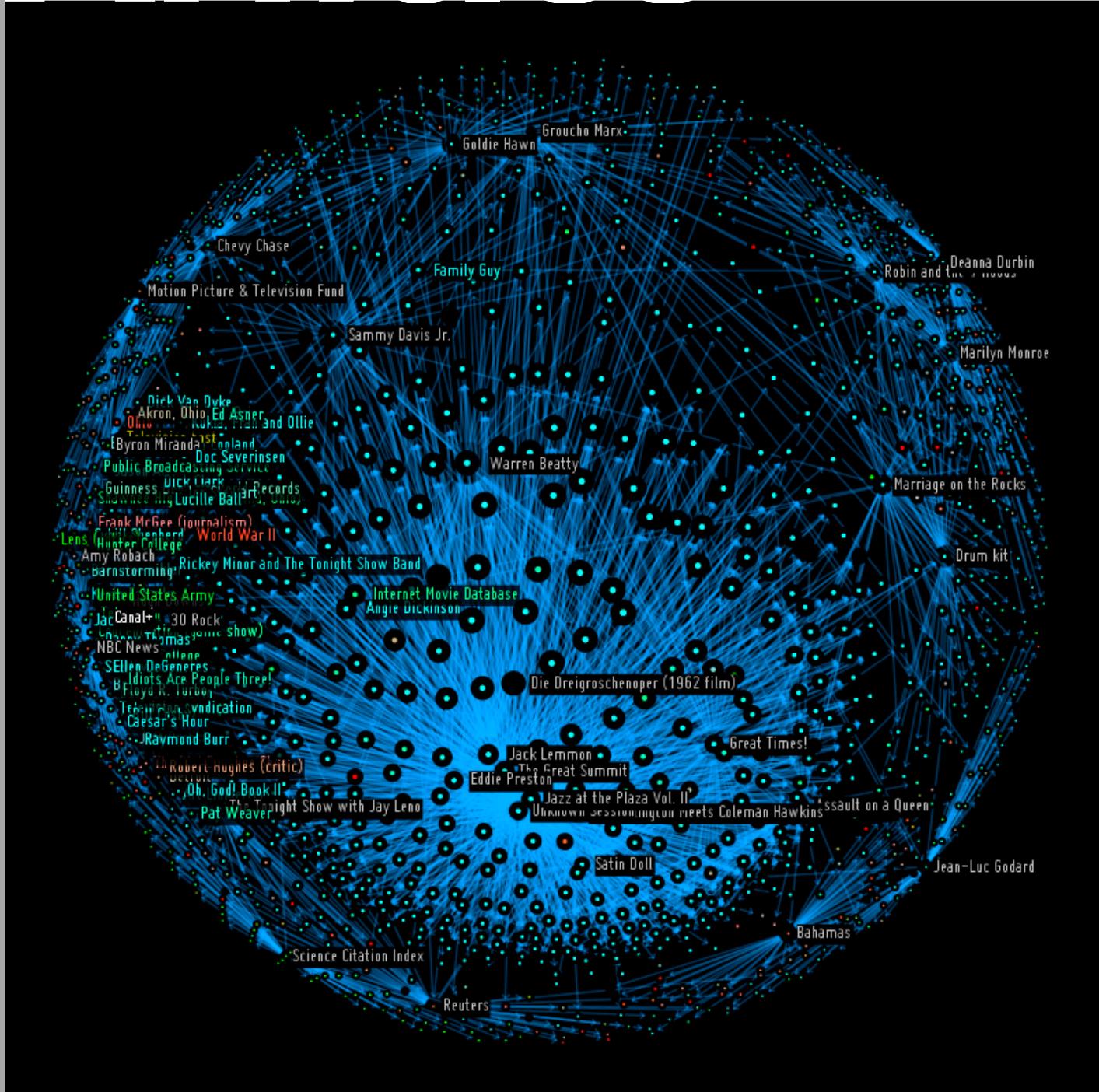


# Articles



I have entered the last chapter of the Generative Design book. M6. Dynamic data structures. A while before I started this chapter I knew that it would be a difficult one. A few months ago I checked all the files from the chapter. Three of seven programs did not work. In all three cases I got a '[Fatal Error]: 1: 1: Premature end of file' warning in my Processing console. I have tried if I could fix the programs but that worked only partially. I got caught up in XML and XML-related problems. Eventually it had given me an overwhelming amount of possibilities that could perhaps solve the problem but I could not track down what the best solution was. I spent a day on the issue. Then I sent an email to the Generative Design colleagues. I have updated them with my current status. And mentioned that it would be very sad if I have to drop out in the last chapter. For now, I have not heard anything from them. If I hear something in the future that's fine. If I hear nothing I'll go searching for other solutions. Which could mean that this chapter evolves differently than described in the Generative Design book. Let's start with it.

