Parametric Design Tool
for W6L32 Common Baseframe

Filip Slotte

Bachelor´s thesis
Degree Programme in Mechanical and Production Engineering
Vaasa 2016

BACHELOR'S THESIS

Author: Filip Slotte

Degree Programme: Mechanical engineering

Specialization: Mechanical Construction Systems

Supervisors: Tomi Vesterbacka (Wärtsilä) and Kenneth Ehrström (Novia)

Title: *Parametric design tool for W6L32 common baseframe*

Date: 4.3.2016 Number of pages: 30 Appendices: 1

Summary

This thesis was made for Wärtsilä Marine Solutions and enlightens the construction process of a parametric design tool for W6L32 generator set´s common base frame.

A parametric tool is useful when an engineer needs to design parts that are similar, but not identical, to each other. By analyzing the conditions that determine the design of the product and setting these as parameters into a CAD program, the engineer can quickly make a 3D-model that meets the specifications set by the company's customers. Using the parametric design features in Siemens NX a user friendly model was created ready to be customized for each new project. This approach requires a different design approach than normal CAD-modelling.

The result is a parametric model for the W6L32 with a design that is easy to alter according to project specifications. Design time is estimated to decrease from one week to 1-2 days. Future development will include making a parametric model for other engine types.

Language: english Key words: parametric design, 3D-design, Siemens NX

EXAMENSARBETE

Författare: Filip Slotte

Utbildningsprogram och ort: Maskin- och produktionsteknik, Vasa

Inriktningsalternativ: Maskinkonstruktion

Handledare: Tomi Vesterbacka (Wärtsilä) och Kenneth Ehrström (Novia)

Titel: *Parametriskt designverktyg för W6L32 generatorsetens fundament*

Datum: 4.3.2016 Sidantal: 30 Bilagor: 1

Abstrakt

Detta examensarbete är gjort åt Wärtsilä Marine Solutions, Project Management, Mechanical Engineering, med syftet att effektivisera 3D-modelleringen av W6L32 generatorsetens fundament, common base frame.

En parametrisk modell är användbar för produkter som är liknande, men inte identiska, i design. Genom att identifiera de parametrar som bestämmer produktens design kan dessa parametrar länkas till mått i tillhörande delar av produkten. Detta gör att modellen lätt och snabbt kan anpassas till projektspecifika mått. Funktionerna i Siemens NX ger en lätthanterlig modell redo att editeras.

Resultatet är en modell för generatorsetens fundament som enkelt kan modifieras för att uppfylla projektspecifika krav. Designtiden uppskattas minska från ca en vecka till en till två dagar. I framtiden ska modellen utvidgas att omfatta Wärtsiläs andra motortyper.

Språk: engelska Nyckelord: parametrisk design, 3D-design, Siemens NX

OPINNÄYTETYÖ

Tekijä: Filip Slotte

Koulutusohjelma ja paikkakunta: Kone- ja tuotantotekniikka, Vaasa

Suuntautumisvaihtoehto: Konesuunnittelu

Ohjaajat: Tomi Vesterbacka (Wärtsilä) ja Kenneth Ehrström (Novia)

Nimike: *W6L32 generattorisettien alustojen parametrinen suunnittelutyökalu*

Päivämäärä: 4.3.2016 Sivumäärä: 30 Liitteet: 1

Tiivistelmä

Opinnäytetyö tehtiin osastolle Wärtsilä Marine Solutions, Project Management, Mechanical Engineering. Tarkoituksena oli tehdä parametrinen 3D-malli W6L32 generaattorisettien alustoille nopeuttaakseen suunnitteluprosessia.

Parametrinen malli sopii käytettäväksi tuotteille, jotka ovat samankaltaisia, mutta eivät identtisiä. Kartoittamalla tuotteen suunnitteluun vaikuttavat parametrit, voidaan nämä syöttää CAD-ohjelmistoon ja sitä kautta rakentaa 3D-malli, joka on helposti muokattava. Hyödyntämällä Siemens NX:in parametriset suunnittelutoiminnot saadaan aikaan 3D-malli, jonka saa helposti muokattua projektikohtaiseen suunnitteluun. Parametrisen mallin tekeminen vaatii suurempaa tarkkuutta kuin tavanomainen 3D-mallintaminen.

Tuloksena on helposti muokattava 3D-malli, joka arvioinnin mukaan lyhentää suunnitteluaikaa yhdestä viikosta 1-2 päivään. Mallia käytetään projektikohtaisen W6L32 alustojen suunnitteluun. Tulevaisuudessa muut moottorimallit saavat omat parametriset mallinsa.

Kieli: englanti Avainsanat: parametrinen suunnittelu, 3D-suunnittelu, Siemens NX

Table of Contents

Appendices

Appendix 1: Manual for Common Base Frame Parametric Model

I. INTRODUCTION

This thesis enlightens the construction process of a parametric design tool. This tool is useful when an engineer needs to design parts that are similar, but not identical, to each other. By analyzing the conditions that determine the design of the product and setting these as parameters into a CAD program, the engineer can quickly make a 3D-model that meets the specifications set by the company's customers. This is a more convenient approach than starting the designing from scratch.

A. Background

The customer for this thesis is Wärtsilä Finland Oy. The division is Marine Solutions, the department is Project Engineering and the team is Mechanical Engineering. One of the things this team designs is common base frames. I worked in this team during the summer of 2015 and was offered a thesis work in order to speed up the designing of common base frames.

The common base frames have formerly been designed by an external engineering company. In June 2015 Wärtsilä decided to end this cooperation and the work was transferred to the mechanical engineering team I worked at. One person on our team had some experience in designing common base frames, so we did a lot of investigation. Using a design guide made by Wärtsilä and looking at old projects we were able to design four base frames. The design process was very time consuming, because for every base frame we had to find a similar one from old projects, find out the differences and figure out how to implement them onto the current frame.

B. Wärtsilä

Wärtsilä is a company providing lifecycle power solutions for the marine and energy market. Marine Solutions, Energy Solutions and Services are the three divisions at Wärtsilä. The company has locations in nearly 70 countries and a 17717 personnel. In 2014 the net sales were 7,779 billion euros. [1]

The mechanical engineering team I worked at is a part of Marine Solutions, meaning they design marine applications. This team focuses on making the project specific drawings for the engine, flywheel and coupling, gearbox, propeller shaft, propeller, generator sets and engine room. The team also assists the rest of the project organization with technical expertise.

II. PROBLEM DEFINITION

The task is to make a parametric tool in the CAD-program Siemens NX 9.0 that designs a project specific common base frame according to the project specification given by Wärtsilä´s customers. The base frame is a welded structure on which the engine and generator are bolted on. Many factors influencing the design, for example: the length, width and height often vary, some steel plate designs can be reused, some steel plates are unique for every project with different hole patterns etc. The base frame design has some required distances between plates and components to minimize vibrations and stress acting on the structure. To start designing every new project from scratch would be very time consuming.

A. Delimitations

Due to various base frame configurations for Wärtsilä´s different combustion engine configurations, me and my supervisors decided to limit this thesis to designing a tool exclusively for the straight six cylinder diesel engine, model name W6L32, since one frame alone has so many possible configurations. We also decided to exclude any accessories such as external lubrication system and non-standard requests by customers, this is what the engineer should focus on. The tool should be used for providing a base frame containing the essentials, ready for adjustment according to the customers' demands.

III. PARAMETRIC DESIGNING

This chapter will describe what parametric designing is, different design approaches, parametric designing in NX 9.0 and NX features useful when making a parametric model.

A. Introduction

Every designer takes part in using and developing representations of a system consisting of smaller parts. When altering one of these parts and it changes the whole system and an entirely new system is created. Normally the system does not respond to changes in a part, all parts have to be evaluated when a change is made to make sure the systems original purpose is still achieved, one minor change can have a major impact on the system as whole. The idea behind parametric modelling is assigning relations between parts and making sure the whole system responds to changes being made. [2]

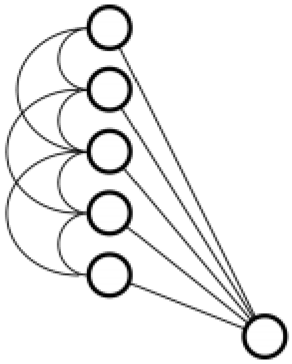
There are two types of modelling systems from a user's perspective: propagation-based and constraint-based. The propagation-based system is common when designing parametric models. It is in its simplest form a spreadsheet with cells that contain a certain value. Some cells have a value inserted by the user while some cells are given a formula which calculates their values depending on values in the user inserted cells. This creates a relationship between cells and ultimately a solution is found for the whole system. [2]

	A
1	1
2	1
3	=A1+A2
4	=A2+A3
5	=A3+A4
	=sum(A1:A5)

(a) User-entered formulae in spreadsheet cells.

	A
1	1
2	1
3	2
4	3
5	5
	12

(b) Spreadsheet display.



(c) The internal data structure. Circles are cells. Lines and arcs are dataflows. In this diagram, arrows are omitted: data always moves downward.

Figure 1: The structure of parametric modelling [2]

Figure 1 shows the fundamental idea of parametric modelling. However, in practice these systems can be very complex. Another issue to bear in mind is that a formula can be written in different ways, the designer must write it in such a way that no misunderstanding is possible. [2]

The constraint based system means giving pre-determined limits for each variable and letting the system solve an equation in order to find the optimal system solution. This differs from propagation based systems because constraint based systems solve a set of continuous and discrete constraints whereas propagation based systems solve unknowns from knowns in a dataflow model. [3] [4]

(a) Equation: $(2X_1)^2 + X_2^2 \geq 5$ ⇒

	$X_2 = 1$	$X_2 = 2$	$X_2 = 3$
$X_1 = 0$	0	0	1
$X_1 = 1$	1	1	1

(b) Equation: $\cosh(X_2) > 2$ ⇒

$X_2 = 1$	$X_2 = 2$	$X_2 = 3$
0	1	1

(c) Allowable combination between X_2 and X_3: (1, b), (2, a), and (3, b) only ⇒

	$X_2 = 1$	$X_2 = 2$	$X_2 = 3$
$X_3 = a$	0	1	0
$X_3 = b$	1	0	1

Figure 2: An example of constraint based design [3]

B. Different design approaches

In his article Matthew Loew explains two different designing approaches, top-down and bottom-up. He describes the differences between them and gives advice on which one to choose. [5]

