Data Modeling for Capacity Requirements Planning Qimonda Portugal

Ana Isabel Duarte Monteiro

Relatório do Estágio Curricular da LGEI 2006/2007

Orientador na FEUP: Prof. João Falcão e Cunha

Orientador na Qimonda Portugal: Eng. Peter Madera

Faculdade de Engenharia da Universidade do Porto

Licenciatura em Gestão e Engenharia Industrial

2007-09 -20

Aos meus pais.

Resumo

O presente projecto foi desenvolvido na Qimonda Portugal, empresa da área dos semicondutores electrónicos, e teve como objectivo o levantamento de requisitos e especificação do modelo de dados para uma solução que viesse a melhorar o cálculo do planeamento de capacidade de produção e de investimento em equipamentos. Simultaneamente desejava-se facilitar os processos de fabrico ao nível do controlo de produção, em particular do escalonamento de produção.

O resultado último deste projecto é um modelo com mais de 30 entidades e 80 atributos. Para este modelo tomar a forma final foi necessário uma análise extensa dos principais problemas que ele deveria ser capaz de responder. Desta análise concluiu-se que os principais problemas prendiam-se com a gestão do produto e dos recursos, como por exemplo problema de agregação e relação entre produtos e problemas de alocação de recursos.

Os utilizadores deste modelo não serão utilizadores humanos mas sim outros sistemas informáticos. Este projecto tem como objectivo integrar um RPC, Repositório Principal de Dados (“Master Data”) que relaciona vários tipos de informação através de vários sistemas. Assim, os utilizadores deste sistema serão aplicações informáticas que também integrem o RPC.

A última fase do projecto está relacionada com o desenvolvimento das funcionalidades que o modelo deve permitir, como por exemplo: funcionalidades que permitam a agregação da data (espaço temporal); ou mecanismos para calcular capacidade disponível. Consoante as necessidades futuras, novas funcionalidades podem vir a ser inseridas no modelo caso seja desejado.

A implementação do sistema ficou destinada a um projecto futuro, que está agora a ser iniciado. Quando o modelo estiver implementado e testado para a fábrica de Portugal é objectivo integrar este sistema em outras fábricas da Qimonda e por este motivo desde o inicio da especificação pretendeu-se que o modelo fosse o mais dinâmico e flexível possível, para depois ser mais fácil de vir a ser integrado em outras fábricas.

Abstract

The project described in this document was developed at Qimonda Portugal, a company in the semiconductors industry. The proposed objective was to collect requirements and specify a data model that could improve the production capacity planning and equipment investment business processes. Simultaneously, this tool aimed to ease production processes, specially concerning production controlling and shop floor control processes.

The project's result is a model with more then 30 entities and 80 attributes. An extensive analysis of all the issues it should solve was held until it acquired its final structure. This analysis pointed that main problems were due to product and resources management such as the aggregation and relation between products and resources allocation.

The users of this model will be other information systems and not human users. As the main focus of this project was to integrate master data which combines various information from several information systems, the users are those systems which will extract and use the master data defined in here.

The last project phase is related with the development of the functionalities that the model should incorporate such as, for example, functions that allow date aggregation (time frames) or mechanisms to calculate available capacity from the underlying more detailed information.

A system implementation was planned for another project, starting right from the end of this one. Firstly, the system should be implemented and tested in Qimonda PT but as soon as it is consistent the objective is to implement it in other Qimonda sites. For that reason the model was, since the beginning, pretended to be as dynamic and flexible as possible in order to have a fast and easy implementation in any Qimonda site.

Agradecimentos

Em primeiro lugar, os meus agradecimentos são direccionados para os meus orientadores de estágio, o Sr. Eng. Peter Madera e o Sr. Prof. João Falcão e Cunha, por toda a disponibilidade e motivação oferecidas e acima de tudo pelos ensinamentos que me foram dados.

Agradeço, à Qimonda Portugal e à Faculdade de Engenharia da Universidade do Porto a oportunidade que me foi dada de realizar este estágio.

Quero agradecer a todos aqueles que trabalharam comigo ao longo destes seis meses, em particular aos Engenheiros: Alexandra Castro; Mariana Menezes; Pedro Rodrigues e Pedro Dias, pela paciência e pelo tempo que disponibilizaram para a minha formação.

Quero também agradecer aos meus pais, ao Pedro e a todos os amigos pela ajuda e incentivo dado ao longo de todo o meu percurso académico.

A vivência destes 6 meses contribuiu sem dúvida em larga escala para a minha formação pessoal e integração no mundo de trabalho.

A todos muito obrigada.

Contents

1 Introduction

1.1 Scope and Objectives of the Internship

The internship project at Qimonda AG is in the scope of the fifth and final year of Industrial Engineering and Management at Faculdade de Engenharia da Universidade do Porto. It began in February 2007.

The main objective of this internship is to specify a new solution to improve Qimonda capacity requirement planning.

The graphic below explains the milestones that were proposed for the internship and the period of time they happen.

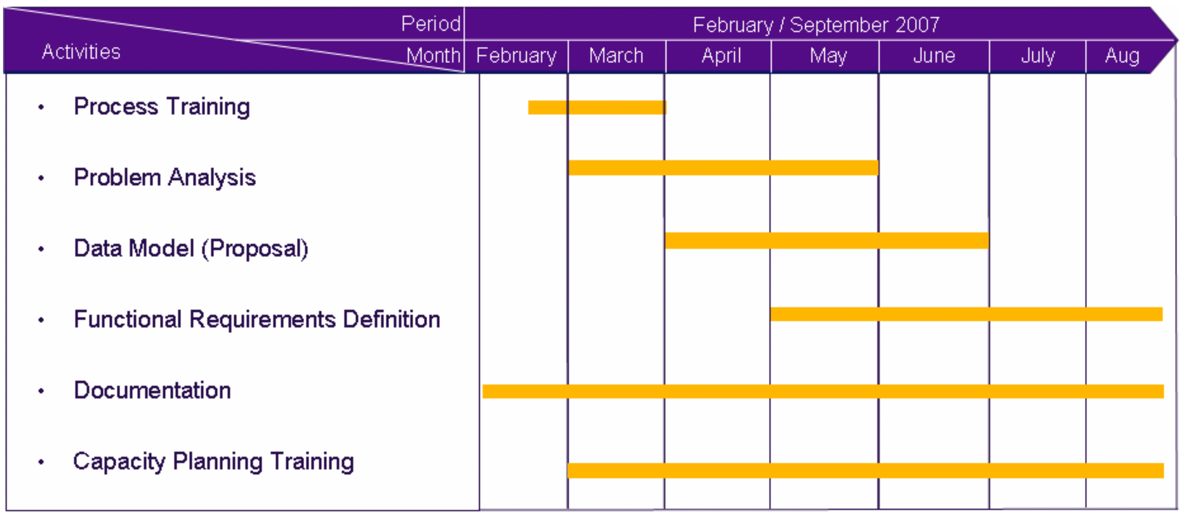


Figure 1 - The Gant Graphic

Process Training has the objective to introduce the company to new collaborators. During these weeks induction training was done. Induction training is a training for the company operators but also helpful for all the others collaborators whose jobs will be related with the production.

Capacity Planning Training was the main period of the internship. Indeed, this project is completely linked with the capacity planning. Because of this, since early stage of the project the capacity planning training was started.

The next step was the Problem Analysis. Problem Analysis consists in the analysis of the main problems of the company related with the three business processes. Once the Problem Analysis started, the Data Model also started. All the phases are completely related.

Definition of Data Model, although not the longest period, was certainty the hardest one. The Resource and Capacity Data Model is the main result of this project so it was the part of the project that was more optimized.

Functional Requirements Definition initially was a chapter with some suggestion of functionalities for the model, but with the curse of the project and the development of the data model was extended and now is an important part of the project.

As this project involves a lot of concepts and assumptions the documentation started since the beginning of the project so that nothing was forgotten.

1.2 The Group Qimonda AG

Qimonda Technology AG was founded on May 2006 when the semiconductor operations of the parent company, Infineon Technology AG, were spun off to form a separate legal entity. Qimonda AG has its headquarter in Munich-Germany.

Infineon Technology AG was founded in 1999 and was also a spin off from the Siemens Technology AG.

The name and brand identity of Qimonda expresses the philosophy and personality of the company, illustrating its vision and values. The word “Qimonda” carries different meanings and allows associations in different languages. “Qi” stands for breathing and flowing energy. Combination of the English word “key” and the latin “mundus” is intuitively understood: Qimonda = “key to the world”. [1]

Figure 2 - Qimonda Logo

Nowadays, Qimonda AG is the third world leader in memories after Samsung and Hynix and its main objective is to become the second one in this ranking. In the picture below is shown the market share in the calendar year of 2006.

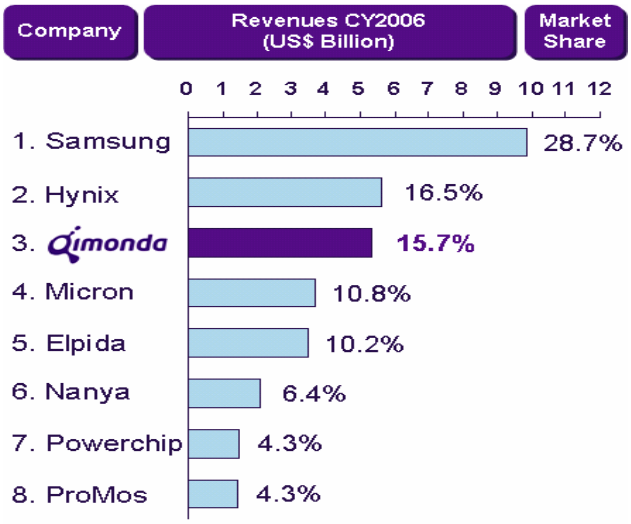


Figure 3 -Memories Industry Market Shares

Today, Qimonda has five production sites around the world: Suzhou-China, Malacca-Malaysia, Dresden-Germany, Porto-Portugal, and Richmond-USA. The company also works with the join ventures Inotera and Winbond, both in Taiwan.

Figure 4 - Qimonda's Sites

Qimonda's production can be divided into two major production parts:

- Front-end – production of wafers
- Back-end – production of components

Richmond is a Front-end site, Suzhou, Malacca, Porto are a Back-end site and Dresden is a Front-end and Back-end site. At this moment, a new Front-end in Singapore is commencing its construction.

In Front-ends wafers are being produced. Wafers are round disc shaped pieces of Silicon that contain the dies, the memory cells of the chip. These are produced either in 200mm or in 300mm of diameter. The number of dies in a wafer can be different depending on the kind of product and the size of the wafer. A wafer can have from 230 dies for a 200mm Wafer until 1700 chips for a 300mm Wafer.

Figure 5 - Operator with a Wafer

As previously mentioned, Qimonda Portugal is a back-end site, and so it receives the wafers from the front-end sites, and then produces the chips. These chips are sent to a Qimonda Distribution Center and will wait there for a customer. That customer could be internal (other Qimonda site) or external. [2]

All the Qimonda sites, productive or not, have the same values:

- Creative – *"We reconfigure the present. We challenge existing rules to enable the future"*
- Passionate – "*We work hard to make them happen*"
- Fast – "*We anticipate opportunities. We execute efficiently*"

1.3 Qimonda Portugal

Nowadays, Qimonda Portugal is the biggest back-end site of Qimonda AG. It is the major national exporter and the second biggest foreign investment in Portugal, with almost 700 millions of euros.

Figure 6: Qimonda Portugal

There are two different types of components distinguished by the package type: the TSOP (Thin Small Outline Package) and the TFBGA (Thin Fine Pitch Ball Grid Array), sometimes also referred to as BOC (Board on Chip). Both expressions, BOC e TFBGA, are used for the same type of components.

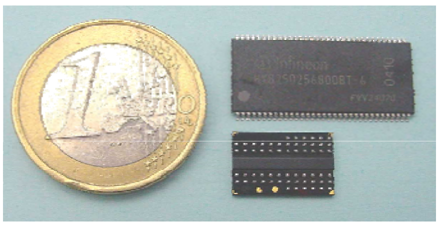


Figure 7 - TSOP and BOC

The FBGA is more recent than the TSOP and it is expected to replace the latter in a few years. Nowadays FBGA is almost 80% of the production volume at Qimonda and TSOP only 20%.

FBGA has many advantages when compared to TSOP: it is smaller; it has the same capacity and lower energy consumption; and it has even better performance. Although the FBGA and the TSOP have the same basic functions, some processes and raw-materials are different.

The production area at Qimonda Portugal (Qimonda PT) is divided in three main areas:

- RDL – Re-Distribution-Layer
- Wafer Test
- Assembly Operation
- Test Operation

Typically the RDL area and the Wafer Test area are Front-end areas, in which wafers are being processed. Though, these areas imply a good return to Qimonda PT they are not the main business and because of that will not be so relevant for this report. Thus, and only for introduction, the RDL has the objective to shape the wafers for a specific kind of components used in multi-chip packages and the Wafer Test is where the wafers are tested.

The components areas where Assembly Operation and Test Operation belong have a specific flow that will be explained in the picture below.

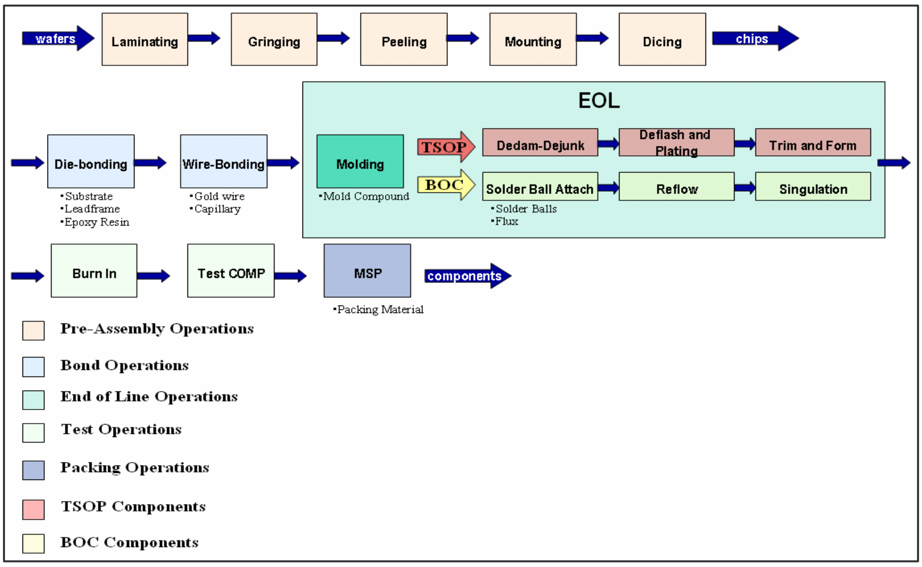


Figure 8 - Component's Flow

As can be seen in the picture, the main areas of the components are Pre-Assembly Operations, Bond Operation that now use to be called Front of Line Operation, the End of Line Operation , Burn in Operation, Test Operation, and Mark Scan and Pack Operation.

The Pre-Assembly: the purpose is to grind the wafer and to cut the individual chips into the final chip dimension. The picture below is a wafer after the Pre-Assembly Operation. In this operation the wafer is cut, although it stays with the same form because the cut wafer is still attached to a carrier and a sticky foil to prevent the chips from falling down.

Figure 9 - Wafer in Pre-Assembly

In the Front of line there are two main operations the Die-Bonding and the Wire-Bonding.

In the Die Bond operation happens the real individualization of the chip, and this last one is attached into the substrate electrical carrier.

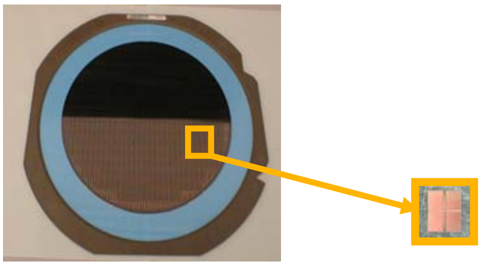


Figure 10 - Wafer in the Die Bonding

In the Wire Bond the electrical connection between the chip and substrate is performed. This connection is done by the gold wire.

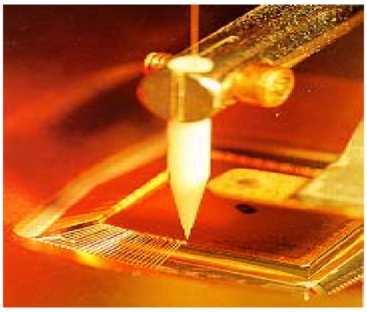


Figure 11 - Wire Bond Operation

In the End of Line the flow is completely different for TSOP and FBGA. The first operation, Molding, is equal to both packages. The Mold Operation has the main objective to encapsulate the chips with a mold compound that will protect the chips against the contact with the environment. The following operations for TSOP will configure the last layout of the chip with the "legs" and in the FBGA will give them the balls.

