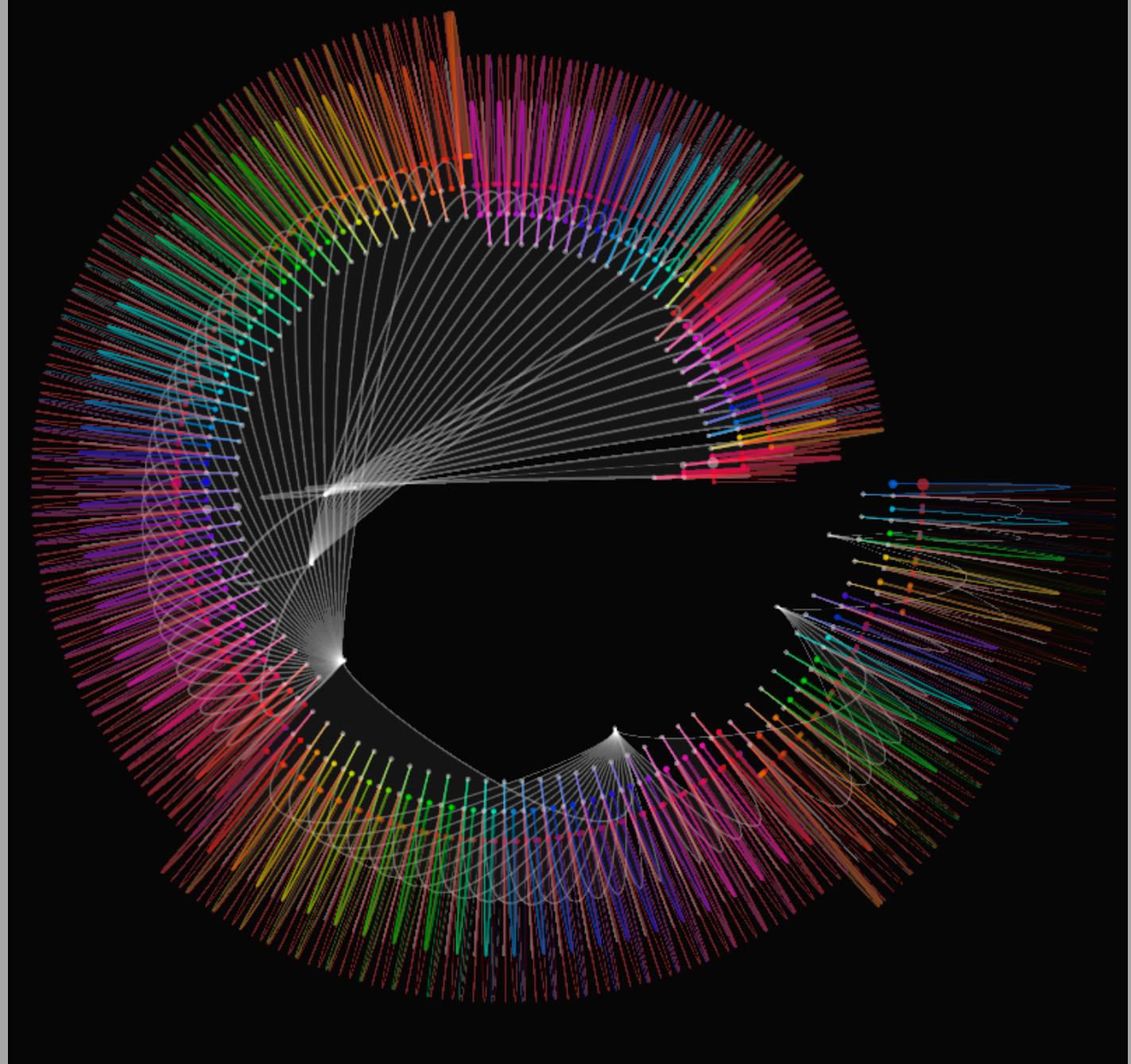


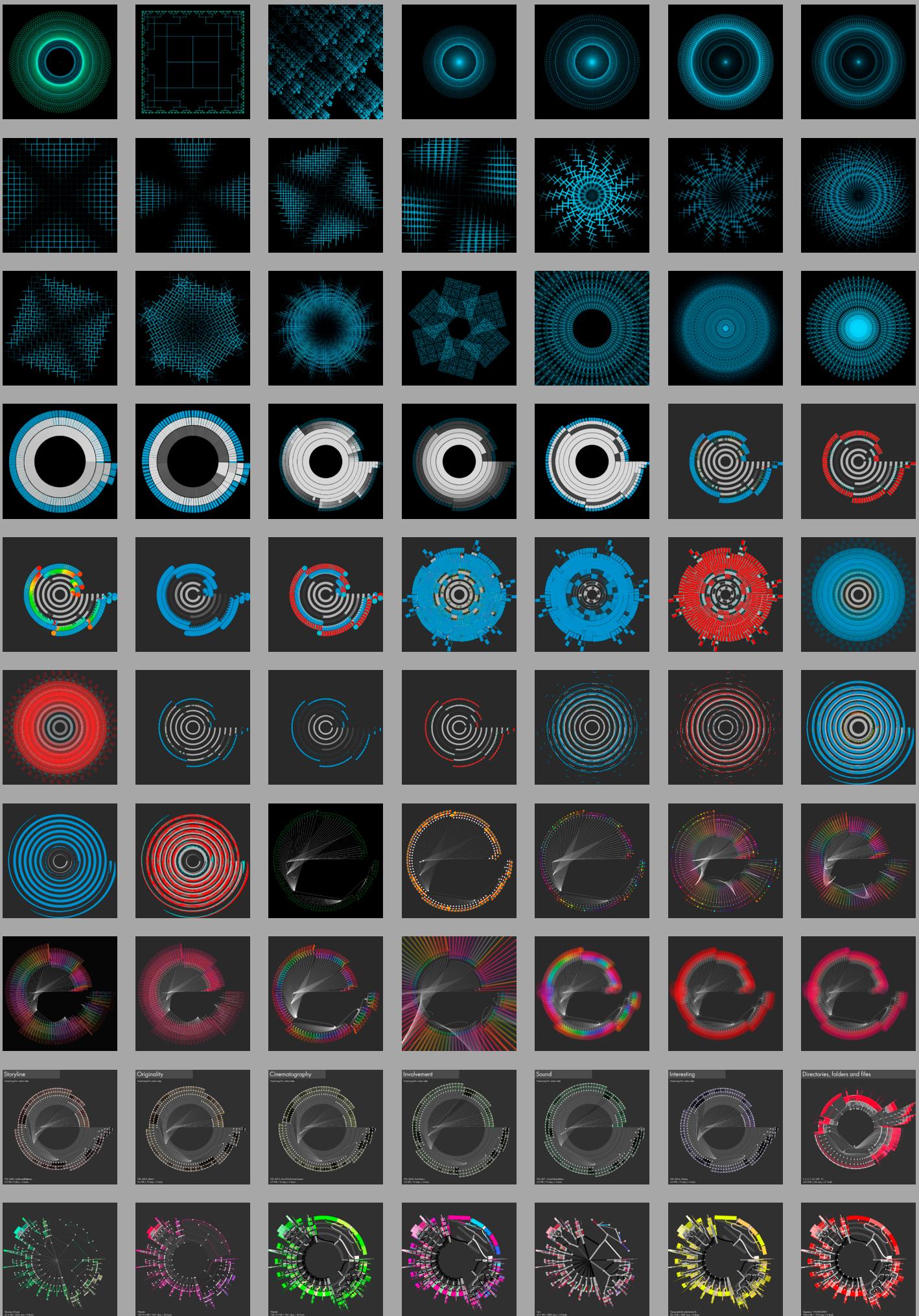
Hierarchy



I never like hierarchy, but in this chapter I'll be glad to make an exception (free interpretation of Groucho Marx's quote). This chapter is about hierarchical structured information using the tree structure as a graphical model. The classical representation resembles an upside down tree-structure. Directories and files are good examples of tree structures. Each directory can contain several different directories. And these directories may again contain directories or files. This hierarchy could go on endlessly. In addition, these hierarchical diagrams are designed to simultaneously show how all data is linked. But they are never clear about the actual information in a document.

One spring evening Faulkner invited a woman with him to see a bride in her wedding dress. Driving over backroads, Faulkner finally turned off into a meadow, where he doused the headlights and drove cautiously forward in the darkness. At length he stopped the car and announced that the bride was in front of them. He switched on the lights, and there in their beam stood an apple tree in full blossom.

William Faulkner, 1897–1962, US novelist.



MyCodeHistory: 22 June 2015

From this chapter on I did not change much of the original programs that were presented in the 'Generative Design' book. When you are interested in learning to program you can find it out yourself. Which is –by the way– much more instructive. In this chapter recursion is presented. Recursion is something new for me. I never like to use it much because I thought that the images it delivers are too symmetrical. And generally I do not like symmetrical images. Symmetry makes everything very balanced. And most of the time that is not what I am looking for. But let me give it a try. I flipped the first image which sketch M_5_1_01.pde creates. It is almost a symmetrical image. At least almost. Because to make it totally symmetrical it should also be vertically mirrored. So that is what I did. And because I thought... well if it can be vertically mirrored than it can also horizontally mirrored.

In fact the images tend to look different from the previous sketch but it is not. Instead of mirroring the recursion four times (which is in fact a rotation of 90 degrees) this image is rotated 128 times with an angle of 2.8125 degrees. It is an interesting image but you do not need recursion for that. In fact using the 1–9 keys you can increase or decrease the amount of the recursion level. And that delivers an interesting symmetrical image build by a pattern.

I have decided to take a step back. How does recursion actually work? And how can I modify it? I replaced the arcs with straight lines. That gives me a square divided into four equal parts. Visually this makes a great difference. And just by coincidence I found out that you can also adjust the branches without changing the T-shape. I am now trying to make more complex images. Just keeping the recursion in the back of my mind. I have left out the dot from the previous images because I don't know what it is good for. Just repeated the drawT line a few times with some divisions. It gives a more chaotic image in the end. But there are all kinds of interesting things going on. And that is what I prefer.

The next sketch is basically using a triangle on its point. The recursion pattern is just the same as in every previous sketch. Again I rotated the recursion 128 times with an angle of 2.8125 degrees. It gives me a kind of 17th Century collar-like image. And than I started over again with a large cross (or plus-sign). That worked out fine. But I found another way to modify that cross (or plus) which on its turn leads to very different images. In the end it gave me again a 17th Century collar-like image. But this time it has a much more interesting structure.

