White-paper on the 3D-graphs and their generator - both known as AnRep3D

Written by Guus Löhlefink,
founder of AnRep3D

Table of Contents

Introduction

AnRep3D is a rather old concept as we developed it at the start of this millennium, but it only became available as a real software-package about three years ago. Back then it wasn't completely mature yet. It took several blog-posts, for which the generator was used, to understand which improvements were absolutely necessary. The AnRep3D-geneator is still not a fancy app for consumers and it won't be. Yet it is a powerful tool for professionals in companies. Sometimes the choice was really between a user-friendly interface and a quick instrument and often the latter was chosen. This doesn't mean that we think AnRep3D-generator cannot be improved. We will use the feedback from our paying customers to improve the generator and perhaps even the graphs.

In this stage of its lifecycle it seemed wise to publish a more comprehensive overview of the 3D-graphs and their generator. This white-paper combines all kinds of examples from our blog, but in a more structured way. We hope it will provide you as a reader with a better insight and help you and your company to use AnRep3D to gain competitive advantage.

1 What is AnRep3D and where did it come from?

To answer the question first, before deep-diving into the background:
AnRep3D is a type of graph, showing (financial) data as a three-dimensional image. The name is an abbreviation of "Annual Reports visualised in 3D-graphs".

At the same time, these graphs have to be created and this is done with the help of a 3D-graph generator. This generator is also called AnRep3D.

Finally, the new company offering the generator (and therefore the resulting graphs) is also called AnRep3D.

If you are not interested in our original dream and the journey we took to get where we are, please skip the remainder of this chapter and move to Chapter 2.

1.1 Why do we need 3D-graphs?

Back in the seventies, graphs were either made with the help of a ruler and a pen/pencil or printed as a collection of ASCII-signs on paper. Then, in the eighties Lotus and some other programs also had – rather complex - options to show graphs. Then something new came up called "Harvard Graphics". It was a small revolution when we suddenly were able to create real, detailed graphs (line, bar, pie and many others) within seconds! The personal computers often needed a special graphical (Hercules) card to be able to draw the graphs on the screen. After a while colour came in. Then Excel pushed the other spreadsheets out of the market and offered even fancier graphs (shade, depth) but without adding any real functionality to the graph itself. As a matter of fact our data exploded exponentially, but we are still looking at the same flat images as we had forty years ago. Even modern programs for visualisation hardly explore the third dimension. We still use series of graphs to investigate something that could be presented in one single 3D-graph! Gamers already embraced 3D-environments, but businesses are rather conservative. AnRep3D wants to change this.

1.2 How it started

I remember visiting the "reality cube" of the University of Groningen in the year 2004 and being really impressed by what I saw. I asked the people there reality cube if it would be possible to visualise abstract data e.g. from measurements. Actually, this question was inspired by Michael Ridpath's novel "Trading Reality", which I read a couple of years before. They told me they sometimes did visualise such data, but it took the team a couple of days or weeks to prepare new software showing those data. This meant that such a single 3D-graph would cost EUR 100 * 40 * 2 * 2 (tariff per hour * hours per week * number of team-members) = over EUR 15000 or, with a more modest fee and less time: EUR 50 * 20 * 1 * 1 = EUR 1000 for a very simple graph (estimation as from 2020). What if it would be possible to provide an automated process and create such a graph in seconds, virtually without costs?

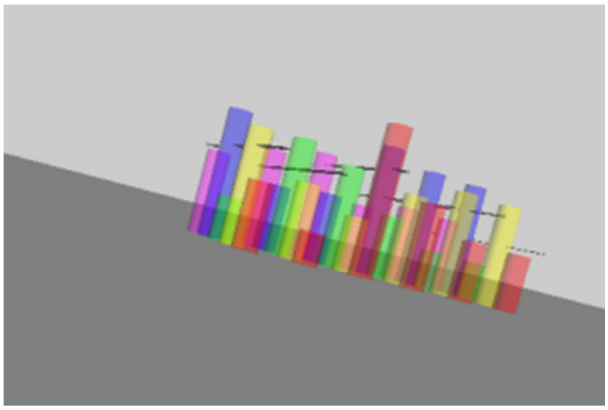
When I got back at the incubator where I was working at the time, I told everybody about the potential of an automated conversion of data into 3D-graphs instead of this lengthy process they told me about. No human labour, just an input-file and an output-file within seconds if we automated the process. Nobody understood why 3D-graphs would add business value (think about the first

comments on mobile phones: useless, no added value, no-one needed such a device, on sewing machines or penicillin - pretty much the same) and actually they even didn't understand what was meant by a 3D-graph (and until now, luckily even an application like Excel still doesn't understand).

1.3 Prototyping and failing

It became clear to me that the only way to educate "my market" was to come up with a prototype but I'm not a software engineer, although I learnt some programming. My company, SCIENTASSIST, had been active for over a decade with all kinds of innovative products. 3D-graphs could be the next innovation. Seeing no alternative, I mastered VRML (Virtual Reality Markup Language – later on called Virtual Reality Modelling Language) as much as I could. Back then this language was the most common one to visualise 3D-shapes on a computer-screen. By the way: don't confuse a "language" with a software application. VRML was rather similar to the original version of HTML and XML, using "tags" to tell the computer what had to be done. For technical background, see Chapter 7.
The first attempts were neither visualising abstract data nor creating the graph in an automated way. The only objective was to make people enthusiast about 3D-graphs. It looked like the screenshot below (of course the real version was in 3D, but this cannot be printed to the pdf-format).

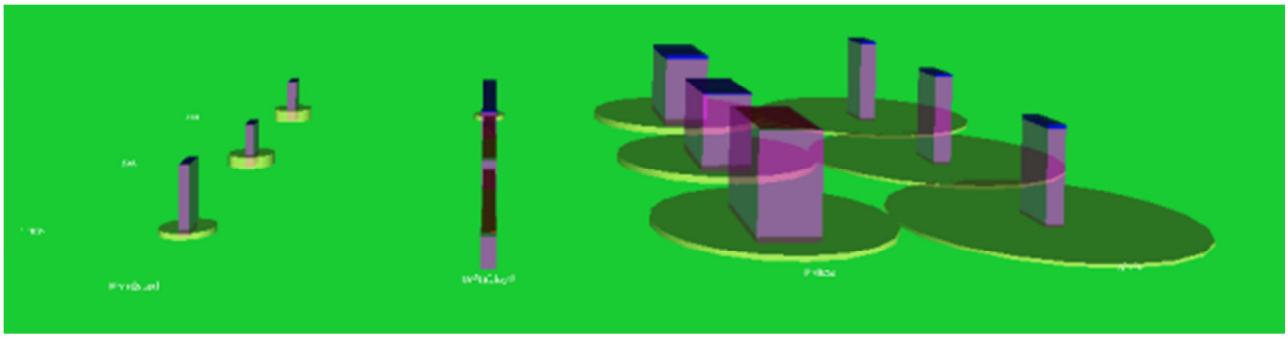
After a while I was able to create a generator in QuickBasic that converted numbers into a 3D-graph in VRML. It worked, but then I discovered al kinds of flaws and started improving. VRBI (Virtual Reality Business Intelligence) was born! My dream was to create a real VR environment, where one could walk literally through a graph. A more modest first step was to use stereoscopy. This time I failed completely as I really don't understand hardware at all. I asked others for help, but the market kept changing at such a high speed that a solution wouldn't survive for over a year and then nobody will pay for it, unless updates keep coming. In the end I had to abandon stereoscopy when the CRT-monitors became obsolete and the flat-screens became the standard. At some point the idea was even to offer a service at a physical location, instead of a licenced software-package, but this would have been a violation of the original plan: give companies full control of their own visualisation.
Although I describe the process in a couple of sentences, it was a painful and frustrating journey which took about two years. At this moment the visualisations were already focusing on financial data. Below a screenshot of the data from a stock-exchange with different funds (the columns) and a related index (the hovering boards).
Again, the real graph was in 3D, but neither a 3D-graph nor an embedded movie can be printed to the pdf-format.



1.4 Entering the 3rd millennium

After a couple of years I decided to start again, taking smaller steps. The objective was to show a 3D-graph on the screen just in the way the popular games did. That was what I was able to realise with some effort. While I was struggling with the technology, several browser plugins for VRML were terminated. Most standalone viewers were incomplete and VRML's successor (X3D) never became popular, so I ended up in a vacuum again!

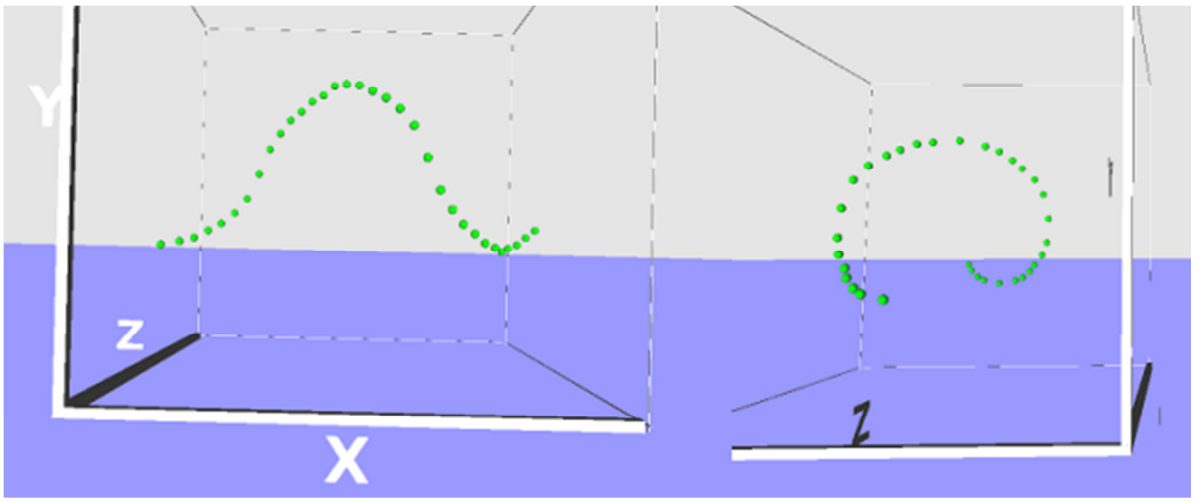
There was one good VRML stand-alone viewer left, but (at least back then) it was hard to work with and rather unstable. I didn't expect my audience (primarily business people) to work with this kind of solution, so eventually I moved to HTML5, the modern Internet language. As a matter of fact, this new version uses JavaScript – something I don't understand very well. Yet I wanted to generate a graph like shown in the screenshot below. A type of graph I already developed in VRML.



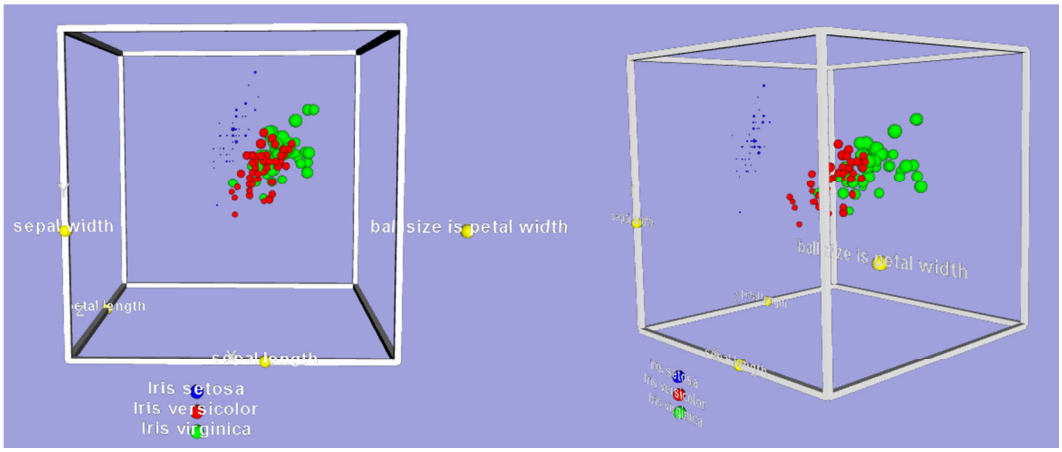
Long story short: fortunately a kind of cloud-service by Fraunhofer came up (called X3DOM), translating my X3D-like scripts into JavaScript on the fly. Finally I was able to jump into the future! We worked together in a team to create a Java-framework for input/output. I leant a bit of Java myself (beware: Java has nothing to do with JavaScript at all - as only the names sounds similar). My knowledge was just enough to re-write all the internal calculations and the X3D-like output.

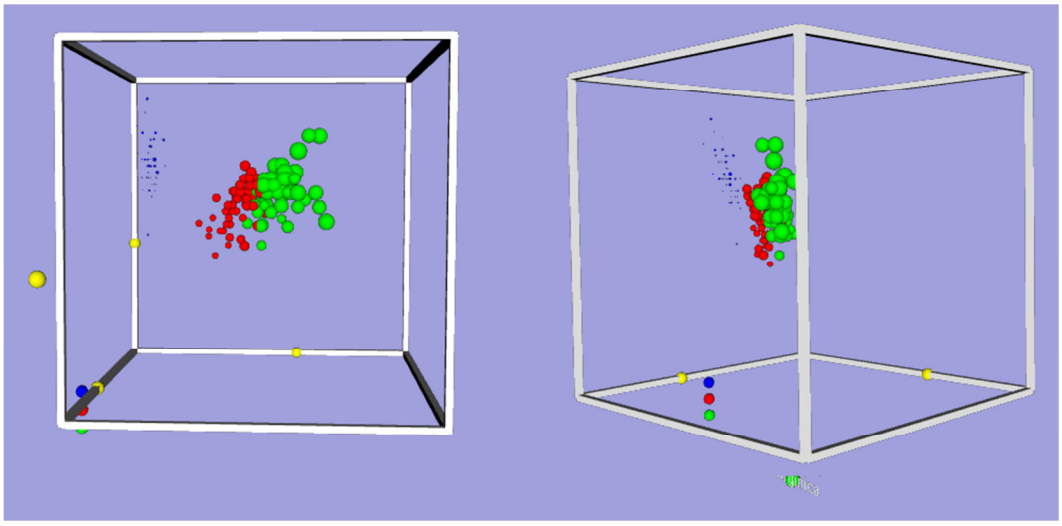
The new type of graph I developed seemed to be a bit of a niche-product. To push the principle of 3D-graphs, it seemed wise to create something easier to understand. The 3D-version of a scatterplot would do the job and so VRBI (Virtual Reality Business Intelligence). Such a type of graph is very useful to show "clusters". Of course a series of flat scatterplots can be used to see clusters from different perspectives, but ***one single 3D-graph*** will replace a dozen of those as the graph can be

rotated to get views from any angle. Below a screenshot (again neither 3D nor animated) is shown. To give a better impression two different views are combined.



Recently after discovering the famous "Iris dataset", I created another VRBI-graph. This graph shows clearly what the power of VRBI is in cluster-analysis. Below a four screenshots from different angles, meant to give an impression of what the real 3D-graph will look like.





1.5 AnRep3D was born (again)

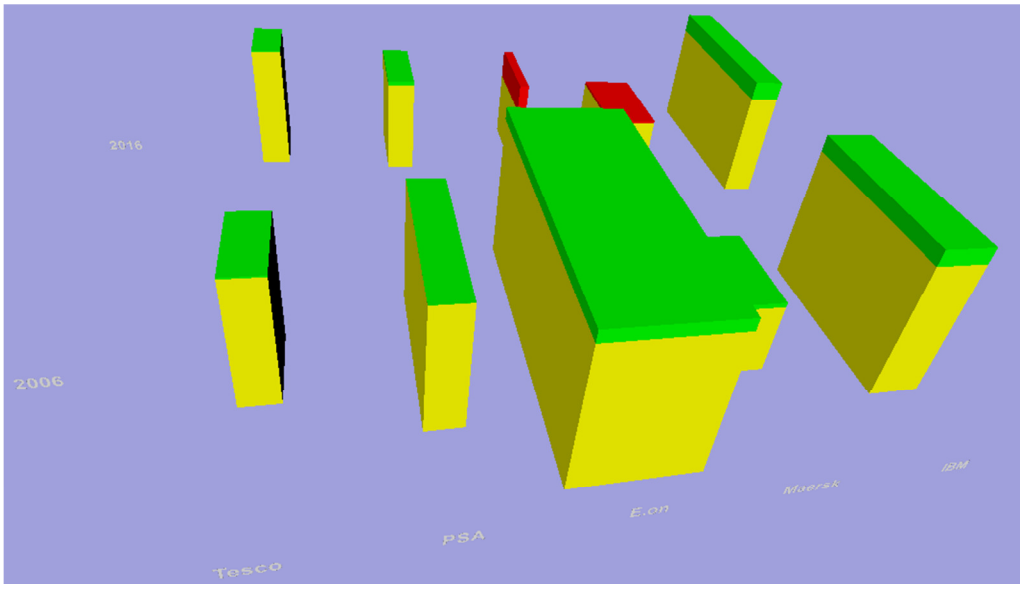
This white-paper is not about VRBI so it will do to explain that in the end I handed over company (SCIENTASSIST) and product (VRBI) to one of my sons.

I went back to my original idea with the help of my sons and narrowed it down: visualising data from Annual Reports for a group of companies through several years (or quarters). After modernising the old attempts, we came up with a more mature product generating HTML5-files as output. This meant the graphs can be shown in a standard web-browser and here we are.

Again, the really technical information is saved for Chapter 7. Chapter 6 explains the generator, but in the next chapters we will fist explain something about the graphs themselves and show a couple of use cases.

2 The heart of AnRep3D: Ratios

An AnRep3D 3D-graph consists of “buildings”, ordered in a Manhattan- or chessboard-like pattern. The shape of a single building is determined by its height, width and depth. This means AnRep3D-graphs are all about ratios as height, width and depth are all determined by the input variables. Every single “building” has its own sizes in three dimensions and therefore it offers three ratios: height/width, height/depth and width/depth. Because all “buildings” in the 3D-graph have their own set of ratios, the ratios can also be compared between the “buildings”. Since these buildings can represent e.g. companies in one direction and year or quarters in the other one, a lot of comparisons between ratios can be made, showing something we could consider as “meta-ratios”.



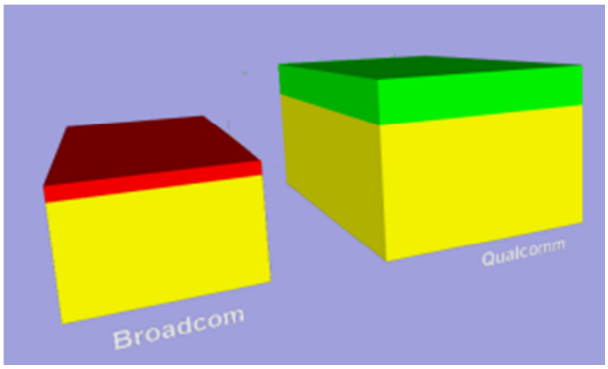
2.1 Original approach

When we started with the AnRep3D-graphs and the generator, we used the “***RPEA***-series” as the input (RPEA is the abbreviation of ***Revenue***, net ***Profit***, ***Equity*** and total ***Assets***). The ratios between these values can be compared between the companies, but also throughout the years or quarters.

Warning: *There is a risk when looking at the graph. It is a 3D-graph, but that doesn’t mean the* ***volume*** *is meaningful. The three dimensions should be compared separately or in combinations (as shapes), but not surface or volume!*

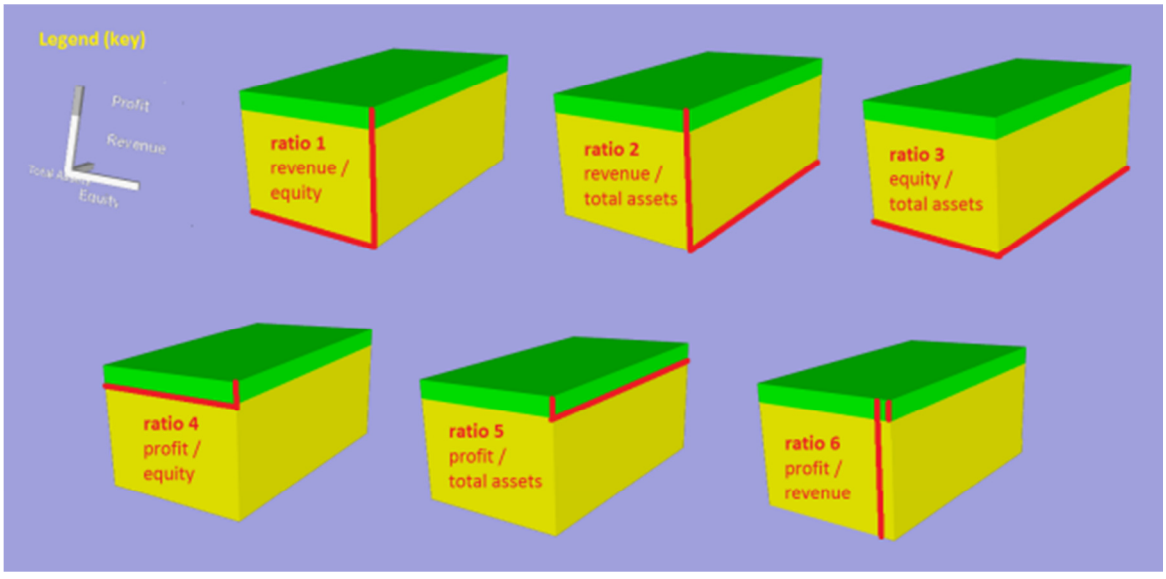
E.g. the shape of the floor indicates a kind of gearing ratio (here Equity/Total Assets, but Equity could be replaced by Liabilities easily to get other ratios but of course this would apply to all companies and all years). A company about ten times smaller than another would be smaller in all three dimensions, but this doesn’t mean it’s 1000 times smaller as the volume would suggest.

R-P-E-A is about four values but we only talked about Height, Width and Depth of the buildings. Yet there is a fourth value shown in the graph as the buildings also can have a **"roof". The top-part of the building will be green, representing the profit-part of the revenue. Sometimes however, there is no profit. In case of a loss it will be *added* to the Revenue as a red "roof" on top of the revenue.



In our standard-approach the yellow part of the height represents the total cost part of the revenue. In case of a loss, this comes on top of the Revenue as the total costs were higher than the Revenue and the total height of the building actually shows the total cost.

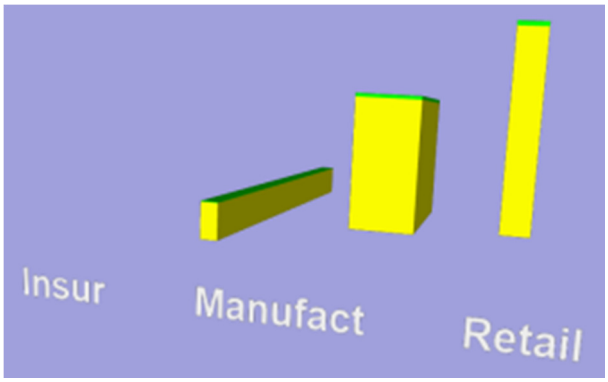
Because of this roof we can distinguish even more ratios. Using the RPEA-series (Revenue, net Profit, Equity and total Assets as input-variables) the return on equity would be the shape of the green rectangle in the front-view. Looking from aside the green rectangle there will show the return on assets. Below an impression of the different ratios in such an AnRep3D-graph is shown.



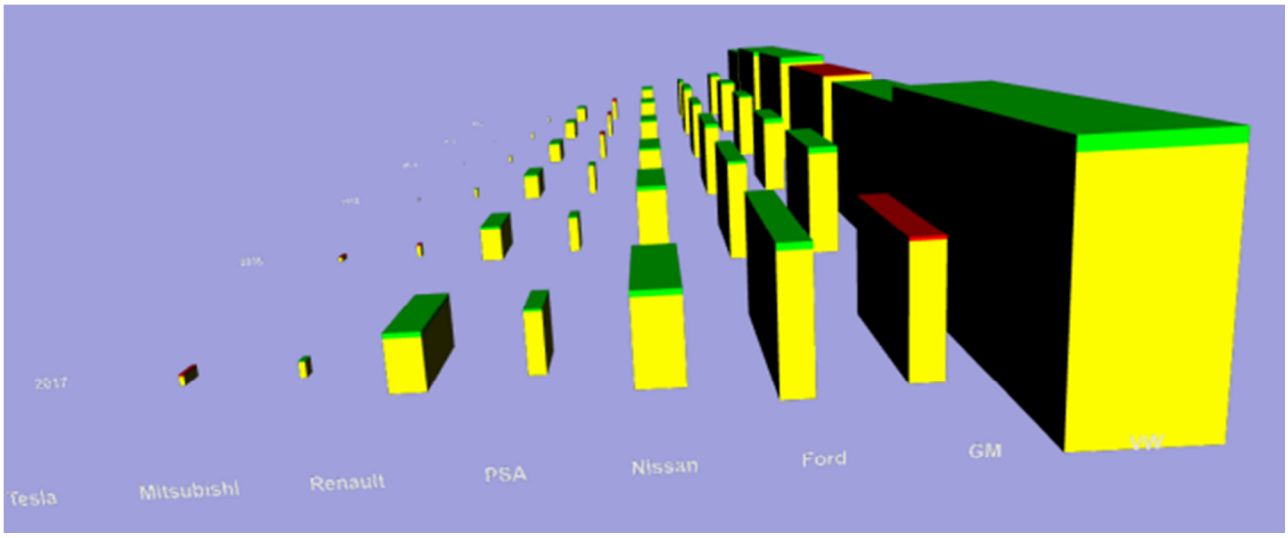
One can see: AnRep3D is a very powerful instrument to interpret a lot of different values at a glance!

It turns out that similar industries often have similar financial ratios as well when looking at equity, total assets and revenue – with profit and loss being rather different. Comparing industries like manufacturing (automotive, machines, consumer goods), insurance and

banking (no examples needed) or more service-oriented companies (like temp agencies and retail) the difference is impressive. Below a partial screenshot of a graph comparing different industries is shown.



The overview of a series of automotive-companies would be a good example of comparison ***within*** a specific type of industry.