Figure 12 – TSOP

Figure 13 – BOC

In the Test Operation, as the name explains, is where the chip is being tested. There are two main operations, the Burn In operation and the Test operation. Burn In operation consist of putting the chips enter the oven where they will be exposed to electric stress under high temperatures with the objective to simulate child mortality. The chips suffer of child mortality and their life is represented in the graphic below. The Burn In test has the objective to age the chip and consequently decrease the number of the futures fails. In the Test Operation the chips will be exposed to more test and the main objective of this test is analyze the speed of each chip.

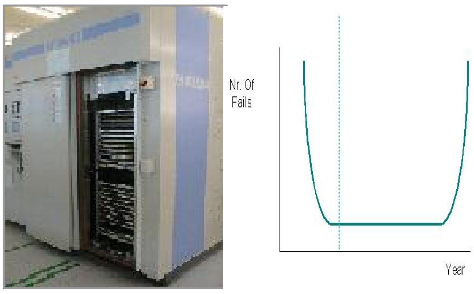


Figure 14 - Burn-in oven

Figure 15 – Tester

The last production operation is the MSP – Mark Scan and Pack. In this MSP operation, as the name explains, the chip is marked, scanned for marking errors and dimensions, and is finally packed. The chips can be packed in reels or trays. Each reel has 1500 or 2000 chips depending on the specification and if packed in trays will have 1500 chips.

Figure 16 – A reel after the final package

All the production area of Qimonda PT (15500m^2) is in clean room conditions. Because of this, it is necessary to implement a lot of security rules. The clean rooms are distinguished by the number of particles that they admit for superior limit. This number is K (1000) particles smaller than 0.5µm per cubic foot. The clean room class is 1K in RDL, the Wafer

Test area, and all Assembly Operation, except MSP which has a 10k clean area class, and the Test Operation and MSP are 100K clean room class.

Figure 17 - Clean Room

Qimonda Portugal is the premium Back-end of the Qimonda AG. The chips for 50% of the consoles memories are produced in the Qimonda PT. Equipment such as Playstation 3 from Sony, X-box from Microsoft and Wii from Nintendo are produced with Qimonda PT memories.

In the picture below are represented some of the equipment and consequently brands that prefer Qimonda chips.

Figure 18 - List of products

1.4 Structure of the report

The report is organized in such a way that the reader can find an explanation of the most important points of the internship in the main body of the document.

In the first chapter, the scope and the objective of the internship are presented as well as a description of the organization.

In the second chapter the importance of a data base and the main reasons that cause this project will be explained.

In the third the main problems that the author faced when he was developing the project will be demonstrated.

In the fourth chapter some examples that can be done with this model will be presented. The fifth the main functionalities that the system provides are explained.

The last chapter the author will do a conclusion of the project.

2 Project Scope

It is a concern that a data model is indispensable for all organizations. Nowadays, from the simplest organization until the biggest company, all the organizations have data bases. So date bases are important issues and in this chapter will be explained the motivations to specify one.

2.1 The Objective - Solution Transition

Qimonda AG has many data bases about different subjects. Therefore, for capacity planning only has one, the dCp – Dynamic Capacity Planner.

The Dynamic Capacity Planner, the dCp, is a Qimonda own tool for capacity planning. dCp was created in the 90's for the capacity planning of the Front-ends sites, years later even in the 90's dCp was adapted to Back-end sites.

There are some reasons that can explain why the dCp adaptation did not happen as well as it was expected, however that reasons are not the objective of this project and because of that will not be mentioned in this report. After a long period (years) of trying to stabilize the dCp it was suggested to specify a new data model for capacity planning. Someone that pretends to specify a data model like this needs a global knowledge about Qimonda. Once that a new data model will be specified since the begging, it was also suggested to specify in the same data model two more business processes – Investment Planning and Shop Floor Control. The specification of this new data model is the objective of this project.

2.2 Model Types

Once presented the main topics of the CRP Data Model, the different type of models that exist will be described in this chapter, the models are: Conceptual, Logical or Physical. Not all are important for this project therefore the main characteristics of them will be demonstrated in the table below.

Type of Model	Model Name	Perspective	Model Description	Entity	Type of Relation
Conceptual	Business Model	Business Owner	Semantic Model	Business Entity	Business
Logical	System Model	Designer	Logical Data Model	Data Entity	Data
Physical	Technology Model	Data/ Database Administrator	Data Design	Table/ Segments	Key/ Pointer

Table 1 - Type of Models

In the Conceptual Model, usually known as Business Model, the designer of the data is interested in the conceptual view. This model is typically used to explore domain concepts and to identify underlying business rules, with project stakeholders.

In the Logical Model, or System Model, the designer of the data is interested in the logical view. This model is used to further explore the domain concepts, and their relationship.

The database administrator is typically more concerned with the physical model and the physical implementation of a relational database. [25]

The present project pretends to be done in a conceptual and logical view. The physical model will be done in a next project.

2.3 Business Processes

As mentioned before, the main objective of this project is to create a solution for improvement of maintenance and management of resource information and to answer the necessity of three business processes:

- Capacity planning
- Investment planning
- Shop floor control

These three main items are the center of the report. All the problems faced and all the functionalities found were based on these topics. Because each business process is very complex a brief presentation about each topic will be presented.

Capacity Planning

Capacity planning is the process of determining the production capacity needed by an organization to meet changing demands for its products. In the context of capacity planning, "capacity" is the maximum amount of work that an organization is capable of completing in a given period of time.

A discrepancy between the capacity of an organization and the demands of its customers results in an inefficiency, either in under-utilized resources or unfulfilled customer demands. The goal of capacity planning is to minimize this discrepancy.

The Planners work is not always easy, if the number of products increase the complexity of the planning will increase exponentially. The Capacity Planner, the person that plans capacity, has to analyze the demand, which means the company resources, and to optimize the way to produce the demand. If the demand is much higher than the capacity, there is some solutions that might be taken such as: not accept the demand, to improve production times, or to do an investment planning.

All the industries need capacity planning, though the ones that are faster need more. Qimonda belongs to the semiconductors industry which is a fast industry. For Qimonda AG capacity planning is indispensable. All the sites, productive or not, have capacity planners. Qimonda Portugal is not an exception, with almost 300 hundred of different products and hundreds of different resources the complexity of capacity planning is high. In Qimonda PT there are three capacity planners; one for Wafer Test and RDL areas, other for Assembly Operation area and the last one for the Test Operation area. Though, they plan their areas individually, indeed the objective is the same thus they are always coordinated to achieve the output proposed.

There are many problems related to the capacity planning; the ones that are more important for Qimonda will be described in the next chapter such as resource allocation and resource dedication. This problem will help the reader to understand the complexity that this task involves and the coordination that it demands.

Investment Planning

Investment planning is the process of determining the investment needed by an organization to meet changing and mid to long-term demands for its products. Once the investment planning is done, the company has information about the amount that it might invest. Then it is necessary to analyze and understand if it is better to invest or not. This decision never is easy and always incurs a risk. Some questions such as if it is better to invest or not or until which percentages of risk the company might invest, are often made. Therefore, they are not objective and they can have different interpretations. Because of all these reasons the investment planning can not be considered only a financial activity. It is indispensable to analyze the investment needed with a concern of all the important areas of the company.

Qimonda AG investment planning has a complex flow. First in the Headquarter, at Munich, the demand forecast is analyzed with the departments of Marketing, Sales and Financing and they do a forecast of the amount that might be demanded of each product. The second step is to distribute the products and the volumes to the production sites. The production sites receive their plan with the volumes and the products that they might produce and when. The production sites analyze the plan, usually the ones that analyze are the Capacity Planners since they are the ones that know the capacities, and then they give their feedback to the headquarter in terms of how much they need to invest in that site (in equipment and headcount). If Munich approves the amount that the site proposes they start the investment, if not they will re-analyze demand and investment until finding the optimum for both.

In Qimonda PT where some equipment can cost until 5 million of euros, investment planning must be very controlled and everything is completely analyzed.

Shop Floor Control

While the other two business processes are more related with the production planning the last one, Shop Floor Control, is related with the executing the production.

The Shop Floor Control is a process of line organization that has many different activities. This data base should be flexible enough to aggregate information of different processes of shop floor control. Therefore, it will be only chosen for this explanation two shop floor control processes: dispatching and scheduling. Some other problems related with this business process will be mentioned in the next chapter.

Dispatching

Dispatching is the act to allocate a lot to a machine. Each lot has a different priority and each machine has its capabilities, thus the processes that do the relation between these parameters are the dispatching. The figure below explains the situation.

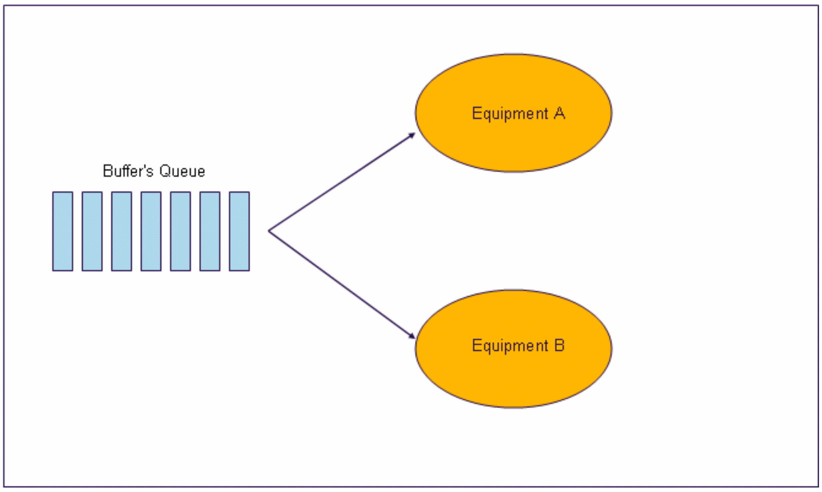


Figure 19 - Dispatching

For the reader to better understand the concept of dispatching an example is provided. In the productive line before an operation there is a queue with different kind of products and different priorities. In the first step of production it is analyzed which lot goes to which machine. The decision depends on equipment capabilities and the specifications of the product. The equipment capabilities that might be considered are for example equipment efficiency, the kinds of product that the equipment processes, the equipment reliability, the equipment's speed and the equipment's capacity.

Scheduling

The Scheduling as the Dispatching are related with the equipment allocation, but in scheduling, the buffer queue is before the equipment and in dispatching is before the operation. In the picture below there is an example showing where the buffer queue is in the scheduling process.

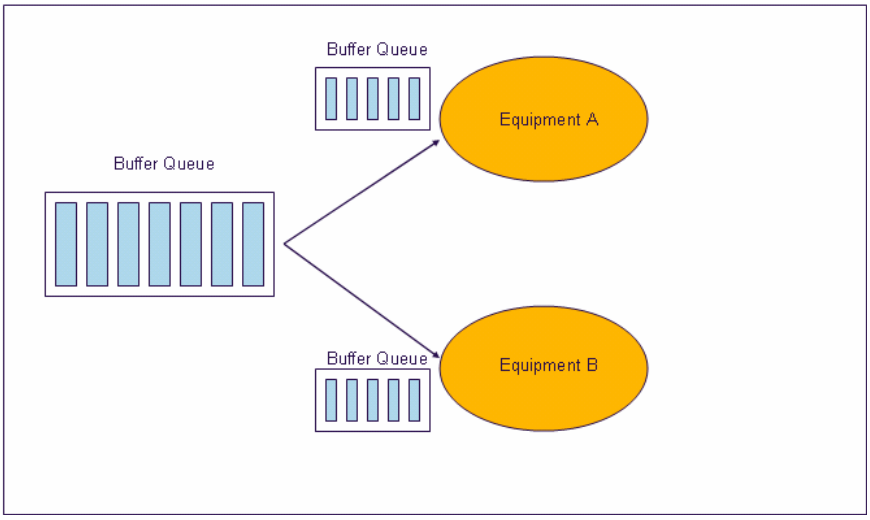


Figure 20 - Scheduling

The main objective of the scheduling is to create a timetable for each machine. With this timetable it is possible to organize much better the production and to know what time each lot must be processed and when the equipment might be in maintenance or setups. This is of great value for global optimization of the production process and minimization of production mistakes.

3 Problem Description

Until this chapter the main objective was to introduce to the reader the environment where this project is incorporated. The company, the industry, and the business processes were introduced. Now it is required to explain the main problems that were considered during the modeling phase of this project. Not only the problems will be explained but also the solutions and the considerations that were taken will be presented.

3.1 Manufacturing Hierarchy

Qimonda has a Manufacturing Hierarchy. Manufacturing Hierarchy is a sorted list of manufacturing areas, where the items are sorted since the highest until the smallest. This list not always is obvious and sometimes some concepts are mixed. Thus, the first problem description is the Manufacturing Hierarchy.

Manufacturing Hierarchy is composed by:

- Location – the location is the biggest level of the Manufacturing Hierarchy. A Location means a physical construction which can be a production site or not. QIMONDA_PT is a Location.
- Manufacturing Level (ML) – A Location has diverse Manufacturing Levels. ML is the first way as line production is divides. Here in Portugal there are five ML – WTEST, RDL, CPREASSY, CASSY and CTEST.
- Area – A ML has diverse Areas. For example CTEST has three Areas – Burn-in, Test and Mark Scan and Pack (MSP).
- Work Center (WC) - A Work Center is a cluster of different Resource Group with the same functionalities. An Area has a group of WC.
- Resource Group (RG) – It is a group of resources with equal or comparable attributes or behaviors.
- Resource (R) – is an equipment, a machine, a tooling, or a fixture.
- Operation (OP) – A Work Center usually has only one operation, nevertheless sometimes has more than one. In Burn-in the operation can be load or unload.

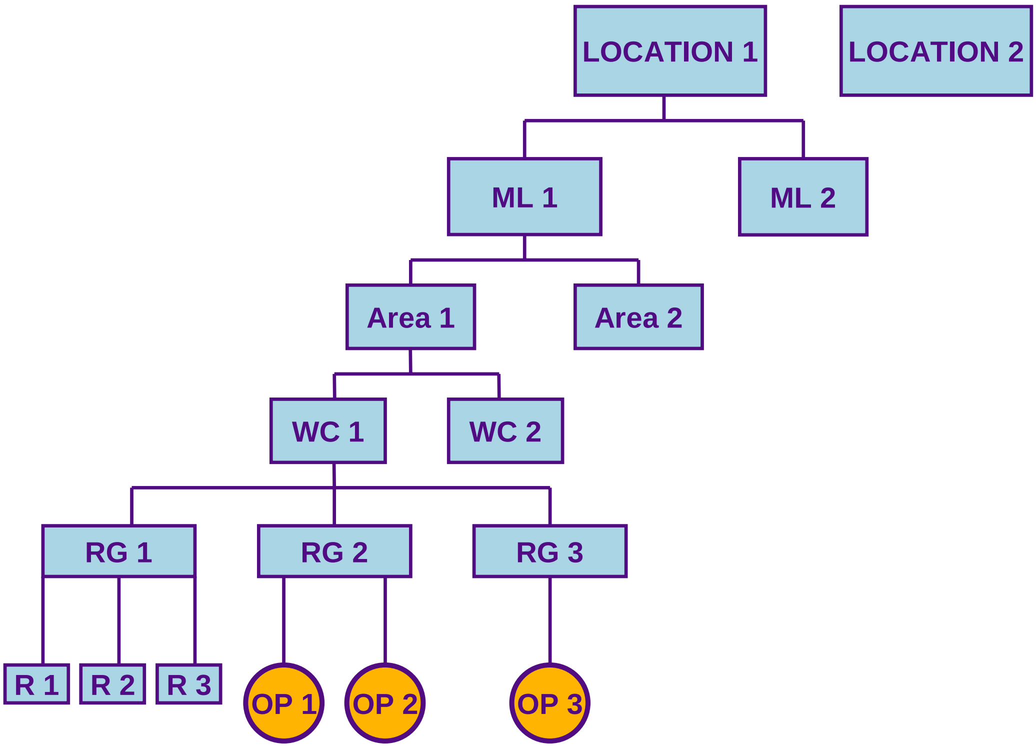


Figure 21- Manufacturing Level Hierarchy

In the current project the concepts Manufacturing Level, Work Center and Area will be represented in the same class – Work Center Class.

Though a first description was done about the Manufacturing Area some other concepts are associated with them. In the Process Chain Area (PCA) there are some names that might be recognized. Each Area has a Process Chain Area (PCA). A PCA is constituted of Prod, IN-Buffer, OUT-Buffer and TRANSIT.

- PROD – in the area where the materials are in production.
- IN–Buffer – is where the material waits for undergoing a production transaction in the subsequent manufacturing level or a receiving stock for shipments from an off-site location.
- OUT-Buffer – is where material is going to be shipped to an off-site location or distribution center.
- TRANSIT – is when the material is in transit between two locations.

In this report all the manufacturing level will be considered as PROD and therefore the PCA will not be modeled explicitly.

3.2 Resource allocation vs. Resource dedication

Though the expressions "resource allocation" and "resource dedication" are frequently associated, they have different meanings.