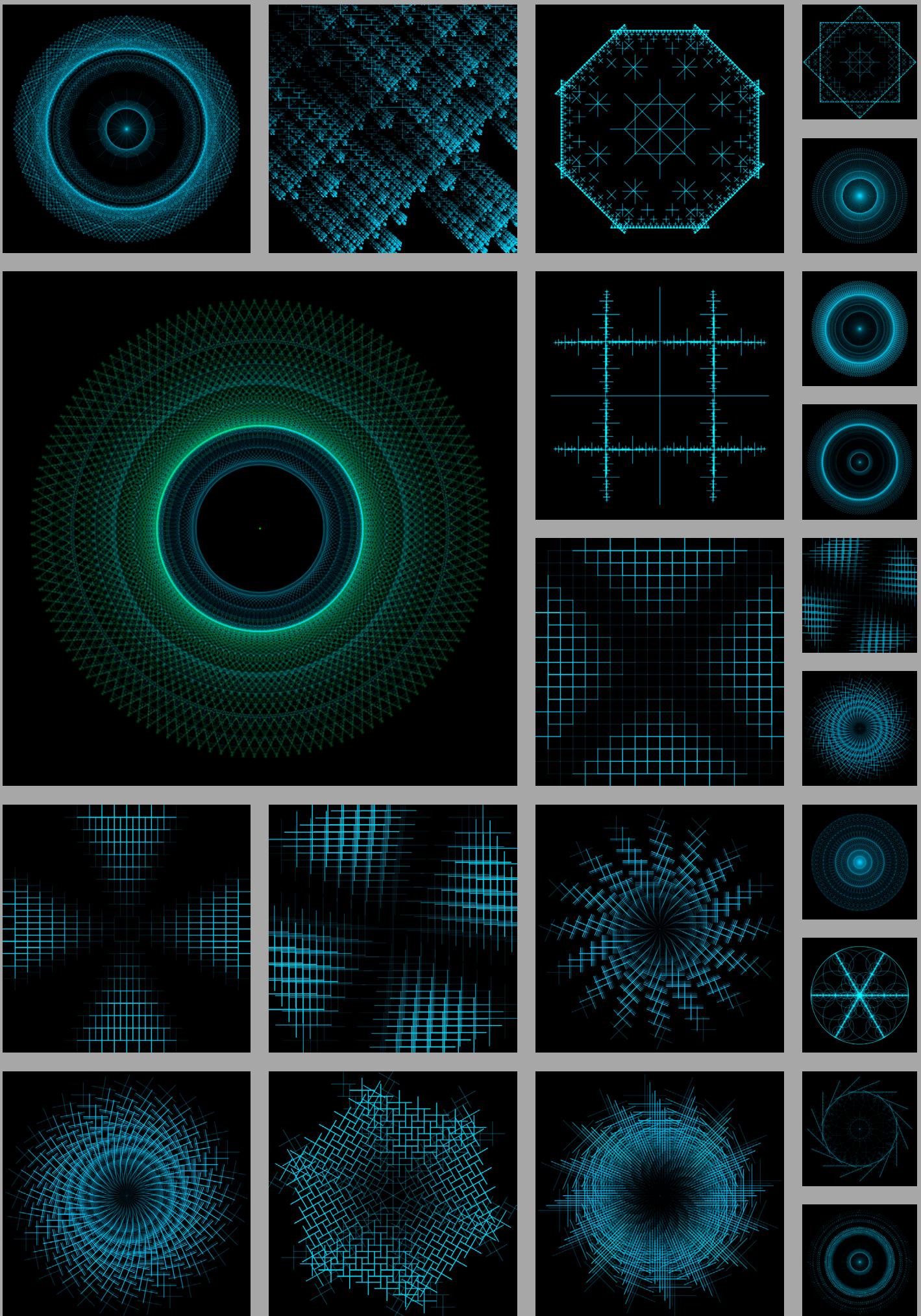
Using the same plus-sign as in the previous sketch I switched off the minimise function. So every recursion cycle draws a plus-sign without changing its size. And because I used an alpha channel the layering makes the signs darker and brighter. This has lead to another range of variations with the plus-sign used as a pavement pattern.

Yet another way of using recursion. A circle. And within that circle are two other circles. And within those two circles are... and so on and so forth. When you rotate the recursion you get very complex patterns. They look a bit too classic to me. I moved the circles out of the large circle. I have five circles arranged in a larger circle. Every circle is accompanied by four smaller circles. And that gives even more different possibilities to get complex patterns. And strange enough the circles generate squares when the recursion is carried through into small details.

Just modifying things a bit can have great consequences for the final image.

An express train to London once made an unscheduled stop at Reading station and Joad, who had just missed his own train, hopped aboard. 'I'm afraid you'll have to get off, sir,' called a poster. 'This train doesn't stop here.' 'In that case, don't worry,' replied Joad. 'I'm not on it.'

Cyril Edwin Mitchinson Joad, 1891–1953, British philosopher, teacher, and radio personality.



MyCodeHistory: 24 June 2015

This part of the tree diagram chapter is completely different than the other ones. It's about reading data from your computer's hard drive. The directories and files are the basis for a tree structure. Now the point is... there are no graphics involved. I only get textual feedback in the console window of the Processing Development Environment. The program displays the entire tree structure in the Processing console with line indentation, which indicates the depth of the directories. So I think its best to write a bit more about what I did instead of showing several tree-structures in ASCII text. I started with looking for a folder which might be interesting to read. I copied our movie reviews to the desktop and specified the path of the default folder to that folder. Then I let the program run. The result is a terrible mess of folders and files. It is an ordered list but it looks very chaotic. But is that because my directory of movie reviews is chaotic?

Looking under the `FileSystemItem` tab of the Processing PDE I found the class `FileSystemItem`. It's a class based on code and ideas of Ben Fry (his book *Visualizing Data* I started reading more than five times but never finished it). I need to get a bit deeper into that before I can say something sensible about it. `FileSystemItem` looks for a file or folder. It finds references to sub-objects that are stored in the array `children`. It also sorts file names alphabetically. It skips '.' and '..' directory entries and symbolic links. It also checks if the child elements exist. Aliases or shortcuts are being skipped. And than a new instance of `FileSystemItem` is generated for all folders and files. But I needed a much more simpler directory. So I made a fake directory. The root folder is called `FakeFolder`. And it contains three `FakeFolders` to begin with.

I noticed that the output in the console gives me two representations of the same `FakeFolder`. One is called `printDepthFirst` and the other one is `printBreadthFirst`. I was not familiar with these two sorting processes so I looked it up. Depth-first search starts at the root of the directory and then it searches as far as possible along each branch before going back. Breadth-first also starts its search at the root directory and then it tries to find the neighbour nodes first before going into the next level of neighbours. You can see the differences in the image on the right page.