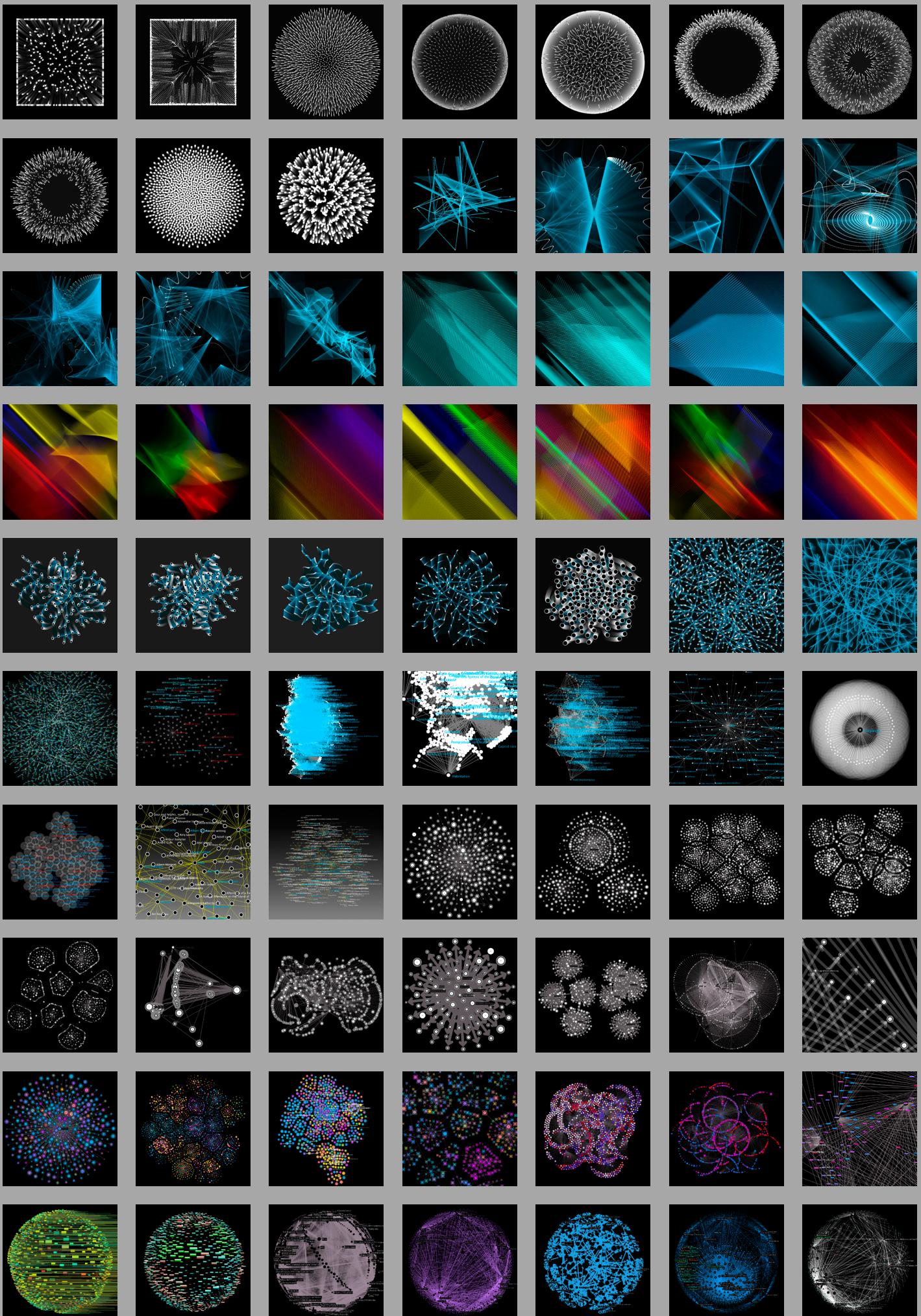
---

One of the things you have to consider when doing a self-study is: 'You have to do everything yourself and (in the worst case) nobody will help you.'

