Norges teknisk-naturvitenskapelige universitet
NTNU

Fakultet for ingeniørvitenskap og teknologi
Institutt for bygg, anlegg og transport
Studieretning Geomatikk

Postadresse
Høgskoleringen 7a
7491 Trondheim
Telefon 73 59 45 80
Telefax 73 59 70 21

MASTEROPPGAVE

VÅREN 2005
For

Stud. techn. Jon Moe

Bevaring av egenskapsdata ved oppgradering av geometri i 3D-modeller

Conservation of attribute data during geometric upgrading of 3D models

Bakgrunn

De fleste land- og offshoreanlegg har i dag 3D-modeller (DAK-modeller) som inneholder store mengder informasjon utenom den rent geometriske, slik som tag-nummer, linjenummer, tilstandsdata med mer. Disse modellene blir sjelden eller aldri oppdatert når geometrien i anlegget blir endret på grunn av senere installasjoner, reparasjoner eller lignende.

Ved hjelp av laserskanner kan en etablere en 3D-modell med korrekt geometri. Denne modellen vil imidlertid mangle all ekstrainformasjon (egenskapsdata) som ligger i den opprinnelige modellen. En vil få fram et mer verdifullt produkt dersom en kan kombinere den nøyaktige 3D-modellen fra laserskanning med informasjon som ligger i de opprinnelige DAK-modellene.

Oppgave

Kandidaten skal kartlegge egenskaper ved eksisterende datamodeller (DAK-modeller) for installasjoner på industrianlegg. Utvalg av anlegg og konkrete modeller foretas i samråd med veileder.

På grunnlag av denne kartleggingen, skal det utarbeides metoder for overføring av informasjon fra eksisterende til nye datamodeller uten at verdifull informasjon går tapt. Videre skal det foreslås en arbeidsflyt for effektiv dataoverføring med god kvalitetskontroll.

Arbeidet skal foregå i samarbeid med firma Hi-Cad AS i Stavanger, som stiller til rådighet arbeidsplass med nødvendig utstyr, programvare og relevante data for oppgaven.

Generelt

Oppgaveteksten er ment som en ramme for kandidatens arbeid. Justeringer vil kunne skje underveis. Eventuelle justeringer må skje i samråd med veileder og faglærer ved instituttet (samt med ekstern samarbeidspartner der dette er aktuelt).

Ved bedømmelsen legges det vekt på grundighet i bearbeidingen og selvstendighet i vurderinger og konklusjoner, samt at framstillingen er velredigert, klar, entydig og ryddig uten å være unødig voluminøs.

Når det gjelder utforming og innhold i rapporter generelt, vises til «Råd og retningslinjer for rapportskriving ved prosjektarbeid og masteroppgave ved Institutt for bygg, anlegg og transport».
(http://www.ivt.ntnu.no/bat/undervisning/rapportveiledning.pdf)

Besvarelsen skal dessuten inneholde
- standard rapportforside
- tittelside med ekstrakt og stikkord,
  (mal til forside og tittelside finnes på instituttetets hjemmesider)
- utlevert oppgavetekst

I tillegg til selve rapporten skal det innleveres en kortfattet (tilsv. 1-2 A4-sider inkl. evt. illustrasjoner) populærvitenskapelig oppsummering av arbeidet, beregnet for publisering på internettet. Denne oppsummeringen leveres elektronisk i form av html-fil og bør redegjøre for hensikten med arbeidet og for gjennomføringen og de vesentligste resultater og konklusjoner av arbeidet. Oppsummeringen innpasses i html-mal gitt av instituttet.

Dokumentasjon som med instituttets støtte er samlet inn under arbeidet med oppgaven skal leveres inn sammen med besvarelsen.

Instituttet vil ha full rett til å bruke resultatene av arbeidet, som om det var utført av en ansatt under den ordinære arbeidsbelastning, såfremt annet ikke er avtalt på forhånd. Bruk av resultatene til publisering etc. kan bare skje i samarbeid med og etter avtale med faglærer og student, og eventuelt ekstern samarbeidspartner.

Innlevering:
Arbeidet med oppgaven starter 14.01.2005.
Besvarelsen i digital form og to innbundne kopier, samt html-oppsummering, skal leveres innen 10.06.2005.

Veileder ved instituttet: Professor Knut Ragnar Holm
Veileder(eller kontaktperson) hos ekstern samarbeidspartner: Håvard Sande

HMS:
NTNU legger stor vekt på sikkerheten til den enkelte arbeidstaker og student. Den enkeltes sikkerhet skal komme i første rekke og ingen skal ta unødige sjanser for å få gjennomført arbeidet.

Med tanke på eventuell deltagelse i feltarbeid, tokt, befaring, feltkurs eller ekskursjoner i forbindelse med masteroppgaven, skal studenten sette seg inn i *Retningslinje for HMS ved feltarbeid og -kurs, tokt, befaringer og ekskursjoner m.m.*(Kapittel 7.13 i NTNUs HMS-Håndbok) og brosjyren *Helse, miljø og sikkerhet ved feltarbeid m.m. ved NTNU*. Begge dokumentene er tilgjengelig på fakultetets HMS-sider på internett (http://www.ivt.ntnu.no/adm/hms/).

Studenten bekrefter med sin signatur å ha lest *Retningslinje for HMS ved feltarbeid og -kurs, tokt, befaringer og ekskursjoner*, samt å ha mottatt brosjyren *Helse, miljø og sikkerhet ved feltarbeid m.m. ved NTNU.*

Instituttet anbefaler at studenten tegner egen ulykkesforsikring. Velferdstinget og SiT har forhandlet frem en forsikringsordning spesielt tilpasset studenter (se fakultetets HMS-sider: http://www.ivt.ntnu.no/adm/hms/).

Dersom studenten i arbeidet med masteroppgaven skal delta i feltarbeid, tokt, befaring, feltkurs eller ekskursjoner, skal NTNUs HMS-regler følges. Studenten bekrefter dette med sin signatur.

Institutt for bygg, anlegg og transport
Dato: 14.01.2005

Dato: 14.01.2005

Knut Ragnar Holm
faglærer

Jon Moe
Stud.techn.

Preface

This master thesis is written at the Division of Geomatics at Department of Civil and Transport Engineering, during the spring of 2005. The thesis spans over special fields ranging from 3D-graphics and databases to surveying and laser-scanning. With a background from both the Department of Computer and Information Science and the Division of Geomatics, I feel the thesis cover a wide range of the subjects from my study.

The work on the thesis has been challenging, but rewarding and I have learned a lot about how to acquire the information needed when it is not easily available. Finding relevant research on the subject has not been easy. The only available information on many of the systems I have studied is what the companies chose to present for advertisement-purposes. Unfortunately this advertisement-information differs in many cases from what the software actually can do and what I have needed.

I want to thank all at Capnor and Håvard Sande in particular for help, support and suggestions, Kari Skårdal at Sørco for providing expertise and help on PDS and Professor Knut Ragnar Holm for guidance and feedback whenever I needed it.

Jon Moe
Trondheim, 10.6.2005

Abstract

Existing models of Norwegian offshore platforms are generally incomplete and lack the accuracy needed to eliminate clashes between the existing parts on the platform and new systems, before construction. Laser scanning is today used to a growing extent to make models with the necessary accuracy and thus resolve clashes prior to construction. However these models only show the surface of the plant and do not have the non-visible attributes or “intelligent information” contained in the existing models of the platform.

I will in this thesis present how the intelligent information from an existing as-built model can be assigned to a scan model, and thus replace the old and inaccurate as-built model used today during the design of new systems.

Table of contents

1 Introduction 1

1.1 Background 1

1.2 Purpose 1

1.3 Structure 1

2 Theory 3

2.1 Plant design 3

2.2 As-built modeling 7

3 Method 17

3.1 The ideal system 17

3.2 Inovx solution 22

3.3 PDS/Cyclone solution 27

4 Implementation 32

4.1 Implementation 32

4.2 Workflow 40

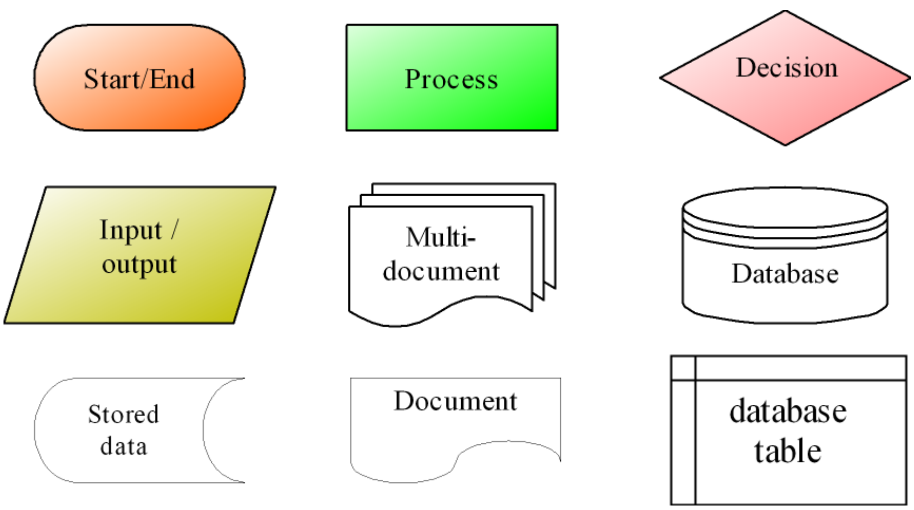
5 Conclusion 45

6 Future work 47

7 Bibliography 49

8 Appendixes 51

Legend for flowcharts



1 Introduction

1.1 Background

Today, most plants in the petrochemical industry utilize 3D-models (CAD) to design, construct and operate the plants and offshore platforms. These CAD models have a lot of information besides the geometrical information such as line-numbers, type of material in the parts, pressure limits etc. These models are rarely updated when alterations such as new installations or repairs are made on the plant.

By using a laser scanner, it is possible to make accurate 3D-models with correct geometry. These models will however lack the extra information which the original models have. Combining this extra information with the accurate models made by laser scanning can result in models with a larger value than the separate models.

1.2 Purpose

The goal of this thesis is to show how intelligent information from existing 3D-models can be assigned to accurate models made by laser scanning, and to suggest a workflow to ensure the quality of the final model is satisfying.

A brief introduction to plant design and as-built modeling is given and different approaches to assigning the intelligent information are discussed. From this discussion an implementation and workflow is suggested and explained. As the solution suggested is not implemented, no tests have been done on the implementation.

1.3 Structure

The thesis starts by explaining plant design and as-built modeling. This is done in the chapters under “Theory”: “Plant design” and “As-built modeling”.

“Method” contains three chapters:

- "The ideal system" which discusses assigning intelligent information in general
- "Inovx solution" which discusses an approach trying to use off-the-shelf software
- "PDS/Cyclone solution" which discusses how the system can be implemented.

An implementation of the system and a workflow is discussed in the chapters under "Implementation" before the conclusion and suggestions for future work are given in separate chapters at the end.

2 Theory

2.1 Plant design

The Merriam-Webster dictionary (Merriam-Webster online dictionary, n.d.) defines a plant as:

> "the land, buildings, machinery, apparatus, and fixtures employed in carrying on a trade or an industrial business"

Throughout the 20th century, engineering and design of plants in the oil and power industries meant manual 2D drafting. Later on it evolved to include the use of expensive plastic models. Over the last 20 years manual drafting has given way to 2D modeling on computers. More recently the 2D models have been replaced with 3D models on all large projects. (Jones, 2003)

Figure 2-1 an offshore platform

With the switch to 3D Computer Aided Drafting (CAD), 2D reports in the form of orthographic or isometric drawings[1] could be generated automatically from the CAD

[1] **Isometric projection** is an axonometric projection. A 3D-object is represented in 2D with an angle of 120° between the projected x, y and z axes. This corresponds to rotating the object by +/- 45° about the vertical axis, followed by rotation of approximately +/- 35.264° = arcsin(tan(30°)) about the horizontal axis, starting from an orthographic projection view. (Wikipedia, May 23, 2005)

models. The model could be checked for interferences at the design stage, reducing clashes during construction, and material take-offs could be generated early in the project to begin purchasing activities.

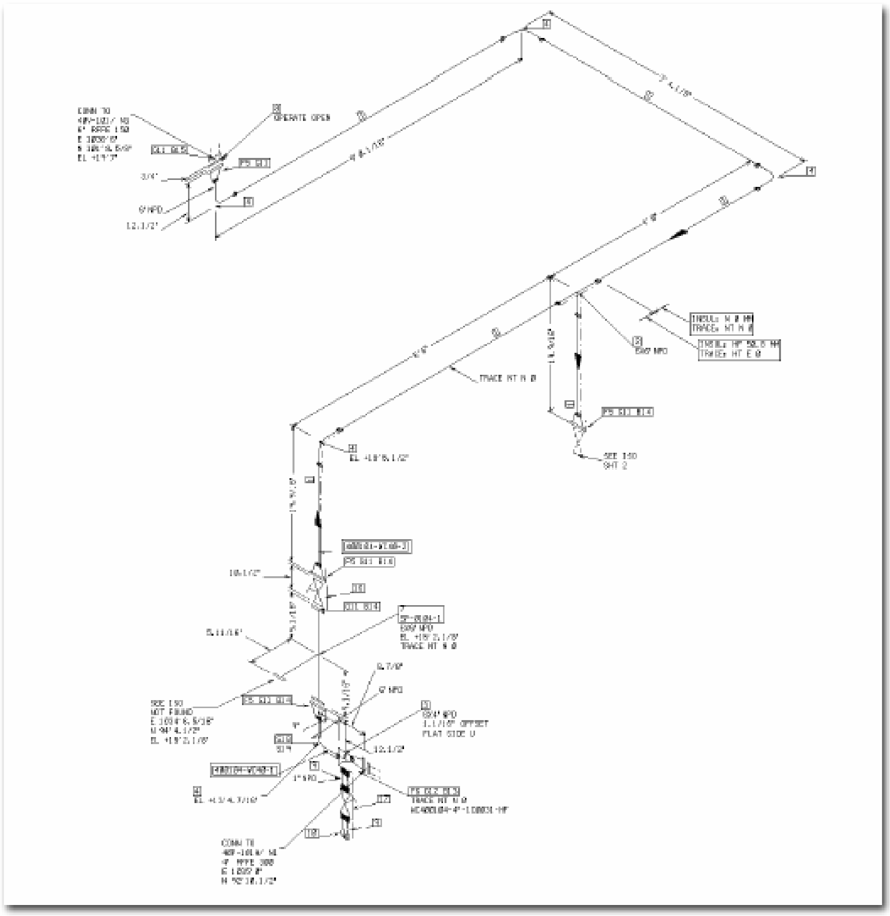


Figure 2-2 iso drawing of a pipeline

Some of the most common CAD-systems today are Autodesk’s AutoCAD (Autodesk, 2005) and Bentley’s Microstation (Bentley, 2005).

