technology to work from coffee shops or myriad other locations**.** Telework is a broader term, referring to substituting telecommunications for any form of work-related travel, thereby eliminating the distance restrictions of telecommuting. All telecommuters are teleworkers but not all teleworkers are telecommuters. A frequently repeated motto is that "work is something you do, not something you travel to". A successful telecommuting program requires a management style which is based on results and not on close scrutiny of individual employees. This is referred to as management by objectives as opposed to management by observation. The terms *telecommuting* and *telework* were coined by American Jack Nilles in 1973.

Long distance telework is facilitated by such tools as virtual private networks, videoconferencing, and Voice over IP. It can be efficient and useful for

companies as it allows staff and workers to communicate over a large distance, saving significant amounts of travel time and cost.
As broadband Internet connections become more commonplace, more and more workers have enough bandwidth at home to use these tools to link their home office to their corporate intranet and internal phone networks.

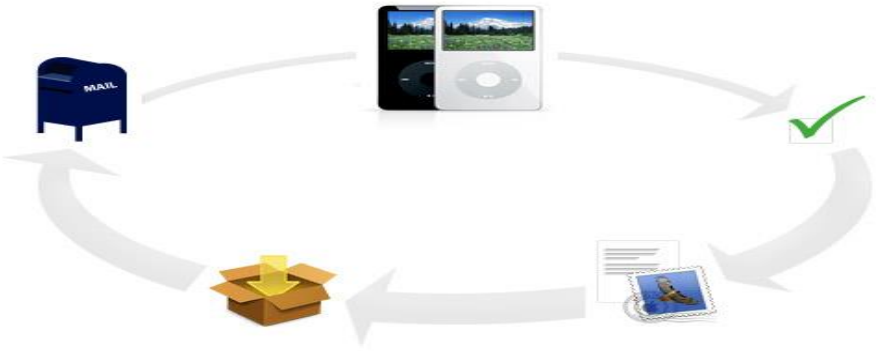
1.3.4 VoIP

Figure 4 VoIP

Voice over Internet Protocol (VoIP) is a general term for a family of transmission technologies for delivery of voice communications over the Internet or other switched networks. The reduction in telephone wiring will obviously lead to decreasing costs because of Voice-Over-Internet protocol. Voice over IP (VoIP) reduces the telephony wiring infrastructure by sharing the existing Ethernet copper, thus reduce the use of metallic waste. VoIP and phone extension mobility also made Hot-disking and more practical.

1.4 Role of IT vendors

- ***Apple***



Four areas of particular attention are product and packaging design, materials, energy efficiency, and recycling. Each aspect of the design cycle provides significant challenges, yet our efforts in these areas have resulted in some impressive results.

Product design: It all begins here. Reducing the environmental impact of our products starts with the product design phase. Design dictates the quantity of raw materials as well as the type and recyclability of materials used. It also determines how much energy is consumed during manufacturing and product use. For example, the amazingly slim 20-inch iMac is made from highly recyclable glass and aluminium and it is so energy efficient it consumes about the same amount of power as a standard light bulb when on.

Materials: Apple helps to safeguard the environment - as well as consumers' safety - by restricting the use of environmentally harmful compounds in our materials and manufacturing processes. In addition to the substances that have already been restricted or eliminated, Apple is removing elemental forms of bromine and chlorine from our products, not just polyvinyl chloride (PVC) and brominates flame retardants (BFRs). The new Mac Book family also uses mercury-free light-emitting diode (LED) displays, with arsenic-free display glass.

Energy efficiency: A devices greatest contribution to greenhouse gas emissions comes from its consumptions of energy over time. Apple has made great strides in recent years to optimize the energy efficiency of our hardware and created tools, such as the Energy Saver feature in Mac OS X, that allow consumers to

manage the power consumption of their computers. Since 2001, Apple desktop computers, portable computers, and displays have earned the ENERGY STAR rating.

Recycling : Apple's holistic, lifecycle approach to recycling includes using highly recyclable materials in products in addition to providing extensive take-back programs that enable consumers and businesses to safely dispose of used Apple equipment. Since our first take-back initiative began in Germany in 1994, we have instituted recycling programs in 95 percent of the countries where our products are sold - diverting over 53 million pounds of electronic equipment from landfills worldwide. Apple is on track to eliminate toxic chemicals from our products. In the2008 Environmental Update Steve Jobs provides an overview on Apple's progress to eliminate mercury and arsenic from displays and Brominated Flame Retardants (BFR's) and Polyvinyl Chloride (PVC) from internal components. Steve Jobs also talks about Apple's policy on climate change, steps taken to improve product energy-efficiency as well as overall recycling performance during 2007.

- ***Wipro***

Wipro Limited, a leading player in Global IT and R&D services, is committed towards environmental sustainability by minimizing the usage of hazardous substances and chemicals which have potential impact on the ecology. It has joined hands with WWF India, one of the largest conservation organizations in the country, to directly deal with issues of climate change, water and waste management and biodiversity conservation.

Figure: 5 Wipro's diverse portfolio of green solutions

- **Green Lighting Solutions**
  - ❖ Complete range of Brightness Management Products for Green Buildings
  - ❖ Ability to integrate lighting and lighting management systems for Green Building performance standards
  - ❖ Role of Lighting for GREEN buildings: 17% – 20% of the overall building's energy usage
  - ❖ Optimize Energy Performance
  - ❖ High efficiency luminaries design
  - ❖ High efficiency light sources - Compact Fluorescent Lamp, LED, etc.
  - ❖ Lighting controls
  - ❖ High efficiency control gear
  - ❖ Personalized controls through task lighting Intelligent lighting systems
- **Green IT Solutions Applications**
  - ❖ E-Freight –An innovative application for the Air Cargo industry that enables efficient, multi-format & paperless interaction between Airlines, Freight Forwarder and Customs
  - ▪ Emission Compliance Management System
  - ❖ An application developed for manufacturing companies
  - ❖ Helps them to control pollution & reduce carbon monoxide emissions
  - ❖ Energy Efficiency Solution
  - ❖ A process & technology application that accommodates the functionality requirements of an end-to-end energy efficiency solution
  - ❖ It is a framework that is designed to help customers to use their energy requirements in the most-cost effective manner
- **Products**
  - ❖ Wipro Green ware
  - ❖ RoHS Compliant (Restriction of Hazardous Substances)
  - ❖ Energy star certification
  - ❖ Energy Conservation mechanism in electronic components
  - ❖ Compliant with environment & safety standards and statutory regulations
  - ❖ Recyclable & degradable packing materials
  - ❖ MPR II certified radiation free monitors
  - ❖ Wipro WEEE Statement

- ❖ Part of 'Quick Start Guide' shipped with all systems from factory
- ❖ WEEE – Waste from Electrical and Electronic Equipment

- **Services**

- e-Waste Disposal Service
  - ❖ Offering a facility to collect retired computers, laptops & servers from willing customers and to dispose them off in a responsible manner
- Eco-friendly Product Engineering Designs
  - ❖ Eco-friendly Engineering Designs that are RoHS compliant & energy efficient
  - ❖ For Telecom & Embedded solution customers
  - ❖ With state-of-the-art labs for environmental testing
- Green Data Centre Energy consumption & Cost are the drivers due to:
  - ❖ Increase in computing demand
  - ❖ Changing cost dynamics
  - ❖ Data Centre Life Cycle Mismatch
  - ❖ Wipro's service offering – Build / upgrade into a Green Data Centre
- **Manage IT Infrastructure**
  - ❖ Optimize server operations & reduce floor footprint
  - ❖ Implement remote monitoring for increased efficiency and improved management
- **Green Testing Lab**
  - Wipro has set up a hardware lab in its Sarjapur campus that will exclusively test products to confirm that they are 'green' compliant.
  - The idea is to maintain & uphold the environmental standards by the Government & Society.
  - Virtualization of Testing.
  - Server Consolidation

- ❖ Allows to run multiple heterogeneous operating systems OR versions of same operating systems simultaneously on a single server —without partitioning or rebooting. It consolidates workload of several under-utilized servers to fewer machines, perhaps a single machine
- ❖ Reduces Cost

- ❖ Reduces cost involved in hardware resources, power, cooling, commercial space & maintenance
- ❖ Reduces Testing Time
- ❖ Simplifies & reduces testing effort

- **Shared Service Consulting**

- ❖ A practitioner's perspective
- ❖ Wividus - A shared service organization
- ❖ Supports 80000+ employees; handles 4000+ transactions per day
- ❖ Conserving resources – paper (95% electronic transactions) & energy
- ❖ Sharing service resources across Wipro businesses
- ❖ Applying Six Sigma & Lean Concepts

➢ ***Google***

Google's mission is to organize the world's information and make it universally accessible and useful. Hundreds of millions of users access our services through the web, and supporting this traffic requires lots of computers. We strive to offer great internet services while taking our energy use very seriously. That's why, almost a decade ago; we started our efforts to make our computing infrastructure as sustainable as possible. Today we are operating what we believe to be the world's most efficient data centres.

The graph below shows that our Google-designed data centers use considerably less energy - both for the servers and the facility itself - than a typical data centre. As a result, the energy used per Google search is minimal. In fact, in the time it takes to do a Google search, your own personal computer will use more energy than we will use to answer your query.

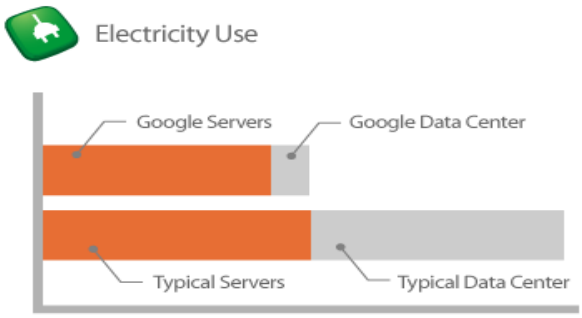


Figure 6 Electricity use

But sustainability is about more than electricity, so we've gone beyond just reducing our energy consumption. Before the end of 2008 two of our facilities will run on 100% recycled water, and by 2010 we expect recycled water to provide 80% of our total water consumption. We also carefully manage the retirement of our servers to ensure that 100% of this material is either reused or recycled. Finally, we are engaging our users and peers to help build a clean and efficient energy future. This broader impact could be significant; if all data centres operated at the same efficiency as ours, the U.S. alone would save enough electricity to power every household within the city limits of Atlanta, Los Angeles, Chicago, and Washington, D.C.

