Automation Techniques for the handling of Photomask Data

J. Gordon Hughes, Leslie J. Drennan (Compugraphics International Ltd., Glenrothes, Fife, KY7 4NT, Scotland)

ABSTRACT

This paper reviews four different strategies that can be used in the automated processing of photomask manufacturing data. The first of these is the automation of specific process steps such as the generation of Barcodes and Titling data, or specific software to perform format conversion, job deck generation or automated fracturing. The second strategy is the automation of complete flows that are specific to a customer or product type. This type of automation can be successfully built using the primitives developed by the first strategy. The third strategy is the automation of generic flow steps that are then applied to all customers. This is a much more powerful solution, but is significantly more demanding in the effort required for the design and implementation. The final strategy, the integration of automation over multiple complete supply chains is a desirable end game. The paper will report on the latest activities of the European Commission Fifth Framework project called Automatic Mask Order Processing System (AutoMOPS) that is developing a prototype implementation of a distributed server architecture that will allow automation and integration within the photomask manufacturing supply chain. The advantages and weaknesses of each of these techniques are reviewed using examples of automation which has been undertaken within Compugraphics over the previous decade. The interaction between these techniques is reviewed, as well as the evolution of solutions from each technique and the systems environment where these techniques are used.

Keywords: Process Automation, Data Fracturing, Supply Chain Integration, AutoMOPS

INTRODUCTION

This paper is designed to provide information on three different aspects of the automation of data processing as it applies to Photomask Manufacture. The first aim is to provide an overview of the software products which have been developed within Compugraphics over the last decade, and to provide an insight into the development process which was used to create them. The second aim is to provide an overview of the European Commission Fifth Framework Programme project entitled, The Automatic Mask Order Processing System (AutoMOPS) which is addressing the role of supplier management and order processing within the Photomask Manufacturing supply chain of the Semiconductor Industry. This can be viewed as a higher level of abstraction to the existing types of Automation which are currently in use at the Compugraphics Glenrothes manufacturing site. An analysis of the different types of automation currently used can provide a framework for reviewing the types of automation which are possible, and to review the interactions between these automation techniques in the context of a Merchant Photomask Manufacturing environment.

The paper starts with a review of the computer environment which is being used at the Compugraphics Glenrothes Mask Shop, as this environment has a significant impact on the way which automation has developed within Compugraphics. This is followed by looking at some examples of Simple Task Automation which were developed about a decade ago, and are still in use today as valuable automation tools. With these tools in place, the paper then reviews some of the customer specific automation flows which have been implemented over the last decade, and then looks at more recent examples of automation where generic automation has been done to allow aspects of the photomask manufacturing process to be automated for all jobs and for all customers. This is followed by a short overview of the AutoMOPS project and what has been achieved to date. The paper concludes by reviewing the interaction of the different types of automation techniques, and looks at the practical details of their implementation within the Compugraphics environment.

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 07/18/2016 Terms of Use: http://spiedigitallibrary.org/ss/TermsOfUse.aspx

THE COMPUTING ENVIROMENT

Figure 1 shows a graph of the number of CPUs which have been used for data processing at the Compugraphics Glenrothes site since general purpose data processing services were first provided during 1985. This graph shows two of the three trends which have been common to all Mask Manufacturing environments over the last decade. For any specific company or site, the timing of specific changes will vary, but the authors believe that the trends shown by the Compugraphics environment are typical of those throughout the industry.

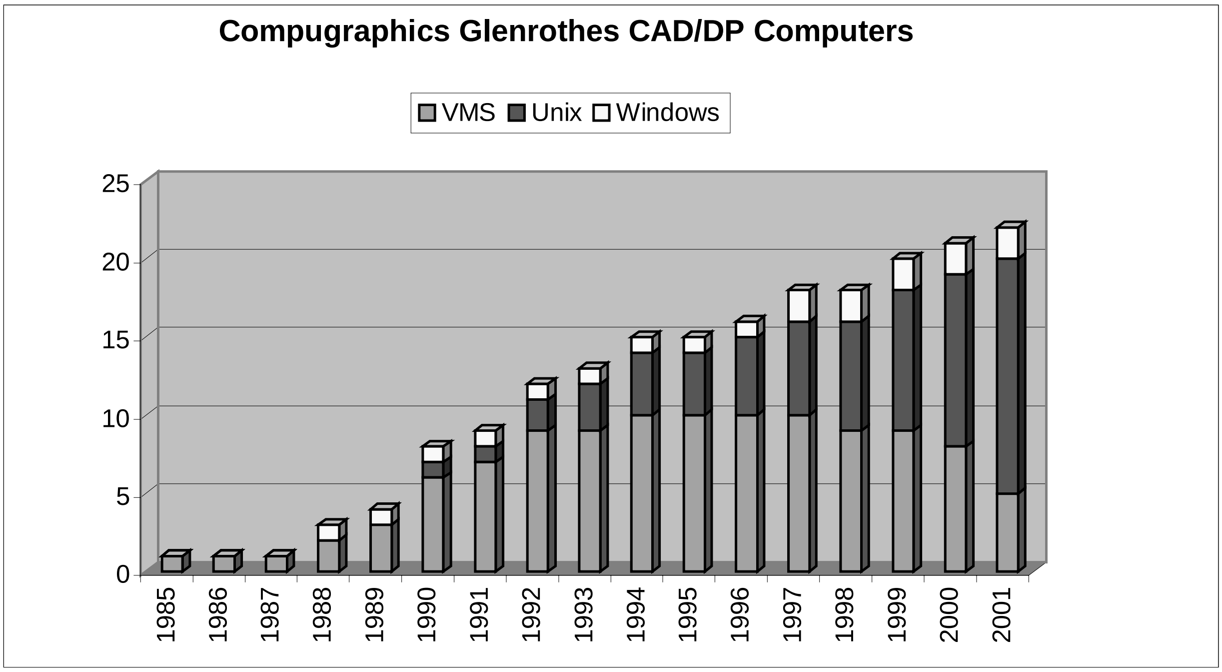


Figure 1 : Computers used for Data Processing at the Compugraphics Glenrothes Site

The first trend is an increase in the number of CPUs used within the manufacturing process. At Compugraphics this has been an area of continuous growth, matching the expansion of the company. The second trend is from the use of VAX/VMS as the platform of choice during the 1980s to Unix as the platform of choice today. Compugraphics is atypical in its continued use of a VMS environment for a large proportion of data processing tasks, although recent years have now seen the number of Unix systems overtake those of the VAX and Alpha systems. The authors believe that the use of VMS during the 1990s provided a stable, secure and robust environment for data processing, which is still difficult to provide using alternative operating systems.