Besides the geometric properties of the model, a lot of information is needed about the plant at the design stage. This information range from information about the project, and what constrains are set for the project, to reference information telling the specific attributes for each part. This information is called intelligent information.

2.1.1 Specification driven design

When the plants are made it is common practice to use components that comply with international, national and local standards and catalogues. These catalogues define and control the parts that make up the plant and ensure that they conform to the specifications. Because most parts of the plant are made of standard components we can utilize this when the plant is designed. By defining all the components available in the design application, the plant can be designed of pre-defined “building blocks”. This is what is called a specification-driven design. When the engineer builds the model he builds it from a library of pre-defined objects rather than modeling each object manually each time he needs it.

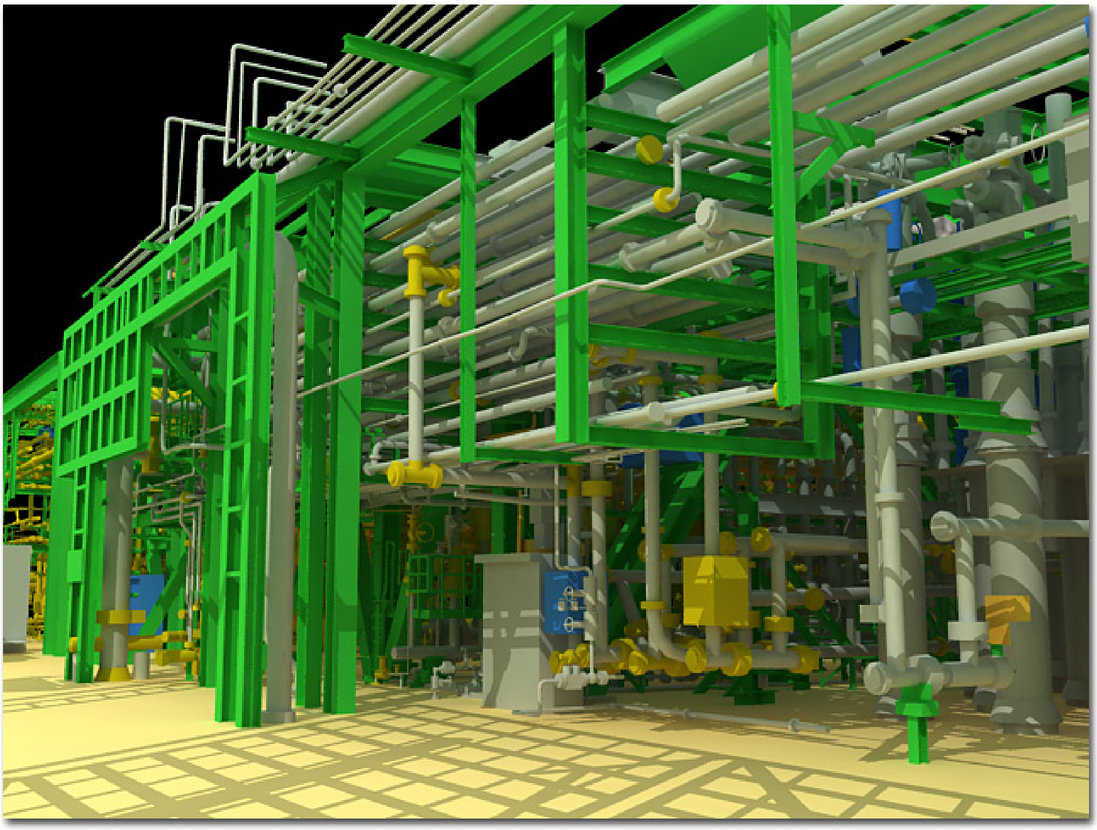


Figure 2-3 an accurate 3D model of a plant (Statfjord B)

Specification-driven design can loosely be compared to building with a well assorted set of LEGO® while normal modeling is more like building the model by cutting, gluing and forming cardboard. The LEGO® approach is less labor intensive, but can only model with standard parts. Specification-driven design systems have underlying design rules that ensure that the specifications are met. Normal modeling however is more labor intensive, but it also has a greater freedom of how the objects are modeled. The drawback is that the final model not necessarily complies with project standards and engineering practices.

The largest providers of specification driven plant design systems in the Norwegian offshore industry are Aveva (2005) with PDMS (Plant Design Management System) and Intergraph (n.d.) with PDS (Plant Design System) and 3D-Smartplant. Both PDS and PDMS have been around since the mid-eighties. (Daratech, 1994)

2.1.2 Intergraph PDS

PDS has evolved from a system based on VAX (Wikipedia, 2005) to a system running on Microsoft Windows® and it is compatible with most popular relational database management systems. Microstation is used as an underlying application that provides CAD functionality. Models are made as Microstation models with a link to the PDS database. Each of these models is divided into model-files, which can only be manipulated by one operator at a time.

2.1.3 Aveva PDMS

While PDS is model-centric PDMS is data-centric. PDMS stores all information into one database. Geometric information which in PDS is stored as a Microstation file, is in PDMS in the same database as the rest of the information. This provides better collaboration and interaction possibilities than the model-centric PDS. The database can be replicated across multiple locations. Whenever a change is made at one location, the change is distributed to the other locations in real time, providing everyone working on the model with up-to-date models. The project database can be divided into partitions that reflect how the operators are divided on different locations. Control, access and the partitions can be changed as changes are needed in the project.

The drawback is that PDMS uses its own proprietary database format. Any interaction with the PDMS database from third part applications requires expensive additional software.

2.1.4 Intergraph SmartPlant 3D

SmartPlant® 3D is Intergraphs new data-centric plant design system. It has the same functionality as PDMS, but has a closer integration with PDS. The move from a model-centric system to a data-centric system reflects how the focus has changed the last decades. From a focus on quickly achieving a model, the focus now is on ease of management and collaboration possibilities across multiple locations.

2.2 As-built modeling

The term "as-built documentation" is a term used for documentation of human-made structures after they have been constructed, as opposed to the design plans made before construction. Structures are never constructed exactly as they are planned. This isn't necessarily a problem. For instance on petrochemical plants the focus obviously is on function. As long as the pipes are fit together and work properly, a couple of millimeters difference is unimportant. However this can be a problem when fitting new systems to an existing plant if the differences are too big, or if parts are missing in the as-built model. Differences as small as a few millimeters, can lead to a serious clash in the model. Such clashes results in a rework of the design plans if the error is found before construction. If the clashes are not found before during construction, the parts must be reconstructed to resolve the clash. This often results in a halt in the construction dependent on this part to be fitted first. Adding to the expensive time where the plant is shut down and out of production.

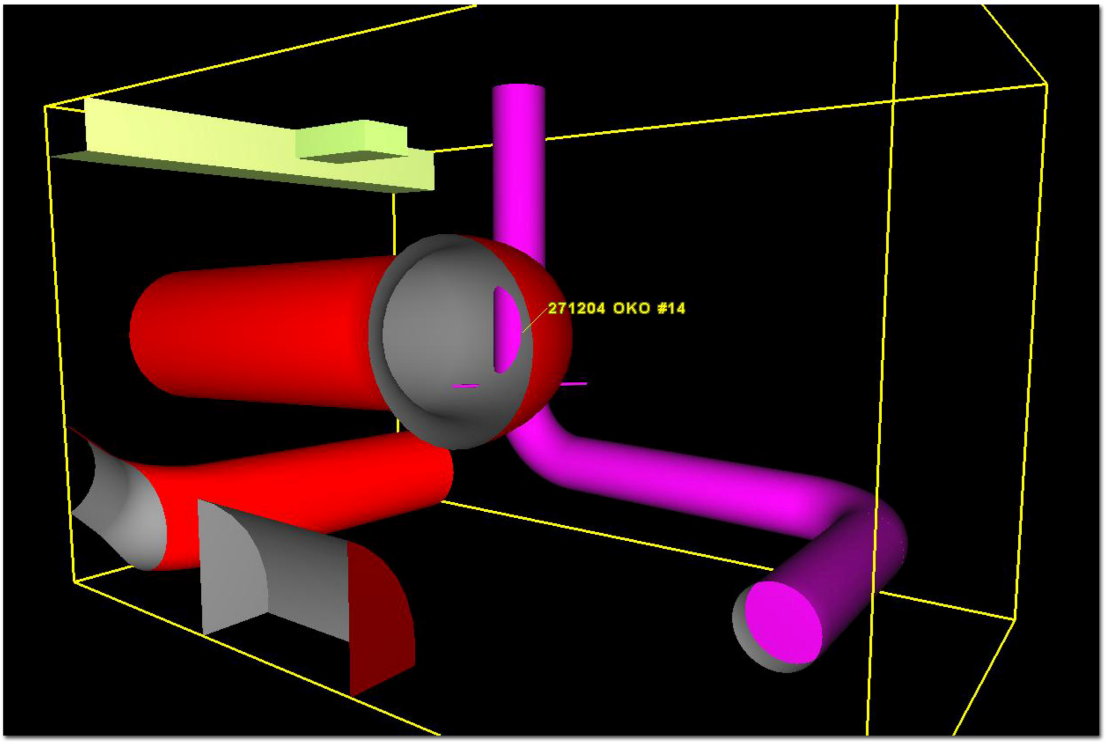


Figure 2-4: A clash between a designed pipe and existing piping (marked with yellow text in the model).

Up to the recent couple of years, as-built models of Norwegian offshore platforms have been made using traditional survey instruments such as total stations. The problem with these models is that they are only as densely modeled as the points the surveyor chooses to measure. If a pipe is not measured by the surveyor, the pipe is not in the as-built model. To make a complete accurate model of a large and complex offshore platform is meticulous work and many offshore companies have several surveyors working on permanent basis to make the measurements needed for the as-built models.

Figure 2-5 Leica total station TPS110C

2.2.1 Photogrammetry

Photogrammetric techniques have been used with good results in some projects. Using photogrammetry requires the operator to mark relevant features in a number of pictures. An accuracy of 0.25 mm or better can be acquired by photogrammetric measurements (Atkinson, 2001 p. 265). When projects are small the workload using photogrammetric methods are manageable, but as the projects grow in size and complexity the workload can become unwieldy.

Some software companies such as INOVx (2003) have made solutions which combine photogrammetric techniques with laser scanning. This way the potentially better accuracy from photogrammetric techniques can be used at important areas while the massive amount of points generated from a laser scanner can be used to model the rest of the areas.

2.2.2 Laser scanning

Laser scanners use a laser beam to measure the distance from the scanner to the surface it hits. By measuring the horizontal and vertical angle (α and β in figure 2-7) it is possible to get the position in space for the spot on the surface where the laser hits. When the beam is moved through the field of view the system creates a dense point-cloud of the scanned objects surface.

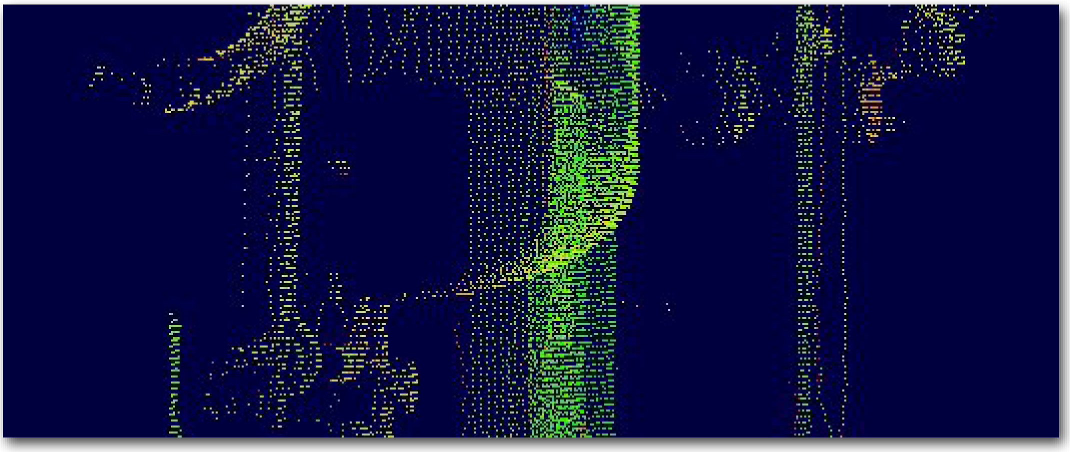


Figure 2-6 point cloud from a laser scan

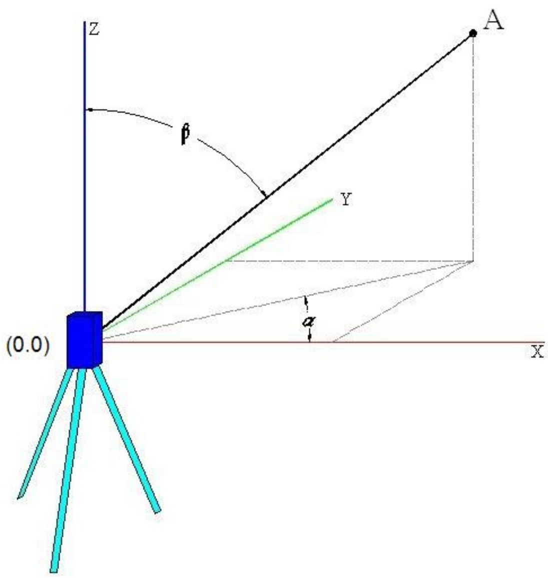


Figure 2-7: The angles α and β are together with the length between the laser scanner and the reflected point A used to calculate the position of A.

Today there are mainly three different laser scanning technologies in use: Time of flight scanners, phase shift scanners and triangulation scanners.