Sustainability is good for the environment, but it makes good business sense too. Most of our work is focused on saving resources such as electricity and water and, more often than not, we find that these actions lead to reduced operating costs. Being "green" is essential to keeping our business competitive. It is this economic advantage that makes our efforts truly sustainable.

Google's five step plan

1. Minimize electricity used by servers
2. Reduce the energy used by the data centre facilities themselves
3. Conserve precious fresh water by using recycled water instead
4. Reuse or recycle all electronic equipment that leaves our data centres
5. Engage with our peers to advance smarter energy practice

- ***VIA***

VIA Technologies, a Taiwanese company that manufactures motherboard chipsets, CPUs, and other computer hardware, introduced its initiative for "green_computing" in 2001. With this green vision, the company has been focusing on power efficiency throughout the design and manufacturing process of its products. Its environmentally friendly products are manufactured using a range of clean-computing strategies, and the company is striving to educate markets on the benefits of green computing for the sake of the environment, as well as productivity and overall user experience.

- **Carbon-free computing :** One of the VIA Technologies' ideas is to reduce the "carbon footprint" of users — the amount of greenhouse gases produced, measured in units of carbon dioxide (CO2) VIA aims to offer the world's first PC products certified carbon free, taking responsibility for the amounts of CO2 they emit. The

company works with environmental experts to calculate the electricity used by the device over its lifetime, generally three years.

- **Solar computing:** Amid the international race toward alternative-energy sources, VIA is setting its eyes on the sun, and the company's Solar Computing initiative is a significant part of its green-computing projects. For that purpose, VIA partnered with Motech Industries, one of the largest producers of solar cells worldwide. Solar cells fit VIA are power-efficient silicon, platform, and system technologies and enable the company to develop fully solar-powered devices that are non-polluting, silent, and highly reliable. Solar cells require very little maintenance throughout their lifetime, and once initial installation costs are covered, they provide energy at virtually no cost. Worldwide production of solar cells has increased rapidly over the last few years; and as more governments begin to recognize the benefits of solar power, and the development of photovoltaic technologies goes on, costs are expected to continue to decline. As part of VIA's "pc-1" initiative, the company established the first-ever solar-powered cyber community centre in the South Pacific, powered entirely by solar technology.

Figure 7: Solar Computing

- **Lead-Free and RoHS computing:** In February 2003, the European Union adopted the Restriction of Hazardous Substances Directive (RoHS). The legislation restricts the use of six hazardous materials in the manufacture of various types of electronic and electrical equipment. The directive is closely linked with the Waste Electrical and Electronic Equipment Directive (WEEE), which sets collection, recycling, and recovery targets for electrical goods and is part of a legislative initiative that aims to reduce the huge amounts of toxic e-waste**.** Driven by these directives, VIA implemented a set of internal regulations in order to develop products that are compliant with these accepted policies,

including the use of non-hazardous materials in its production of chipsets, processors, and companion chips. In 2001, they focused on lead-free manufacturing, introducing the Enhanced Ball Grid Array (EBGA) package for power efficient VIA processors and the Heat Sink Ball Grid Array (HSBGA) package for their chipsets. In traditional manufacturing processes, lead is used to attach the silicon core to the inside of the package and to facilitate integration onto the motherboard through tiny solder balls on the underside of the package. VIA's lead-free manufacturing technologies do not require a lead bead, and the solder balls now consist of a tin, silver, and copper composite

- ***IBM***

In May 2007, IBM unveiled Project Big Green -- a re-direction of $1 billion USD per year across its businesses to increase energy efficiency. New products and services are expected to reduce data centre energy consumption and transform clients' technology infrastructure into "green" data centres, with energy savings of approximately 42 percent for an average data centre. As part of Project Big Green, IBM is building an $86 million green data centre expansion at its Boulder, Colorado location and will consolidate nearly 4,000 computer servers in six locations worldwide onto about 30 refrigerator-sized mainframes running the Linux operating system.

Project Big Green outlines a five-step approach for data centres that is designed to improve energy efficiency:

1. Diagnose: evaluate existing facilities -- energy assessment, virtual 3-D power management and thermal analytics.
2. Build: plan, build or update to an energy efficient data centre.
3. Virtualize: Virtualize IT infrastructures and special purpose processors.
4. Manage: seize control with power management software.
5. Cool: exploit liquid cooling solutions -- inside and out of the data centre.

By investing in systems that deliver better performance per watt, businesses can make significant long-term savings and reduce their carbon footprint. Project Big Green invests in delivering continual advances in power-performance for each new generation of its server and storage technologies, enabling clients to

run the same business workload at lower cost and with reduced environmental impact

- ***Sony***

Sony has developed an environmentally friendly prototype battery that runs on sugars and can generate enough electricity to power a music player and a pair of speakers, the Japanese company said. The bio battery's casing is made of a vegetable-based plastic. It measures an inch and a half along each edge and works by pouring sugar solution into the unit; where enzymes break it down to generate electricity. Test cells had an output of 50 mill watts. Sugar is a naturally occurring energy source produced by plants through photosynthesis. It is therefore regenerative, and can be found in most areas of the earth, underlining the potential for sugar-based batteries as an ecologically friendly energy device of the future.

1.4 Green Computing Tips

- Use LCD monitors instead of CRT monitors, which consume a lot more electricity. LCD monitors uses three times less when active, and ten times less energy when in sleep mode.
- Use laptops instead of desktop computers, also cuts down on energy usage. The Everex Step Note NC1501 is touted as the world's most energy efficient notebook computer, using only 12W peak power. By comparison, a desktop model uses 200-400 watts.
- If a laptop is not feasible, look for the Energy Star label when purchasing a computer. New US government regulations make this more important than it's been for the past fifteen years.
- Disable your screen saver. Burn-in is not an issue with modern monitors, and screen savers can prevent your monitor and computer from going into idle/sleep mode.
- Enable the power management features on your computer, to turn off components such as the monitor, fans and hard drive when idle. On Windows, go to Control Panel / Power Options. On OS X, go to System Preferences / Energy Saver.
- Switch off the monitor, printer, scanner and other peripherals when not in use.

- **Tips for Green Home Computing:**
  - Don’t check your email on a PC as far as possible – use a mobile device
  - Never leave your PC switched on at the wall, or on standby
  - Take that CRT monitor to the recycling centre
  - Always switch off speakers, modem, monitor at the wall socket if not using
  - Use natural ventilation in the computer room
  - Only connect to the internet when you know you will use the connection
  - Get all family members to log on to the WiFi network at the same time
  - Consider buying a newer, more energy efficient computer or low power notebook
  - Surf at cafes where they only have a single WiFi modem
- **Tips for Green Office Computing**:
  - User blade servers that run very low temperature chips to save cooling
  - Tell employees to switch everything off at night
  - Use smart thermostats in the server room to save cost
  - Use low power thin client PCs that use on-demand applications
  - Switch to LCD screens to cut power usage
  - Only buy Green label PCs and hardware that can be completely recycled.
  - Recycle all internal paper, and reprint on the back of used single side waste.

Chapter 2

Literature Review

Energy Conservation has become a worldwide issue in now days. Because if the requirement of the energy is less, than cost is automatically is reduce. As we know that the computer required more energy to perform the task. If computer required more power or energy, than industries or organizations will be demanding a lot of energy. If demand of energy is increasing than cost of energy is also increasing. So far sake of world environment it is necessary to decrease the demand of energy by reducing the power consumption of computer.

So now day's energy conservation has become a critical issue in Morden electronic system. In computer system energy wastage is occur in both software and hardware .Green computing proposed the solution of energy conservation by application of different techniques at software and hardware levels. Optimization for energy conservation can be made for software and hardware levels.

The hardware level optimization for energy conservation is mostly achieved by the Electrical engineer. Because hardware level optimization are achieved through circuit design by implementing smaller silicon process geometries, auto idle detection circuits and active well biasing techniques[2]. Software level optimization for energy conservation is mostly achieved by software engineer. The software optimization is achieved in operating system, through green complier and green scheduler. Another way for energy conservation is construct the energy efficient graphical user interface *(GUI*) design. In this the researcher provides specific suggestion to *GUI* design [6].

The other way for energy conservation is software development life cycle (*SDLC*).In this we can use such type of matrices and tools which reduce the energy consumption in *SDLC* cycle.

So we can say that there are different other areas in computer in which we can reduce the energy consumption.

Because I am software engineer, so I will do the work in the field of computer software. As we know that software development cycle is the frit step of any software which is running on the computer.

So in this thesis my problem is how to reduce the energy consumption in software development cycle (*SDLC*).Different researcher provide framework of *SDLC,* which reduce the energy consumption. I will refer these all research paper for finding the solution of my problem.

2.1 Why Conduct a Literature Review

Before explaining how the literature has been conducted, it is necessary to justify why literature review has been chosen for this study. Literature review is conducted for variety of reasons, as follows:

- This thesis report must raise the IT community's current understanding and knowledge about Green IT and contribution of exploration in the current Body of Knowledge (BoK) (Levy and Ellis, 2006).
- This literature review brings the attention of IT community where more research work is required and what is needed to be known.
- The literature review methodology is chosen to give collateral evidence of the research problem.
- One of the reasons for doing literature review is to ensure the validity of the evaluated results.
- Furthermore literature review builds a strong theoretical foundation from available resources (See Table 1 for resources) which helps to explain the problem with strong arguments and reasons (Levy and Ellis, 2006).
- Additional reasons for using the literature review are; to justify the significance of the information security problem in Green IT; to develop the relationship between Green IT ideal solutions and actual practices of it.
- This literature review approach also helps to identify the recommendations for future research about how to make Green IT more secured from information security perspective and what controls need to be implemented and what additional care to be taken while implementing practicing Green IT Literature Review Process.
- The systematic literature review has been chosen because it ensures that complete relevant literature about Green IT has been gathered. One of the step of literature review process, known as, literature input process, gives a very good sign about literature gathering completion when you are not finding as new concept and thoughts. Webster and Watson (2002) also say, "A

systematic search should ensure that you accumulate a relatively complete census of relevant literature.