That is (in short) what there is to tell about this program. There are some other issues which I did not mention. You can find more information about the program in the Generative Design book (M.5.2 Reading data from the hard drive).

Logic is one thing, the human animal another. You can quite easily propose a logical solution to something and at the same time hope in your heart of hearts it won't work out.

Luigi Pirandello, 1867–1936, Italian dramatist, novelist, poet and story writer.

```

/Users/henklamers/FakeFolder
printDepthFirst
0 -1<-->0 (0) FakeFolder
1 files

printBreadthFirst
0 -1<-->0 (0) FakeFolder
1 files

/Users/henklamers/Desktop/FakeFolder
printDepthFirst
0 -1<-->0 (0) FakeFolder
1 0<-->1 (1) FakeFolder_01
2 1<-->2 (2) FakeFolder_01_01
3 2<-->3 (3) FakeFile_01_01_01.txt
4 1<-->4 (2) FakeFolder_01_02
5 4<-->5 (3) FakeFile_01_02_01.txt
6 1<-->6 (2) FakeFolder_01_03
7 6<-->7 (3) FakeFile_01_03_01.txt
8 0<-->8 (1) FakeFolder_02
9 8<-->9 (2) FakeFolder_02_01
10 9<-->10 (3) FakeFile_02_01_01.txt
11 8<-->11 (2) FakeFolder_02_02
12 11<-->12 (3) FakeFile_02_02_01.txt
13 8<-->13 (2) FakeFolder_02_03
14 13<-->14 (3) FakeFile_02_03_01.txt
15 0<-->15 (1) FakeFolder_03
16 15<-->16 (2) FakeFolder_03_01
17 16<-->17 (3) FakeFile_03_01_01.txt
18 15<-->18 (2) FakeFolder_03_02
19 18<-->19 (3) FakeFile_03_02_01.txt
20 15<-->20 (2) FakeFolder_03_03
21 20<-->21 (3) FakeFile_03_03_01.txt
22 files

printBreadthFirst
0 -1<-->0 (0) FakeFolder
1 0<-->1 (1) FakeFolder_01
2 0<-->2 (1) FakeFolder_02
3 0<-->3 (1) FakeFolder_03
4 1<-->4 (2) FakeFolder_01_01
5 1<-->5 (2) FakeFolder_01_02
6 1<-->6 (2) FakeFolder_01_03
7 2<-->7 (2) FakeFolder_02_01
8 2<-->8 (2) FakeFolder_02_02
9 2<-->9 (2) FakeFolder_02_03
10 3<-->10 (2) FakeFolder_03_01
11 3<-->11 (2) FakeFolder_03_02
12 3<-->12 (2) FakeFolder_03_03
13 4<-->13 (3) FakeFile_01_01_01.txt
14 5<-->14 (3) FakeFile_01_02_01.txt

```

MyCodeHistory: 1 July 2015

Sunburst diagrams. Something I never used in my work. I first copied all comments of the Generative Design book into the program. Usually that works best to get an idea of what the program does. But this time the program is still not very clear to me. Most of the time I find out how everything connects during working and modifying the program. That is for sure not the case with this code. But first I have to solve something else. In the previous session I thought I could use a FakeDirectory with FakeFolders and FakeFiles. But when I read that FakeDirectory in the Sunburst diagram program it was pretty boring. It might have something to do with the fact that all directories, folders and files are having the same directory-structure. So I took the wrong path. You might say 'But why don't you fill the FakeDirectories with random FakeFiles?' Good question... but that is a lot of work. By the time your tree in the console window looks interesting you might be a few days further in time. Maybe I could use my movie rating directory files again. I created them with a very simple Processing program that counts variables together and gives you a number in the end which represents the quality of the movie you have seen. On the right page you find an example of the output. We have seen almost 100 films since the day we started rating movies. And I hope that this directory might be a good candidate to use as a sunburst diagram.

It turned out that the movie directory seemed not as good as I expected. And there were a lot of bezier errors in it. At least I thought it were bezier errors. But when I looked for a bezier function in the program I couldn't find one. I have to find out why these rendering errors appear. But first I must reorganise the film review directory. I think we get a directory of 10 levels deep. Our rating is ranked from 0–10. I think a 10-points rated movie can go in the inside ring. All the other points will be outside rings.

Might as well turn out that I just need to reverse that order in a later stage. This means that the top folder should be filled with movies that are rated 10. I just call the folder 10. In that folder should be a 09 folder. In that folder should be a 08 folder and so continuing until we arrive at a folder which is named folder 00. Although I am almost sure that we did not see a movie which has zero or no points at all. Looking at the image the program creates, there is something wrong. The inner circle is white. But it should be filled with five movies which scored 10 points. Strange enough they are located one ring deeper. Oh the inner ring is shaped by the folder called FilmReviewData_GDV_02.

I changed the path to FilmReviewData_GDV_02_10 but the result stays the same. What happens when I copy the folder named 10 to the desktop and point the Sunburst diagram to that? That seems to work. So I have to change the name of the top folder from FilmReviewData to 10. The only thing I have to find out is why the diagram has strange unwanted lines (or rendering errors) in it. I have no idea by which code they are generated. The only thing I can think of is that some Java files in the directory have influence on the shape. Or maybe these are rounding errors? After cleaning up the directories and files they are almost gone. Otherwise I have to remove them in Photoshop.

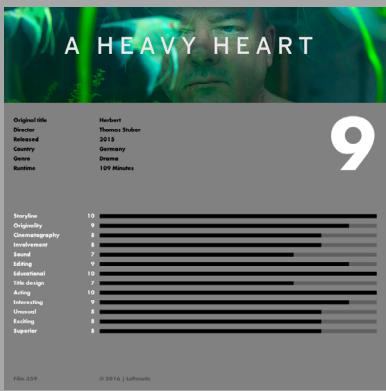
Changed the color range. Used all colours from the color spectrum for the files. Used grey for folders. I halved the thickness of the arcs for folders. And I used radius in a linear way. All circles are now of the same thickness. So the inner red ring has 5 films with 10 points. Than we have 7 films with 9 points and so further and so on.