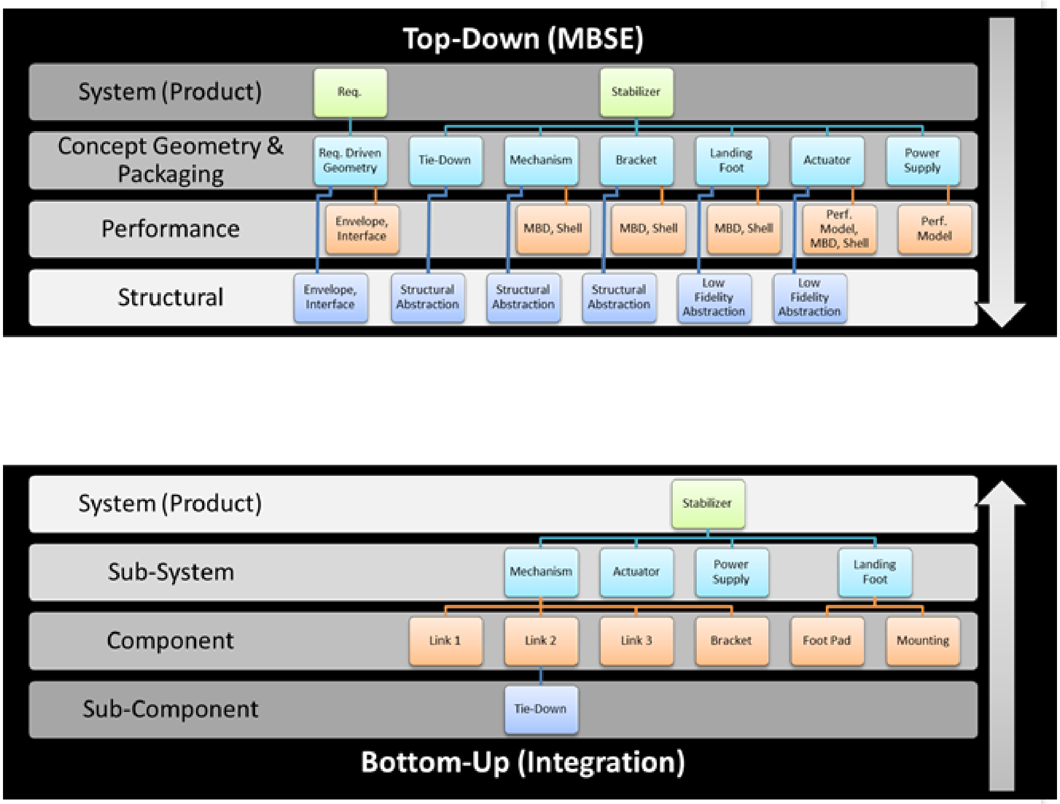


Figure 3: Comparison of top-down and bottom-up approaches [5]

As seen from Figure 3 the top-down approach starts with taking the requirements of the system into consideration. The designer might start drawing simple geometry or abstract models of the product by making a design concept, then checking for performance and finally analyze the structures. This method allows a faster development cycle when creating different design concepts. It also gives flexibility for the designer to select appropriate tools and features. [5]

The bottom-up approach can be compared to building with Legos. You start with basic, low-level and often predefined components. Combining them with other components and giving them constraints and relations with other parts will create the final system solution. This method is very useful for well-proven design concepts which need only minor modification compared to the top-down approach where several design concepts are often made and evaluated. [5]

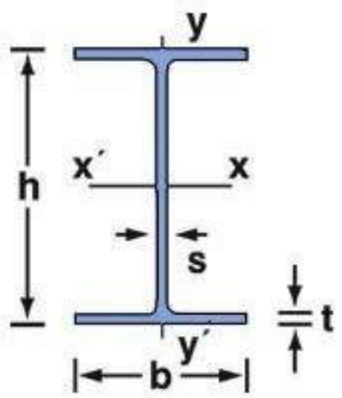
For this thesis the bottom-up approach was chosen since a common base frame consists of pre-defined parts and dimensions.

C. Parametric designing in NX 9.0

This sub-chapter will cover the NX features that were used when making the parametric model. In order to better explain this an example with a simple beam was created. Some of these aspects differ from traditional designing and must be taken into consideration in order to get a fully functioning parametric model.

In our example we will design a variety of simple beams according to the table in Figure 4.

CCORDING TO PRODUCTION: DIN 1025 / EN 10034
STEEL QUALITY: DIN 17100 / EN 10025



	WEIGHT	DIMENSIONS				ΔΙΑΤΟΜΗ	ΡΟΠΗ ΑΝΤΙΣΤΑΣΕΩΣ	
	kg / m	mm				cm2	cm3	
		h	b	s	t	F	Wx	Wy
80	6,000	80	46	3,8	5,2	7,64	20,0	3,69
100	8,100	100	55	4,1	5,7	10,30	34,2	5,79
120	10,400	120	64	4,4	6,3	13,20	53,0	8,65
140	12,900	140	73	4,7	6,9	16,40	77,3	12,30
160	15,800	160	82	5,0	7,4	20,10	109,0	16,70
180	18,800	180	91	5,3	8,0	23,90	146,0	22,20
200	22,400	200	100	5,6	8,5	28,50	194,0	28,50
220	26,200	220	110	5,9	9,2	33,40	252,0	37,30
240	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	47,30
270	36,10[illegible]	270	135	6,6	10,2	45,90	429,0	62,20
300	42,200	300	150	7,1	10,7	53,80	557,0	80,50
330	49,100	330	160	7,5	11,5	62,60	713,0	98,50
360	57,100	360	170	8,0	12,7	72,70	904,0	123,00
400	66,300	400	180	8,6	13,5	84,50	1160,0	146,00
450	77,600	450	190	9,4	14,6	98,80	1500,0	176,00
500	90,700	500	200	10,2	16,0	116,00	1930,0	214,00
550	106,000	550	210	11,1	17,2	134,00	2440,0	254,00
600	122,000	600	220	12,0	19,0	156,00	3070,0	308,00

Figure 4: A sketch and dimensions of an I-Beam (http://www.alibaba.com/product-detail/European-standard-I-beam-IPN-Beam_699420537.html)

1) Sketch constraints

Constraint are a way of defining dimensions and geometric relations in a 2D sketch. In NX there are two types of constraints: geometric and dimensional. Geometric constraints describe the geometric relation between lines, arcs etc. Such relations can be parallel and perpendicular for example. Dimensional constraints describe the size and location of geometric shapes such as distance to sketch origin and length of line to name a few. When creating a sketch NX will by default automatically make both geometric and dimensional constraints. This allows the designer to use a “shape before size” design philosophy, first the shape is created disregarding the dimensions. When the shape is drawn some key dimensions can be added leaving the auto-dimensions that are left. The sketch must however be fully constrained to avoid problems when making a parametric model. NX will notify whether the sketch is under-, fully- or overconstrained. [6]

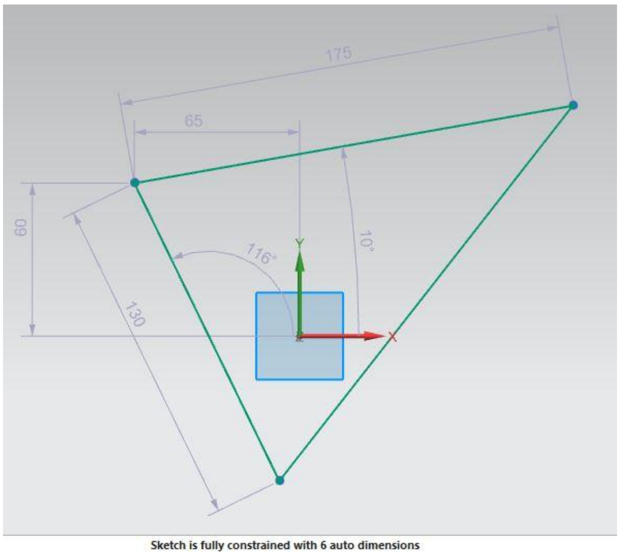


Figure 5: Auto-constrained sketch.

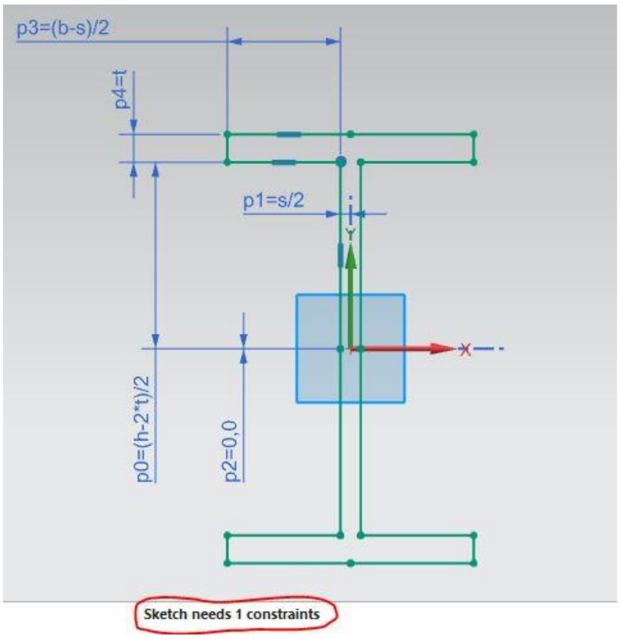


Figure 6: Under constrained sketch, too few dimensions describing the sketch

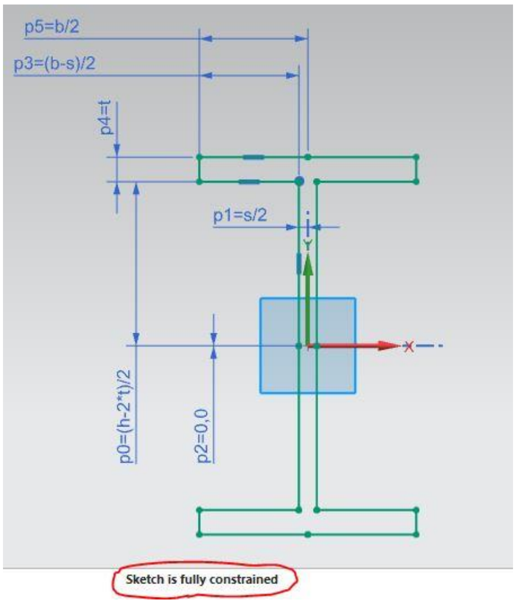


Figure 7: Fully constrained sketch

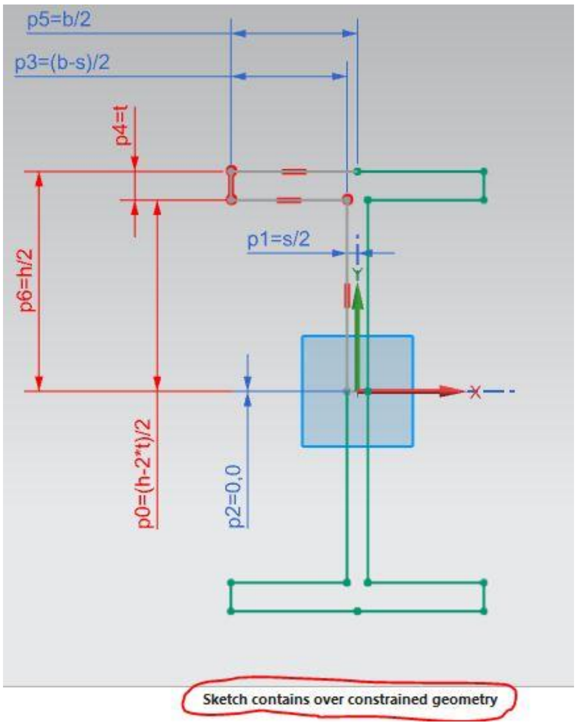


Figure 8: Over constrained sketch, too many dimensions describing the sketch

2) Expressions

As seen in figures 6, 7 and 8 there are mathematical expressions describing the dimensions of the sketch. In our example we can make the variables h, b, s and t in Figure 4 and give them values according to the table.