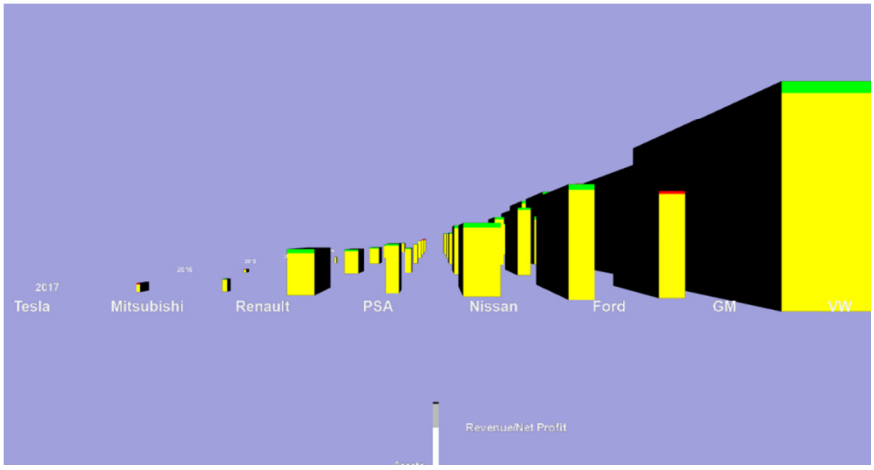


2.2 Real 3D-graphs are available

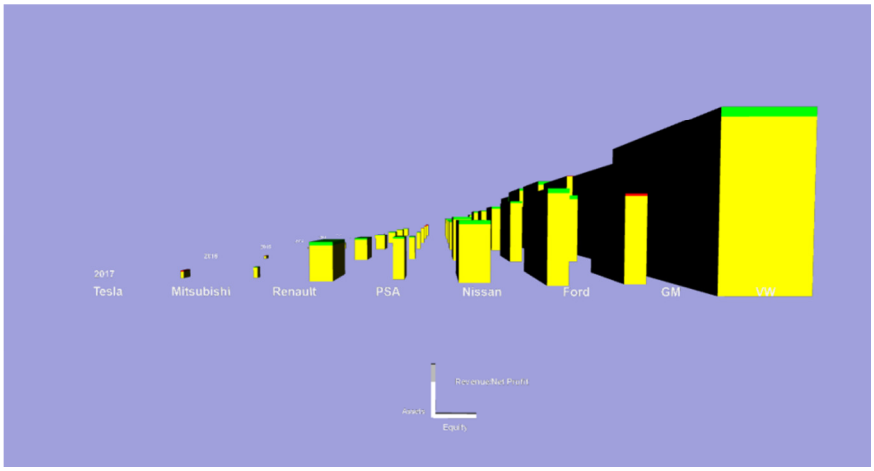
Of course screenshots are a very poor representation of the real 3D-graphs. That's why this white-pater comes with a couple of real 3D-graphs discussed. The format is .htm meaning the graphs can be viewed in normal web-browser. An online connection is needed because a cloud-service (X3DOM) is used. Open these graphs by selecting them in the file-explorer and double clicking.

2.3 Manipulating the 3D-graph

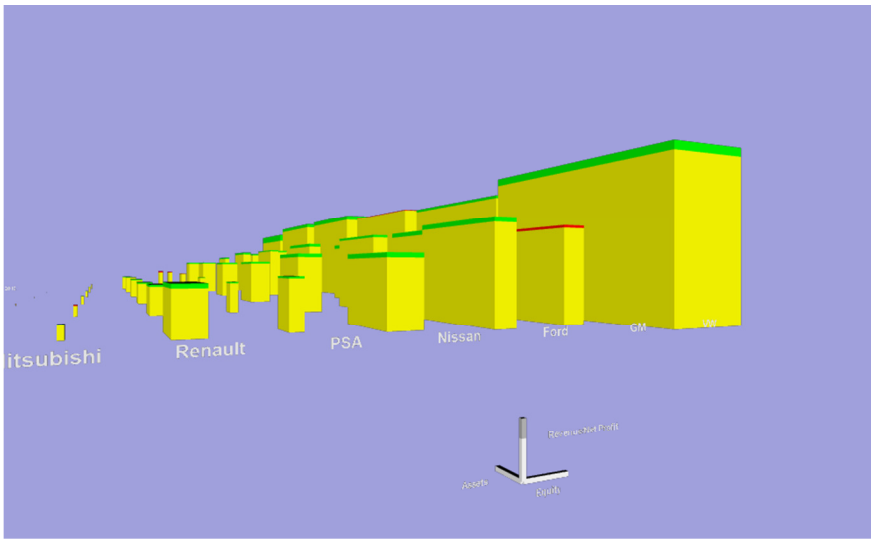
To manipulate such a 3D-graph some exercise is needed. Below a rather large 3D-graph is shown, holding the series of automotive companies for a couple of years. It is a large graph and as the legend (key) is a little bit in below the graph and to the front it will be helpful to ***zoom*** out.



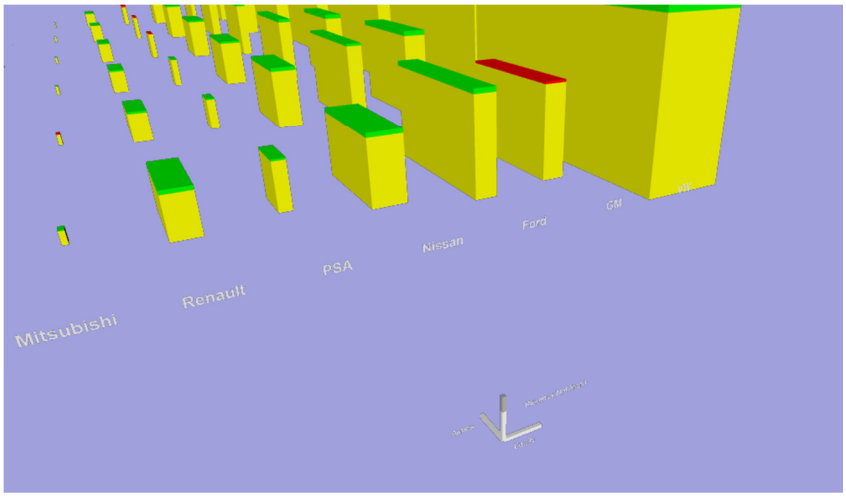
To do this, hold the right mouse-button while moving the mouse up (down would zoom in). The image will be like the pictures below (the legend is completely visible now).



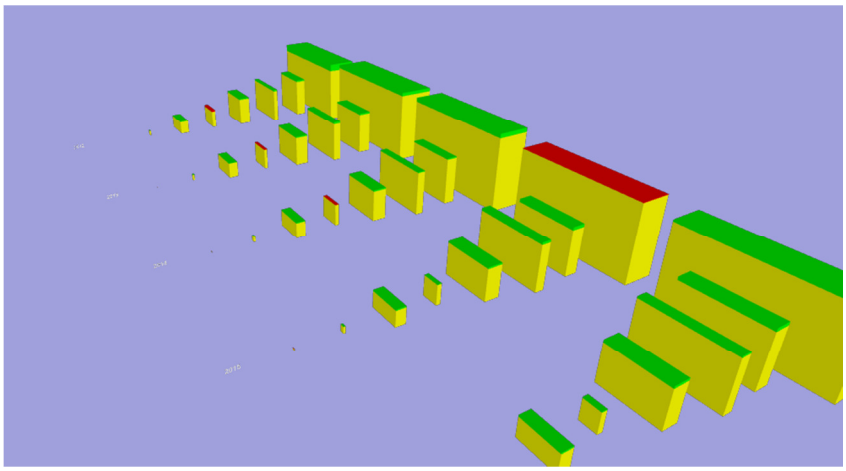
To rotate or tilt the graph, just click and hold the left mouse-button while moving left, right, up or down. Moving the mouse to the left rotates the graph counter-clockwise.



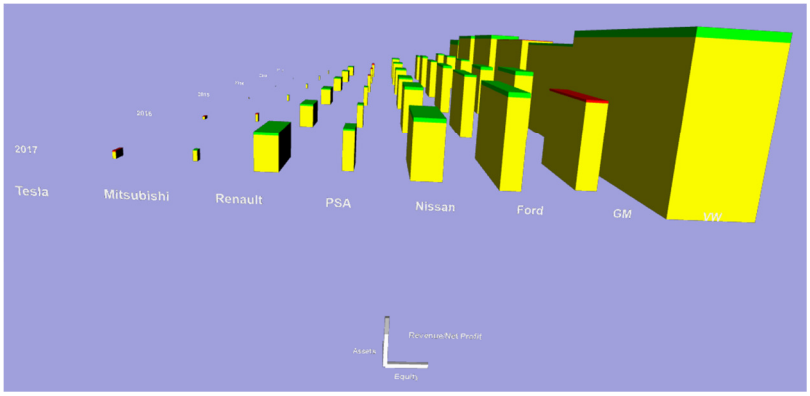
Moving the mouse up after moving to the right (rotating the graph) will tilt the graph as well. The result is shown below.



Of course all kinds of combinations of zooming (in, out), rotating and tilting can be applied, but one other adjustment is very important: **re-centring**. By double-clicking at a point somewhere in the 3D-graph, the centre will be redefined. The screenshot below was taken after a double-click somewhere in the top-left corner when the situation was as in the screenshot above.



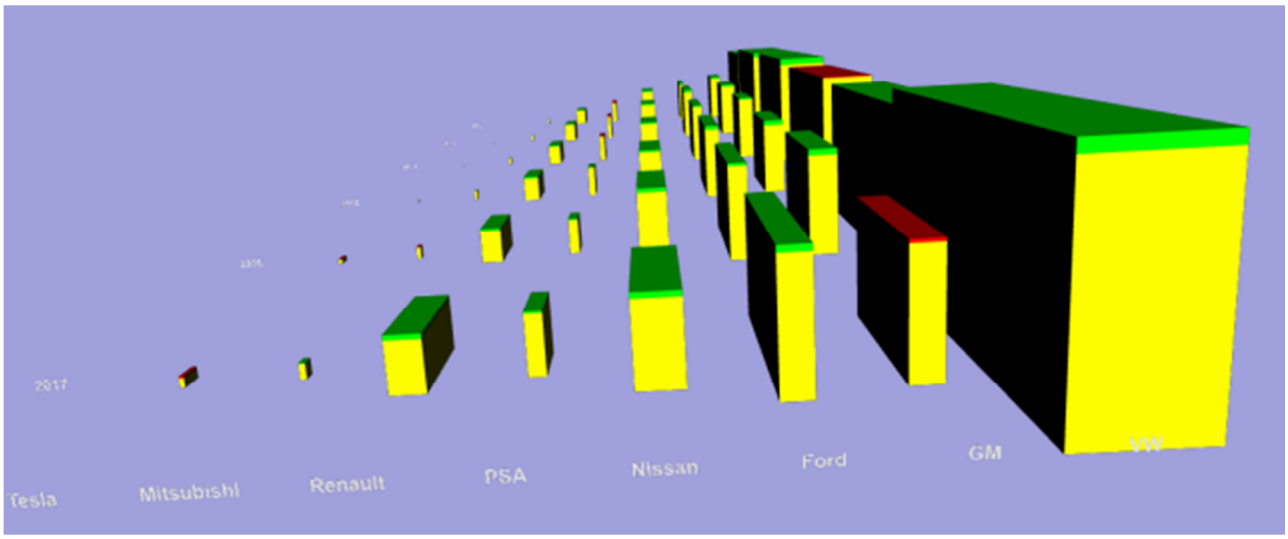
After several combinations of zoom, rotating, tilting and re-centring, the screenshot below was taken.

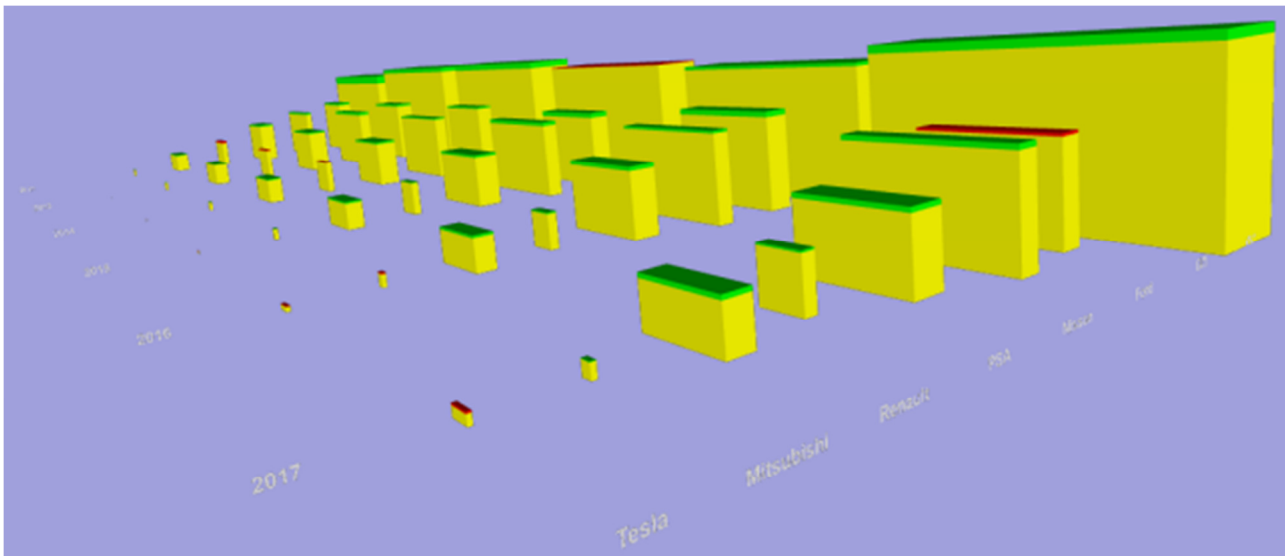


3 Financial applications

3.1 Standard input

A good example of the original use of AnRep3D would be the 3D-graph for automotive (showing Revenue and net Profit as height and roof, Equity as width and total Assets as the depth of the buildings). Creating this graph can be a bit of a challenge of course as broken book-years and different currencies. The latter can easily be solved with the help of exchange-ratios.
Here, ten different automotive-companies are presented with their values from 2012 to 2017 and ordered from higher to lower revenue. Below two screenshots are showing two different angles of the 3D-graph.

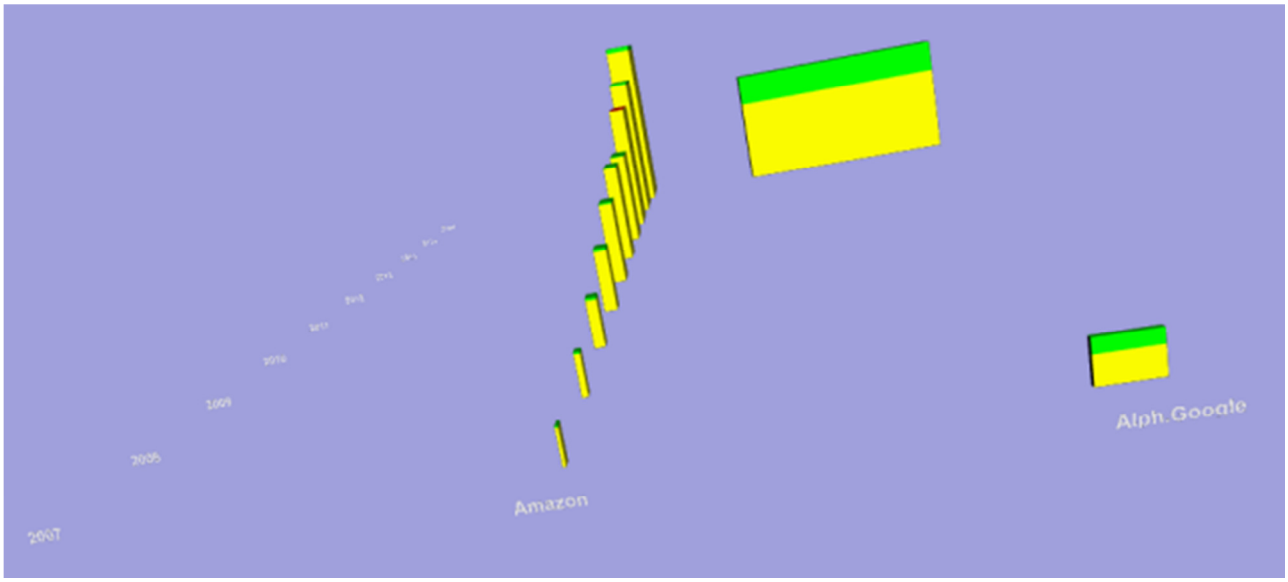




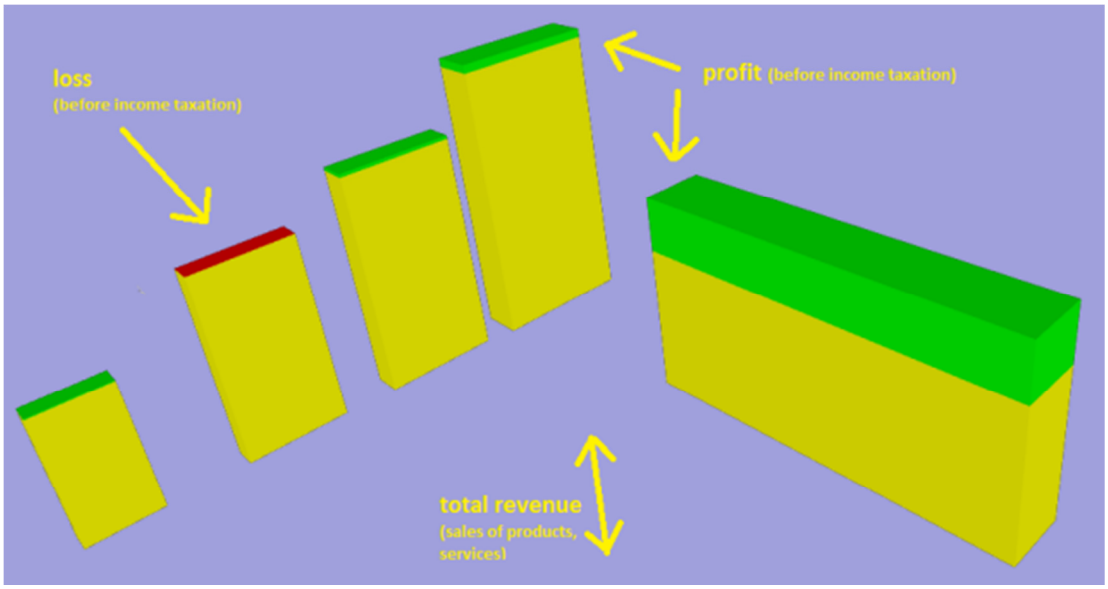
3.2 Alternative values for the "roof" and other aspects

This time the graph is showing Amazon over a period of ten years (2007 – 2016). Another company was needed as a reference for start and end of the period only. It could have been any large company, but Google (2006 - later becoming Alphabet 2016) is often referred to when talking about the new Tech Giants. As two years would do, all other values (2007 – 2015) were put in as zeroes. Instead of Revenue, (net) Profit, Equity and (total) Assets (abbreviated as RPEA), here Revenue, *Profit before taxation*, Equiy and *(total) Liabilities* is presented. Using "profit before taxation" (or gross profit, operating profit or EBITDA) instead of net profit with usually show a thicker "roof".

Replacing Total Assets with Liabilities is a bit odd. We know Equity cannot exceed Total Assets as it is a part of it (after subtracting the Liabilities). Yet the Equity can be higher than the Liabilities although it won't happen very often, but Google/Alphabet turned out to be an example! The "building" is perpendicular to the common direction (for RPEA).



Be aware that this graph isn't just a simple bar-graph in 3D, as *all sizes do have a meaning*. The bottom-view shows the changing width and depth, representing the equity and total liabilities throughout the years. Another screenshot shows revenue and profit before taxation in more detail.



Here we see that Amazon's profit increases more than the revenue (in a relative sense), although it's still less impressive than Alphabet's profit. Yet there was a profit most of the time. Only 2014 shows a thin red roof indicating a loss.

This time it is interesting to present the values used in the Input-file (of course all the individual values can be found in the annual reports - please ignore the first line for now, as it is the "parameter-line"):

Amazon_Google-Alph.csv - Notepad

File Edit Format View Help

```
2, 10, 1000, 10, 6
Amazon;2007;14835;660;1197;5288
Amazon;2008;19166;901;2672;5642
Amazon;2009;24509;1161;5257;8556
Amazon;2010;34204;1497;6864;11933
Amazon;2011;48077;934;7757;17521
Amazon;2012;61093;544;8192;24363
Amazon;2013;60903;506;9746;30413
Amazon;2014;88988;-111;10741;43764
Amazon;2015;107006;1568;13384;51363
Amazon;2016;135987;3892;19285;64117
Alph.Google;2007;16594;5674;22690;2646
Alph.Google;2008;0;0;0;0
Alph.Google;2009;0;0;0;0
Alph.Google;2010;0;0;0;0
Alph.Google;2011;0;0;0;0
Alph.Google;2012;0;0;0;0
Alph.Google;2013;0;0;0;0
Alph.Google;2014;0;0;0;0
Alph.Google;2015;0;0;0;0
Alph.Google;2016;90272;24150;167497;28461

Revenue;Income bef. taxation;Equity;Liabilities
```

3.3 Cash Flow, Current Assets and Current Liabilities per quarter

The default-set to be shown in the 3D-graph is Revenue, Profit, Equity and total Assets (RPEA). The values as the name AnRep3D suggests are usually taken from Annual Reports. However, there is no reason to avoid quarterly updates. Then, Profit can be replaced by something like Cash Flow from Operations (roof) and by taking Current Liabilities instead of Equity (width) and Current Assets only (depth), we have a completely different set of ratios (like current ratio and cash flow ratio)!

Amazon, Alphabet (still called GOOG at the stock exchange) and Apple seem like an interesting set of candidates to follow for a couple of quarters!

This time the data are not coming from Annual Reports (form 10-K) but from froms 10-Q. Fortunately the structure is very similar. Below a collage of screenshots from the three important statements (Cash Flow, Income, Balance - Amazon, Q2 2018) is shown as an example.

CONSOLIDATED STATEMENTS OF CASH FLOWS
(in millions)
(unaudited)

	Three Months Ended June 30, 2018
OPERATING ACTIVITIES:	
Net cash provided by (used in) operating activities	7,449

AMAZON.COM, INC.
CONSOLIDATED STATEMENTS OF OPERATIONS
(in millions, except per share data)
(unaudited)

	Three Months Ended June 30, 2018
Net product sales	$ 31,864
Net service sales	21,022
Total net sales	52,886

AMAZON.COM, INC.
CONSOLIDATED BALANCE SHEETS
(in millions, except per share data)

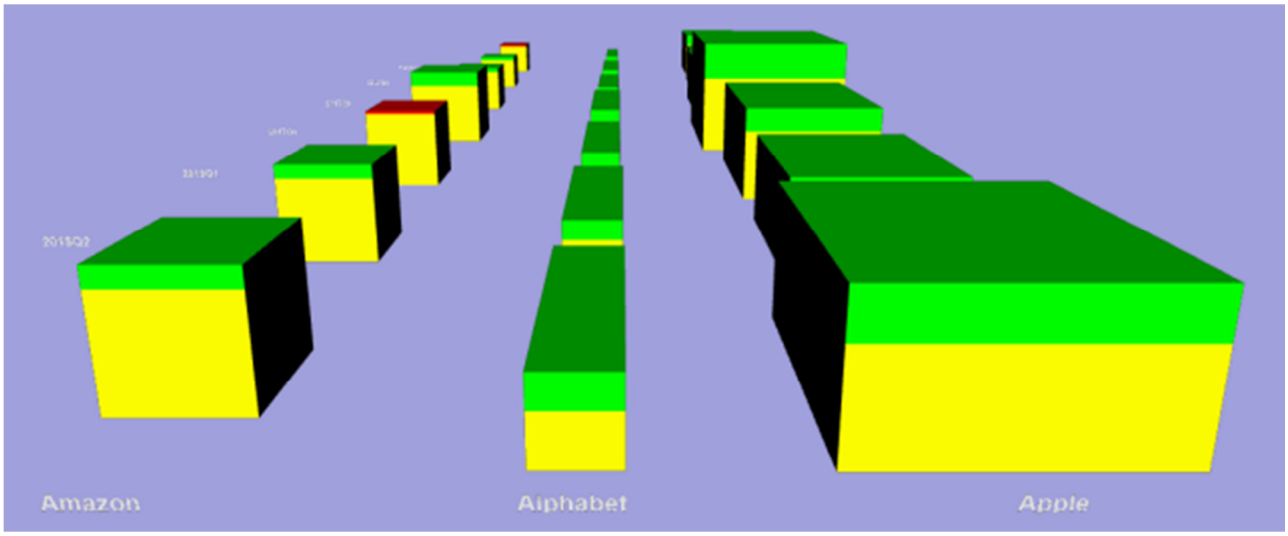
	June 30, 2018 (unaudited)
ASSETS	
Total assets	$ 134,100
Total current assets	54,481
LIABILITIES AND STOCKHOLDERS' EQUITY	
Total current liabilities	50,801
Long-term debt	24,638
Other long-term liabilities	23,666
Commitments and contingencies (Note 3)	
Total stockholders' equity	34,995
Total liabilities and stockholders' equity	$ 134,100

The relevant numbers got a red square around them and those values were put inti the input-file which was fed to the AnRep3D-generator. The others were used for checking purposes but not used in the input-file.

Apple has more or less the same ratio of Current Assets (depth) to Current Liabilities (width) as Amazon, although Apple's amounts are both twice as large.

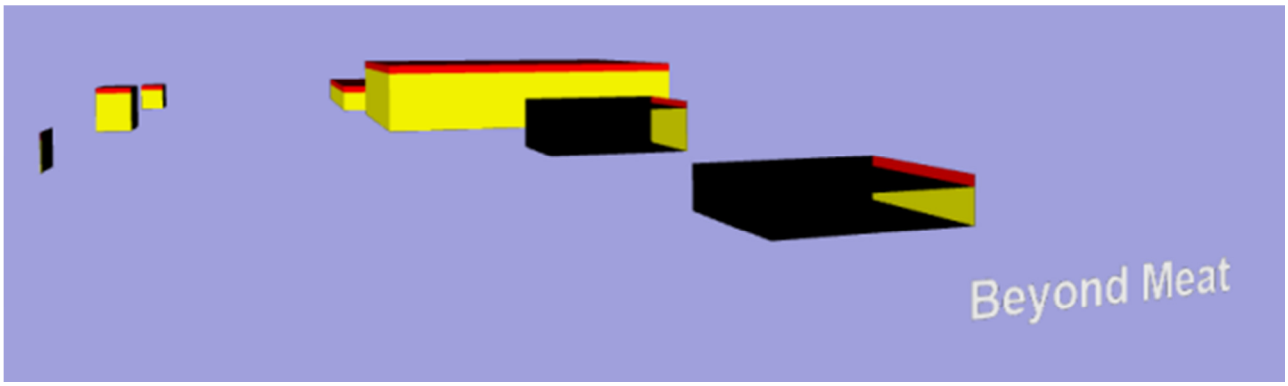
Alphabet, on the other hand, has a much better ratio as the current assets (depth) are about the same as Apple's, but its current liabilities are much lower!

Neither Alphabet nor Apple shows a negative cash flow during the period from the start of 2017. Yet Amazon did, as the two red roofs show.