The third trend is more difficult to graph, although it is obvious that a CPU purchased today will be many times the power of a CPU purchased during the 1980s (for a similar cost). The authors' assertion is that the increase in power of CPUs over the last decade has followed Moore's Law, and although the graph shown in figure 1 shows a linear upwards trend, the reality is that over the last decade there has been an exponential growth in the computing power which has been applied to the processing of photomask data.

As a result of the mixed Solaris and VMS environments existing within Compugraphics, an important aspect of automation is that where possible, tools should be portable between these two environments. Most of the software development has been done using the traditional programming languages C[1] and Imp[2], both of which have compilers for VMS and Solaris. The Imp programming language was developed at the University of Edinburgh as a language to implement Operating Systems where efficiency was a major consideration. Imp is a block structured language with a portable exception handling system, useful facilities for the handling of text strings and complex data structures and easy

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 07/18/2016 Terms of Use: http://spiedigitallibrary.org/ss/TermsOfUse.aspx

to use interfaces to operating systems procedure libraries, which are useful when writing systems applications. As a result it became a popular language for general purpose programming. It also happens to be the first major programming language used by the authors, who both gained degrees in Computer Science from the University of Edinburgh.

Software development started at Compugraphics in 1989. Thanks to one of the author's backgrounds (JGH) of working in a software house, many of the issues which are required by the QS9000 and BS7799 standards, such as version control, tracking and documentation, have always been a feature of the software developed internally at Compugraphics.

Another key strength of the software development environment at Compugraphics is that the software development staff have always worked alongside the Data Processing staff, and this has resulted in very high levels of interaction between the software users and the software developers. In most cases, the software development staff have had a dual role of both software development and data processing work depending on the priorities of the current work load. The development of internal software applications is the dual responsibility of both the Data Processing and IT departments.

Prior to looking at examples of automation, it is important to consider two building blocks which are prerequisites to all other types of data processing automation. These are the System and the Application building blocks. At Compugraphics the Systems Building Blocks consist of the use of the Compaq VMS and Sun Solaris operating systems as well as other systems utilities for these environments. The Applications building blocks which play an important role in further automation are the Lattice Logic Ltd. Shapesmith[3,4] and Transcription Enterprises Inc CATS software systems. The Shapesmith software was used for processes which were automated prior to the purchase of the CATS software at Compugraphics, although the CATS software is also capable of performing all of the automation which is currently performed by the Shapesmith software.

AUTOMATION OF SIMPLE TASKS

This paper presents three older examples of the automation of simple tasks with the data processing environment at Compugraphics. These examples are significant as they represent software which has been proven to be useful over a long period of time, and are also examples of software which have been integrated into other types of automation flow.

The first example is a command called TITLE which allows the user to enter a series of text strings at the terminal or from a data file and this information is used to generate MEBES data files which contain the human readable text. Prior to any software developments within Compugraphics, when human readable text was required, this would involve a graphical editing session to generate a customized Calma GDS II database[5], which was then fractured to produce the required MEBES data. Following the philosophy of the Lattice Logic Chipsmith silicon compiler[6], one of the first programming tasks was to write the TITLE command as "silicon compiler" for text data. Figure 2 shows an overview of the way that the TITLE command is structured. The user views the TITLE command as a single entity, where text is input and MEBES data is output, however internally the TITLE command generates a file of Caltech Intermediate Format[5] (CIF) and uses the VMS Digital Command Language (DCL) to invoke the Shapesmith software to automatically perform the conversion. The title command was first produced in 1989, and has been extended to add additional characters such as microns and copyright, as well as additional options to control the placement and size of the text.

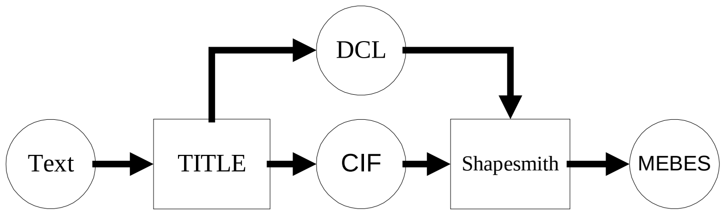


Figure 2: The structure of the TITLE command

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 07/18/2016 Terms of Use: http://spiedigitallibrary.org/ss/TermsOfUse.aspx

The second example is a command called BARCODE, which has a very similar structure to the TITLE command. This command also provides the user with a "text in" and "MEBES data out" application. Figure 3 shows the similarities to the TITLE command, but in addition to using DCL to invoke the Shapesmith software, this is also used to invoke the TITLE command to generate the human readable text beside the barcode data. This is an important example of how simple task automation can be re-used to provide building blocks for further levels of automation.

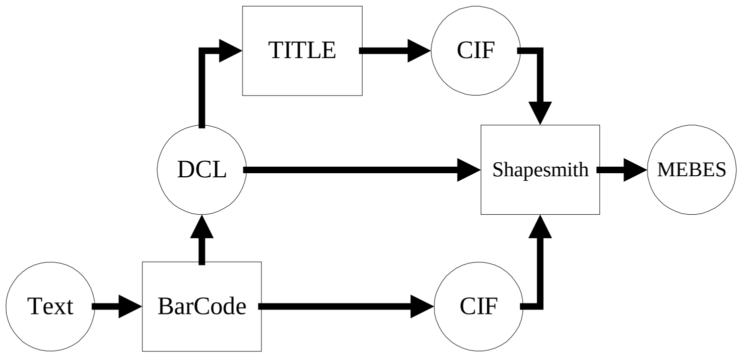


Figure 3: The structure of the BARCODE command

The initial release of the BARCODE command was to generate GCA barcodes, but this has since been extended to generate 8 types of barcode for all major reticle types. Although other barcode generating software is now available, the Compugraphics BARCODE command provides the users with a single easy to use interface, rather than different user interfaces depending on the type of barcode being generated.