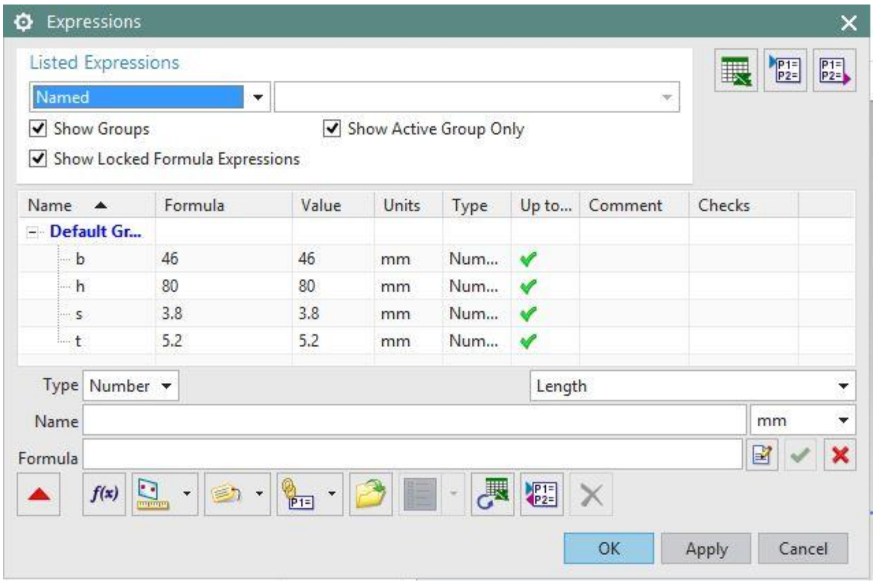


Figure 9: The Expressions window

The variables can now be given values manually. They can also be used in mathematic functions to form relations between them, see Figure 1. These variables can be used in any part of the model to describe a dimension in millimeters. Variables can be assigned with a large variety of quantities such as mass, volume, time etc. A variable can also be defined as a measurement taken from the model itself which is useful when using Interpart links. Every variable can be given a range of values for which it satisfies the design intent, these are called Requirements. In our beam example we could set “mass” as a variable that measures the current beams mass. A requirement can be made, the beam must not be heavier than 500 kg. NX then highlights this error and we can fix it. A Requirement not meeting set values can be classified as failed, warning or information depending on how serious the error is. This has to be decided by the designer of the parametric model.

3) Interpart links

When modelling an assembly consisting of many components it can be useful to link dimensions between parts using Interpart links. The dimension now changes along with the linked dimension. A mathematical expression using a linked dimension can also be made.

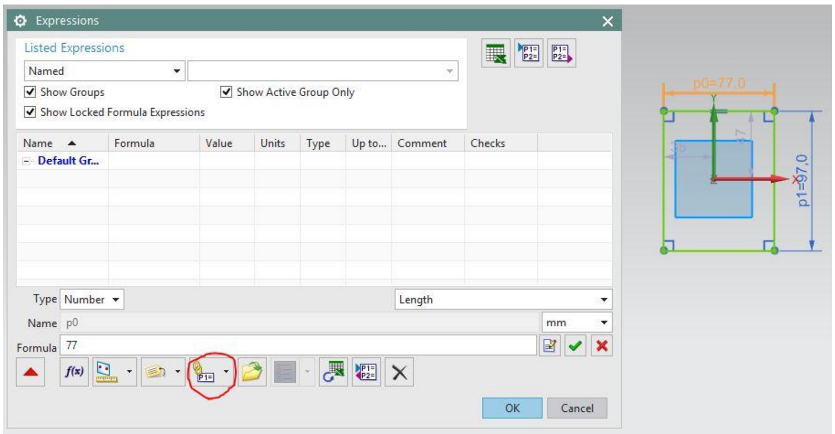


Figure 10: Interpart link, the highlighted dimension will be linked to the beam

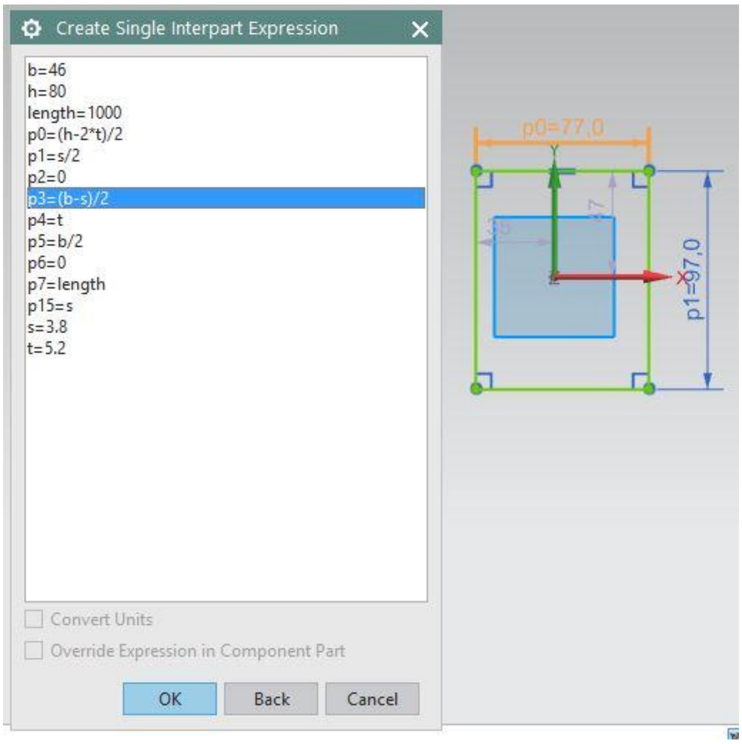


Figure 11: The desired dimension is chosen from the beams Expressions

4) Wave linking

Wave linking means creating a link between a source geometry and a target geometry enabling the option of modifying one without altering the other.

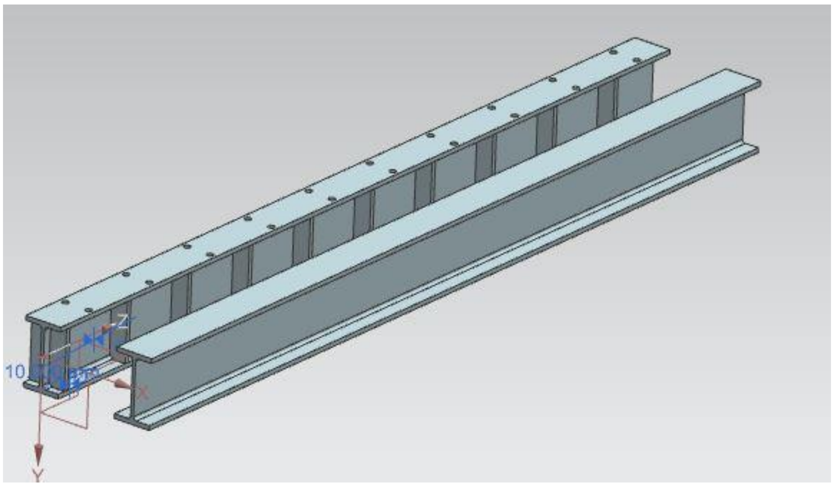


Figure 12: Wave link

In Figure 12 we see two beams with identical geometry, the left beam is part of an assembly, the right one is not. If the designer would use a top-down design approach the left beam would be designed first to fit the system requirements, then by making a wave link a pre-machined beam could be modelled. In this case the left beam would be source geometry, leaving out anything but the beam structure, and the right beam would be target geometry. This would allow for example material to be added to the target geometry without adding it to the source geometry. [7]

If a bottom-up design approach is used the right beam would be source geometry and the left would be target geometry. As seen from figure 12 adding holes to the target geometry does not affect the source geometry, but if we were to modify the source geometry by for example making the beam shorter, these changes would show in the target geometry. This approach is ideal when for example a welded model is built first and then machined, which is the case for this thesis. [7]

5) Suppress by expression

Features in a model can be suppressed using an expression. This will create a new variable which can be given a simple if-statement whether the feature should be visible or not. One condition in the right beam of Figure 12 could be if the beam is shorter than a certain length, no holes are drilled. The suppression can also be manually inserted with a value deciding the suppression.

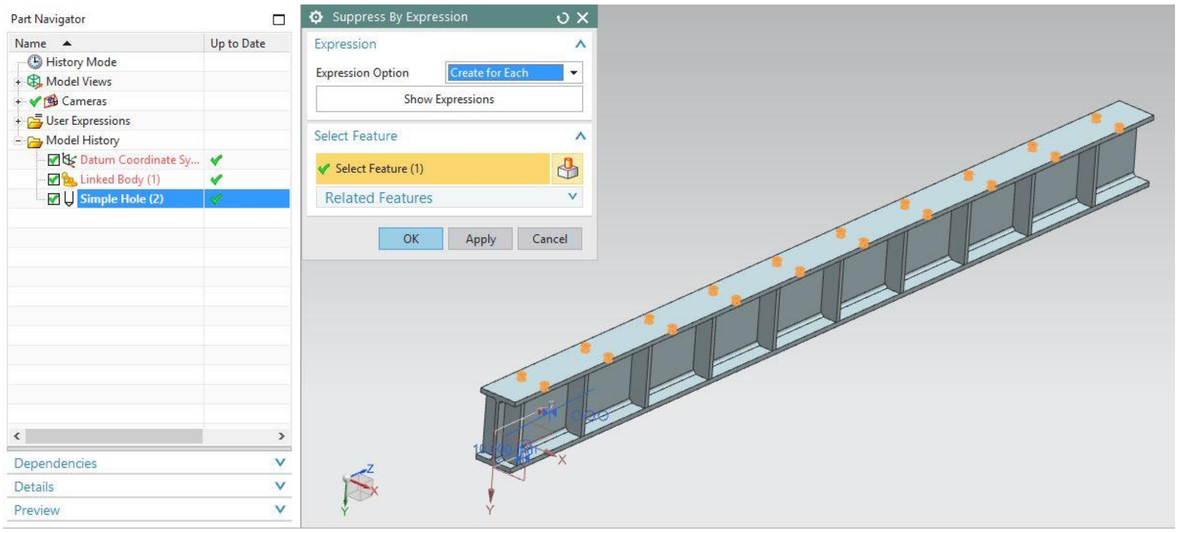


Figure 13: Holes are hidden with "Suppression by expression" command

6) Re-use library

The Re-use Library function allows the user to store standard parts, product templates, design features, curves etc. in an organized and accessible way for later reuse. [8] Re-using features and components reduces design time since the desired feature does not have to be designed again from scratch. The same library can be accessed by multiple users making sure the design process is following company standards and minimizes user interpretation regarding design attributes. If the designer notices something that is often used when designing, it could be useful to include it in a library for later reuse. [9]

