Software Methodology and Automation for Semiconductor and IP's Flow

Prepared By :

Rajat Gupta

12MCEC08

Internal Guide

Prof. Vipul Chudasama

Nirma University

External Guide

Mrs. Jyoti Kumar

ST Microelectronics

Department Of Computer Science And Engineering

Institute Of Technology

Nirma University

Ahmedabad

MAY-2014

Software Methodology and Automation for Semiconductor and IP's Flow

Major Project

Submitted in partial fulfillment of the requirements

For the degree of

Master of Technology in Computer Science and Engineering

PREPARED BY :

Rajat Gupta

12MCEC08

Internal Guide

Prof. Vipul Chudasama

Nirma University

External Guide

Mrs. Jyoti Kuamar

STMicroelectronics

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INSTITUTE OF TECHNOLOGY

NIRMA UNIVERSITY

AHMEDABAD

DECLARATION

This is to certify that,

I, **Rajat Gupta**, **12MCEC08**, a student of semester IV Master of Technology in Computer Science Engineering, Nirma University, Ahmedabad , hereby declare that the project work **Software Methodology and Automation for Semiconductor and IP's Flow** has been carried out by me under the guidance of Mrs. Jyoti Kumar , ST Microelectronics India Private Limited,Greater Noida and Prof. Vipul Chudasama, Department of Computer Science and Engineering, Nirma University, Ahmedabad. This Project has been submitted in the partial fulfillment of the requirements for the award of degree Master of Technology (M.Tech.) in Computer Science and Engineering, Nirma University, Ahmedabad during the year 2013 - 2014.

I have not submitted this work in full or part to any other University or Institution for the award of any other degree.

Rajat Gupta(12MCEC08)

CERTIFICATE

This is to certify that the Major Project entitled **Software Methodology and Automation for Semiconductor and IP's Flow** submitted by **Rajat Gupta(12MCEC08)**, towards the partial fulfillment of the requirements for the degree of Master of Technology in Computer Science Engineering of Nirma University of Science and Technology, Ahmedabad is the record of work carried out by him under my supervision and guidance. In my opinion, the submitted work has reached a level required for being accepted for examination. The results embodied in this major project, to the best of my knowledge, have not been submitted to any other university or institution for award of any degree or diploma.

Mrs. Jyoti Kumar
External Guide,
STMicroelectronics India Ltd.

Prof Vipul Chudasama
Internal Guide,
Nirma University

Seema Garg
Mentor,
STMicroelecttronics India Ltd.

Prof. Vijay Ukani
PG Coordinator - CSE,
Nirma University

Dr. Ketan Kotecha
Director,
Nirma University

Dr. Sanjay Garg
HOD - CSE,
Nirma University

ACKNOWLEDGEMENT

First and foremost, sincere thanks to **Mrs. Jyoti Kumar** Manager, STMicreoelectronics India Private Limited, Greater Noida. I enjoyed his vast knowledge and owe him lots of gratitude for having a profound impact on this report.

I would like to thank my Mentor, **Mrs. Seema Garg**, STMicroelectronics India Private Limited, Greater Noida for his valuable guidance. Throughout the training, he has given me much valuable advice on project work. Without him, this project work would never have been completed.

I would also like to thank my Internel guide **Prof. Vipul Chudasama**, Institute of Technology, Nirma University, Ahmedabad for her valuable guidance.

I would also like to thank **Dr. K.K.Kotecha**, Director, Institute of Technology, Nirma University, Ahmedabad for providing me an opportunity to get an internship at Intel Technology India Private Limited, Bangalore.

I would like to thank my all faculty members for providing encouragement, exchanging knowledge during my post-graduate program.

I also owe my colleagues in the ST Microelectronics, special thanks for helping me on this path and for making project at Intel more enjoyable.

Rajat Gupta(12MCEC08)

Contents

List of Figures

Abstract

The project aims to develop a system is for any type of document but nothing particular. This system is revolutionary to document processing. The uniqueness is being dekstop based and for general-purpose.

The proposal is to characterize an efficient methodology to accomplish enhancement and proficiency change by acknowledging general manual generation.test substance streamlining for the Manual era in order to attain betterof Quality final items.

The center is on the Automation of Standard cell document era. Right now these sorts of archives are created physically so they require more manual exertions, time and precision is additionally not guaranteed.so The proposal is to computerize the documentation process . So by utilizing distinctive S/w instruments and calculations, the documentation procedure could be robotized which can decrease the manual deliberations, diminish the process duration and increment accuracy.this robotization incorporates objective appraisal of diverse innovation archives

Chapter 1

User Manual Generator

1.1 Introduction

The Standard Cells Doc Automaton is a comprehensive automation tool to generate standard cells user manual, automatically on a click of button and in no minute time. Indirectly this saves enormous amount of time, what usually needed if any manual created manually. Also all the specification and inputs used for setup this automaton are already reviewed and in agreed on standard (with designer), hence it minimizes frequent human intervention, number of reviews and other steps needed to make the manual quality. This leads to consistent, error free and more quality deliverable. As Other benefit, rather than generating manual in .PDF format only, this tool also delivers .XML (projectteam.xml) file. It makes possible the customization of manual easier and faster in case any new section or update need in current document so indirectly reusability also achieved.[4]

1.2 Abstract View of System

- As according to the principal of software engineering, any software system should take at least one input and must produce some desire output (at least one) in expected amount of time and cost.
- This Doc Automation system also follows the same principal; it takes some inputs at run time and produces .XML and .PDF as output, Input used at run time and Outputs.

- Supporting files are basically the setup required .txt files used to build this automation system, depending upon which entire system coding logic developed.
- These all supporting files are reviewed and provided by designer, at the time when any techno need to be automated; these are fixed and placed in single location, System supporting input files

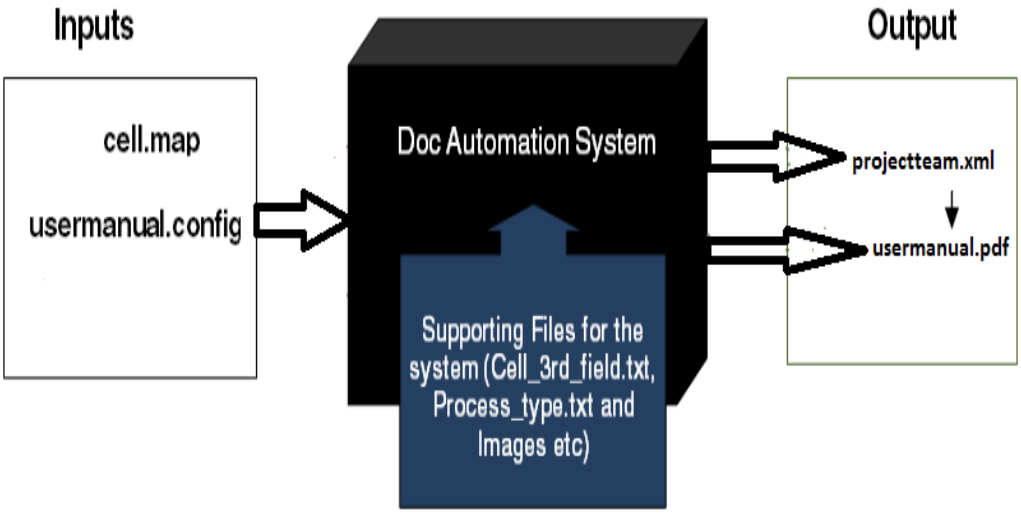

Figure 1.1: Abstract view of Tool

[4]

1.3 Inputs

The tool take two input file cell.map and config.conf . In these two files cell.map is mandatory file while config.conf is optional. If user is not providing the config.conf file then he need to provide library name and techno name as input. There are some supporting files containing various information about the library. The user has a provision to override the supporting file from GUI by selecting supporting file.