The third example is a command called DOCATS. In the early days of the CATS software there was a very limited range of control commands that could be specified within the CATS Include File language (.cinc). Compugraphics also identified a requirement to be able to use the CATS software as a command line driven interface, which could then be used from within other programs to provide further levels of automation. Figure 4 shows the software's internal structure. A .cinc file is generated based on the user's command line specifications, and uses DCL to invoke the BATCAT command to submit a batch job to process this data. From a user perspective, the command provides the ability to take a set of MEBES files in, and produce a set of output MEBES files which have been processed according to the command line options specified. The software will automatically perform the CATS WRITEFILE command and where appropriate, the software will automatically perform a logical XOR function to ensure that the output data is still a valid representation of the input data. The following example is the command that can be used to reverse (and XOR) a set of data files for a job:

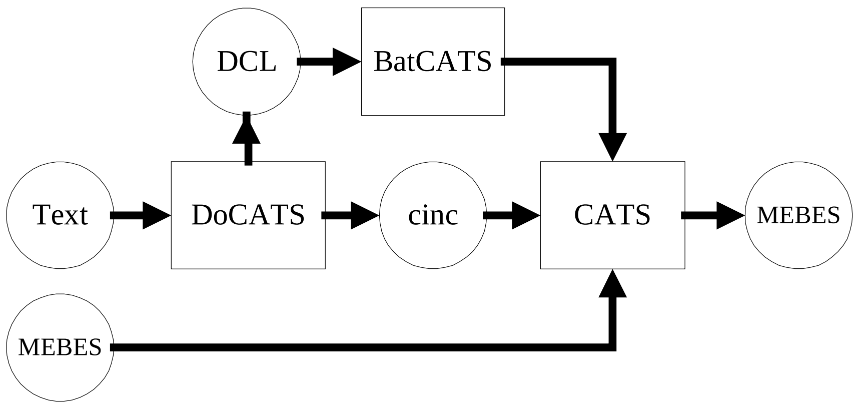


Figure 4: The structure of the DOCATS command

```
$ DOCATS C*.%% /FORMAT=MEBES/REVERSE=YES/NAME=D%%%%%%%%.%%   CINC_file_name
```

Another advantage of the DOCATS command is that the Compugraphics CATS default settings are automatically applied to all jobs which are processed on the batch queues. The software was first released in 1991, and subsequent changes have been required with new CATS releases and changes to the Compugraphics system configurations or CATS default settings.

There are many other examples of simple task automation software, in particular a large number of programs associated with job deck manipulation, however hopefully these examples with provide a flavor of the implementations and their

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 07/18/2016 Terms of Use: http://spiedigitallibrary.org/ss/TermsOfUse.aspx

dependencies. In particular new task automation depends on existing systems, applications and task automation software. More complex forms of automation can be achieved by the implementation of multiple simple tasks. It is important when implementing these tasks that there is a clear definition of what the automation is expected to achieve, including well defined interfaces and how the software will fit into the existing process flows. By providing a command line interface to all of the software, this makes the re-use within other programs easier. Note that this does not preclude the possibility of alternative graphics interfaces to specific applications, the Pinebush Hyperplot software is a good example of what can be achieved for a mixed graphical and command line driven program.

AUTOMATION OF CUSTOMER SPECIFIC FLOWS

Once the tools are in place for simple Task Automation, these can be extended to provide automation for customer specific processes. An example from the early 1990s of this is a command called xxxREL (where xxx represents the internal Compugraphics customer code). This software is used to release customer provided ROM data to manufacturing, and the software will perform all of the tasks necessary, only requiring the operator to check that the software has run correctly. The internal structure of this software is shown in figure 5. The ROM data may be presented in a number of different data formats, usually MEBES, but if not then the DOCATS command can be used to perform the required conversion. The ROM data for the mask writing system needs to be joined with existing Scribe data, and a job specific job deck is constructed from templates. Certain product types require barcodes and/or human readable text to be produced, which are generated using the BARCODE and TITLE commands. The initial implementation was for two different data formats being used to generate 2 different product types, produced by three different writing systems with the masks being supplied to 2 waferfab lines. The software has been subsequently extended to cover 7 product types being processed for 5 waferfab lines, and is still in use today, a decade after it's initial implementation.

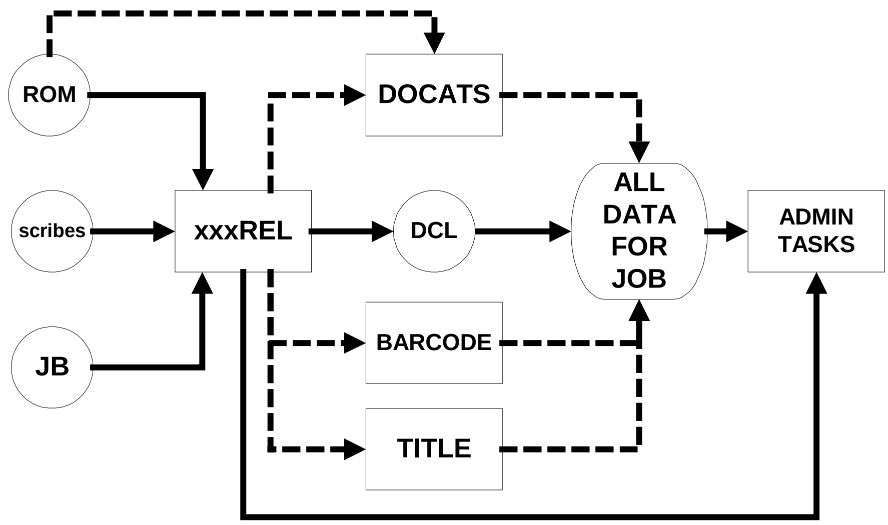


Figure 5: The structure of the xxxREL software

A second example, dating from the mid 1990s is a command called xxxOR2JB where the software will read an order form (a text file) from a customer who supplies MEBES data, generate the required job deck from the order form and use the DOCATS software to optimize the data for the mask writing tool being used. Other Task Automation software is used to complete the administrative tasks which are required as part of the job data preparation. The initial implementation was for a single product for a single site, and this has since been extended for several different product types being sent to several different sites. An important issue with this type of automation is that software changes are required if the customer makes any changes to their order form layout.