7) Part families

When handling models that are mostly the same but have small variations Part Families can be a helpful tool. It is basically a spreadsheet with the design parameters listed, very helpful in our beam example, Figure 4. When the parameters are defined a spreadsheet can be created and the different parts designed very quickly. Note that a good parametric model must be created before using making a Part Family. [10]

DB_PART_NO	OS_PART_NAME	h	b	s	t	length
		80	46	3,8	5,2	1000
IPE	80	80	46	3,8	5,2	1500
IPE	100	100	55	4,1	5,7	2000
IPE	120	120	64	4,4	6,3	2500
IPE	140	140	73	4,7	6,9	3000
IPE	160	160	82	5	7,4	3500
IPE	180	180	91	5,3	8	4000
IPE	200	200	100	5,6	8,5	4500
IPE	220	220	110	5,9	9,2	5000
IPE	240	240	120	6,2	9,8	5500
IPE	270	270	135	6,6	10,2	6000
IPE	300	300	150	7,1	10,7	6500
IPE	330	330	160	7,5	11,5	7000
IPE	360	360	170	8	12,7	7500
IPE	400	400	180	8,6	13,5	8000
IPE	450	450	190	9,4	14,6	8500
IPE	500	500	200	10,2	16	9000
IPE	550	550	210	11,1	17,2	9500
IPE	600	600	220	12	19	10000

Figure 14: The beam example written as a Part family spreadsheet making different length beams

8) Product Template Studio

As seen in Figure 9, this Expressions window looks pretty straight forward. But what if there were several hundred parameters and the end user only needs to fill in a few? To avoid mistakes and make the use of the parametric model more streamlined Product Template Studio can be used. The designer can make a window that looks like the NX interface in which only the values that are needed occur, allowing for easy, accurate and fast usage. Product Template Studio also allows information to be accessed in the dialog box, for example a link to a design guide, saving time in searching for information. [11]

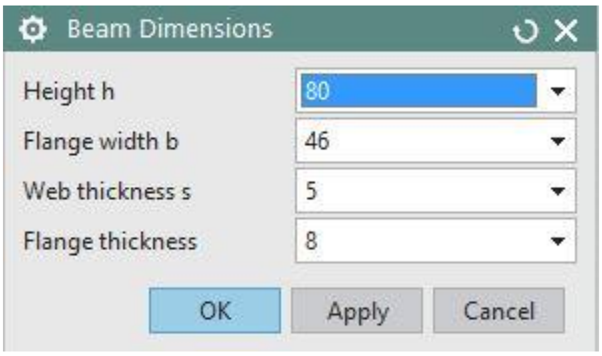


Figure 15: Dialog box made with Product Template Studio

IV. THE COMMON BASE FRAME

The common base frame is the structure on which the generator set is bolted on. Because every generator set is project specific, so is the common base frame. Factors that determine the dimension of the base frame are: engine model, oil sump dimension, flywheel and coupling dimensions, generator dimensions and external components. This chapter will describe the process and the parts needed to build a common base frame

Figure 16: Wärtsilä W20 generator set on a common base frame [12]

A. Internal order specification

The Internal Order Specification (IOS) contains project specific data about what the customer has ordered. From here the base frame designer looks for the type of engine and its specifications for example deeper oil sump (meaning a higher frame) and external components (wiring, piping etc.)

B. Center of Gravity calculation

In order to determine the dimensions of the common base frame a Center of Gravity (CoG) calculation is made using a calculation program. By inserting information about the engine model, generator, flywheel and coupling the program determines the length, height and width of the frame. When making a 3-D model of the frame the center of gravity is measured and inserted into the program. The locations for the rubber elements, lifting pins and jacking screws are iterated until they match the CoG tolerances and design guidelines.

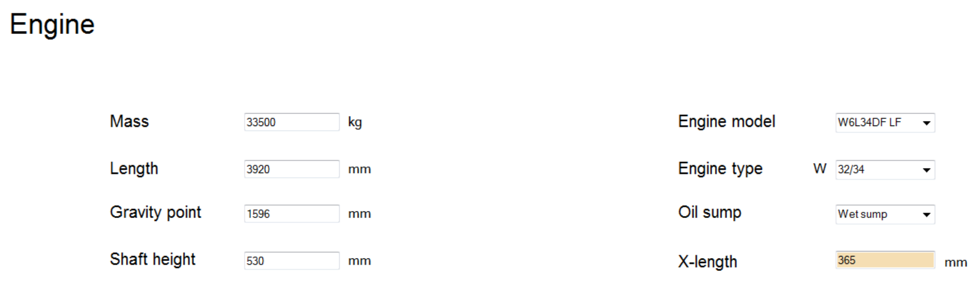


Figure 17: Information about an engine

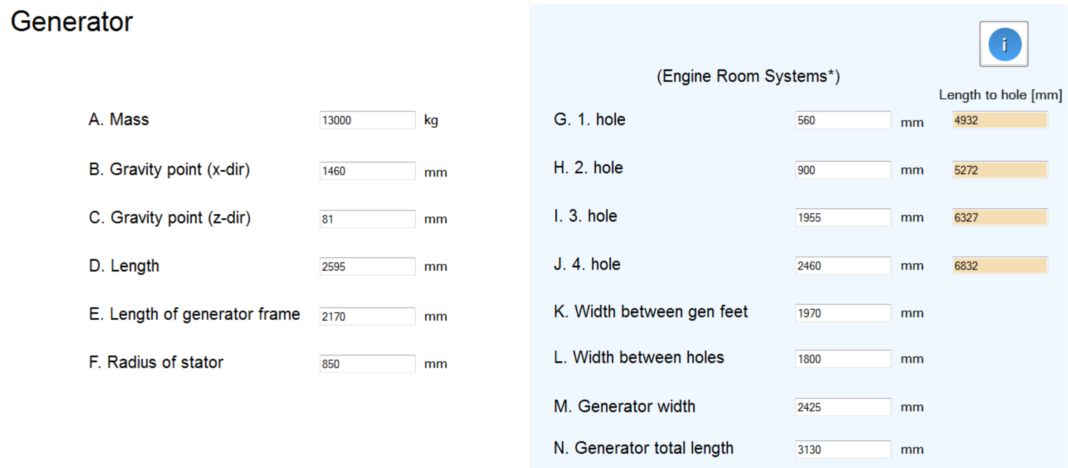


Figure 18: Information about a generator

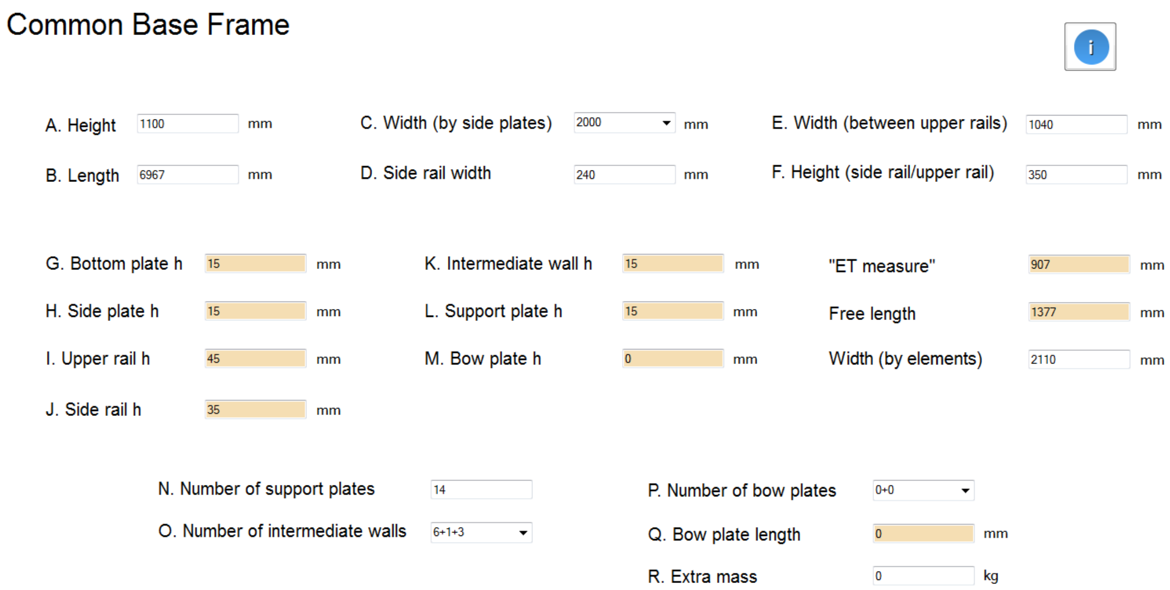


Figure 19: Calculated dimensions of a common base frame

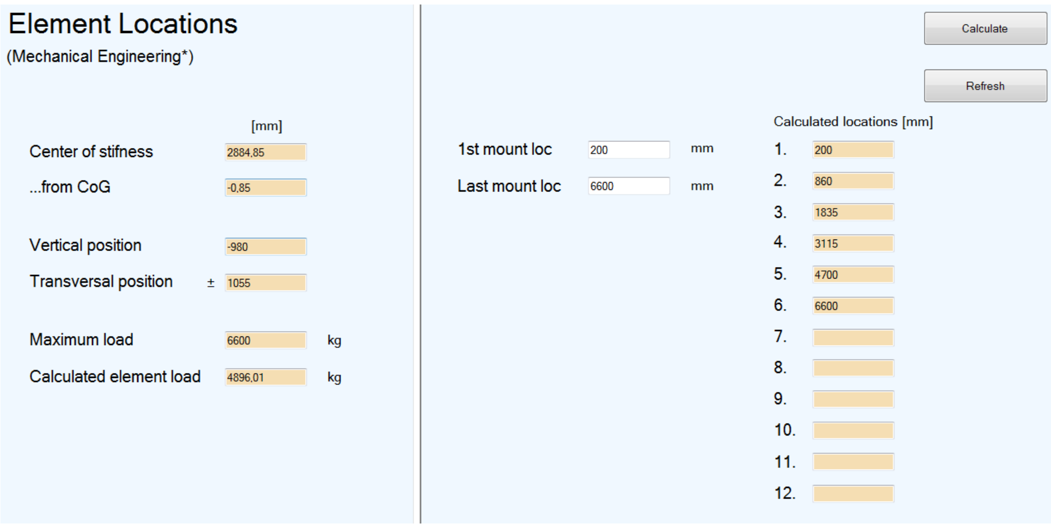


Figure 20: Element locations calculated

In figures 17-20 the user interface of the Center of Gravity calculation tool is shown. These are the values that the design engineer works with. The calculations for the lifting pins and jacking screws are similar to Figure 20.

C. Common base frame components

This subchapter will describe the components that are needed to build a base frame and how their dimensions are decided.