3.4 Negative equity

Visualising some interesting IPOs, we encountered a surprise. Beyond Meat, the IPO with a very high return on its shares had a negative equity! It was very clear from the quarterly statements and we decided it would be best to just set the negative equity to zero. However, we forgot to do so and put the negative value in the input-file without thinking. Then another surprise came in. The AnRep3D-generator was quite able to deal with a negative equity after all. The "buildings" with a negative equity somehow became transparent. From some angles they are black, as if they don't really exist. From other angles the loss (red roof) is visible.



Although is it rare, it seems AnRep3D is able to deal with negative values not planned for. The profit/loss part represented by the roof was designed for positive (green) and negative(red) values, but fortunately the generator doesn't crash with a negative equity.

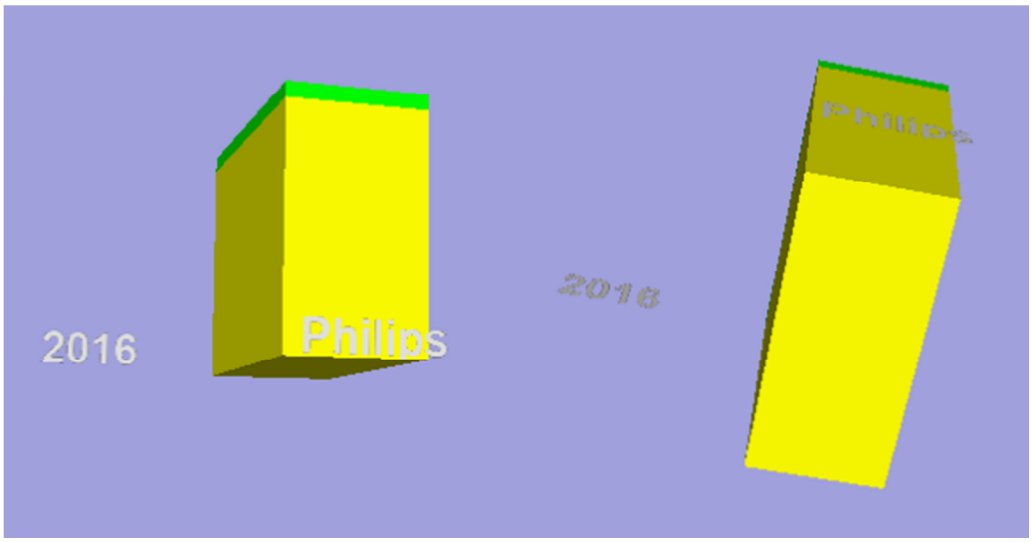
After the IPO, Beyond Meat suddenly had a lot of money and that's why the buildings to the rear are normal. They show a positive equity.

3.5 Different industries and their shapes

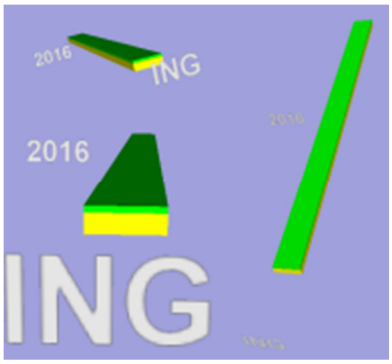
To generate 3D-graphs, the free demo-generator can also be used. Instead of several buildings only one will appear in a graph. For now that's ok as we are comparing different types of industries to see their different shapes in AnRep3D. Taking different screenshots will do to show what is meant. Again, we used the basic RPEA-set (Revenue, net Profit, Equity and total Assets).

Below an example is shown of a graph generated with the free demo-generator. It's about Royal Philips, a well-known Dutch manufacturer focusing on medical equipment. The picture below is a collage of screenshots taken from different angles.

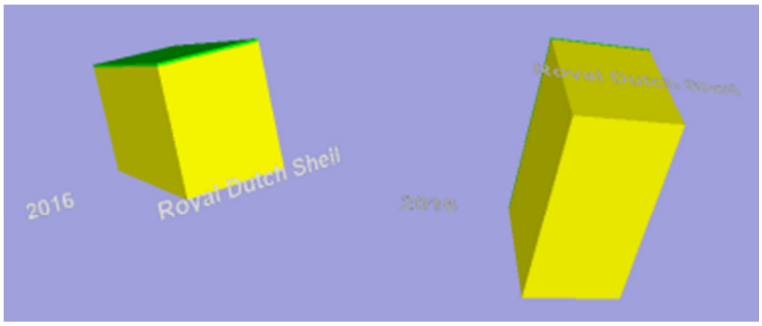
A manufacturer needs a lot of factories and machines and therefore equity will be rather high (and the gearing low), making a building with the ground-plan of a shoe-box.



If we take some other industries, we will see buildings with different shapes. ING is a bank and banks usually have only very little equity as they work with money taken from the market and placed somewhere else in the market. Almost all the money they lend to households and companies for e.g. mortgages and business loans is money brought in as savings and deposits. The money has to be paid back and therefore it is a Liability. The money the bank lends to other parties is an Asset. Because the difference is small, Equity is low and the buildings are very narrow and deep. A bank needs a lot of money to create some revenue and therefore the height is modest. The building looks like a shelf.

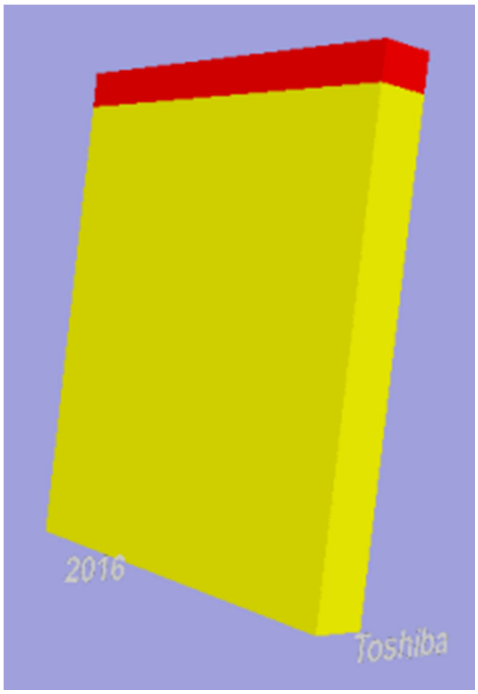


Royal Dutch Shell on the other hand, an oil-major, is similar to a manufacturer. A lot of investments have to be done in oil-rigs and refineries. The ground-plan looks like a shoe-box but the building is rather tall.



Then we take a very different one. Randstad is a temp-agency, so mainly a service-provider. Sometimes people will say that this type of industry makes a lot of money with one employee and a fax-machine. Indeed the Revenue (height) is impressive when compared to Equity (width) and total Assets (depth). The relative profit however, is modest as the operational costs are high.

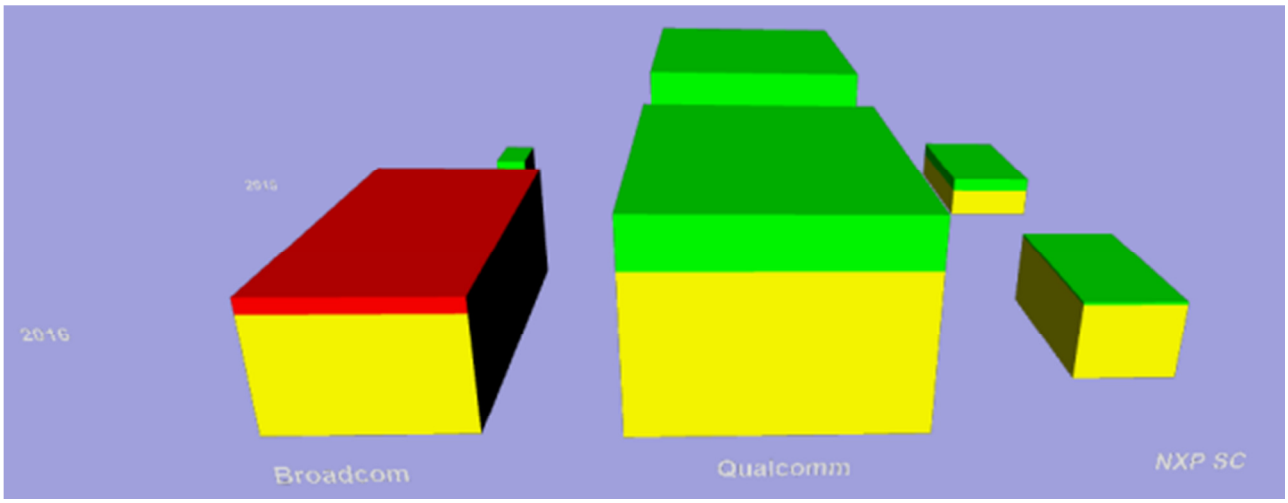
Finally, we show a manufacturer with a very modest Equity: Toshiba. It suffered some losses, eating out the Equity. The height and depth of the graph are almost the same, as the revenue is about the same size as the total assets. Unfortunately the equity, which already is very small compared to the total assets, doesn't differ a lot from the loss.



Indeed Toshiba seems to be a candidate for a take-over as some analysts say unless it's able to extract a lot of money from the market to boost the equity – or to return to a healthy profit of course. Toshiba is mainly a manufacturer, so we should compare it with Royal Philips or Royal Dutch Shell and not with a bank like ING and certainly not with a HR-services and temp agency like Randstad. The revenue is quite good but the gearing is high and therefore a loss will have more impact.

3.6 Mergers and Acquisitions

This time we will do it the other way round: first the 3D graph (output of the AnRep3D-generator) will be presented. Then an explanation will be given.



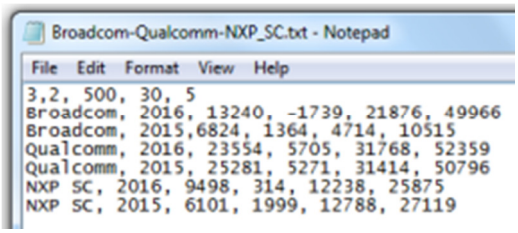
What do we see? Three companies in two years. Every "building" has sizes related to values from the annual report. Again total height is Revenue with the height (thickness) of the roof

representing Profit (negative if it's red instead of green, so a actually a loss). Then the width represents the Equity and the depth the total Assets. If a building is taller than another, this means the revenue was higher (than for another year or another company). A deep, narrow building means a lot of debt and so on.

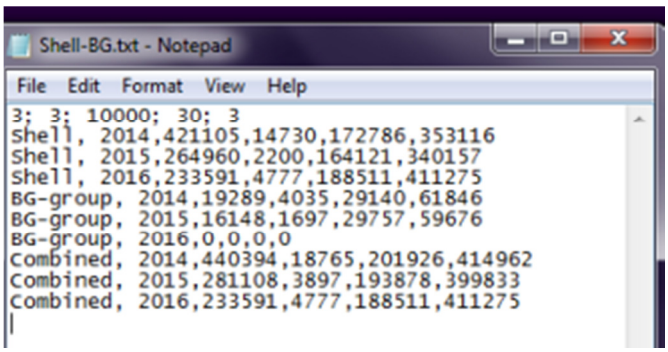
Now why are Broadcom, Qualcomm and NXP semiconductors shown in this graph? Well, that's simple: Qualcomm wanted to buy NXP semiconductors and Broadcom wanted to buy Qualcomm. So when this graph was created we were looking at a cascade of potential take-overs.

NXP is the smallest, looking at equity and total assets. Then Qualcomm would be larger but Broadcom had to be the richest party with the highest equity. Like in nature the smaller fish is eaten by a larger one, which in turn will be eaten by a really large one. As we can see Qualcomm is larger than NXP indeed: both equity and total assets are twice as high. But for Broadcom it's different. It wasn't larger than Qualcomm in 2016 and in 2015 (to the rear) it was even smaller than NXP! Actually the 2016 company is not the same one as in 2015, because it's actually just Avago, using the name of the company it took over! This new Avago-Broadcom combination shows a loss in 2016, where the others show a profit.

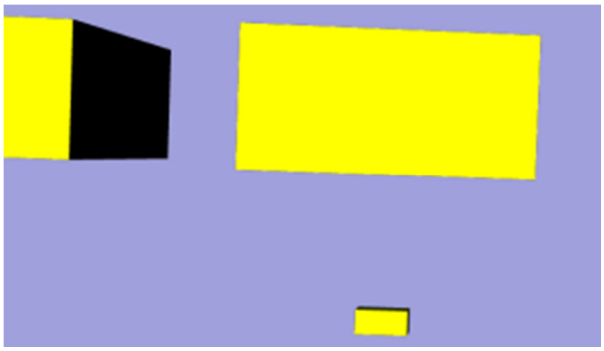
Also remarkable, is NXP having a high profit in 2015, but a lower profit with much higher revenue in 2016. The input-file used (values taken from the annual reports over 2016) are shown in the input-file like below (again: please ignore the first line as this is the parameter-line).



Another example of a takeover (this time a completed one) is Royal Dutch Shell taking over BG. The latter was much, much smaller. BG-group's revenue was only about 6% of the revenue of Royal Dutch Shell, but its profit in 2015 was about 77% of Shell's profit. Almost the same in absolute values! Equity and total Assets were very different again – BG showing only 1/5th or 1/6th of Shell's values. The input-file for the 3D-graph was as shown below:



Talking about ratios: if we have a look at the shape of the bottoms for both (2015 taken), it seems to be very similar for both, meaning the gearing is not very different, although the absolute amounts are.

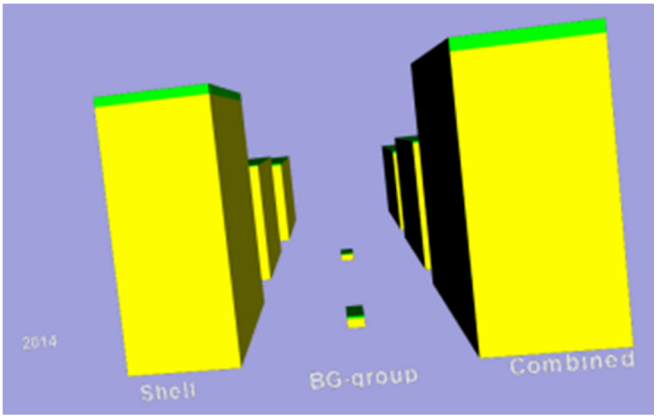


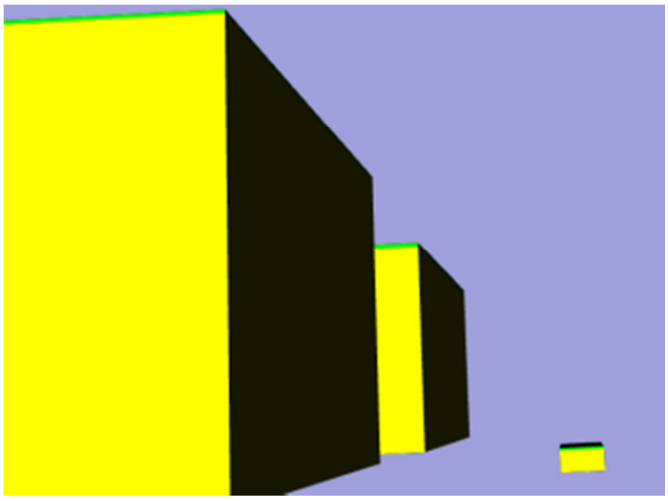
Looking at the front-side, we see a huge difference between the revenue-levels.

However, the profit (the thickness of the green roof) is more or less of the same size for 2015. This means the *absolute values of the profits of both companies were not very different!*

For the years before the merger, an artificial combination was calculated and presented in the graph next to the real companies. In 2016 BG didn't exist anymore.

This means the sum of the revenue, the sum of the profit, the sum of the equity and the sum of the total assets was taken for 2014 and 2015. Formally the same was done for 2016, but then Royal Dutch Shell published the consolidated values and the values for the BG-group became all zero, so for 2016 the combination is equal to Royal Dutch Shell.





Of course we cannot simply add the values, because mergers and acquisitions usually lead to an outflow of money to shareholders. Apart from this, it was the period in which the oil-price dropped and put a lot of pressure at the results. Still it's interesting to show a combined set of values. And although Royal Dutch Shell is more like Goliath, with BG-group being more like David, the profit was an interesting part of the comparison.

4 Applications in energy

Although AnRep3D was designed for the visualisation of financial information (as the name suggests: it's the abbreviation of "Annual Reports visualised in 3D"), the generator and the 3D-graphs created are actually "agnostic". The generator doesn't care what is fed to it, but to get a meaningful 3D-graph the variables have to be related somehow to visualise ratios. Yet finance is not the only area with ratios. E.g. energy has a mix as we know several energy-sources. As a matter of fact it would be nice to have ten-dimensional graphs (to visualise coal, oil, natural gas, nuclear, solar, wind, hydro, geothermal and biofuels including waste) but for now we have to live with three or four. Especially now, with all the climate-discussions going on, the focus is on fossil and renewables. Nuclear energy is in between as it doesn't produce CO_2 while operating, but it's not widely accepted either. Within the fossil section coal produces about twice as much CO_2 per Joule as natural gas, so there is another distinction. Then renewables are not a homogenous group either. Biomass can be a risk when it competes with food. In the end, depending on the subject it can be interesting to use visualise energy-mixes with the help of AnRep3D. We already suggested to introduce two different product-lines, with EnRep3D (Energy Representation in 3D) as a parallel one. For now we will only illustrate the possibilities.

4.1 The energy-mix in four European countries

The source-data for the energy-part of this graph were obtained from iea.org (the International Energy Agency). To be able to compare the consumption, we have to consider the size of the population and those numbers were taken from sources like Wikipedia, Indexmundi and Worldometers. Knowing the population allows us to calculate the "per capita" consumption.

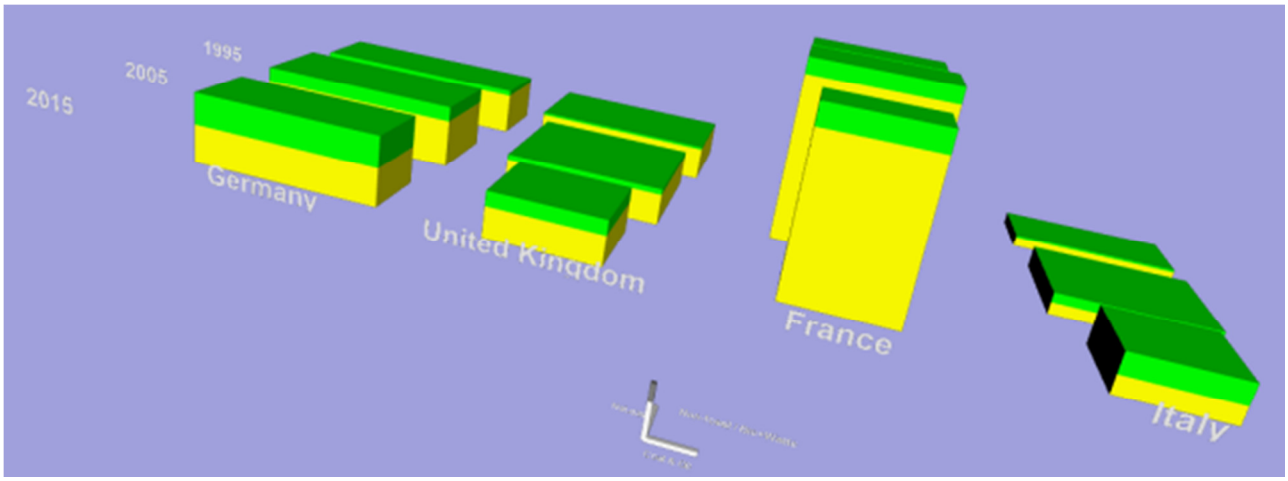
First we take a couple of EU28 member-states. At the moment this was written, the United Kingdom was still a part of the EU28.

The jumps in time are rather large, to have really impressive changes visible in the graph. Dimensions defined in the graph are all presented in the same units of energy, like ktoe (kilotonne of oil equivalent) or PJ (PetaJoule). As it's about ratios it doesn't matter which one. The meanings of the dimensions (also shown in the legend, hovering in front of the graph) are:

- Height: non-fossil sources (like nuclear, hydro, geothermal, solar wind and bio+waste)
- Width: coal and oil combined
- Depth: natural gas

The **yellow** part of the height represents sources without *direct* CO_2 emissions (nuclear, hydro, geothermal, solar and wind) and the **green** part – although non fossil – is the section which is a part of the CO_2-cycle and therefore more disputed than the regular renewables like solar and wind. The reason why natural gas is presented separately from coal and oil is because coal & oil are seen as the "old economy" now. Natural gas is fossil as well, but more of a transition source. Its related CO_2-emissions per unit of energy are lower, but certainly

not negligible. Be aware that natural gas includes LNG (Liquefied Natural Gas). All four countries will have values in every dimension. Remember that all values are “per capita” (representing the average energy-consumption of an inhabitant).



What is remarkable in this 3D-graph? Although the total energy-consumption per inhabitant (GJ per capita: sum of height, width and depth) is not extremely different (in 2015 Italy’s consumption was about 2/3rd of Germany’s), France has a very different shape, because the non-fossil part (height) is 2.5 – 4.5 times higher than for the other countries. The green part is not very different, but the yellow part is (4 – 8.5 times higher), mainly because France has a lot of nuclear energy. Germany is still heavily relying on coal and oil, although the consumption went down in twenty years. The UK however, did much better. All countries using less energy from coal and oil in 2015, compared to 1995. Of course the crisis helped to reduce the energy consumption. This explains why natural gas also went down. Otherwise a shift towards this source would have been more likely. Between 1995 and 2015 the green part (bio-fuel and waste) increased for all four countries. It would be interesting to investigate what the composition of this section is. The usage of potential food should be avoided and the same applies to the usage of agricultural land for biofuels instead of food. On the other hand, collecting waste from garbage could be useful, as long as the recycling of valuable materials – including plastics – is at a high level. Toxic exhausts should also be avoided.

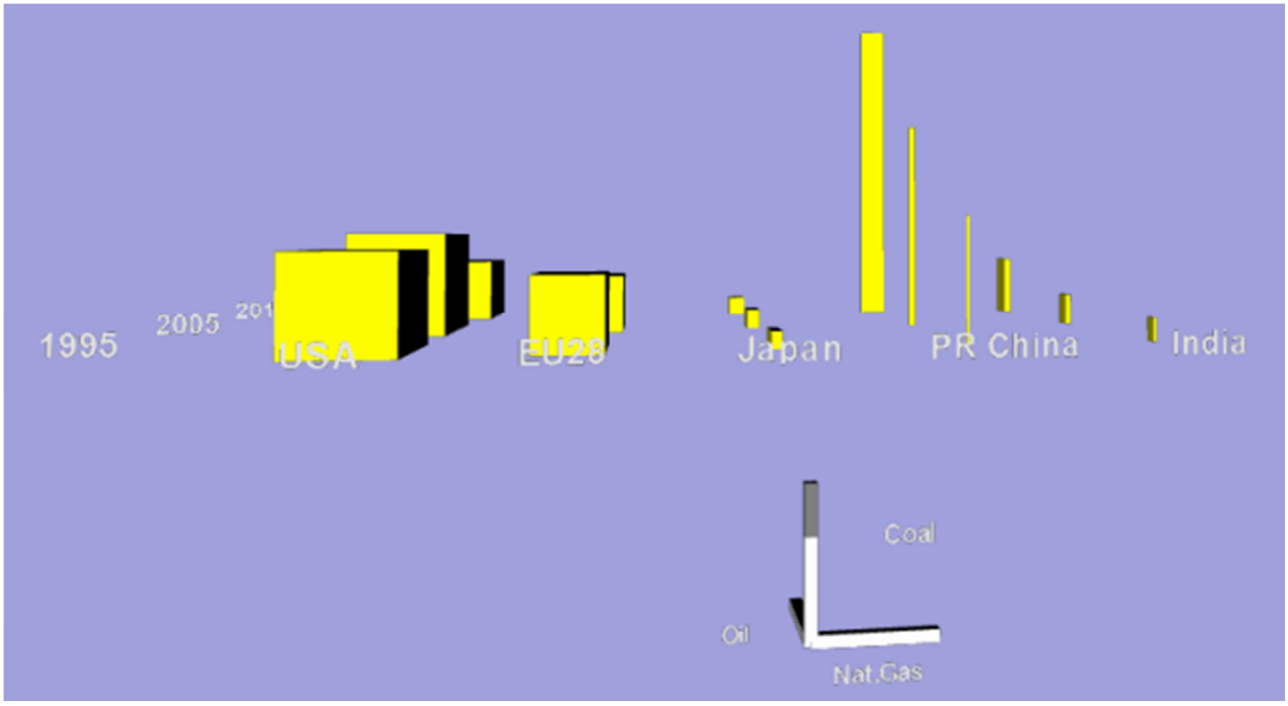
4.2 The energy-mix at a global level

Now we will look at the mix of fossil fuels only, at a global level. If we look at CO_2-emissions, the main contributors are coal, oil and gas, we know coal is roughly twice as bad as natural gas and oil is in between (roughly 100, 77 and 55 kg CO_2/GJ).

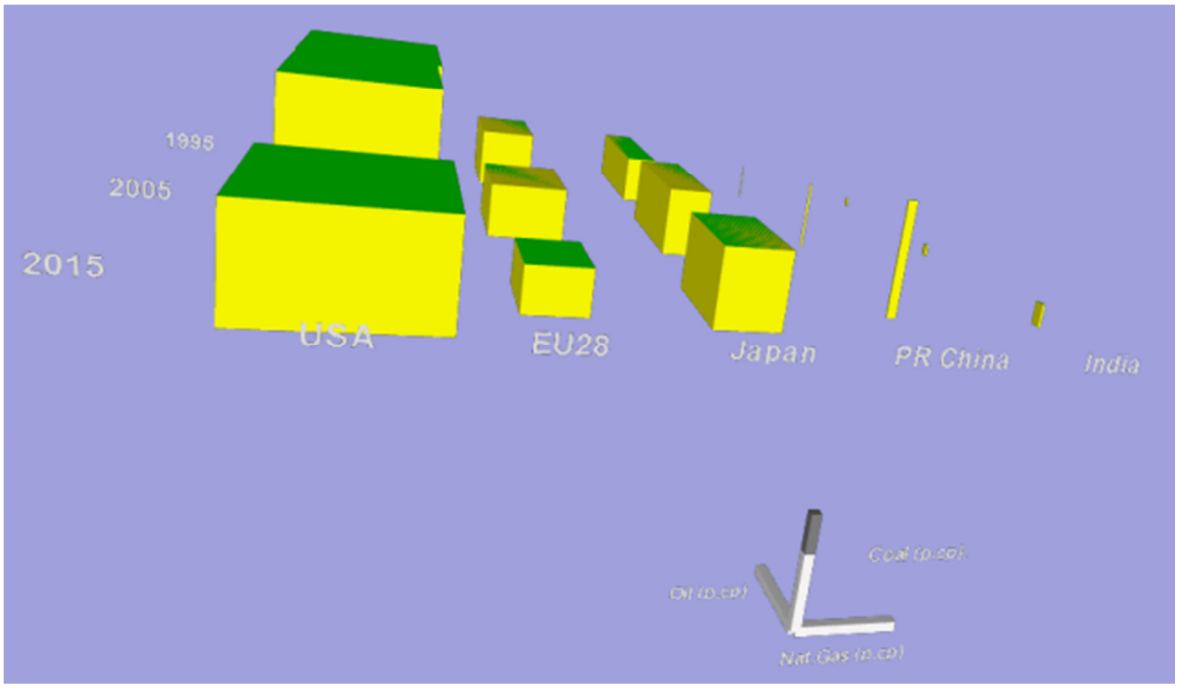
This means the composition of the fossil part is very important, as an amount of energy in coal could be replaced by the same amount of energy in natural gas, but with half the CO_2 being emitted. It’s still fossil energy, but for the transition it is an intermediate step. Let’s have a look at the data from the IEA again but this time only with coal, oil and natural gas as dimensions as the operation of nuclear energy stations and renewables do not have a net contribution to the CO_2-emissions. To give a fair overview, two approaches are shown

below: total use and use per capita, because the former tells us something about the global impact, but the latter is a more honest measure.