2.2 Literature Review Input

This section describes how literature has been search and gathered, with the help of specific approaches and techniques, introduced by Levy and Ellis. Literature review input process is the foundation of a quality literature review which is deep and broad, rigor and consistent, valid and clear, effective and synthesize, not a simple compilation of related material. If literature input is wrong, or of low quality, impertinent or inappropriate then whatever the data analysis or evaluation method is used, a quality and valid result cannot be achieved.

2.3 Literature Resources

It is true that importance of past literature resources cannot be denied. The previous research work and studies helps the other researchers to take advantage of it while deriving the new knowledge. To take advantage of previous work, it is necessary to look for literature resources (See Table 1) because a literature from quality resource can be confidently referred in owns new concepts for various purposes for example, to give direction to the reader or to prove the validity of the study. Table 1 is the list of databases, where the searching has been carried out. Important criterion for searching the relevant material inside and outside the IT/IS outlets, has been followed as discussed in the following sub section “Search Techniques”, where searched keywords are searched in all fields including the full text.

Table 1: List of Literature Databases

1.	ACM (Digital Lab
2.	EBSCHost
3.	Elsevier (ScienceDirect)
4	Google Scholar
5.	IEEE (Comp Soc&Xplore)
6.	ProQuest (ABI/INFORM)
7.	SAGE
8.	Springer

2.4 Research Parameter

In order to gather manuscripts relevant to the subject matter, under investigation, following different high level keywords of Green IT are used for searching which are as follows:

- Green Computing
- Green IT
- Climate Saver Computing
- Green Threat
- Green Technologies
- Going Green
- Green Grid
- Green IT threat to security
- Green IT and Information security

Levy and Ellis (2006, p.190) and Webster and Watson (2002) suggestion, about the keyword search, has also been followed. Different keyword or phrase has been used to search the literature. Buzzwords have been avoided as a keyword that appear and disappear in the literature. Search technique is not stick to a specific keyword. Further techniques have been discussed in below sub heading.

2.5 Search Techniques

To achieve the high degree of literature quality, following search techniques has been followed:

- Searching has been started from the Journals guided by Levy and Ellis (2006, figure. 2) and Webster & Watson (2002).
- Selected conference proceeding compiled by (Levy and Ellis, 2006, figure. 3) has also searched for the applicable literature. The literature input has also been gathered from umber of literature database vendors.
- Backward and Forward search techniques, (Webster and Watson; 2002 and Levy and Ellis 2006) has been used.
- Most the searched worked is carried out electronically. Except the few books which are borrowed from the library or some purchased articles.

2.6 Literature Review

After searching the literature review input, the second daunting task is to manage the gathered literature for data analysis and evaluation. All electronically searched literature is primarily separated according to their subject matter. Different electronic folder was maintained, to keep the same subject matters aligned. This management technique helps me to look only into that folder which I required for literature analysis and writing. In green computing most of the research work has mainly focus on environmental sustainability in term of computer hardware. But as we know that from the help of previous paper, software is play important role in energy consumption. Software features are responsible for co2 emission; software has an indirect effect on the environment by operating and managing the underlying hardware running on it [1].

"A Green model for sustainable software engineering" presented by Sara s. Mahmud et all paper focus on achieving green and sustainable software by building a green and sustainable software model, that will aid software engineers in development process of software and include the recent approaches taken by software to insure the safely of the environment.

This paper presents the two level models. The first level is a hybrid software engineering process between sequential, iterative and agile software development process. This model motive is to create a green and sustainable software process. In this first model the researcher describe that how we can make each stage of software development life cycle is green. For full fill this motive the researcher use different type of approaches, and finally present a green model of each stage of software development life cycle[1].

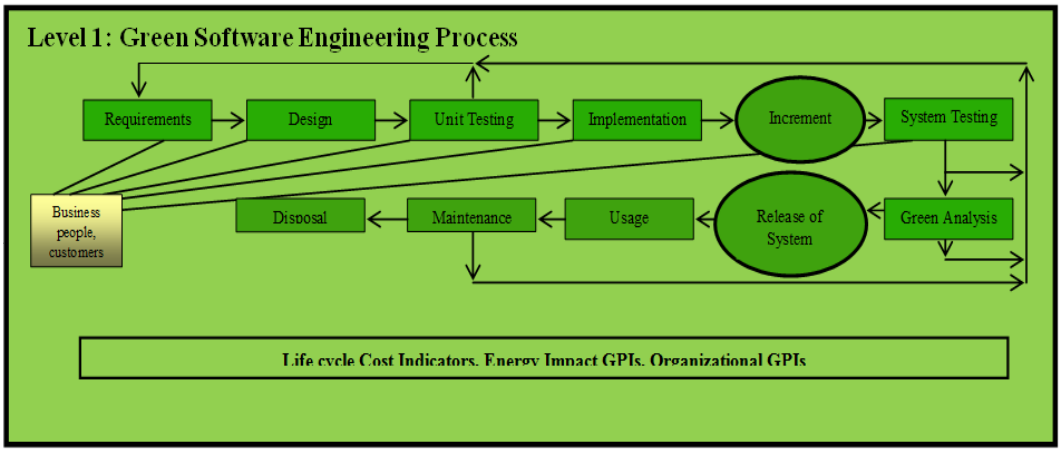


Figure:8 Level 1 of our Proposed Green Model

The second level model defines how software itself can be work as a tool to promote green computing by specifying all the approaches that have been taken. In this model the researcher define different types of tools and matrices that help limit the energy wastage of running application.

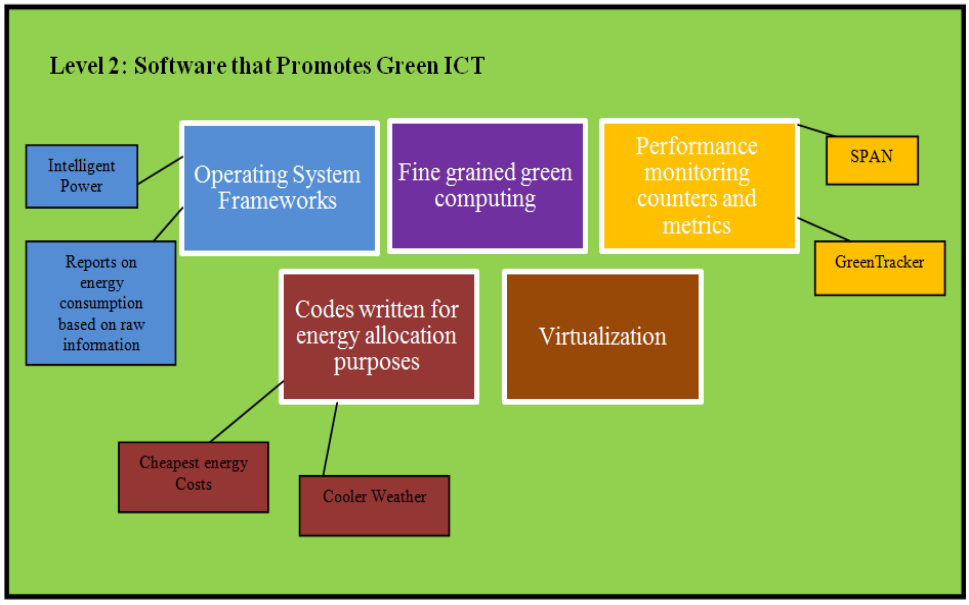


Figure 9: Level 2 of our Proposed Green Model

The other paper is “An approach towards sustainable software development”, This is presented by Senath and Raghavendra is describe the changes in existing software development life cycle and gives the suggestions, which can lead to lower carbon emission , paper and power use thus helping organizations to move towards greener and sustainable software development.

Firstly this paper describe that power consumption of different type of monitors such that *CRT, LCD, LED* .After analysis of all data, the paper give the result that LCD monitors consume less power compared to *CRT*. After that this paper gives a Green software development model. That model tries to refine the different phase of software development by introducing a set of suggestions. These suggestions are following by each stage of software development life cycle Requirement analysis. Design, implementation, coding, testing and maintain ace. [4]

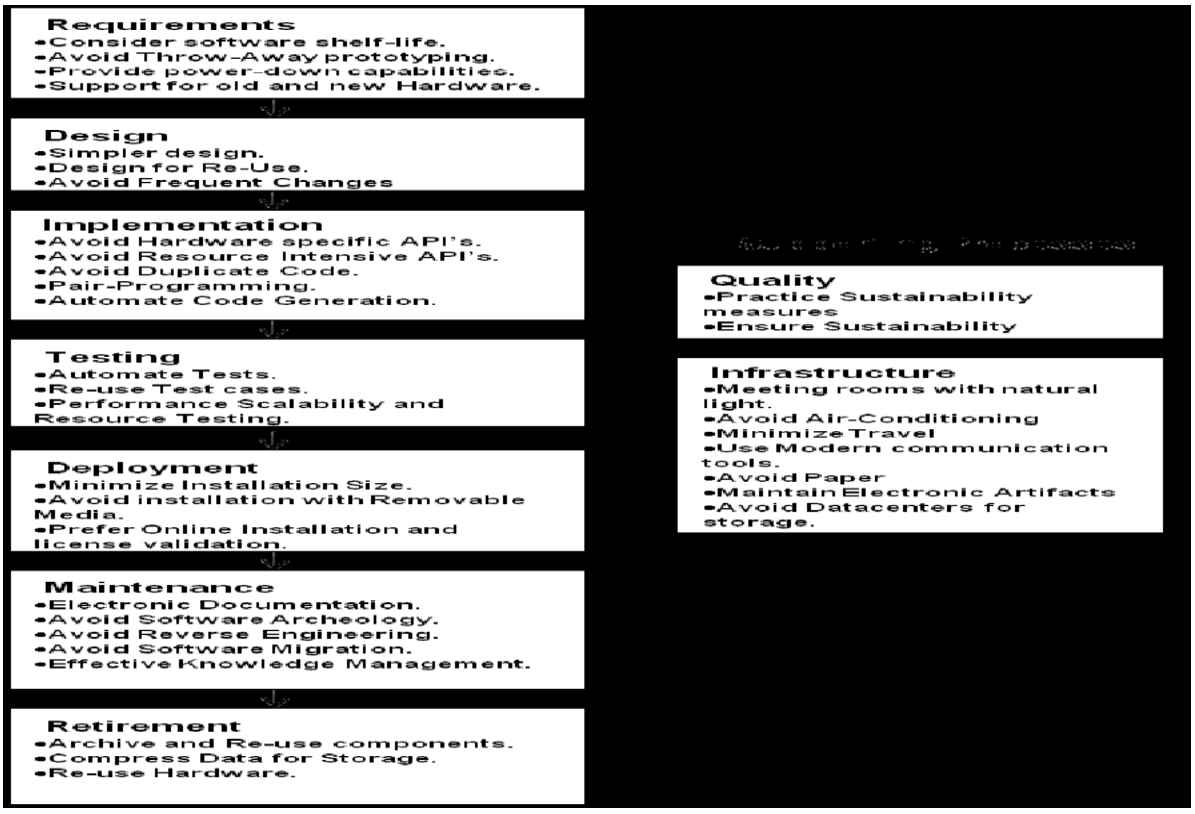


Figure 10 Proposed Sdlc Model

These all suggestions are helpful to avoiding the huge amount of carbon emission, paper and power use and helping the company and organization to moves towards green.