Concerning production planning, a resource-allocation decision is a plan for using available resources. The plan has two parts: the basic allocation decision and the contingency mechanisms. The basic allocation decision is the selection of resources which are allocated to some products and not to others. [16]

The resource dedication, which is also known as capability, is used whenever there's a technical specification. An example to help understand this situation: a product can only be produced in a specific model of equipment (e.g. dual die products). Resource dedication can also be considered as a modeling approach when a customer specifies that the product must be produced in a specific model of equipment. Often customers have many specifications that are included in the resource dedication.

A final example that compares all the concepts: there are 4 machines (availability) but only 3 can produce product A (capability). Based on the planned demand only two machines are "reserved" to produce the product A (allocation). But due to the volume of the planned demand, actually only one machine is used (utilization).

3.3 Resource Priorities

Machines usually have more than one tool group associated to make the production more flexible. With this potentiality, a specific equipment can produce a range of different products only changing the tool.

Not only the machines have more than one tool- but also some operations have more than one resource group associated, so in one operation the same product can be done in more than one equipment group and each equipment group can use more than one tool group. This flexibility is very positive for production although sometimes the information is not as organized as desired and sometimes it is not obvious which resource should be chosen. This situation can be improved with a relation between equipment and tools.

In today's dCp based capacity model, this relation is done by the creation of a combi machine group. A combi machine group is a relation between a machine and a compatible tool. A combi relation defines a simultaneous utilization, an AND relation. For example: when it is said that a Combi_1 will be used this means that it will be used simultaneously the resource 1 AND the tooling 1. Although a combi is a great way to group a resource and a tool it is not enough to relate priorities, this means that with only a combi the user does not know what combi might choose first.. As the priority relation is important, Parent Machine Group was created to define an alternative usage, an OR relation. The Parent Machine Group defines which combi will be used. For example: when there are both the Combi_1 and the Combi_2 and both can be used, this is shown as an alternative.

With the objective of satisfying this necessity (alternative combinations) it was created a Resource Priority Class that relates the machine, the tool and priority. This can be done if a sorted number is associated to each resource and tool. The machines and tools with the same number belong to the same group.

3.4 Distinction of main resources and tooling

As previously mentioned, production needs equipments and tools. Each equipment group can have more than one tool group. And a tool group can be in more than one equipment group.

Nevertheless both, equipment and tools are resources and can be included in the resource group. Though this is a good simplification, it is necessary to create an attribute in resource group to distinguish them. Thus, it is necessary to create the attribute Resource Type where the machine has the definition of main resource, and the tool the definition of “tooling”. This distinction is essential for capacity planning since without this the Capacity Planners could not distinguish them.

3.5 Alternative Scenario

As mentioned before, Qimonda PT receives from the Head Quarters a production plan that defines what the site should produce in the next weeks. Although, the Semiconductors Industry is very fast and sometimes needs to change the initial plan, every time it is necessary to change the plan the Capacity Planners receive an e-mail asking if it is possible to produce a specific quantity of a product in a specific date and if for this demand to be possible is necessary abdicate to produce any other product. This analysis is vast and in this fast industry it is very usual.

Every time that Capacity Planners want to simulate the capacity with the new instructions they need to create a new file and specify it. It is pretended that this model will be able to do an alternative scenario without a big effort.

Alternative scenario, in this model, will be the opportunity for the Capacity Planners to insert in the model the new data that they received and the model returns the capacity that this new specification will spend. Although the example of the new volumes of products will probably be the most usual scenario, the model will be flexible enough to analyze with other data, such as with a different number of resources, or different process times.

Certainly this solution will optimize the Capacity Planners work and will be a great improvement in the company.

3.6 Yield Modeling

There are many parameters that define the operation and the products; one of them is the Yield. Yield is the percentage of good units of a product that go out of an operation. Each product has a different yield and that might change.

The Yield can be analyzed in different perspectives, such as the yield of a wafer, the yield of a lot or the yield of an operation.

In this project the yield will be specified in the operation level. At this level of specification it is possible to obtain detailed information about the production. If the user needs information in other levels such as Work Center or Area a business logic can be applied to aggregate the information.

3.7 Good Chips per Wafer

Other important parameter is the Good Chips per Wafer. Good chips per wafer are the number of dies in a wafer that are able to be produced into components. This number characterizes the product.

In the model is also useful not only to keep the information about the good chips per wafer but also to create a business rule that would be able to calculate it.

The good chips per wafer need to be calculated based on the yield. In this way the important yield for this functionality is the wafer yield. Thus, the number of good chips per wafer is the product between the chips per wafer and the wafer yield. This formula is presented below.

$$GCPW = CPW \bullet (Yield)^{WAFERTEST}$$

GCPW → Good Chips per Wafer

CPW → Chips per Wafer

As the number of good chips per wafer is directly related with the Yield and the Yield changes along the time, this means that the number of good chips per wafer also changes along the time.

3.8 Demand and Location

The Demand can be understood with different meanings: as the market demand or the demand that the headquarter sends to the production sites. In the present project the second interpretation will be used. This means, the demand is the demand for one product in one

Location. A Location can have more than one demand depending on the number of products that the site produces.

The Demand depends on the period of time (Date) as a result of different dates a Location will have different demands. This is easy to understand because the demand of a specific product in a specific week will probably be different in the next.

A production plan is the aggregation of different demands for the same period of time. Each Location has its plan.

3.9 Product

Certainly everyone can give an example of a product. The concept of product is linked to everyone since they were a child. Therefore, product is much difficult to be defined. There are many definitions about product, for this report was chosen one from Kolter [15], "*a product is anything that can be offered to a market that might satisfy a want or a need*".

This definition helps the reader to understand the meaning of product but is not enough to define the Qimonda products. It is not so simple to define what Qimonda considers product. Because of this some question to help define products were posed.

1. Is a product different if it comes from a different Frontend?
2. Is a product different if its package is different?
3. Is a product different if it has a different packing?
4. Is a product different if the final customer is different?
5. Is a product different if the Marking is different?
6. Is a product different if its quality is different?
7. Is a product different if its supply chain is different?

The answer of the first question - *Is a product different if it comes from a different Frontend?*- is NO!. The place where the wafer is produced does not influence the final product, and components from wafers from different places can be together in the same lot.

The answer of the second question - *Is a product different if its package is different?* - is YES!. If a product is a TSOP or a BOC is completely different. The package gives different characteristics to the product and the TSOP flow and BOC flow are different so it implies to become different products.

The third question - *Is a product different if it has a different packing?* - is a tricky one. It is not so clearly as the previous ones but the answer is NO!. If a product is packed in reels or in trays this does not make a product different. For example: If someone wants to buy a Shampoo, he can buy it with small size or with big size. The size of the product's packing does not matter. In both sizes the product is the same. So the product will not change with the packing. In the case of Qimonda, if a product is packed in reels or in trays the product is the same.

The fourth question is also a tricky one - *Is a product different if the final customer is different?* - this question has different answers – NO/YES. If the question was understood as there are two customers that want to buy the same product – the product will not change with the customer. For example, if two different persons go to the supermarket and buy a shampoo, the shampoo will not be different because the customers are different. But, there is other point of view, if it is considered that the customer does product specifications. For example, two different persons want to buy the same car model but one chooses some extras that the other one does not choose, in this way the products are different because they are satisfying different needs (Kotler definition). Thus, when a customer does product specifications the final product will be different if not the product will be the same. With the product specification the answer of the fifth question is started - *Is a product different if the Marking is different?* – the answer is YES!. A customer marking is a product specification so as it was demonstrated product specifications makes products different.

The sixth question - *Is a product different if its quality is different?*- the answer is YES!. If two products have different qualities they are different. For a customer it makes a difference to buy a product with good quality or medium quality.

The seventh question is directly related with the question number one. So the answer is NO!. It does not matter where the product come from if when it arrives is the same.

3.10 Product Group / Product

In this chapter it is pretended to present two new concepts. They are the product group and the product. As the name suggest the product group is a group of products of the same type. The difficulty in this distinction is to know which one concept might be used in each situation.

In a general way the product group is in the aggregation that interests the customer, and the product is a version with more detail about the production. An example will be used to help the reader understand this distinction. A customer goes to the supermarket to buy two bottles of water with the same size and from the same brand. For the customer these two bottles are equal because satisfy the customer in the same way. Although, the bottles were produced at different dates and for the company in the line production has to be possible to distinguish one of the other. This latter concept is the equivalent of the product concept at Qimonda. The product is the name that identifies a specific product in the line production. This concept is completely essential for the production controllers that are the ones that need the information from the Shop Floor Control Processes.

Nowadays the product is characterized by a number, this number is such a code and each unit has a meaning associated. This code is known as Baunumber. A product group has many products (baunumbers) associated.

This problematic will be mentioned in other places in this report. The relation between the class of the Product Group and the class of the Product will require a business rule. This business rule has the objective to aggregate products from the Product Class with the same characteristics in the same Product Group Class. In this way the model will be completely flexible and dynamic. At same time the information can be used in the two levels of aggregation that is a great facility. This problem will be described in the next chapter.

3.11 Product Aggregation Problem

The definition of a Product Group is a conjunction of attributes of a Product.

As the definition said the Product Group and the Product are connected by Attributes. So the Product Attribute class has relation with the class Product Group and the class Product.

Other problem that was defined is what the attributes are that define the Class Attribute. In a first view can be said the name of the attribute and the value. For example

CLASS PRODUCT_ATTRIBUTE	
NAME	VALUE
GENERATION	512 M

Table 2 – Product Attribute Class - Name and Value

This description is enough if one generation only has one value. But this is not the reality.

CLASS ATTRIBUTE	
NAME	VALUE
GENERATION	256M
GENERATION	512 M

Table 3 -Product Attribute Class with more than one value

So, for the Attribute Class to be defined two attributes are needed, because it is possible to have more than one value for the same name. From an implementation point of view this question can be solved with the help of a new attribute (a code) that distinguishes each entry. But the importance in this report is to secure that both attributes will be always used to define the Product Attribute Class. In the example below will be described two different products:

PRODUCT – 7931	
GENERATION	512M
SHRINK	D11
PACKAGE	FBGA 84

Table 4 -First Product

PRODUCT – 7932	
GENERATION	512M
SHRINK	D11
PACKAGE	FBGA 60

Table 5 -Definition of two different products

These two different products belong to the same Product Group if only the Generation and the Shrink are considered.

PRODUCT _GROUP - 512M D11
PRODUCT – 7931
PRODUCT – 7932

Table 6 - Product Group 512M D11

At the same time that the products are aggregated into the same product group, more information of the product is possible to be used in others tasks.

The Attribute Class will improve all the model's users. For sure this is one of the most important classes and one of the classes that will keep more information.

The Attribute Class is a great improvement. The Attribute Class will save so much time once that will be simple adding new attributes without changing anything else. This class will enhance the flexibility of the model.

3.12 Product Relation

There are many issues related with the product. The product relation is one more. Although the name of the product in the Product Group Class is the same since the product enters in the factory until it is shipped, for the Product Class this is not true.

Indeed every time that a product enters in a new Work Center will suffer a transformation that can be physical or not. Every time that the product suffers a transformation, it will acquire new characteristics. If the characteristics of the product are different the product needs to be different too. This problematic is complex, and will be also referenced in the chapter five. In the present chapter it is important that the reader understands that every time a product enters a new Work Center it will be transformed into another product or products depending on the Work Center.

As mentioned before, the name of the product is a code with lot of information and if the product acquires new information, it needs to change the name for the new one that aggregates the new information. This sequence of names that the product suffers since its entry into the factory until it will be shipped needs to be kept. In the model this information will be kept in a new class.

At this moment in Qimonda there is not a class that keeps this information as in the proposed data model. This is a great improvement and will be very useful. This class will be used for the business process shop floor control. The class is the Product Relation Class.

3.13 Equivalent Products

Qimonda PT produces hundreds of different products with different process time complexity between them. Sometimes at the same time in the same resources two different products have completely different outputs because one is much more complex than the other.

It is not advisable to compare products with different complexities, thus a reference product was considered. A reference product is a product that will be the reference for the others. The importance of this parameter is higher, consequently is the Qimonda Heads Quarter that decides the reference product.

Equivalent Product is very useful to compare different productive units that produce different products. With reference product it is possible to know what each production site produces in equivalents (reference product) and compare to the others.

Nowadays the reference product is “512M T90 PG-FBGA-60 x8”. This product has complexity 1.0 (one). All the other products have related complexities with this product. For example: if there is a Product X with a complexity of 2 (two), it will be at the same (in this analysis) to produce one unit of Product X or two units of the reference product.

This product is dynamic and can be changed if there is necessity. Thus, a new class to support this information will be created; the class - Reference Product Class. With this class being also time dependant, it is possible to keep the information about the historical of the reference.

Reference Product class is not enough, a class to agglomerate all the product complexities is also necessary – Product Complexity Class. As the Reference Product, in the Product Complexity is common to change the complexity of a product. For example when a new product is introduced in the production line its complexity is higher than when it is standard.

If one product is more complex than other product, it means that one product will spend more time to be produced than other. Because of this the maximum capacity of one resource must be calculated in the reference product unit.

Once the user knows the reference capacity of a product, it is possible to transform the reference capacity in the physical capacity. For this to be possible, it is only required to divide the reference capacity of the product by its complexity. This calculation is done per

Resource Group. The maximum capacity that one resource of one resource group can produce in equivalent units is stored in a new class – Maximum Capacity of Reference Class.

Another concept that might be considered is the Maximum Capacity to Produce. This kind of capacity is different than the latter one because this one considers the availability parameter as for example standby and downtime. This Capacity is always equal or minor than the Maximum Capacity of Reference. This concept also needs to be kept.

The capacity that is needed to satisfy the demand must be calculated in equivalents. This information is a result calculated by a business logic functionality and maintained in the class – Local Starts Result.

The spare capacity will be a functionality and the result of this will be present in a class. This class can be the same of the Maximum Capacity to Produce because both only depend on the resource group and the time. That class will be the Resource Group Capacity Parameters.

3.14 Resource Exceptions Modeling

A product in an operation can be processed in more than one Resource Group. This is a problem of resource allocation. But there are more considerations that should be taken. In a resource group not all the resources can process all the products that are consider for the resource group. In this scenario, it is not a problem of resource allocation but a problem of Exceptions. In the Exceptions model there are the permissions and the exclusions.

To better illustrate this problem an example will be used. There are three products (Product A, Product B and Product C) and all of them can be processed in the Resource Group A. But the Resource Group A has the Resource 1 and the Resource 2. If an exclusion is considered so all the products can use both resources unless the Product A that can not use the Resource 2. On the other hand other situation can be considered. It happens when all the products can use the Resource 1 but only the Product B can use the Resource 2, in this way this is a permission.

For this new concept be specified in the model is needed a new class - Exceptions Class. This class will be linked with product, operation and resource. And the class returns if this relation between the resource the operation and the product is allowed, disallowed or for engineering only.

3.15 Operation Sequence

With the complexity of the line production and the diversity of the products, it was necessary to standardize the sequence of production. This standard sequence is known as enterprise route.

An enterprise route is the sequence that the product needs to do to be produced; each product has an enterprise route. An enterprise route has many routes and one route has many operations. This description is simple but the complexity become higher when a product can have many enterprise routes and an enterprise route can have many routes and one route can have many operations.

In this chapter the relation between the operations and the routes is analyzed. An operation can be in more than one route in order to identify the sequence of one operation in one route an attribute is not enough. For example; the operation X can be the first operation of product A but can be the second of the product B because the product B needs an initial inspection, so an additional attribute that translates the number of the sequence is not a hypothesis.

The best way to qualify the order in the sequence is to create a new class that adds a new entry for each combination of product and operation. Two different products can have the same order in one operation but they will have different entries, for example the product A and the product B have for the first operation the operation X but this does not mean that they will have the same attribute, the attribute will be different but the value of the attribute will be the same. In this way the model becomes more dynamic. For example in the product A, which has the operation X as its first operation, was identified a qualify problem and that implied a new flow in which the operation X will be not the first but the third. Here the user only needs to change the attribute of this specific relation and any other product from the same product group will be interfered. The class that keeps this information is the Operation Sequence Class.

3.16 Route Sequence

The main objectives of the Route Sequence are the same as of the Operation Sequence and both sequences are related. Route Sequence pretends to identify the order of one route in an enterprise route. As it happened with the Operation Sequence in the Route Sequence it is necessary to create a new class that keeps the information between the product, the enterprise route and the route.

In the Route Sequence it is not enough create a new class because it is possible to have equivalent routes to achieve the same results. This means that there are equivalent routes for the same enterprise route. Firstly, it is important to understand the meaning of equivalent route. An equivalent route is a route with the same objective and result that other but one is a little bit different of the other in particular aspects, such us its speed. With this description is possible to understand two important characteristics: one is that they have the same result so in the same enterprise route they have the same sequence number. But if they are equivalents but not equal it is important to distinguish them for the users can choose the one that they prefer. So, in the class of the route sequence it is necessary to create a new attribute to identify alternative routes. The attribute will be the Priority Number. The Priority Number has the purpose to distinguish the alternatives routes. In this way the principal route will have the attribute Standard and the other the Alternative.