2.2.3 Time of flight scanner

A time of flight scanner measures the time it takes from a laser pulse is sent from the scanner to it gets reflected from a surface back to the scanner. As the speed of light is constant the distance to the surface is calculated from the formula:

$$d = \frac{c \cdot t}{2} \quad (1)$$

Where d is the distance, t is the time and c is the speed of light.

Optimal effective range[2]	1-100 m
Accuracy, position[3]	6 mm
Accuracy, distance[4]	4 mm
Scan speed (max)	1800 points/sec

Specifications for HDS3000 time-of-flight scanner (Leica, 2004)

Figure 2-8: Leica HDS3000, time-of-flight laser scanner

[2] Optimal effective range is defined by the distance where the reflected signal has decreased to 10% of the sent signal.

[3] The positional accuracy is for the position calculated from the angles in figure 2-7 at 50 m.

[4] The accuracy in distance, calculated from the formula (1) to a surface 1-50 m away.

2.2.4 Phase shift scanner

Phase shift scanners utilize a continuous laser beam rather than the pulsing laser beam used in time of flight scanners. This allows a much faster sampling speed with typical scan speeds ranging from 100,000 to more than 500,000 points/sec. In order to measure the distance the beam is modulated with a reference wave. The range is then calculated by measuring the difference in modulation between the emitted beam and the reflected beam. Both amplitude modulation and frequency modulation can be used as a modulation-method.

Optimal effective range	1m – 25m
Accuracy, position[5]	7.5mm
Accuracy, distance[6]	5mm +240ppm
Scan speed (max)	500,000 points/sec

Specifications for HDS 4500 53m range model phase shift scanner (Leica, 2004)

Figure 2-9: Leica HDS4500 phase shift scanner

[5] The positional accuracy is for the position calculated from the angles in figure 2-7 at 10 m.

[6] The ppm value equals the range noise standard deviation.

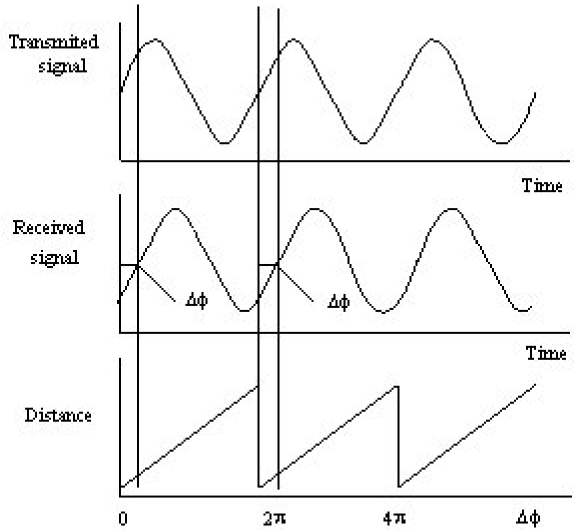


Figure 2-10: The difference in the modulated wave between the transmitted signal and the received signal is used to calculate the distance (using amplitude modulation in this example)

2.2.5 Triangulation scanner

Triangulation is a system where a light-beam is projected onto the object surface. One or more CCD-cameras are used to record the position where the light hits the surface. The angle of the light leaving the laser is internally recorded, and the fixed base-length between the camera and the laser is known from calibration (figure 2-11). From this measured position it is then possible to calculate the height of the object. These methods are however only available for relatively small objects, as the accuracy is highly dependent on the distance between the light emitter and the CCD-camera. The accuracy of the points measured is better than the laser scanning techniques mentioned earlier, with an accuracy defined with a standard deviation of 0.3 mm at 2 m and 0.6 mm at 5 m (Boehler, Heinz & Marbs, 2001)

Triangulation is a method developed mainly for the design and manufacture of cars and human scanning and is thus mainly produced in sizes fit to scan such objects. The technique has been used with success in cultural heritage recording to measure the surfaces of sculptures and architectural surfaces. (Boehler, Heinz & Marbs, 2001) Scans conducted on architectural surfaces have however had a focus on recording the details in the surface of specific walls for cultural documentation rather than documenting the structure of the whole “building”, which is the main focus when scanning large petrochemical plants.

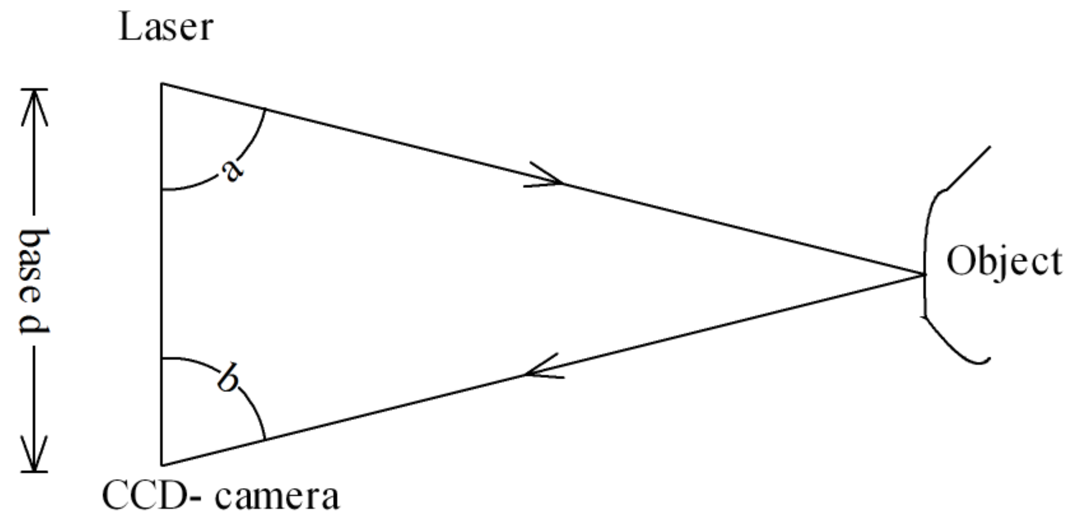


Figure 2-11: The angles a and b, and the base distance d is used to calculate the coordinates on the reflected surface

2.2.6 Using the point clouds

In plant design projects the final result is usually a CAD model representing the real world, but the point cloud or the referenced pictures can be used raw during the design work. Measurements can be made directly in the point cloud, or in the referenced pictures with the appropriate software, and can provide important information for the designer. Even if a point cloud provided as-is during the design makes the correct geometric information available during design, the geometric information is not easily accessed. The designer has to actively measure every distance and position he needs when needed.

An as-built model that would be easier to use, is a complete model made of surfaces or volumes, not a collection of point clouds. To make such a scan model, surfaces of the objects need to be modeled and placed in the point cloud. A models' surface can be generated directly from the point cloud as a mesh, or 3D primitives such as boxes or cylinders may be fitted to represent the point cloud. These 3D primitives can be fitted both automatically by selecting a selection of points, from which the application calculates the best fit for the object, or the objects can be placed manually. When objects are placed manually, the modeler decides the orientation and extent of the object by evaluating the point cloud.

When 3D-primitives are fit to the point cloud the accuracy of the model becomes better than that of each point in the point cloud. As a surface is fitted to the point cloud this surface will interpolate the points and cancel the effect of random variations in the accuracy of each point.

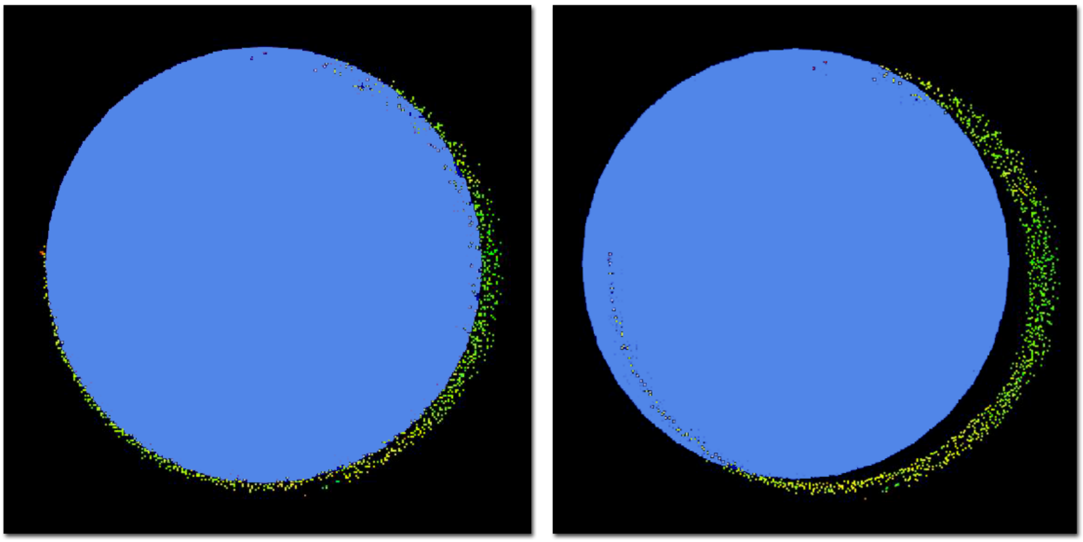


Figure 2-12: manually fitting a pipe to the point cloud

Cyclone (Leica Geosystems, 2005) and 3D-PlantLINx (INNOVx, 2003) are some of the applications purpose-built to produce scan models from point clouds.

A scan model is initially “dumb”, meaning it does not contain any intelligent information. It rather just occupies the volume taken by plant parts. Line-numbers, operating temperatures, pressure limitations and other relevant information, has to be gathered from other systems when designing a plant. Since all these attributes are just as important as, or even more important than the geometry, the designers usually design the plant in a specification-driven plant design system.

The problems arise when this initial model differs from reality. The intelligent attribute information is usually correct, but the geometry of the model will differ in most systems. Pipes tend to be affected by gravity and slope in the middle of long spans. Equipment added after the plant was built might not have been added in the model. An unforeseen error in the plans could have been corrected during assembly and a mistake made by the construction-workers might have been solved by a quick fix. There are many factors like these in real world that can make the initial as-built or design model inaccurate.

The engineers who design new systems need to know what parts the existing plant is made of in order to answer questions like: “How will the new pipes be connected to the existing pipes? Will the support system be able to support the added weight?” Questions like these need to be answered all the time when the engineer is planning the system. When the plant was initially designed all this information was entered into the plant design.

A scan model shows the engineers where the equipment and parts is on a plant and how it looks. This model can not tell the engineers what the parts are and their attributes other than those derived from looking at the surface of the parts. Because of this lack of intelligent information in the scan model, the engineers design the new systems in the existing as-built model based on the initial design models, where they have all the intelligent information available. These initial as-built models may, or may not have been updated after the initial construction of the plant. If serious changes have been made to the design model, the possible conflicts between the real world and the 3D model are resolved at a later stage.

If we could assign the intelligent attribute information from the design models to the accurate scan models, the designers would only need to look at one model to get the correct information. Today the engineers have to evaluate several models from different companies to get all the information needed. Different contractors might have built different systems and made the corresponding design models. These models are not necessarily collected into one model. They rarely have the accuracy needed to resolve possible clashes with confidence and a scan model is needed in addition to the design models.

3 Method

3.1 The ideal system

In this chapter I will discuss in general what approach should be taken to assign intelligent information from a plant design system to a scan model.

3.1.1 Acquiring the intelligent information

Most importantly one should be able to access the data. Both the existing as-built models with their intelligent information and the scan models need to be read. How much information needs to be read from each system depends on how much of the different information is altered when assigning the intelligent information. As a minimum, the system must read the information it needs to alter and the information needed to place the model back to the plant design system it came from.

3.1.2 Acquiring an accurate scan model

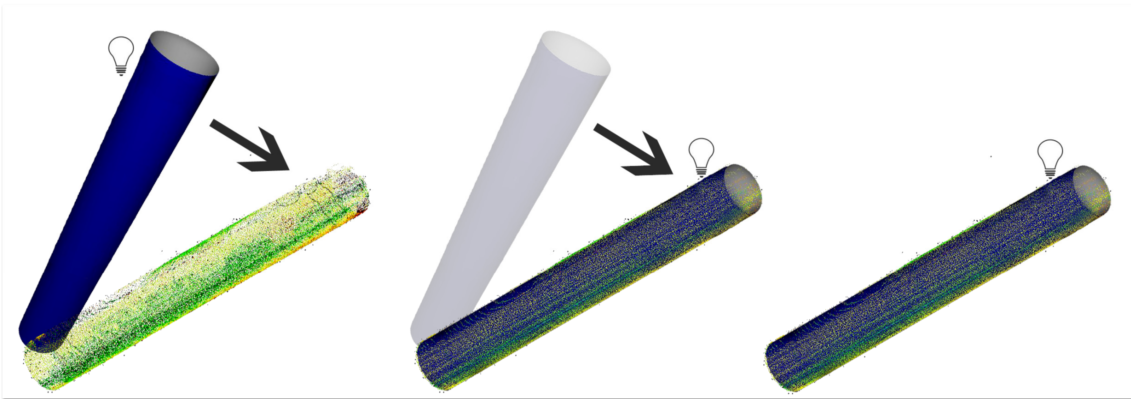


Figure 3-1: Moving a pipe-object with its intelligent information to fit the point cloud. (Intelligent information represented by a light bulb.)

Ideally one could import the models from a plant design system in a model space with matching point clouds and fit the already intelligent objects to the point cloud. Only objects lacking the desired accuracy should be moved. It would not be necessary to assign the intelligent information again after the objects have been moved, as long as the intelligent information could be preserved for each object when they are moved into place. This approach would be good if only some of the objects were misplaced in the model.

The experience however, is that every existing as-built model has errors. (Sande, H. personal communication, May 2005) Most important is missing parts in the model. If the as-built model does not contain all objects present on the plant, the as-built model is close to useless for clash checking. One can not be sure all clashes are found and resolved if there are parts on the plant that are unaccounted for in the model.

Even if the as-built model is complete there are many sources for errors (Sande, H. personal communication, May 2005):

- Errors can be introduced during construction. Or rather changes made to the initial design might not lead to an update in the as-built model. Pre made parts might not fit as well on the construction site as planned in the design model, resulting in ad-hoc solutions to make the parts fit anyway.
- Reference grids are often not defined with the desired accuracy on the platform. When these grids are used to stitch together models from different parts of the plant, the result is errors that vary without control within the model.
- Nature might introduce some errors. Changes in temperature might expand or contract pipes and structure. Gravity will pull down on long stretches of pipe making them slope in the middle.
- Thick layers of paint or insulation will also produce differences in the geometric parameters of an object such as diameter, length and thickness.