The next approach is this field is "The green soft model: A reference model for green and sustainable software "This is presented by Stefan Neumann and et al. This paper first gives definition of the term "Green and sustainable software" and "Green and sustainable software engineering" after that this paper give conceptual reference model, that is "Green soft model". This model include a

cradle-to-grave product life cycle model for software products, sustainability matrices and criteria for software, software engineering extensions for sustainable sound software design and development as well as appropriate guidance.

This reference model contains a life cycle of software product [12]. The second part of the green soft model is called sustainability criteria and matrices. It covers common metric and criteria for the measurement of software quality [13]. The third part is component procedure model. This part classifies procedure models that covers acquisition and development of software, maintain ace of IT system and user support [9]. The last component of the model contains recommendations and tools. These all tools help to achieve the sustainability in the software [10,14].

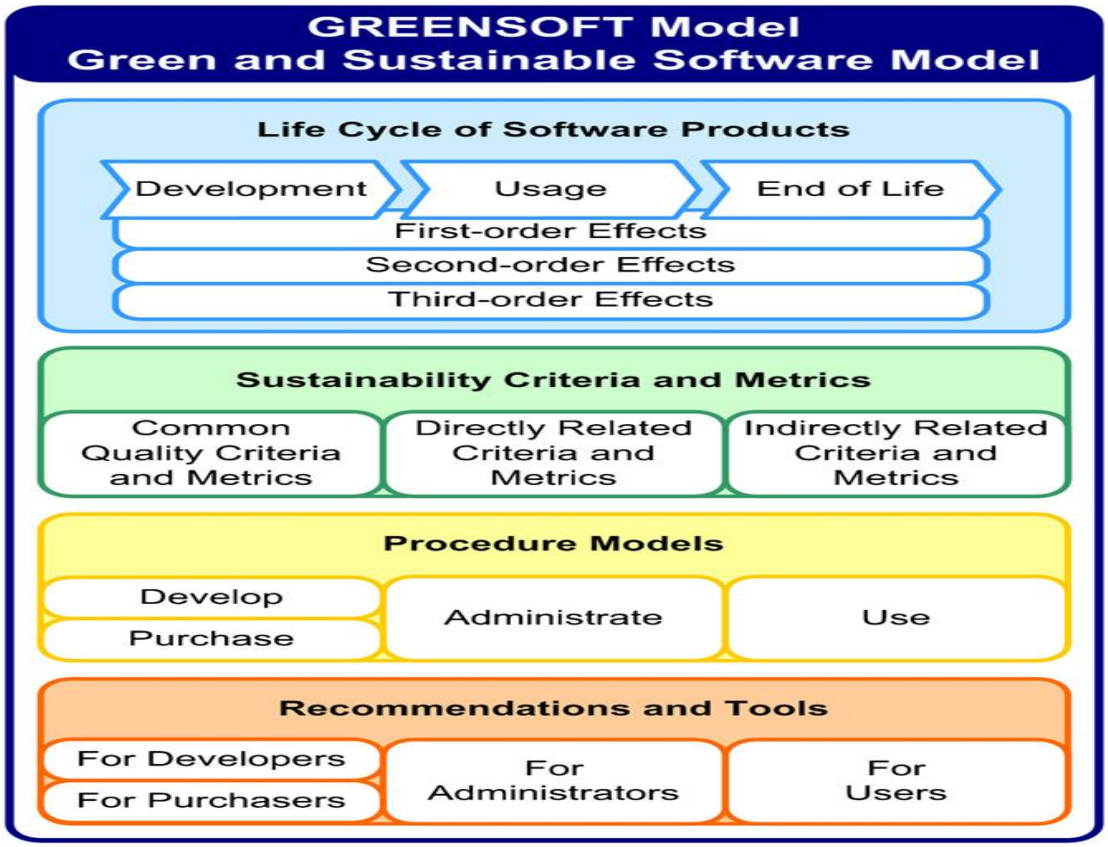


Figure11 The GREENSOFT Model: a reference model for "Green and Sustainable Software".

The next paper is "software level green computing for large scale system". This is presented by Faiza faktar et al. This paper describe that how we can conserve the energy trough hardware and software. The papers describe that, through the

circuit design and implementation we can get the energy conservation in the hardware.

The main focus of this paper is identification of energy conservation for software and there utilization at complier & scheduler. For scheduling its uses the distributed interactive engineering tool box (DIET). And for compilation distributed green complier is used. This is hardware independent and it is use existing distributed complier. Basically main focus of this paper is that how we can reduce energy consumption through the complier and scheduler. In green scheduling, we analysed that the active processes for energy requirements. In green complier we analysis the programme at compile time and perform code reshaping during transformation [16 17].

In this paper there are different type of strategies describe which is helpful for make the compiler green. These are:

a. Cache skipping
b. Use of register operands
c. Instruction clustering
d. Instruction re-ordering and memory addressing
e. Dynamic power management

In this paper researcher proposed the Generic algorithm for distributed green complier (DGC).After that researcher proposed the work floe diagram of distributed green complier.

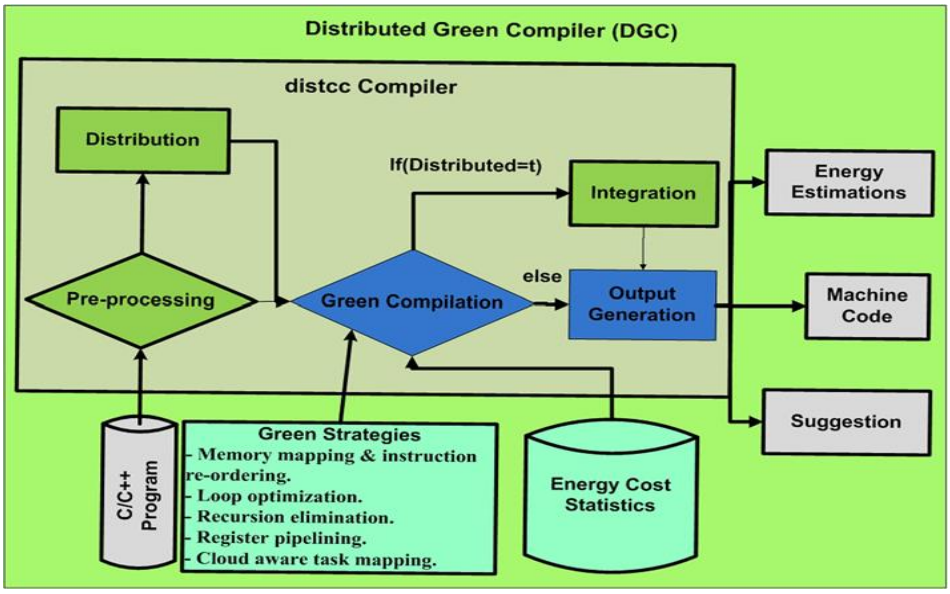


Figure 12 Workflow diagram for proposed Distributed green compiler (DGC).

To make the scheduler green this paper use greedy based algorithm [18]. The greedy based algorithm is used to efficient *VM* allocation to processers. The work of the *VMS* is proper utilization of core processing and reduces the power consumption [15].

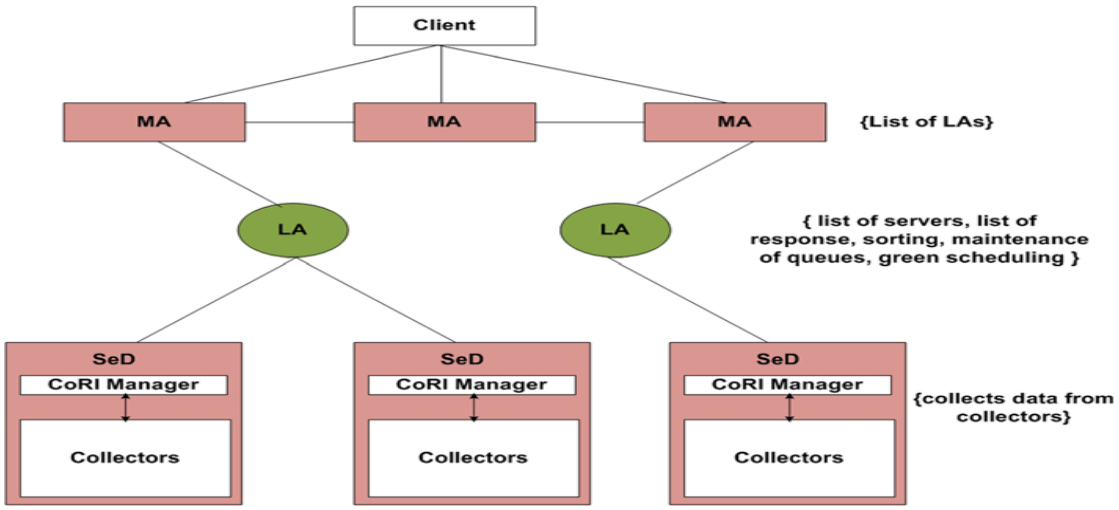


Figure 13 Proposed DIET Scheduler with green aspects.