Concluding, in the same enterprise route the ones that are first are the ones with the smaller numbers. In case of Routes with the same sequence number, the first to be chosen is the Standard one and the other is the Alternative. In this class each new combination between product, enterprise route, sequence number and priority number means a new entry. It is not necessary to create two primary keys because when the model will be implemented only is necessary to create a new attribute such as Route Code Number and each combination has a different number in this attribute.

3.17 Resource Attributes

In the chapters before the problems related with the products were mentioned. Indeed the product is one of the most delicate entities of the model. Not less important in this project are the resource objects.

In the present project the main two areas of the model are the product and the resources, and all the others classes are related directly or indirectly with these areas.

As was done in the product group, the relation between the resource and the resource group is done by an extra class to help this description – the Resource Attribute. Though the bases of the two classes are similar, they are completely different. The resource attribute class can change along the time; this means that the resource attribute class is connected with the date class.

This class will give much more flexibility of the model and will save much time in model maintenance. But the relation between the Resource Group and the Resource can not be indirect (Resource Attribute Class) but also in the directly way (both classes connected).

The classes related with resource will be mentioned frequently in the report because they have many functionalities associated.

3.18 Resource Conversions and Set up times.

Qimonda Portugal produces daily a range of different products. It is frequent that different products in the same day need to share common resources – so resources conversions are needed.

The resource conversions give to Qimonda an increase of flexibility but decrease the efficiency because they consume time, thus reduce available capacity. If on the one hand it is mandatory to have flexibility, and only this way it is possible to produce a range of different products, on the other hand if the conversions are too much the output will decrease. Consequently, the conversion time (Set Up Time) is a variable that needs to be optimized.

The Set Up Sequence Class keeps the information about the products that can be produced in a resource group. The problem in this class is if the relation between the products might be considered as one to many or many to many. Indeed, both approaches are correct, as it will be explained with the help of examples.

In the first approach (one-to-many relation), if the Resource Group A processes product 1 and finishes, the next product that can be produced belongs to a variety of products (the way that is chosen the next product is a functionality and that will be explained in other chapter). This alternative implies that the user needs to input all the options that a Resource Group can have and its Set Up Times. Thus, a huge manual work is needed.

In the second option (many-to-many relation), for example, products P1 and P2 can be converted to P3 or P4 both with a Set Up Time of 40 minutes. This is also correct but if it is pretended model a conversion from P1 and P2 to P1 or P3 it is not correct because from P1 to P1 there should not be a Set Up Time. This approach would imply that this model would require additional exceptions to model not required set up for the same product sequence, in this example from P1 to P1. The objective of the project is to simplify modeling, therefore we chose the first alternative (one-to-many relation). In this way is avoided the exceptions.

3.19 Recipe Model

One of the problems mentioned before was the set up time. If in one way it is very important to optimize the set up conversions, in order to make this possible it is crucial to optimize all the tasks related with the equipment conversions. A range of different tasks is being studied for this optimization such as SMED – Single Minute Exchange of a Die.

One of the tasks that must be done for the equipment conversion is the change of the recipe. The recipe is the program that the product will execute in the resource, for example in the Burn in Area the recipe in the ovens describes the hours that the product needs to be inside the oven and which are the electronic tests that the product might be submitted to.

This information needs to be kept by product, operation and date. In the model there is other class that is done by product; operation on a date which is the Process Time Class. In this way the information about the recipe will be stored in the Process Time Class.

3.20 Date Aggregation

Date Class certainly is the class with more relations with other classes. All the three business processes represented in the model need this class and they utilize its information. Because of the different necessities of each business processes the date class keeps the information in different formats. For example, the classes with information about the Shop Floor Control, such as the classes that keep information about scheduling needs to keep second information. On the other hand the Capacity planning only needs weekly information and the investment planning monthly or yearly information. As demonstrated, it is necessary to keep all the information in different formats and it is also necessary to keep information and conversions between the Business Year (September to September) and Calendar Year (January to January).

This class needs to be associated with a business rule to processes all this demands. This subject will be worked in the chapter five.

3.21 Investment Planning

Today, if one organization wants to survive competition it is necessary that it has a clear objective – a mission. Different organizations have different missions, some pretend to be national leaders, other world leaders, other are non-profit organizations. Qimonda as a profitable organization pretends to be the world leader in memories. This is an ambitious mission and requires a huge investment.

As mentioned before, investment planning is one of the crucial parts of this model. For the investment planning to be consistent it is essential to keep in an efficient way as much information as is possible.

The resources from the same resource group along the time have different prices. The price of each resource is an attribute that might be modeled. The resource price will be an attribute of the resource attribute class.

The investment done per Resource Group will be stored in a new class the Capacity Resource Group Class. This class will keep the money invested until now in the resources from a specific resource group. In the same way that is needed to keep the information about the Resource Group the information about the investment of a Work Center might be also kept.

The introductions of new products involve costs. These costs come from new tools for each specific product and sometimes from completely new equipment. It is interesting to keep the information about the investment necessary to introduce a new product. This information will be stored in the class – Capacity per Resource Group per Product.

With all this information the company can keep all the information with a format that is easy to understand.

The classes mentioned in this problem description will be explained in the chapter five.

3.22 Engineering Corridor

Qimonda Portugal not only produces salable products. Almost 10% of the production is Engineering Products. Engineering Products are products that are produced by engineers and these products are not for external sales.

There are different types of Engineering Products. Some of them have a similar behavior as the productive products, others not. Once that Engineering Products are mandatory and they need a special care, an Engineering Corridor was created where this kind of products might be produced.

There are different alternatives to model the Engineering Corridor. One of the options is to allocate some resources for the Engineering Corridor. Therefore the Engineering Products Volume are not constant, this means, that some weeks will have spare capacity and another weeks are overloaded. Every time there is spare capacity the line looses performance, because this capacity could be allocated to production. So that this model become more dynamic a resource attribute was created that transmits the percentage of each resource that is allocated to Engineering Products and to Productive Products. With this attribute spare capacity can be controlled. The outcome is represented as the ... class.

3.23 Actual and Forecast Parameters

In a company with this dimension forecasting is mandatory. Almost all parameters have forecast values. It would be perfect if forecast was always equal to reality, although this is mostly not true. Indeed there are a lot of other reasons that can not be planned that cause the gap between the forecast and the reality.

Forecast parameters are planned based on the actual values. In current and traditional planning systems, every time that the forecast is different from the actual value, only the planned value is kept in the system. This means that there is not a historical about all the actual values that truly happened. Thus, in this chapter we propose to create a new attribute where this information will be kept. With the help of the attribute will not be a huge effort to keep this information and it can be very important when the planners pretend to improve the way they are planning the values. This attribute will be provided in the Resource Attribute Class.

It is also pretended to create a functionality where the users (the capacity planners) can choose the forecasting values or the actual values. There are many different forecasting functions that can be useful with these different values; these functions will be explained in the chapter five.

4 Proposed Solution

After the main problems encountered in current planning processes, in this chapter the solutions will be presented. The main solution of this project is the proposed data model although there are other important points that are presented here because they are also solutions. The users of the solution are one example of this.

As the model that will be presented in the chapter 4.2 is very complex, in the main report only two main areas of the model will be discussed, these areas are the Product Group and the Resource Group.

4.1 The Users

Qimonda uses a vast number of planning systems such as dCp or SAP, but indeed a significant part of the data is still being calculated in Excel files. Without any doubt Excel is a wonderful tool especially because it is very friendly and with easy debugging. However, when a lot of different information is pretended to be included the excel files start to be not enough for the necessities. Because of this a Master Data solution is of importance. The Master Data as the name means is a huge data base that has the objective of replacing a series of different smaller data bases. One of this smaller data bases relevant to this project is the CRP Data Model.

The CRP Data Model will be the capacity part of the Master Data. As the reader knows this project pretends to answer the capacity planning, investment planning and shop floor control necessities. Thus the model will keep a lot of information, which is much diversified and it will be useful for a range of different solutions.

This is a particular case of this project, the users of the model are not persons but they are rather other solutions that use the underlying data for other purposes. This means that this model will support many other models that will integrate the output of the CRP in their inputs. Since the users are solutions it is pretended to keep the model flexible enough in order to enable integration of new systems every time that it is pretended and without influencing the others users.

At this moment the systems that will be connected with the CRP are represented in the picture below and they are: Capa Modeling Solution, Investment Planning Solution, Factory Planning, Supply Chain Planning Solution, Master Planning, Financial Forecasting, Scheduling Solution, Shop Floor Control Tools, Dispatching Solution and Shop Floor Simulation.

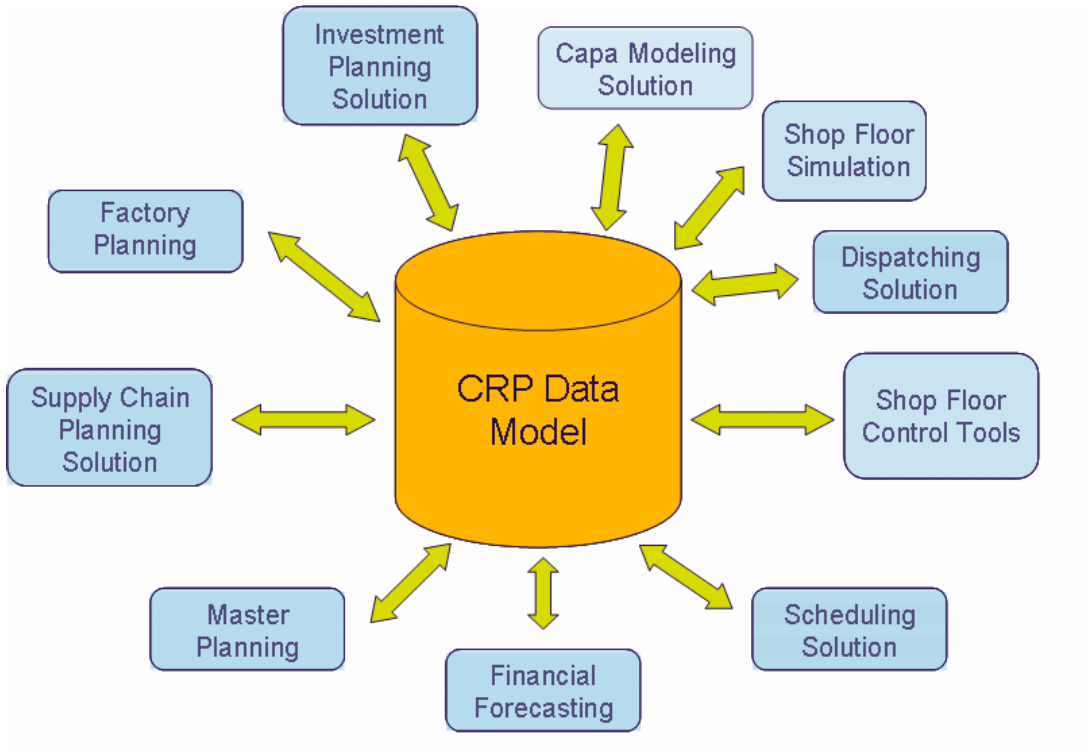


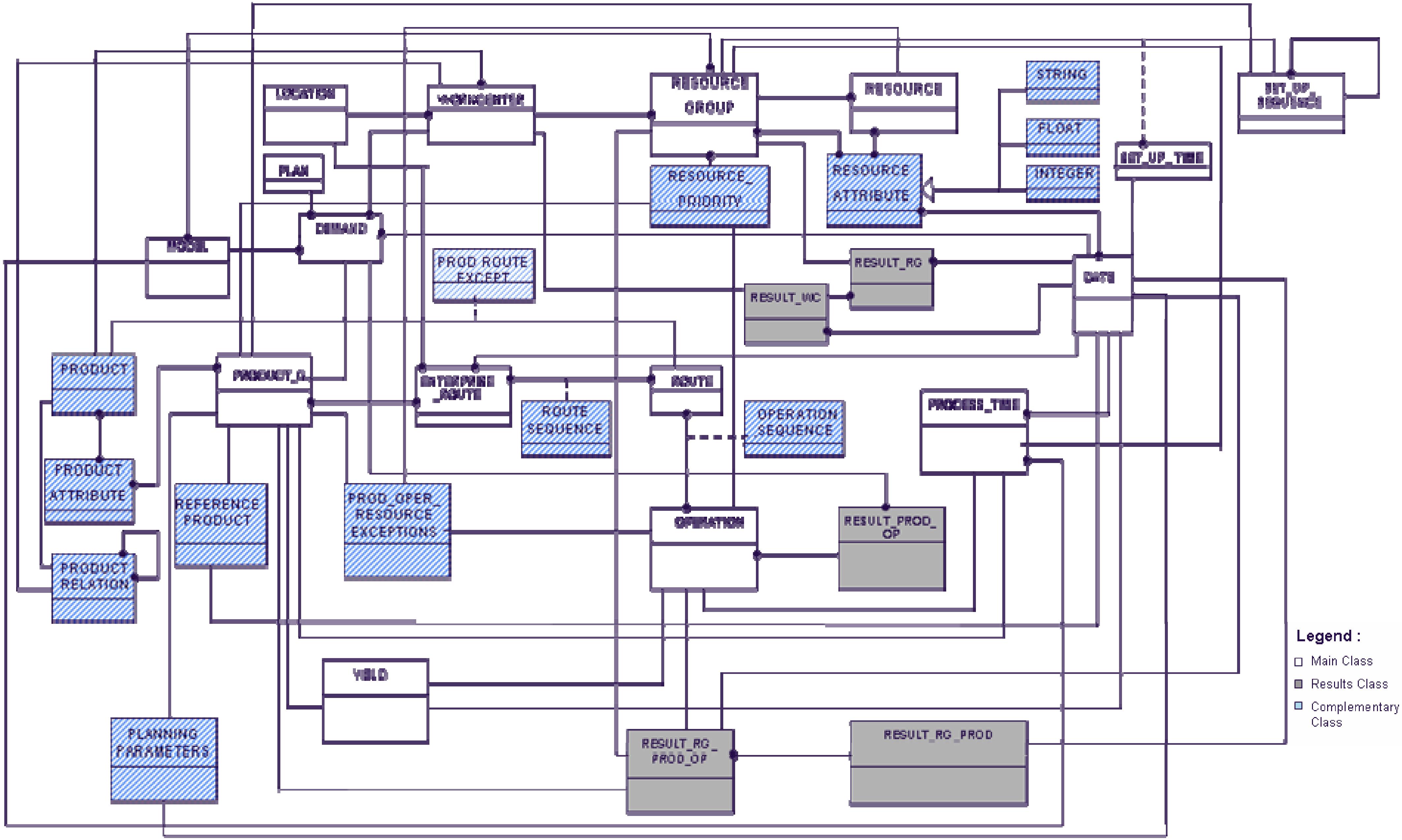
Figure 22 - User of the Model

In the chapter number five we reference some functionality that needs interfaces with the user. As was mentioned this project will not have connection with interface users, therefore, some of the functionalities that were developed need interfaces with the users. This kind of information not will be implemented in a first phase of the project, although it was decided that these ideas will stay reference in this project and can be use for next versions.

4.2 The Model

In this chapter, the model will be presented and its classes will be explained.

There are three different kinds of classes, the main classes, the result classes and the complementary classes. The main classes can be understood as the bases of the project, with these classes the model can be completely understood and the main functionalities can be kept. The complementary classes are the secondary classes, the objective of these classes is to keep information about a specific part of the model and complementary it with the information kept by the main classes. The result classes as the main can explain keeps the information about the results that the model produces.


LOCATION
WORKCENTER
PLAN
DEMAND
MODEL
RESOURCE GROUP
RESOURCE
STRING
FLOAT
INTEGER
SET_UP SEQUENCE
SET_UP_TIME
RESOURCE_ PRIORITY
RESOURCE ATTRIBUTE
PROD ROUTE EXCEPT
RESULT_RG
RESULT_WC
DATE
PRODUCT
PRODUCT_G
ENTERPRISE _ROUTE
ROUTE
ROUTE SEQUENCE
OPERATION SEQUENCE
PROCESS_TIME
PRODUCT ATTRIBUTE
REFERENCE PRODUCT
PROD_OPER_ RESOURCE EXCEPTIONS
OPERATION
RESULT_PROD_ OP
PRODUCT RELATION
YIELD
PLANNING PARAMETERS
RESULT_RG_ PROD_OP
RESULT_RG_PROD
Legend :
Main Class
Results Class
Complementary Class

Once that the model has more than thirty classes not all the classes can be explained in the main report. Two areas of the model were chosen to be presented in the main report; these areas are the Product Group and the Resource Group. The reason for choosing these areas is because they are very important for the model. The other classes are explained in the appendix A.

4.3 Product and Resources examples

For those who don't work in Qimonda the subject of this project sometimes can be ambiguous. There are a lot of new concepts with particular meanings that can be difficult to understand without a good underground. For this reason and also because not all the classes can be described in the main report this chapter has the objective to exemplify two main parts of the project – the product group and the resource group. These classes were chosen because they are very important in all model, without resources and product a factory does not make sense.