---

To help yourself, you must be yourself. Be the best that you can be. When you make a mistake, learn from it, pick yourself up and move on.

David James Pelzer, 1960–, American author.



MyCodeHistory: 17 July 2015

Due to the unsolved problems that are coming up in this chapter my plan is now just to continue and see where I end up. And to be honest that is not really different from how I continued to work from the beginning of October 2013. I pushed the boundary 100 pixels inwards. And the number of nodes are increased from 200 to 400. It is strange that when everything has stopped moving a pattern is recognised. It's not exactly the same pattern every time but it's pretty similar. I doubled the amount of nodes again from 400 to 800. But the final static pattern remains the same.

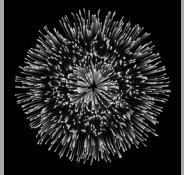
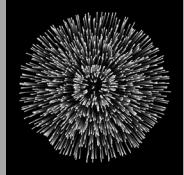
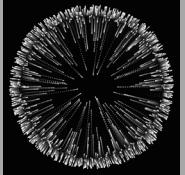
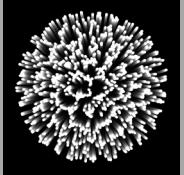
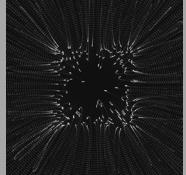
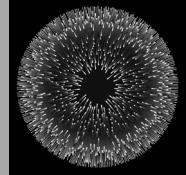
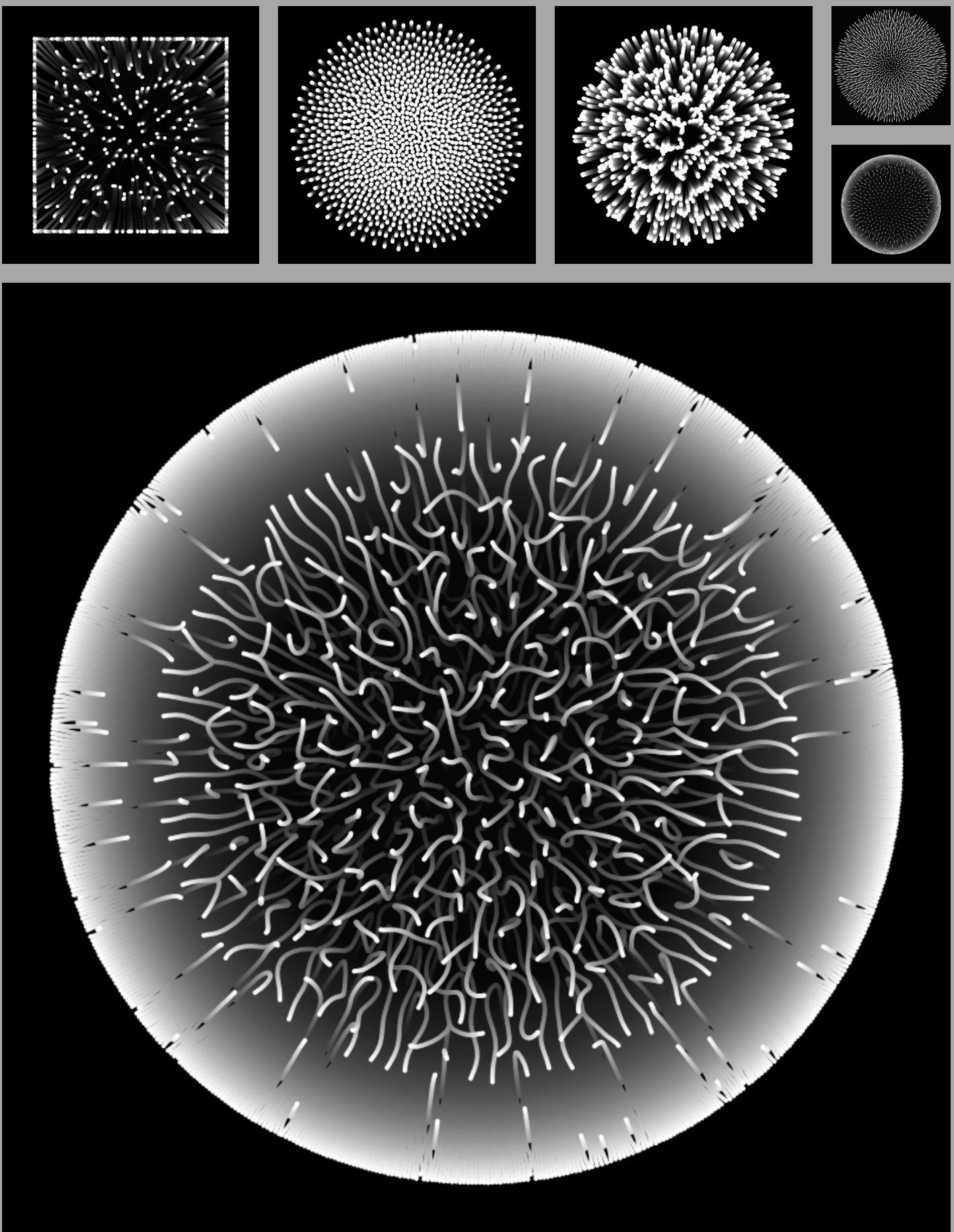
The diameter of the nodes is now reduced to 3 pixels. The number of nodes is doubled again to 1600. The program is not running smooth but that is not important at this stage. For me it's more important that I can understand a little of the mechanisms and makes nice graphics. Animation comes later. I reduced the diameter of the nodes to 1 pixel. 3200 Nodes in use. The 1 pixel diameter I find less successful and the strange thing is that the nodes still create a lot of symmetry. What will happen when I change the overall settings of the variables in the Node class? This certainly offers a lot of interesting almost symmetrical images. The maximum velocity (speed) is reduced from 10 to 2. Brought damping, radius, strength and ramp up and down to judge the effect of these variables.