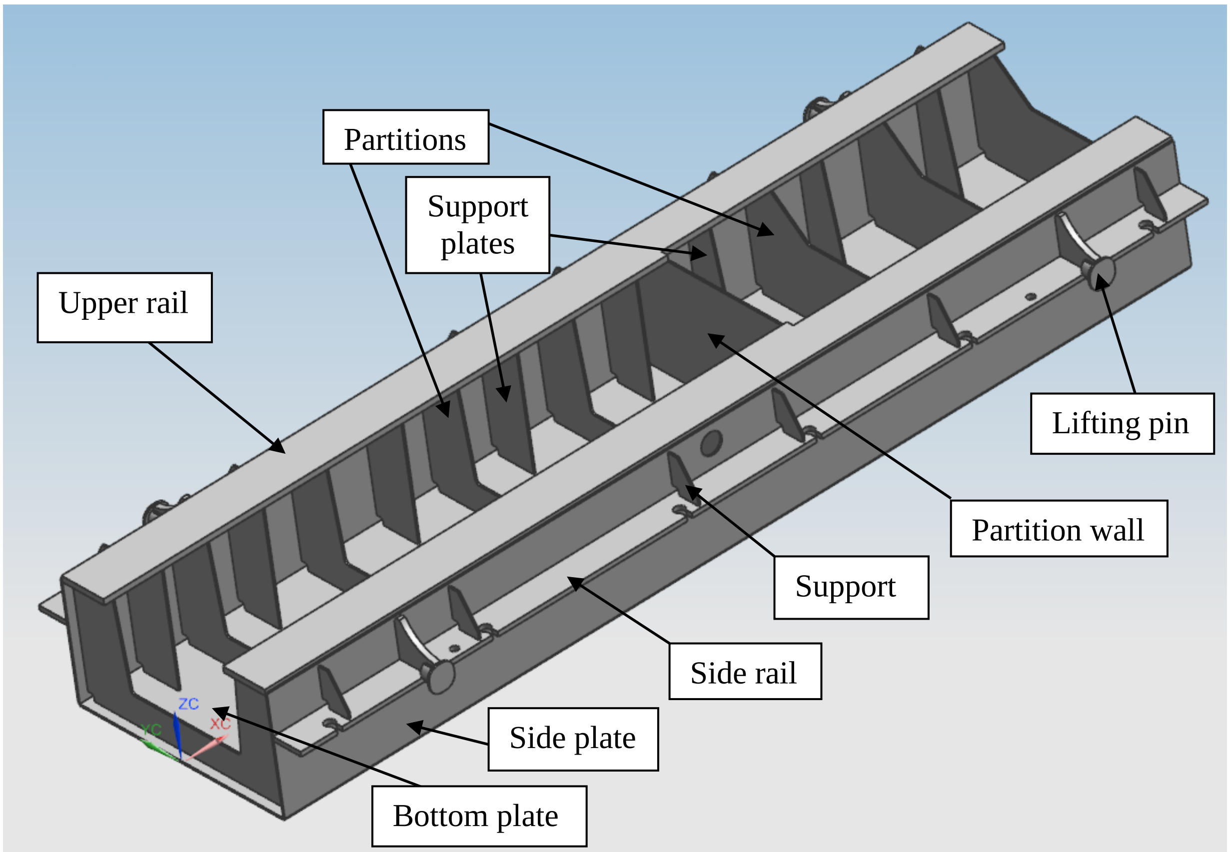


Figure 21: A 3-D model of a welded common base frame

1) Base plate

The base plate is a steel plate located at the bottom of the base frame. Its dimensions change for every project since the length and the width of the frame alters. The length of the base plate is the same as the base frame length. The width of the plate is left a little bit smaller than the base frame width for welding purposes.

2) Side plates

A side plate is located on each side of the base frame. A hole is made at a certain location depending on the engine model for assembly purposes. The length of the side plate is the same as the base plate, the height of the plate is a little smaller than the base frame height for welding purposes.

3) Side rails

The side rails are located on the side of the base frame containing cutouts for the rubber elements. Later holes for jacking screws will be drilled. There can be six or eight side rails per base frame depending on the number of lifting pins. The W6L32 has two lifting pins per side dividing the side rail into three pieces. The side rails are made of a thicker steel plate and the lengths are decided by the position of the lifting pins.

4) Upper rails

Two thick steel plates located perpendicular on top of the side plates. The engine and generator are bolted to these plates. The length is equal to base plate length, the width depends on base frame and generator width. The top of these plates will be machined to fit the engine and generator, not shown in Figure 21.

5) Partitions and support plates

Positioned inside the base frame for added structural stability. Dimensions must be such that they fit inside the tunnel formed by the baseplate, side plates and upper rails. There are normally four different partitions and supports on a frame. However, if the engine has a deep oil sump different partitions have to be used in order to fit the engine.

6) Lifting pins

Used when lifting the generator set. The CoG calculation gives the optimal positions for these. On a W6L32 generator set there are two lifting pins on each side, other models have three per side. Three pins allow the base frame to be lifted without a generator, this however causes problems with the W6L32 due to lifting wires damaging the side of the engine. Lifting pins are the same for each project.

7) Supports

Located on top of the side rails, a certain distance from the rubber elements, adding structural stability. There are normally one support per element, in this case six. More can be added if necessary. Supports are the same for each project.

8) Partition wall

Separates the engine side from the generator side, placed along with the partitions and supports. Other models require bow plates for added stability and these are welded to the partition wall, the W6L32 does not need these. If the common base frame´s weight is too low according to the design guide, additional weight can be added to the partition wall.

9) Bow plates

Some models require bow plates for added structural stability. These are bent steel plates fitted on top of the partitions closest to the partition wall. The number of plates is chosen from the design guide. The W6L32 does not have bow plates.

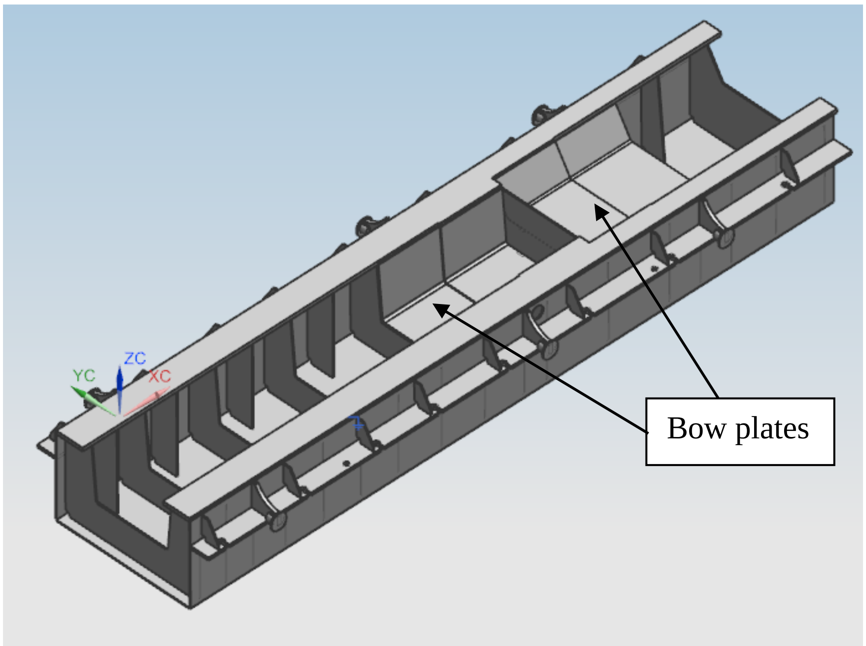


Figure 22: A base frame with two bow plates on each side

10) Other components

This subchapter will focus on external components that influence the design of a standard base frame.

The mounting holes for the engine are drilled along a milled pocket on the upper rail. Seven threaded holes and one general hole need to be made. There are standard hole locations for each engine model.

The flywheel cover is an arc formed cover mounted over the engines flywheel to protect its operators. Holes need to be drilled on a milled pocket on the upper rail to mount the cover to the frame. Five to seven threaded holes depending on the length of the needed flywheel cover.

Holes for grounding the engine are drilled on a machined pocket near the holes for the flywheel cover, six threaded holes.

Machined pockets are located at the generator hole positions. They are needed when mounting the generator on a set of Vibracon machinery mounting chocks. Some clearance must be left on the sides for minor adjustments.

In order for the generator to be aligned correctly, adjusting plates are needed. Holes for these are drilled near the generator pockets.

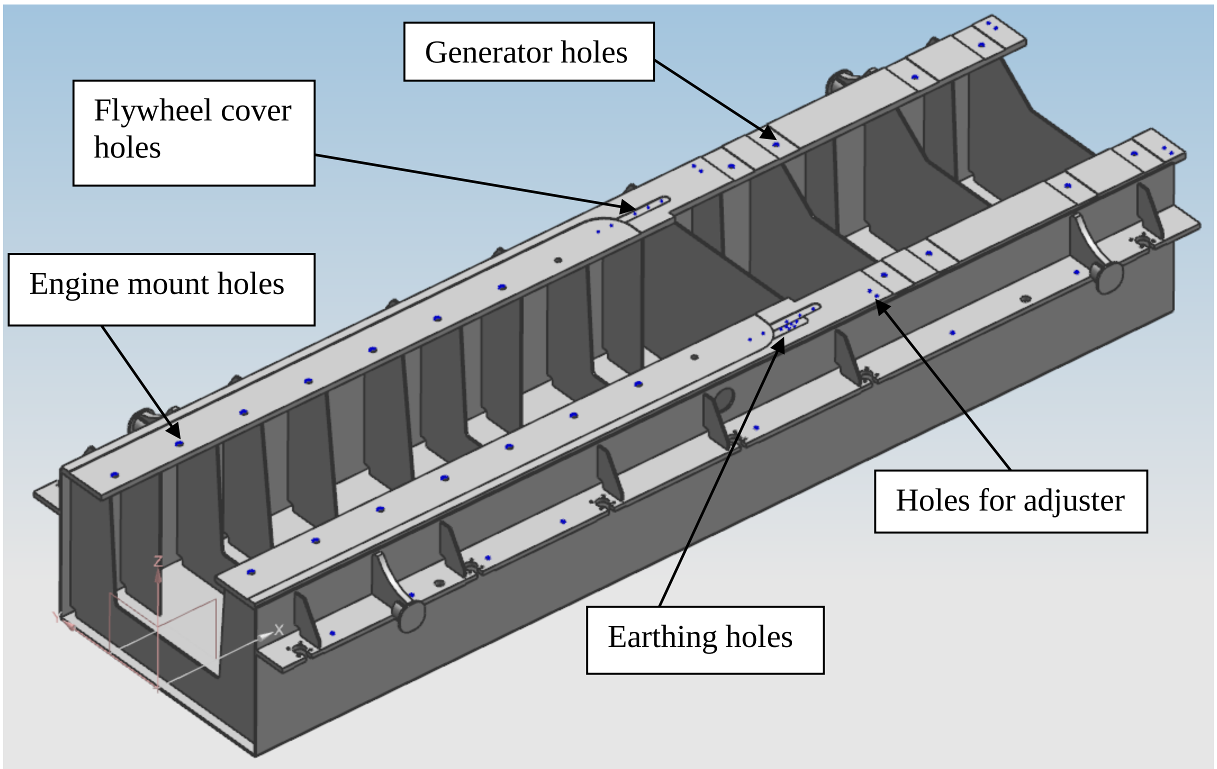


Figure 23: A machined common base frame

V. DESIGNING THE PARAMETRIC MODEL

This chapter will focus on how I built the parametric model starting from giving the individual components parameters using Expressions, hiding components and features with the Suppress function, putting the pieces together a welded and machined assembly and finally making a 2-D drawing of both assemblies.