Product Group

Product Group is a particular area of the model. To exemplify this concept two examples will be demonstrated. The first example is about the relation between different product attributes, product group and product and the second example is about the product relation. The main classes that will be used in the example are the classes that belong to the diagram below.

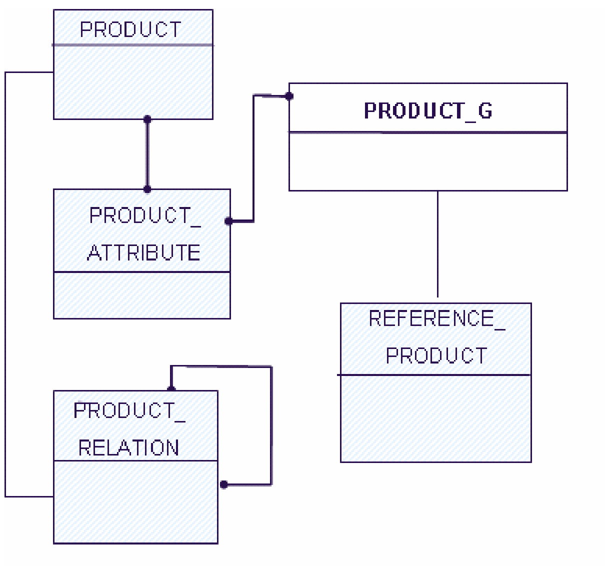


Figure 23 - Product Classes

First Example:

In the first example we pretended to describe the relation between the product group, product and product attribute. In the chapter three the concepts of product group, product and product attribute was already introduced, though in this chapter this concepts will be reminded as the product group is a class that group different products with the same product attribute. This relation where the product group and the product are not directly linked gives to the model a huge flexibility, in this way, a product can not belong to any product group and the product group where the product belongs will depend on the granularity that was chosen. This relation will give a good dynamic to the model.

Flexibility is so important because this data base will aggregate different business processes and the granularity of each one require are different. In this way for the capacity planning and investment planning the product group does not have products, for the planners the important is to know only the attributes the product group. On the other hand for the shop floor control processes, for example for scheduling, are important to know the product and the product group. The example below can better explain this relation and the way that these classes are connected.

Product Attribute		
#Cod	Name	Value
1	Generation	512M
2	Generation	1G
3	Shrink	T90
4	Shrink	T11

Product Group
#Name
512M T90
512M T11
1G T90
1G T11

Product
#Name
49300246
49300247
49300249
49300290

Table 7 - Product Group Classes

Each table represented means a class – product group, product and product attribute. As the product group is a group of products with the same attributes then it can be said that the product group 512M T90 has two attributes identified, they are the generation 512M and the Shrink T90. With both attributes and with this granularity only one product group can be defined, but many products can belong to this product group for example the 49300246 and the 49300247.

Other potentiality of this model is that one product attribute can define more than one product group for example the attribute generation 512M define the product group 512M T90 but also the product group 512M T11. This facility gives to the model much more flexible and in this way the model needs to keep less information because one attribute defines more than one product group.

Although a combination of product attributes only can have one product group, a product group can have many products with the same products attributes for example the product 49300246 and the product 49300247 both have the generation 512M and the shrink T90.

Second Example:

Along the production line the product is being modified. Every time that the product suffers a transformation, this transformation can be physical or not, it will be considered a new product. This concept was introduced in the chapter three but will be exemplified in the picture below. In the Product Relation Class the key attribute is the Relation Type Attribute, this attribute can be a source or a result depending the type of relation.

	Product Relation	Product	WorkCenter	
	#Relation Type	#Name	#Name	
OUT	Source	49300246	DIEBANK	Relation 1
IN	Result	69300395	PREASSY	Relation 1
IN	Source	69300395	PREASSY	Relation 2
OUT	Result	49300246	DIEBANK	Relation 2
OUT	Source	79300985	ASSY	Relation 3 / Relation 4
IN	Result	99314647	TEST	Relation 3
IN	Result	99314649	TEST	Relation 4

Table 8 - Product Relation

A baunumber is a code that defines a specific product. One kind of information that the code keeps is the place where the product is in each moment. So, to guarantee that the product is always identified, its name needs to change every time it enters in a new work center.

For example, when a product is in the DIEBANK (die bank is the first area where the wafer are stored before enter in the production line) has the name of 49300246. When it enters in the PREASSY (the next work center) the 49300246 will be transformed in 69300395 (Relation 1). Therefore the inverse relation (Relation 2) is allowed in the model (In in PREASSY Out in DIEBANK) this relation doesn't make sense for Qimonda because it is pretended that this transformation goes with the product flow and the product never goes back in the flow.

One product in Work Center can be also transformed in two or more products in the next Work Centers (Relation 3 and Relation 4). In the picture below, this relation is exemplified – the 79300985 goes out to the ASSY and when it enter in the TEST enter as two different products 99314647 and 99315649 but both belong to the same Product Group. Usually this distinction is caused by quality specifications.

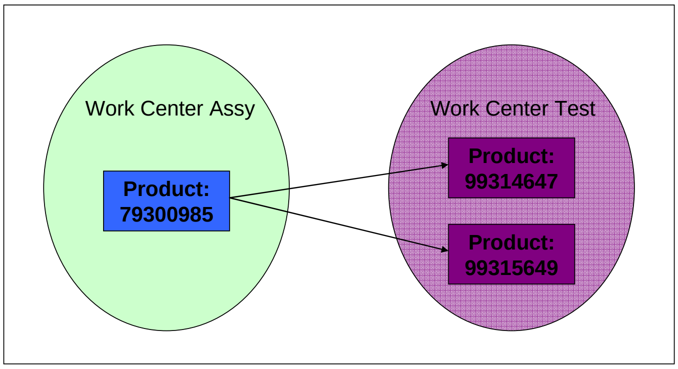


Figure 24 - Product Relation

Resource Group

Certainly the Resource Group area is the part of the model that implies more functionality and problem descriptions. For this reason, the Resource Group is chosen to be described with examples.

The picture below is the class diagram of the resource group area. The resource group area as the product group area has a class of the attribute – Resource Attribute Class. Though, the Resource Group and the Resource are linked by the Resource Attribute, they are also directly linked. Although all the product attributes are string in the resource attributes this fact is not true. Indeed a resource attribute can be a string, a float or a integer, thus with the objective to solve this problem in the resource attribute was done a generalization in three classes, which are the String Class, the Float Class and the Integer Class.

Another important point of the resource area is that the date needs to be kept. On the contrary of the product, that does not change with the date, the resource attributes can change with the time. For example the resource status can be "received" when the resource enters in the factory and will change to "install" when is released for production.

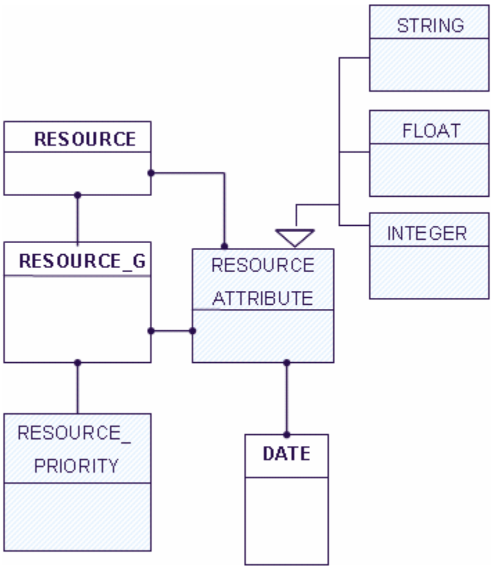


Figure 25 - Resource Classes

First Example:

The first example is about the resource priority. In this example some new classes are introduced: Resource Priority, Product Group, Operation and Resource Group. In the table below these classes are represented in tables and each line represented a related entry.

Resource Priority	Product Group	Operation	Resource Group
#Priority Number	#Name	#Name	#Name
1	512M GH11	UNLOAD	PTB-11
1	512M GH11	UNLOAD	BLU-(MR9210)
2	512M GH11	UNLOAD	PTB-12
2	512M GH11	UNLOAD	BLU-(MR9220)
1	1G T90	WIRE BOND	TSUNAMI 3100
2	1G T90	WIRE BOND	SHINKAWA
3	1G T90	WIRE BOND	ESEC 3088ip

Table 9 - Resource Priority Example

For the Shop Floor Control Processes a priority relation between the resources and the products is important to be created. What in the table before is demonstrated is for one product group in one operation there are a priority number for each combination of tool and equipment or only for equipment when this one does not need a tool.

In the first case for the same product group in the UNLOAD operation the priority is to be produced in the equipment BLU-(MR9210) with the tool PTB11. If this option is not possible the second priority will be used, the BLU-(MR9220) but with the tool PTB-12. If in the first alternative only the equipment is not available it is possible to use the equipment of the second alternative with the tool of the first alternative. With this option the model becomes much more flexible and the process will be optimized. It is important not forget that this potentiality only is used if the equipments are the same type, if the machines are of different types this can not be used. Nevertheless, this is a production issue; what the model pretend is to facilitate this option.

If the operation hasn't tools associated the case is easier, if the first priority is not available the second priority will be used. For example, in the table before for the product group of 1G T90 the priority is produced in the TSUNAMI 3100 equipment or in the SHINKAWA if the first is not available or in the ESEC 3088ip as the last option.

Second Example:

The second example of the resource is about a functionality of the model that is pretended to be implemented. The table below represents three classes: the resource class, the resource attribute class and the resource group. The resource attribute in this case is the float class but for simplification it will be called resource attribute.

Resource
#Name
SHINKAWA 01001
SHINKAWA 01002

.

Resource Attribute		
#Cod	Name	Value
1	Availability	90%
2	Availability	80%
3	Availability	85%

Resource Group
#Name
SHINKAWA
SHINKAWA
SHINKAWA

Table 10 - Resource Attribute

In this example there are two resources that belong to the same resource group – Shinkawa. The first resource Shinkawa 01001 has 90% of availability, the second Shinkawa 01002 has 80% of availability. The model calculates the availability of the resource group Shinkawa as the average of the availabilities of the resources – in this example is 85%.

5 Functionalities

In the chapter five will be presented the functionalities that are pretended for the model.

Although it was mentioned that this model will not be connected with interfaces some of functionalities that now will be presented will require the interfaces. This was done as suggestion for a next version of this model which can be though to create interfaces.

5.1 Resource Group Functionality

There are some functionalities that might be considered in the resource group category, this means in the class that are related with the resource. The main two functionalities thought are an automatically counter of the resources and the average of the availability.

The first functionality, the counter, was thought because all the resources of a resource group are not bought at the same time. Thus, every time that was bought a new resource was necessary to actualize manually the number of resources in the resource group. This hypothesis needs a big effort and can incur in errors. In this way, for this necessity be satisfied was created a functionality that counts automatically the number of resources that a resource group has and every time that there is a new entrance in the resource class the model identify the correspondent resource group and addict the resource in the resource group. The inverse (when a resource move out of the factory) is also valid.

A resource group can have a lot of different resources and with the shop floor control systems is possible to know the availability that each resource has. The functionality that is propose is analyzed all the resources that belong to the same resource group, and keep the value of the availability of each one and to do the average of these values. This functionality can only be done with the help of a business rule.

Sometimes is possible that a resource group has no resources, for example if the resource become obsolete the resources will move out from the company but can be interesting to keep the information about that resource group. This option only is possible with the business rule which aggregates the information from resource to resource group turned off. Thus, always that information about the resource group is needed and there are no resources, the relation between resource and resource attribute will be ignored and the information will be given by the resource group class.

5.2 Exception Functionality

In order to allow an exception, it is necessary that a rule exists. What is pretended in this functionality is to certify that first there is the rule and then the exception.

When is pretended to produce a product it is needed the relations between Product – Operation and between Operation and Resource Group. Only after this, it will be certified if this relation has an exception or not. If it is the case, in addition to the normal rule, it is needed an exception business rule to allocate or disallow a product in the correct resources.

This means that the algorithm needs to evaluate first the generic rule and then the exception rule on top.

5.3 Resource Group Capacity

The class Result of a Resource Group to Produce (RESULT_RG_PROD) will be an output class. This class objective is to transmit the result of what is the capacity of a resource group. To acquire this result this class needs the functionality to sum the results of the class Result of a Resource Group to Produce in an Operation (RESULT_RG _PROD _OP) that belongs to the same Resource Group.

5.4 Resource Attribute

There are a lot of different attributes for the same Resource. Some of these attributes are inputs for some outputs. Thus, it is fundamental that the same attribute is always written in the same way (same name and with the same combinations of capitals letters). Because the manual entries may lead to frequent errors, there will be created a combo box with all the names of all the attributes and in this way the user only needs to select a proposed attribute.

The attributes proposed for the combo box are represented in the table below.

Resource Attribute	Resource Type
Model	string
Vendor	string
Resource	string
Functionality	string
Operation	string
Resource_Amount	integer
Availability	float
EquipmentUptime	float
Downtime	float
EngineeringTime	float
StandbyTime	float
ProductiveTime	float
UnscheduledDowntime	float
ScheduledDowntime	float
Production_Corridor	float
Engineering_Corridor	float

Table 11 - Resource Attribute Oprion

One of the resource attributes is the Availability. But the availability can be calculated in two different forms. The availability can be an input written by the Capacity Planner or can be a result of the engineering hours, maintenance hours and total hours. This different type of hours is explained in the appendix B (Equipment time categories). If the Availability attribute is composed of values then the Availability is that value. If the Availability has no value, the functionality will be active and the availability will be the result between the uptime and the engineering hours. The functionality is used for the Resource and the Resource Group.

Time categories is a name of some attributes of the resource attribute class. This class has two attributes very important for the time categories – the attribute unit of measure (percentage, hours, …) and the attribute type (input, final, …).

The time categories are an input (attribute type) that the user inserts in the model; this input can be in percentage or in hours (attribute unit of measure). As it is a user input, it can be interesting that the user has the autonomy to choose if he wants to insert the time categories in hours or in percentage or both. If the user pretends to put only one of the categories will be easy to understand. An example: if the user inserts that the uptime is 2 hours is easy but if the user pretends that the uptime will be 2 hours and also 20 % of the time we have not sensibility for this. So, it is necessary to create a new attribute type that gives the sum when the user pretends to insert the time categories in two formats. This new attribute unit of measure will be given in hours or in percentage and it will be a specification of the user. At the same time it will be the functionality that converts one input into the final format by summing this one with the other, in this way we will understand that 2 hours and 20% of the time is the same of 30% of the time or 6 hours.

To the same attribute we can have more than one type as it was explained in the chapter 3.5 (type: “input” or “final” for example). To be user friendly once that this new attribute type “final” always needs to be the result when the user pretends to have two categories, we also pretend that it will be the same even when we only have one input. Indeed, if we pretend the result in hours and we gave the input in hours the time will be the same, but if we give the time in hours and we gave the input in percentage this attribute converts the input. The attribute type “final” is always a result attribute no matter if we have both inputs or only one.

5.5 Time Categories Functionality

Qimonda has different types of Time Categories (all the categories are explained in the appendix A). These Time Categories are an example of some attribute that are inserted in the Resource Attribute with the same characteristics that were explained in the chapter before.

What is pretended is to create a functionality that sum some of these time categories and give the result in other time category. In particular one of the sums is the UnScheduledDownTime type “final” with ScheduledDownTime type “final” and returns the result in the DownTime type “final”. The other aggregation is EngineeringTime type “final” with StandByTime type “final” and ProductiveTime type “final” resulting in the EquipmentUptime type “final”. For the attributes ScheduledDownTime, UnScheduledDownTime, EngineeringTime, StandByTime, ProductiveTime, though the values that we are interested in are their “final” values, they are “inputs” for the DownTime and EquipmentUptime attributes.

5.6 Capacity Necessary to Produce per Operation in Equivalents

This functionality will calculate the available capacity to produce per operation in equivalents. This result is expressed in the class Result per Product and Operation (RESULT_PROD_OP) in the attribute Equivalent Volume In. This attribute is a result (product) between the Physical Volume In and the complexity of that product.

Keeping the information in equivalents transmits more sensibility because all the operations have the results in the same unit.

5.7 Spare Capacity

The spare capacity is the excess of capacity that is not necessary for the production, the spare capacity use to be presented either in equivalents or in percentage. If one hand is fundamental decrease the spare capacity because this means that there equipment that is not utilized, on the other hand if there is not any spare capacity if there is a problem the line can not react.

To calculate the spare capacity is needed the help of a functionality. The Spare Capacity is the result between the maximum capacity that one resource group can produce less the value of the local starts result in equivalents for the same resource group. If the spare capacity is positive it means that is possible to do the plan, if it is negative it means that there is not enough capacity to do the plan. The area with less spare capacity is the bottleneck

An additional option is to divide the spare capacity by the maximum capacity of a resource group. Thus the result is given in percentage and transmits more sensibility.