This first graph won't be a surprise. Everybody knows China uses a lot of coal – far more than the USA – but not a lot of oil and natural gas. Be aware that here the past in front and the future to the rear. This is just a choice as the numbers can be fed to the generator in any order, creating different arrangements in the 3D-graph. The graph shows that the consumption of coal rose quickly in China between 1995 and 2015. For India it's far less impressive and the other countries are more or less stable.



For the second graph, the time-axis is reversed: the "present", in this case 2015 (completing energy-statistics on a global level takes a couple of years) is to the front and 1995 is to the rear. Here we see the average coal-consumption (height of building) *per inhabitant* and this time China is still high. Here, the USA is surprising. The population increased by about 20% between 1995 and 2015 but the coal-consumption went down (after an intermediate increase). The EU and Japan show more or less the same pattern and none of them seems to replace the coal with oil (depth of building) or natural gas (width of building). All managed to keep their fossil energy-usage more or less stable or even reduce it. On the per capita level, *all of them reduced* their energy-consumption.

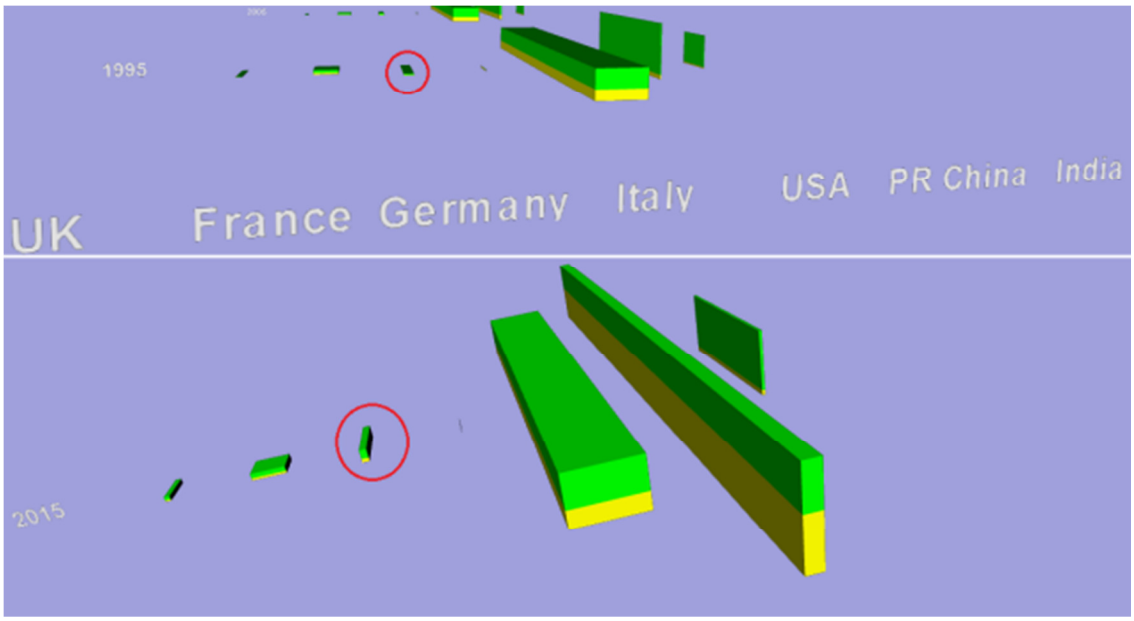


India and China saw their energy-consumption doubled or tripled over two decades (the in total increase is stronger than for the value per capita). We cannot judge them for that, as they are simply developing later than the others, but we can ask them politely to use either less coal or get rid of the CO_2 – and in the end they will!

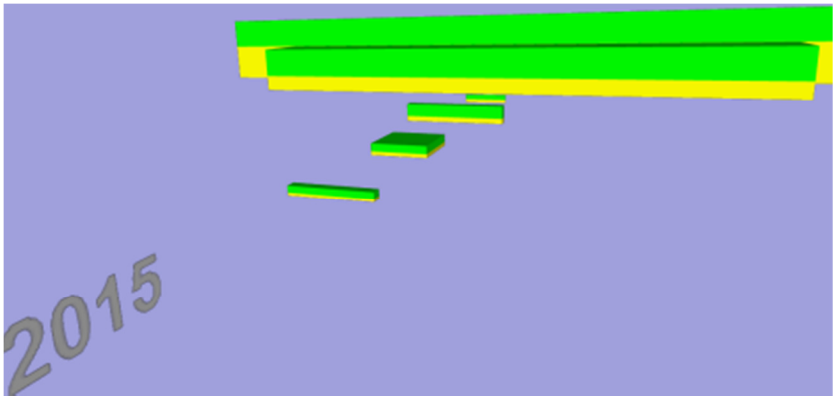
4.3 "Energiewende"

Now that we have shown rather easy examples, we can move on to a more complex mix. This time we compare the total of fossil fuels (depth), nuclear energy (width) and all the "renewables" (height). Renewable energy is actually a strange mix and therefore we will split it in a non-biological section like solar, wind, hydropower, geothermal and several other types (yellow part of the height) and biofuels and waste (green part of the height).
The first picture is a combination of two screenshots from the same graph. In the top-half we see the Germany "building" for 1995 (within the red circle). The USA and PR of China are also in the graph and therefore the "Germany building" is quite small. Again: the width represents the part of nuclear energy (total value in PJ) and the height is the amount of renewables (the green part being the bio-energy component).

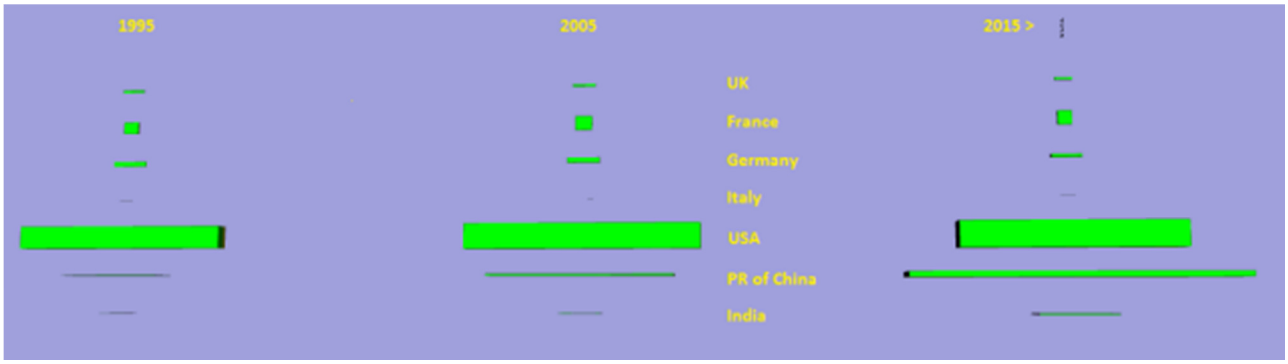
The bottom half of the picture shows Germany again, but now for 2015 (again in a red circle). This time the building is much narrower, indicating an enormous reduction of the nuclear energy. It's the impact of the "Energiewende". At the same time, the building is much higher than in 1995 because of the increase of renewable energy. The green roof is really thick so the majority of those renewables are still biofuels and not wind or solar (found in the yellow part underneath). Italy is next to Germany and hardly visible as it displays a very narrow building. To be honest, there is no width at all as Italy doesn't have nuclear energy.



If we turn the graph to see the side of the 2015 buildings, it becomes clear that most of the energy consumption in Germany is still fossil-based. For France, showing more or less a square building, fossil and nuclear are about the same. See screenshot below.



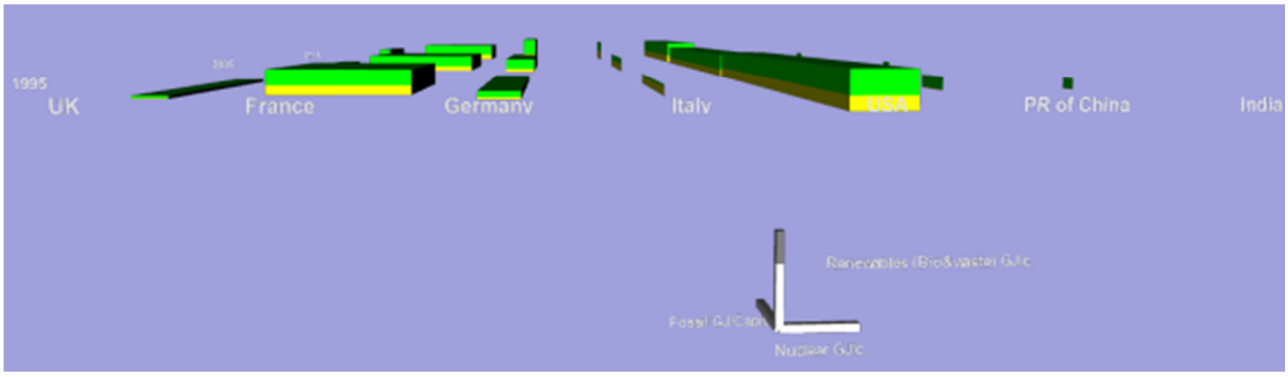
A better way to compare all countries in the graph – when looking at the usage of fossil and nuclear energy – is to take the top-view. In the real 3D-graph the names of the countries are in front of the lanes and easily checked, but because this is only a fixed screenshot the names were put in manually.



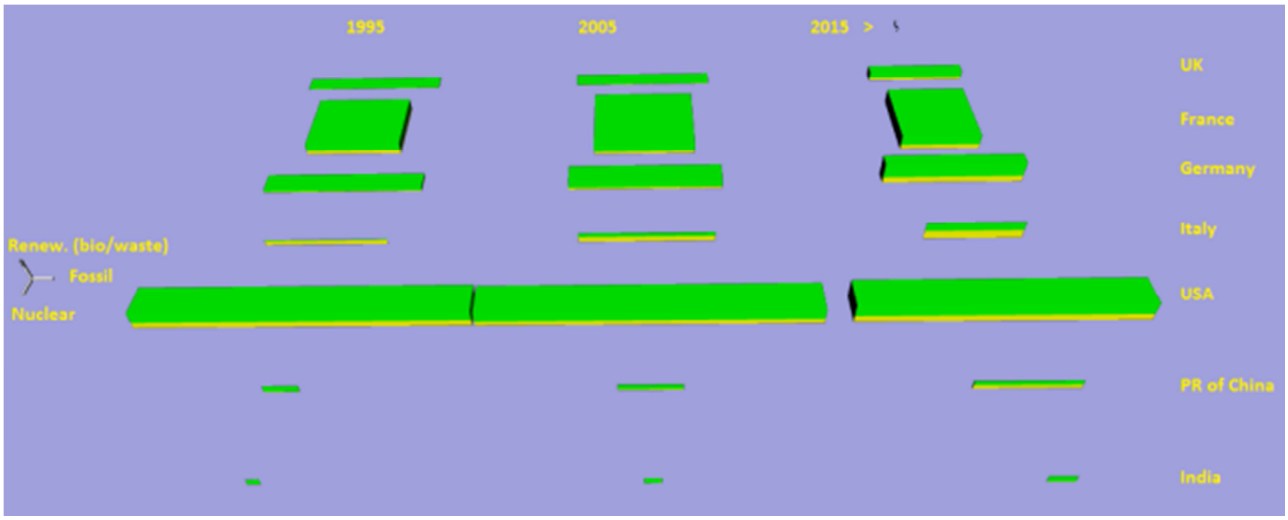
This top-view shows us that the fossil consumption of both Germany and the UK is higher than for France. At the same time we know that France uses about twice the amount of nuclear energy of Germany and the UK taken together. That's why France has very low carbon-emissions! The absolute amount of nuclear energy consumed by France is more than a half the nuclear consumption of the USA, although the USA has a much larger population! The shape of the USA-building resembles the shape of the one for Germany because the ratio of fossil to nuclear is very similar: much more fossil than nuclear energy. The People's Republic of China consumes more or less the same amount of fossil fuels as the USA (in 2010 it was less, but in 2015 it was already about 30% more than the USA), but its population is at least three times as large. And India, with a population that is also over one billion, still consumes less than half of the USA-amount of fossils.

It's time to switch to the "per capita" graphs again, showing us the average consumption of an individual in every country. This *won't change the shape of the buildings* as the mix of fossil, nuclear and renewables will remain the same, but the relative sizes of the buildings will be different after all!

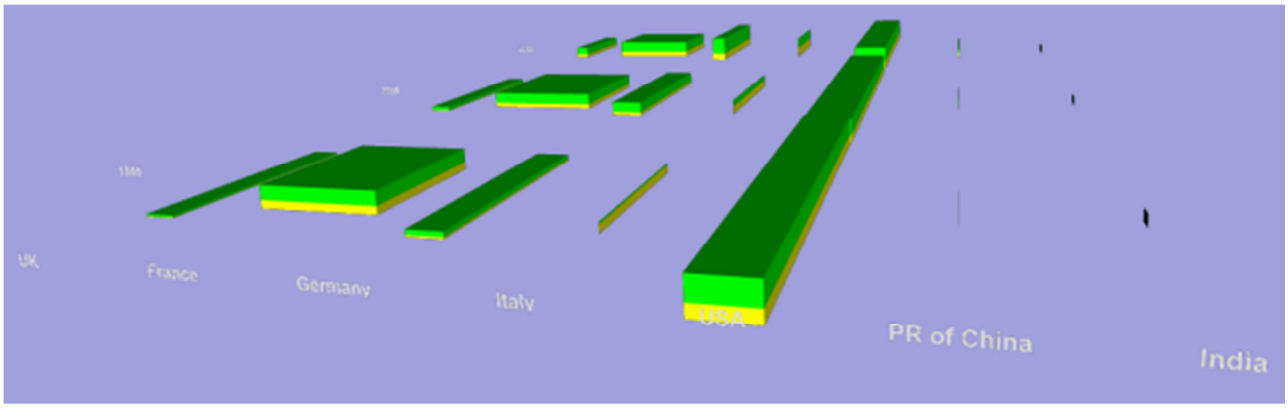
The first screenshot of the "per capita" version shows the legend to the front of the graph. The values are the same as the previous ones for the countries, but divided by their population for the year of consumption. Suddenly the size of the USA-building is closer to the ones for Germany or the UK, but both China and India are hardly visible any more. They have a very large number of inhabitants and the energy-consumption of the average individual is much lower than for the USA or even countries in Europe.



An oblique top-view illustrates clearly that the average inhabitant of France consumes more nuclear energy than the average American from the USA. But although France was rather good at renewable energy in 1995, the USA was doing better.



Moving from the side to the front, we can see in 2015 individuals in Italy and Germany are on top when it comes to renewable energy and for Italy it's even about 50% of non-bio & waste (yellow part of the height – e.g. hydro, wind, solar). In this graph the most interesting part is not the comparison between countries, but the change in the mix over the years!



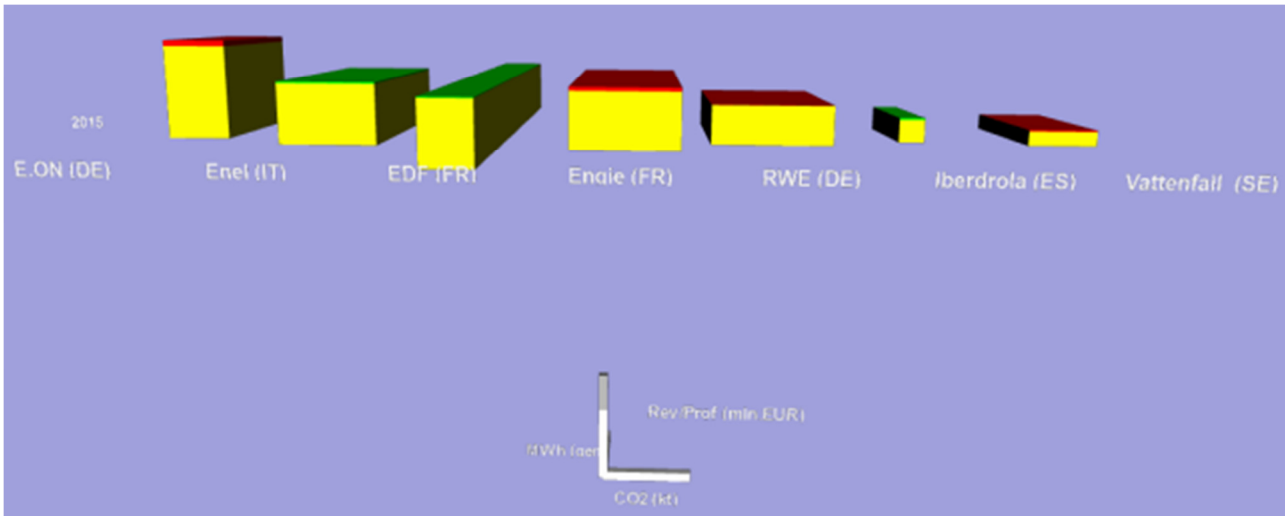
5 Other uses of the AnRep3D-generator and 3D-graphs

AnRep3D could have different names for different markets and purposes of course, like EnRep3D (meaning Energy Representation in 3D and this might be the best alternative as it sounds very similar). Or something like CuCaRep3D (Cure and Care Reporting in 3D) and maybe even TelCo3D. We would need a lot of different manuals. For now the manual is still focusing at Annual Reports. Yet it is good to show some alternatives, despite the name of the generator. When the generator can move from finance to energy, other data will be processed as well. In his chapter some additional ideas for other areas will be presented.

5.1 A mix of finance, energy and CO_2

AnRep3D will process any set of numbers. It's up to the analyst creating the graphs to have meaningful combinations. Although this example is about companies, finance, energy and CO_2 will be combined this time. The companies selected are delivering mainly electricity and they are working throughout Europe. The start was a paper from Prospex on the top-ten of electricity producers and traders in Europe. It's about 2015 and of course we are five years later now in this white-paper, but that doesn't matter. The Prospex-paper holds TeraWatt-hours (TWh) produced and traded, revenue, profit and also CO_2 emissions. For AnRep3D we know the Revenue and Profit part already, as this will be the height of a building in the 3D-graph. Instead of Total Assets (depth) the number of TWh produced is selected and instead of equity (width) CO_2 was taken. It's all a matter of taking the right scale and millions of Euros, MWh of electricity and kiloTonnes of CO_2 turned out to match well (remember that the scaling factor affects all three dimensions at once, so the ratios have to be right upfront). Because CO_2 mentioned is only related to electricity produced (not to the amount only being traded), I selected the companies with over 75% of own production. Since some companies also trade e.g. gas, the revenue is not clearly connected to the electricity, but more like an indicator of the size of the company. The profit then would be an indicator of success.

In the end this combination of *Revenue, Profit, MWh of electricity and kilotonnes of* CO_2 *provides a very interesting 3D-graph*. Two German companies (E.on and RWE) produce more or less the same amount of electricity (MWh), but their CO_2 emissions are very, very different. It's similar for two French companies (EDF and Engie), but here EDF produces even much more energy with lower CO_2 emissions. Vattenfall (Swedish for waterfall) suggests green energy, but their relative CO_2-footprint is somewhere between Engie and RWE. The Italian company Enel is similar to Engie, but the difference there is that for the Italians there is a profit instead of a loss. Iberdrola finally, makes a profit and has a good CO_2 to MWh ratio! It's because of a mix of nuclear energy, natural gas and renewables. Only EDF's ratio is better as the majority of their electricity comes from nuclear energy. You can have a look for yourself.



5.2 Agriculture in 3D – minerals, livestock and farmland

To be honest, this subject was quite a struggle. In Europe we have very nice and publicly available statistics. It was quite easy to get the data for the Nitrogen (N) and Phosphate (P) ***surplus*** per hectare (about 2.5 acres) of farmland. But then gaps and negative surplus-values were in (that's possible of course – it means there is a lack of the mineral). Europe is quite large, so a selection of countries had to be made and the question was also which years to report. For clarity: nitrate and phosphate are useful compounds in agriculture, but a surplus will pollute land and water, causing damage to the environment. That's why Europe collects those data and has legislation on some minerals. Manure is a fertiliser, but when the amount of minerals exceeds the need of the farmland, it will be more of a poison. As more and more manure was deposited on the farmland, all kinds of side-effects came up as a result of this dumping. To understand the negative impact of the surplus this article can offer some insight. Currently we have a climate crisis, but in the past we have dealt with acid rain and the ozone depletion. Acid rain was and is actually connected to the Nitrogen issue.

In the end it all comes down to the area of farmland related to the number of livestock held in this area. For decades agriculture became more intensive in several European countries and as a result the amount of manure went up. We don't want to limit the number of animals – on the contrary with e.g. "mega-stables" coming up – but at the same time the surface of a country won't increase (apart from legal tricks and claiming land back from the sea perhaps). How to get rid of the dung, holding these minerals? The easiest way is to dump it on the farmland, but this is not a sustainable solution. Yet, manure is not the only cause of the surplus and sources outside of agriculture are attributing as well.

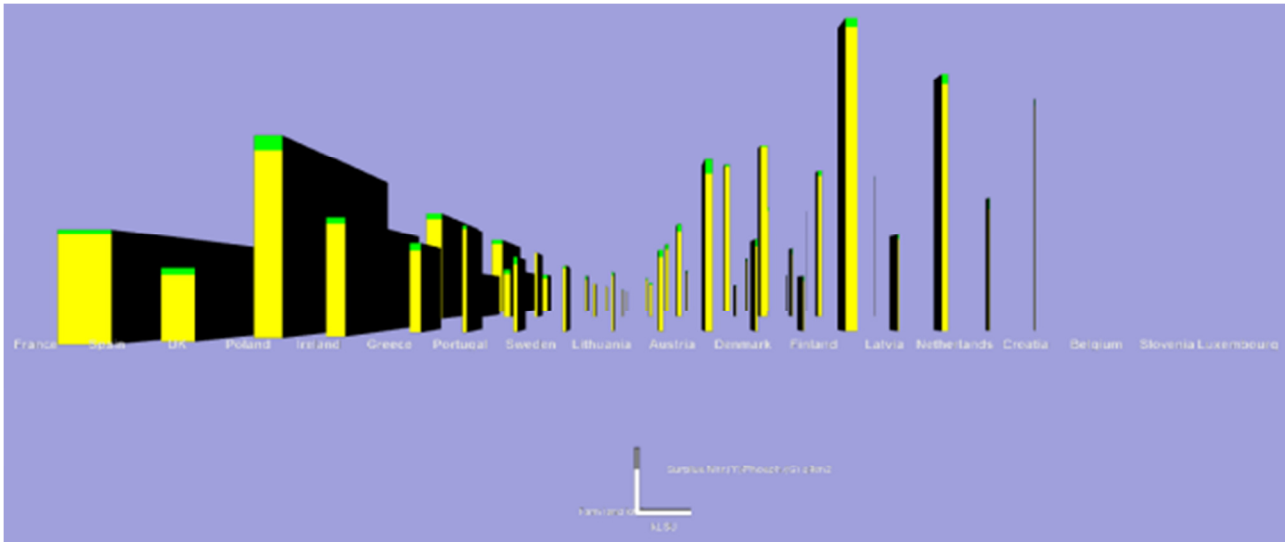
A long time ago, the EU came up with laws to reduce the surplus and here we are! To see whether these laws are successful, we put a couple of values in a table and generated a 3D-graph with the help of the AnRep3D generator. The data were mainly obtained from Eurostat.

It made sense to use the height of the AnRep3D “buildings” for nitrate (N – rather large values) with a green roof for phosphate (P – more modest values). To avoid confusion: N is the chemical symbol for Nitrogen and P for Phosphorus, but we use them to identify the derived compounds eg. NH_4^+, NO_3^- and PO_4^{3-}.