What mechanism determines the direction of the nodes? It's not entirely clear to me. I have pushed the boundary just outside the display window. Therefore, the nodes return back into the display window. That's actually not what you want because it makes the final picture very busy. The damping and scale variables are apparently very decisive for the result.

The result of this chapter is (or should be) that a number of Wikipedia nodes are sometimes linked. Perhaps it is worth trying to see if the start position of each node has an influence on the result. The starting position is now the center of the display window. But not exactly the center. There is a random factor of -1 and 1 pixels offset. What happens if I continue to pull those positions further apart? In the x direction it makes only a difference at the starting position. The end result remains the same. It also seems that not much is changed when you increase the offset as far as 1600 pixels for the x and y positions. Again it turns out that the starting position is shifted from the center. I ended with a series of images in which I tried to find out where chaos begins and ends within the nodes.

---

It emerged that this program is also a nice tool to create flower-like structures. In the context of this assignment that is not very functional but it delivers me a different result than I had expected at the beginning when I started to work with this program. I see that also as a quality.



MyCodeHistory: 22 July 2015

This program is about springs and nodes. When I began to work with it I never thought this program could be transformed into an interesting drawing tool. I have adapted the two nodes and the spring a bit so they can be used to create compositions. At that moment I was working with a pen and a Wacom tablet. And it appears that drawing with the mouse is easier than using the pen. The pen is too sensitive. The mouse can almost fix its position to the Wacom tablet. And in that manner, you can easily draw straight lines.

I have doubled the length of the spring to be able to work outside the display window. But suddenly, the program has a will of its own. At least a will that I have not consciously programmed. Spontaneously formed sine waves appear. No idea where they come from. Another strange thing is that every image has a cross made by two lines in the middle of the display window. I think the images would be much better without it.

I've copied the line and ellipse drawing functions into the mousePressed block. The initial draw spring and draw nodes functions are commented out. That means that no cross is drawn anymore until I left- or right-click the mouse button. A small note: It seems like this experimentation has nothing to do with this actual program. But the only thing I'm trying to look at are the boundaries of what is possible within the program. The result is always unpredictable. But the resulting images are always related to what has been put into the program. Except when things come into view which I think are not necessary. For example: I think the first node point in every image is not necessary. So I also commented that out.

Right now I am trying to understand where these sine waves come from. I think it has something to do with the damping and stiffness. I have set them both extremely low and that means that the spring jumps very slow and is also difficult to stop. It remains almost endlessly going up and down. So probably damping and stiffness are responsible for those sine waves.