A. Defining user inserted parameters

NX will automatically make a parameter in the Expressions window for each measurement for each part. The user however only needs a few of these parameters, so the important ones must be defined.

The important parameters and their impact on the base frame design are listed in the table below. To clarify; of course there are other things influencing the design of the parts. For example the engine mounting holes influence the design of the upper rail since they are drilled there. The point of this table is to define what changing dimensions influence which part. If I were to make a parametric model for all Wärtsilä engine configurations I would have to make a table of the engine hole positions and have NX choose from there. But since this thesis is for one engine type only the table is much simpler.

Parameters	**Influences the design of**						
Welded assembly	Base plate	Side plate	Side rail	Upper rail	Partition	Support	Partition wall
Length	x	x	x	x			
Width	x			x	x	x	x
Heigth		x			x	x	x
Element position 1			x				
Element position 2			x				
Element position 3			x				
Element position 4			x				
Element position 5			x				
Element position 6			x				
Jacking screw position 1			x				
Jacking screw position 2			x				
Jacking screw position 3			x				
Jacking screw position 4			x				
Jacking screw position 5			x				
Jacking screw position 6			x				
Generator hole width				x			
Generator hole 1				x			
Generator hole 2				x			
Generator hole 3				x			
Generator hole 4				x			
Upper rail offset				x			
Upper rail generator side width				x			
Lifting pin position			x				
Lifting hole 1			x				
Lifting hole 2			x				
Machined assembly							
Flywheel cover length				x			
Generator total width				x			
Generator length from edge to hole 1				x			
Generator length from edge to hole 4				x			

Figure 24: Parameters and their impact on design

B. Making the parametric parts

Now that we know what influences the dimensions of the parts, the parts themselves have to be made. I started by entering the parameters in Figure 24 into the Expressions window in the welded assembly. I went on by making the individual parts from scratch and linking their dimensions to the parameters in the welded assembly using Interpart links (see chapter 3.3.3).

The base plate and the side plates were made in a similar way. A rectangle was drawn and the two variables stated in Figure 24 were made to define its dimensions.

The side rails were a bit trickier to make. First of all, there are three unique pairs of side rails on each frame and their length depend on the lifting pins position. Side rail 1 length is now defined as a measurement from base frame origin plane to the lifting pin plane minus half the thickness of the lifting pin. Side rail 2 length is the distance between the two lifting pin planes minus the whole thickness of the lifting pin. Side rail 3 length is the measurement from the second lifting pin plane to a plane at the end of the frame. The element cutouts were positioned according to the inserted values in the welded assembly. A Suppression by Expression was added to the cutouts: if a cutout is further away than the length of the side rail it is suppressed and the cutout is placed on the next rail, this was done with an “if” statement. As seen in the figure below the first cutout´s “if” statement gives the value 1, which means it should be active, the second has the value 0 and is suppressed.

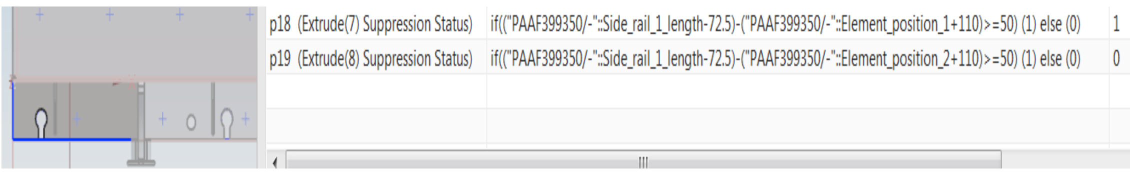


Figure 25: Logical conditions for the element cutouts

The upper rail length is the frame length plus the parameter “upper rail offset” that can be added if necessary. The engine side has a different width than the generator side depending on the generator feet width. For engine models with bow plates there would also be cutouts to fit these.

The partitions, support plates and partition wall were not made from scratch, but taken from old projects. If I had to make new parts from scratch Part families would be useful. Since there are standard dimensions for a frame, old parts can be used. This was solved by loading each possible part into the welded assembly and using Suppress by Expression. Only the ones that fit the current frame remain visible.

C. Placement of the components

When placing the components into the welded assembly the user would normally give constraints between parts to make sure they are placed correctly. My colleague Aki Laurila was making a similar model for the W20 base frames and advised me to define different planes and constrain the components to them rather than actual parts. He found that this is the best way to avoid NX getting confused when parts change in dimension. I have had similar experiences in previous projects so this approach was chosen.

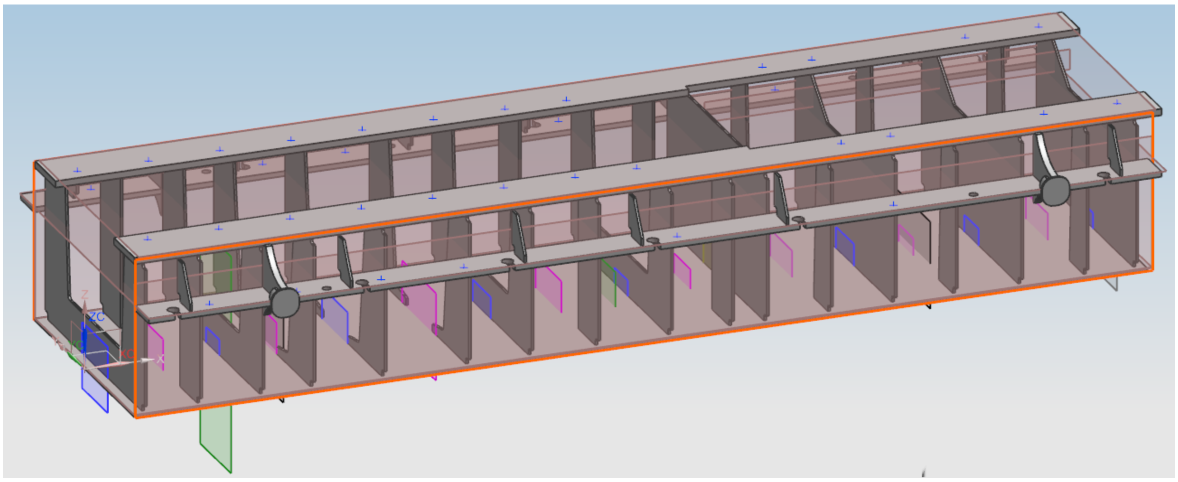


Figure 26: Different planes

In Figure 26 some of the planes are shown, the blue ones are partitions, the pink are supporting plates and the green on the left is the lifting pin distance from the center of gravity, the green in the middle is the distance from frame coordinate system (located at the left bottom of the frame) to the center of gravity. The highlighted orange planes represent the side plates, all the partitions and supporting plates are constrained to fit perpendicular to this plane. A similar plane is made for the base plate and every other major part. The planes themselves are positioned using basic parameters as length, width and height combined with mathematical expressions to place them according to the design guide.

By constraining parts to planes the user can easily move around the partitions and supporting plates without having to open the Expressions window and figuring out how much some part has to be moved. Simply double clicking on the desired plane and dragging it along the frame to place it where it fits the best, even aligning planes with one and other. A good example of this is the fact that the lifting pins need a partition or supporting plate immediately behind them for structural stability. Simply place the lifting pins at an ideal position and drag the partition or supporting plate closest to this location so they align perfectly.

D. Making the machined assembly

When all the parts are placed in the right position it is time to prepare for the machined assembly. First it must be made certain that the frame follows the design guide. There can for example not be an engine hole directly on top of a partition, it would destroy the partition. A certain distance is required and to make this easy all the drilled holes are displayed as a sketch on top of the upper rail. After making sure everything is according to the design guide a new part was made parent to the welded assembly, the machined assembly.

Almost every machined feature is somehow linked to an Expression in the welded assembly, for instance generator hole positions or element locations. This allows the user to only insert the value once and typing errors are minimized. The only features that are not linked to an old expression is the flywheel cover length, lengths from the generators holes to its edges. The flywheel cover length is decided when a 3D-model of the generator is imported to the assembly to check for a suitable flywheel cover from a range of standardized ones.

My first idea to do this was to make a Linked body as described in chapter 3.3.4. My colleague Aki Laurila gave me advice to just import the welded assembly into a parent assembly and do all the machining there. The machining does show on the welded assembly, but not in the welded assembly drawing. The user loads the machined assembly, the welded is there as a child component, the user saves every parametric part (base plate, side plate, side rails and upper rail) under a different name.

E. Making the 2D drawing

The 2D drawing is quite simple, needed measurements are added to both the welded and machined drawings. After looking at some old drawings I realized it would be good to make sure there is room to add more views, especially to the machined drawing in case there are more features added to the frame.

One problem I encountered with the welding drawing is when the automatic parts list was made. At Wärtsilä NX stores all the parts in a software called Teamcenter. In Teamcenter you can add data for the specific part such as weight, material, and intended use. When making a drawing and calling for the automatic part list NX lists every object in the assembly. Since I have loaded all the possible partitions and supporting plates into the assembly, NX displays them in the parts list though they are suppressed. My colleague Aki Laurila advised me to manually suppress the parts not being used since we don´t know another way. Another issue with the welded drawing was adding the length of the welds. I tried making this automatic by linking the weld dimensions to expressions in the model, that didn´t work for some reason, so the weld lengths have to be added manually.

F. Product Template Studio

When a parametric model is made it is time to design the user interface. The user needs no programming experience due to the simplicity of Product Template Studio. Simply load the created model, choose which parameters should be visible in the window, drag and arrange them, give them proper titles and choose whether they are key in values or chosen from a list etc. When right clicking the part in NX an option called “edit reusable component” opens the window you have created.