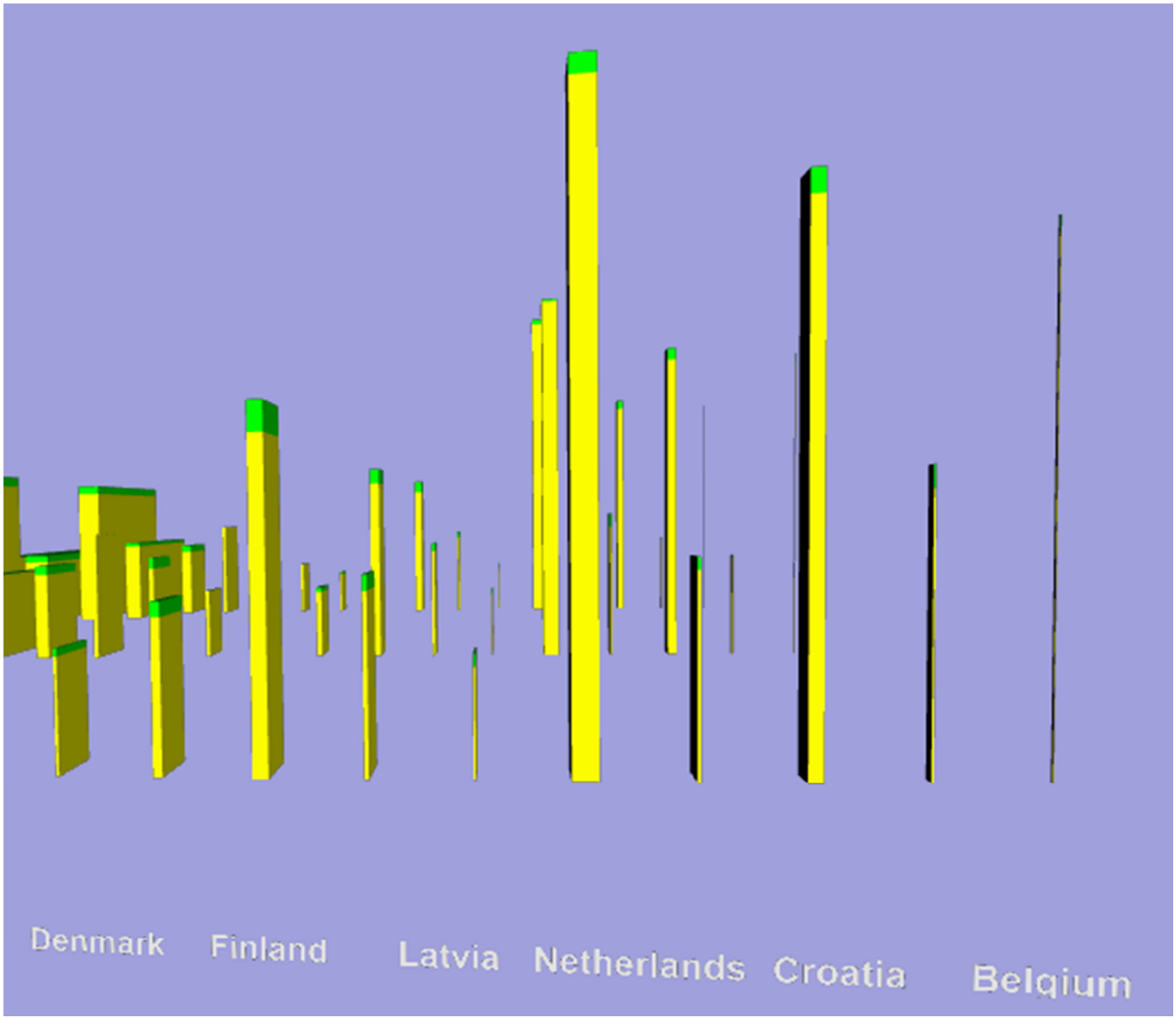
Since we know the real issue with minerals on farmland is the number of animals in relation to the available amount of farmland, the width of the buildings could represent the number of animals and the depth the surface (square kilometres – 1 km^2 is about 247 acres) of farmland. So far so good, but then other issues arose. The agricultural land numbers includes wasteland and woods, but we all know most of the manure won’t get there. In the end it turned out the surface reported by the European statistics covers for permanent grassland, permanent crops and arable land. The others seem to be subcategories. The years for farmland didn’t match quite well with those of the minerals, but in the end 2013, 2014 and 2015 were selected, applying interpolation for the minerals.

The next issue was about the animals. It’s nice to count them, but if one country has mainly cattle, the next one goats and sheep and another one mainly chickens? It’s not a fair comparison. Fortunately there is a standardised unit, correcting for the impact of the animal and this is the Live-Stock Unit (LSU). At first the numbers made no sense and didn’t match with the reference-values. Then it became clear the LSU-tables offer two units: numbers (just a headcount and not LSU at all) and real LSU. After having all this straightened out, the input-file was ready in a couple of minutes.

The only choice left was about the size of the values. The N- and P-values were converted to gram of surplus per square kilometre (g/ km^2). The surface of farmland (arable, permanent crops and permanent grassland) was already converted from Hectare to km^2. Only the LSU could either be in kLSU (kilo-LSU = thousands) or in hLSU (hecto-LSU = hundreds). The former offers a compact graph, the latter provides a better overview for smaller countries, but is very wide. The solution was to generate two different 3D-graphs. The one with kLSU is compact enough to show all the countries in it in a glance whilst the other one provides more detail. A couple of countries have very high values for the nitrate-surplus, with the Netherlands being the number one. Be aware that because of the parallax the names of the countries are not in front of their “buildings” in the screenshot.

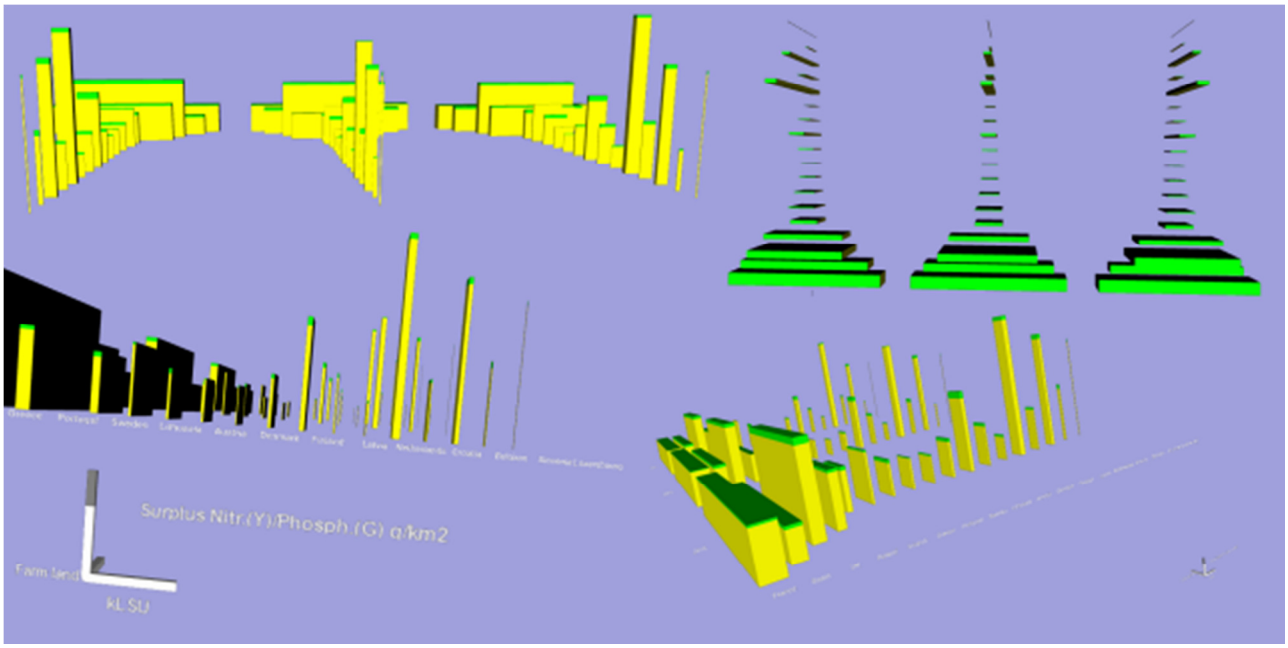


For more detail a zoomed screenshot is shown below.



The N- and P surplus-values were converted to gram per square meter of surplus (g/ km^2). The surface of farmland is in km^2. LiveStock Units (LSU) are a way to translate different kinds of animals (geese, sheep, cattle, horses of even mooses) to a uniform value which can be used in calculations. Here we use thousands of them (kLSU = kilo-Livestock Units)

The 3D-graph (showing the N and P surplus in relationship to the area of agricultural land and the number of LiveStock Units) is available. Below we present a couple of screenshots from different angles.



What do we learn from the 3D-graph behind the screenshots? We can see that larger countries seem to have a smaller surplus than the smaller ones. (As the order is from large surface to small from left to right, it's easy to recognise.) One of the reasons is that they also have a lower Livestock to Area ratio (kLSU/km^2). However, it's not a linear relationship. E.g. the UK has an issue with high surplus-value, despite the large surface of agricultural land. Spain, with a higher number of (k)LSU and a comparable surface has a lower surplus. Greece has about the same surface for agriculture as Ireland, but has a larger surplus despite the much lower LSU-count. For Spain the surplus went up from 2013 (front) to 2014 and 2015 (mid and rear), but Poland managed to reduce the surplus in 2014, but it went up again in 2015. If we can concentrate on the kLSU/km^2 ratio (the shape of the "buildings" as seen from the top), we can see Denmark, the Netherlands and Belgium have a high ratio and their surplus is (very) high as well. Countries like Sweden and Austria have lower ratios (lower kLSU, larger area) and their surplus-values are much lower indeed.

Finally the green part is interesting. The surplus-value for Nitrogen (mainly ammonia and nitrate) is represented by the yellow part of the building. The green part represents the P part (almost solely phosphate). Phosphate can endanger water quality by causing algae bloom. We can clearly see that Spain has a higher P-surplus than France with the N-ratio being the opposite. The same applies to Denmark in comparison with the Netherlands. Of course the surplus is also related to the amount of mineral needed in the soil. With a low level of certain minerals, the surplus could be lower – but not necessarily for both N and P. Then, the composition of the manure (coming from different types of animals that is) can also cause differences between the surplus-values for N and P.

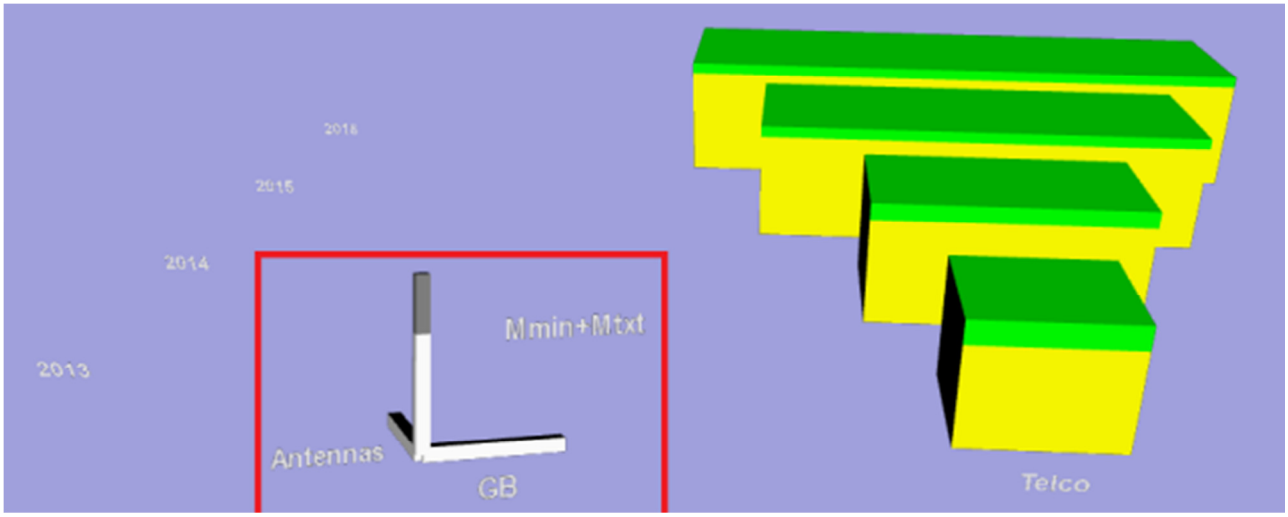
5.3 AnRep3D in Telecommunications (TelCo)

From a consumer-perspective TelCo is about minutes, messages, megabytes and perhaps antennas, so those are the subjects taken for this example. The data were obtained from a Dutch marketing site, presenting minutes, texts and MB in thousands which were converted to Mmin (Mega-minutes or millions of minutes of phone calls), Mtxt (Mega-messages: millions of text messages) and GB (Gigabytes or billions of bytes). Then the website of a Dutch regulator provided the number of antennas active during a couple of years. It turns out there is a lot of them. Not millions, so no Mega-antennas but just their absolute number. The combination was available for four years.

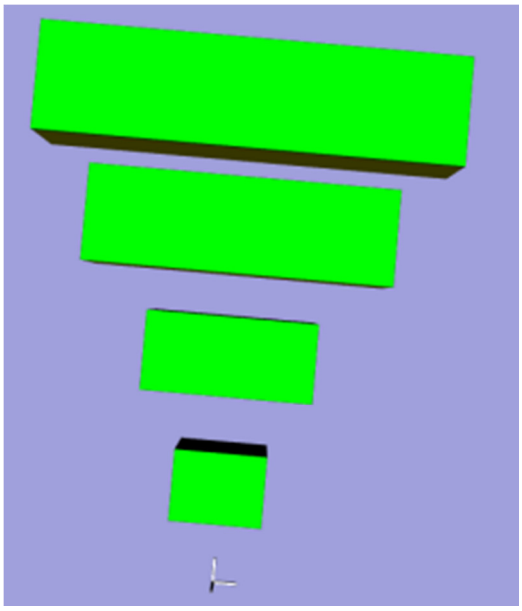
First the input-file in Excel is shown, just before saving it to a .csv-file ("MS-DOS format", Excel still says after more than twenty years of Windows).

	A	B	C	D	E	F	G	H
1	1	4	1000	20	5	Mmin+Mtxt	GB	Antennas
2	Telco	2013	28239	5271	33935	25874		
3	Telco	2014	29956	4077	63077	29080		
4	Telco	2015	30816	2848	114579	35892		
5	Telco	2016	33238	3016	158673	40328		

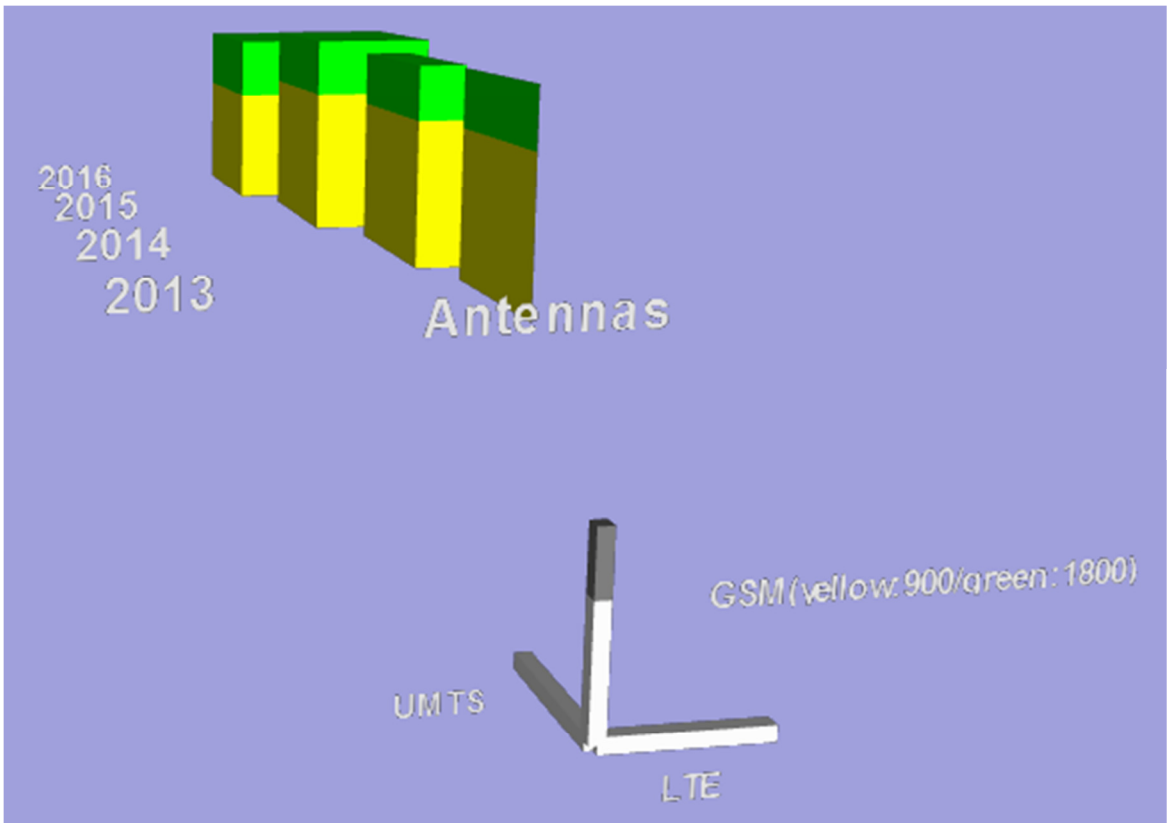
Then the output: the 3D graph in html (to be viewed in a web-browser). The legend/key is put in the screenshot as a picture in picture, to provide some guidance.



The graph is very different from the previous ones. Data-consumption has grown explosively, it seems. At the same time the number of text-messages went down, although not as impressive as one would expect. Most people replaced them by app-messages which will be in the data-segment. However, at the same time the Internet of Things (IoT) applications came up and sometimes a simple mobile phone sim-card is used to let those devices send their messages. Probably this put a halt to the decline of the text messages. The real IoT growth is in the data-segment of course.



The number of antennas has increased slightly. So how do they keep up with the growth in data-usage? Were these antennas oversized from the beginning? In the report of the regulator there is a classification in GSM (900 or 1800), UMTS and LTE. The latter is for real IoT-data. Counting the antennas per category, an even more interesting graph could be created. A screenshot is shown below.



Double-click the screenshot to see the live 3D-graph in your browser. For instructions on manipulation, see 3D-graph above

The LTE are coming from nowhere in 2013 to more or less the same number as all GSM-antennas (and nearly the same as UMTS) in 2016. All the growth in the number of antennas

is coming from the LTE-group, but is there a relationship between data usage and LTE-antennas? Or was a lot of capacity available in the first place and has the percentage of occupation of the bandwidth gone up? For now these questions won't be answered as this whitepaper is primarily meant to show the power of AnRep3D.

5.4 A healthcare example

The data for this example are about healthcare in the Netherlands. This is because nearly everything in this country is measured and documented, so the result will be interesting.

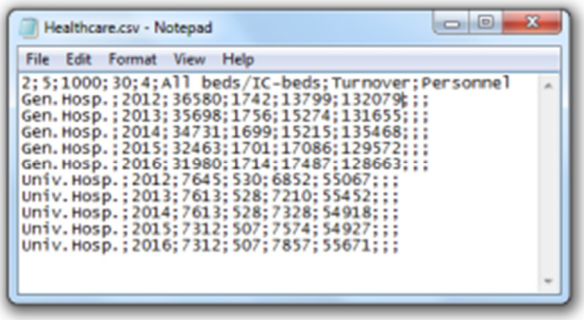
To obtain two sets (still in time: annual values) the data was split between:

- General Hospitals and
- University Hospitals.

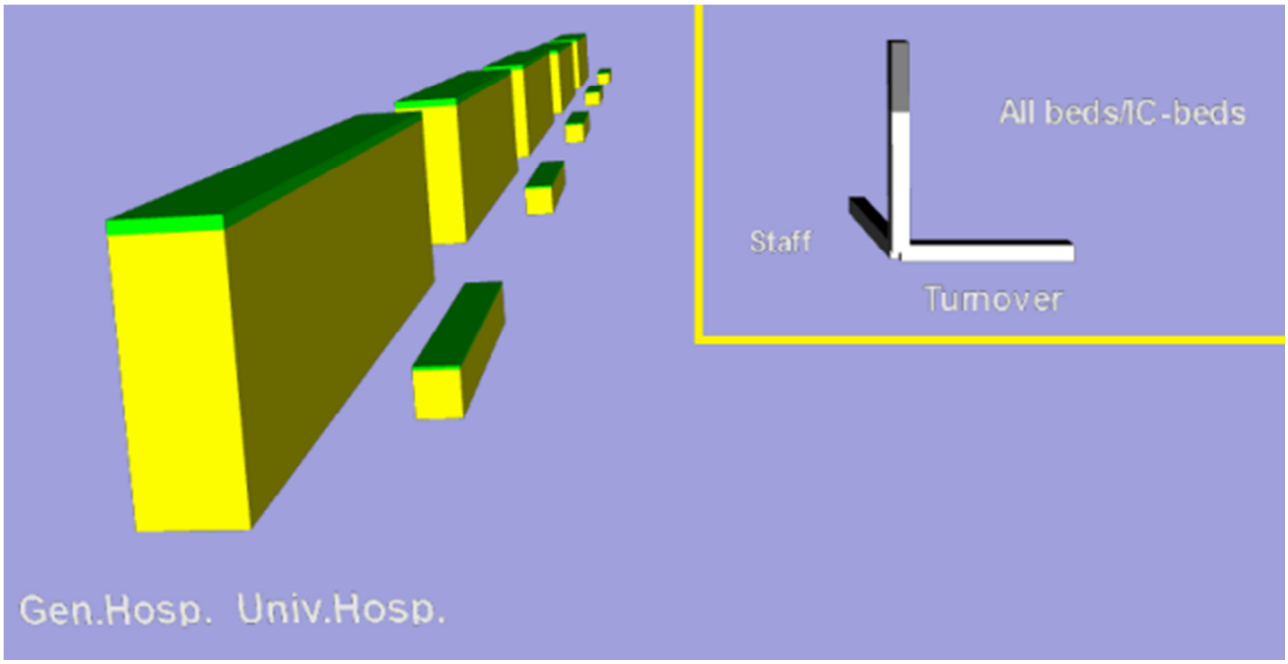
For both a set of characteristic values was obtained:

- Total number of beds, of which beds in intensive care mentioned separately
- Turnover (in millions of Euros. Hospitals do not have real "revenue" or "sale" values.)
- Staff, measured in Full-Time Equivalents, rather than "bodies".

The data was obtained from a [(Dutch) site on healthcare](), with some small adjustments made. The input-file will not be very different from the previous ones. It's just numbers. Below a screenshot is shown. The additional semi-colons are irrelevant. The data were entered and ordered in Excel and saved as a .csv and doing so the additional semi-colons came in. *Of course the first line is still the parameter-line!*



The output is like always an html-file to be viewed in a web-browser, where the 3D-graph will be shown and can be translated, rotated and zoomed in or out.



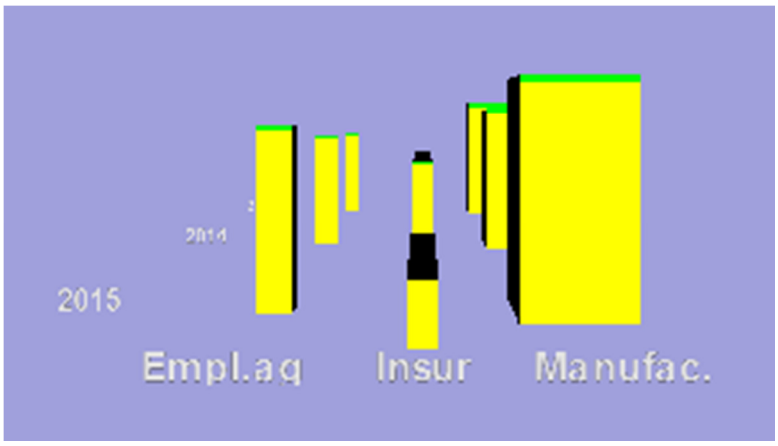
Interesting, but not surprising is that for General Hospitals and University Hospitals the ratio between turnover (width) and staff (depth) is not very different. The (large) university hospitals are together about half the size of the other hospitals for both the money and the people. This is because the wages are the greater part of the total budget, despite the expensive devices.

On the other hand, the number of beds is much lower in University Hospitals. More like a quarter of the number of the General Hospitals. Apart from Outpatients and Daycare (the latter may or may not be reflected in the number of beds, but is certainly related to the staff) the patients in a University Hospital have usually more complicated health issues. Also interesting is that the changes over time are rather small. In a spreadsheet they look more impressive.

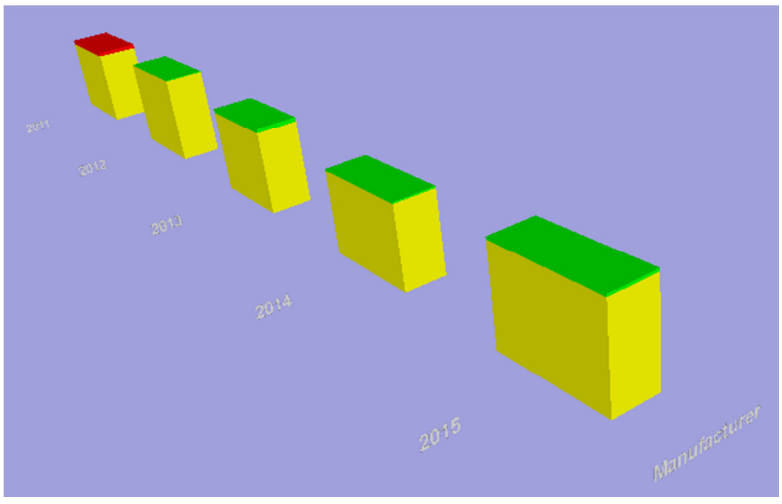
6 Abstract from the AnRep3D-manual

6.1 Introduction

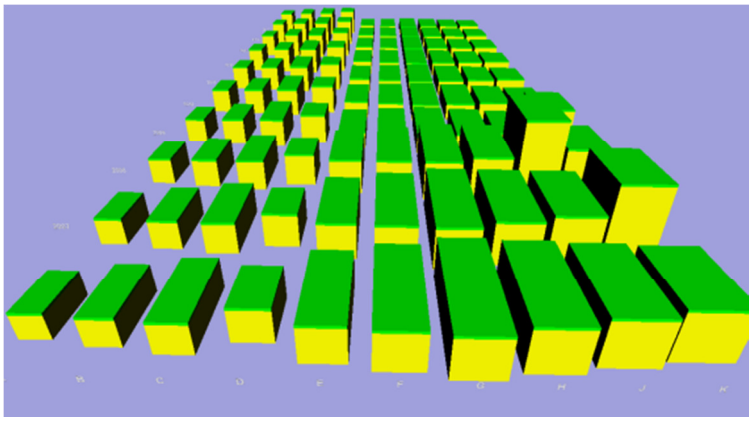
Of course there is a manual, coming with both the paid version (as you will know there are two options for payments: money or creativity) and the free demo. Parts of this manual serve as a kind of walkthrough for the usage of the AnRep3D-generator. That's why an abstract is offered in this white-paper.
It´s easy to explain the generator in a technical way but the concept of the graphs it creates has to be understood first. The 3D-graph – the output of the generator in .htm-format - will show a number of "buildings", arranged like a Manhattan map or a chessboard.



From the left to the right a number of companies will be presented. Such a row will be repeated from front to back for different years. So if only one company is available in the input-file, only one lane of buildings will be shown from front to back.



Although the number of buildings and years is not really limited, it is wise not to exceed an area of 10 x 10 (companies x years). Otherwise the "city" will become too vast for a quick overview. However, it *is* possible to travel through the graph and look at parts of it.



Now what does a *single building* show exactly? Actually it is a special kind of 3D-graph developed several years ago by the founder of AnRep3D.

The base of the building is a rectangle, representing e.g. the total ***assets*** in one direction and e.g ***equity*** in the other. This means the equity-ratio of the company is reflected in the shape of the rectangle. As "equity" will never exceed "assets" (because it is part of it and there will always be some liabilities) a higher equity to asset-ratio will present a more square-like base. On the other side, a very narrow, long rectangle reflects a poor equity-ratio (high gearing).

The height of the building could represent the ***revenue***. A *green* layer on top (the "roof") shows the ***profit*** as part of the revenue and therefore the yellow part equals the total of ***costs***, because revenue – profit = costs. Sometimes there is no profit, but only a loss. A ***loss*** is represented by a *red* layer on top of the revenue – indicating the costs were even higher than the revenue!