In this last version I will try to look for other changes. And I must admit it is very difficult to make any change because the program is very tight written. If you try to change something

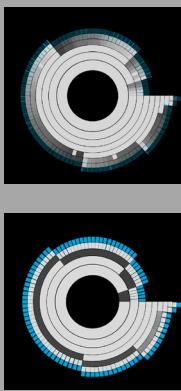
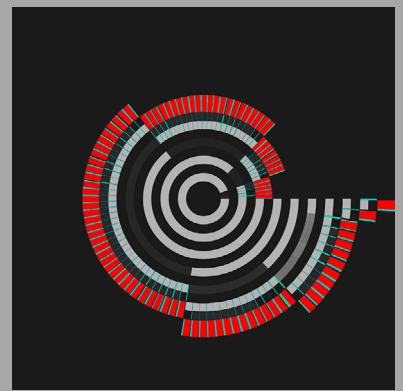
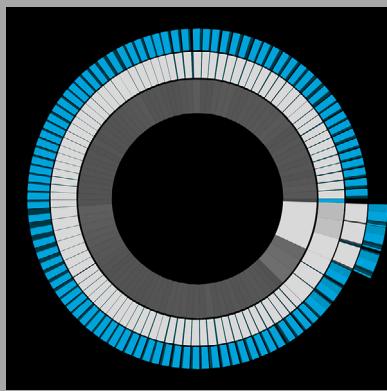
it has an effect on everything. Main thing is that any modification of the diagram is closely connected to the data. When you see it in that light the version with the circles is not very functional.

From here on I knew that the variations I created did not make very much sense but it is interesting to see how far you can deconstruct the image. Of course this hasn't anything to do with visualising data. It has everything to do with deconstructing or repeating data. Made one variation with circular movements. Used the alpha channels for overlays. But again this is a useless data visualisation. They are not readable and do not represent the document structure which I have made with the movie review directories and files.

After working with the sunburst diagram I think the problem is that anyone can do this. Just choose a directory and let the program do its work. Another point is that the program is very hard to modify to something you would really like to have. As I said before the programming is very tight. Everything is strongly connected with the data. And that is a good thing. But it gives you less room to design something special. And if you do change something things will get very confusing. For instance when you change strokes for fills the diagram is completely changed to something which looks interesting. But looking it from the data's point of view it is a useless change. Another thing is that this diagram gives you an overview of the directories, folders and files in a root directory. But it is difficult to see which file is where in the diagram. Maybe my reaction is a bit too premature. Let's see how the next program will evolve.