- **Cell.map** : This file contains the list of cells for a library.The name cells contains the information its polybiasing, process type, cell optimization.

- **Usermanual.conf** : This file contains the **Product Name Library Name Library Type Library Architecture Library Track Techno Product Version History about the total no. of cell s** and also some information like **Features** and **Application** of library.

- **supporting file** : There are multiple text file containing information about cell process type, cell optimization , cell height, library short description and library architecture.

This tool generates two output files:

1.4 Output

The tool generate two output files

- **XML** : The xml file is the intermediate file. This file is generated by merging various intermediate xml file. This intermediate file is used so that after the extraction the information for the library if user want to edit the text or need to embed the information can be edit in this xml.

- **PDF Manual** : The pdf file is final output of the tool it contains Library short description, cell image, its application features and the cell description.

Chapter 2

IO Metal Porting

2.1 Introduction

IO Metal Porting is a tool used for generating the schematic, layout, symbol, of the cell. The main idea for this tool is to generate the skill script to edit the layout, schema tic, symbol, abstract which was already generated. The size of the library are very large and there are lot of cells in each library so to change each cell manually is very difficult. So to automate this process we write a script to make those changes using skill.

2.2 Views in Virtuoso

2.2.1 Symbol View

A Symbol perspective is a pictorial representation of a cell. Schematic perspective incorporates pins, image representation, marks and a choice box. Pins are enter and yield of an image. The state of the image shows the cell capacity. Marks in the image are utilized to add to the documentation of the outline. Choice box in the image, characterizes the territory of the image that an example will be selectable. A symbol view for an inverter looks as shown in figure below:

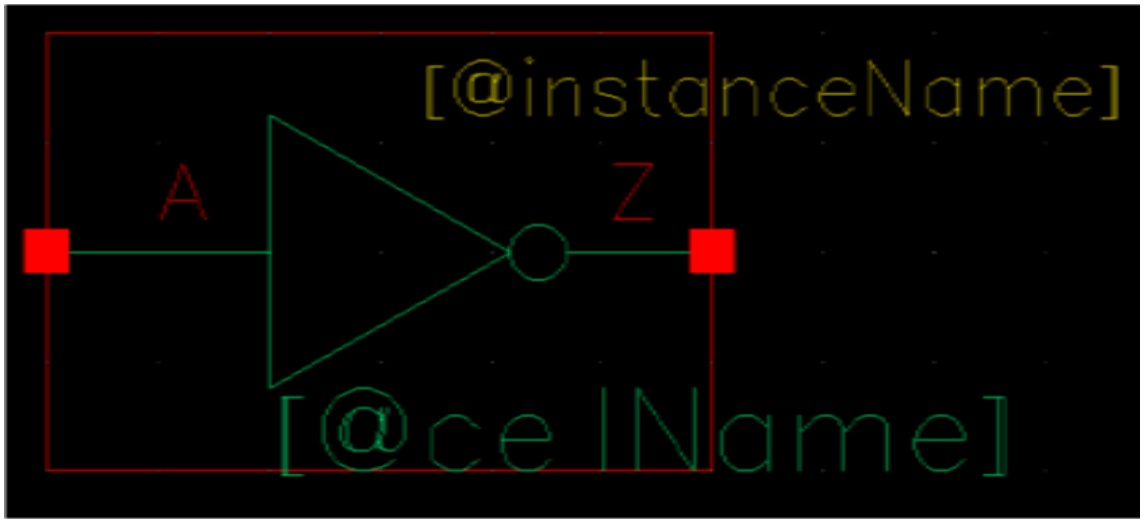

Figure 2.1: Symbol View

[3]

2.2.2 Schematic View

A schematic normally uproot all points of interest that are not significant to the data the schematic is expected to pass on, and may include impossible .A schematic is a disentangled representation of an electrical circuit. It indicates the distinctive parts of the circuit as rearranged standard images, and the force and sign associations between the gadgets. Schematic is the representation of a cell at the transistor level. A schematic perspective incorporates segment occurrences, wires and pins.

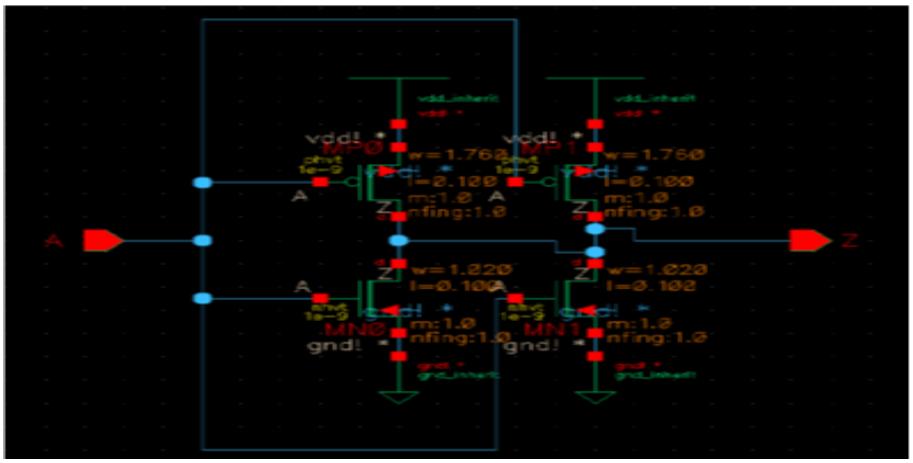

Figure 2.2: Schematic View

[3]

2.2.3 Layout View

Layout view is the actual physical representation of the electrical circuit of the cell that goes on the silicon. Different layers are used in a CAD environment to draw the physical structure keeping in mind a set of rules that need to be followed. The layout must pass a series of checks in a process known as verification. The two most common checks in the verification process are Rule Checking (DRC)(Design Rule Check),and Layout Versus Schematic (LVS).

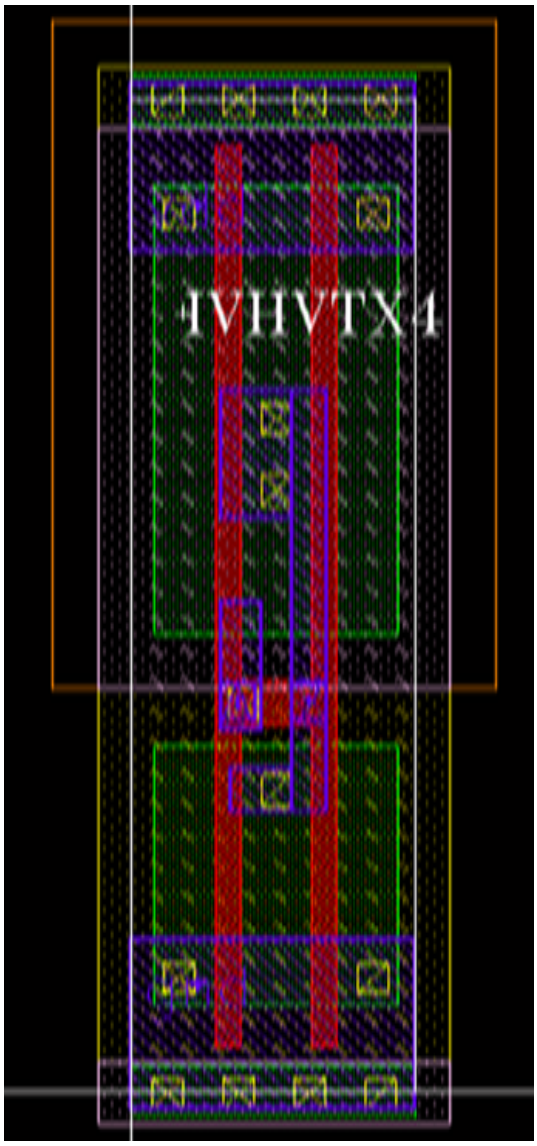

Figure 2.3: Layout View

[3]