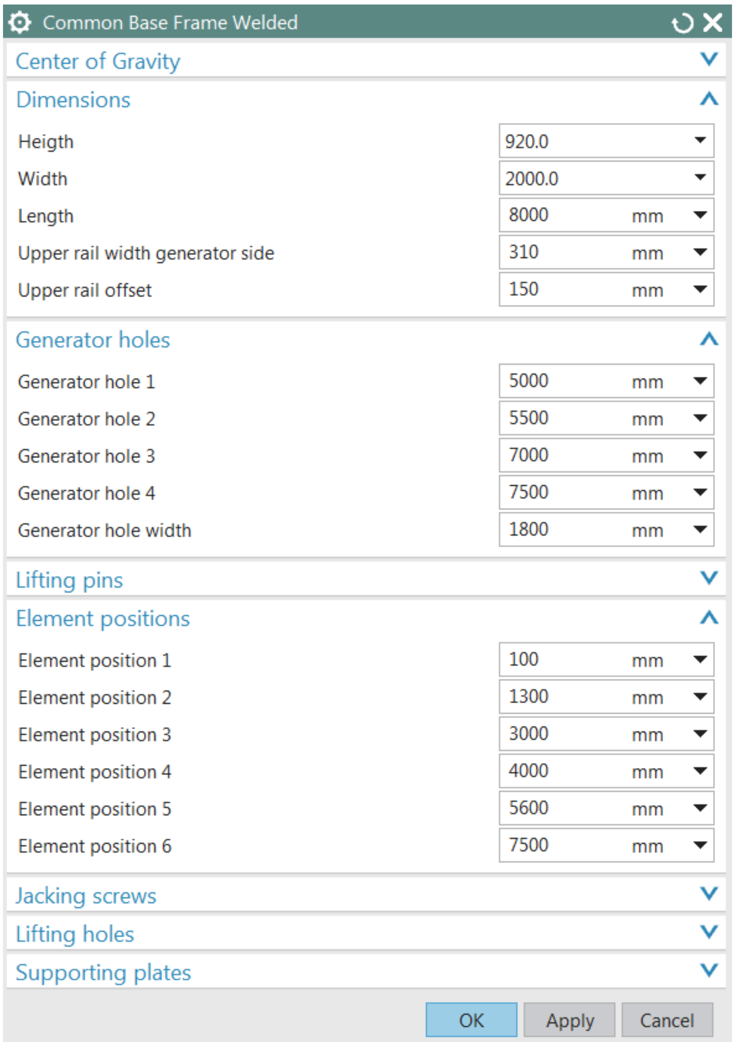


Figure 27: Parametric model user interface made with Product Template Studio

G. Requirements

As mentioned in 3.3.2, a model can be equipped with different design conditions called Requirements. Taking an example from the design guide: the distance between a partition or support and an engine hole must be a certain range of distances. This can be checked by adding a length measurement as a variable and giving that measurement a range of values that defines when it is good. Requirements were made for each measurement stated in the design guide. In the requirements window the current expression is shown allowing the user to see why it failed or got a warning.

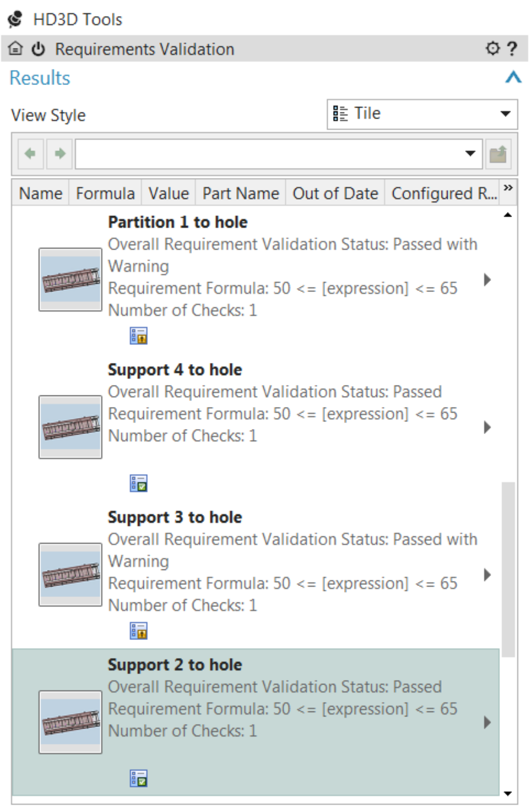


Figure 28: Requirements showing failed and passed requirements

VI. Result

This chapter will describe the working process of the model, the testing process, input from colleagues and the future of this model.

A. The working process

A manual of how to use the model is found in Appendix 1. The process was made fairly simple. The designer need to prepare by filling in the CoG calculations with ideal values. These values theoretically fill the design requirements, but the calculation tool does not know if for example parts collide or come too close to each other.

After making the parametric parts project specific by saving them under a new part numbers the designer inserts these theoretical values into the dialog box in the welded assembly. By clicking “apply” the designer can instantly validate the created model and alter it if needed. Next step is to check that the Requirements listed are met. After this the machined assembly´s dialog box is given the project specific values and the machining is done.

When the 3D-model is within design guidelines it is time to make the drawings. Two template drawings were made, one for the welded and one for the machined assembly. The designer has to save this template drawing under a new drawing number and load the newly made part into the drawing. If everything goes as planned the old measurements attach to the new model, but often this is not the case. After some manual work re-attaching all measurements in both drawings the welded assembly´s automatic part list has to be modified. As said, it lists all the 43 parts that are loaded in the model, even if they are suppressed in the model. This can be changed by editing a suppressed parts properties setting it to be “reference only”. This process is quite quick since multiple parts can be set at once. Once this is done the drawing is complete.

B. Testing the model

An old project was chosen for testing purposes. Following the process described in the previous subchapter some minor features did not work as intended and had to be remade. After fixing these issues I could focus on the process as whole. I noticed that this model is much quicker than the old way of working, which meant editing every feature in every part separately, a very time consuming process to say the least. My colleague Xiaotian Bi told me it takes about a week to make a new frame. My model has not been tested time-wise, but design time can be reduced by at least several days since the parts update quickly and the user does not have to edit any sketches or other features.

C. Colleague input

A meeting was held in 2.3.2016 where I presented the model to my colleagues and asked for their input. The participants were my supervisor Tomi Vesterbacka and my co-workers Xiaotian Bi and Aki Laurila. During this meeting I walked them through the design process and they got to ask questions and give advice on how to improve the model and way of working. My colleagues were very impressed with the model. They were positive to its effects on reducing design time for the common base frame. Some improvements were brought up: the measures in the drawings must be fixed so they work even when the model changes. We speculated that the measurements could be defined by planes rather than actual part geometry, but this has to be investigated further. One aspect that also has to be improved is when making the first iteration with the theoretical values from the CoG calculation. If these values don´t work the calculation needs to be changed, meaning inserting new values into the calculation tool, updating the tool, inserting these values into the model, validate the model and possibly go back and make more changes. This kind of back and forth flow of information is exposed to risk of typos and other human errors, the designer must be very careful with this step. My colleagues were however satisfied with the model as it is we agreed the improvements will be made in the summer of 2016.

D. The future of the model

My team thought this kind of development in the design process was very good and they wanted to take this even further. A W6L32 project had just come in and Xiaotian Bi will be testing the model on that project. After that it can be properly evaluated by someone who does not know the model inside out. The model should be so easy to use that you do not have to know how it was built, just how to use it to do your tasks. We also agreed on that my summer will be spent expanding the model for W7L32, W8L32 and W9L32 engines. After making sure that parametrization works for these engine types, the v-motors will get their own models.

VII. Conclusion and reflections

Looking at the problem definition and delimitations and evaluating the result I would say I met the set goals. The tool creates a basic model ready to be modified according to customer demands. Interaction with the model is easy, just open one menu and alter the values. The fact that I get to expand this model to other engine types tells something about the success.

Finding information about parametric modelling was both easy and hard. I found very little theory on the basics of parametric modelling, and the little I found was mainly written by people in architectural or software design. I decided to include the architectural parts, since it is easier for a mechanical engineer to understand. Finding practical examples of parametric modelling was easy, Ally PLM´s webinars make it easy to understand the model designing with practical examples. YouTube also contains lots of hands on examples with varying quality.

When making a parametric model one must have a very well-structured way of working, this cannot be said enough. It is a good idea to name all the different features that you are working with. It takes time, yes, but it will be worth it if and when you have to troubleshoot your model to find out why it is not working like it should.

I have no idea how widely spread parametric modelling is in today's industry, but I guess its importance will increase even more in the future. When talking to people about parametric designing they are somewhat afraid. "No one has to know what lies behind the solution, you just enter values without knowing what you are doing". This is not the point of parametric modelling. It is about capturing the knowledge that your team has and putting it into effective use eliminating "manual" tasks. Computers are good at calculating and evaluating, but someone has to teach them how and make sure they do it right. This is what people should do. People should do what machines cannot: study, read, investigate, come up with new ideas and leave the manual "boring" labor to the computer. By good documentation knowledge will stay within the team even with people changing jobs. I like to think that if we don´t do this someone else will, and then we are facing bigger problems than not knowing exactly how a model is built.

List of sources

[1] Wärtsilä, “About: Wärtsilä,” 2016. [Online]. Available: http://www.wartsila.com/about. [Accessed 15 January 2016].

[2] R. Woodbury, S. Williamson and P. Beesley, “Parametric Modelling as a Design Representation in Architecture:,” Canadian Design Engineering Network, Toronto, 2006.

[3] L. Lin and L.-C. Chen, “Journal of Engineering Design,” 2002. [Online]. Available: http://portales.puj.edu.co/jaguilar/Constraints.pdf. [Accessed 2 February 2016].

[4] Wikipedia, “Wikipedia: Parametric design,” 2014. [Online]. Available: https://en.wikipedia.org/wiki/Parametric_design. [Accessed 2 February 2016].

[5] M. Loew, “CAD/CAM: enigneering.com,” 2013. [Online]. Available: http://www.engineering.com/DesignSoftware/DesignSoftwareArticles/ArticleID/6576/Bottoms-up-or-is-the-Top-down-A-Working-Definition-of-Systems-Engineering.aspx. [Accessed 15 January 2016].

[6] R. H. Shih, Parametric Modeling with UGS NX 6, 1:st ed., Mission, Kansas: SDC Publications, 2008.

[7] Wärtsilä, *Instruction for Wave Links (Power Point presentation),* 2013.