In the result analysis show that distributed green complier (DGC) conserve clock cycle 30% to 405 by applying few green strategies and distributed interactive engineering tool box (DIET) help to reduce few amount of energy consumption, carbon emission.

The next paper is related to cloud computing. In this paper green computing is use in cloud computing. The paper is “power management in cloud using green

algorithm". This is presented by R.yamini. This paper describes that how we can do the proper power management in cloud using green algorithm. Cloud computing is a delivery of service not a product. The cloud computing provides three types of services:

A Infrastructure as a service

B. Plate form as a service

C. Software as a service

To provide all these services the cloud computing required more power. So this paper describes the solution of power management by using green computing. This paper uses a green algorithm for power management. Firstly this paper proposed architecture of a green cloud computing platform. This architecture supporting energy efficient services allocation in green cloud computing infrastructure. This architecture basically consists of following parts:

a. Consumer/Broker
b. Green resource allocator
c. Virtual machine
d. Physical machine

After that this paper gives an energy model. The motive of this model is:

1. The determination of optimal points from profiling data.
2. Energy aware resource allocation using Euclidean distance between the current allocation and the optimal point at each server [3].

After that for the proper utilization of processer this paper gives two task consolidation algorithms:

a. ECTC (energy consolidation and task consolidation)
b. Max utile (maximum utilization)

In result the task consolidation algorithm play an important role. Because by using this algorithm we can do the proper utilization of the processers by proper resource utilization. And we know that the resource utilization directly related to power consumption. So by the help this model we can save few amount of energy [3].

Now the next paper is" Power efficiency for software algorithm running a graphics processors". This is presented by Bjorn Johnson and et al. This paper describes the power efficiency of a range of graphics algorithm in different GPU's. For measuring power consumption they built a power measuring device that samples currents at a high frequency.

In this the researcher built a power management station, which measure power of the PCI express bus (which can deliver up to 75w) and on the graphics card two power connectors , which in the case can deliver up to 75+150w. This sum is to max of 300w [5].

The main motive of researcher in this paper is to finding the answer of some question:

1. What are power characteristics of different graphics algorithm solving the same problem on different graphics architectures?
2. Is energy directly proposal to frame time.
3. What does the power consumption look like inside a frame?
4. What does the power consumption look like during an animation?
5. Can power optimization of software algorithm, become a new subtopic in graphics.

   To solving these all answer the researchers use the different type of rendering for example forward rendering, backward rendering etc. After that they use the different types of shadow algorithm namely-shadow mapping(SM), shadow volumes (SV), variance shadow mapping etc.

The next paper is "Sustainable development sustainable software and sustainable software engineering". This is presented by Timo johnnan and et al. This paper gives an overview in the field of sustainable in computer science. This paper presented a life cycle model, helps to develop green products.

To obtain the green and sustainable products, this paper use some techniques know from conversional design, software engineering software development. This model consists of:

1. Life cycle of software products
2. Procedure models
3. Recommendations for actions and tools

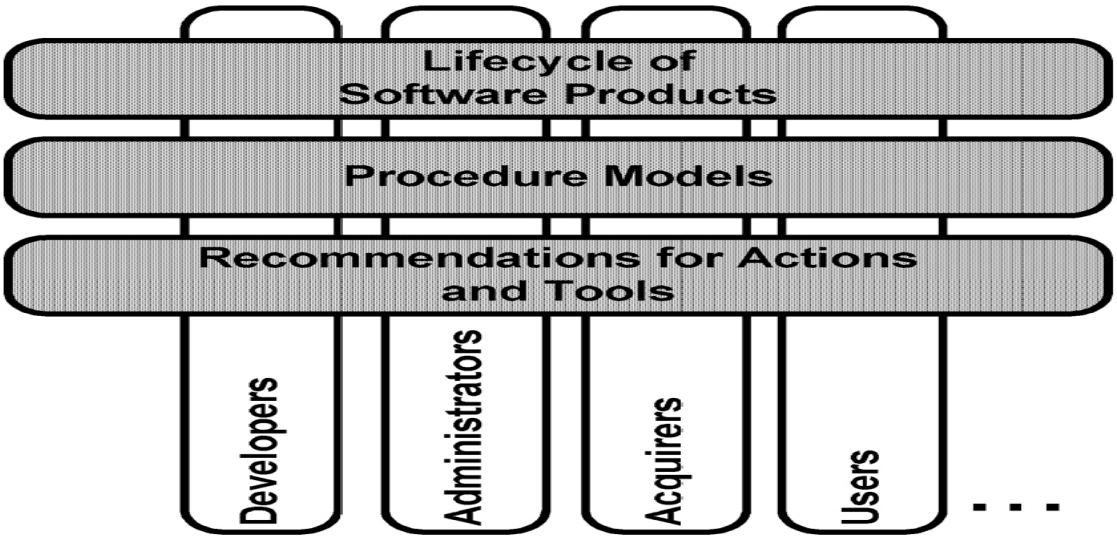


Figure 14 Proposed model

Basically this paper use cradle to grave approach and develop a life cycle of software product in general and of green software products in particular. In next step this paper uses the generic approach in order to integrate a sustainable aspect in to standardized model of software engineering. After that this paper defines some metrics for models, tools and software product to make them measurable and comparable regarding sustainable aspects.

In the result this paper presented a frame work, which specifies that how we can get green products. For this purpose this paper uses the green matrices, tools.

This is literature review in green computing. After carefully study of all previous work we can say that different paper proposed different types of approaches in green computing. But at last the motive of all the paper is to reduce energy consumption, carbon emission, and resource utilization.

After study of literature review we can say that green computing is an interesting area for research and this is very useful for future happy world.

Chapter 3

Proposed Framework

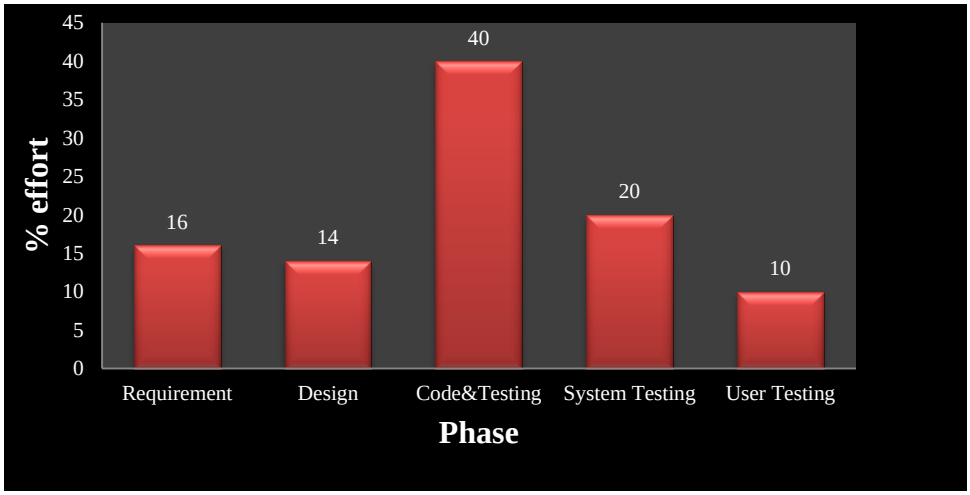
We think that computers are less polluting and consume a little energy. But at this time we are wrong, because according to a survey $ 250 billion per year used up on powering computers worldwide only about 50% of that power is spent computing – the rest is wasted idling. In this paper we are proposing our own software development life cycle modal which is composed by six stages. These stages are Green requirement analysis, Green design, Green implementation, Green testing, Green maintenance and green analysis. The last stage green analysis is very important because all the life cycle stages are similarly connected to the green analysis phase. The reason behind it When one stage is completed, then green analysis phase check that, this stage follow the green computing rules or not. In green analysis phase we will use the different important concept like green scheduler and compilers, fault tolerance, cloud computing, loop optimization, grid computing, software tools and matrices. Our green modal follows the iterative approach instead of the sequential approach, because if we want to change in any phase then we can do easily. But this is the disadvantage of the linear sequential life cycle modal. So we will refer to iterative life cycle model.

As population has increased, energy use has also increased. According to the report from US Environment Protection Agency, data centre in US consumed about 60 billion kWh in 2006, which was near about 1.5 percent of the total US electricity consumption. In green computing we are analysis that how much energy is consume by a device or process and what ways we can use to reduce energy consumption. We know that software and hardware both are required energy consumption although software does not consume energy directly it takes energy indirectly through hardware utilization. So it is important that we have energy factor in our mind when we are development the software. In our software model approach we are including the green analysis in each software development lifecycle phase, by this approach we can know that how we can make each phase green and how we can reduce energy consumption of this future software. In our software development model we are introduce six life cycle phases these phases are green analysis phase, green design phase, green coding phase, green testing phase, green maintenance phase and all these phases are similarly connected to green analysis phase. These phases are connected to green analysis phase because green analysis phase include some basic techniques and applications, by using these

techniques we can get energy efficient software. In first section we will study that how we can implement greenness in each software development life cycle phase. After that in section second we will study which methods will be implemented in green analysis phase and how can use these methods, applications to reduce energy consumption. Software cost estimation methods first try to know the size of the software to be built. Based upon this size the expected effort to be put is measured. Estimated effort further is utilized to calculate the time duration and cost of the project.

Table 2: Energy Consumption by Each Phase

Sr. No.	Activity	Standard work effort %
1.	Requirements Phase	16
2.	Design Phase	14
3.	Code and Unit testing	40
4.	System Testing	20
5.	User acceptance Testing	10



Energy consumption of each phase

3.1 Steps in green software lifecycle model

This is my proposed green frame work. This model consists of six stages. All these stages are software development life cycle. The main motive is us how we can make this software development cycle green. For this *I* will use some concept of green computing.