2.2.4 Abstract view

This view is necessary for automatic layout (placement and routing) tools. It only contains information on cell boundary, routing obstacles, and I/O pins. To generate abstract view for inv

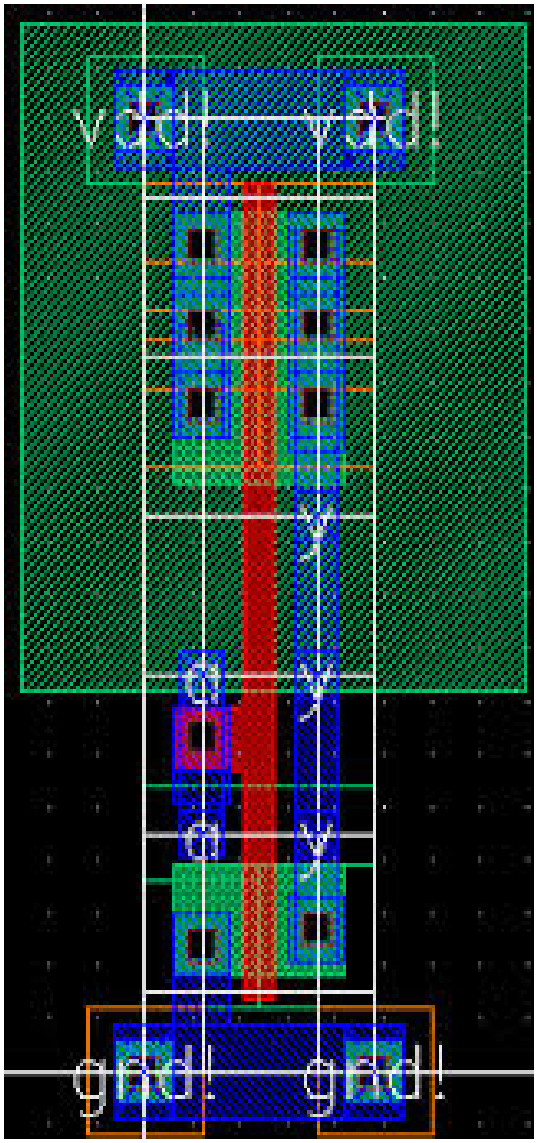

Figure 2.4: Abstract View

[3]

Chapter 3

Roadmap Generator

3.1 Introduction

This is excel based tool to generate the roadmap of the technology. Tool is used to generate the DP/Product/Technology Roadmaps in PowerPoint Slide format taking most updated data from the Mysql Schedule Database (SDB). The slides are used for program management and reviews. The tool displays the form where the user needs to select the tool using the radio button. Then user select the start date and end date, then select the techno, milestone and then generate the road map.

Chapter 4

Features Implemented

4.1 UM Generator

Following are some of the features implemented in SC Automation Tool

4.1.1 Requirement Gathering

- **Description**

  Assess the business and technical feasibility for the proposed system.Identify the people who will help specify requirements and understand their organizational bias. Define the technical environment into which the system or product will be placed. Identify "domain constraints" that limit the functionality or performance of the system or product to be built. Gathered all requirement from the previous tool and also understand the basic use of the tool from the teams who will use this tool. Identify ambiguous requirements as candidates for prototyping

- **Expected Output**

  The expectation is to obtain all the information about the tool. All feasibilty and the enhancements that need to be done in the tool, enhanced features those were not present and some new funtionality required.

- **Result**

  All the requirements are sucessfully gathered and incorporated in the tool.

4.1.2 Activity Diagram

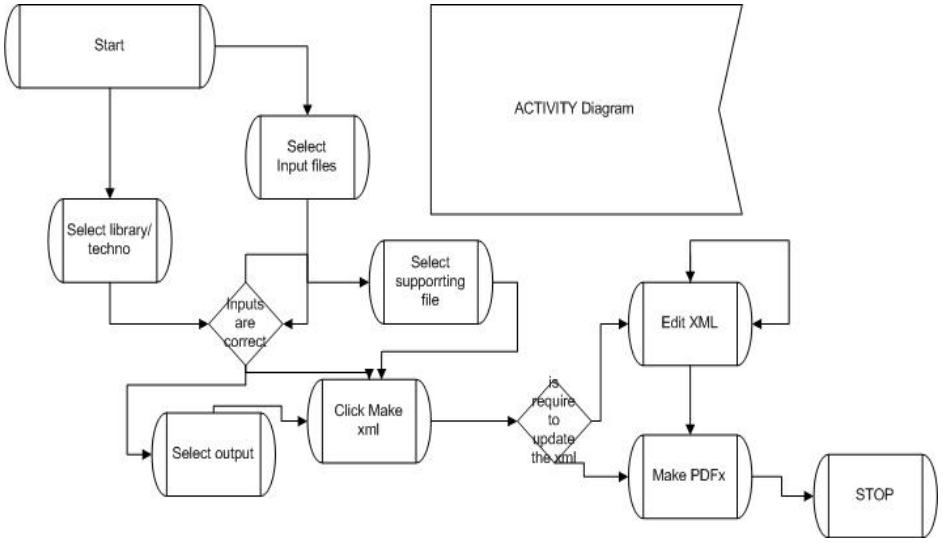

Figure 4.1: Activity Diagram

[4]