Such a graph may seem complicated, but offers the possibility to judge a company´s situation at a single glance. The equity to asset ratio is easily compared between years and companies by looking at the base of the buildings. All kinds of other ratios will be recognised as well, e.g. the sides of the (flat) roof showing the return on assets (or return on equity).

Of course it's not mandatory to feed the generator with total assets and equity only. Assets and liabilities will also do and instead of the net profit the gross profit or e.g. EBIDTA or Cash flow from Operations can be taken – usually providing a larger "roof". There is a legend in the graph, hovering in front of the buildings but at a lower plane. Its labels will be read from the input-file and will explain what is visualised in the buildings.

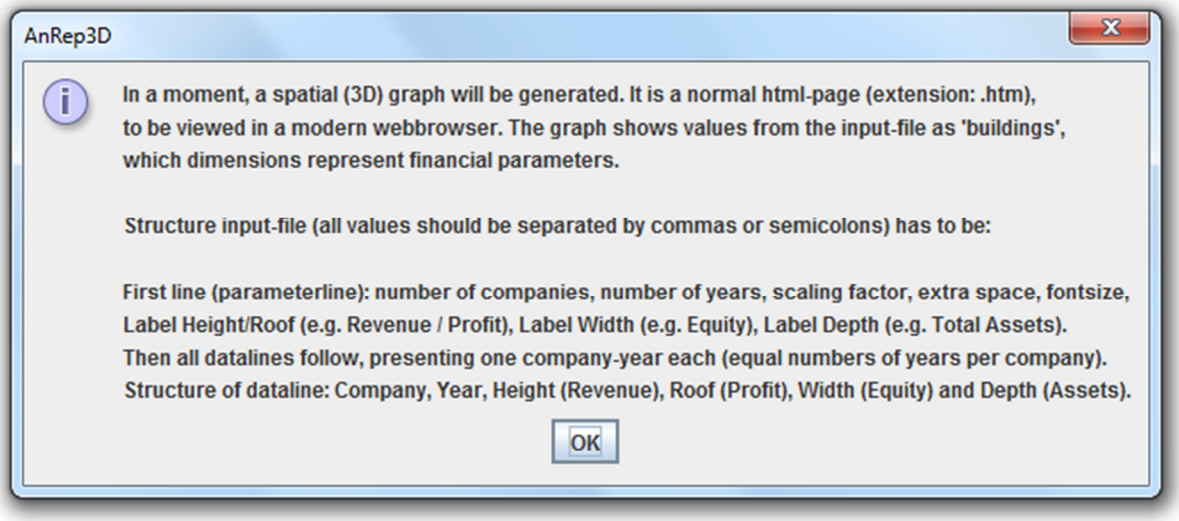
For now we will ignore these possibilities and concentrate on the set ***revenue, profit, assets, equity***. That's how we will offer data to the generator in the examples.

6.2 Generating a 3D-graph

To get some feeling about how the generator works, just convert a ready-made input-file into a 3D-graph yourself (examples come with the generator - also the free demo-version). The steps from receiving the zip-file until the 3D-graph being shown (and manipulated) on the screen are explained below:

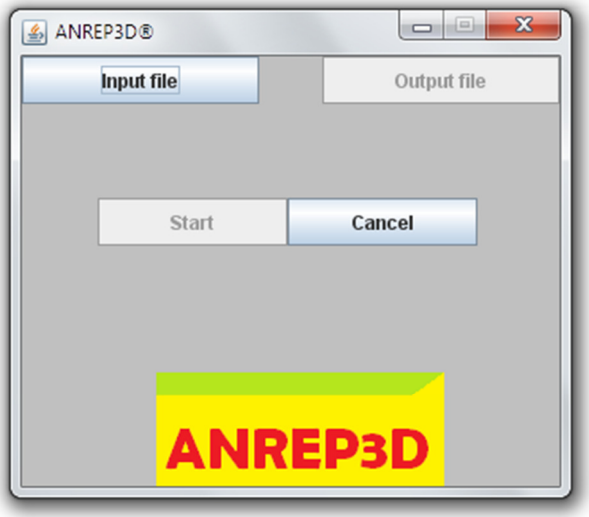
1) The generator is a kind of standalone "app" so it doesn't need any formal installation. Just create the right folder, e.g. C:\AnRep3D and unpack the zip-file in this folder.

2) Activate the generator by (double-) clicking **anrep3d.jar** and this screen will come up:

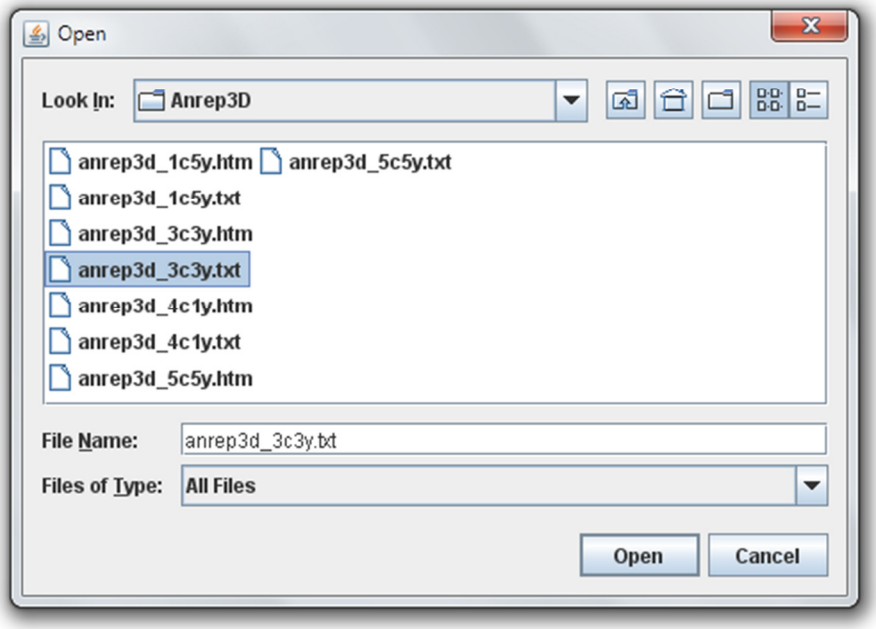


Actually it presents a very compact manual – to be ignored most of the time.

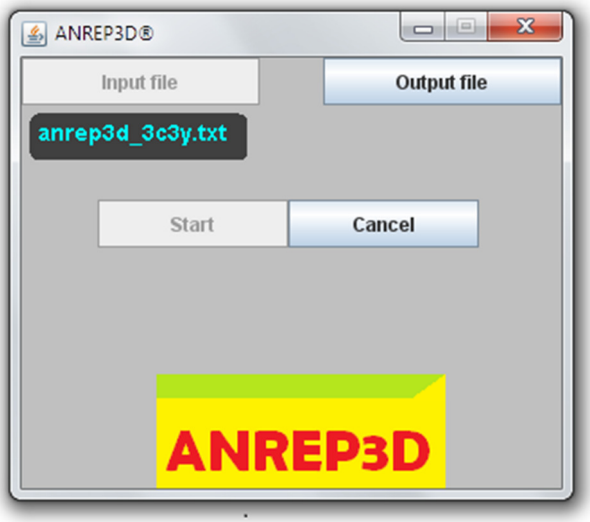
3) Click OK and click the button "Input file" on the next screen.



4) Now an Explorer screen comes up, starting in the folder where the generator (the .jar file) is. The examples of input-files are in the same folder and will be shown.
The **.txt** files are the input-files (any file-extension representing a normal plain ASCII-set will do). The **.htm**-files will also be visible, but are examples of output-files and won't be accepted as a source.

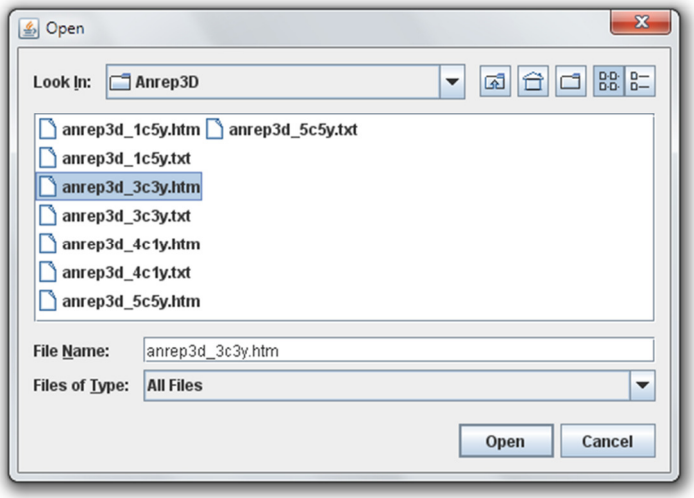


5) Select one of the .txt-files (any extension presenting a plain text-format will do – also a **.csv**-format without quotes, but for now we go with a .txt-file.
As soon as the file is selected, by double clicking - or selecting and clicking the button "open" - the next screen will appear. The name of the selected file will be shown below the "Input file" button. By the way: 3c3y is short for "three companies over three years".

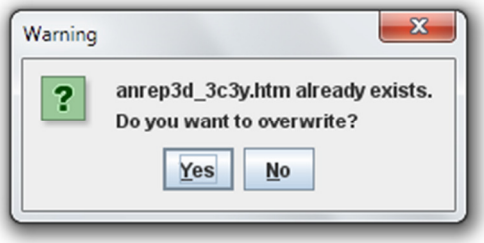


6) Now click the button "Output file" and the same Explorer screen will come up again.
This time we can enter a name for a ***new*** file ***or overwrite*** an existing one.
As the output-format is .htm, only an .htm-file will be overwritten. If the name of an existing file with another extension is used, the output-file will have the same name but with the extension changed to .htm, making it a different file. This means it is easy to select the input-file again to create an output-file with the same name but another extension (.htm).

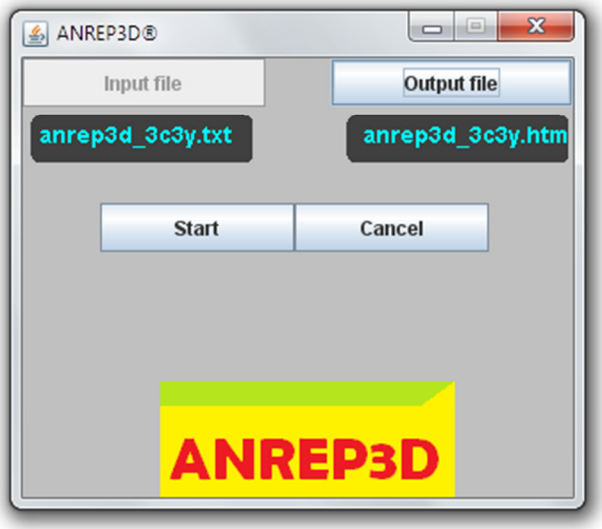
This time, we will just select the existing .htm-file (anrep3d_3c3y.htm) corresponding to the input-file (anrep3d_3c3y.txt), although you might enter a filename of your own choice in the empty bar of course, like e.g. ***demo3c3y.htm***



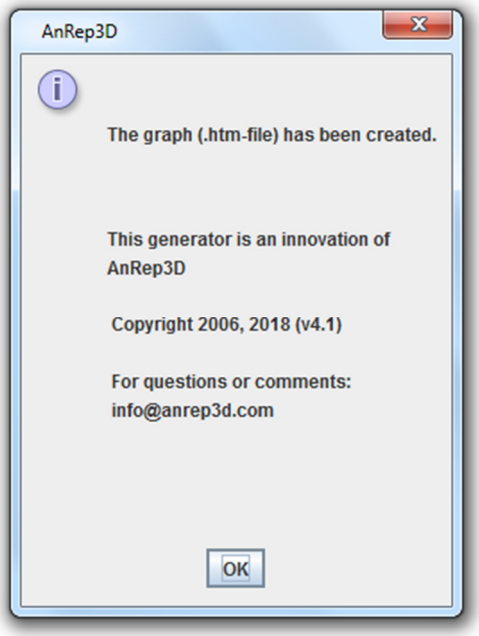
7) Before overwriting an existing file (in case you didn't enter your own filename but selected the existing one as we advised above), the generator checks if it's not a mistake.



8) Click yes and this time the name of the output-file will appear under the button with "Output file" on it.

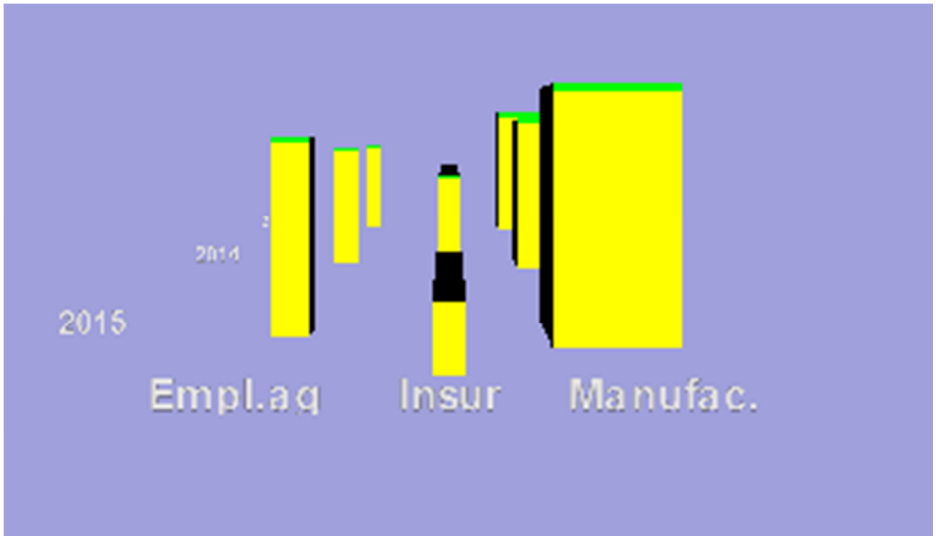


9) If you still made a mistake, just press the "Cancel" button to stop the session.
If you want to proceed, click the start-button and the data in the input-file will be processed.
The result will be a graph in .htm-format, written into the output-file.
As soon as the processing is completed, a confirmation-screen will appear. Your version will refer to your licence.

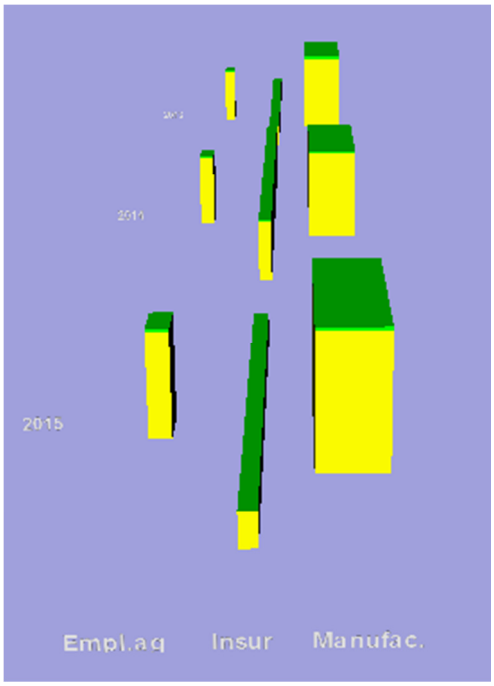


10) This is only a message, so click "OK" to proceed and the generator will close itself again. The graph is available (like it already was as an existing .htm-file was overwritten in this example).

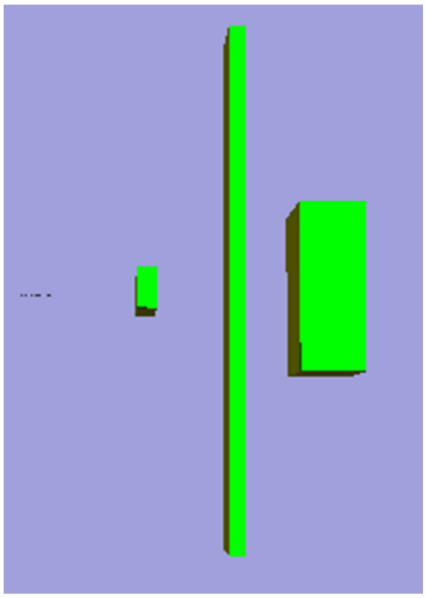
11) Go to the folder where you unpacked the .zip-file and double click the (most recent) .htm-file (either overwritten or created) and the graph will appear in the browser (of course the .htm-extension has to be associated with your favourite browser).
The screen will look like the picture below.



By clicking somewhere in the graph, holding the button and moving the mouse, the graph will tilt and/or rotate. The result might be like the screenshot shown below.



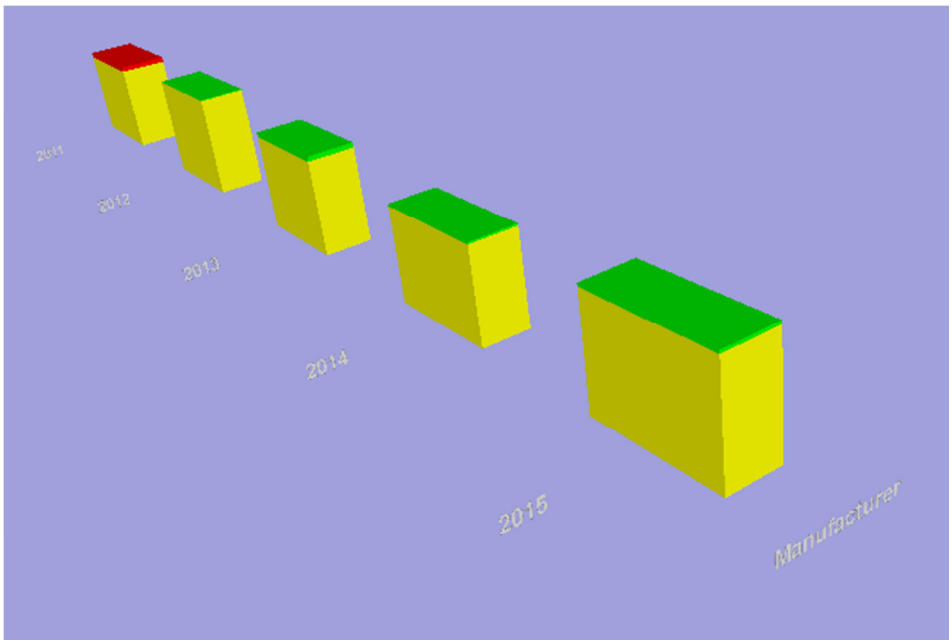
The green "roofs" indicate a positive profit. If we tilt the graph even further, we will see the top of the roofs and as their shapes are equal to the base, they reveal the equity-ratio.



The ratio of equity (width of the building) to total assets (length of the building) is clearly visible now. The insurance company has a high gearing (leverage), because the length of the building (total assets) is over twenty times the width (equity)!

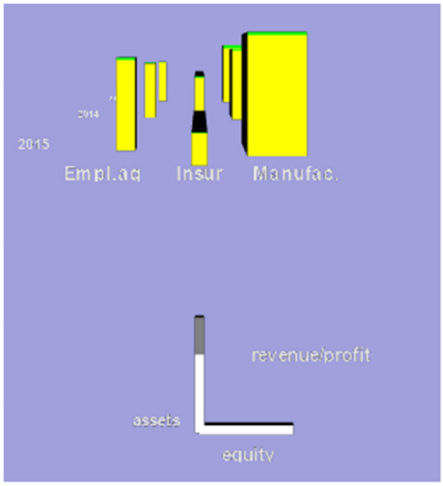
We can also see the Manufacturing company not only having more assets than the Employment agency, but also much more equity. Yet the revenues don't differ a lot. However, their gearing is slightly different.

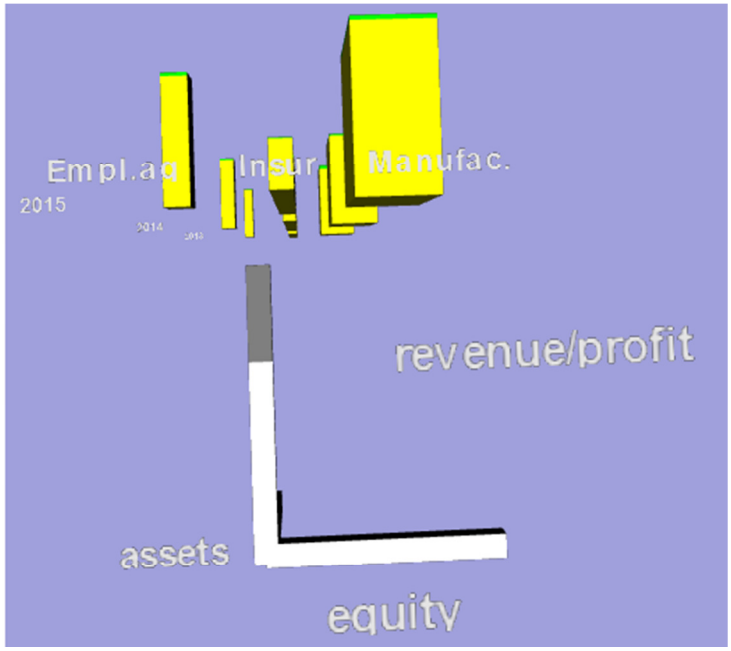
Looking at a single company over a couple of years (e.g. the graph ***anrep3d_1c5y.htm***, created from anrep3d_1c5y.txt), we are able to look at the different revenues through a series of years. There was a loss in 2011, shown by the red roof. It wasn't a small loss because the roof has a visible thickness, but the next year (2012) a small profit was reached and the year after (2013), there was even a substantial profit.



We can also compare the equity to assets ratio throughout the years, but a screenshot just cannot show the power of a real 3D graph! So take a look at the real 3D-graph in your browser.

Although the default meaning of height (roof), width and depth of a building is Revenue (Profit), Equity and Total Assets, other values can be chosen. The roof has to be a part of the total height (unless it's a negative value, than it will be on top of the building and in red), but that's the only limitation. To tell the observer what is shown, there is a 3D legend in front of the buildings, showing the meaning of the three dimensions. The labels will be read from the input-file. Below are two screenshots showing the legend – one in a regular view and the other one presenting a close-up.





6.3 The input-file

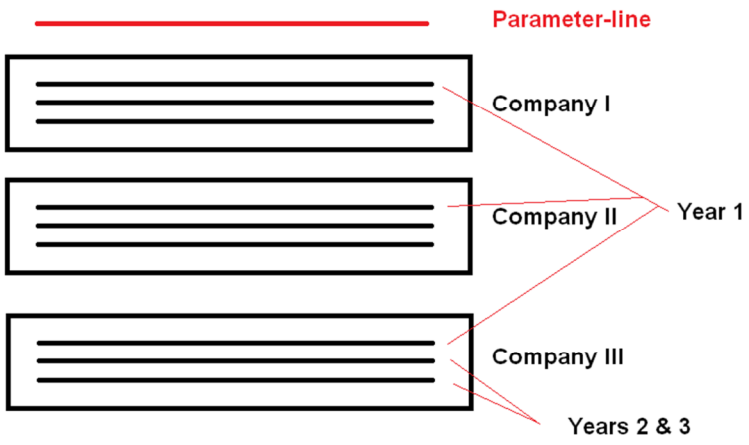
Now let's take a closer look at the input-file, feeding the generator with data together with some parameters giving it directions. Before we start it is very important to know all individual values in the input-file should be separated by comma's (,) or semicolons (;). If a .csv-file (MS-DOS type) is created by e.g. Excel, usually the values will be separated by semicolons. When entering values by hand comma's might be more convenient, but even a mix is allowed. Of course the decimal separator has to be a point, to avoid confusion.

6.4 The first line is a parameter-line

The first line does not contain the real data but offers a set of parameters to the generator telling it how to read the file and how to shape the graph.

The data-set consists of lines representing companies and years. For every company the same number of lines has to be provided (if no values are available, zeroes will do). This number of lines is equal to the number of years. The number of blocks is determined by the number of companies.

The general structure of an input-file is presented below. Here the first two values in the parameter-line would be 3,3 (3 companies = blocks with 3 years = lines each).



No automatic detection is provided (yet), to keep the generator lean and cheap. It is possible to select a lower number of companies than actually present. One or more blocks at the end of the set will be ignored. It's also possible to add some comment or explanation after the last line of coordinates without getting an error as this line won't be read. If data – e.g. one year for one company – is missing, just put in zeroes to keep the structure intact.

6.4.1 Number of companies, number of years

The number of companies and the number of years (the same for each company) both have to be mentioned first and indeed those numbers are the first and second value at the parameter-line. So 4,2 means four companies with two years each – eight data-lines expected in total (four blocks of two lines). However, 2,4 means two companies with four years each – quite a difference (two blocks of four lines) although still eight data-lines have to be present.

6.4.2 Scaling

After another comma or semicolon the scaling-factor has to be mentioned. Usually the values in an annual report will be in thousands or even millions. To obtain values preferably in the range of zero to one hundred (0 – 100), the scaling-factor is able reduce the values this range. If e.g. revenue, profit, equity and assets are like 16225, 179 , 2899, 7595 these values have to be reduced by a magnitude of a thousand and therefore the scaling-factor could be 1000. Of course 200 or 500 could also do very well, creating larger graphs. Please try different values to discover which resulting range (values divided by the scaling factor) suits you. Don't use decimals.
Be aware that the scaling factor applies to all directions to avoid unintended distortion of the ratios shown.

6.4.3 Additional space between buildings

The space between the "buildings" will be derived automatically from the input shaping them. However, you might need some extra spacing e.g. to fit the labels. That's why there is a spacing-factor as well. This factor is a percentage of extra space added and has to be between 0 and 100. Zero will add 0% of extra space, so no change and 100% will double the space between the buildings. Again, don't use decimals.

6.4.4 Font-size

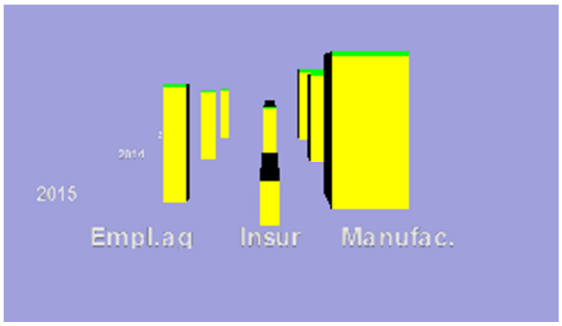
Finally a value for the font-size has to be provided. Although a value between 1 and 10 is allowed, very small fonts can be illegible. Four (4) is advised as a default. To be able to fine-tune the size, decimals are allowed here. Use a point as a decimal separator.