5.8 Date Aggregation

In the Date Class is needed to relate the daily information and the weekly information. What is pretended with this functionality is that all the daily information will be converted in weekly information (Calendar and Fiscal Week). When the weekly information is in Calendar Week, can be converted into Fiscal Week and when it is in Fiscal Week can be converted into Calendar Week.

With the help of this business rule is easier to associate a day in a week and the days that a week has. When there is the information in weeks will be easy to convert in both calendars (Calendar Week and Fiscal Week). The same approach is valid for monthly and quarterly associations.

All the days in this class must be available in the data base in order to the aggregations work correctly. Only in this way it is possible to have information about the days that a week has.

5.9 Investment Planning

The model is created with the objective to help the users to optimize their jobs. Thus, it is created some functionalities that save some human work and advise the users for some notes. This is the case that this business rule will help a lot the user.

As mentioned before if the spare capacity is negative that means that is not possible to do the plan, though it was also mentioned that is always necessary a spare for the line produce. This necessity will be the Reference Spare. The reference spare is the minimum of spare in equivalents per resource group.

Every time the Spare Capacity is minor than the Reference Spare the model will calculate the number of increasing resources that is needed to buy for the Spare Capacity to be equal to the Reference Capacity. This number is calculated in first way like a decimal number and in a second way rounded. An example: if there are two equipment and two different scenarios, one with a need of 0.5 equipment and the other one with a need of 1.4 equipment this needs are completely different. Though, a necessity of 0.5 the rounded number is 1 and with a necessity of 1.4 the rounded number is also 1. Indeed these necessities are completely different and probably with the first one the Capacity Planner will decide not to buy the equipment and in the second one maybe he chooses to buy two machines. Because of this it is important to keep both results. The rounded number can be written by the Capacity Planner when he doesn't agree with the round number proposed by the model.

The money spent is the product between the rounded numbers of resources with the price of the resource. The result is the investment per Resource Group. Sometimes it is important to have the investment per Work Center, in this way the functionality will be the sum of all the investment per Resource Group that exist in that Work Center.

The introduction of a new product implies some cost, sometimes being sunk cost (cost of equipment that only can be use to produce that product). This cost needs to be studied carefully, to know if it is good or not to introduce the new product. Because of this analysis the attribute that gives the investment per Product Group was created. This investment will be the sum of the investment per Resource Group that only was bought to produce that specific product.

5.10 Set Up Times

The Set Up Time and the Set Up Sequence are modeled in the step of the Resource Group. This is correct once that for the same resources group all the resources have the same products that can be produced with the same set up times. But different resources can produce at the same time different products. Thus, the products that the resource can produce next are different and the set up times are also different. An example, the Resource1 and the Resource 2, both from the Resource Group A, are producing P1 and P2 respectively. If next, both want to produce the P3 the Set Up times will be different. So, though these classes are modeled with the granularity of the Resource Group it is needed a functionality that does the propagation of information in the resource granularity. The main reason of this functionality is however to provide disaggregated data to target systems.

The second functionality is once we have the list of product that might be produced next in a resource (Resource granularity – first functionality), the functionality will return a sorted list with all the Products that are allowed to be produced in that Resource Group and their set up times. The first value of this list is the fastest set up and the last value is the slowest. With the second functionality the user can chose the products in a friendly way.

5.11 Business Processes and Required Classes

This model pretends to specify three different kinds of business processes, Capacity Planning, Investment Planning and Shop Floor Control.

Once that we have a list of different classes we will present a matrix with all the classes and the business processes where they belong.

Name of Class	Capacity Planning	Investment Planning	Shop Floor Control
Product Group	X	X	
Product			X
Product Attribute	X	X	X
Reference Product	X	X	
Product Relation	X	X	X
Plan	X	X	
Demand	X	X	X
Location	X	X	X
Model	X	X	X
Work Center	X	X	X
Resource Group	X	X	X
Resource Priority	X	X	X
Resource			X
Resource Attribute	X	X	X
Enterprise Route	X	X	X
Route	X	X	X
Operation	X	X	X
Route Sequence			X
Operation Sequence			X
Process Time	X	X	X
Date	X	X	X
Yield	X	X	X
Planning Parameters	X	X	X
Prod Operation Res Exceptions	X	X	X
Product Route Exception	X	X	X
Result Resource Group	X	X	
Result Work Center	X	X	
Result Product Operation	X	X	
Result Resource Group Operation	X	X	
Result Resource Group Product	X	X	

Table 12 - Classes and their Business Processes

5.12 Forecast Functionalities

As mentioned before in the chapter 2.28, it will be useful to keep information about the forecast values and the actual values.

There are two main functionalities that might be considered in this chapter: the calculation with actual values or forecasting values and the forecast functions.

In the first functionality what is pretended is for a special date in the past, for example calendar week 22, the model returns the values that were planned for that date and the actual values that really happened. What is pretended with this functionality is to have the possibility to come back to a specific week and calculate based on the actual values or the planned values and in this way the user can compare with the plan.

In the second functionality the user will have the possibility to choose if he pretends or not to do a forecasting function. There are many functions that can be inserted in this option but it is not the objective to explain them. It is only pretended to explain that they exist and they can be done.

6 Conclusion

The main and expected objective of this project was the specification of a model that can improve capacity planning, investment planning and shop floor control processes such as scheduling and dispatching. This objective was completely achieved and the result is presented by a solid and global model with more than 30 entities and 80 attributes.

During the project development some difficulties were found. The main difficulties were linked with three factors: the first is that Qimonda plays in a fast industry and so words like flexibility and dynamic were mandatory and sometimes this necessity wasn't easy to be implemented. The second main difficulty is related with the product mix that is produced in the factory. At each moment the production line has dozen of different products which in total makes an output of 13milions units produced per week. This complexity needs a huge organization in the production line, so it is important that some issues are optimized such as the product relations, resource allocations and resource dedication and prioritizing resources. The third main point that should be referenced is the fact that all the entities are intimately related which means that all the issues of the model are related (the model has more than 60 relations). So, every time that it is pretended to add any entity or to optimize a set of entities there is a snow ball effect, this means, that many other entities need to be reformulated because one entity was changed or added. This problematic caused that a lot of time was spent in the model specification.

Another particular issue of this project is the fact that the users of the model are not persons but other solutions that use the underlying data for other purposes. This means that this model will support many other models that will integrate the output of the CRP in their inputs. Since the users are solutions it is pretended to keep the model flexible enough in order to enable integration of new systems every time that it is pretended and without influencing the others users.

Since the beginning of the project, it was not pretended to implement the achieved solution. The implementation of the project will be responsibility of a future project. Between the specification phase (this project) and the implementation phase other important project has started. This intermediary's project has two main objectives: in a first phase will have the objective to do an evaluation criteria about the main topics that might be considered in the implementation, in a second phase the objective is to do a market research about a solution that can implement the present project, if any solution is found (this is a possible scenario) it will be necessary to create a team to be responsible for the implementation of the model. This team can be found in the local IT team or in an external company.

As soon as implementation is finished the developed tool is intended to be used in other Qimonda sites around the world and being responsible for the development of the basis of a project with this importance has been since the beginning and will continue to be a grateful and enriching experience.

7 References and Bibliography

[1] Wikipedia. “Qimonda”. 2007 March. <http://en.wikipedia.org/wiki/Qimonda>. Accessed 2007 March 6.

[2] Qimonda AG. Core Values. <http://intranet.qimonda.com/Downloads/CF/C/Comp_Pres/375,5,Our Core Values>. Accessed 2007 March 6

[3] Wikipedia. “Data Modeling”. 2007 February. <http://en.wikipedia.org/wiki/Data_modeling>. Accessed 2007 March 8.

[4] Wikipedia. “Standard Data Model”. 2007 February. <http://en.wikipedia.org/wiki/Standard_data_model>. Accessed 2007 March 8.

[5] Wikipedia. “Ontology”. 2007 February. <http://en.wikipedia.org/wiki/Ontology_%28computer_science%29>. Accessed 2007 March 8.

[6] Data Modeling Using EXPRESS-G for IFC Development. <http://www.iai-international.org/Model/documentation/Data_Modelling_Using_EXPRESS-G_for_IFC_Development.pdf>. Accessed 2007 March 8.

[7] Wikipedia. “STEP”. 2007 February. <http://en.wikipedia.org/wiki/STEP_%28ISO_10303%29>. Accessed 2007 March 8.

[8] Wikipedia. “ISO 10303”. 2007 February. <http://en.wikipedia.org/wiki/ISO_10303-11>. Accessed 2007 March 8.

[9] Wikipedia. “UML”. 2007 February. <http://en.wikipedia.org/wiki/Unified_Modeling_Language>. Accessed 2007 March 8.

[10] UML. “Unified Modeling Language”. 2007 January. UML home page <http://www.uml.org/>. Accessed 2007 March 8.

[11] Wikipedia. “Use Case”. 2007 March. <http://en.wikipedia.org/wiki/Use_case>. Accessed 2007 March 20.

[12] Wikipedia. “Class Diagram“. 2007 March. <http://en.wikipedia.org/wiki/Class_diagram>. Accessed 2007 March 20.

[13] Cunha, João. “Apontamentos das aulas de Sistemas de Informação“. 2004.

[14] Ramos, Nogueira. “Desenhar Bases de Dados com UML”. 2006. Edições Sílabo.

[15] Wikipedia. “Product (business)”. 2007 April. <http://en.wikipedia.org/wiki/Product_%28business%29>. Accessed 2007 April 16.

[16] Wikipedia. “Resource Allocation”. 2007 March. <http://en.wikipedia.org/wiki/Resource_allocation>. Accessed 2007 March 27.

[18] Infineon Technology. “Technical Regulation No.25 – Reliability Availability Maintainability (RAM)”. 2002 . <http://intra.muc.infineon.com/hl_tr/tr25e/tr25.pdf>.

[19] OMG – Object Management Group. 2007 February 2. OMG home page. <http://www.omg.org/gettingstarted/what_is_uml.htm>. Accessed 2007 March 8.

[20] Product Data Management Enablers Specification, 2006 November, OMG home page documents,<http://www.omg.org/technology/documents/formal/product_data_management_enablers.htm>. Accessed 2007 April.

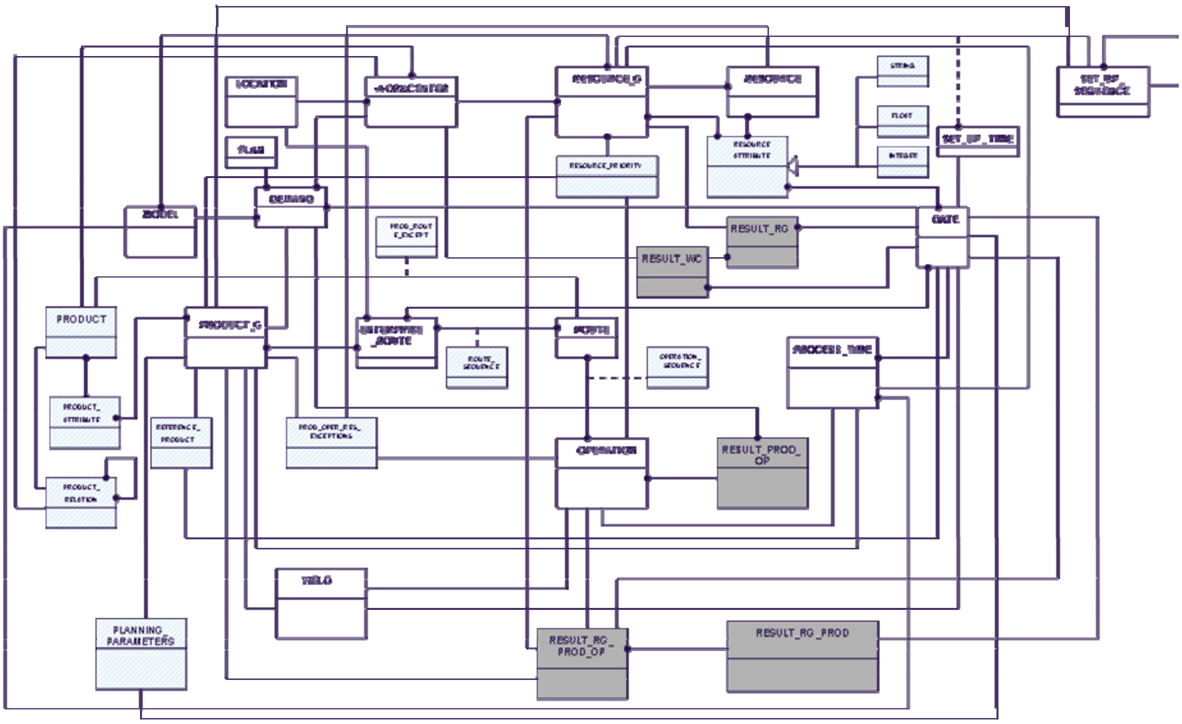
[21] Teradata Magazine. “Leveraging the Industrial Logical Data Models as Your Enterprise Data Model”. 2007 January. <http://www.teradata.com/t/page/162502/index.html>. Accessed 2007 March 12.

[24] Wikipedia. “Capacity Planning”. 2007 June. <http://en.wikipedia.org/wiki/Capacity_planning>. Accessed 2007 June 28.

[25] Ambler W. Scott. “Data Modeling 101”. November 19 2006. <http://www.agiledata.org/essays/dataModeling101.html>. Accessed 2007 March 12.

APPENDIX A: Diagram of classes

In this chapter will be described the characteristics of all the classes. All classes attributes are here described.



Legend :

- Main Class
- Results Class
- Complementary Class

Resource_Group Class

The resource group class serves to keep information related to a group of manufacturing resources with equal or comparable attributes or behaviors.

Convention	Example
RESOURCE_GROUP	-
NAME [string] MODEL [string] RESOURCE_AMOUNT [integer] FUNCTIONAL_TYPE [string] VENDOR [string] RESOURCE_TYPE [string]	C-BLU-(MR9220) MR9220 1 LOAD_UNLOADER MIRAE MAIN_RESOURCE
WORKCENTER [1 – N] RESOURCE [N -1] RESOURCE_ATTRIBUTE [N-N] PROCESS_TIME [1 – 1] RESOURCE_PRIORITY [1 – N] MODEL [N – N] CAPA_RG [1 – 1] CAPA_RG_PROD_OP [1-1] SET_UP_SEQUENCE [1-1] SET_UP_TIMES [N-1]	CASSY HITACHI INSTALL 10 1 MODEL_1 - - 1G T11 TFBGA 68 30 MINUTES

Set Up Sequence Class

The set up sequence gives the products that can be produce next in a resource group.

Convention	Example
SET_UP_SEQUENCE	-
SET_UP_SEQUENCE_TYPE [string]	FROM TO
RESOURCE_GROUP [1 - 1] PRODUCT_GROUP [1-1] SET_UP_TIME [1- N]	C-BLU-(MR9220) 512M T90 TFBGA 60 30 min
SET_UP_SEQUENCE [1-N]	1G T11 TFBGA 68

Set Up Times

Convention	**Example**
SET_UP_TIME	-
SET_UP_TIME [float]	20 minutes
SET_UP_SEQUENCE [1-N] RESOURCE_GROUP [1-N] DATE [1-1]	1G T11 TFBGA 68 C-BLU-(MR9220) 21-05-2007

Resource Class

The resource class serves to keep information related to a specific resource.

Convention	**Example**
RESOURCE	-
NAME [string]	EPFBLU-0001
RESOURCE_ATTRIBUTE [N - N] RESOURCE_GROUP [1 - N] PROD_OP_RES_EXCEPTIONS [1-1]	27 – 03 -2007 C-BLU-(MR9220)

Resource Attribute Class

The Resource Attribute Class keeps the information about the attributes of the resources. This information can be done in different ways, as a string, as a float or as an integer. It is very important to keep the information in the correct format because some of them will be used for calculus. Because of this the Resource Attribute Class has a generalization in three classes: String, Float, and Integer.

Convention	Example
RESOURCE ATTRIBUTE	-
DATE [N - N] RESOURCE_GROUP [N - N] RESOURCE [N –N]	27 – 03 -2007 C-BLU-(MR9220) C-BLU-(MR9220) -1

Convention	Example
STRING	-
NAME [string] VALUE [string] TYPE [string]	STATUS Installed ACTUAL_INPUT; ACTUAL_FINAL; FORECAST_FINAL
RESOURCE_ATTRIBUYR [1 - 1]	

Convention	Example
FLOAT	-
NAME [string] RESULT [float] UNIT_OF_MEASURE [string] TYPE [string]	AVAILABILITY 95 PERCENTAGE; HOURS; MINUTES ACTUAL_INPUT; ACTUAL_FINAL; FORECAST_FINAL
RESOURCE_ATTRIBUYR [1 - 1]	

Convention	Example
INTEGER	-
NAME [string] RESULT [integer] UNIT_OF_MEASURE [string] TYPE [string]	RESOURCE_AMOUNT 5 - ACTUAL_INPUT; ACTUAL_FINAL; FORECAST_FINAL
RESOURCE_ATTRIBUYR [1 - 1]	

Work Center Class

In a first view the WORKCENTER was defined with many RESOURCE_GROUP and many OPERATIONS. If this is true we will have:

RESOURCE_G	WORKCENTER
MR9220	BIN
MR9210	BIN
BIO – MTX	BIN

OPERATION	WORKCENTER
LOAD	BIN
UNLOAD	BIN
BURNIN	BIN

In this way we don't know which OPERATION correspond to each RESOURC_G. So RESOURCE_G needs to be directly related with OPERATION.