4.1.3 Sequence Diagram

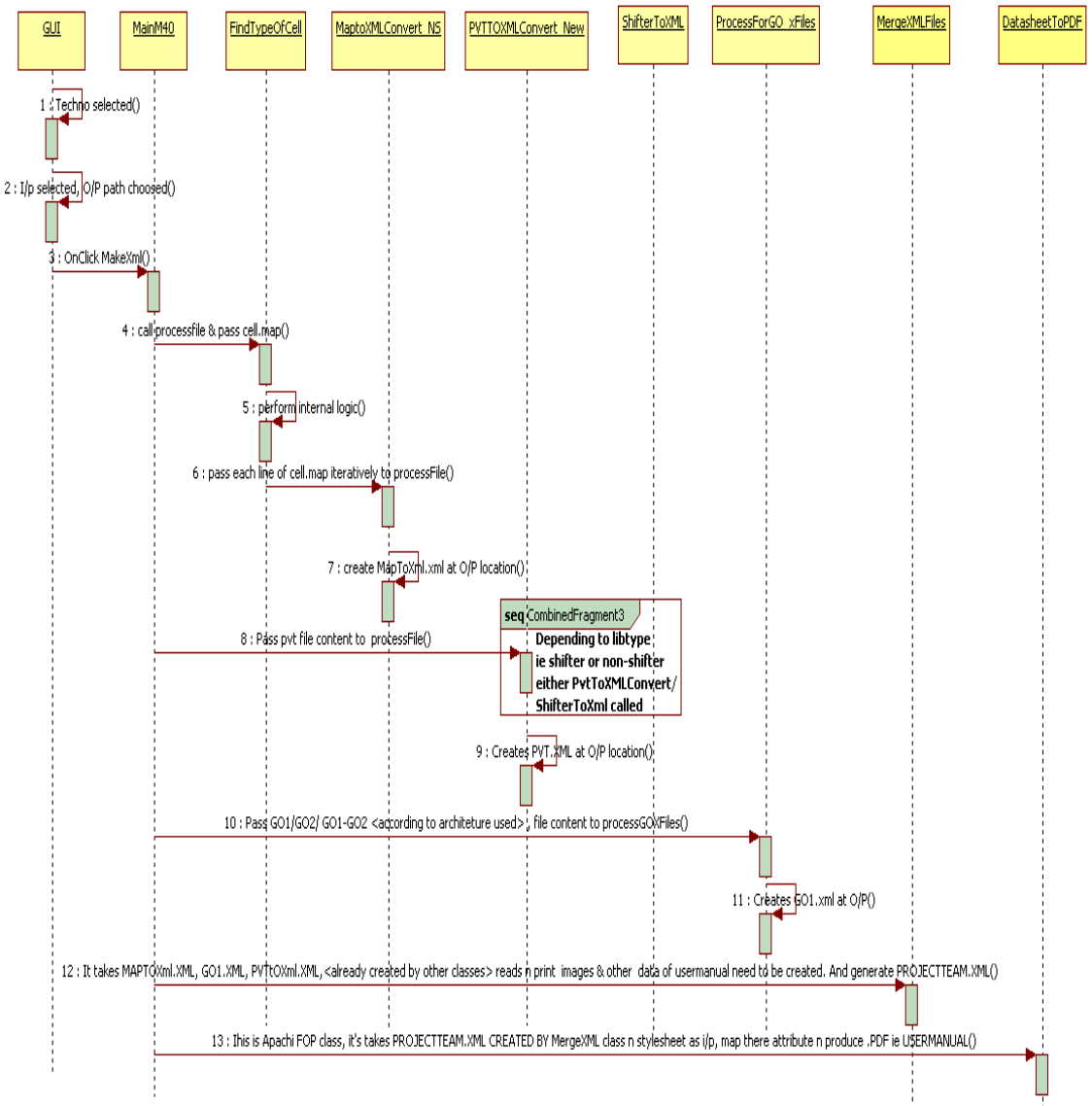

Figure 4.2: Sequence Diagram

[4]

4.1.4 Overview Diagram

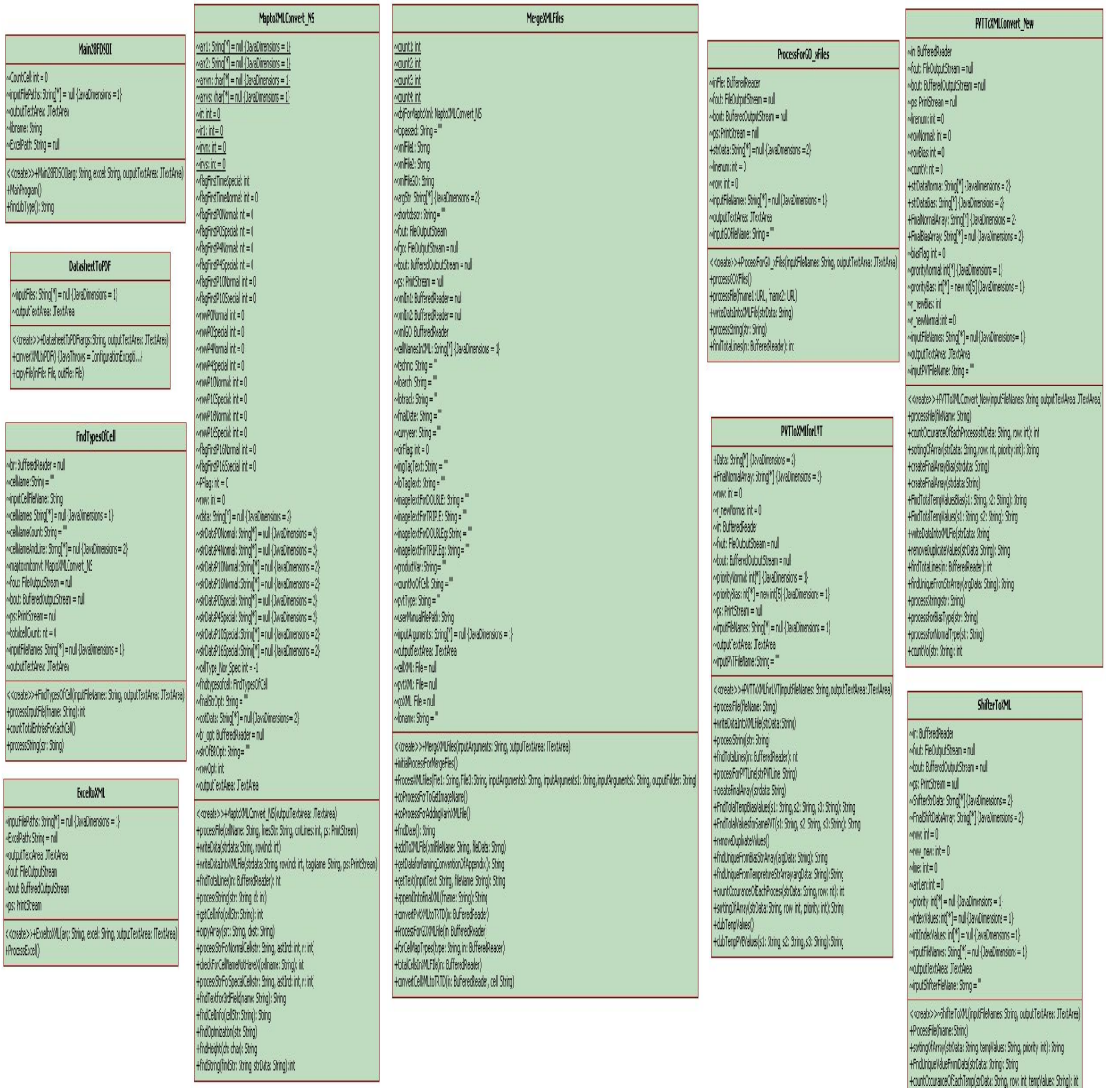

Figure 4.3: Overview Diagram

[4]

4.1.5 Designing of data structure

Design the data model for the system. Also designed class for the parsing and data population in the data model. Different classes for the extraction of data from the input files and the supporting files. Feature of overriding the supporting files from the external files.

4.1.6 GUI

Designed a GUI which can support all the requirements of the tool, It contains the support for selecting cell.map, conf config. It also having text field to write the Library name and drop down to select the techno name only available if the user does not select the conf.config file.

It also contains a check box, if users check the checkbox then a panel display which allow user to select the supporting file which he needs to override with external file. And if it uncheck the checkbox then panel will be invisible. There are buttons which are used to make xml , make PDF, and reset button to generate the xml and PDF.