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 07/18/2016 Terms of Use: http://spiedigitallibrary.org/ss/TermsOfUse.aspx

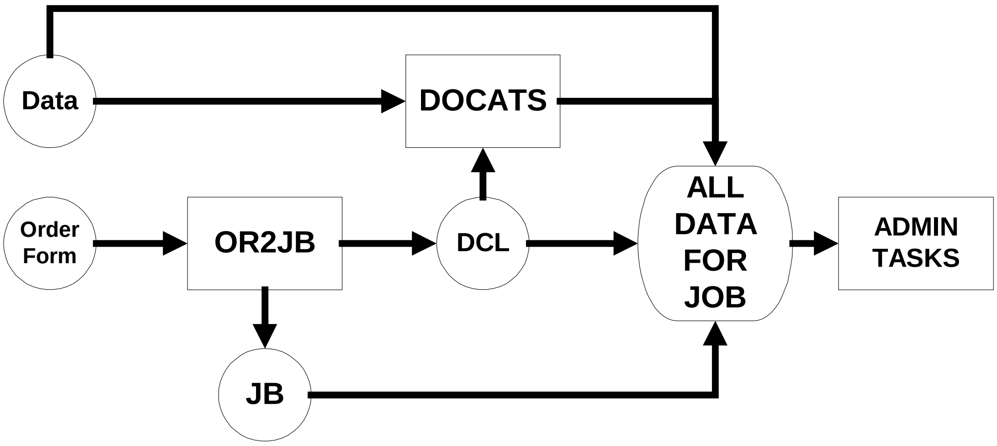


Figure 6: The structure of the xxxOR2JB software

A third example is really a set of several commands, mostly dating from the late 1990s. Each command is called DO_xxx and provides the automation of a major customer input. In most cases the data is provided as Calma GDS II data, although a few implementations are for customers providing MEBES data. The software analyses the customer order form (in some cases converting the Postscript back into text), and uses this information to generate .cinc files for the CATS processing of the information, and the job deck file required by the job. Again the stability of the customer order form is an important issue, and consequently many of these implementations have been done as joint projects with customers to ensure compatibility of working practices. In most cases the software also implements multiple different product types, usually 1X Master masks or Reticles, but is some cases, multiple types of Reticle. It is also often the case that the .cinc files often have to optimize the data for multiple types of mask writing tool.

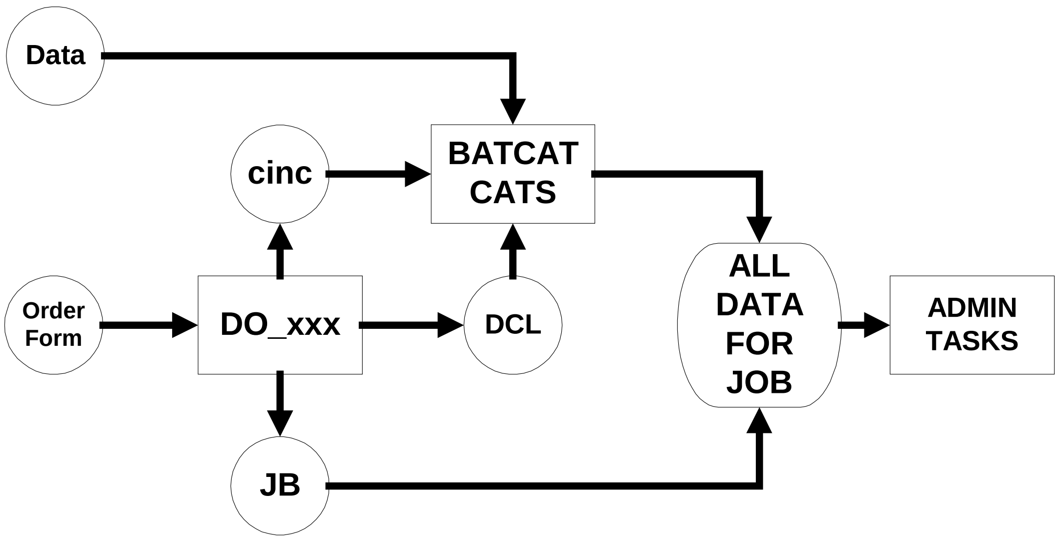


Figure 7: The structure of the DO_xxx software

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 07/18/2016 Terms of Use: http://spiedigitallibrary.org/ss/TermsOfUse.aspx

As can be seen from these examples, this type of automation can take longer to implement, and is only suitable for major customers where there will be sufficient payback on the investment required. It is important that the person undertaking the implementation has a sufficient knowledge of the customer requirements. This type of automation only works well for customers who have standardized requirements of well defined product types, and as mentioned previously, is very dependent on the stability of the customer order forms.

AUTOMATION OF GENERIC FLOWS

A more powerful but more challenging type of automation is the production of generic software which can be applied to a series of tasks for all customer jobs. Figure 8 shows two examples of Generic Task Automation software which can be found in use at Compugraphics, namely the FOR_WRITER and the 2MPOFDB programs.