It is time to introduce a refresh function key. I have to restart the program every time without a change made in the program itself. Perhaps it is better for me to focus more on the jumping mechanism instead of making different images.

I have introduced a for loop because I thought it might be interesting to do something with a line pattern. That actually produced better images than I expected. So I think I'm going to expand this pattern. I did not change much. The difference is this time that I made the spring very short. I also have everything repeatedly redrawn.

The last variations were made with more color. I choose red, yellow, blue and green. I now use the spring class to generate patterns. Damping and stiffness are almost off. Which means that they are set extremely low.

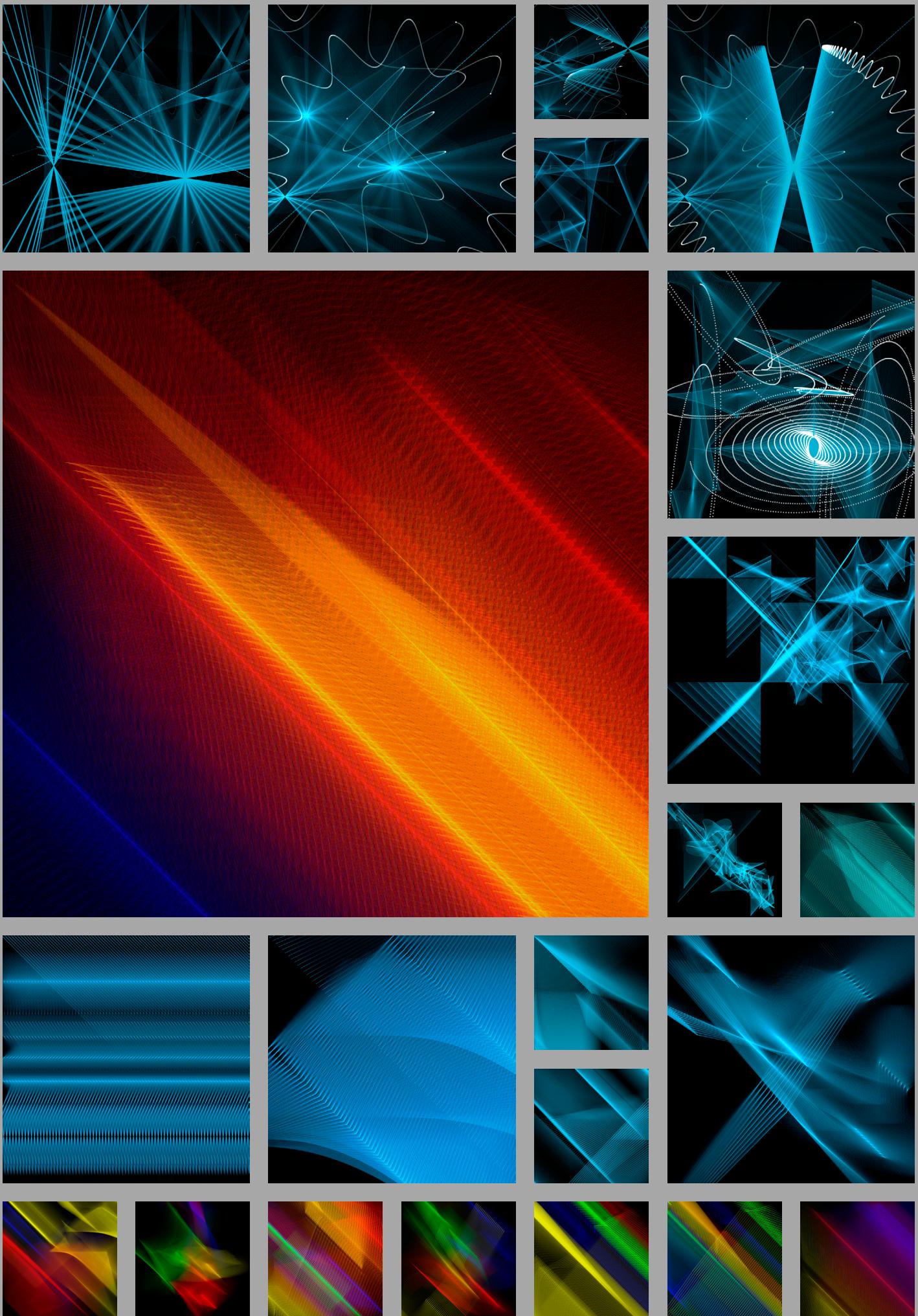
---

This session has created some spring-like pattern images. In short: production-wise it was a good session. But I need to go a bit deeper into the spring class. Maybe I can pick that up in the next program.