There is also a provision in a tool to generate the PDF if user already have the well-defined xml.

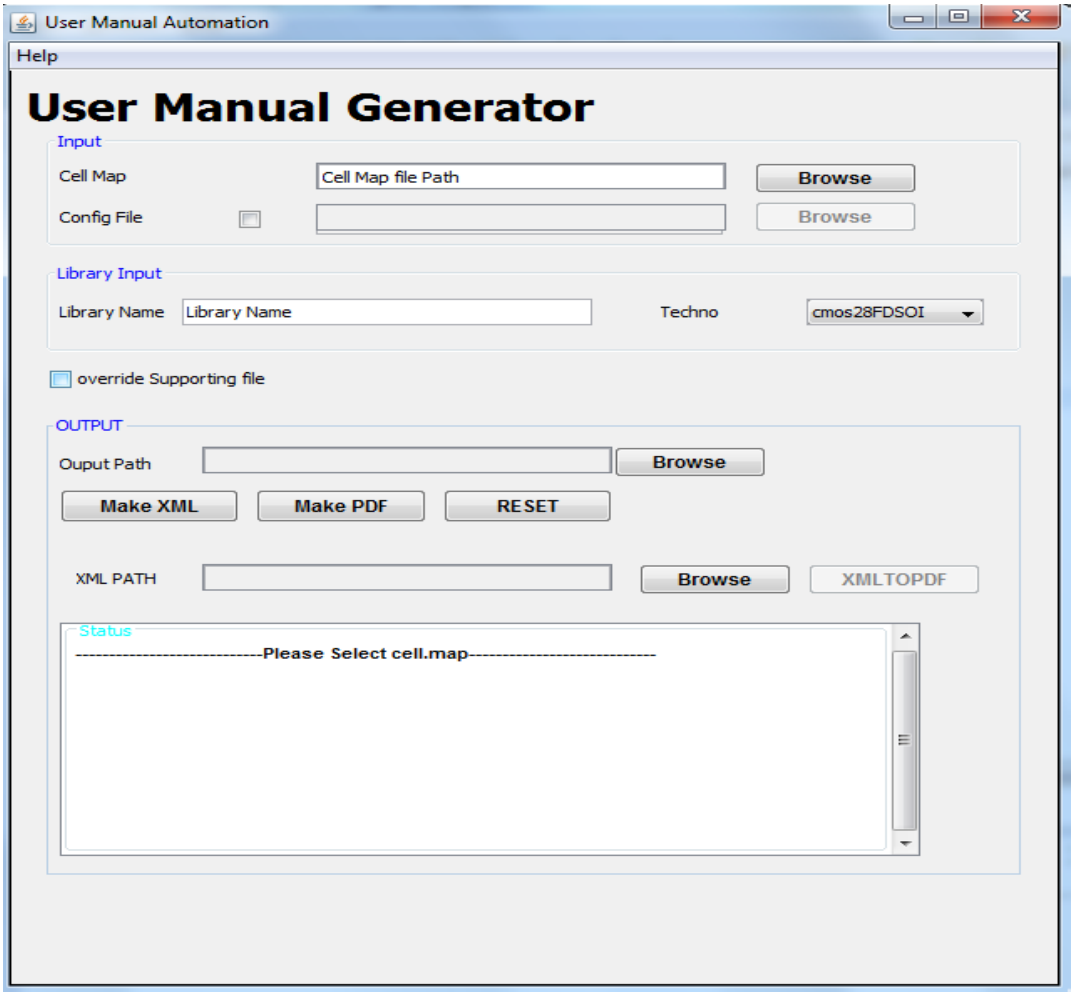

Figure 4.4: GUI of Tool

[4]

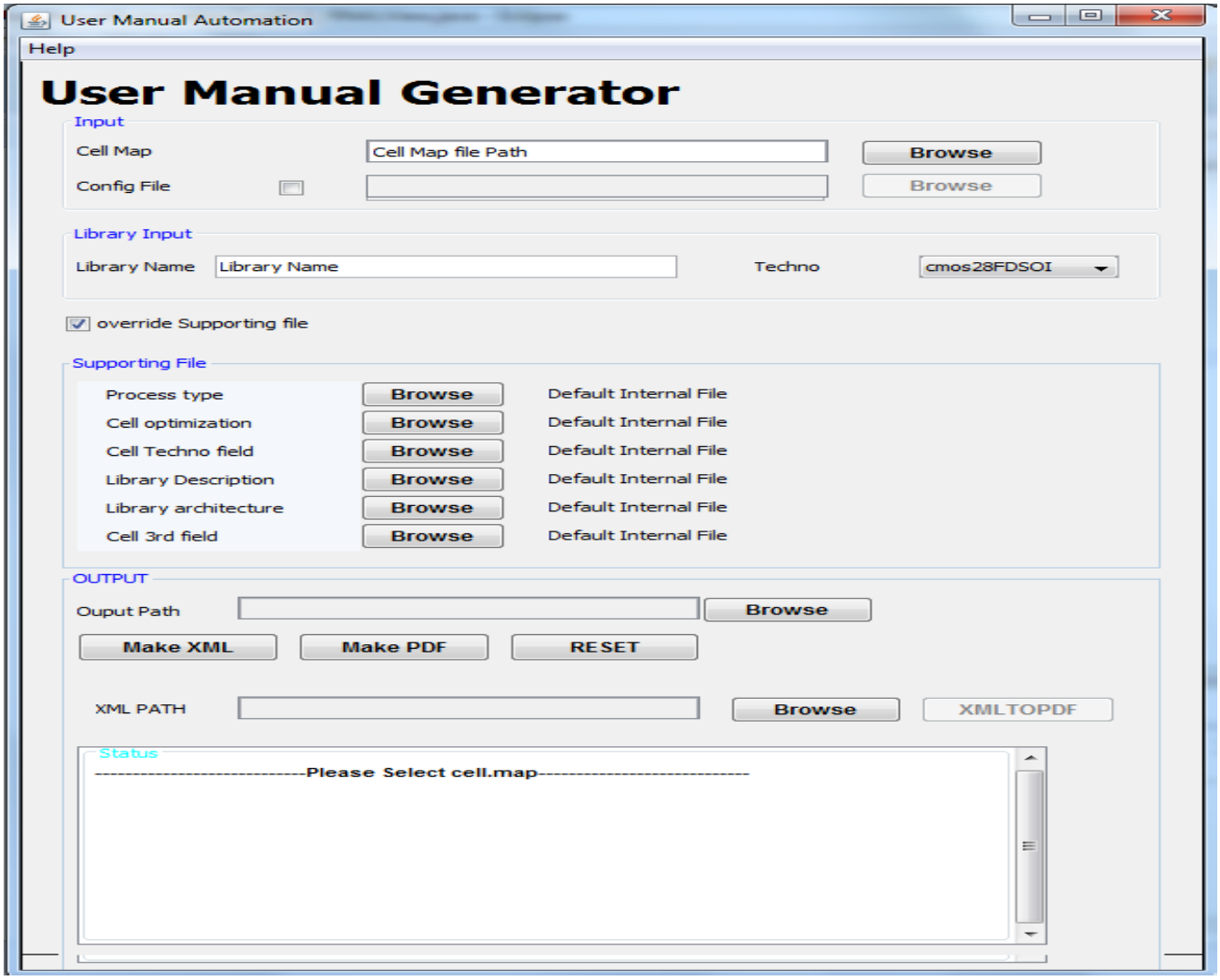

Figure 4.5: GUI with supporting file

[4]

4.1.7 DITA compline XML

Design a DITA compline xml for the tool. In DITA everything is divides into topic. So every section is a topic then there is a title and the body. All other data that is tables, images, paragraphs are come inside the body tag. The main advantages of dividing all the section into topic increase the reusability and also define the regularity in the structure of the xml. So to design such xml I used JAXB parser which is used to design a data class for the above structure, so to write the xml we can directly use this data class and generate the xml with the help of JAXB marshaller. This is a technique to generate the xml from a data class in a specified structure

4.1.8 FOP for XML to PDF

For converting the generated PDF from XML we used Apache FOP which takes xsl and xml as input and generate PDF. The XSL contains the designed information of the xml. It consist the design for the first page, headings, table.