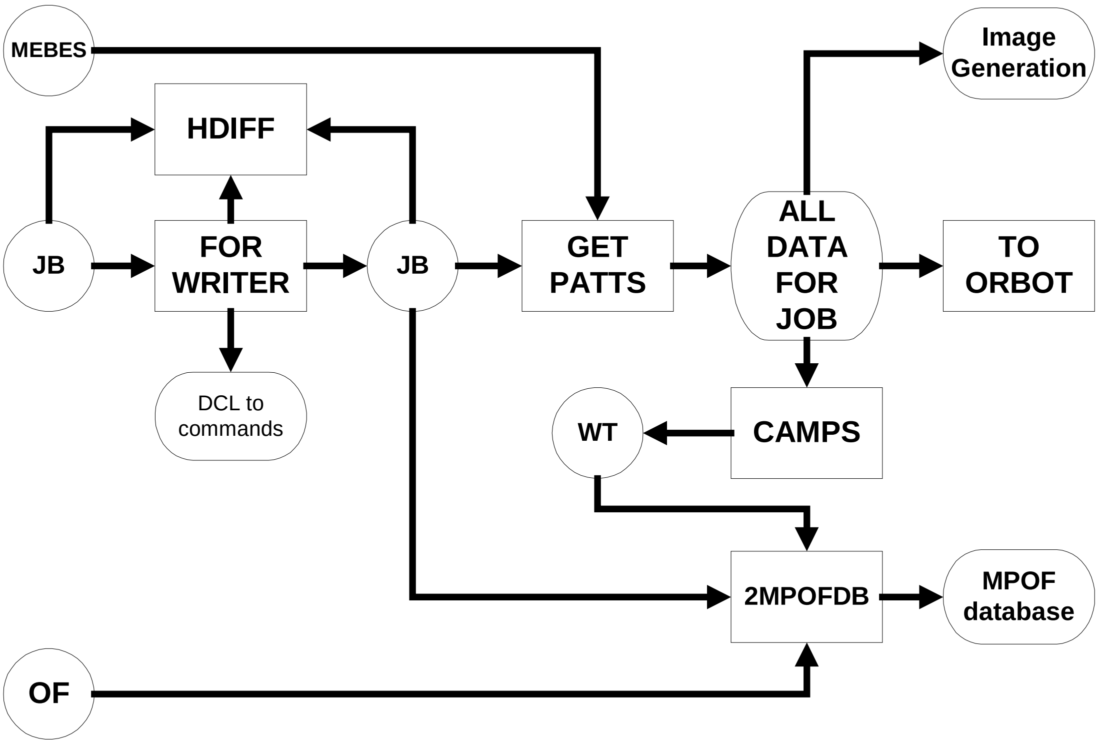


Fig 8: The structure of the FOR_WRITER and 2MPOFDB commands

The FOR_WRITER software provides a standard back-end flow which is used for all jobs prior to manufacture. This flow consists of the following sub-tasks. First the FOR_WRITER software checks the input job deck for compatibility with the selected writing system; if necessary, the software will alter the job deck to suit the selected writing system, and also add product and writing system specific SPC information to the job deck. The FOR_WRITER software then calls a Compugraphics generated program called HDIFF which is used to present the operator with the differences between the input and the output job decks with any differences highlighted. The FOR_WRITER software will then call the GET_PATTS command to ensure that all the MEBES data files which the job requires are present. This may include copying customer specific files from standard libraries, and to also check that there are no extra MEBES data files present in the job directory. Once all the data is available for the job, the FOR_WRITER software will adjust the file protections to allow the clean room staff access to the data, and runs the CAMPS software to generate an estimated write time file which also includes information on the percentage of data coverage. The FOR_WRITER software will automatically send the data to other systems for the preparation of the Inspection data, using the TOORBOT command if requested by the operator, and finally invoke the 2MPOFDB command.

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 07/18/2016 Terms of Use: http://spiedigitallibrary.org/ss/TermsOfUse.aspx

The 2MPOFDB command consists of a number of configuration files which are derived from Customer specifications and quotations. On a per-job basis, the job deck, the write times file and, for major customers, the customer order form, have their information extracted and used to complete the administrative database, called the Mask Production Order Form (MPOF) DataBase, which controls the customer requirements. For major customers, there is a special module provided for each customer which handles the data extraction from their order form. In cases where there is no order form the information is extracted from the configuration files using a matching system based on the ideas behind the Prolog[7] programming language. For example from the job deck the software knows about the customer name and the mask size, and this is used to filter the information in the configuration files. If only one item of information is found, then this will be used, otherwise the operator is required to clarify which option applies for this job. By using this method of data matching the amount of input required by the operator for a job is significantly reduced, and since the introduction of this software in 1997, errors associated with masks being made with the wrong requirements have been almost totally eliminated.

Within Compugraphics we have regression test suites which are used before any of the software is released, but in the case of Generic applications, these have proved to be particularly useful. Any error made with a piece of software which is used with every job would be potentially catastrophic.