RESOURCE_G	OPERATION
MR9220	LOAD
MR9210	UNLOAD
BIO – MTX	BURNIN

If this relation is granted we don't need the directly relation between OPERATION and WORKCENTER.

Convention	Example
WORKCENTER	-
WORKCENTER_NAME [string] AREA_NAME [string] MANUFACTURING_LEVEL_NAME	 burn_in CTEST
RESOURCE_GROUP [N - 1] LOCATION [1 –N] PRODUCT [1 – N] PRODUCT_RELATION [1-1] RES_WC [1-1] DEMAND [N-1]	C-BLU-(MR9220) QIMONDA_PT 79039334 - - 5000K

Process_Time Class

The process time class serves to keep the time in minutes per K that one resource group in one operation needs to produce a product in a specific data.

Other information that might be kept in this class is the name of the recipe that the problem will do in the resource.

Convention	Example
PROCESS_TIME	-
TIME [float] UNIT_OF_MEASURE RECIPE_NAME	10.5 Minutes AX58
DATE [N - 1] RESOURCE_GROUP [1 - 1] PRODUCT _GROUP[1 -1] OPERATION [1 - 1] MODEL [1 - N]	27 – 03 -2007 C-BLU-(MR9220) 512M GH11 PG-TFBGA-136-53 x32 LOAD MODEL_1

Product Group Class

The product class serves to keep information related to a group of products.

A PRODUCT GROUP has many ENTERPRISE_ROUTES, an ENTERPRISE_ROUTE has many ROUTES and a ROUTE has many OPERATIONS. So a PRODUCT GROUP has many OPERATIONS. This relationship is granted without any directly relation between the PRODUCT GROUP and OPERATION.

Convention	Example
PRODUCT_GROUP	-
NAME [string]	512M GH11 PG-TFBGA-136-53 x32
PROCESS_TIME [1 -1]	10
ENTERPRISE_ROUTE [N - N]	DBG_DDR2_PRINT
PLANNING_PARAMENTERS [1 – 1]	
YIELD [1 – 1]	98%
CAPA_RG_PROD_OP [1 -1]	1000
RESOURCE_PRIORITY [1 – 1]	2
DEMAND [1 – 1]	5000
PRODUCT_ATTRIBUTE [N-N]	GENERATION 512
PRODUCT_REFERENCE [1-1]	512M T90 PG-FBGA 60 x8
PROD_OP_RES_EXCEPTIONS [1-1]	
SET_UP_SEQUENCE [1-1]	1G T11 TFBGA 68

Product Class

The class Product serves to keep the information to a specific product.

Convention	Example
PRODUCT	-
BAUNUMBER[string]	79039334
PRODUCT_ATTRIBUTE[N -N]	GENERATION 512
WORKCENTER [N-1]	DIEBOND
PRODUCT_RELATION [N-N]	ASSY – 79039334
ROUTE [1-1]	ROUTE_1
PROD_ROUTE_EXCEPTIONS [1-1]	

Product Attribute Class

The class Attribute serves to keep the information about the attribute that defines the product.

Convention	Example
PRODUCT_ATTRIBUTE	-
NAME [string] VALUE [string]	GENERATION 512M
PRODUCT[N -N] PRODUCT_GROUP [N-N]	79039334 512M GH11 PG-TFBGA-136-53 x32

Product Relation Class

This class keeps the relations between the different Baunumbers. Always that A Baunumber enters in a Work Center it transforms itself into another Baunumber. This particular relation between Baunumber and Work Center is stored in this class. That way the model was modeled with the help of one relation between the class PRODUCT_RELATION and itself. In this way we are imposing the hierarchy.

Convention	Example
PRODUCT_RELATION	-
RELATION_TYPE [string]	SOURCE RESULT
PRODUCT [N -1] WORKCENTER [1-1]	79309334 ASSY
PRODUCT_RELATION [N-N]	

Reference Product

The Reference Product is the product that we consider that it has complexity 1 (one) and the others are calculate based on the Reference product.

Convention	Example
REFERENCE_PRODUCT	-
PRODUCT [string]	512M T90 PG - FBGA 60
PRODUCT_GROUP [1 -1] DATE [1 - 1]	512M T90 PG - FBGA 60 26/4/2006

Result per Resource Group Class

This class keeps the results that are calculated per Resource Group. This class gives us information such as, Spare Capacity and the Local Starts (Maximum Reference Capacity of one Resource from a Resource Group). All these attributes have the results in equivalent product.

Convention	Example
RESULT_ RG	-
MAX_CAPA_EQUI [integer] SPARE_CAPA_EQUI [integer] LOCAL_STARTS_EQUI [integer] NR_EQUIPMENT_NEED [float] NR_EQUIPMENT_INVEST [integer] INVESTMENT [integer]	1000 20 1200 3.1 3 152000€
RESOURCE_GROUP [1 -N] DATE [N – 1]	HITACHI 25/04/2007

Result Work Center class

This class serves to keep the information about how much the investment is done in a specific Work Center.

Convention	Example
RESULT_WC	-
INVESTMENT [integer]	900000 €
CAPA_RG [1 -1] DATE [1 - 1]	- 26/4/2006

Enterprise_Route Class

The enterprise route defines all the routes that a product can do. But, a product can do more than one enterprise route, sometimes the enterprise route is changed for another, in this case it is important that the class keeps the date information. Because of this the class ENTERPRISE_ROUTE is connected with the class DATE.

Different products can have the same ENTERPRISE_ROUTE at the same time.

Convention	**Example**
ENTERPRISE_ROUTE	-
NAME [integer] SEQUENCE_NUMBER [integer]	DBG_DDR2_PRINT 1
DATE [N - N] PRODUCT_GROUP [N -N] ROUTE [N – N] LOCATION [1-N] ROUTE_SEQUENCE [1-1]	27 – 03 -2007 512M GH11 PG-TFBGA-136-53 x32 DBG QIMONDA_PT 2

Route Class

The route class serves to keep information related to the route that a product did in a period of time.

Convention	**Example**
ROUTE	-
NAME [integer]	DBG
OPERATION [N – N] ENTERPRISE_ROUTE [N – N] PRODUCT [1-1] PROD_ROUTE_EXCEPTIONS [1-1]	LOAD DBG_DDR2_PRINT 79309334

Operation Class

The operation class serves to keep information related to each operation in each area.

Convention	Example
OPERATION	-
OPERATION [string]	LOAD
ROUTE [N – N]	DBG
PROCESS_TIME [1 – 1]	10
LOCAL_STARTS [1 – N]	5000
RESOURCE_PRIORITY [1 – 1]	1
YIELD [1 – 1]	90%
CAPA_RG_PROD_OP	-

Location Class

The Location Class serves to keep the information about where is the place that the product will be produce.

Convention	Example
LOCATION	-
NAME [string]	QIMONDA_PT
LOCATION_CODE [string]	QPT
TYPE [string]	Backend
COUNTRY [string]	Portugal
ENTERPRISE_ROUTE [N -1]	ER_2
DEMAND [N - 1]	13000k
WORKCENTER [N – 1]	ASSY

Demand Class

The Demand Class serves to keep the information about what will be the volume that we need to have in output for each product.

As it was previous explained we have two different types of demands, demands for the production corridor and demands for the engineering corridor. The type of demand characterizes the type of the products and where the products might be produced, in the engineering corridor or in the production corridor.

Convention	Example
DEMAND	-
DEMAND [integer] DEMAND_PRIORITY [string] DEMAND_TYPE [string]	13000 1 ENGINEERING; PRODUCTIVE
PRODUCT_GROUP [1 -1] LOCATION [1 - N] DATE [1 -1] PLAN[1- N] LOCAL_STARTS [N – 1] MODEL [1 – N]	512M GH11 PG-TFBGA-136-53 x32 QIMONDA_PT 27 – 03 -2007 VRFC 2000 MODEL_1

Date Class

The Date Class serves to keep temporarily information.

Convention	Example
DATE	-
DATE_FROM [date] CALENDAR_WEEK [integer] CALENDAR_MONTH [integer] CALENDAR_QUARTER [integer] CALENDAR_YEAR [integer] FISCAL WEEK [integer] FISCAL_MONTH [integer] FISCAL_QUARTER [integer] FISCAL_YEAR [integer]	27 – 03 -2007 13 03 Q1 2007 26 06 Q2 06/07
ENTERPRISE_ROUTE [1 -1] PROCESS_TIME [1 – 1] DEMAND [1 – 1] RESOURCE_ATTRIBUTE [1 – N] CAPA_RG [1 – N] YIELD [1 – 1] PLANNING_PARAMETERS [1 – 1] CAPA_RG_PROD_OP [1-1] CAPA_RG_PROD [1-1] SET_UP_TIME [1-1]	ER_1 10 5000k GENERATION 512M - 90% - - 5 min

Plan Class

The Plan serves to keep the information of what type of plan each demand is associated.

Convention	Example
PLAN	-
NAME [string] TYPE [string]	VRFC0705 VRFC
DEMAND [N - 1]	12000k

Yield Class

The Yield serves to keep the information about what is the percentage of good units associated with an operation.

Convention	Example
YIELD	-
YIELD [float]	88%
OPERATION [1-1] PRODUCT_GROUP [1- 1] DATE [1 – 1]	LOAD 512M GH11 PG-TFBGA-136-53 x32 27 – 03 -2007

Result per Product and Operation Class

A product, since it begins in the production line until it goes to the shipping, passes for different operations. Each operation has a Yield (the percentage of good units) associate so in each operation there are some loses. Because of that it is important to know approximately how many units we need in the entrance of each operation so that in the end we have the demand that we want. This volume is represented in the RESULT_PROD_OP class.

In this class we have two attributes: the first one Physical Volume give us the result in the amount of the product. And the Equivalent Volume give us the result in equivalents units.

Convention	Example
RESULT_PROD_OP	-
PHYSICAL_VOLUME_IN [integer] EQUIVALENT_VOLUME_IN [integer]	1000 1207
DEMAND [1 -N] OPERATION [N -1]	1000 LOAD

Result per Resource Group, Operation and Product

This class serves to store two kind of different information: the product complexity and the physic capacity for one Resource Group in one Operation.

Convention	Example
RES_RG_PROD_OP	-
PRODUCT_COMPLEXITY [integer] CAPA_PHISICALS [integer]	1.46 100
PRODUCT_GROUP [1 -1] OPERATION [1 -1] RESOURCE_GROUP [1-1] DATE [1-1] RES_RG_PROD [1 - N]	512M T90 PG-FBGA 60 x8 DIEBOND HITACHI 23-04-2006 -

Results of a Resource Group per Product Class

This class serves to keep the information about the capacity in physical products in a Resource Group.

Convention	Example
RES_RG_PRODUCT	-
CAPA_PHISICAL [integer] NEW_INVESTMENT_PG [integer]	5000 800000€
RES_RG_PROD_OP [N -1] DATE [1-1]	 23-5-2006

Model Class

Scenario Class serves to keep the information of alternative options that the Capacity Planner can do.

Convention	Example
MODEL	-
NAME [integer]	MODEL_1
PROCESS_TIME [N -1] RESOURCE_GROUP [N -N] DEMAND [N – 1]	12 HITACHI 1000

Planning Parameters Class

Convention	Example
PLANNING_PARAMETERS	-
GOOD_CHIPS_PER_WAFER [integer]	450
PRODUCT_GROUP [1 -1] DATE [1 -1]	

Resource Priority Class

If we consider that the RESOURCE_PRIORITY only has directly relation with RESOURCE_G and PRODUCT we will have:

RESOURCE_G	RESOURCE_PRIORITY
MR9220	1
MR9210	2
BIB -9220	1
BIB – 9210	2

And

RESOURCE_G	OPERATION
MR9220	UN_LOAD
MR9210	UN_LOAD
BIB -9220	UN_LOAD
BIB – 9210	UN_LOAD

In this way I will not know if the RESOURCE_PRIORITY is for the LOAD or UNLOAD operation. So it is necessary the RESOURCE_PRIORITY directly related with OPERATION.

RESOURCE_G	RESOURCE_PRIORITY	OPERATION
MR9220	1	LOAD
MR9210	2	UNLOAD
BIB -9220	1	LOAD
BIB – 9210	2	UNLOAD

Convention	Example
RESOURCE_PRIORITY	-
PRIORITY_NUMBER [integer]	2
RESOURCE_GROUP [N -1] PRODUCT_GROUP [1 -1] OPERATION [1 -1]	HITACHI 512M T90 PG-FBGA 64 x8 DIEBOND

Product Operation Resource Exceptions Class

Convention	Example
PROD_OPER_RES_EXCEPETIONS	-
EXCEPTION_TYPE [string]	DISALLOWED
RESOURCE [1 -1] PRODUCT_GROUP [1 -1] OPERATION [1 -1]	

Operation Sequence Class

Process Sequence Class serves to keep the operations as a sorted list in a route.

Convention	Example
OPERATION_SEQUENCE	-
SEQUENCE_NUMBER [integer]	5
ROUTE[1 -1] OPERATION [1 -1]	ROUTE_1 BURNIN

Route Sequence Class

Route Sequence Class serves to keeps the route s as a sorted list in an enterprise route.

Convention	Example
ROUTE_SEQUENCE	-
SEQUENCE_NUMBER [integer] ROUTE_PRIORITY [string]	6 STANDARD
ROUTE[1 -1] ENTERPRISE_ROUTE [1 -1]	ROUTE_1 ER_2

APPENDIX B: Comparison of Data Modeling Languages

The main objective of this project is to create a data modeling for capacity requirements planning.

Firstly we need to understand the meaning of data modeling. Data modeling is the process of creating a data model by applying a data model theory to create a data model instance. A data model theory is a formal data model description. When we are doing data modeling, we are structuring and organizing data. These data structures are then typically implemented in a database management system. In addition to defining and organizing the data, data modeling will impose (implicitly or explicitly) constrains or limitations on the data placed within the structure. Managing large quantities of structured and unstructured data is a primary function of information systems. Data models describe structured data for storage in data management systems such as relational databases. They typically do not describe unstructured data, such as word processing documents, email messages, pictures, digital audio, and video. Early phases of many software development projects emphasize the design a conceptual data model. Such a design can be detailed into a logical data model. In later stages, this model may be translated into physical data model. [3]

Standard data model is a data model that is widely applied in some industry, and shared amongst competitors to some degree. They are often defined by database vendors or operating system vendors and thus used by default whether suitable for given purpose or not. When in use, they tend to constrain software architecture significantly, as it becomes impossible to make decisions that require data distinctions not made in the standard model, without substantial effort in changing data gathering and building a so-called data warehouse. [4]

Ontology is a data model that represents a set of concepts within a domain and the relationships between those concepts. It is used to reason about the objects within that domain. Ontologies are used in artificial intelligence, the semantic web and information architecture as a form of knowledge representation about the world or some part of it [5]. EXPRESS is the data modeling language of Step and standardized as ISO 10303-11. An EXPRESS data model can be defined in two ways, textually and graphically. For formal verification and as input for tools such as SDAI (Standard Data Access Interface) the textual representation within an ASCII (American Standard Code for Information Interchange) file the most important one. The graphical representation on the other hand is often more suitable for human use such as explanation and tutorials. The graphical representation, called EXPRESS-G, is not able to represent all details that can be formulated in the textual form. Express is similar to programming languages such as PASCAL. Within a SCHEMA various data types can be defined together with structural constraints and algorithmic rules.

A main feature of EXPRESS is the possibility to formally validate a population of data types- this is to check for all the structural and algorithmic rules.[6]

Step (Standard for Exchange of Product model data) is an International Standard for the computer interpretable representation and exchange of industrial product data. The objective is to provide a mechanism that is capable of describing product data throughout the life cycle of a product, independent from any particular system. The nature of this description makes this suitable not only for neutral file exchange, but also as a basis for implementing and sharing product databases and archiving. Typically STEP can be used to exchange data between CAD, CAM, CAE, PDM/EDM and other CAx systems. STEP is addressing product data from mechanical and electrical design, analysis and manufacturing, with additional information specific to various industries such as automotive, aerospace, building construction, ship, oil & gas, process plants and other. STEP is developed and maintained by the ISO technical committee. [7], [8]

UML (Unified Modeling Language) in the field of software engineering the UML is a standardized specification language for object modeling. UML is a general-purpose modeling language that includes a graphical notation used to create an abstract model of a system, referred to as a UML model. UML is not restricted to modeling software. UML is also used for business process modeling, systems engineering modeling, and representing organizational structures. UML is extensible, offering the following mechanisms for customization: profiles and stereotype. The semantics of extension by profiles has been improved with the UML 2.0 major revision. [9]

UML is not a method by itself; however, it was designed to be compatible with the leading object-oriented software development methods of its time.