4.2 IO Metal Porting

Following are the task did for the IO Metal Porting tool.

4.2.1 Layout view

Write the script for performing some operation for editing the layout of the tool. Write skill script for the small modules to find the unique list, replacing string with other and more scripts performing such operation

4.2.2 symbol

Write the script for performing some operation for editing the symbol view of the tool. Write the script for sorting the pin list.

4.3 Road Map Generator

Following are the task did for the Road Map Generator.

4.3.1 Grid generation

- **Problem**: In this tool initially the grid blocks are generated one by one and it increase the time of the generation
- **Solution**: Save the grid in the template and the time of generation of the roadmap the grid get copied from the template and generate the grid. This decrease the time for the generation of the road maps and enhances the functionality

4.3.2 Testing

- **Problem**: Some issues were there in the tool , like internationalization problem and other
- **Solution**: Test the tool and solve those errors by correcting the by correcting the tool.

Chapter 5

Tools and Technology

5.1 UM Generator

5.1.1 Swing

Swing is the essential Java GUI gadget toolbox. It is a piece of Oracle's Java Foundation Classes (JFC) an API for giving a graphical client interface (GUI) for Java programs[5]. Swing was produced to give a more modern set of GUI segments than the prior Abstract Window Toolkit (AWT). Swing gives a local look and feel that imitates the look and feel of a few stages, and additionally underpins a pluggable look and feel that permits provisions to observe and feel pointless to the underlying stage. It has more effective and adaptable parts than AWT. Notwithstanding well known segments, for example, catches, check boxes and marks, Swing gives a few propelled parts, for example, tabbed board, scroll sheets, trees, tables, and records. Dissimilar to AWT segments, Swing parts are not executed by stage particular code. Rather they are composed totally in Java and consequently are stage free. The expression "lightweight" is utilized to portray such a component

Architecture

Swing is a stage free, Model-View-Controller GUI schema for Java, which takes after a solitary strung programming model additionally this skeleton gives a layer of deliberation between the code structure and realistic presentation of a Swing-based GUI.

Foundations

Swing is platform-independent reason is that it is completely written in Java. Complete documentation for all classes can get from Java API Guide.

Extensible

swing is an exceedingly measured based building design, which takes into consideration the "stopping" of different custom executions of specified skeleton interfaces: Users can give their own particular custom implementation(s) of these parts to override the default usage utilizing Java's legacy system. Swing is a segment based schema, whose parts are all at last inferred from the javax.swing. Jcomponent class. Swing protests nonconcurrently fire occasions, have bound properties, and react to an archived set of routines particular to the segment. Swing segments are Java Beans parts, agreeable with the Java Beans Component Architecture particulars.

Customizable

Given the automatic rendering model of the Swing system, fine control over the points of interest of rendering of a part is conceivable. As a general example, the visual representation of a Swing part is a synthesis of a standard set of components, for example, an outskirt, inset, beautifications, and different properties. Normally, clients will automatically redo a standard Swing segment, (for example, a Jtable) by allotting particular fringes, colors, foundations, opacities, and so on. The centre part will then utilize these properties to render itself. In any case, it is likewise totally conceivable to make one of a kind GUI controls with very modified visual representation.

Lightweight UI

Swing's large amount of adaptability is reflected in its inborn capacity to override the local host working framework (OS's) GUI controls for showing itself. Swing "paints" its controls utilizing the Java 2d Apis, as opposed to calling a local client interface tool compartment. Therefore, a Swing part does not have a relating local OS GUI segment, and is allowed to render itself in any capacity that is conceivable with the underlying representation Guis. Then again, at its center, each Swing part depends on an AWT compartment, since (Swing's) Jcomponent augments (Awt's) Container. This permits

Swing to connect to the host OS's GUI administration skeleton, including the pivotal gadget/screen mappings and client connections, for example, key presses or mouse developments. Swing basically "transposes" its own (OS-freethinker) semantics over the underlying (OS-particular) segments. Along these lines, for instance, each Swing segment paints its version on the realistic gadget because of a call to component.paint(), which is characterized in (AWT) Container. Yet dissimilar to AWT parts, which appointed the painting to their OS-local "heavyweight" gadget, Swing segments are answerable for their rendering. This transposition and decoupling is not only visual, and stretches out to Swing's administration and provision of its own OS-autonomous semantics for occasions terminated inside its part regulation chains of importance. As a rule, the Swing building design designates the errand of mapping the different kinds of OS GUI semantics onto a basic, yet summed up, example to the AWT compartment. Expanding on that summed up stage, it builds its rich and complex GUI semantics as the Jcomponent model.

5.1.2 JAXP

The Java API for XML Processing (JAXP) is for transforming XML information utilizing requisitions composed within the Java programming dialect. JAXP influences the parser gauges Simple API for XML Parsing (SAX) and Document Object Model (DOM) so you can decide to parse your information as a stream of occasions or to manufacture an article representation of it. JAXP likewise underpins the Extensible Stylesheet Language Transformations (XSLT) standard, providing for you control over the presentation of the information and empowering you to change over the information to other XML records or to different configurations, for example, HTML. JAXP likewise gives namespace help, permitting you to work with Dtds that may overall have naming clashes. At long last, as of adaptation 1.4, JAXP actualizes the Streaming API for XML (Stax) standard[2]. Intended to be adaptable, JAXP permits you to utilize any XML-consistent parser from inside your requisition. It does this with what is known as a plug ability layer, which gives you a chance to module a usage of the SAX or DOM API. The plug ability layer likewise permits you to module a XSL processor, giving you a chance to control how your XML information is shown.

5.1.3 XSLT interface

The XML Stylesheet Language for Transformations, or XSLT, takes into consideration change of a XML archive into different types of data[8]. JAXP gives interfaces in bundle javax.xml.transform permitting requisitions to summon a XSLT conversion. This interface was initially called Trax (Transformation API for XML), and was created by a casual coordinated effort between the designers of various Java XSLT processors.
Main features of the interface are

- A production line class permitting the provision to select progressively which XSLT processor it wishes to utilize techniques on the industrial facility class to make a Templates item, speaking to the aggregated type of a template. This is a string safe question that might be utilized more than once, in arrangement or in parallel, to apply the same template to numerous source archives (or to the same source report with diverse parameters)

- A technique on the Templates article to make a Transformer, identifying with the executable kind of a format. This can't be granted across over strings, in any case it is serially reusable.

  The Transformer offers techniques to set format parameters and serialization decisions (for example, whether yield should be indented), and a methodology to truly run the change.