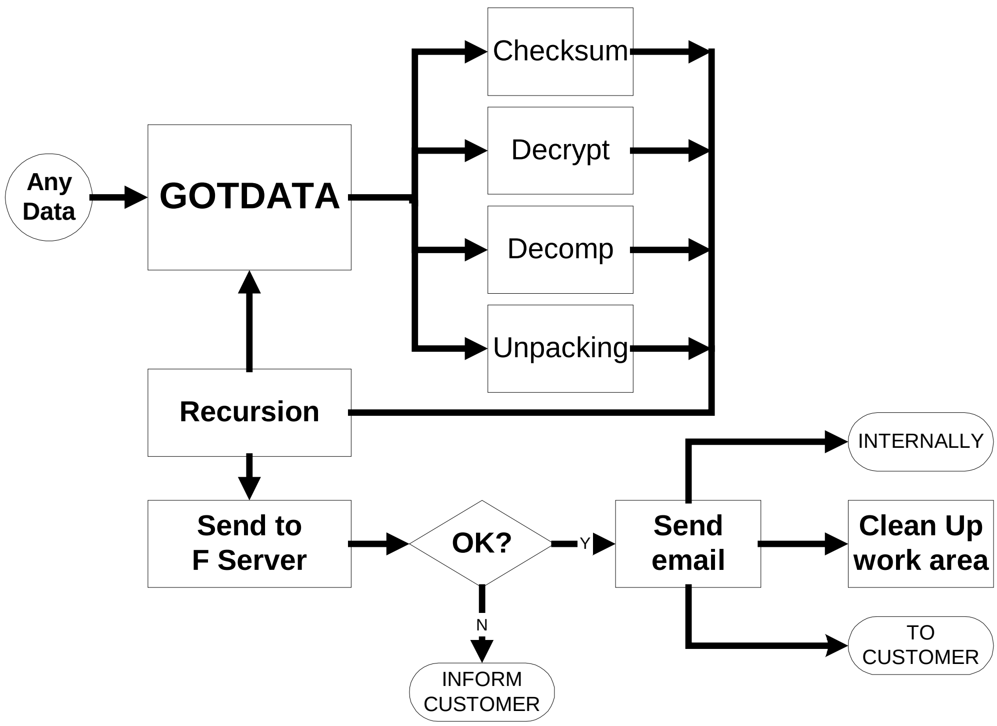


Fig 9: The structure of the GOTDATA command

A final example of Generic Automation from Compugraphics is a recent command called GOTDATA which provides a standard mechanism for the treatment of all incoming customer data. This data may require to have a checksum applied, to be decrypted, decompressed or unpacked. Further more, these operations may be required in any order, as different customers have different preferences for the order that these operations are performed on outgoing data. The program needs to apply itself recursively until the data handling is complete. Once the data handling is complete, the data is moved from our De-Militarized Zone (DMZ) Network on to a file server on our Internal Network, and this transfer is checked by the operator. If the operator confirms that the transfer is OK, then confirmation e-mail is sent internally to production control and externally to the customer if required. The data on the DMZ Network will also be automatically

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 07/18/2016 Terms of Use: http://spiedigitallibrary.org/ss/TermsOfUse.aspx

deleted. The software also maintains a log file of all its operations, which gives very good traceability. The two challenges of this software are the implementation of the recursion and the fact that a customer can start a new transfer while the GOTDATA software is processing the data from a previous one. Both of these issues are implementation details that are left as an exercise for the reader!

This type of Generic Automation results in more complex software than previous types of automation, and as a result this software is even more time consuming to implement, all of the examples given have been incrementally developed over a number of years. In order to do the implementation, the programmer needs to have a very clean understanding of the processing and systems which underlie the tasks being automated. As these programs are covering all cases for all customers it is important to have an escape mechanism which can be used if there is a need to process new types of job, for example the ability in the 2MPOFDB software to suppress the use of customer specific modules, and run directly from the configuration files. Any Generic Automation software is built on the existing Systems, Applications and Task automation tools which are available. In spite of the cost of implementing this type of automation, it also has a very powerful pay back when implemented correctly.

AUTOMATION WITHIN A SUPPLY CHAIN

The Mask Industry Quality Assessments, reported at previous BACUS symposiums[8], show that photomask ordering is an error prone process. After any data processing requirements, there may typically be 50 items of information required by the clean room staff to correctly make a simple mask, and this number increases significantly in the case of newer technologies such as OPC or Phase Shift masks. The increasing complexity of these newer technologies is a further potential source of error. It is generally accepted that the use of automation is a means of reducing error rates. Some evidence to confirm this is provided by the fact that Compugraphics have consistently scored better than average when benchmarking against the Mask Industry Quality Assessments, the authors argue that this is thanks to the automation which is already in place at Compugraphics. There is also general agreement that the use of Standards can help to reduce error rates, improve cycle times and result in better time to market. These arguments were the basis of a proposal made to the European Commission to fund a project titled, The Automatic Mask Order Processing System (AutoMOPS). These arguments were accepted, and project IST-1999-01332 was started in February 2000, with a funding of 2.1M€ over 2 years. The AutoMOPS consortium consists of two software companies, Nimble who produce administrative software and Sigma-C who produce data handling software; and two photomask supply chains, one consisting of the Infineon Technology Mask Shop supplying to the Altis Semiconductor waferfab; and the other consisting of the Compugraphics Mask Shop supplying to the Alcatel Microelectronics waferfab.

One of the first tasks of the project was the development of a Generic Meta-model to capture the interactions within the Photomask ordering supply chain. The starting point of these discussions was the diagram that is shown in figure 10. It is interesting to compare this diagram with diagrams produced by the SEMI Design Concept to Silicon Data Path Task Force[9]. The left hand and two right hand boxes match the Task Force's diagram, while the 4 remaining boxes represent the "Dark Space" identified by the Task Force. The diagram in figure 10 represents the AutoMOPS project's attempts to chart this Dark Space. The Fracture operation is typically the conversion of a data format such as Calma GDS II into a machine specific format such as MEBES. After the data files are created for the writing system, it is necessary to make up a file with machine control information, such as the MEBES Job Deck, and this is the Job Preparation operation. The Image Enhancement operations include tasks such as Optical Proximity Correction or Laser Proximity Correction. The Mask Layout operation is the generation of non-design specific information for the photomask, such as scribe and Barcode data. Within the AutoMOPS consortium, there are several different types of interaction between the members of the supply chain, for example in some cases the link between the designer and the mask shop is to the left of the Dark Space, where as in other cases this interface is to the right of the Dark Space. Indeed there are examples of every permutation of these boxes between the designer and the mask shop! Another important aspect of the AutoMOPS project is that this diagram can be used to chart the logistical and financial interactions of the supply chain, as well as the technical challenges of the data handling aspects.