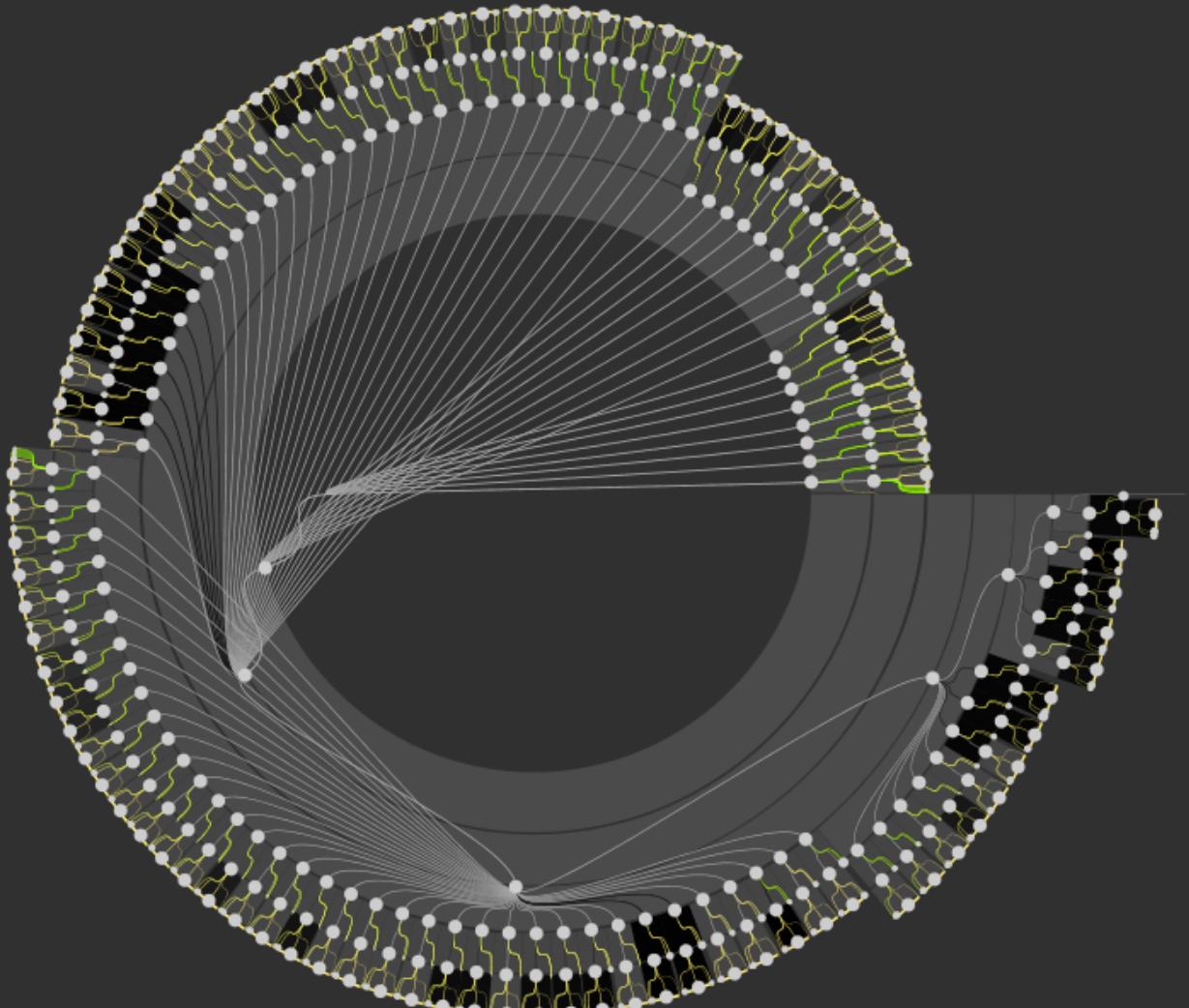


9

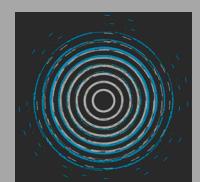
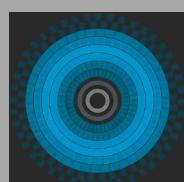
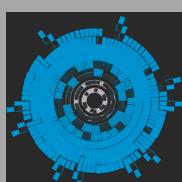
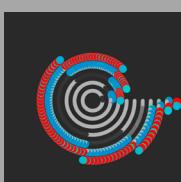
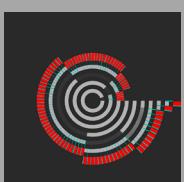


Cinematography

Visualizing film review data



FRD_0073_ManWithAMovieCamera
2.9 MB | 14 days | 4 levels



MyCodeHistory: 5 July 2015

The second part of the sunburst diagrams is a modification of the two classes FileSystemItem and SunburstItem. This makes it possible to create a tree-like image based on the sunburst diagram of the previous program. Again I copied all the comments of the Generative Design book version into the program itself. Changed the background color to black and did some minor changes. Not that I could change much because that FileSystemItem class of Ben Fry looks pretty daunting (or let me call it challenging). I've used the same directory structure as in the previous program. The movie rating directory that is.

I forgot to give the dots a color. In fact they have a color but you won't see them because the dots are the same color as the background. The next thing I am going to do is to update our movie directory with three new films. And I will also change the order of the directory. In the earlier version the top-level of the directory was filled with films who received 10 points. I will reverse that. So the top-level (the inner part of the circle) will be filled with films who have scored zero points and the lower level (the outer part of the circle) will be filled with films which scored 10 points. It makes a minor difference in the shape of the diagram if you compare it with the M.5.3 tree diagrams chapter in the Generative Design book. Line quality is terrible. Have to find another way to improve that.