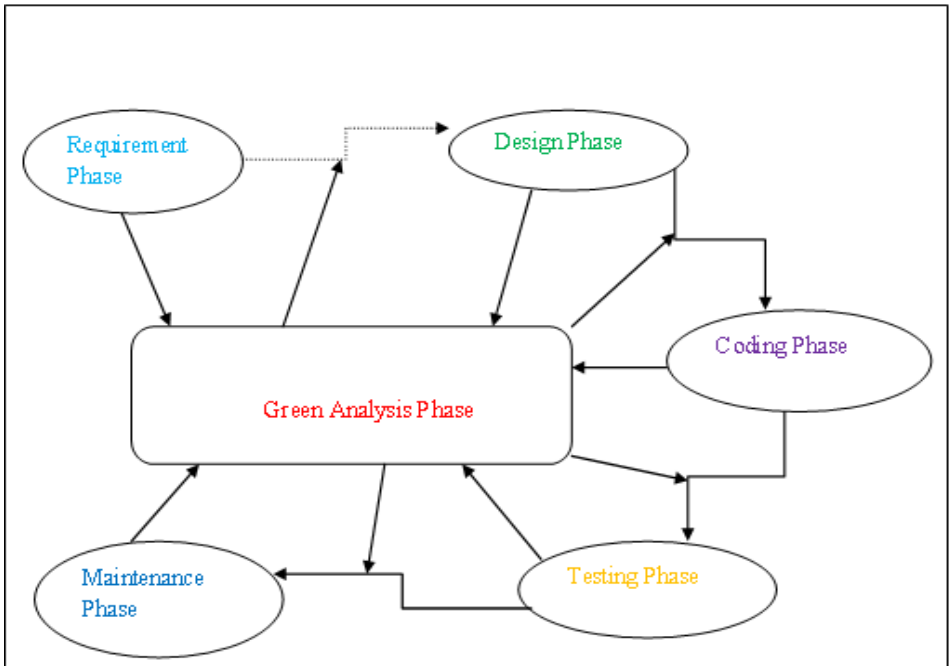


Figure 15: Proposed Green model

3.1 Green requirement phase:

We can say that this is a planning phase. Because in this phase software engineers team gathering the requirements from the customer. After collecting requirements of customer, requirement analyses are usually done by good skilled and experienced software engineer in the organization. This is a basic process of software engineering. In our purposed requirement modal we will add a new step. In this step we will study that how we will make requirement phase energy efficient. For this purpose we use software tools and matrices. Software matrices are an evaluation criteria indicating how green and energy efficient a system or software. They measures that how much amount energy wasted or use efficiently. Software tools can be used to keep track, identify and limit the amount of energy used by running application [20]. Cloud computing is a delivery of computing as a service not as a product. There are many open source hardware and software level resource available as a services by different vendors for example currency converter, calculator making use of these readymade resource in program will be beneficial in term of cost and time compare to reprogramming them.

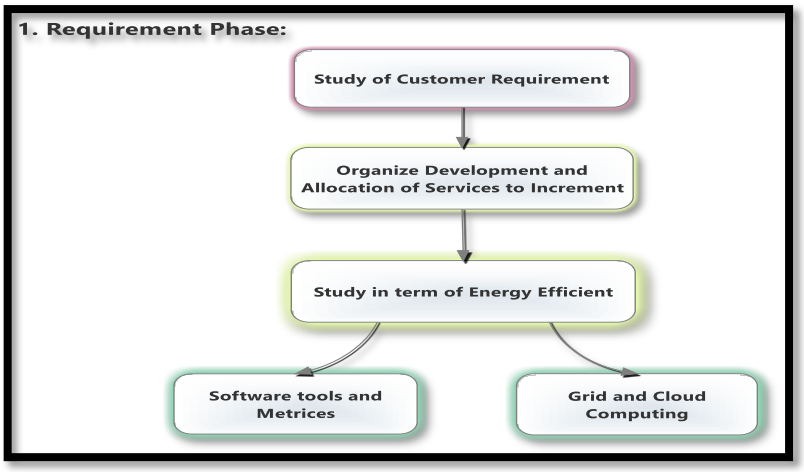


Figure 16: Green requirement phase

3.2 Green architectural design phase

Green design is co-existing with respect to product and process designs, hence green design tool must be able provide an optimal green design that has minimal environmental impact with respect to both product and process. For this purpose the green design tool uses and builds upon the concept of subassembly. We can define subassembly it is a unit assembled separately but design to fit with other unit in a manufactured product. This can't be disassembled without permanent damage. By the help of subassembly we can save time and money in lot of ways. Greenness attributes describe the greenness of a product design with eleven top level attributes namely subassembly, reusability, label, internal joints, component Varity, component identification, recycled content, chemical uses, additives, surface finishes, external joints and hazard level of material [7].

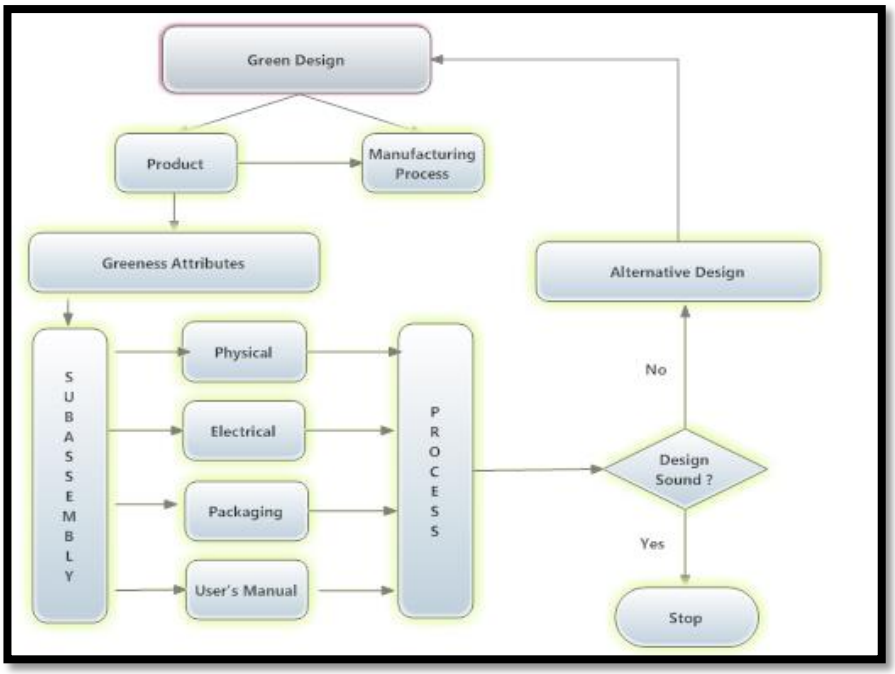


Figure 17: Green Design Phase

3.3 Green coding phase

In this phase a software developer return the code according to client requirements. We can make the coding phase energy efficient by following way;

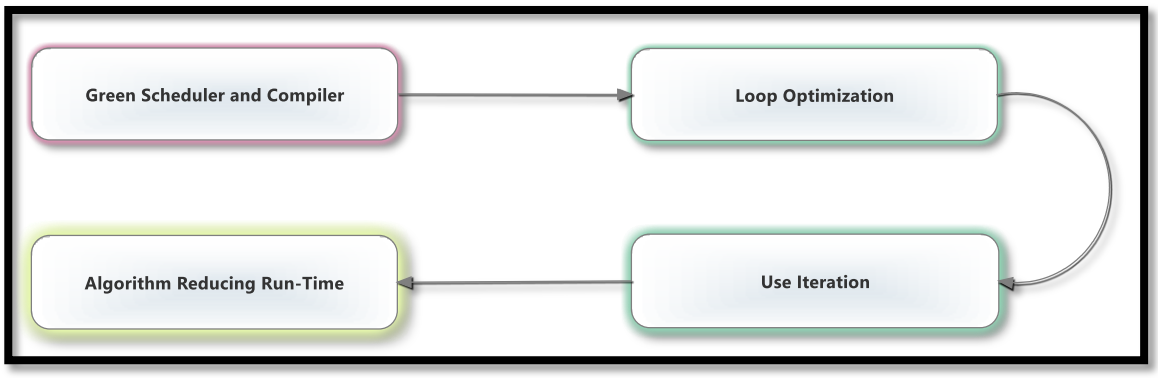


Figure 18: Green Coding Phase

- **Green compiler & scheduler**:

  A distributed green compiler (DGC) is a hardware independent compiler & it uses an existing distributed compiler. It distribute source code of software our a network, reshape binary code by applying green strategies during code transformation at compile time & give green suggestion to software programmer for energy reduction. For scheduling distributed interactive engineering tool box (DIET) scheduler is used. For DIET scheduler a new

algorithm is used. This used algorithm introduced green aspect in scheduler to effectually make use of resource in such a way that consumption of power and carbon dioxide (CO2) emission is reduced

- **Loop Optimization:**

We know that loop play an important role in programming. In loops, replication improves the performance but it required high energy consumption. A best approach can be skipping of cache operation during unnecessary replication. In this instruction clustering is a way that allows a compiler to execute pair or cluster of instruction in one cycle. It will reduce the running time of program and reduce energy consumption.

- **Loop Transformation:**

The term loop optimization include modify loop body or the control structure of the loop. They are usually variable since operate on a subset of code that is typically executed frequently. The transformation is use for source level energy reduction. In this the basic idea is to reduce the size of the loop body in order to decrease transformation produces positive effect in term of energy I-cache and D-cache misses.[22]

- **Algorithms which reduce energy:**

After that we can use energy cost data base. The energy cost data base can be use in code parsing and parse tree generation algorithms.

4.4 Testing phase

The software testing phase is used to check that the software follow all the requirement of the customer or not. Or can emphasize on discovering faults or defects in the software where the behaviour of the software is incorrect. In this phase we can use the different type of testing techniques for requirement testing, unit testing, integration testing, and system testing. In given fig. is a popular testing process use in software engineering. In our proposed modal we can add High speed testing techniques .in this we will introduce some techniques that can speed the testing process and provide faster feedback to the team and customer.