---

That is one good thing about this world... there are always sure to be more springs.

Lucy Maud Montgomery, 1874–1942,  
Canadian author.



MyCodeHistory: 30 July 2015

In this program nodes are connected by springs. The node and the spring class are both contained in the Generative Design library. That can be downloaded at [www.generative-gestaltung.de](http://www.generative-gestaltung.de). To begin with, I have not changed much in the program. I only changed the color of the background. The springs and nodes I've kept the same as in the original program. Added traces to all objects.

In the second phase of making variations I decided to reduce the diameter of the nodes. That makes the total image a bit more advanced. But probably less clear. And there is a kind of remaining image (as in after image) that creates the illusion as if it is generating a silk-like material. Not a wrong effect at all.

Maybe it's a bit less sophisticated but I still checked how the image looks when you increase the nodes and leave the line thickness of the springs thin. It all looks a bit more massive but the image is not uninteresting.

I increased the number of nodes to 1000. That makes a chaotic impression. It is also unfortunate that the nodes stay within the display window. As a result there is a build up of nodes at the sides of the display window. They can not move anymore to some place else. That's why I increased the boundaries of the nodes and springs outside the display window.

I've commented out the nodes. In that way I can double the number of springs without the result getting chaotic. At some point I felt the need to reduce the number of springs. I also added some transparency. On the pivot points of the springs artificial nodes are being formed. But they do not have the functionality of the real nodes (which are now turned off). In other images, I turned the nodes on and increased the size slightly.

Time for a new variation. I wanted to introduce color only into the springs. I could do that with an if statement. But in hindsight it was better to remove the lines and to introduce only color in the nodes. After a few variations, I have turned on the springs again. It brings some guidance in the chaos.

I reduced the boundary. Ellipses are replaced by rectangles. The largest distance between two nodes is only 2 pixels. So basically negligible. I have also added another additional color and reduced the diameter of the rectangles. Has a rectangle a diameter?