Spend a lot of time to check how the colours are set. It is still not 100% clear to me how that exactly works. I think I could make a special chapter about that issue. But I will not. I think also Ben Fry's FileSystemItem class needs some more attention from me. Finally I did manage to improve the line quality.

My question is now... why are the outside dots always filled with the same color? Another strange thing are the fileArcScale and folderArcScale variables. Are they in use? I have commented them out and the program behaves normal. I don't see any change in the diagrams. And the program doesn't make use of arcs. I kept the brightness of lines and folders on 100. But when I put dotBrightness to zero in the initialisation it still displays the same. It doesn't make any difference. Changing dotSize during the initialising phase does make a difference. Setting hueStart to zero and hueEnd to 360 makes everything red except for the lines and the directories and subdirectories. Setting hueStart and hueEnd both to 360 also makes everything red. Setting hueStart to 359 and hueEnd to 360 gives me the total range of hue's.

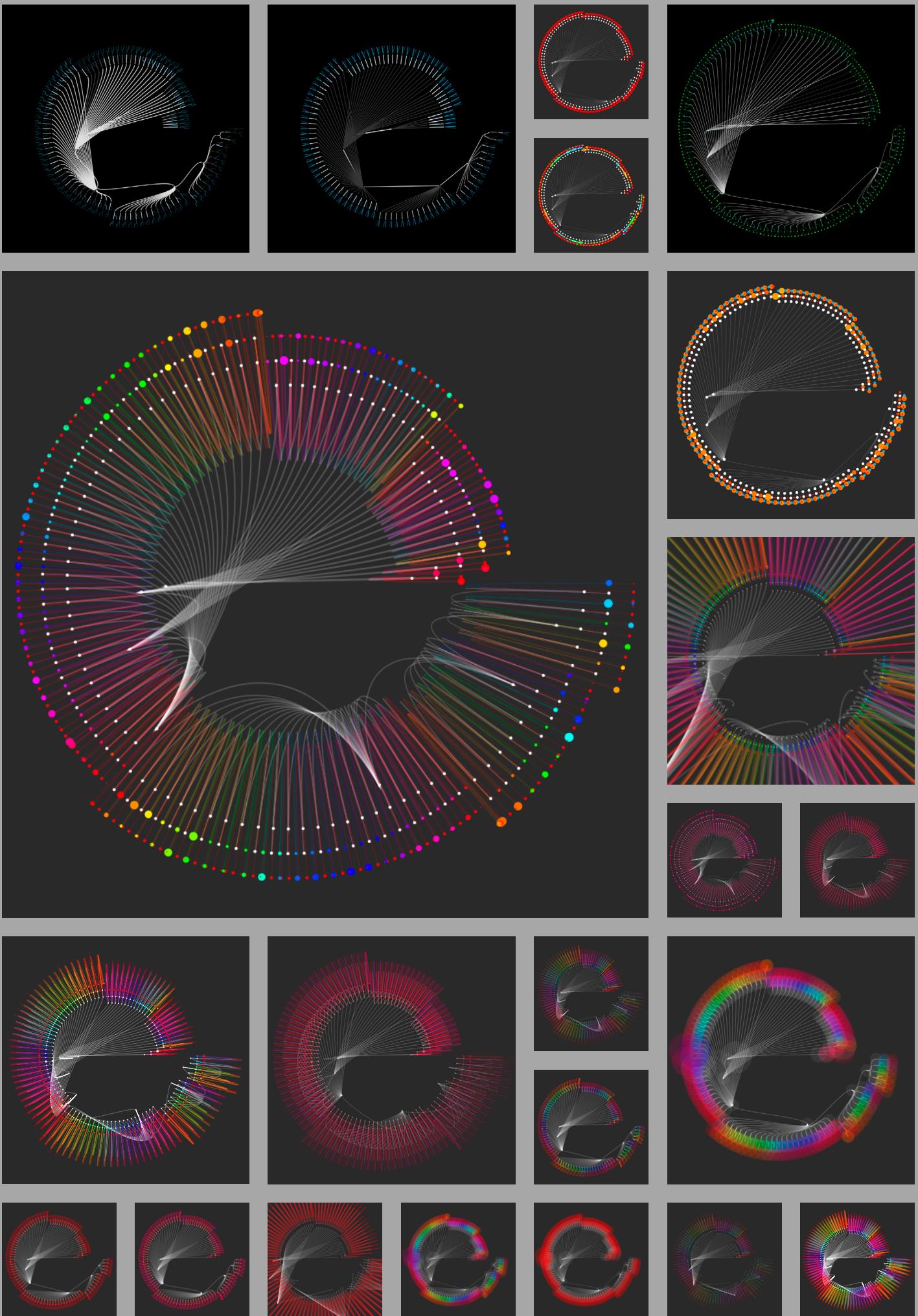
I have replaced calcEqualAreaRadius (which should return radii to have equal areas in each depth) for calcAreaRadius (which returns radii in a linear way). It is not clear to me why calcEqualAreaRadius does not return equal area radii while calcAreaRadius does. This gives more room to see the individual dots. Also increased the size of the diagram. The bezier version of this diagram will be the first version of the diagram which may look nice but is totally not functional.