Although UML is a widely recognized and used modeling standard, it is frequently criticized for the following deficiencies:

- Language bloat. UML is often criticized as being gratuitously large and complex. It contains many diagrams and constructs that are redundant or infrequently used. This criticism is more frequently directed at UML 2.0 than UML 1.0, since newer revisions include more design-by-committee compromises.
- Imprecise semantics. Since UML is specified by a combination of itself (abstract syntax), OCL (well-formedness rules) and English (detailed semantic), it lacks the rigor of a language precisely defined using formal language techniques. In some cases, the UML abstract syntax, OCL and English are contradictory, in other cases they are incomplete. The imprecision of the UML specification has consequences for users and tool vendors alike, such as causing incompatibilities among tools due to unique interpretations of specs.

After this description about some important concepts we decide to use in this project the UML (Unified Modeling Language) model.

UML – Unify Modeling Language

For many years, the practice of software development was exempt from many of these modeling issues. This is simple to understand; the software in the past was easily created and easily changed. Today, software systems have become very complex. While professional architects might build a doghouse without a design diagram, they would never construct complex building without first develop an array of architectural plans, diagrams and some type of a mock-up for visualization. For this reason, developers need a better understanding of what they are building, and modeling offers and effective way to do that. At the same time, modeling must not slow things down. Customers and business users still expect software to be delivered on time and to perform as expected on demand.

Despite the many reasons and virtues behind modeling, a great majority of software developers still do not employ any form of abstraction higher than that of source code. Because, sometimes the actual complexity of the problem or solution does not warrant it. In other cases, developers choose not to model because they simply do not perceive a need for it until much too late.

The software industry has adopted the Unified Modeling Language as its standard means for representing software models and related artifacts. Software architects, designers and developers use UML for specifying, visualizing, constructing and documenting all aspects of a software system. Key leaders from IBM Rational led the original development of UML. Today, UML is managed by the Object Management Group (OMG), which consists of representatives throughout the world to help ensure that the specification continues to meet the dynamic needs of the software community. Adopting a standard notation such as UML is an important step in taking a model-driven approach to software development.

UML is more than just a graphical notation standard - it is a modeling language. As with all languages, UML defines syntax (both graphical and textual, in this case) and semantics (the underlying meaning of the symbols and text).

As it was mentioned UML has a graphical representation of system's model. The model also contains a "semantic backplane" – documentation such as written use cases that drive the model elements and diagrams.

There are three prominent parts of a system's mode:

- Functional Model : showcases the functionality of the system from the user's Point of View – include use case diagrams
- Object Model: showcases the structure of the system using objects, attributes, operations, and relationships – includes class diagrams
- Dynamic Model – showcases the internal behavior of the system – includes sequence diagrams, activity diagrams and state machine diagrams.

In the present project we pretend to develop the functional model and the object model. [10]

<u>Use cases</u>

In software engineering and system engineering, a use case is a technique for capturing functional requirements of systems and system-of-systems. Each use case provides one or more scenarios that convey how the system should interact with the users called actors to achieve a specific business goal or function. Use case actors may be end users or others systems. Use cases are separate and distinct from UML use case diagrams, which allow one to abstractly work with groups of use cases. The main objective of the Use case is to be understood by someone not understands informatics language. [11]

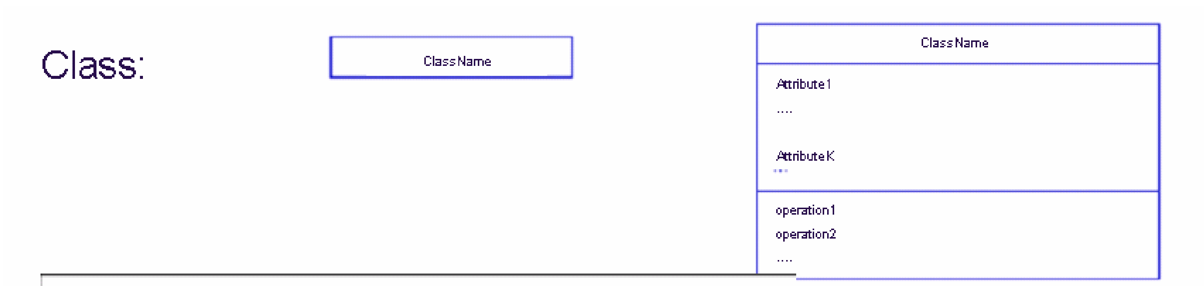
<u>Class diagram</u>

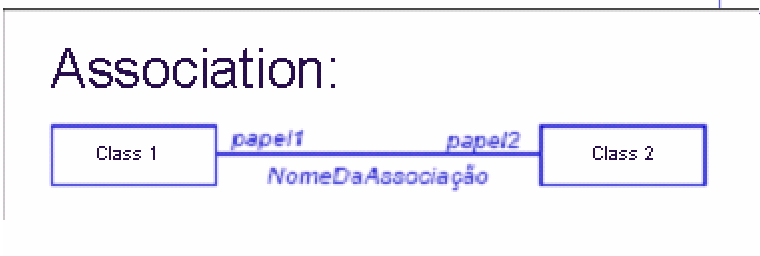
Class diagram is a type of static structure diagram that describes the structure of system by showing the system's classes, their attributes, and the relationships between the classes.

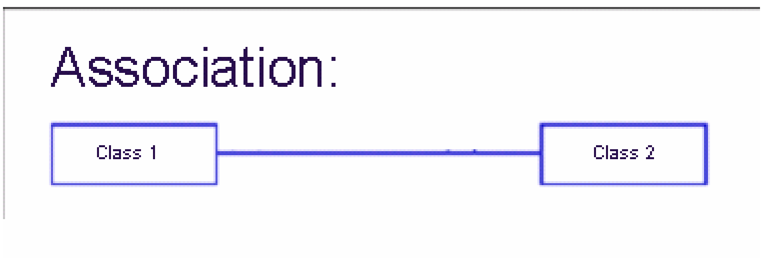
A class in the software system is represented by a box with the name of the class written inside it. Classes may also be used to represent domain or other non-software elements. An optional comportment below the class name can show the class's attributes. Each attribute is shown with at least its name, and optionally with its type, initial value, and other properties [12].

<u>UML Convention</u> [13]

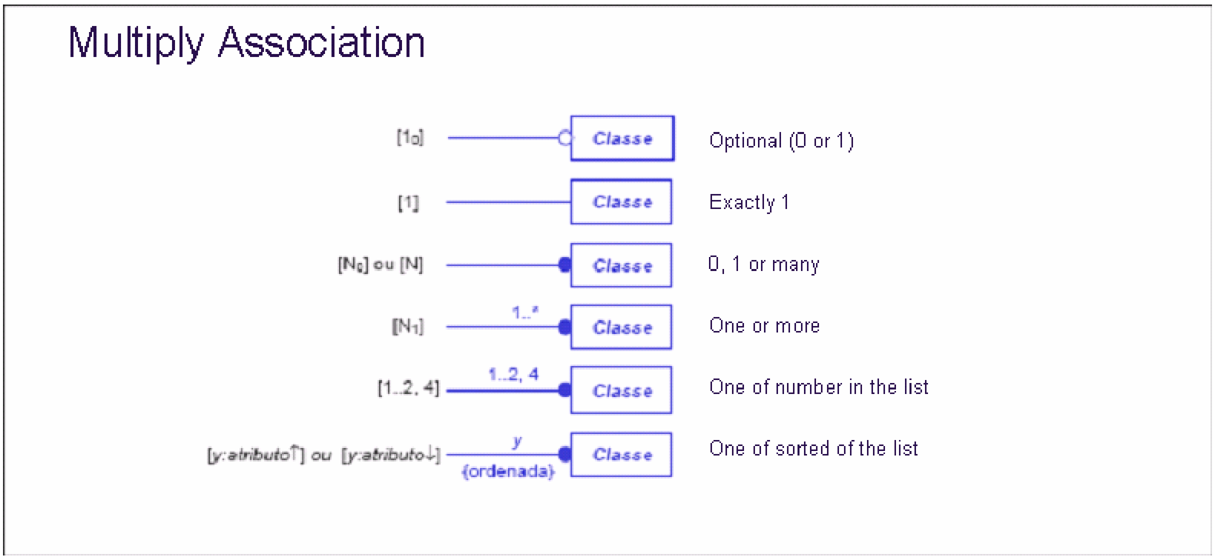
To understand the Class diagrams it is need to understand first the UML Convention.



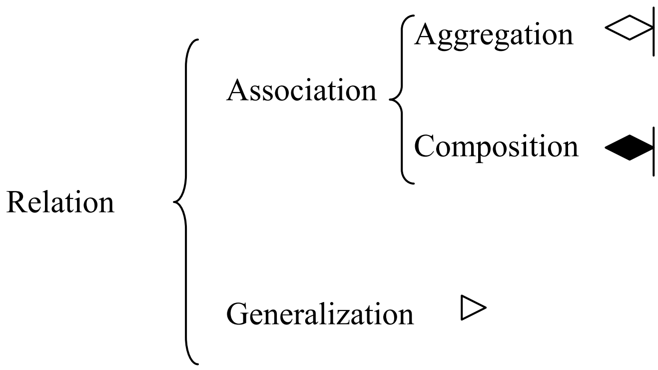




In class diagrams it was use a convention to demonstrate the relation between the classes. The figure below demonstrates the different kinds of association:



There are also different kinds of relations between the classes: [14]



APPENDIX C: Equipment Time Categories

Qimonda AG has hundreds of different equipments and all of them have different time categories. What we pretend to explain in this document is the different times categories that Qimonda use for its equipments.

The figure down helps to understand.

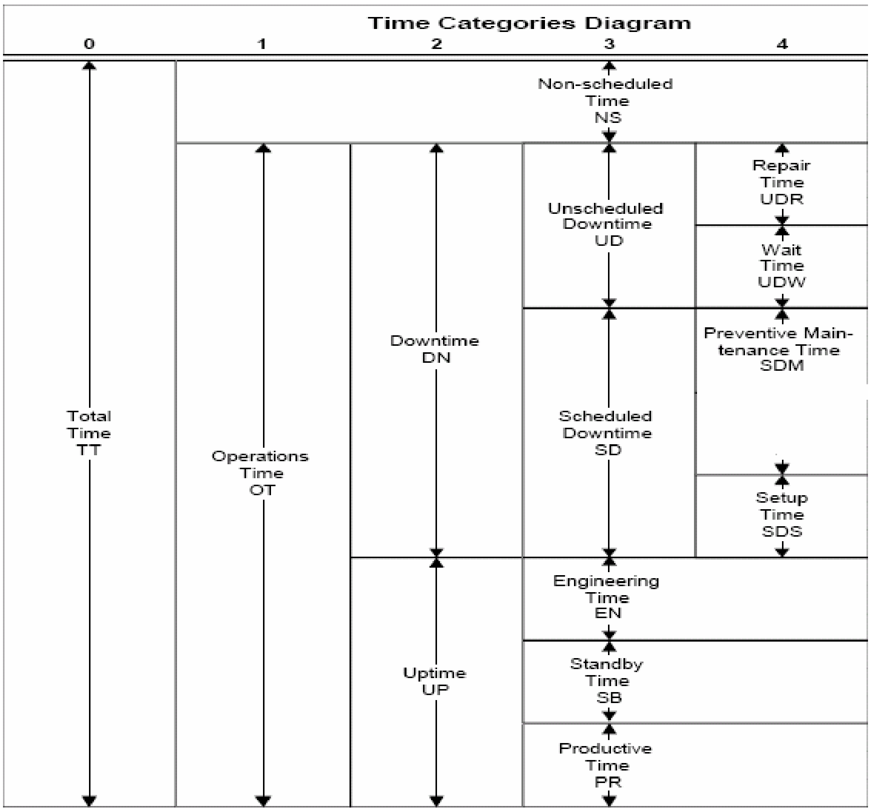


Figure 26: Qimonda Equipment Time Categories

Total Time (TT): is the total time in hours [h] during which a production machine is observed, e.g. 24hours a day or 168 hours per week.

Non-scheduled Time (NS): is a period of time in [h] or the share of TT in [%], when the equipment is not scheduled to be utilized in production. In Qimonda Portugal is use to program a % time per week to do NS according to averages of others weeks.

Operation Time (OY): is the time in [h] or the share of TT in [%], for which it is planned to operate a production machine. It is a total minus non-scheduled time

Downtime (DN) : is a period of time in [h] or the share of OT in [%], when the equipment is not in a condition to perform its intended function. It does not include any portion of non-scheduled time.

Unscheduled Downtime (UD): is a period of time in [h] or a share of OT in [%], when the equipment is not in a condition to perform its intended function due to unplanned down events. Such as technical failures, unplanned measures to secure operation, unplanned shut down of supply infrastructure or due to input materials, which are outside of the specification.

Scheduled Downtime (SD): is a period of time in [h] or the share of OT in [%], when the equipment is not available to perform its intended function due to planned downtime events. The scheduled downtime state includes: maintenance delay, production test, preventive maintenance, change of consumables/chemicals, setup, facilities related to downtime (infrastructure).

Uptime (UP): is a period of time in [h] or the share of Operations Time in [%], when the equipment is in a condition to perform its intended function. It includes productive, standby, and engineering time, and does not include any portion of non-scheduled time.

Engineering Time (EN): is a period of time in [h] or the share of OT [%], when the equipment is in a condition to perform its intended function (no equipment or process problem exist), but is operated to conduct engineering experiments. The engineering state includes: process engineering (e.g. process characterization), equipment engineering (e.g. evaluation of equipment and components of equipment).

Standby Time (SB): is a period of time in [h] or the share of OT in [%], other than non-scheduled time, when the equipment it in a condition to perform its intended function, chemicals, and facilities are available, but it is not operated. The standby state includes: no product or lot is available, no support tools, no operator is available, no input from information systems.

Productive Time PR: is a period of time in [h] or the share of OT in [%], when the equipment is performing its intended function. The productive state includes: regular production (including loading and unloading), work for third parties, rework, engineering done in conjunction with production; may or may not be product units (e.g., split lots and new applications), random samples for quality assurance.

After this description we can realize that the difference between the operation time and the productive time is significant. Thus, as we (capacity planners) can not minimize the engineering and schedule maintenance time because it is not our responsibility we need to minimize the setup time.

Setup time is significant and can be reduce with a good planner. One way to reduce it is minimize change of tools in the equipments. Although this is easy to understand to construct this is very complex. Equipment allocation is a delicate issue and involves a lot of subjects.

APPENDIX D: Consulted Data Models

OMG – Object Management Group

The Object Management Group (OMG) is the world's largest software consortium with an international membership of vendors, developers, and end users. Established in 1989, its mission is to help computer users solving enterprise integration problems by supplying open, vendor-neutral portability, interoperability and reusability specifications based on Model Driven Architecture (MDA).

Business Motivation Model (BMM)

The Business Motivation Model specification provides a scheme or structure for developing, communicating, and managing business plans in an organized manner. Specifically, the Business Motivation Model does all of the following:

- It identifies factors that motivate the establishing of business plans.
- It identifies and defines the elements of business plans.
- It indicates how all these factors and elements inter-relate.

Among these elements are those that provide governance for and guidance to the business — Business Policies and Business Rules.

Business Process Modeling Notation (BPMN)

Business people are very comfortable with visualizing business processes in a flow-chart format. There are thousands of business analysts studying the way companies work and defining business processes with simple flow charts. This creates a technical gap between the format of the initial design of business processes and the format of the languages, such as BPEL4WS, that will execute these business processes. This gap needs to be bridged with a formal mechanism that maps the appropriate visualization of the business processes (a notation) to the appropriate execution format (a BPM execution language) for these business processes.

Inter-operation of business processes at the human level, rather than the software engine level, can be solved with standardization of the Business Process Modeling Notation (BPMN). BPMN provides a Business Process Diagram (BPD), which is a Diagram designed for use by the people who design and manage business processes. BPMN also provides a formal mapping to an execution language of BPM Systems (BPEL4WS). Thus, BPMN would provide a standard visualization mechanism for business processes defined in an execution optimized business process language. [19]

Product Data Management (PDM) Enablers

A Product Data Management system (PDM) is a software tool that manages engineering information, supports management of product configurations, and supports management of the product engineering process. The engineering information includes both database objects and “document” objects – sets of information stored in files that are opaque to the PDM system. This information may be associated with specific products or specific product designs, or more generally with product families, production processes or the engineering process itself. The engineering process support usually includes workflow management and concepts of engineering change and notification. In many manufacturing organizations, the PDM is the central engineering information repository for product development activities.

This specification is intended to provide standard interfaces to Product Data Management systems, or other systems providing similar services, from other manufacturing software systems, primarily those supporting various aspects of product and process engineering, and those supporting manufacturing planning. [20]