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 07/18/2016 Terms of Use: http://spiedigitallibrary.org/ss/TermsOfUse.aspx

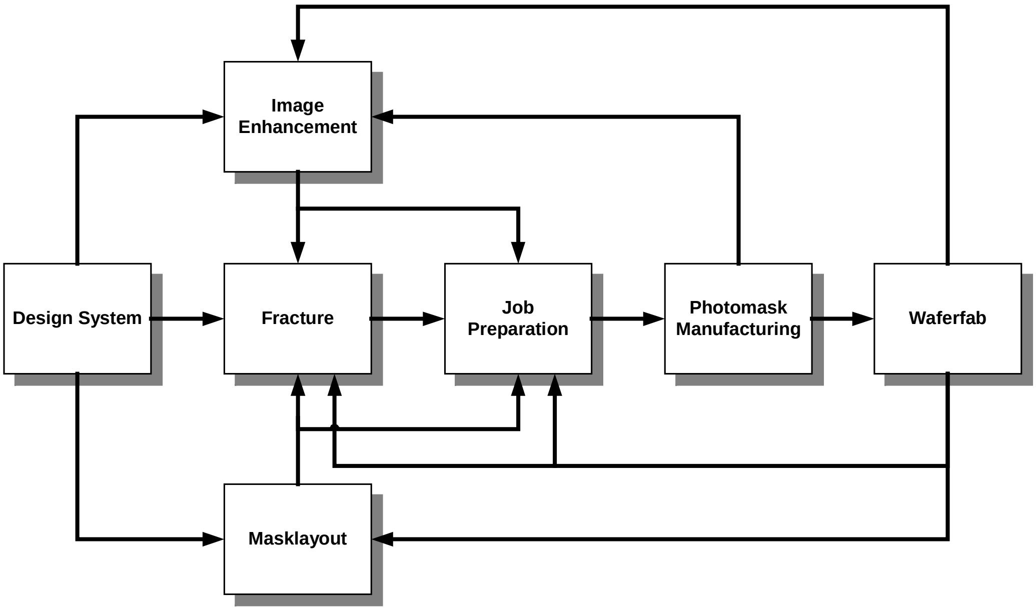


Figure 10: A model of the Photomask Supply Chain

The AutoMOPS project advocates the use of standardization to allow a customer independent solution to be provided. A result of this is that most of the AutoMOPS partners are actively involved with the SEMI Standards Program, in particular the European based Photomask Qualification Terminology Task Force[10] and envisage the use of an XML based version on the SEMI P10 Standard[11] as the most appropriate way to convey the technical information about the mask requirements.

Following the completion of the Generic Meta-model, this was used to define the specifications of an AutoMOPS Server system which would manage each of these task specific boxes from figure 10 at each site within the Supply Chain. Further details of this design of the AutoMOPS server was presented at the European Mask Conference 2000[12]. The first prototype implementation of this server was demonstrated in May 2001, with links to the Infineon-Altis Work In Progress system and the Sigma-C JOBMAN software. A refined implementation of the server is now under development and this will be demonstrated within the Compugraphics-Alcatel Supply Chain.

Further information about the AutoMOPS project can be found at the project's Internet Web site, www.automops.org .

CONCLUSIONS

Initially it was thought that a structure such as a stack or a pyramid would be a good representation of the interaction between the different types of Automation techniques. Figure 11 shows a diagram which reflects the experience at Compugraphics of these different techniques. A starting point is the Systems and Applications software which provides the basis for the generation of any Task Automation. Generic Process Automation is at the upper end of a continuous spectrum of automation complexity that ranges from Simple Task Automation through to Generic Process Automation. Any type of Task Automation can also be re-used within Customer Specific Automation, for example the xxxREL software now also uses the FOR_WRITER and 2MPOFDB software to complete the processing stages. A desirable end goal is that of Supply Chain Automation, which can use standards to promote non-customer specific solutions. Any effort which is devoted to the production of Customer Specific Automation is time and effort which may be better spent reaching the more desirable long term goals of Generic or Supply Chain Automation, although the authors are sure that Customer Specific Automation will continue for pragmatic or logistical reasons.

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 07/18/2016 Terms of Use: http://spiedigitallibrary.org/ss/TermsOfUse.aspx

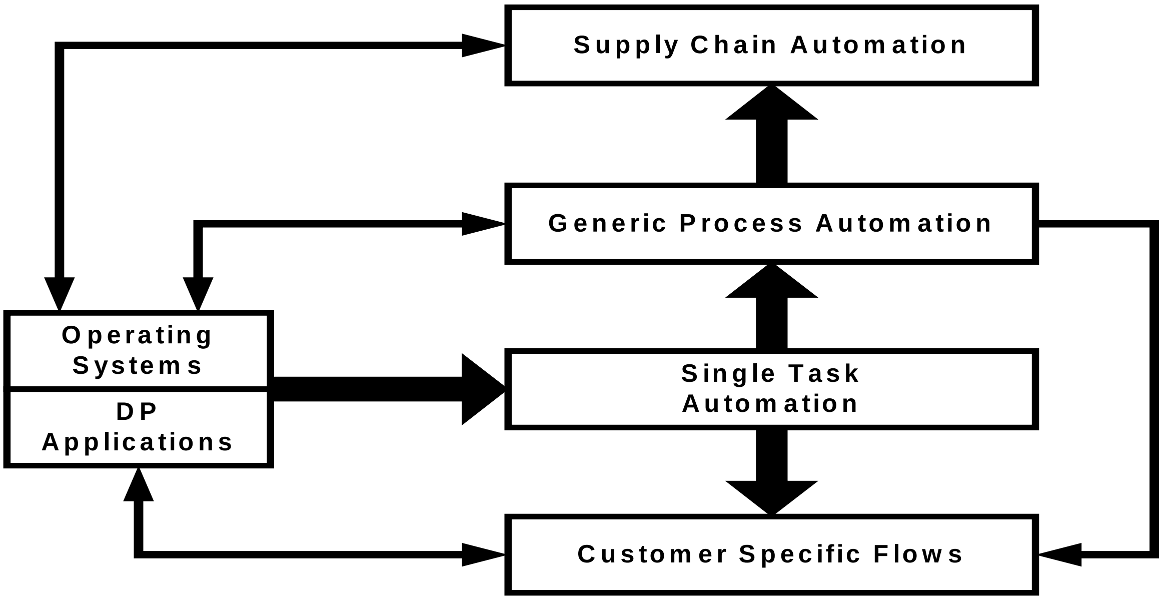
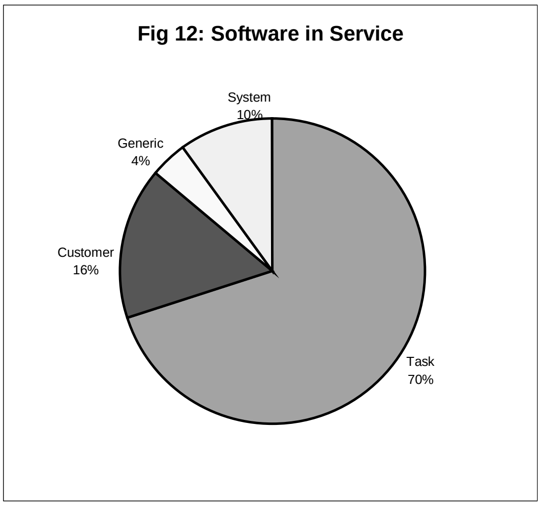
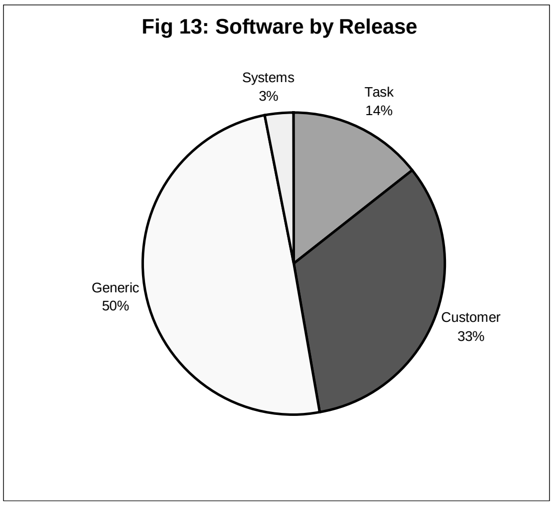