I adjusted the line thickness and the begin and end of the lines. Modified the transparency a bit. Small adjustments on the size of the diagram to let it better fit into the display window. Pulled everything outside the border of the diagram. Again not very functional because you cannot trace the links between folders files and directories. And that is where these diagrams are designed for. I ended with the default settings of lines and some extravagant adjustment of dots and transparency. However, you can still trace the relations between directories, folders and files.

It is easier to make beautiful not functional images than functional ones.

Design must be functional, and functionality must be translated into visual aesthetics without any reliance on gimmicks that have to be explained.

Ferdinand Porsche, 1875–1951,
Czech Republic automotive engineer.



MyCodeHistory: 11 July 2015

I intend to start again with our movie review data directory for this tool. Therefore, it is necessary to explain the movie directory a bit further. We have started watching movies and reviewing them using a simple program in January 2015. Our movie review data program consists of thirteen components that all represent movie quality on a scale from 0 to 10. The components are: Storyline, Originality, Cinematography, Involvement, Sound, Editing, educational, Title design, Acting, Interesting, Unusual, Exciting and Superior. When we have seen a movie we give points looking at these components. I showed one of the reviews when I was writing about the previous program. But here's another one.



Of all the movies we have seen I want to see all ratings of one component in one sunburst diagram. In my opinion that means I need to modify the directory structure. All movies are now in one directory and they are all arranged on date. But I want all scores of Storylines (from 0–10) in one directory. Roughly, I will have to place 100 movies manually in a new directory called: 'Storyline'. And the same I have to do with the other components. Originality gets a directory with all the scores of all originality components of 100 movies. Cinematography gets a separate directory with all the scores of all cinematography components of 100 movies. That means that I have to put 1300 files manually in separate directories.

That took me two days. It is very boring administrative work and that slows me down and it takes even more time.

Maybe I could have swapped those directories using AppleScript but I did not know how long it was going to take to make a script for it. I don't have much experience with AppleScript. Doing it all manually I was sure it was finished after two days (although I did not know that it would take me so long). Then it's a matter of running the sunburst tool for 13 times. Feeding it with the different directories and make screen dumps. All inner rings represent the 10 scores. The rings to the outside represent the 9, 8, 7, 6, 5, 4, 3, 2 and 1 scores. The outer ring represents the 0 scores. I was surprised because the result is visually as dull as the work of the past two days.

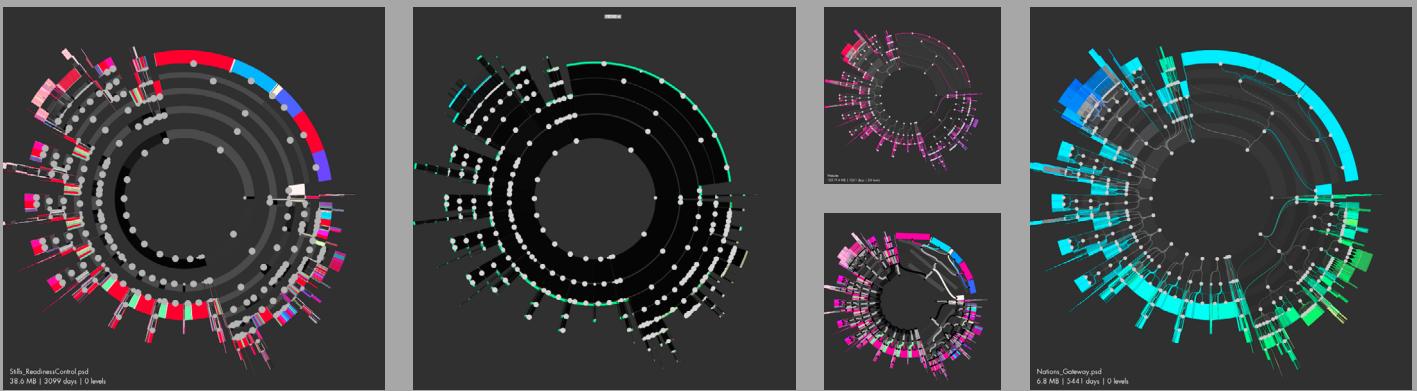
Because the visuals were all rather disappointing, I just looked what happened when I pointed the sunburst tool to the entire file structure of Generative Design Variations directory. That image was much better than the results of the past few days. The same, I tried with my internal accounting (bookkeeping) directory. And with the directory of my loftmatic website. It all looked very interesting. I copied the file structure of the movie review data from my desktop to the Generative Design Variations folder.