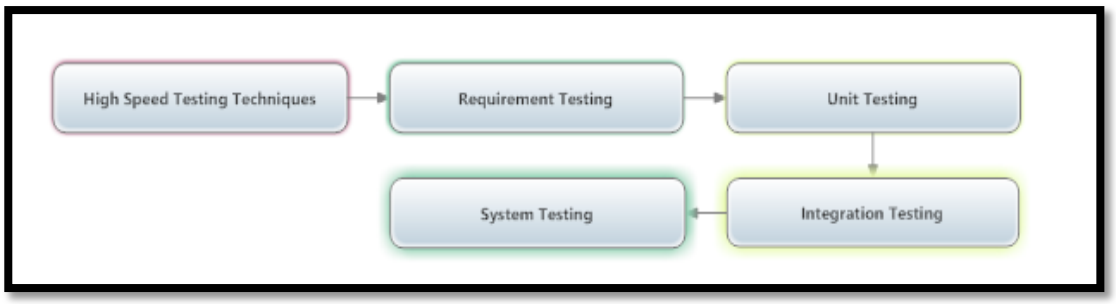


Figure 19: Green Testing Phase

- **Risk based testing:** This is a type of software testing. In this software team specify the features and function on which software may be fail. There are different types of risk based testing like business or operational, technical, external testing.

- **Pair wise testing*:*** Pair wise testing is an economical alternative to testing all possible combinations of a set of variables. In pair wise testing a set of test cases is generated that covers all combinations of the selected test data values for each pair of variables. Pair wise testing is also referred to as *all pairs* testing and *2-way* testing. It is also possible to do all triples (3-way) or all quadruples (4-way) testing, of course, but the size of the higher order test sets grows very rapidly.

4.5 Maintenance phase

Software maintenance is a phase which involved in changing a system after it has been delivered. The maintenance phase is an expensive phase, in this phase cost is proportional to energy .In maintenance phase it is important that when developer, develop this software, he will written code easy and more under stable because if

program is easily understood than it will much faster and easier to apply changes.

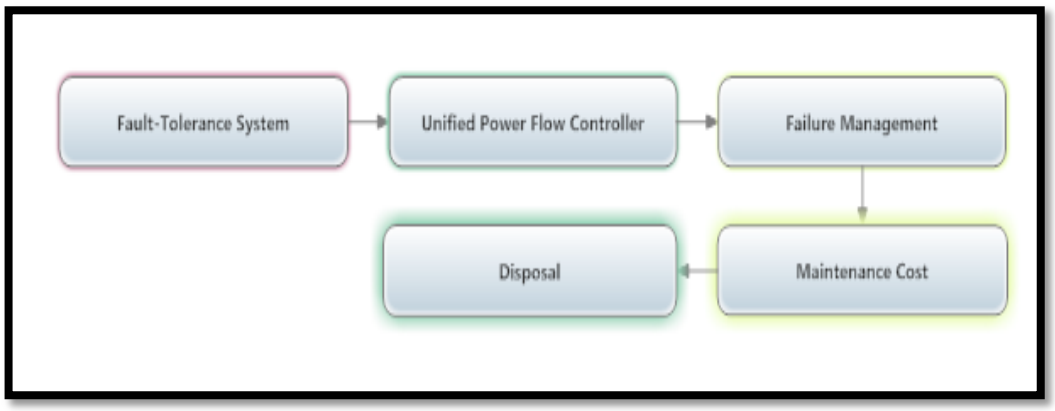


Figure 20: Green Maintenance Phase

We can make the maintenance phase green by including following matrices:

- **Fault Tolerance*:***

This provides the environmental sustainable benefit to minimize environmental waste through allowing the system to function even with defect instead of shutting down and requiring repair. For fault recovery we can use the unified power flow controller (UPFC) [23].

- **Failure Management*:***

A good failure management can carry out a successful failure process and as a result reduce environmental waste by preventing future failure.

- **Disposal Stage:**

- Software recycling in term of reusing the software code for future projects.
- Hardware recycling in term of following reusing and recycling before disposing the equipment and material.
- Products that can be used repeatedly should be bought to save natural resources.

This is my frame work. This frame work uses the different types of approaches and tools. The motive of this frame work is to reduce the energy consumption, co2 emission and reduce resource utilization.

Chapter 4 Simulator and results

Simulator

4.1. Follow the steps to install Cygwin

---

1. Go to http://www.cygwin.com/
2. Click the Install or Update Icon
3. Choose RUN from the window.
4. Choose RUN again from the window (If it pops up)
5. Choose NEXT from window
6. Choose Install from Internet option and click NEXT button from window.
7. Enter the PATH for Cygwin software and keep default selections for all other options. And click NEX to continue
8. Select a folder where you want to save a copy Cygwin installation files in the window.
9. Choose your connection type.
10. Select the nearest downloading website from the window.
11. By default Cygwin installation will not contain some of the additional utilities that we need in order to run GCC for Simple scalar. So to install these extra utilities, make sure that you have included Devel window (Just click once to change it from default to install) and click NEXT.
12. Click Finish.

After successful installation of Cygwin Open Cygwin and type "make" command as shown in the following window.
NOTE: If you get "make: *** No targets specified and no makefile found. Stop." that means all the additional utilities also installed successfully. If you get message such as "bash: make: command not found" that means either you haven't selected "Devel" tools while installing Cygwin or errors occurred while installing the software. So you may need to re-install Cygwin again including "Devel".
Cygwin installation is complete.

4.2 Follow the steps to install Simple scalar on Cygwin

- ----------

For Cygwin on Windows/PC platform

1. Get the Simple Scalar package from http://www.simplescalar.com

2. Go to Tools in the Downloads section to the left and download simplesim-3v0d.tgz.

3. Download the package directly into Cygwin's root directory (c:\cygwin).

4. Open the Cygwin and make sure that you are at Cygwin root directory as shown below. Just type cd / and press enter to make sure you are at Cygwin root directory.

5. Type ls and press enter to see if the downloaded file is there.

6. Type "tar –xzvf simplesim-3v0d.tgz" and press enter as shown below to untar and unzip the package to install Simple scalar.

This will create a subdirectory simplesim-3.0 with the source code for all simulators described above.

7. cd simplesim-3.0 press enter

8. And then type "make config-pisa" at the prompt and press enter as shown below.

This step will set up the files for building the PISA target. The other alternative is an Alpha target.

9. Type "make" and press enter as shown below.

At end of the results, it should display something like "**my work is done here**" as shown below.

10. Type makesim-tests and press enter as shown below.

This step is to verify that the simulators built OK

The Simplescalar installation is complete.

4.3 Follow the steps to install GCC cross compiler for Simple scalar

----

In order to be able to compile programs to run on the simulator, you need a port of cross-compiler and libraries for Cygwin.

Download the cross compiler tar file from http://www.eecg.toronto.edu/~moshovos/ACA05/hw/ss-gcc.usrlocal.tar.bz directly into Cygwin's root directory.

Open the Cygwin and type cd / and press enter to make sure you are at Cygwin root directory.

Type “tar –xvf ss-gcc.usrlocal.tar.tar” and press enter as shown below to untar the package to install cross compiler. (Note: this is just a tar file not a zip file. So the option should be only –xvf not -xzvf)

Congratulations!. Now you have installed all of the components to run and compile files for Simplescalar.

To test the cross compiler, lets do the following:

To compile c programs

sslittle-na-sstrix-gcc<your c program>

To run simplescalar

cd /simplesim-3.0 (to go to simplesim-3.0 directory)

./sim-safe <compiled c program for example a.out>

To store the simplescalar out to a file

./sim-safe –redir:sim /test/testfilea.out

-redir:sim -> is to reditrect the output to a file
/test/testfile -> path and output file name
a.out -> compiled c program

Type cd/simplesim-3.0

Hello World test:

```
#include<stdio.h>

main()
{
printf("Hello World");

}
```

Save as .c file in simplesim-3.0

To check if the file is there, type ls in cgywin.

sslittle-na-sstrix-gcchelloworld.c

run program: ./sim-safe a.out (will run last compiled program)

4.4 Our Simulation Results:

IN our proposed model currently we are working only Coding phase. So our motive is to proof all the parameters or concepts those we are using in our proposed framework. So we are showing each parameter one by one. The first parameter which we are using is green Complier and green Scheduler. For this we are given the algorithm.[23]

Algorithm for proposed Distributed green compiler (DGC)

```
Input:
 1. A C program.
 2. Switches D for distribution
 3. Machine IPs for distribution.
Output:
    1. An energy conservative executable.
    2. Green suggestion.
    3. Energy statistics report of program.
Begin
    while(End of file){
          pre-process source code.}
    If(D){
          Whilse(End of file){
              Distribute on network}
    }
    while(End of file){
        Loop optimization, Dead code and recursion elimination, Un-optimized block
        identification, Energy statistics calculation
        Other Compilation process for intermediate code generation}
    If(D){While(End of file){
                              Integrate}
                  }
 Generate output
End
```

Features analysis of DGC with existing compilers

Features/Compilers	DGC	Encc	Coffee	Mrcc
Distributed	☼	×	×	☼
Hardware Independency	☼	×	×	☼
Cloud Aware Task aping	☼	×	×	☼
Energy Cost Calculations	☼	☼	☼	×
Loop Optimization	☼	×	☼	×
Dynamic Power Management	×	×	×	×
Instruction Reordering	×	☼	☼	×
Recursion Elimination	☼	×	×	×
Register pipelining	☼	☼	☼	×

In Scheduler, when we are using Greedy Capacity Algorithm then it not only reduce the energy consumption but also efficiently utilize the cloud resources. So this is called green Scheduler.

```
Begin
    while (true)
    {
      for( i = 1 to i <= queue1.length( ))
      {
        If  ci >= 1 && queue1.length( )>0)
        {
            If check capacity vm on ci
            {
                Schedule vm on ci AND task on VM
                ci – 1
             }
        }
      }
    }
End
```

This is greedy capacity algorithm. The list of VMs on machine is maintained in queue with LA and tasks in queue1.The major aim of this algorithm is to utilize all processing cores in a machine, which reduces the power consumption.