Figure 11: Interaction of Data Automation Techniques

Figure 12 is a diagram which shows the proportions of different types of automation which have been produced and are currently in use within Compugraphics. This shows that Task Automation represents the highest proportion of the software, with Generic solutions representing the smallest proportion. However figure 13 shows the same categories, but this time plots the number of software releases (changes) made to the software over the first 6 months of this year. The amount of effort which the more complex Generic tools require to support them now becomes clearly visible. When doing further analysis of the reasons for change, about 80% of the changes found were caused by either changes in operational practice within Compugraphics or changes associated with customer's requirements or operational practices. These exist in equal proportions, but the start of this year has seen the introduction of major new equipment and alternative material suppliers at Compugraphics, so the Compugraphics reasons may not be typically as high. Also when the customer reasons are analyzed, two thirds of these changes were uncontrolled, in other words we were reacting to a change made by the customer without prior notice.





Downloaded From: http://proceedings.spiedigitallibrary.org/ on 07/18/2016 Terms of Use: http://spiedigitallibrary.org/ss/TermsOfUse.aspx

The authors conclude that the evidence presented in this paper shows that the use of automation provides a valuable tool for improving of the speed and accuracy of data handling tasks, and that we have shown that there are different types of automation which can be implemented, which require different levels of resource and commitment, in particular:

- Simple Task Automation is the fastest form to implement
- Customer Specific Automation is only a viable approach for larger customers with sufficient investment payback
- Generic Automation is more powerful but harder to implement and support
- Supply Chain Automation using standards is the real way forward

ACKNOWLEDGEMENTS

The authors would like to thank the following individuals and organizations for their assistance with the work described in this paper.

- Lindsay Gill, Leslie Forrest and Blair Martin for their work in the generation of software for Compugraphics, as well as all members of the Compugraphics Data Processing department for their suggestions and feedback.

- The European Commission for their support of past research projects 5014, 8892(TRAM), 23928(OPTIMA) and the currently running IST-1999-10332(AutoMOPS).

- The staff from the other companies who are members of the AutoMOPS consortium, namely: Alcatel Microelectronics, Altis Semiconductors, Infineon Technologies, Nimble, and Sigma-C, along with the other Compugraphics staff working on this project.

REFERENCES

1. Kernighan & Ritchie, “The C Programming Language”, 2nd edition, Prentice Hall, 1988
2. Robertson, “The IMP-77 Language”, Edinburgh University Department of Computer Science, Internal Report CSR-19-77, 3rd Edition 1983.
3. Lattice Logic, “The SHAPESMITH documentation set”, Version 1, Lattice Logic, 1985
4. ES2, “The SHAPESMITH documentation set”, Version 4, European Silicon Structures, 1989
5. Rubin, “Computer Aids For VLSI Design”, Addison-Wesley, 1987
6. Gray, Buchannan, & Roberson, “Designing Gate Arrays Using a Silicon Compiler” 19th DAC, 1982.
7. Clocksin & Mellish, “Programming in Prolog”, Springer-Verlag, 1981
8. Grenon, “1999 Mask Industry Quality Assessment”, BACUS 1999, SPIE Vol. 3873 Part 1, 1999
9. SEMI TFOF, “Design Concept to Silicon Data Path Task Force”, 16-Jul-2001
   Distributed with the Microlithography Committee Meeting Minutes dated 17-Jul-2001
10. Jonckheere, Vandenberghe, Wiaux, Verhaegen & Ronse, “Reticle quality needs for advanced 193nm lithography” PMJ 2001, Proc. SPIE 2001, in press
11. SEMI Standard, P10-0301 Specification of Data Structures of Photomask Orders, SEMI 2001
12. DeRidder, Filies, Rodriguez, & Kuijken, “AutoMOPS: B2B and B2C in mask making”, EMC 2000, Proc. SPIE 2000.

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 07/18/2016 Terms of Use: http://spiedigitallibrary.org/ss/TermsOfUse.aspx