Two applied interfaces Source and Result are portrayed to address the data and yield of the change. This is a to some degree whimsical use of Java interfaces, since there is no longing that a processor

will recognize any class that executes the interface - each processor can pick which sorts of Source or Result it is primed to handle.

In practice all JAXP processors help the three standard sorts of Source and the three standard sorts of Result and maybe distinctive executions of their own.

5.1.4 JAVA JAXB Parser

JAXB is a section a bit of JAX set of Apis. JAXB grants Java designers to get to and process XML data without expecting to know XML or XML planning. For example,

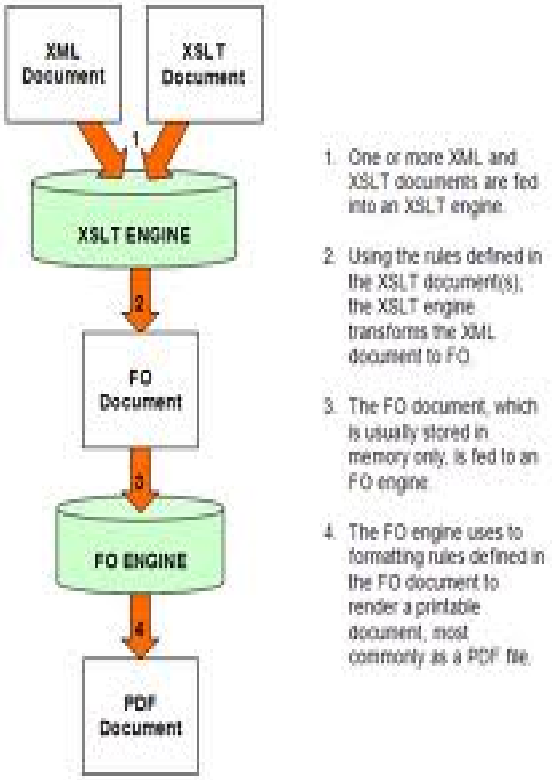

Figure 5.1: PDF Generation Flow

[2]

there's no urging motivation to make or use a SAX parser or form callback procedures. It makes use of a given example of a XML record and hence makes the obliged Java classes, identifying with that plan. Present day example focuses on XML representation of Data, for effortlessness of exchange web-organization and generally speaking. In case we have a xml representation of a database, we can use JAXB to make a tree of Java challenges and work with them in Java phrase. This is known as XML-Databinding.[6] JAXB stays for Java Architecture for XML Binding and has been open since Java 1.5 days, however became totally some bit of the lingo with Java 6.0. The framework seriously relies on upon included metadata (annotations) in your source code to characteristically plot XML record to a Java class heirarchy of leadership. This makes working with XML significantly all the more straightforward, your code more generous and less accuse tolerant to the normally cushy nature of XML data sources with missing parts and/or changing part orders.

XML is a different leveled data structure, consistently encoding a tree structure . Java

classes can furthermore be composed in such a way (and normally are) that they structure a tree pecking request. JAXB makes use of that and handles all (or to the degree that it can do so) changes to understood primitive data sorts and settled classes, achieving a totally populated data/class tree when completed. It moreover backs the Java Collection Framework and subsequently can characteristically populate and remove the different sorts of records available.

5.1.5 DITA XML

It is a standard that is defined and maintained by the OASIS DITA Technical Committee.[10] The name derives from the following components:

- Darwin: it uses the principles of specialization, which is in some ways similar to the naturalist Charles Darwin's concept of evolutionary adaptation.
- Information Typing.
- Architecture: DITA is an extensible set of structure.[11]

x

Features and limitations

- Topic orientation: DITA substance is made as themes, every a singular XML record. Ordinarily, every subject blankets a particular subject with a solitary goal, for instance, a theoretical point that gives an outline, or a procedural theme that demonstrates how to achieve an undertaking. Substance ought to be organized to look like the record structure in which it is held.
- Maps : A DITA guide is a compartment for points used to change an accumulation of substance into a production. It gives the subjects grouping and structure. A guide can incorporate relationship tables (reltables) which characterize hyperlinks between points. Maps could be settled. Maps can reference themes or different maps, and can hold a mixed bag of substance sorts and metadata.[9]
- Content reuse :Topics might be reused crosswise over numerous productions. Parts of substance inside points might be reused through the utilization of substance references (conref), a transclusion instrument.

- Metadata: DITA incorporates far reaching metadata components and characteristics, both at subject level and inside components. Restrictive content permits sifting or styling substance focused around traits for group of onlookers, stage, item, and different properties. The contingent preparing profile (.ditaval document) is utilized to distinguish which values are to be utilized for restrictive transforming.

- Information typing: DITA incorporates far reaching metadata components and characteristics, both at subject level and inside components. Restrictive content permits sifting or styling substance focused around traits for group of onlookers, stage, item, and different properties. The contingent preparing profile (.ditaval document) is utilized to distinguish which values are to be utilized for restrictive transforming.

  A (General) Task theme is proposed for a system that depicts how to fulfill an assignment. A Task point records an arrangement of steps that clients take after to handle a proposed result. The steps are held in a taskbody component, which is a specialization of the nonexclusive body component. The steps component is a specialization of a requested rundown component. Idea data is more target, holding definitions, guidelines, and rules. A Reference subject is for points that depict order linguistic use, programming directions, and other reference material, and typically holds point by point, genuine material.

- Specialization: DITA permits including new components and properties through specialization of base DITA components and characteristics. Through specialization, DITA can suit new subject sorts, component sorts, and characteristics as required for particular commercial ventures or organizations. Specializations of DITA for particular commercial enterprises, for example, the semiconductor business, are institutionalized through OASIS specialized councils or subcommittees. Numerous associations utilizing DITA additionally create their specializations.

  The extensibility of DITA grants associations to practice DITA by characterizing particular data structures and still utilize standard devices to work with them. The capability to characterize organization particular data architectures empowers organizations to utilize DITA to advance substance with metadata that is serious to them, and to authorize organization particular runs on archive structure.

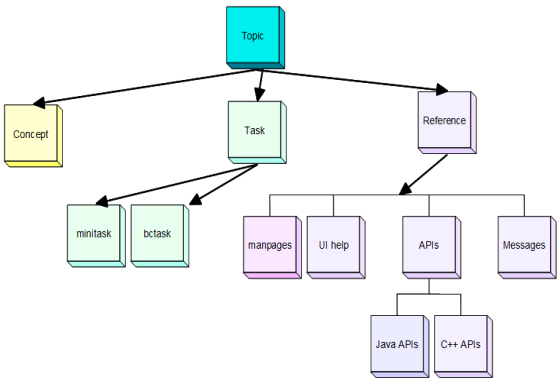

Figure 5.2: Topic Heirarchy

[9]

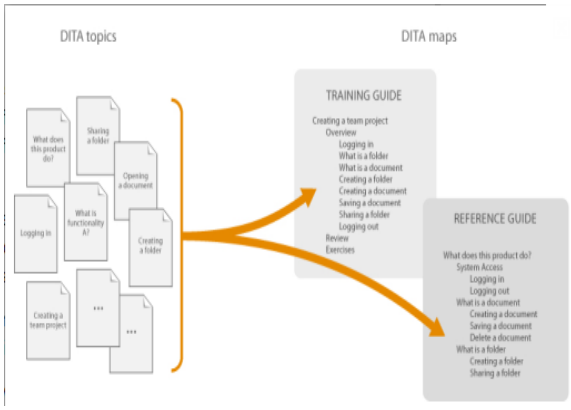

Figure 5.3: content Reuse Flow

[10]

5.2 IO Metal Porting

5.2.1 Skill

The SKILL programming dialect gives you a chance to redo and enlarge your the earth. Ability gives a sheltered, large amount programming environment that naturally handles numerous conventional framework programming operations, for example, memory administration. Ability programs might be quickly executed in the Cadence environment. Ability is perfect for fast prototyping. You can incrementally accept the steps of your calculation before consolidating them in a bigger program.[8]

Capacity administration mistakes are tenaciously the most well-known reason refered to

for calendar defers in customary programming advancement. Aptitude's programmed stockpiling administration alleviates your project of the trouble of express stockpiling administration. You increase control of your product improvement schedule.