Such errors can easily become a problem when fitting a new design to a narrow area as in figure 3-2.

Placing the as-built model into the point cloud and then fitting each object in the as-built model would be a bigger task than to make a scan model from scratch. The modeler must for each object find the corresponding set of points from the point cloud and then fit the object in question to the chosen points. The main difference between the two approaches is how aware the modeler must be of the surrounding points when modeling. To understand what the points in a point cloud represent and model this correctly is not easy and requires a lot of training. At the same time having a good understanding of the points for neighboring pipes and structures is even harder.

If this approach is taken, the operator has to for each object in the model first decide which points in the point cloud matches the object. Then the operator has to see how well the object fits these points and adjust the object so it fits the points with the desired accuracy. Any objects missing in the design model still need to be modeled. Which means the application should provide good modeling possibilities for point clouds.

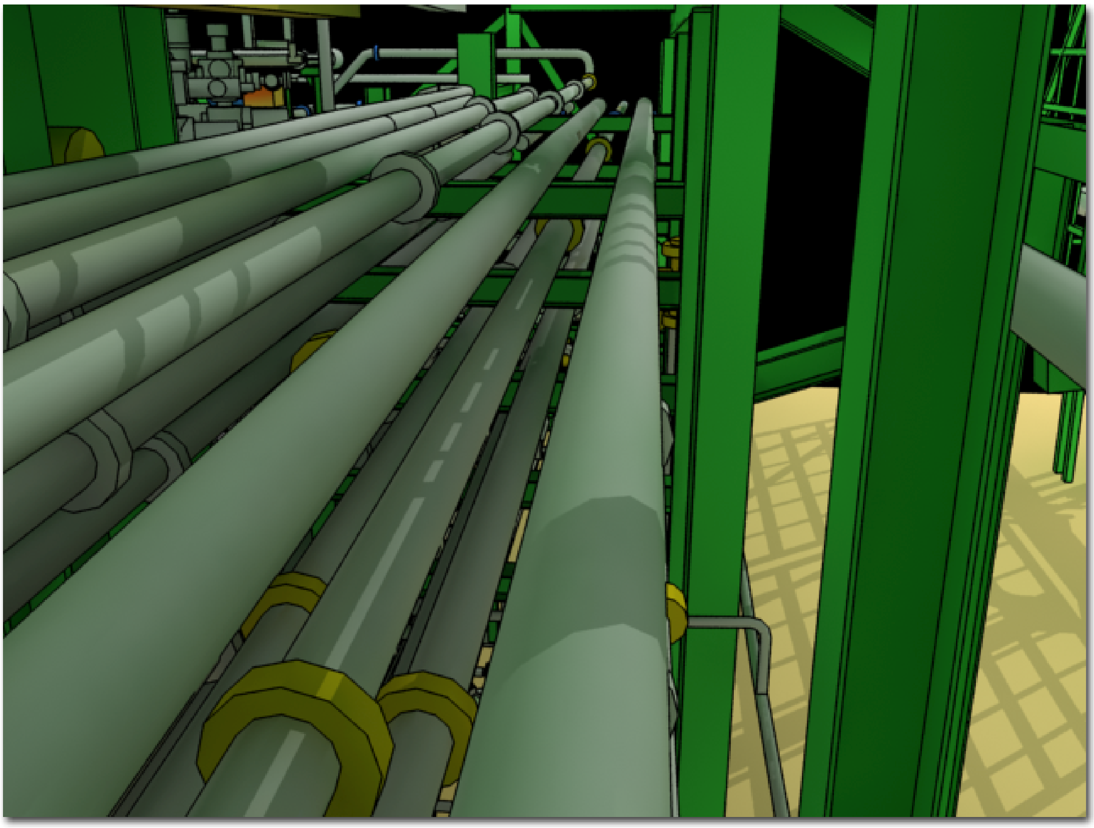


Figure 3-2: Dense pipe-gates make it difficult to choose the correct pipe when assigning intelligence.

Many scan models exist today with the desired accuracy and as the technology matures, more will be made. The following approach utilizes these models. Assigning intelligent information is done on already made scan models, eliminating the requirement for reading point clouds. Having a scan model made beforehand, means the operator doesn't have to deal with the accuracy of the model while assigning intelligent information to the model.

The only requirement for the scan models is that they can be read in the application that assigns the intelligent information. This way the scan models can be made with the modeling software best fit, usually the software in use already. The company does not need to train people to use new modeling tools and no alteration has to be made to the existing workflow of producing scan models.

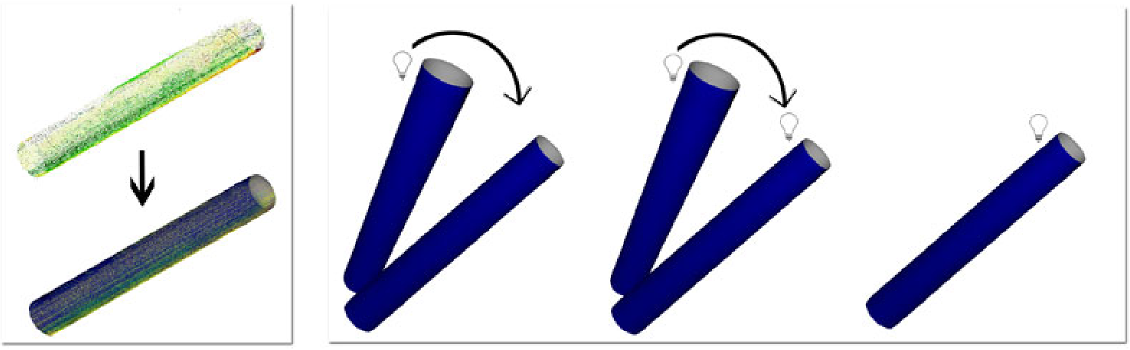


Figure 3-3 making a scan model and adding intelligent information in separate applications

The application that assigns the intelligent data must be able to read the scan models and the models from a plant design system with the intelligent information associated to the model.

When a scanned model of the plant already exists, the application does not have to be able to handle point clouds, which gives a wider range of possible CAD-tools to implement it on. Both AutoCAD and Microstation have a development interface and can be used for this task. They also have a database interface which is needed to work with intelligent data.

3.1.3 Importing intelligent scan models back to the initial system

To make all the work done with assigning intelligent information to the models useful for the engineers, the models need to be imported back to the initial plant design system. The engineers must be able to use the new model as they would use the initial model with all its intelligent information, and be confident that the geometry of the model they work on is correct and complete. All constraints present in the plant design system must be present the same way as they did prior to the upgrading of the models and the same intelligent information must be accessible through the model.

3.1.4 Managing the changes

When intelligent information is assigned from an as-built model to the scan model it is important to keep track of the changes made on the models. Questions such as: "Which models have been processed and which are due? How many and which objects in the scan model have gotten intelligent information assigned?" needs to be answered before the quality of the work can be assessed.

When intelligent information is assigned from the as-built model to the scan model, the status of both objects should be updated. The object in the as-built model should get a status telling that its intelligent information has been assigned to an object in the scan model. Objects in the scan model should likewise get its status updated.

3.2 Inovx solution

In this chapter an approach using off the shelf software to assign intelligent information to the model is discussed and explained.

Inovx has the only software available with both modeling capabilities and the desired attribute manipulation functionality I could find. They deliver a system with four modules:

- 3D-PlantLinx, which is the modeling environment of the system
- RealityLinx which is the model-browser environment
- KnowledgeLinx which is a module that adds attribute and database manipulation possibilities to RealityLinx
- CAD-Linx which is their system for importing and exporting intelligent information from other systems.

Upon request we received a license for 3D-PlantLINx and a license for RealityLINx. The license we received for RealityLINx only provided us with the possibility to import and view models. It did not allow us to manipulate or import any intelligent information. As a result of these limitations I could not evaluate these features, but have relied on documentation, mail correspondence and phone conferences with developers and support at Inovx.

RealityLINx is a viewer with the ability to show CAD models and attributes assigned to these models. To assign the attributes KnowledgeLINx or CAD-LINx is needed. CAD-LINx is not developed as a product for which Inovx provides a license, but is rather a set of internal tools Inovx uses to provide conversion services for its clients. It relies on XMpLant[7] and its XML definitions of plant-objects to translate models with intelligent information between different plant design systems and Inovx' own database format. KnowledgeLINx is an integrated part of

[7] XMpLant is a generic converter for intelligent process plant information made by Noumenon Consulting Ltd. It maps information from one proprietary system to a neutral XML model and then back out to another. Thus, it can convert intelligent plant information between any systems which have an XMpLant interface. (Neumenon, n.d.)

RealityLINx and can be unlocked with a proper key. The main functionality for KnowledgeLINx is to provide the tools needed to add custom attributes to models in RealityLINx.

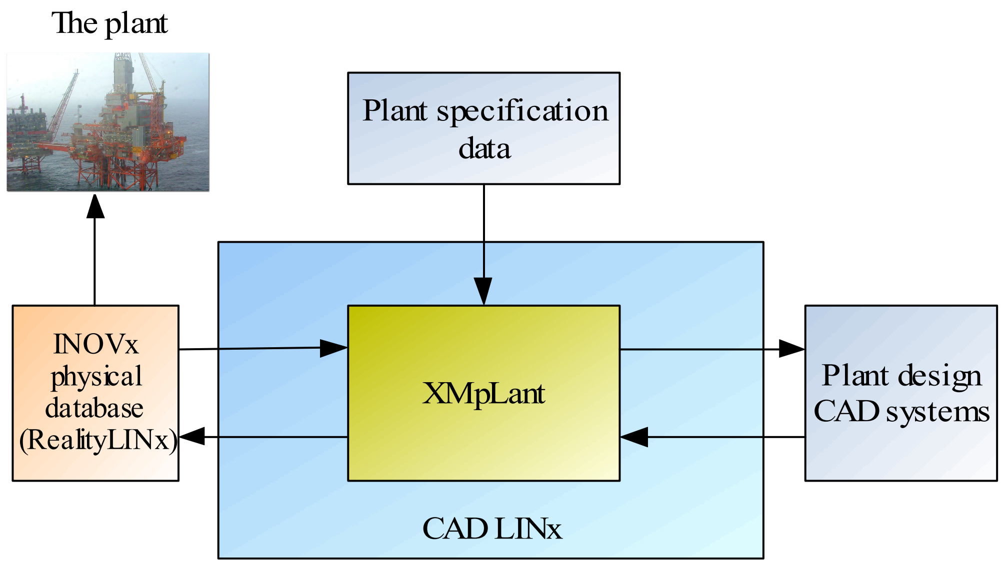


Figure 3-4, how CAD LINx works (Inovx, 2003)

3D-PlantLINx was provided with all features enabled and could be fully evaluated. To get the evaluation as thorough as possible the application was tested both by me and by an experienced modeler at Capnor. She compared it with the CAD and laser scan applications that she had experience with, while I evaluated the possibilities of the application towards intelligent models.

The first impression of the module is that it performs well. It handles large models and large point clouds with ease and has advanced functionality for choosing which objects to show. This is important when working with large plant models and point clouds. When it comes to modeling in the point cloud it relies heavily on assisted modeling.[8] 3D-primitives are fitted to the point cloud automatically after the operator has defined what type of primitive the object is and a selection of points that represent this object. Unfortunately, sometimes this method does not insert the 3D-primitives into the point cloud correctly.

[8] Assisted modeling is when the dimensions and alignment of an object is placed automatically into the point cloud. After selecting one point or a selection of points, the application evaluates these points and surrounding points to decide the best fit between the object and the points.

To evaluate the placement of the 3D-objects in the point cloud, an easy way of deciding how well the object is aligned is needed. In Cyclone this is done by using a limit box that cuts away all points outside the box and slices through all objects that goes through the box. This way it is possible to look inside an object and evaluate how the scan points are distributed outside an object, compared to the inside. The objects' surface should ideally cut through the cloud of points leaving an equal amount of points over and under the surface. By doing so, it is reasonable to believe that the object is well aligned. 3D-PlantLinx lacks this possibility.

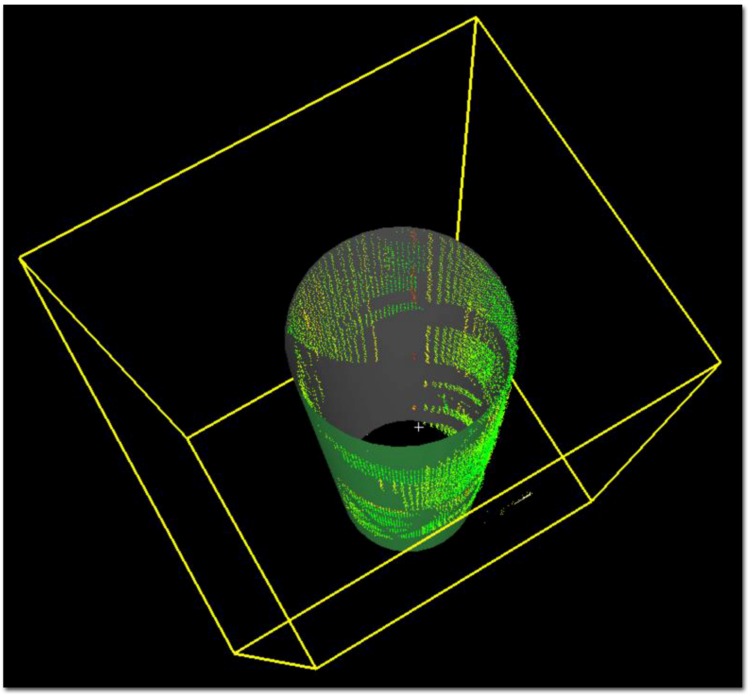


Figure 3-5: Limit-box slicing through pipe and point cloud, showing points on both the inside and the outside of the pipe

The other major problem with aligning objects in 3D-PlantLINx is the absence of a manual manipulation mode. When one are modeling and aligning objects it should be possible to do this in a natural way. That is, one should be able to rotate and move objects by moving the mouse. In 3D-PlantLinx this is done by entering the rotation axis for the rotation and an angle defining how much the object should be rotated. There is however a way to move an object with the mouse. This movement can be restricted to one or two directions as shown in figure 3-7, so this feature works well in 3D-PlantLinx.