The next concept which we are using in our framework is code transformation.For example when we are using the code transformation concept in coding phase, then According to Carlo Brandolese et al. [24] we get result

```
main () {
int i, j, a, b[8], result = 0;
for (i = 0; i < 8; i++)
scanf(f, "%d", &b[i]);
scanf(f, "%d", &a);
result += foo(a, b);
printf("result = %d", result);
}
int foo(int a, int b[8]) {
int i, sum = 0;
for (i = 0; i < 8; i++)
sum += a * b[i];
return sum;
}
```

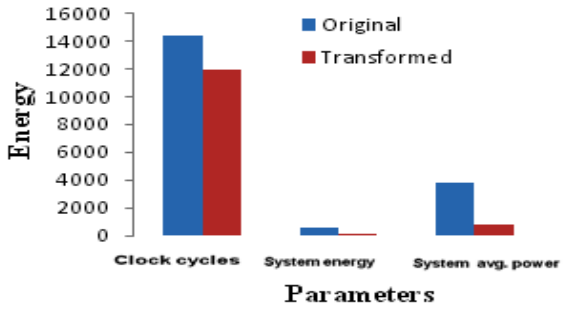
⇒

```
main () {
int i, j, a, b[8], result = 0;
for (i = 0; i < 8; i++)
scanf(f, "%d", &b[i]);
scanf(f, "%d", &a);
if (cvd foo(a)) result += sp foo(b);
else result += foo(a, b);
printf("result = %d", result);
}
int foo(int a, int b[8]) {
int i, sum = 0;
for (i = 0; i < 8; i++)
sum += a * b[i];
return sum;
}
int sp foo(int b[8]) {
return 0;
}
int cvd foo(int a) {
if (a == 0) return 1;
return0 ;

}
```

Table 3: Result Analysis

Parameter	Original	Transformed	%
Clock Cycles	14393.00	11981.00	-16.76
Processor Energy (¹J)	3.44	3.25	-5.64
Processor avg. Power (mW)	23.90	27.09 1	3.35
System Energy (¹J)	544.00	90.00	-83.45
System Avg Power (mW)	3778.66	751.31	-80.12



This is the result when we are using the loop transformation concept, by the help of this table we can say that if time period is reduce then energy is automatically reduce. For simulate all these results which is given in table we have to use simple scalar tool. By the help of this tool we will calculate the cache miss rate, cache hit rate, read penalty, and write penalty etc. parameters.

These all parameters are very helpful for calculating clock cycles, Processor Energy, Processor avg. Power (mW). By the help of sheiu-chakarbati model we will calculating all these parameters. For this we will use different type of formula's:

For calculating clock cycles We adopt the model used in [Hennessey and Patterson 1996] and assume that the hit_time or the number of cycles per hit is 1, and that the miss_penalty or the number of cycles per miss is 42, 44, 44 and 48 for line sizes of 16, 32, 64 and 128 respectively. Thus increasing the line size reduces the miss_rate but increases the miss_penalty. The execution time (in cycles) is another important performance metric. If the execution time of a task assuming no cache misses is known, then the execution time (in cycles) is given by [Li and Henkel 98].

Cycles = Cycles_w/o_cache + N_miss_rd * read_penalty + N_miss_wr * write_penalty + N_miss_fetch * fetch_penalty[25]

where Cycles_w/o_cache is the execution time of a program running on the processor core without cache misses, N_miss_rd is the number of read misses, N_miss_wr is the number of write misses, N_miss_fetch is the number of instruction fetches, read_penalty is the number of cycles for a read miss, write_penalty is the number of cycles for a write miss and fetch_penalty is the number of cycles for an instruction cache miss. For a fixed size instruction cache, the term N_miss_fetch * fetch_penalty does not change with the data cache configuration and can be considered as a constant.

If we assume that for the data cache, the read and write penalties are the same,

then the number of cycles can be approximated to

Cycles = Cycles_w/o_cache + N_misses * miss_penalty + constant,

Similarly for calculating processor energy we will use different type of formulas.[26]

$$MISS\_R = \frac{\sum_j mr(j) * trip(j)}{\sum_j trip(j)}$$

$$CYCLES = \sum_j C(j) * trip(j)$$

$$ENERGY = \sum_j E(j) * trip(j)$$

So in given proposed model we are using difference type of such type of concept. In future we will try to give the framework for each phase of software development cycle. This will be helpful for us to minimize energy reduction and e-wastage in Software development life cycle.

The next concept which we are using is Loop Optimization.

Loop Optimization: Loop optimization is an important concept in programming. Because this explain that how we can decrees execution time. This improves the performance of object programmes by transforming existing code in to more efficient code.

```
void left (a, distance)
char a[];
int distance;
{
int j, length;
length = strlen(a) - distance;
for (j = 0; j < length; j++)
a[j] = a[j + distance];
}
```

When we will compile given source code. Then we will get optimized code and unoptimized code which is produce by compiler. The optimized version (compiled with the –O option) contains fewer total instructions and fewer instructions that reference memory.

In this unoptimized code the loop is 13 instructions long and uses eight memory references. But in optimized loop the loop is six instructions long and uses two memory references.

Table 4 Result analysis of loop optimization

Programme	instructions	Memory reference
Optimized Code	6	6
Unoptimized Code	13	8

By the help of given result we can say that in optimization code instruction size is less compare to unoptimized code so optimized code take less time compare to unoptimized code in programming. As we have discussed that if time is less then consumption of energy is also less. So we can say that loop optimization is play important role to energy reduction.

References:

1. Sara S. Mahmoud and Imtiaz Ahmad "A Green Model for Sustainable Software Engineering" International Journal of Software Engineering and Its Applications 4, July, 2013 pp.55-74.

2. Faiza Fakhar1*, Barkha Javed1, Raihanur Rasool1, Owais Malik1 and Khurram Zulfiqar2 "Software level green computing for large scale systems" Journal of Cloud Computing: Advances, Systems and Applications 2012.

3. R.Yamini **"**Power Management in Cloud Computing Using Green Algorithm"IEEE-International Conference On Advances In Engineering, Science And Management March 30, 31, 2012.

4. Sanath.S.Shenoy, Raghavendra Eeratta "Green SoftwareDevelopmentModel:An Approach towards Sustainable Software development" IEEE-International Conference On Advances In Engineering, Science And Management (ICAESM -2012) pp. 128-133.

5. Björn Johnsson, Per Ganestam, Michael Doggett and Tomas Akenine-Möller "Power Efficiency for Software Algorithms running on Graphics Processors" IEEE/ACM International Conference on Green Computing and Communications pp.230-233.

6. Keith S. Vallerio, Lin Zhong, and Niraj K. Jha, "Energy-Efficient Graphical User Interface Design" IEEE/ACM International Conference on Green Computing and Communications pp.226-229.

7. Bahiru Kassahun, Manjini Saminathan, Janine **C.** Sekutowski "Green Design Tool**"** Third International Conference on Intelligent System Design and Engineering Applications pp.432-435.

8. Stefan Naumann∗, Markus Dick, Eva Kern, Timo Johann"The GREENSOFT Model: A reference model for green and sustainable softwareanditsengineering" 2011 Elsevier,pp205-211.

9. M. Dick, S. Naumann, Enhancing software engineering processes towards sustainable software product design, in: K. Greve A.B. Cremers (Eds.),

EnviroInfo 2010, Integration of Environmental Information in Europe, Proceedings of the 24th International Conference on Informatics for Environmental Protection, October 6–8, 2010, Cologne/Bonn, Germany, Shaker, Aachen, 2010, pp.706–715.

10. M. Dick, S. Naumann, A. Held, Green web engineering, a set of principles to support the development and operation of "Green" websites and their utilization during a website's life cycle, in: J. Filipe, J. Cordeiro (Eds.), WEBIST 2010– Proceedings of the Sixth International Conference on Web Information Systemsand Technologies, vol. 1, Valencia, Spain, April 07–10, 2010, pp. 48–55.

11. M. Dick, S. Naumann, N. Kuhn, "A model and selected instances of green and sustainable software", 9th IFIP TC 9 International Conference, HCC9 2010 pp. 248–259.

12. U. Tischner, B. Dietz, S. Maselter, E. Schmincke, M. Prosler, F. Rubik, B. Hirschl,How to do EcoDesign? A Guide for Environmentally and Economically Sound Design, Verlag form, Frankfurt am Main, 2000.

13. International Organization for Standardization, Software Engineering – SoftwareProduct Quality Requirements and Evaluation (SQuaRE) – Guide toSQuaRE, 2005.

14. J. Fischer, S. Naumann, M. Dick "Enhancing sustainability of the software lifecycle via a generic knowledge base" 24th International Conference on Informatics for Environmental Protection,October 6–8, 2010, pp.716–725.

15. Ishfaq Ahmad (2012) Handbook on Energy-Aware and Green Computing., University of Texas at Arlington, USA;Sanjay Ranka, University of Florida, Gainesville, USA.

16. Heath T, Pinheiro E, Hom J, Kremer U, Bianchini R (2002) Application Transformations for Energy and Performance-Aware Device Management. International conference of Parallel Architectures and Compilation Techniques, pp121–130.

17. 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.

18. Duy TVT, Sato Y, Inoguchi Y (2011) “A prediction-based green scheduler for datacentres in clouds”. IEICE Trans InfSyst E94-D(9) pp1731–1741.

19. S. Murugesan, “Harnessing green it: Principles and practices,” IEEE IT Professional, pp. 24–33, January 2008.

20. E. Macii, M. Pedram and F. Somenzi, “High-level power modeling, estimation, and optimization,” *IEEE Trans. On CAD,* (1998) 1061–1079.

21. W. Ye, N. Vijaykrishnan, M. Kandemir and M.J. Irwin, “The design and use of SimplePower: a cycle-accurate energy estimation tool,” *Proc. of IEEE Design Automation Conference*, (2000) 340–345.

22. W.–T Shiue and C. Chakrabarti, “Memory exploration for low power embedded systems,” *Proc of IEEE Symp. On Circuits and Systems,* (1999) 250–253.

23. M.B. Kamble and K. Ghosse, “Analytical energy dissipation models for low power caches,” *Proc of IEEE Symp. Low Power Electronics and Design,* (1997) 143–148.

24. . Brandolese, W. Fornaciari and F. Salice, “Code-level transformations for software power optimization,” *CEFRIEL, Tech. Rep. N. RT-02-004,* (2002).

25. Wen-Tsong Shiue,Sathishkumar Udayanarayanan,Chaitali Chakrabarti “Data Memory Design and Exploration for Low Power Embedded Systems” 37th IEEE/ACM Design Automation Conference.pp-1-18.

26. Wen-Tsong Shiue, Chaitali Chakrabarti “Memory Exploration for Low Power,

Embedded Systems” 37th IEEE/ACM Design Automation Conference pp. 1-6.