6.4.5 Legend-labels

For height, width and depth labels will be provided in the parameter-line. Even if the default values Revenue/Profit, Equity and (Total) Assets will be presented, it is wise to add those labels. After the font-size a comma or semicolon has to be added and then the labels will follow – also separated by commas or semicolons. Using four labels (a separate one for the roof) turned out to be confusing, so the roof label has to be a part of the label for height. E.g. use something like **Revenue/Profit** or **Revenue [Profit].** Within the company-labels and the labels for the legend, special characters like ( ) { } [ ] | / and : (colon) are allowed. Of course a comma (,) or semicolon (;) should be avoided as those characters are separators, read by the generator itself!

6.4.6 Example

The parameter-line in the input-file anrep3d_3c3y.txt we used to present an example in chapter 1 (the screenshot of a fixed position is repeated below) has the following parameters at its first line:
3, 3; 1000, 30, 3, revenue(profit), equity; assets
Remember that both commas and semicolons are allowed as separators and here a mix was used to emphasise the free choice.



Three companies: each with a set of values for three years. Hence the “3, 3” at the start of the parameter-line.
The values had to be divided by one-thousand to get a graph fitting in the window so the scaling factor was set to 1000. There was a need for 30% extra spacing, mainly to give the labels some room. Finally a font-size 3 was chosen. Instead of the scaling factor 1000, smaller values could have been in the data-lines, using decimal points. It’s allowed, but as most annual reports provide integers, it’s more confortable to avoid decimals, using the scaling factor.

6.5 Data-lines

All relevant parameters have to be in the first line and the *lines after this line* will provide the data to be shown in the graph. The number of lines has to be the product of the first two values in the parameter-line, because every company has the same set of years. In the example above 3, 3 meant three companies with three lines each, hence nine lines.

6.5.1 A data-line has a fixed structure

First the company is mentioned, then the year e.g: *Insur, 2016* but of course that's not enough as we need the four values determining the size of the building and the roof. After the year the values for the height (e.g. revenue), roof (e.g. profit), width (e.g. equity) and depth (e.g. total assets) of a building have to follow. So in this example the complete sequence on a data-line will be: **Company, Year, Revenue, Profit, Equity and Assets**

As an example in real values:
Insur, 2015, 6011, 128, 2823, 73468

or the other valid option
Insur; 2015; 6011; 128; 2823; 73468

or even something like:
Insur,2015;6011,128,2823;73468

6.5.2 Labels

The company-name will be presented as a label. We didn't describe the company as "Insurance company" but only as "Insur" for a good reason. We want the labels to be compact and therefore used abbreviations. Of course it is possible to use larger labels like "Insurer" or even "Insurance company" and raise the space between building and/or reduce the font-size to avoid overlapping labels. Just try and discover what works out well.

Only every first company-name is relevant
If we are completely honest, companies and years are not read from every single line.
For every company-block the company-label is taken from the first line in the block.

So if we would enter something like:
Insur, 2015, 6011, 128, 2823, 73468

but the next line would have a different company-label:
Who_Knows, 2014, 13797, 361, 2746, 89979

The company-label will still show "Insur" because the label is derived from the first line in the block and the "Who_knows" entry is just ignored.

Year-labels are read from the first block
Something similar happens with the year-labels. This time it's a little bit more complicated.

As every company-block should have the same set of years, the labels will be derived from the first company-block only. Let's consider two company-blocks:

Empl.ag, 2015, 19219, 519, 3862, 7309
Empl.ag, 2014, 17250, 340, 3313, 6778
Empl.ag, 2013, 16568, 231, 2908, 6608
Insur, 2015, 6011, 128, 2823, 73468
Insur, 2014, 13797, 361, 2746, 89979
Insur, 2013, 5899, 168, 2930, 76515

If we would change the year-labels in the second block, like this:

Empl.ag, 2015, 19219, 519, 3862, 7309
Empl.ag, 2014, 17250, 340, 3313, 6778
Empl.ag, 2013, 16568, 231, 2908, 6608
Insur, 1871, 6011, 128, 2823, 73468
Insur, 1970, 13797, 361, 2746, 89979
Insur, 1241, 5899, 168, 2930, 76515

The graph would still be the same as only the year-labels from the first company-block (the Employment Agency) will be used. The ones marked in red will be ignored.
So be sure the year-values are the same – and in the same order for that matter! – for every company-block. Otherwise the graph could offer a comparison of apples and oranges. The example below illustrates such an error:

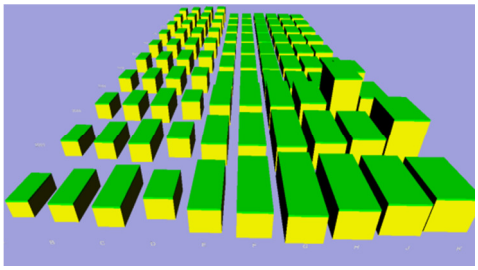
Empl.ag, 2015, 19219, 519, 3862, 7309
Empl.ag, 2014, 17250, 340, 3313, 6778
Empl.ag, 2013, 16568, 231, 2908, 6608
Insur, 2015, 6011, 128, 2823, 73468
Insur, **2013**, 13797, 361, 2746, 89979
Insur, 2014, 5899, 168, 2930, 76515

For the second block (Insur), the order is reversed for the second and third year. As a result the 2014 building for the Insurance company will be shown next to the 2013 building of the Employment Agency. This could lead us to wrong conclusions. Even the time-range for the company itself could be confusing. If e.g. 2013 had as loss (negative profit) and 2014 the profit was positive again – so an improved situation – it would seem to be the opposite. Of course this will not happen by just entering a wrong year-label, but only if the whole line actually was switched to another position. That's why we advise to always put the real labels (Company and year) in front of the line. Doing so, the input-file will be easier to read and understand for a human user!

6.5.3 Some practical advise

As mentioned before, never forget to enter the mandatory parameters on the first line with the *data-lines only starting from the second line.* Theoretically there is not a real limit for the number of companies and years to be provided. However, we advise however to limit the number of companies and years to a reasonable number. Ten companies and ten years (one hundred data-lines!) will already present a pretty crowded graph, but it could still be a useful one of course. Try and find the

best combination and be aware of the fact that very large graphs can react slowly or even lag when manipulated in the browser-window.

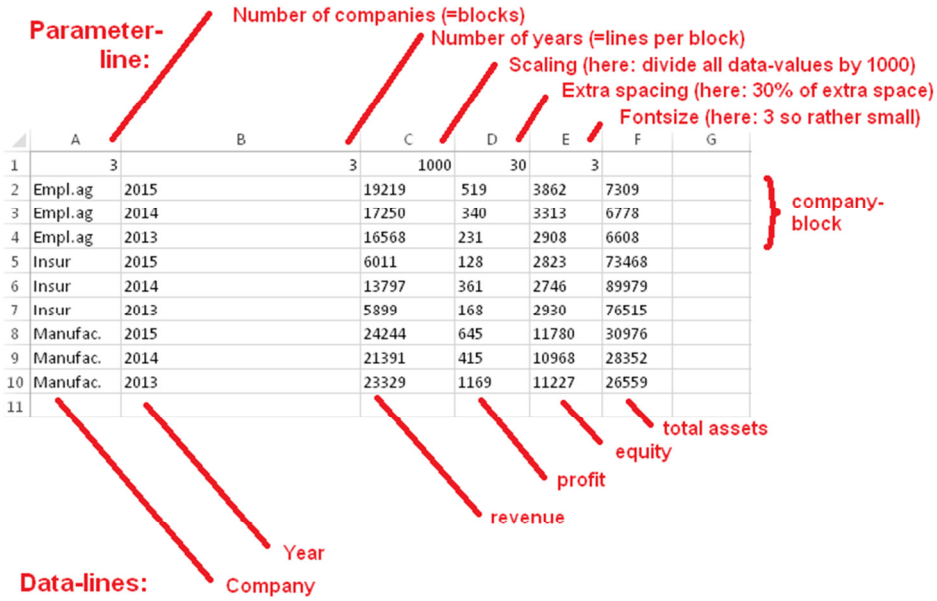


6.5.4 Creating an input-file with Excel

Of course a readymade input-file is nice to give the generator a first try, but it is not what the generator was purchased for. Presumably the objective was creating your own spatial (3D) graphs, based on your own sets of data. Although entering these data by hand in notepad is a possibility, nowadays most data are stored in Excel-sheets.

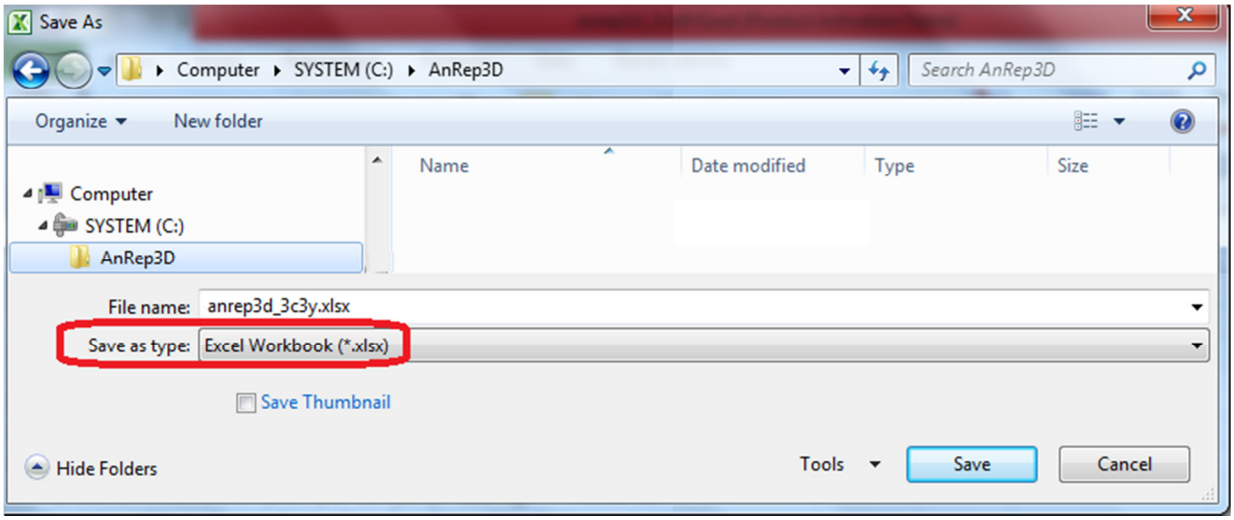
To be able to present Excel-data to the generator, the set should be stored as a .csv-file. This file will contain semicolons as a default. Be aware of the comma being a separator as well, so a decimal point has to be a *point* indeed! But then decimals are never allowed in the parameter-line, except for the font-size. In the data-lines decimals can be applied.

Below an example of the input-file ***anrep3d_3c3y.txt*** is given, now shown in an Excel-grid, with comments added.

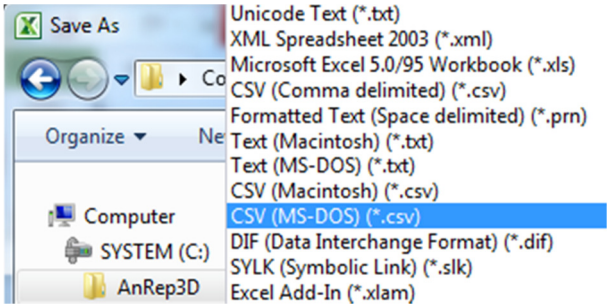


	A	B	C	D	E	F	G
1	3	3	1000	30	3		
2	Empl.ag	2015	19219	519	3862	7309	
3	Empl.ag	2014	17250	340	3313	6778	
4	Empl.ag	2013	16568	231	2908	6608	
5	Insur	2015	6011	128	2823	73468	
6	Insur	2014	13797	361	2746	89979	
7	Insur	2013	5899	168	2930	76515	
8	Manufac.	2015	24244	645	11780	30976	
9	Manufac.	2014	21391	415	10968	28352	
10	Manufac.	2013	23329	1169	11227	26559	
11							

To save such an Excel-worksheet as a .csv-file, go to "save as", look for the right folder and enter the desired file-name. Before saving, click at the bar offering different formats.

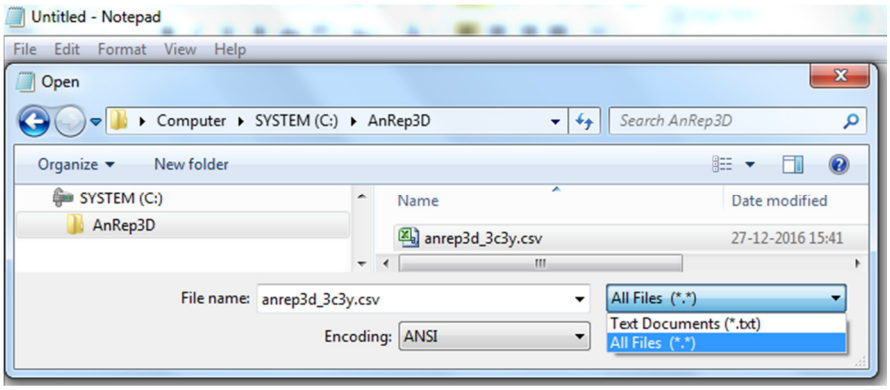


Then, select one of the available .csv-formats, e.g. the CSV (MS-DOS) (*.csv)

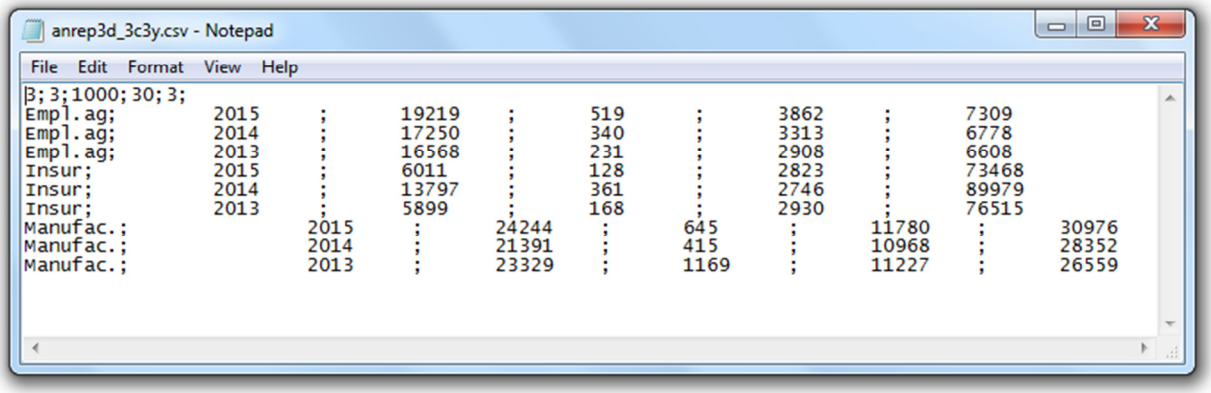


There will be a warning about loss of data. Be sure you already saved the Excel-original before proceeding with the .csv-format, because this format does not support all Excel-features. E.g. other tabs will be lost when savings as a .csv-file.

The .csv-format is a general format, but often it will be associated with Excel. Nevertheless it can be read by a simple editor like Notepad (right-click and choose "open with").
Many Notepad-versions will only look for .txt-files however, so when opening from Notepad, set the file-type to "all files" and it will also show the .csv-files too.



The result is a clean .csv to be read by the AnRep3D-generator.



Remember that semi-colons are fine as a separator. The generator will understand comma's and semicolons and even a mix of both.

To change setting in the parameter-line, just use the .csv-version to edit. It's also possible to keep the original data – without a parameter-line – in Excel and add the parameter-line afterwards in the .csv version, using Notepad.

The description above is just an example of how to create a data-file. You can also create the set in e.g. Word and copy-paste them to an empty file in .txt-format. And of course other Office-packages like ***Libre Office*** will do as well.

6.5.5 Setting the parameters in th parameter-line

We will use the input-file AnRep3D_3c3y.txt as an example. It has this parameter-line:

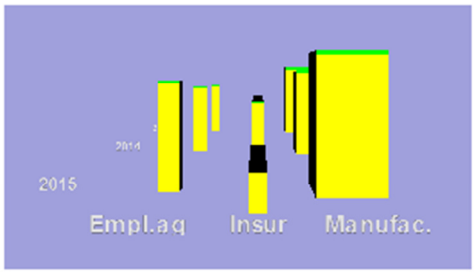
3, 3; 1000, 30, 3, revenue(profit), equity; assets

(the semi-colons are in to show they as a separator as valid as a comma).

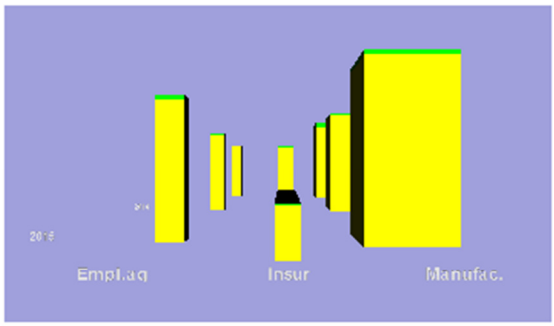
The first two values – **3, 3** – have to represent the number of companies and years, so we cannot alter them freely as long as we stick to the three companies and three years as found in the data-lines.

6.5.6 Changing the scaling-factor

The next parameter-value however – **1000** – could be changed to 500 or 2000 but it will change the appearance of the graph. The graph at the top is the original one (with scaling 1000).



scaling = 1000



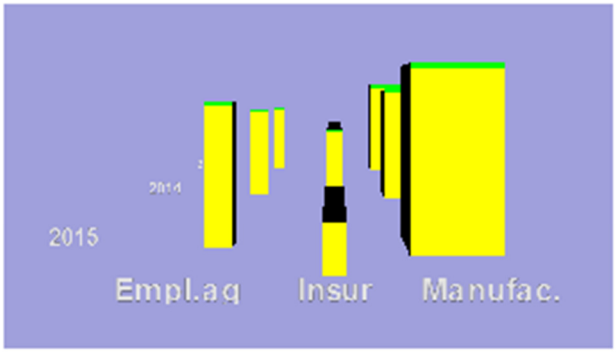
scaling = 500

scaling = 2000

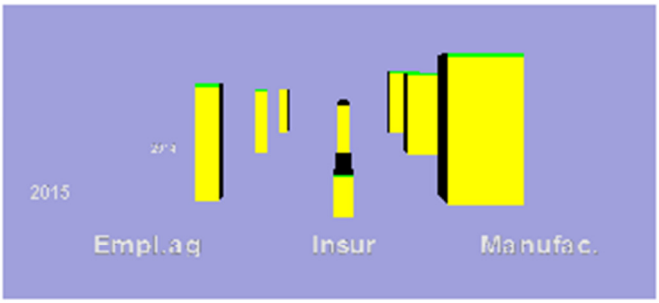
With a smaller scaling factor (like 500) the original values will be reduced less and therefore the "buildings" in the graph will be larger. At the same time the graph will move a little bit more to the background, because the distances are related to the size of the buildings. Although the font-size remains unchanged, the labels look slightly smaller because of the larger distance to the observer. A higher scaling-factor (e.g. 2000) will reduce the values more and as a result the "buildings" will be smaller. At the same time the space between the buildings will be reduced and as a result the labels – even with the same font-size – will overlap each other. This can be corrected by reducing the font-size of course, but also by adding more space by raising the value of the spacing-parameter as discussed in the next section.

6.5.7 Varying the spacing-factor

The value after the scaling factor, so the fourth value at the parameter-line, is the spacing-factor. When calculating the graph, the space will be related to the size of the buildings. Usually this will do, but with larger labels (or relatively large labels), sometimes more space is needed to prevent the labels from overlapping. The spacing parameter is a percentage, presented as a number, ranging from 0 to 100 and offering 0 - 100% of extra space between buildings (and between the graph and the observer). Below a screenshot of the original graph (extra space 30%) is compared with an alternative version (extra space is 60%). The distance between building is larger but also the distance between buildings and labels and even between graph and observer.



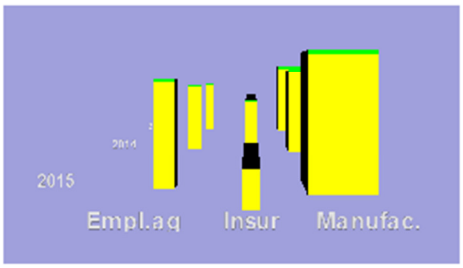
extra space 30%



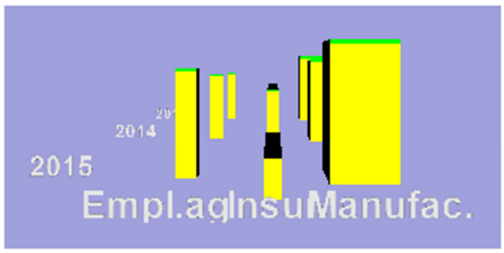
extra space 60%

6.5.8 Changing the font-size

When the font is small, it can be hard to read the texts. On the other hand a larger font-size could cause an overlap of the labels. Of course this can be corrected by adding extra space.
The opposite is also true: if extra space is added a larger font could be useful.
In the example below the original graph (font-size 3) is shown to the left. The graph to the right was generated from the same data-set with only the font-size set to 5. As a result the labels overlap each other in the same way as with a reduced scaling-factor.



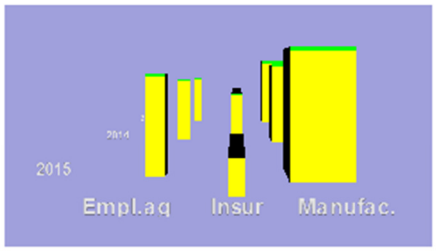
fontsize 3



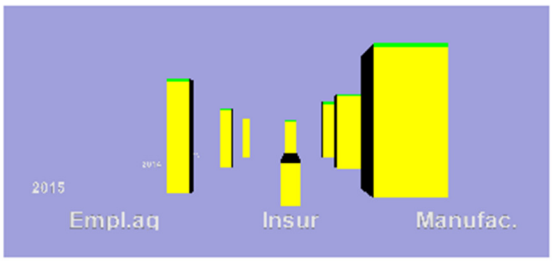
fontsize 5

6.5.9 Balancing scaling, extra space and font-size

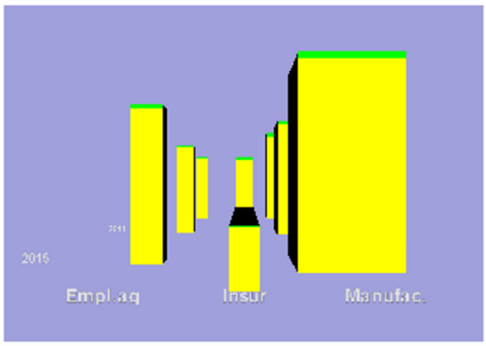
By varying scaling, extra space and font the right balance can be obtained.
Below some examples of different parameters applied to the same data-set.



scaling = 1000 extra space = 30% fontsize = 3



scaling = 500 extra space = 60% fontsize = 5



scaling 500 extra space = 0% fontsize 3

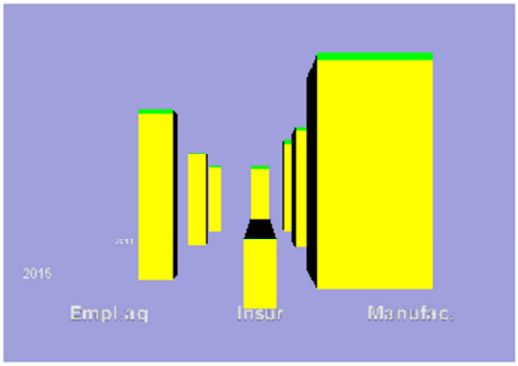
6.6 Manipulating the graph

If a 3D-graph would be nothing more than the screenshots shown in the previous section, the extra dimension would not add a lot of value. In reality the graph can be zoomed in and out, rotated and tilted to get the preferred angle needed to get a better understanding.
We already noticed the graph can be closer or further away as a result of different parameters applied.

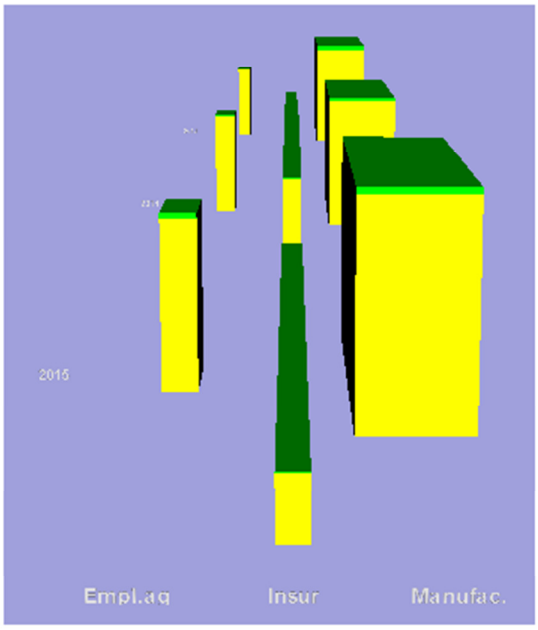
By *left-clicking* the mouse while pointing somewhere in the graph and moving the mouse at the same time, the graph can be tilted and rotated. By *double-clicking* somewhere in the middle of the graph (easier when tilted first) the graph will zoom in.

Below the original position of one of the alternative versions of the graph is shown to the left. To the right the same graph is shown, but now the front-side is tilted down and the graph is closer because it was zoomed in.

The manipulations described are not invented by AnRep3D but are just general WebGL/HTML5 functionality, so please have a look at this standard to find out more.