Figure 3-6: rotation in 3D-PlantLINx is defined by entering the needed values to define the rotation in this "tool box".

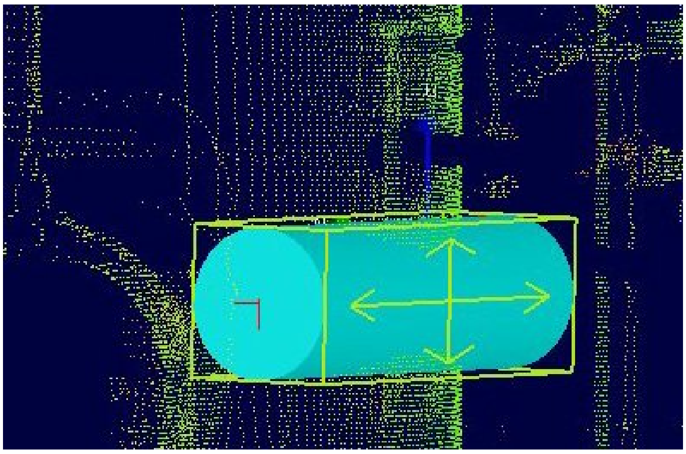


Figure 3-7: movement perpendicular to the objects' axis can be done manually by using the mouse.

There is also the issue of backwards compatibility from RealityLINx to 3D-PlantLINx. As the system works now, it is possible to upgrade the model made in 3D-PlantLINx to the format RealityLINx uses, but not the other way around. As only models in RealityLINx can have intelligent information assigned, any attribute-manipulations need to be done in RealityLINx, thus ruling out a "move objects with intelligence to fit the point cloud" – approach.

For these reasons it would be best to keep modeling in Cyclone as the know-how on this software already is well established and it has better overall features than 3D-PlantLINx. The main reason together with the modeling features for evaluating Inovx' solution, was its ability to read intelligent information from plant design systems such as PDS and PDMS and write information back to these. These functions are not a part of the products Inovx offer, but are rather a service Inovx performs for its clients.

As a whole, Inovx has very close to all the features needed to assign intelligent information. It can make scan models, read and write PDS- and PDMS-models, manipulate the geometry of objects in both scan models and models from PDS/PDMS and it has advanced tools for manipulating intelligent attribute information. Some of the features are however not fully available as a product. Even if 3D-PlantLINx is not used to make the scan models, RealityLINx can still be used to load the models, and with some alterations, to assign the intelligent information. The basic functionality is present and can easily be adjusted to solve our problem. A tool to monitor the progress is also needed to assure the quality of the work done.

Inovx are currently considering implementing these features into their software, but will not conclude within the timeframe of this thesis. The success of this approach depends on Inovx and whether or not they see the value in implementing the features needed.

3.3 PDS/Cyclone solution

In this chapter I will explain an implementation of a system that can assign intelligent information from a PDS model to a scan model made in Cyclone.

Because the PDS-system available is in use only admission through a trained PDS administrator was allowed. A sample model for which scan models already exists was exported from the system. The Microstation model files were copied and the tables containing intelligent information for the model were exported to a local Microsoft SQL server database. In other words all data were provided outside PDS, but only limited access to these data was given through PDS.

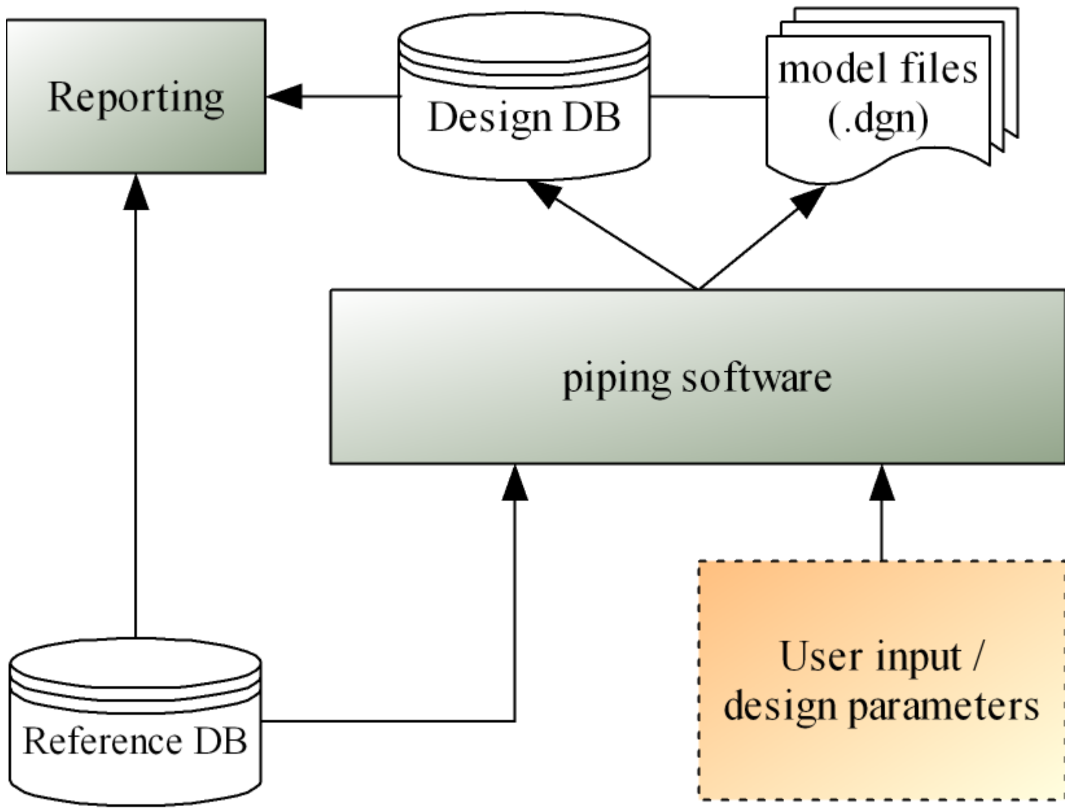


Figure 3-8: Piping software overview, PDS. A summary of the figure in (Intergraph, 2002, p.114)

The scan models used are made in Cyclone and exported to binary format .coe files. These files are then converted to .dgn files in Microstation.

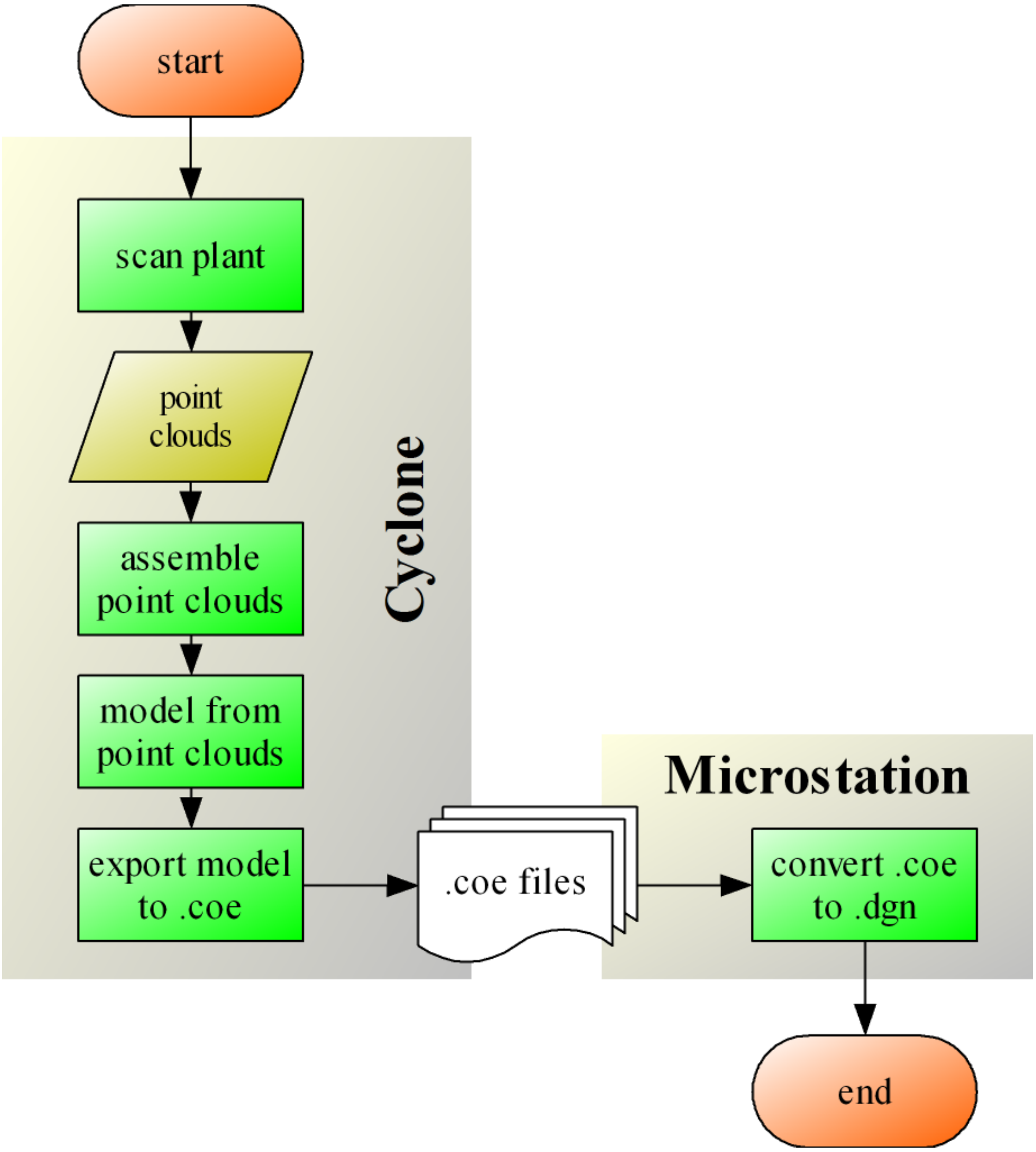


Figure 3-9: Acquiring a scan model readable in Microstation

Each graphical object in the as-built model, with intelligent information, has a database link assigned in the .dgn-file. From this link it is possible to make a query on the database getting all intelligent information associated with the object.

3.3.1 Requirement specifications

1. The application should read the .dgn files and show the objects contained in this, as both a list of objects and as a visual model. Intelligent information connected to the model should be accessed through a database connection. This information does not need to be openly viewable as the application should be able to handle all manipulation on the database automatically, when intelligent information is assigned between the models.

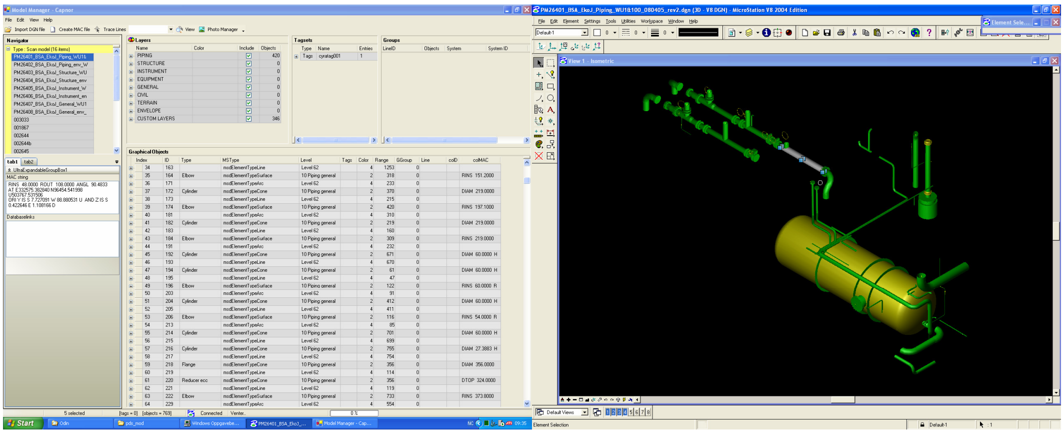


Figure 3-10: A model opened as a list of objects with their attributes on the left side and as a graphical view in Microstation on the right side.

2. Assigning the intelligent information is done by selecting objects in the graphical view not marked as "processed". The corresponding object from the scan model can then be chosen in the graphical view by comparing the two models. When the corresponding object is chosen the application can assign the intelligent information by copying the mslink-information and altering the necessary tables in the database.

3. Processed objects in the models should be distinguishable from unprocessed objects, both in the list view and in the graphical view. Objects should only be processed once. An object with intelligent information should only assign its information once and an object in the scan model should only get intelligent information from one object in the PDS model.

4. When all objects from the PDS model, with intelligent information, have assigned their information to the corresponding objects in the scan model, the model needs to be exported back to PDS. Depending on how much we wish to reorganize the models we need to alter some tables in the database before we export the models back to PDS.

When a plant is made in PDS, the model is usually divided into disciplines. Each discipline is then subdivided into areas. As each .dgn model only can be accessed by one user at a time, these areas are divided into a number of new models depending on how many operators are working on the model. (Intergraph, 2002 p.29) This results in a lot of small model files, and thus a big number of models to manage.

5. If the result after the processing is a set of files with information equal to the files copied from PDS, but with geometric properties from the scan files. These files could be entered back to PDS without any alterations on the database.
6. If the final model files with intelligence are organized as the scan files or in any other appropriate way, some alteration on the database is needed. The database needs to get the new .dgn files added in its table of models and new tables in the design database needs to be added for each new .dgn file.