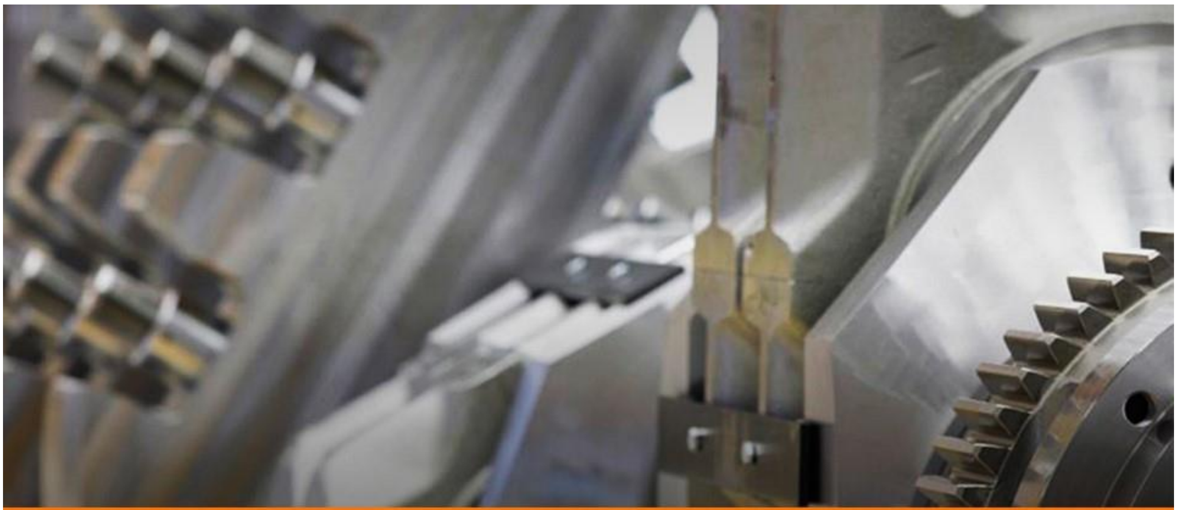
[8] Siemens PLM Software, “NX Re-Use Library,” [Online]. Available: http://www.plm.automation.siemens.com/en_us/products/nx/for-design/knowledge-re-use/library.shtml. [Accessed 18 January 2016].

[9] Ally PLM, *Lunch Bytes - Reuse Library (webinar),* Ally PLM, 2013.

[10] Ally PLM, *Ally PLM Lunch Bytes - Part families (webinar),* Ally PLM, 2014.

[11] Ally PLM, *Ally PLM Lunch Bytes - Product Template Studio (webinar),* Ally PLM, 2014.

[12] Wärtsilä, “Wärtsilä Generating sets,” 2016. [Online]. Available: http://www.wartsila.com/products/marine-oil-gas/engines-generating-sets/generating-sets/wartsila-genset-20. [Accessed 2 February 2016].



MANUAL FOR CBF PARAMETRIC TOOL

Filip Slotte

WÄRTSILÄ

Preparations

Have the following things filled in and ready:

- CoG calculation
- Generator drawing

- Please note that this currently works for W6L32/34 (26.2.2016). More configurations will come.

WÄRTSILÄ

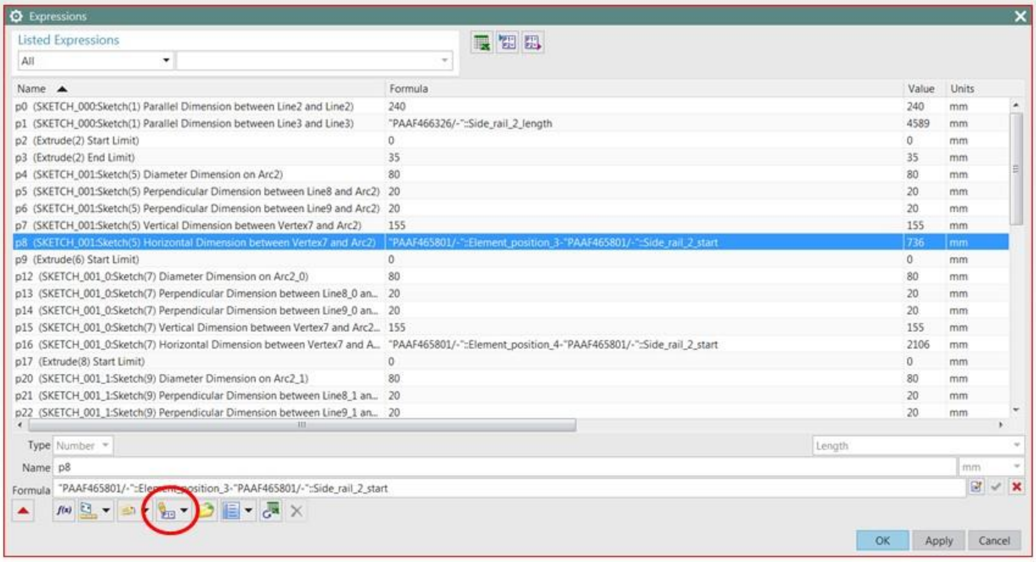
New part

Do like this:

- Open PAAF409578
- Make PAAF399350 unique, save PAAF409578 as a new part
- Make the following components unique: baseplate, sideplate, side rail 1, side rail 2, side rail 3, upper rail

- Make the new baseplate work part
- Open its expression window (menu, tools, expressions, edit multiple interpart expressions)
- Make sure that everything is linked to your welded assembly part number
- Do this for sideplate, side rail 1, side rail 2, side rail 3 and upper rail
- *This was intended to be automatic, but didn´t work for the individual parts*

Changing expressions



Under this menu, choose "edit multiple interpart expressions". Change to your welded assembly part number.

Tips and tricks

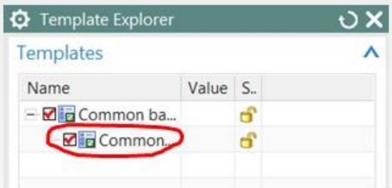
How the model is designed to work

- Try to keep the lifting holes on side rail 2 and/or side rail 3
- Side rail 1 should have 2 elements
- Side rail 2 should have 3 elements
- Side rail 3 should have 1 element
- If a lifting pin comes too close to an element it will suppress it
- Adjustment of the generator is now 5mm along x-axis and 5mm along the y-axis

Welded assembly (1/5)

Do like this:

- Right-click Machined assembly and select "Edit reusable component"
- Double click this ->



- Fill in all the values according to CoG
- *Note that you must calculate "ideal" positions first, then check it in the model so there is no collision between components*
- *All the holes are shown as a sketch for easy evaluation*

Welded assembly (2/5)

The menus:

Center of Gravity		
Distance to CoG	3550	mm

Insert the calculated CoG

Dimensions		
Heigth	920.0	
Width	1800.0	
Length	7247	mm
Upper rail width generat	200	mm
Upper rail offset	50	mm
ET measure	937	mm

Enter height, width and heigth

Decide upper rail width on generator side and upper rail offset

Generator holes		
Generator hole 1	5000	mm
Generator hole 2	5500	mm
Generator hole 3	7000	mm
Generator hole 4	7500	mm
Generator hole width	1800	mm

Enter the distances to generator holes from CBF origo and hole width

Welded assembly (3/5)

The menus:

Lifting pins		
Distance from CoG	2650	mm

Enter the ideal positions

Element positions		
Element position 1	100	mm
Element position 2	1300	mm
Element position 3	3000	mm
Element position 4	4000	mm
Element position 5	5600	mm
Element position 6	7500	mm

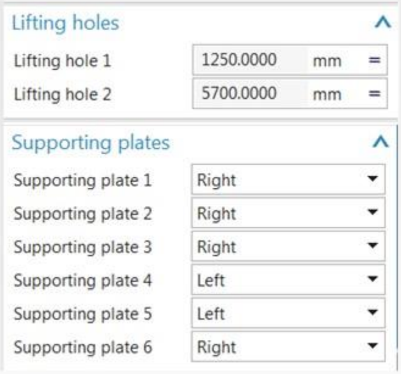
Enter the ideal positions

Jacking screws		
Jacking screw 1	200	mm
Jacking screw 2	1000	mm
Jacking screw 3	3500	mm
Jacking screw 4	4500	mm
Jacking screw 5	6000	mm
Jacking screw 6	7800	mm

Enter the ideal positions

Welded assembly (4/5)

The menus:



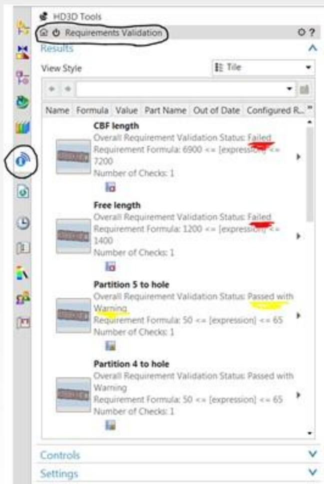
Enter the ideal positions. You might have to click the "=" icon and choose "make constant". Can also be controlled by moving a plane. If you want to move them by planes, don´t touch this.

Decide on which side of the element the supporting plate should be

Welded assembly (5/5)

Requirements:

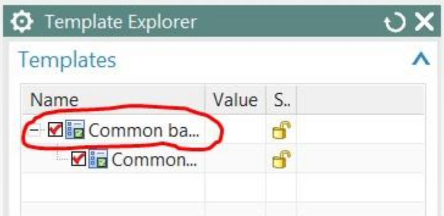
- Open the shown tab
- Choose "Requirements validation"
- Here design requirements are shown
- These are mainly partition distances to engine holes
- Move the components until requirements are met (partitions, supports, partition wall)



Machined assembly (1/2)

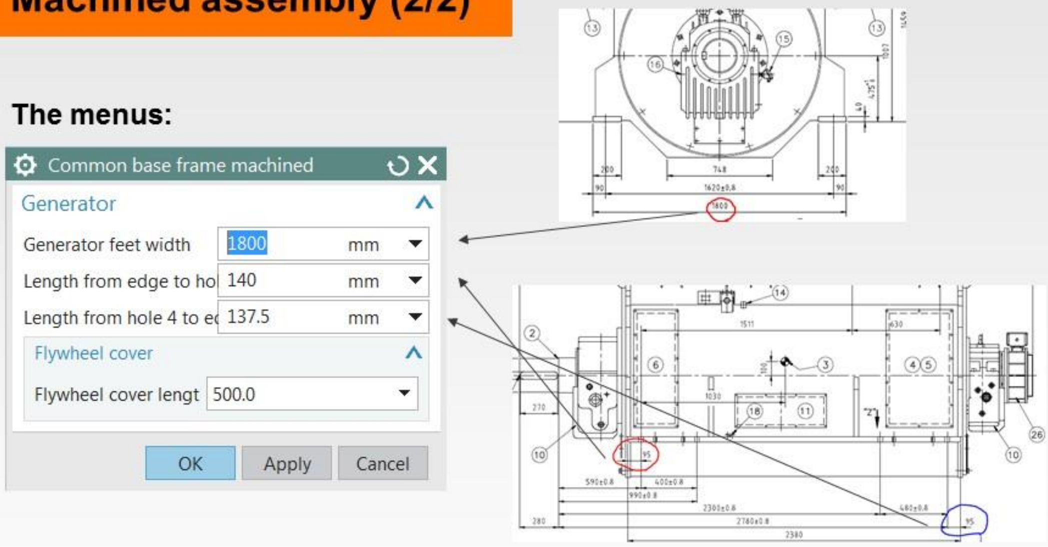
Do like this:

- Make the machined assembly the work part
- Right click and open ”Edit Reusable Component”
- Click this ->



Machined assembly (2/2)

The menus:



Making a drawing

Do like this:

- Open DAAF314360 (welded drawing)
- Save it as a new drawing
- Replace the drawings part with your new welded assembly
- Follow the same procedure with DAAF32102
- Measurement may have to be re-defined manually

Welded assembly part list

Do like this:

- Select all the parts that are suppressed
- Right click->Properties->Component is reference only