The next day I opened the Generative Design Variations file structure in the sunburst tool. I worked on that since October 2013 and it contained at that moment 36260 items. It looked very different compared to the results from the previous day. The results from the day before. In fact that is right because I added the movie review data directory and files the day before. It even looked much better. I worked a few hours with the sunburst tool to create variations using the Generative Design Variations directory, my accounting directory and the loftmatic website directory. That led to better results.

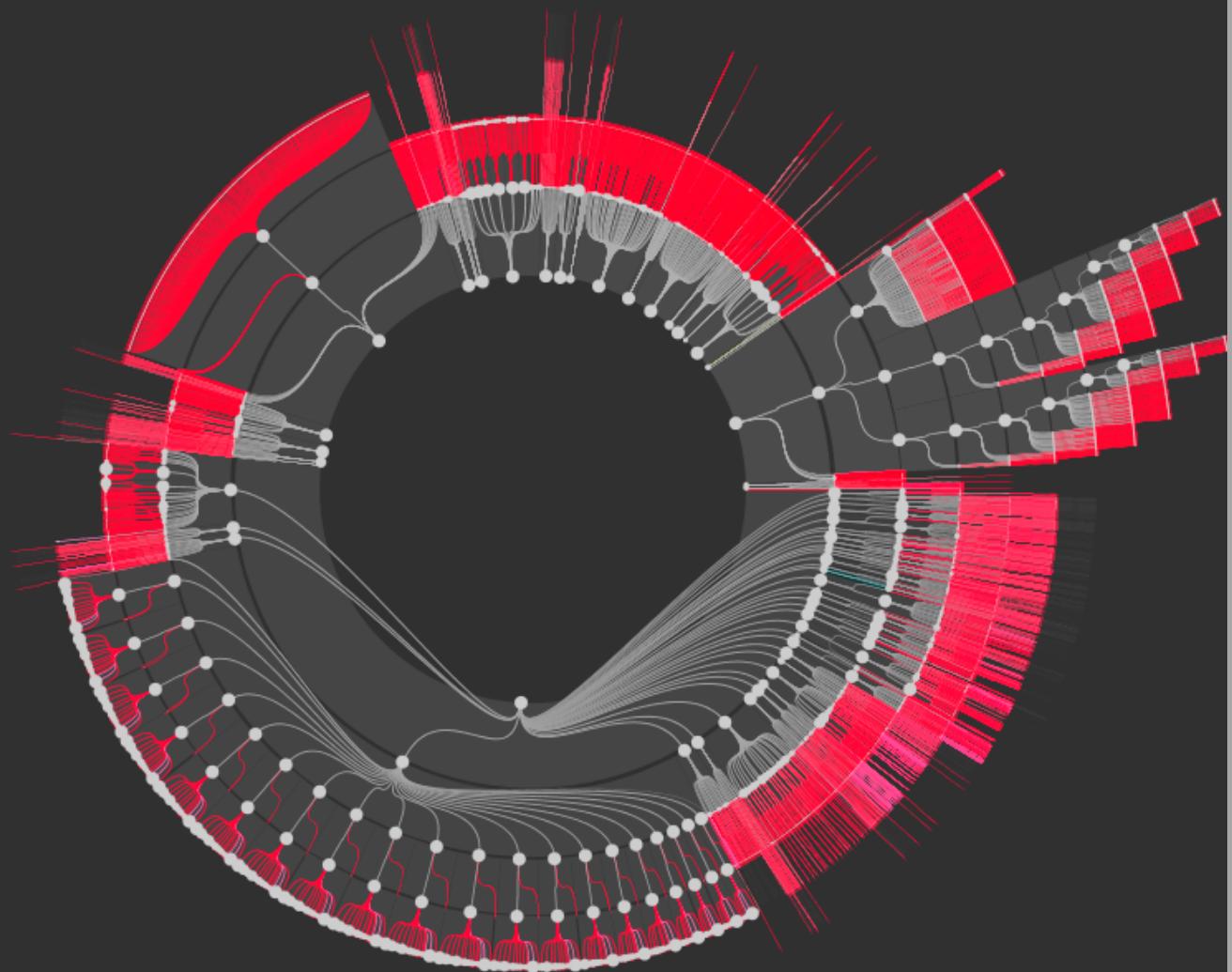
Of course, there's a certain type of person who feels that anything which becomes mainstream has to be rejected immediately. And that's part of the indie-alternative snobbery and hierarchy and elitism.

Alexander Paul Kapranos, 1972—, Scotisch singer and guitarist.

Looking back at this session, I must confess that the sunburst tool only leads to good results if you have a complex file structure. The advantage of the sunburst tool is that anyone can use it. But that is an immediate disadvantage because you get almost always similar results. Less complex directories deliver different results than complex directories with lots of levels, folders and files but the design language is recognisable. And as a designer you don't always want similar results. In fact I rather would never have similar results if I don't need them. So I think that every tool that you use should have more variables to change the visualisation. And that is the nice thing of a programmed tool because you can always change the code to your specific needs.



Directories, folders and files



P_3_2_1_02_GDV_10
24.0 MB | 346 days | 61 levels