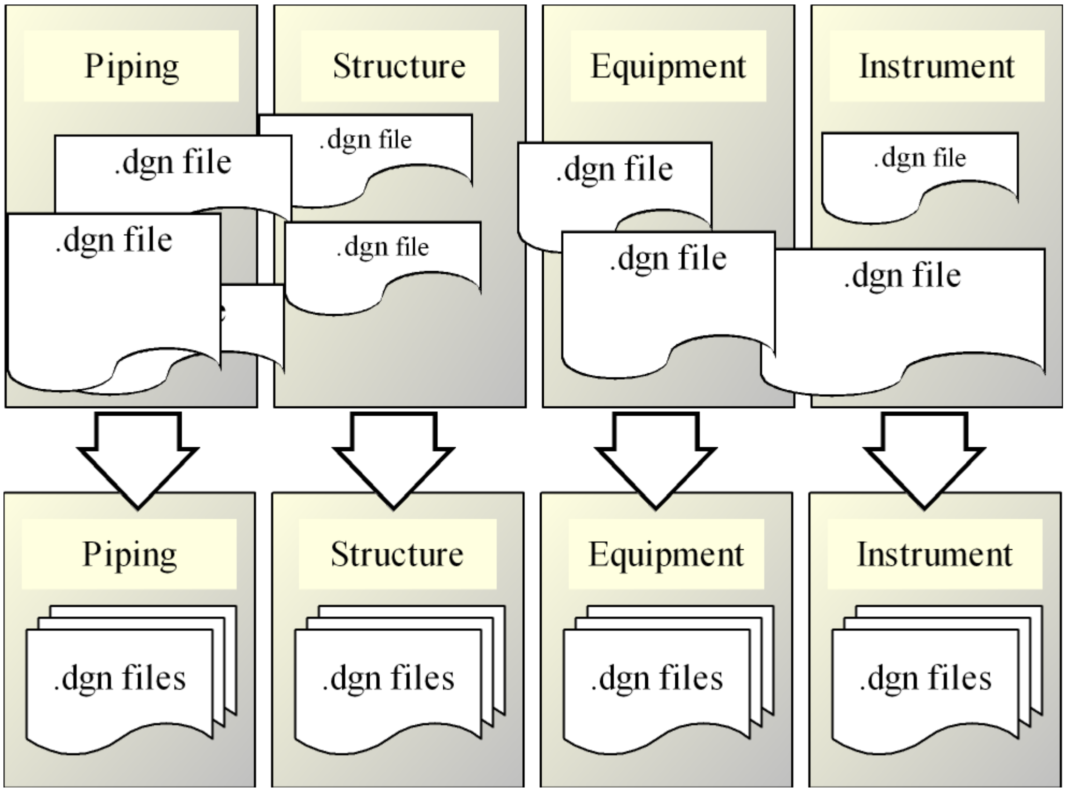


Figure 3-11: Reorganizing the .dgn files from PDS to get more manageable model-files.

4 Implementation

In this chapter I will discuss how a system made to assign intelligent information to a scan model can be implemented.

4.1 Implementation

4.1.1 Accessing the data

After researching PDS and PDMS it was decided to focus mainly on Intergraphs' PDS because this was the only system where expertise was available in-house. Both the .dgn-files and the Microsoft SQL-server database are easily accessible with the software available at Capnor.

A model from PDS was exported so we could read the model. The model we chose was a relatively small model and contained 50 .dgn files. These files were in Microstation v.7 format. To get the information from the database we exported the tables in question from the PDS system into new Microsoft SQL server databases, one database for each main-part of the database. (Design database, Project database and Reference database)

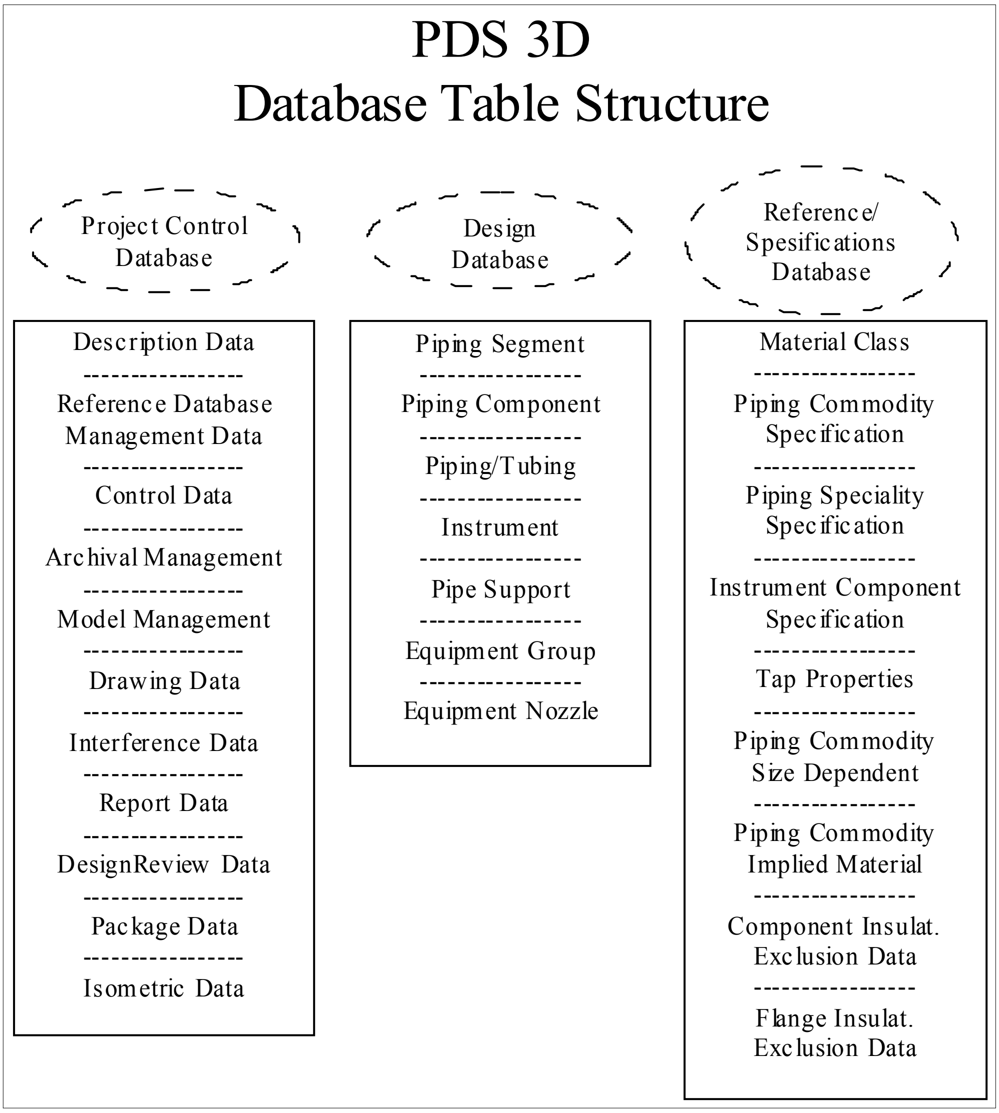


Figure 4-1: Overview over what information the different databases in PDS contains (Intergraph, 2002, p. 56)

We used PDS models from a plant were we already had scan models. The scan models were first modeled in Cyclone and then exported to .coe format. .coe is a file-format that Microstation can read and use to create a new .dgn-file.

4.1.2 Assigning intelligence

Microstation has the ability to link to a database, but a new application needs to be implemented to alter and move these links. This application must be able to cooperate with Microstation and provide all the functionality needed, which Microstation can not provide. This is mainly functionality directed towards database manipulation, mslink manipulation and management of the changes made.

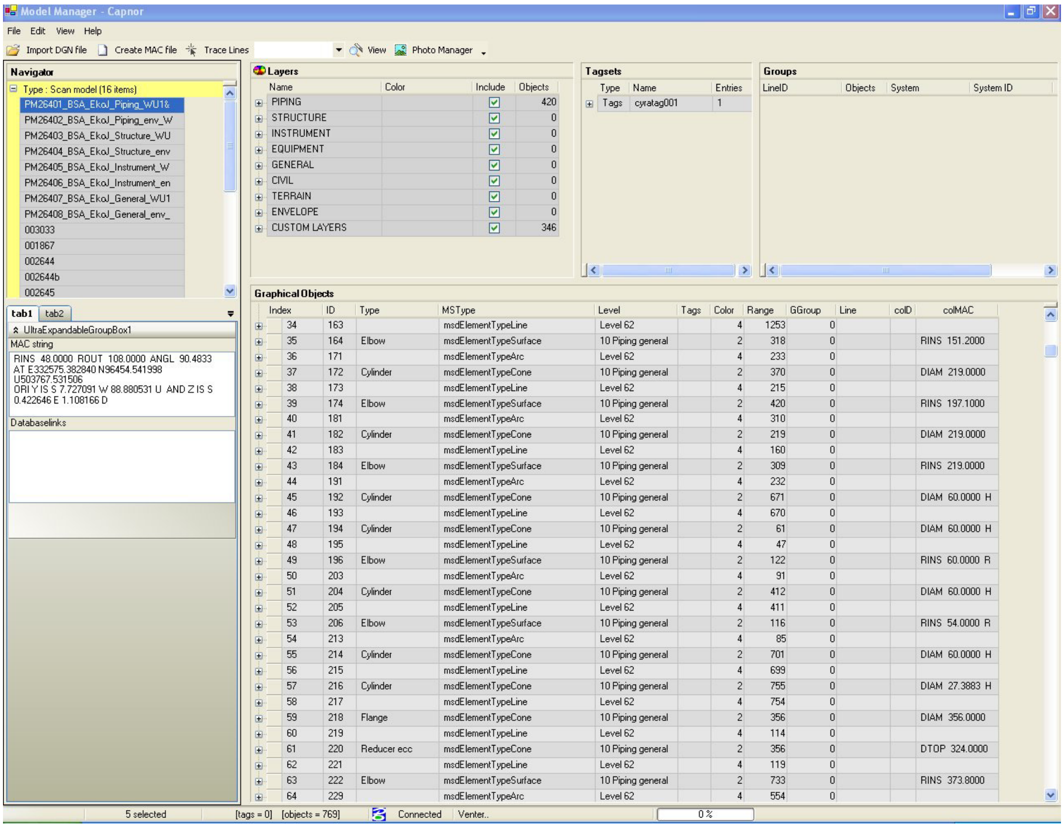


Figure 4-2: Screenshot of the list of objects from a .dgn file

First of all the application loads a scan model (a .dgn file) and lists all objects in the file and their properties. If there is mslink-information on any objects, this information is also listed. At the same time as the file is loaded, a Microstation window is opened with the model in 3D-view. PDS models can now be loaded as a reference in the same view. These reference models can be edited and manipulated in the same way as the master model. After all objects in an as-built model are processed, the PDS model can be “de referenced” and a new PDS model can be referenced, while leaving the scan model as a master model.

A scan model usually covers a larger area and more objects than a single model from PDS, an example of this is shown in the figure 4-3 and 4-4 where the PDS model shows one pipeline and the scan model shows the whole wellhead area. The complexity of the models will however vary between projects. Some PDS models may contain as much or even more than the model shown in figure 4-3, depending on the practice in the specific project and the purpose of the model.

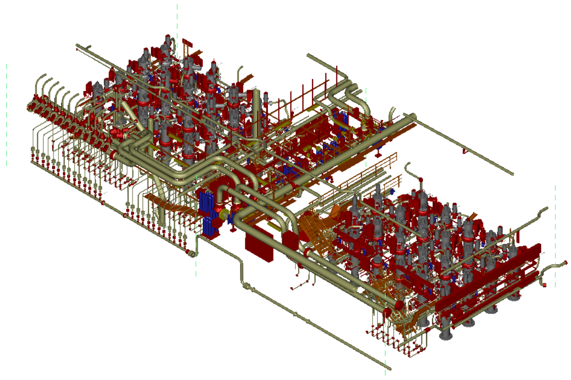


Figure 4-3: A complete scan model of a wellhead area contained in a single .dgn-file.

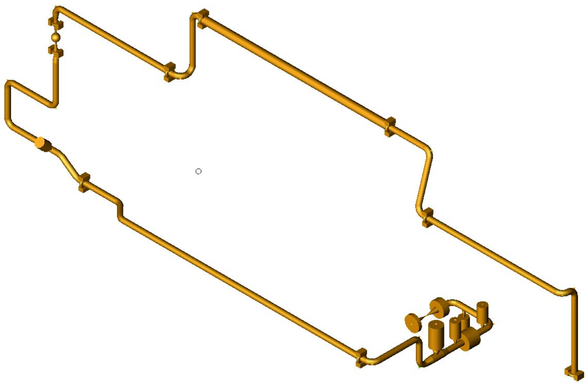


Figure 4-4: The contents of a single .dgn file from PDS, a single pipeline in the wellhead area.

When an object is selected from the list in the application, the corresponding object in the 3D-view will be highlighted.[9] This way it is possible to identify which object in the scan model corresponds to the highlighted object from the PDS model.

[9] Highlighting is here a change of color. Either by the default highlight color in Microstation or by temporary changing the material of the object so that the object is distinguishable from surrounding objects.

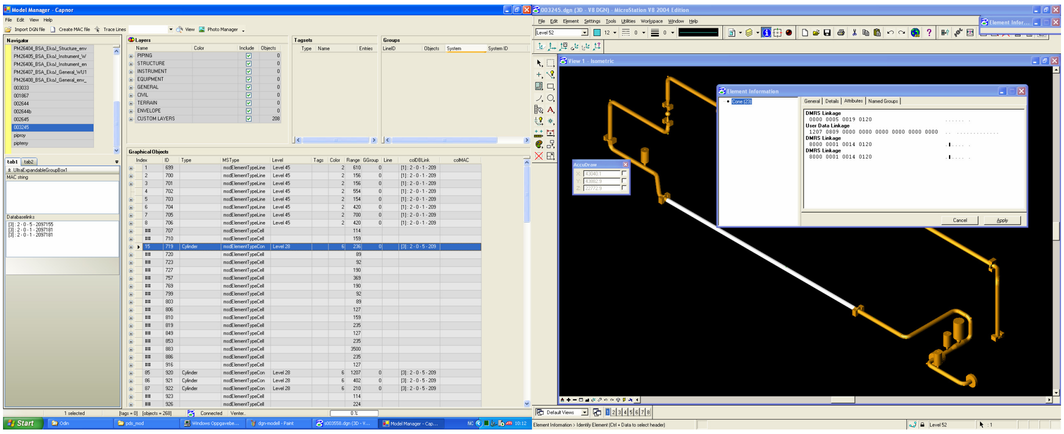


Figure 4-5: Objects highlighted both in the list and in the graphical view.

With the object from the PDS model highlighted, the corresponding objects from the scan model can be identified in the graphical view. The objects from the scan model might not match the extent of the object chosen from the PDS model, leading to grouping and ungrouping of different objects in the scan model, before we have a match between the two models. Microstation has the needed functionality for combining and dividing objects into groups, and these features will be used where such instances occur.