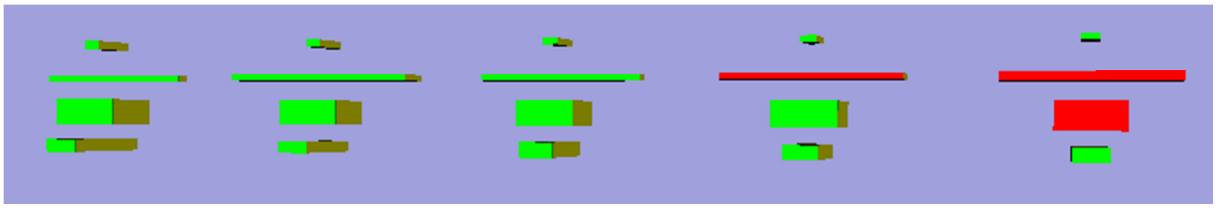
scaling 500 extra space = 0% fontsize 3



scaling 500 extra space = 0% fontsize 3

now brought closer and slightly tilted

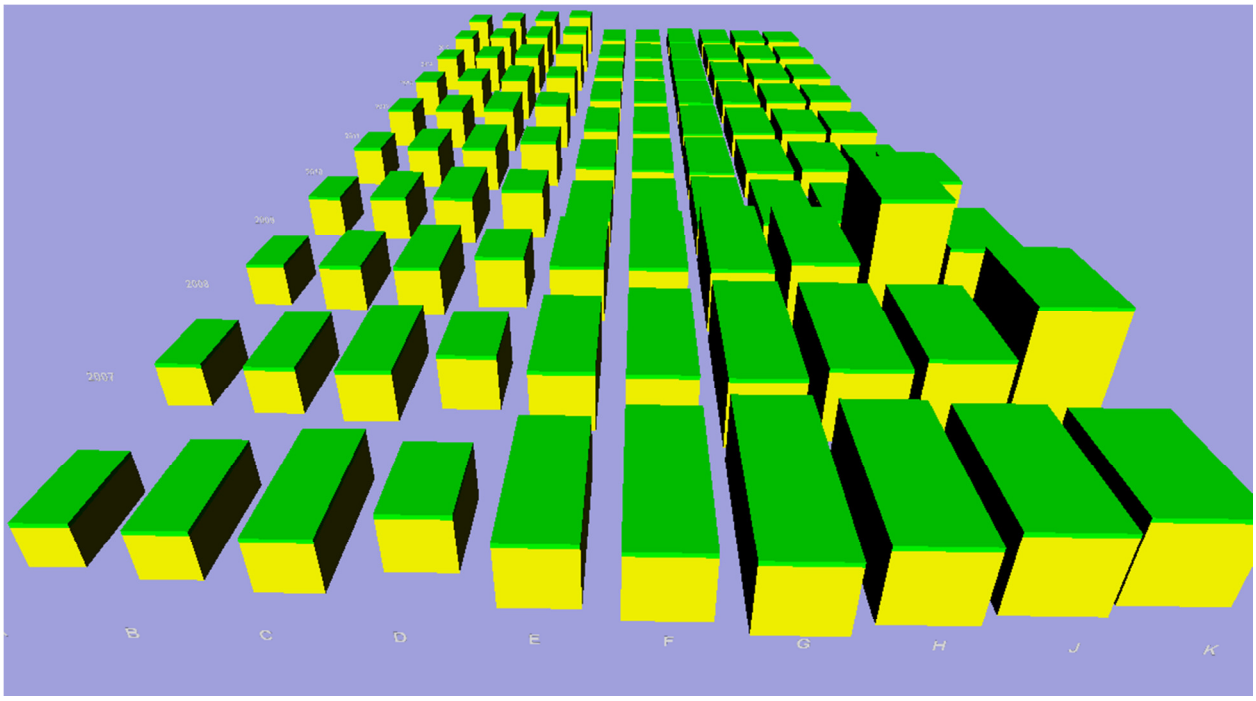
Tilting the graph will also help to see patterns of profit and loss throughout the years. The red roofs represent loss, the green ones profit (four companies during five years shown).



6.7 Adding companies and years

Until now we used an input-file with only three companies and three years. The generator is able to process a much higher number of lines. Actually the limit is the resulting graph. If it becomes too crowded, it will be hard to discover special patters, trends or exceptions. At the same time, manipulating the graph might get difficult because it will respond slower.

To illustrate the power of the generator a graph is generated using ten (fictive) companies called A to K (the confusing I was left out) during ten years – presented from past to present.
The graph was tilted a little bit to provide a full overview. All roofs are green because none of these fictive companies ever showed a loss.

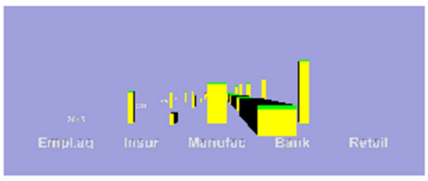


Although it is possible to process a hundred lines of data, the graph will not be very informative but the choice is completely yours!

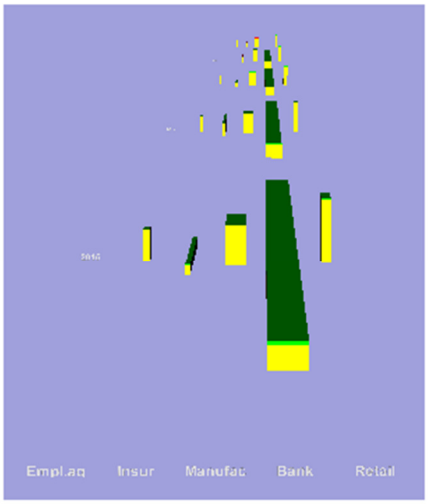
Now let's get back to five very different companies, by adding a bank and a retailer to our original series with an employment agency, an insurance company and a manufacturer. At the same time we add two years so the first two parameters will become **5,5** (of course it could be 3, 5 or 5,3 or any other combination for that matter, but we just took equal numbers again). The retailer fits very well in the series, but we get a very strange graph because of the bank.
From the most recent crisis we already know banks have a remarkably equity to assets ratio (and therefore a high gearing), because their core-business is to lend and borrow money. Money remaining in the bank does not generate any revenue. Yet it is surprising to see the shape of the "building" representing the bank. It's a small but quite normal bank, although it had to be saved by the government. The shape is much like the insurance company, but the bank is larger and therefore it influences the graph strongly.

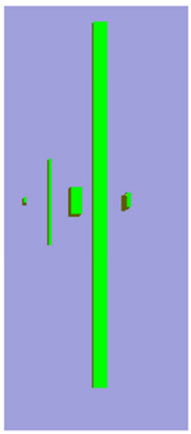
The distance between the buildings depends primarily on their size. To obtain a chessboard-like plan, the building with the largest width and the building with the larges depth (not necessarily the same one) will determine the size of the area available for each and every building. As the bank has an extremely high value for "assets", it forces every field in the chessboard-like pattern to become very, very deep (seen from front to rear).

The three screenshots below are taken from the same graph, but in different positions.



scaling = 1000 extra spacing 50% font = 5





To the left the starting position of the graph is shown. In the next one, the graph is slightly tilted. The third screenshot presents a top-view for only one year. Now the impact of the relative low equity-ratio in combination with a very high value for the assets compared to the other companies, becomes very clear.

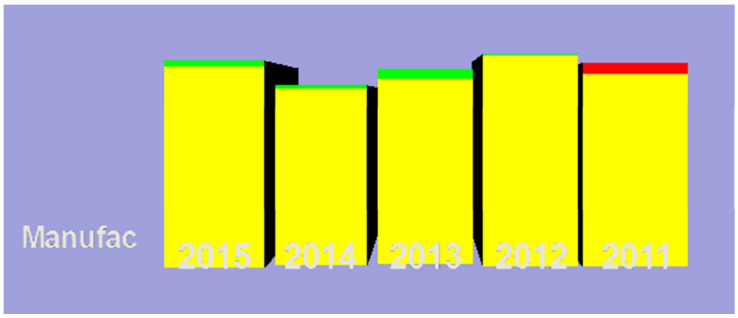
By looking at the graph a lot of discoveries can be done. The discoveries could lead to hypotheses to be tested at the actual numbers. Visualisation should only be the first step of course!

6.8 Deviating from the original idea

Although the AnRep3D generator was designed with special ideas in mind, it turned out to be rather resilient and as a result all kinds of modifications are allowed. In this paragraph we will discuss some alternative use cases.

6.8.1 Switching companies and years

We emphasised the structure in the data-lines in the input-file should be company first, year next. As a matter of fact it could be reversed as well. Sometimes it is easier to do so. But remember, this will only change the arrangement of the buildings in the graph, not the shape of the building.

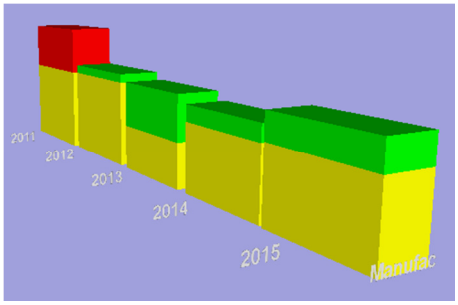


The related input-file looks like this:

```
5,1;1000,0, 4
2015    ,Manufac,          24244   ,       645     ,       11780   ,       30976
2014    ,Manufac,          21391   ,       415     ,       10968   ,       28352
2013    ,Manufac,          23329   ,       1169    ,       11227   ,       26559
2012    ,Manufac,          25085   ,       226     ,       11174   ,       29079
2011    ,Manufac,          22704   ,       -1295   ,       12350   ,       29395
```

6.8.2 Taking EBIT, EBIDTA or gross margin instead of profit

It can be useful to take another measure for the performance of a company. These values are often much higher and will provide a larger roof. The legend will show what is represented in each and every dimension, but don't forget to add the labels to the parameter-line!



6.8.3 Showing profit without revenue

If return on assets and return on equity are much more interesting than the revenue, the revenue could even be replaced by gross profit (or EBIDTA and so on), entering simply the same value for the next position (the roof). The buildings will be lower and completely red or green of course. Again, it would be wise to make a note in the input-file.

6.8.4 Replacing Equity or Assets with Liabilities

For some purposes it can be worthwhile to replace equity or assets with liabilities. Taking the latter option, the base of the buildings will represent the gearing. Combining Assets and Liabilities will change the shape of the base a lot. Often, it will be more like a square (although not completely of course).

6.8.5 Thinking completely outside the box

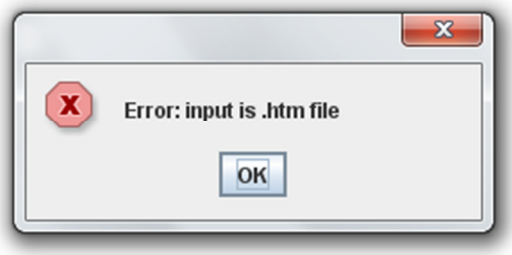
Nobody forces you to enter the values as shown in this manual. Actually we only offer a 3D graph able to show a couple of values in comparison (within a building and between buildings). So if you want to use the *height for customer rating* and the *width for unit-price* and the *depth for volume sold* – with the buildings representing different products, nobody will complain about it.

6.8.6 Using the legend

For creative new ideas, but also for a standard graph, it's wise to choose the labels for the legend well. Labels for height, width and depth can be provided. The height consists of two parts (or actually three: the total height, the roof and the remaining yellow part). The label will describe the full height, but the meaning of the roof can be added, using e.g. brackets like ( ), { } or [ ]. A slash / or colon : is also an option but never use a comma or semicolon in a label!

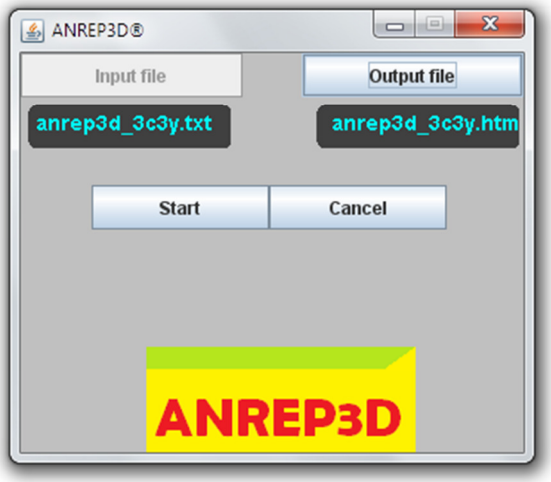
6.9 Errors

As soon as an .htm-file output-file) is selected as an input-file, by double-clicking or the "Open" button, an error-message will appear.

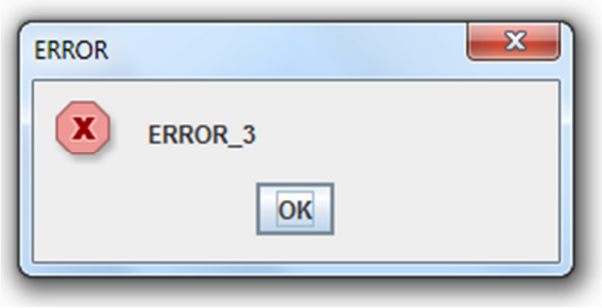


So no chance of selecting an input-file by mistake!

If you made a mistake anyway, there is still a chance to correct it by pressing the "Cancel" button. This will abort the whole session.



If the input-file isn't structured the right way, a generic error-message will come up.



In such a case please check for commas and/or semicolons, the number of entries in each line and the structure of the parameter-line.

6.10 Disclaimer

The AnRep3D company and its developers are not responsible for any damage or other potentially harmful effects caused directly or indirectly by installation or use of the generator-program or associated files.

Never draw conclusions from a generated 3D-graph only. Consider a discovery to be a hypothesis and check it against the original data first!

All Company-names and Trademarks mentioned are the property of their respective owners.

Please have a look at our website: https://anrep3d.com
or at our blog: https://anrep3d.wordpress.com
Contact us at mailto:info@anrep3d.com

7 Technical facts about AnRep3D

7.1 Again, some history

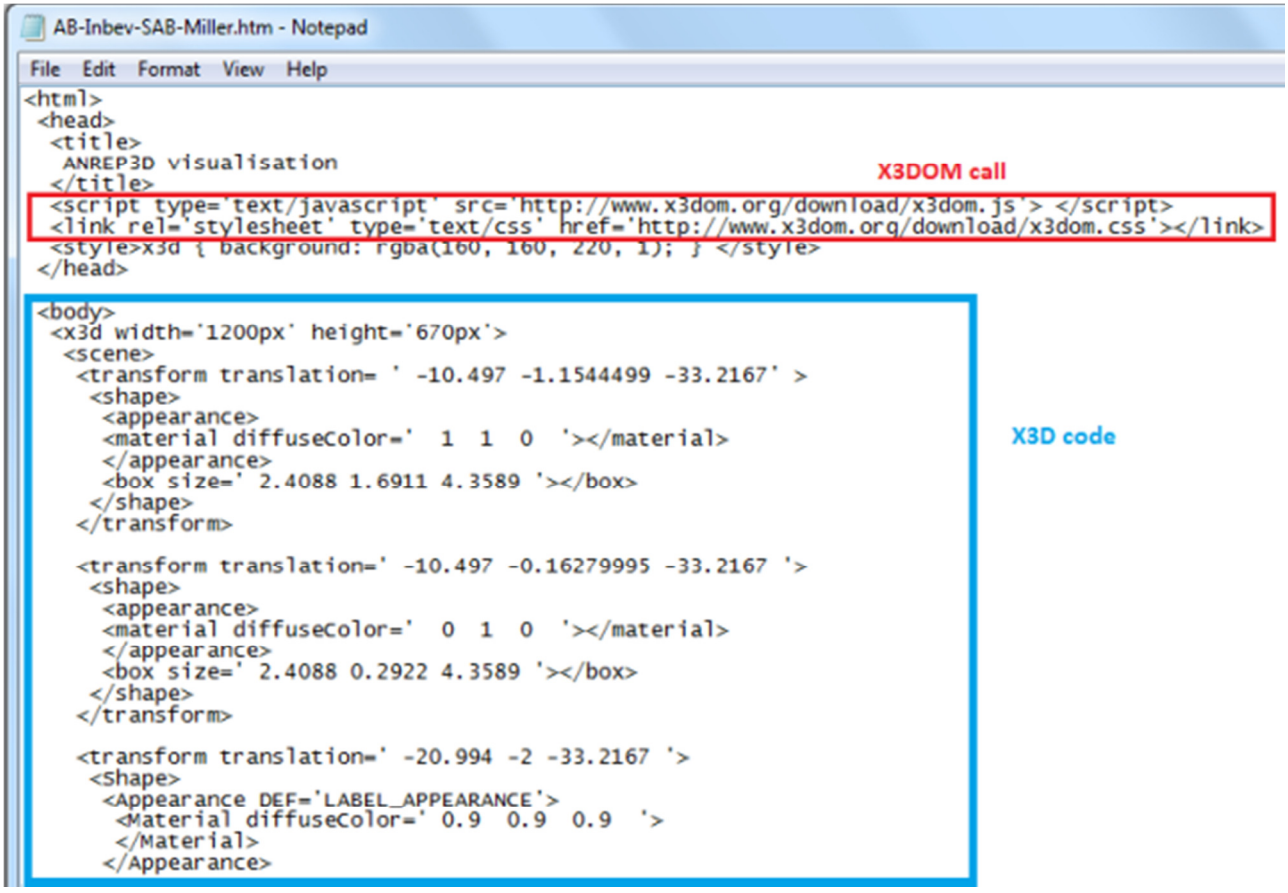
In the past, the AnRep3D-generator created graphs in the independent language VRML, used for 3D models. VRML is a formally obsolete language, able to present 2D and 3D models. Formally, because it has a successor called X3D. Yet VRML is still around and X3D as a pure language never became very well-known let alone broadly accepted. Since AnRep3D-graphs are 3D models, choosing VRML made sense in 2006 but there was also a disadvantage which became more important over the years. VRML can only be read by specific viewers or in a web-browser if a plug-in is installed. The number of viewers available over the year declined and on top of this, a kind of crusade against browser plug-ins started. People neither wanted to install a plug-in, nor did they download a viewer. In the end this meant nobody was looking at the AnRep3D-graphs (and VRBI for that matter). Switching to X3D wouldn't have solved the issue as it needs a viewer as well and hardly any browser-plugins are available.

In 2016 we knew we had to work on an alternative, to keep AnRep3D alive. To be able to show 3D models in a browser, HTML5 was (and still is) the best standard. Yet HTML5 relies heavily on JavaScript. JavaScript is a terrible language (not related to Java at all by the way) but became more or less the standard in dynamic webpages anyway and it looked like we had no choice.

Luckily, Germany has always been very active in Virtual Reality and related subjects like VRML. Of course the research-groups there encountered the same problems as AnRep3D did and the Fraunhofer Institute for Computer Graphics Research (IGD) created a very smart solution. It offers a cloud-service called X3DOM, able to convert a file with an X3D-like content into Javascript in realtime. This means when our "3D-graphs" (actually an .htm-file with a lot of X3D-code in it) are being viewed, in the background the X3DOM cloud-service is called to translate the code upfront.

7.2 Output-files being translated by X3DOM.org in real time

Looking at the output-file of the AnRep3D-generator one will see something like this:

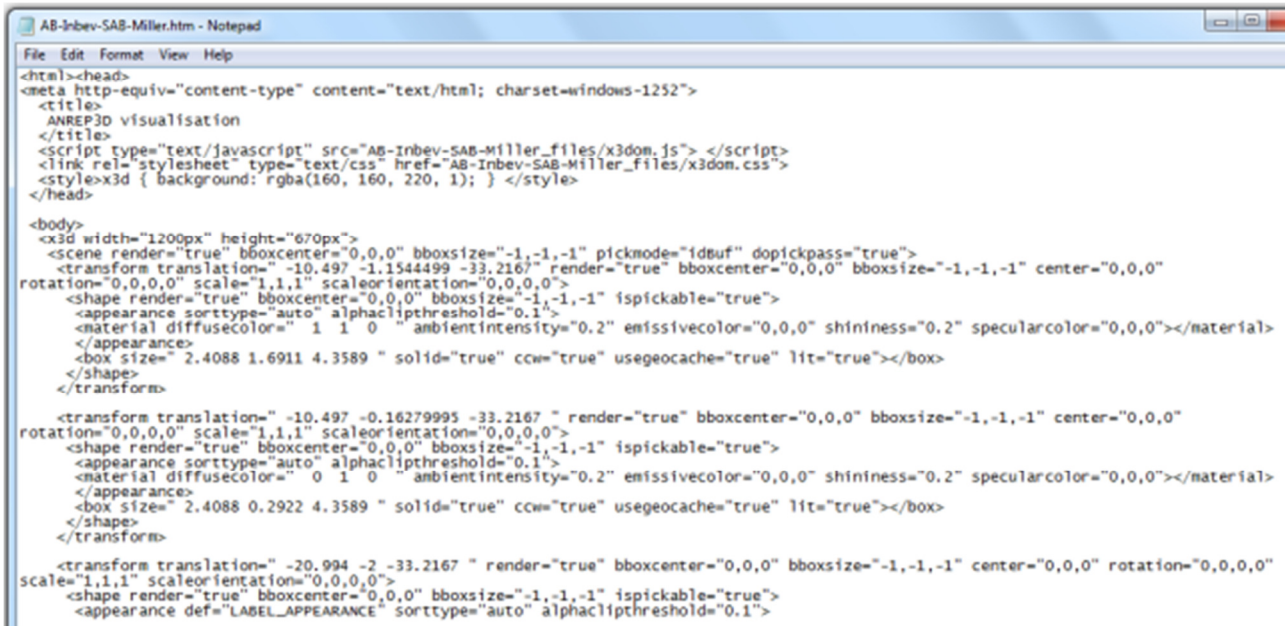



```
<html>
 <head>
  <title>
   ANREP3D visualisation
  </title>
  <script type='text/javascript' src='http://www.x3dom.org/download/x3dom.js'> </script>
  <link rel='stylesheet' type='text/css' href='http://www.x3dom.org/download/x3dom.css'></link>
  <style>x3d { background: rgba(160, 160, 220, 1); } </style>
 </head>

 <body>
  <x3d width='1200px' height='670px'>
   <scene>
    <transform translation= ' -10.497 -1.1544499 -33.2167' >
     <shape>
      <appearance>
      <material diffuseColor='  1  1  0  '></material>
      </appearance>
      <box size=' 2.4088 1.6911 4.3589 '></box>
     </shape>
    </transform>

    <transform translation=' -10.497 -0.16279995 -33.2167 '>
     <shape>
      <appearance>
      <material diffuseColor='  0  1  0  '></material>
      </appearance>
      <box size=' 2.4088 0.2922 4.3589 '></box>
     </shape>
    </transform>

    <transform translation=' -20.994 -2 -33.2167 '>
     <Shape>
      <Appearance DEF='LABEL_APPEARANCE'>
       <Material diffuseColor=' 0.9  0.9  0.9  '>
       </Material>
      </Appearance>
```

The second part within the blue box is the actual graph, more or less in X3D. With some alterations it would be possible to generate pure X3D, to be read by a viewer. The first part however, surrounded by a red line, calls for a translation. As the file itself is already in the html-format, the translated result can be understood by the browser.

This worked so well that we never thought about it, until last May (2018). The X3DOM cloud-service was down during a couple of days for the first time ever and as a result the AnRep3D-graphs didn't work! We were shocked, especially because we never realised X3DOM was a cloud-service translating the AnRep3D code in real-time! After the weekend https://x3dom.org was up and running again and has been stable again.

Fortunately the *converted code can be saved locally to become independent from the cloud-service*, but to be honest: we never did until X3DOM went out of service for a couple of days. Below an example is shown of the original code (above) after calling the X3DOM-service.



Apart from the translated html-file, a folder will be present holding two files: ***x3dom.css*** and ***x3dom.js*** The former is a cascading style sheet and the latter holds JavaScript-code. HTML-file and folder (with two additional files) together provide the local graph. Having this, there is no need for a connection to the Internet and X3DOM.org any more. The only difference I noticed is the translated and stored graph being larger: only a blue background is visible, but scrolling down the graph is still there.

7.3 A short note on JavaScript and Java.

Despite "Java" being in the name, JavaScript has nothing to do with the Java programming language. It's an example of "mimicry" used as a marketing strategy and copied from nature, where animals or plants will mimic properties from another being.

The AnRep3D-generator was written in Java, but any other language would have done (e.g. C++). All one needs is the complied programme – never mind how it was created. The only issue is, the Java-package should be available on your device to be able to run it. Java is available for free at a separate Oracle-website: https://java.com/en/download/ Languages used by Microsoft are preinstalled in Windows, but Java is not.

7.4 WebGL

To make things even more complicated, there is another layer in place. HTML5 can present graphics, but to be able to interpret the code and show the result at the screen, somewhere deep in in the system WebGL is active. The name is short for Web Graphics Library and with the help of this JavaScript API (Application programming interface – again JavaScript!) web browsers won't need plug-ins to be able to show graphics presented by HTML5.

8 The AnRep3D business model

8.1 Companies

The AnRep3D business model has been very simple until now. Companies can purchase a licence, allowing them to have up to five concurrent users working with the generator within one Legal Entity – not covering for subsidiaries or holdings. The payment is a one-off.

We are not naive and know that five concurrent users could even cover companies with hundreds of thousands of employees. Nobody will be creating AnRep3D-graphs fulltime, so if one person is using the generator for a couple of seconds (it's really that quick) another one can take turns. That's why the Legal Entity limit was put in place.

Apart from the one-off payment, the licence and is perpetual for the purchased version (except when the terms and conditions are violated, then the licence expires immediately). Of course the normal model would be to have a recurring annual fee of 10 – 20 % of the initial amount. However, we know we are a niche-player and we prefer as many companies to buy a licence and even better: actually work with the AnRep3D-generator! The one-off payment of a couple of hundreds of euros will just help us to earn back all the time invested. An annual fee would cause us trouble because of the administrative burden, rather than being a benefit. In the end we would rather change the world by sharing AnRep3D than get rich by it.

If ideas for improvement come up which are broadly shared, then new versions have to be developed. Existing customers will get a discount, but then a new business model could be in place. Until now we are happy with the current version of the generator and the graphs and would like them to be used all over the world!

8.2 Consumers

Coming from a B2B background, it is very hard to sell something to consumers. B2B reflexes are different and the result would either be to spend too much time on an individual customer, getting in conflict with unknown regulators or disappointing customers. In short: we don't sell to consumers.

Yet, we realised that in the end individuals, rather than companies, will work with AnRep3D. We already offered a free demo with full functionality from the start. The only limitation is that this demo-generator only shows one "building" (one company, one year) as it will only read one data-line. The question is what will happen if somebody is really interested in AnRep3D, used the demo and wants to have the real generator.

Recently we came up with a solution. Instead of paying money such a person can send in a creative 3D-graph, created with the help of the demo-generator. Although the specific graph will remain the intellectual property of the creator, we obtain the right to show it for marketing purposes. The reward will be a personal licence.

8.3 Investors, resellers, business angels and take-overs

Let's face it: AnRep3D is a small niche-player and a family-held company. If we want to be successful, something bigger should back us. Several options can be thought of, but our limiting factor is time. We work as consultants in finance, marketing and technology and that's how we make a living in the

first place. It would be nice if AnRep3D would be as big as some other companies in Data Mining or Visualisation, but right now we are not. Even a simple investor won't help, unless we recruit people to work on marketing, sales, development, user-experience and so on. HR, legal and finance would come in place as well. This means a lot of money to be burnt, but at the moment we don't even know where the 3D-graphs will be in a couple of years. A business angel would be better as then a network and knowledge would come in together with the money.

A simpler model would be to have added value resellers, making money not primarily by selling licences (although we are willing to share the revenue on a 50/50 basis), but by offering consultancy and education. This model could be very successful and we would like to be in touch with candidates who want to take a chance.

Finally, a take-over would also be possible. However, we are not in for the money (although a million euro would be nice). As we want to change the world, we have to be sure AnRep3D is going out into the world, rather than be killed and buried. Then there is a complication as our family has another product, based on the same 3D-technology: VRBI. A take-over would probably mean that AnRep3D and VRBI have to be sold in a package-deal.