Aptitude likewise controls famously mistake inclined framework programming errands like rundown administration what's more unpredictable exemption taking care of, permitting you to concentrate on the applicable subtle elements of your calculation or client interface plan. Your projects will be more viable on the grounds that they will be more concise.

The Cadence environment permits SKILL program advancement, for example, client interface customization. The SKILL Development Environment holds capable following, debugging, what's more profiling apparatuses for more goal-oriented projects.

Aptitude powers your speculation in Cadence engineering on the grounds that you can join existing usefulness and include new capabilities.

Aptitude permits you to get to and control all the parts of your nature's domain: the User Interface Management System, the Design Database, and the charges of any incorporated configuration apparatus. You can even inexactly couple exclusive configuration devices as discrete courses of action with Expertise's interprocess correspondence offices.

5.3 Road Map Generator

5.3.1 VBA

Visual Basic for Applications (VBA) is an execution of Microsoft's occasion driven programming dialect Visual Basic 6 and its copartnered coordinated nature (Ide).
Visual Basic for Applications empowers building client characterized capacities (Udfs), mechanizing methods and getting to Windows API and other low-level usefulness through dynamic-join libraries (Dlls). It supersedes and develops the capacities of prior requisition particular macro programming dialects, for example, Word's Wordbasic. It might be utilized to control numerous parts of the host requisition, including controlling client interface characteristics, for example, menus and toolbars, and working with custom client structures or dialog boxes[12].
As its name proposes, VBA is nearly identified with Visual Basic and utilization the Visual Basic Runtime Library, yet it can regularly just run code inside a host requisition as opposed to as a standalone program. It can, be that as it may, be utilized to control one provision from an alternate through OLE Automation. For instance, it is utilized to naturally make a Word report from Excel information, which are consequently gathered by Excel from surveyed perception sensors. VBA can utilize (however not make) (Activex/COM) Dlls, and later forms include help for class modules.

5.3.2 Web Service

Web administration will be utilized to associate the databases and information from the databases will be taken to the devices as needed. The web administration is utilized as an interface between the Census Tool and the Mysql database. The database for the Census Tool is put away in Mysql. This database has the data of the amount of assets specifically workers, sub cons and trainees at diverse destinations dealing with distinctive projects.
The web administration cooperates with the database in Mysql and produces a xml record with the comparing information. The web administration has been coded in Java for each one table in database there is a relating class in Java.

Chapter 6

Literature Survey

Investigating the Use of a SC Automation, gives the confirmation that As engineering keeps on growwing and get more unpredictable, programming engineer will be confronted with harder difficulties to completely create the product item inside the time given to them. To stay aware of this pattern, engineer should reliably search for approaches to enhance their creating practices. A SC mechanization is an apparatus that can help a client to create manual. So before any association chooses to resolve to create an instrument, it might be shrewd to first investigate its utilization inside your creating surroundings to guarantee that it is suitable to your product creating needs.

Investigation of web advancement Framework [1], There are numerous existing web provision improvement skeletons which straightforwardness the errand of web provision advancement. Struts , Cocoon and OXF (Open XML Framework) are samples for these skeletons. These schemas give numerous libraries to empower the Web provision designers to concentrate on the business forms usage instead of concentrating on programming dialect and environment related issues. In a manner the systems recommend the detachment of attentiveness toward the advancement. The business protests and methods might be more focused on by the engineers. The straightforward static pages' survey of a web requisition is carried on another stage where a large portion of the information are dynamic, perused from databases, where there exists high communication of the framework (sites) with the customers (programs). Without the structures the idea of element page creation, advancement of substance and business object-thick sites is obviously conceivable. However this requires truly programming which brings about extremely slip inclined locales that are created in long time with high extend plan. The assistance of

the structures are simpler to acknowledge as of right now. Object-social mapping (O/R mapping) for steadiness is mapping of the item turned world to social databases where the data is held in social structures. There is an impedance confound between the both planets. However mapping is important for different reasons. These days item arranged dialects like Java, C#, C++ are the basically utilized dialects by undertakings for programming activities. The provision and business information are demonstrated in an Object-arranged way. Then again, in social databases the same information is communicated in social tables which have sections and lines. In numerous ventures the mapping from the Object-Oriented world to the social world is physically coded in the provision programs. This is a safe way however it is likewise extremely lapse inclined and obliges a great deal of hand-work. There is a requirement for computerized mapping which digests from make, upgrade and erase operations. There exist systems and devices for this programmed mapping undertaking. Classes of the OO dialect are mapped to social tables. Sleep OJB (Objectrelationalbridge) and Castor JDO are a percentage of the samples for these systems.

Systems for Agile Software Developing: An Industrial Experience [3], presents the experience in performing procedures for test mechanization in lithe programming advancement Scrum. We could watch the impact of dexterous values on group association and advancement. These encounters indicated that it is plausible to extend cooperation in coordinated effort to tackle issues, hunt down coordinated devices and keeping the administration and association of a product process.the robotization is an asset to robotize dreary errands, report programming, lessens expense of venture utilizing open source apparatuses, and errand distribution in little parts. Then again, the developers' mentality in these situations are exceptionally paramount to investigation new methodologies and devices. A few lessons educated could be concentrated and they could help other programming specialists when performing dexterous robotization methodology in a comparable environment.[3]

Chapter 7

Conclusion

This report includes overview of UM Automation tool, framework for web development , IO metal Porting and road map generator.

Automation of the process is basic need in todays scenario so the tools like UM Automation, IO metal porting help. By usage of automation tool the document generation becomes easier. By using this tool tedious manual efforts can be reduced, the automation can be run on different techno for the document generation.

Bibliography

[1] Xiaofeng Yu, Yan Zhang, Tian Zhang, Linzhang Wang , "A model-driven development framework for enterprise Web services"

[2] "JAXPJava, A. P. I. "for XML Processing." (2004)."

[3] "ST internal documents and Training manuals on LIBRARY VIEWS"

[4] "SC Automation User guide"

[5] Loy, Marc, et al. Java swing. O'Reilly, 2012..

[6] Ort, Ed, and Bhakti Mehta. "Java architecture for xml binding (jaxb)." Sun Developer Network (2003).

[7] Other ST Sources.

[8] "Skill lang user guide"

[9] "Darwin Information Typing Architecture (DITA) Version 1.2". Retrieved 10 October 2012."

[10] "OASIS Darwin Information Typing Architecture (DITA) TC". OASIS. Retrieved 5 October 2012."

[11] "Frequently Asked Questions about the Darwin Information Typing Architecture". IBM Corporation. Retrieved 10 October 2012.

[12] Chapra, Steven C. Power Programming with VBA/Excel. Prentice-Hall, Inc., 2007.