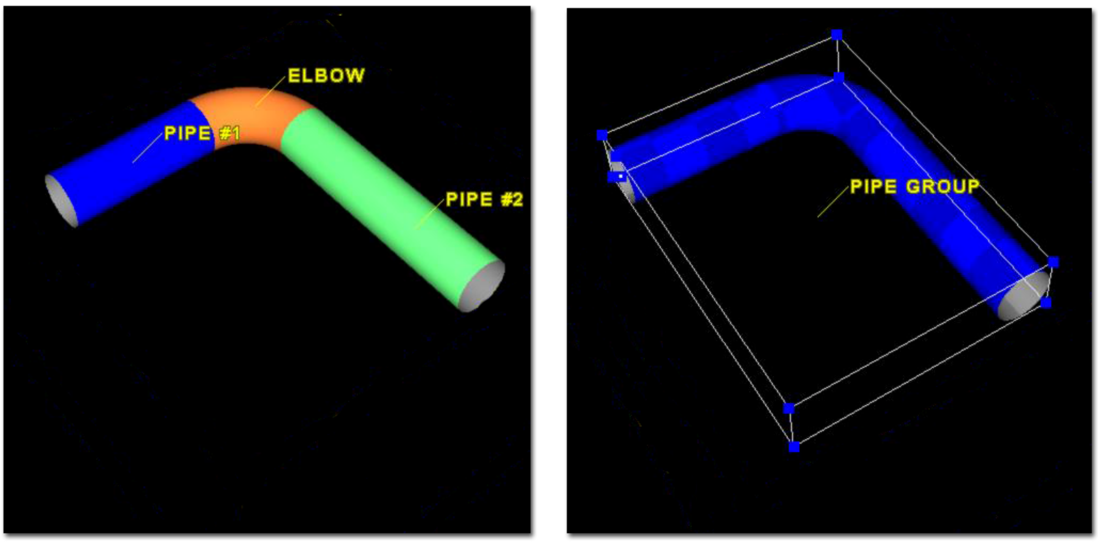


Figure 4-6: On the left: A pipe divided into the primitives it is made of.
On the right: The parts grouped into a single pipe

While researching, I have not been able to determine how the database link information links each object to the database. The documentation available (appendix 1 & (Intergraph, 1998)) suggests the information in the mslink-information contains a query on a table mscatalog, but this table is not present in the PDS system I have access to. I know that PDS is able to locate all intelligent information for a specific object from the information in the .dgn file, but I do not know how to access this information. I tried to contact Intergraph through the local PDS administrator, but her contacts at Intergraph could not offer a solution to the problem.

I will assume that all information in the .dgn file related to the intelligent information is contained in the mslink-information of the object. This means that by moving the mslink-information from the selected PDS object to the matching object in the scan model, all information that needs to be moved in the .dgn file has been transferred. For the transfer of the intelligent information to be complete some alterations in the database are needed.

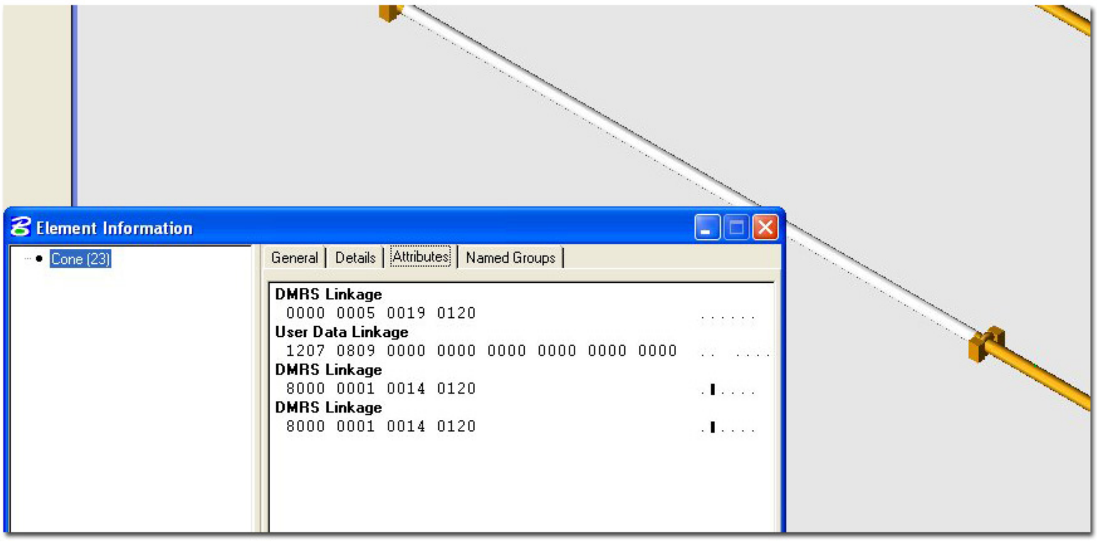


Figure 4-7: The mslink-information containing the database linkages to the white pipe in the background. Each "DMRS Linkage"-field contains one link to the database.[10]

Assuming there is an mscatalog table or that the table link to the .dgn object in the database is similar, this table needs to be altered when the intelligent information is

[10] The database linkages are represented in the mslink-information as hexadecimal numbers. Information about how these numbers are decoded is available in (Intergraph, 1998).

assigned. The object's mslink-information in the .dgn file contains two numbers in particular which are used to construct two queries on the database. The first query is based on an entity number. If this number is 20 the query would be:

Select * from mscatalog where entitynum = 20

The second number results in a query such as (with a number of 589):

Select * from <attribute table> where mslink = 589

<attribute table> is determined from the tablename column in mscatalog for the record where the entitynum is 20. The attribute table is the table in the design database which corresponds to the type of equipment the object is. If the object is a pipe segment, (centerline of pipe) the query would list the row for that particular pipe segment from the segment data table, pdtable_12. When the object has several linkage entries in the mslink-information, the object is linked to several tables in the database.

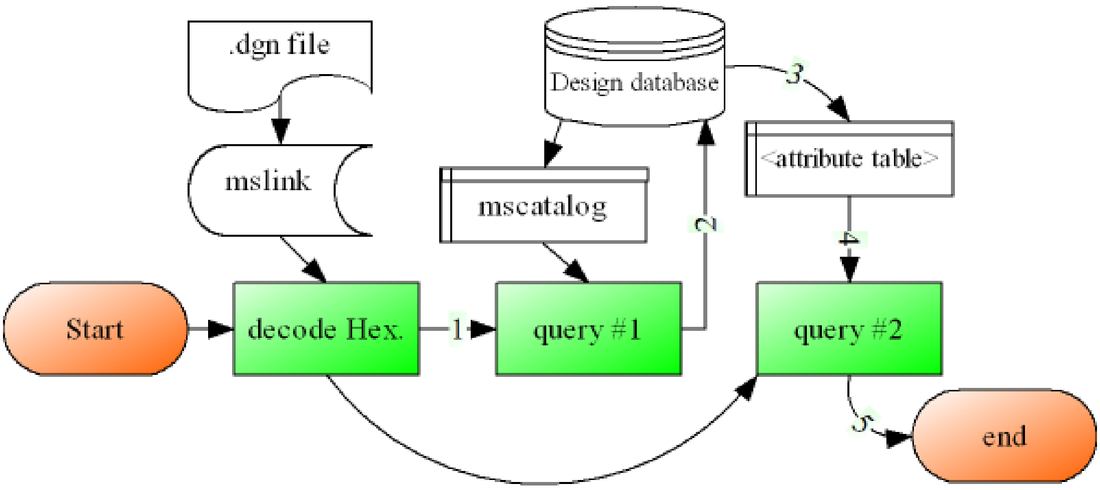


Figure 4-8: Decoding of mslink-information and queries performed on the design database.

In the design database the tables are organized in models. One set of tables for each model. (.dgn file) When a new scan model is loaded a new model should be added in the database with a matching set of tables. For each new object that gets intelligence assigned, the corresponding row in all tables from the PDS model is copied and pasted into the new model. This new model should also be added to the table in project database where all the models are listed (the table: pd_113). This should be enough to insert the new scan model with all intelligent information assigned back into PDS.

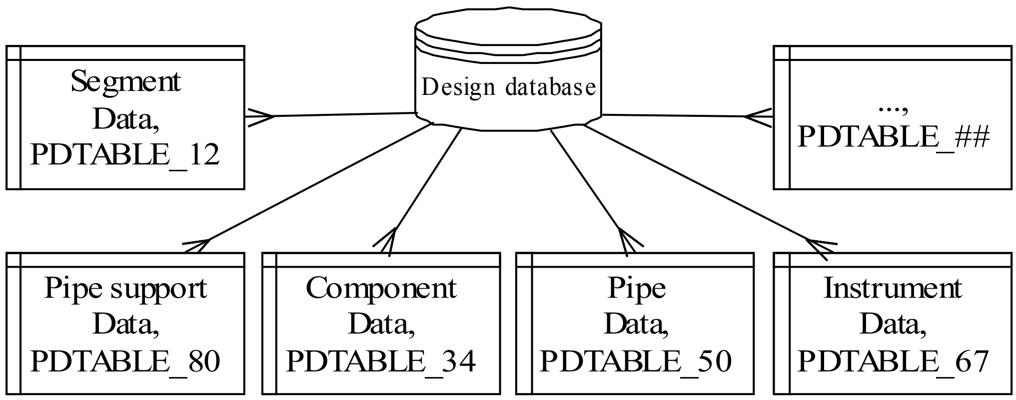


Figure 4-9: Overview over some of the tables in the design database in PDS. Each model has a set of these tables in the database.

4.1.3 Managing alterations

As PDS models gets finished it is important to keep an overview over which models have been and which models still need processing. When the application loads a new model it will add the model and all its objects to a database. If the model already exists in the database it is not added. The database keeps the status of all models and their objects. Initially objects have the status “unprocessed”. As objects get selected for processing they get the status “in process” and when they are finished they get the status “processed”.

Other information about the models such as by whom and when the plant was scanned, who made the scan model and when it was made could also be entered into the database for later administration of the models.

4.2 Workflow

This chapter discusses the workflow from scanned plant to an intelligent scan model.

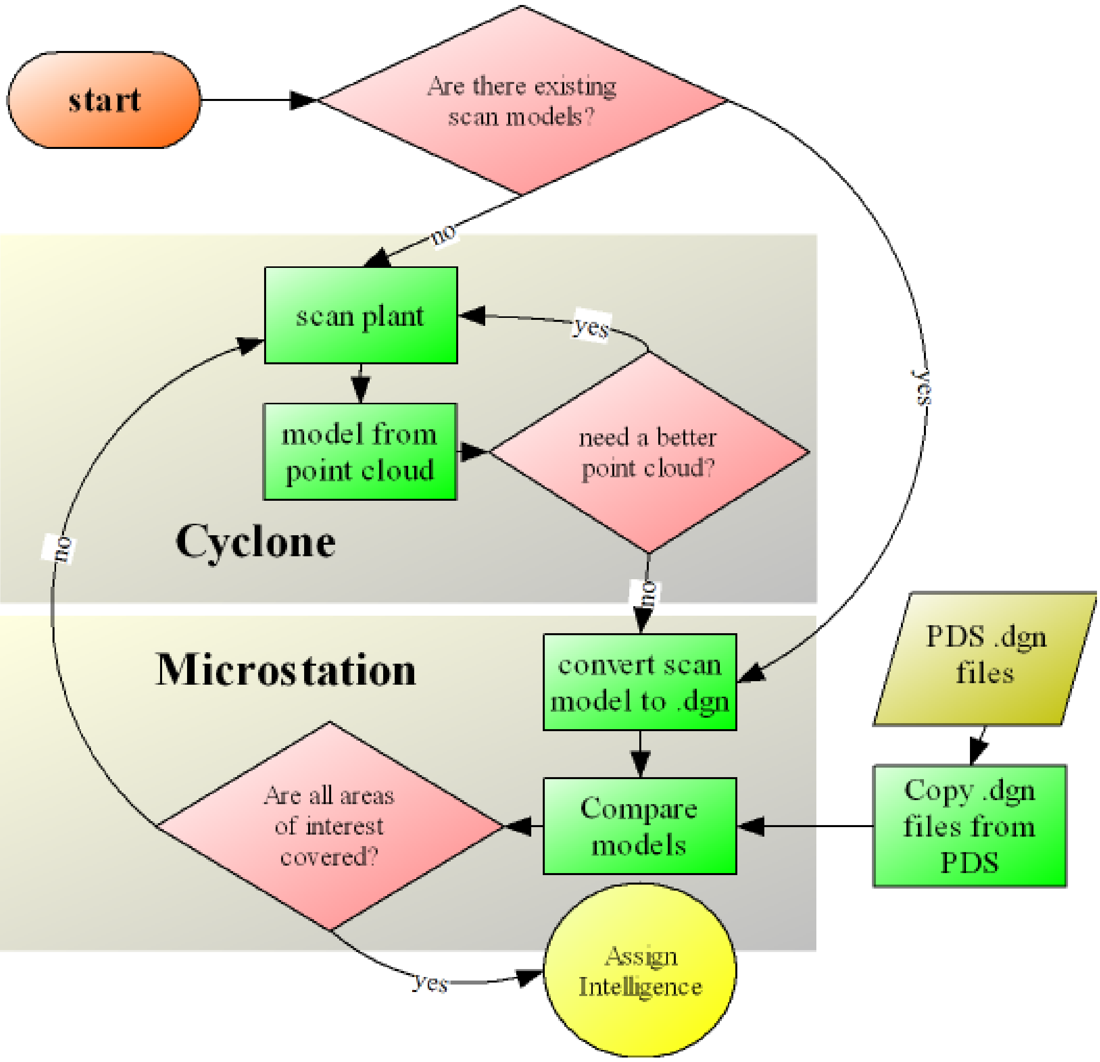


Figure 4-10: Workflow for acquiring the scan model.

If there are no scan models matching the area covered by the PDS model, the area needs to be scanned and modeled. Scanning and modeling follows the workflow already in use for making scan models: The area is scanned and the point clouds are referenced to each other and to the existing grid. Then the point cloud is processed to make it manageable for the modelers' computers. After the point clouds have been processed, modelers fit 3D-primitives to the point cloud to make an accurate scan model.

If the modeler during modeling finds that he lacks points in important areas, these areas need to be scanned. Models made today are only as accurate as the client wants them. To cut down on the costs in the final scan models today, details are not modeled accurately but boxed into close fitting boxes or cylinders. The scan models are going to need a greater level of detail when intelligent information is added to the model. Parts such as pressure gauges, valves and small pipes connected to instruments and machinery are boxed today, to tell there is equipment in the specified area. Before intelligent information can be added to a scan model, all such parts must be modeled. A single object in the scan model can not represent several parts on the plant, as they sometimes do today. Each object in the initial as-built model has to be uniquely mapped to an object in the scan model, so that the objects can be uniquely identified in the model.

When the scan models are finished they are converted to .dgn and compared to the .dgn files from PDS. This comparison could reveal areas in the PDS model which are absent in the scan model and needs to be modeled. If all areas are covered the models are ready to get intelligent information assigned.

Figure 4-11: exporting the tables of interest from the PDS database to a separate database

Before the intelligent information can be assigned to the scan models, the part of the database that is of interest must be exported from PDS and into a separate database. This "work database" contains all tables in the design database corresponding to the

.dgn files copied from PDS. Tables associated with the model from the project database and the reference database is also exported into this database.

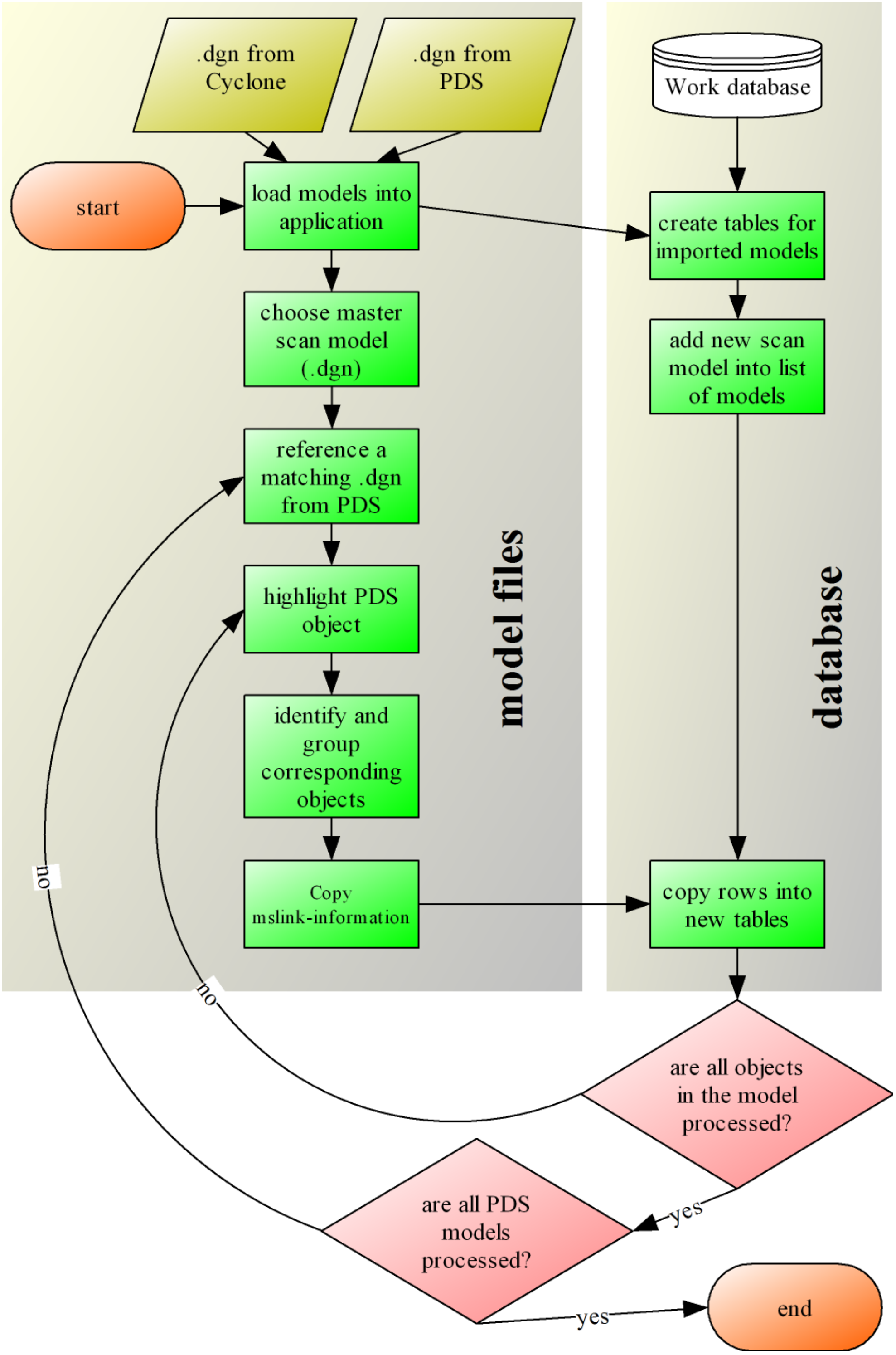


Figure 4-12: Workflow for assigning intelligent information to the scan model.

Intelligent information is added to the scan model with the custom application. First all .dgn files from PDS and .dgn models converted from Cyclone are loaded into the application. A list of all models is generated and the status of all the models is set to “unprocessed”. Then a scan model is selected from the list and loaded into the model view. This model is the master-file, which PDS .dgn files are referenced to. These referenced models can be edited the same way as the master model. Referencing the PDS models has the advantage that the scan model stays while PDS models can be added and removed as they are processed.

When a PDS model is referenced to the scan model, all objects in the model are listed with their attributes. Selecting an object from this list gives the object a new color in the graphical view rather than using the highlighting functionality in Microstation, because it is necessary to have the PDS object highlighted until its intelligent information is assigned to a scan object. With the PDS object selected, the modeler identifies the corresponding objects from the scan model. These objects are then combined to match the object from PDS. 3D-manipulation functionality in Microstation is used to make the combined objects in the scan model.

After combining the matching objects into a group the mslink-information is copied from the PDS object to the group of objects in the scan model. The objects are marked as “processed” in the list and get a new color in the model view to show that they are processed. Any alterations to the database are done automatically by the application when the mslink-information is moved. If there are still more objects in the PDS .dgn file with the status “unprocessed”, a new object is chosen from the list of objects and the process of identifying and combining objects is repeated. This makes it easy to identify any objects in the model still waiting to be processed. A quick view on the model or the list of objects is all it takes to identify and find the objects that need processing. After all the objects in a PDS .dgn file are processed, the file is detached from the scan model and a new .dgn file is referenced.

When all the .dgn files from PDS are processed, the model is imported back to PDS. The .dgn files with intelligent information are copied and the design database tables are added to the design database. Tables in the design database which have been changed or have gotten new rows in the process are altered in the PDS database to reflect these changes.

5 Conclusion

Laser scanning has the last few years started to replace the traditional surveying methods on offshore platforms. Surveying projects which earlier could take years to complete, can today be finished in months. The problem is these new methods of surveying produce completely new models, while the old surveying techniques used the existing models as a basis for their models. The result is these new scan models are not used during the design and planning of new systems because they lack the intelligent data needed. They are only used to control for clashes after the system has been designed using old existing models. By adding the intelligent information from existing models to scan models, the scan models can be used as a basis for the design of new systems.

An approach to getting the intelligent information from a plant design system and assigning this information to a scan model has been suggested. The method suggested uses Microstation and its .dgn format as a basis to show the models graphically and to choose the matching objects to get the intelligent information assigned. To manage all alterations done on the models, a separate application is implemented on top of Microstation. This application lists all objects in the models and keeps track of which objects have been altered and which objects are due.

The application provides the tools necessary for an operator to assign the information from an existing model with intelligent information, to a new scan model. A graphical view of the model is used to identify and match objects from the different models. As objects assigns or get assigned intelligent information they are marked by the application to mark them as “processed”. When the intelligent information is assigned, the application adds the necessary information in the database automatically.

The quality of the final model is dependent on the accuracy of the final model and that all objects have gotten the correct intelligent information. The accuracy of the model is equal to the accuracy of the scan model, which usually is 2mm for most modeled surfaces. Whether all objects have gotten the correct intelligent information is dependent on how easy it is to identify matching objects. It is easy to check that all the objects have gotten intelligent information, in the workflow suggested, but a workflow for checking that this information is the correct information, has not been suggested. This is not necessarily a problem, but the system needs to be implemented and tested before we know how big this problem is. As long as the procedure for assigning the intelligent information is easy-to-follow the problem should be minimal.

6 Future work

As the approach suggested has not been fully implemented and tested this is the first work that needs to be done.

The approach I have presented is an approach to assigning intelligent information from a PDS system to corresponding scan models. As an increasing number of plants and platforms are modeled in PDMS the method should also be implemented to facilitate PDMS and other plant design systems available now and in the future. A switch from PDS to PDMS from the system suggested and partly implemented, would require a completely rework of the system, and how the intelligent information is assigned to the models. Using XMpLant and its XML format as a basis for how the intelligent models are saved and the intelligent information is accessed would make the application more robust to new applications and changes made in the plant design systems.

XMpLant represents all plant items as generic objects. The generic objects are classified in a public domain XML schema. This means that the core is flexible and new objects can be classified without changing the program. XMpLant utilize a mapping subsystem to define the mapping of attributes and their values between a native system and the neutral model.

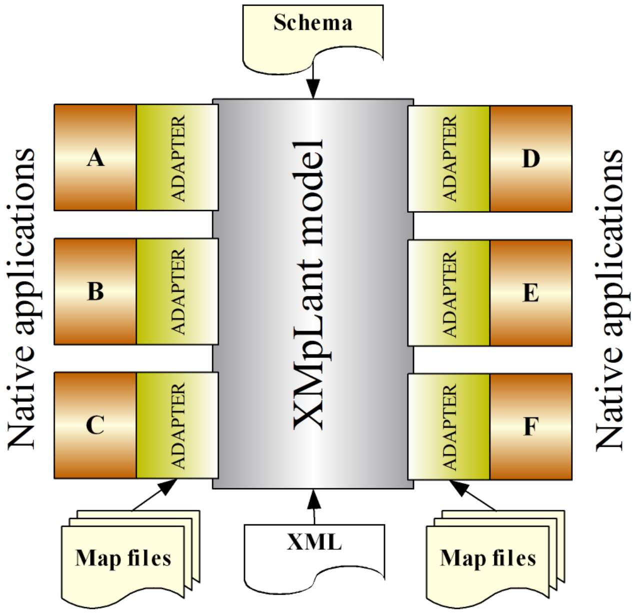


Figure 6-1: How XMpLant works. (Neumenon, n.d.)

If the application interfaces directly with the XmpLant XML model instead of the converted .dgn files and different databases, it would be more robust towards changes made in the different design systems such as PDS, PDMS and 3D-PlantLINx. Variations between projects are handled by the Map files which map the models and their attributes to the XML model and no changes need to be made in the application.

The approach suggested, relies on some meticulous matching done by the operators. This work could to some extent be automated. Rather than selecting the matching objects in the scan model manually for each PDS object, the application could suggest close objects which have similarities with the object. The type, orientation and size of the object together with the position could produce enough attributes for a matching algorithm to work in most cases. Only special cases where the matching algorithm did not find a match, or where the match found by the matching algorithm is evidently wrong, would need to be resolved by the operator.

7 Bibliography

Atkinson, K. B. (2001) *Close range photogrammetry and machine vision,* (1st ed.) Caithness, UK: Whittles.

Autodesk, Inc. (2005) *AutoCAD product information,* Retrieved June 06, 2005 from http://usa.autodesk.com/adsk/servlet/index?siteID=123112&id=2704278

AVEVA Solutions Ltd. (2005) *Plant design management system (PDMS)*, Retrieved June 06, 2005 from http://www.aveva.com/library/datasheets/pdf/VPD_pdms.pdf

Bentley Systems, Inc. (2005) *Microstation,* Retrieved June 06, 2005 from http://www.bentley.com/en-US/Products/MicroStation/

Boehler, W., Heinz, G. & Marbs, A. (2001) *The potential of non-contact close range laser scanners for cultural heritage recording.* CIPA Working group VI. Mainz: University of applied sciences & Roemisch-Germanisches Zentralmuseum.

Daratech, Inc. (1994) *3D Plant Design Systems: benefits and paybacks,* Cambridge, MA:

INOVx Solutions, Inc. (2003) *advanced plant lifecycle management solutions,* Retrieved June 6, 2005 from http://inovx.com/

Intergraph Mapping and Geospatial Solutions. (1998) *Analyzing the MGE database linkage on Microstation graphic elements.* Retrieved June 06, 2005 from http://imgssupport.intergraph.com/TechNotes.asp

Intergraph (April 2002) *PDS 3D theory, user's guide.* Available on appended CD.

Intergraph Corporation (n.d.) *Smartplant® 3D,* Retrieved June 06, 2005 from http://ppm.intergraph.com/smartplant/3d/

Jones, B. (2003) *Hydrocarbon Engineering,* Farnham Surrey, United Kingdom: Intergraph process power & offshore.

Leica Geosystems (2004) *HDS3000 3D laser scanner,* Retrieved June 06, 2005 from http://hds.leica-geosystems.com/products/HDS3000_specs.html

Leica Geosystems (2004) *HDS4500 3D laser scanner,* Retrieved June 06, 2005 from http://hds.leica-geosystems.com/products/products_pdfs/HDS4500_specs_25m.pdf

Leica Geosystems (2005) *The HDS software family,* Retrieved June 06, 2005 from: http://hds.leica-geosystems.com/products/products-soft.html

Merriam-Webster online dictionary. (n.d.) *Plant (noun)*, Retrieved June 06, 2005, from http://www.m-w.com/

Neumenon Consulting Ltd. (n.d.) *XMpLant,* Retrieved June 06, 2005 from: http://xmplant.com/

Wikipedia, the free encyclopedia. (May 07, 05) *VAX,* Retrieved June 06, 2005 from http://en.wikipedia.org/wiki/VAX

Wikipedia, the free encyclopedia. (May 23, 05) *Isometric projection*, Retrieved June 06, 2005, from http://en.wikipedia.org/wiki/Isometric_projection

8 Appendix

8.1 PDS documentation on CD

PDS documentation is provided on the CD appended with permission from Intergraph (all right reserved). The documentation is provided for academic use only and may not be used for commercial purposes.