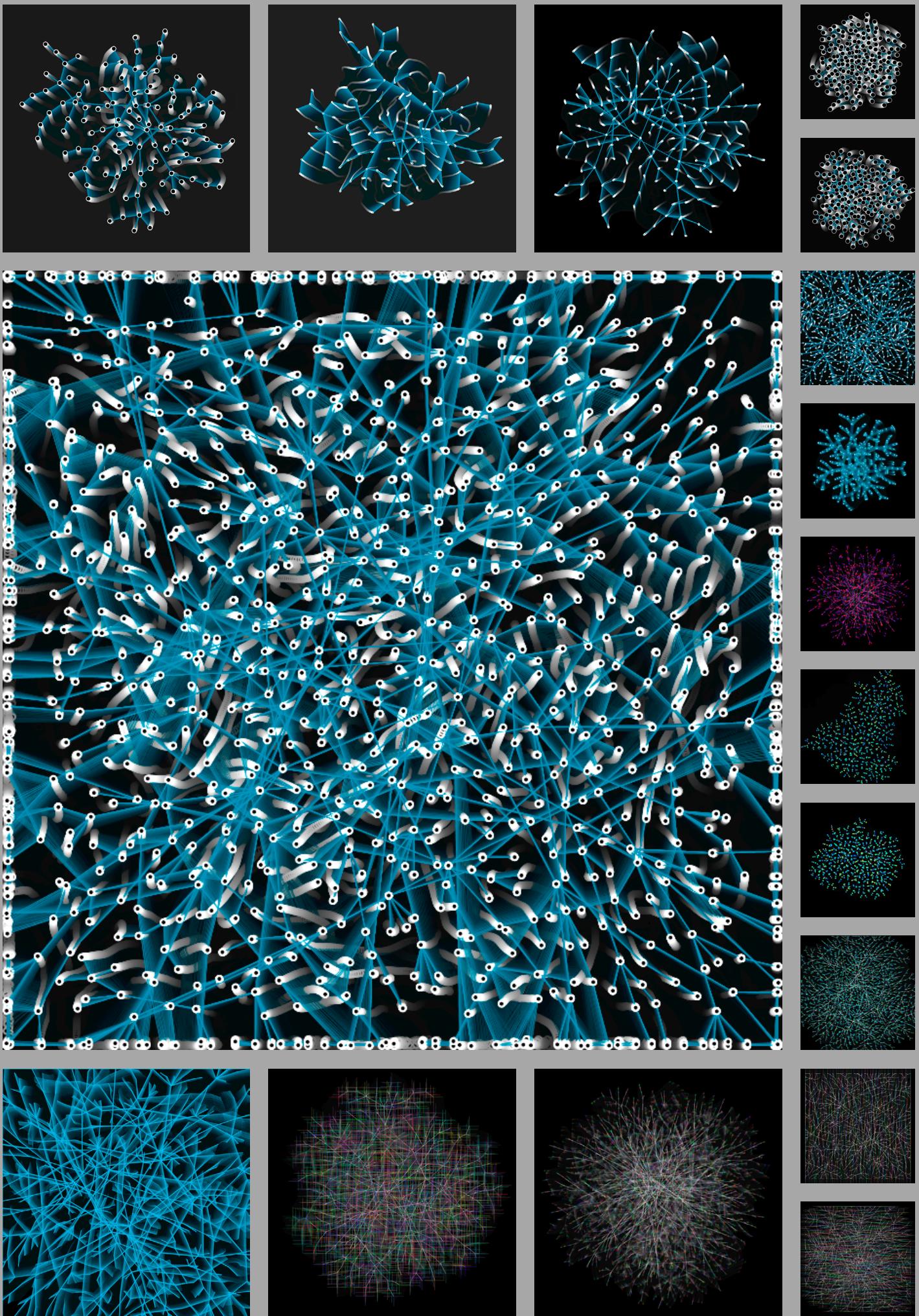
A variation with only lines as nodes. Both horizontally and vertically. I have increased the number of nodes to 3200. Complete nonsense of course but I was curious to see how it would look like. I experimented a bit with the frame rate but actually I did not get any better results. Frame rate is really not relevant if you do not animate these images. Which easily could be done.

What would happen if I make excessively long lines. Strangely enough this results in (some cases) a kind of three-dimensional image. However, these are two-dimensional graphics. I do not work in the z-direction. When the image comes to a standstill it reveals that the display image is not entirely filled. I think that is not an objection. But I made a version where that boundary is raised. I also doubled the alpha channel.

Another variation uses a cross (or plus) at the node positions. Basically, you can put every possible shape in that node position. But in this case I have chosen a cross. This leads at times to very natural images. Especially if the lines are very small. I compare it with overblown dandelions. Why do I always have to make comparisons? Because while doing that it seems you can make something intangible more tangible. I write intentionally 'it seems'. I will come back to that later.

---

This was a very sensible session with very different images when you compare them with the previous sessions. The following two programs are more theoretical. They discuss the use of the wiki API, JSON and XML.



MyCodeHistory: 1 August 2015

This is the first theoretical exercise of two which I need to complete to finish the last exercises of the Generative Design book. Getting data from the Internet is not new to me. I first came into contact with the Internet in 1992. I was working at Océ Research & Development, Venlo, Netherlands. But I will not go into that any further because it is not relevant for this exercise. I usually download a csv or tsv file for data visualizations. But downloading data through an API is new to me. The first thing I tried is to type in a URL from the [Generative Design](#) book.

In fact the browser changes the same url to an https: url. It shows me a page which is the HTML representation of the JSON format. Then I have the question: What is JSON format? Checked that. JSON (JavaScript Object Notation) is a format for data exchange. JSON is based on a subset of the JavaScript programming language. JavaScript? But I need XML. Anyway... JSON apparently has a lot of advantages above XML. It is compact. People and computers can easily understand it. It maps simply to the data structures used by most programming languages (numbers, strings, booleans, nulls and arrays). Until so far, this all sounds very familiar. Another advantage is that almost all programming languages have features or libraries that can read or write JSON structures. The point is now that if you're not careful you're ending up doing other things than completing this session. You'll get on the path that leads to figuring out how JSON, PHP (Hypertext Pre-processor) and AJAX (Asynchronous JavaScript And XML) works. Or shall I try Perl perhaps? So for the time being this seems not a good idea. In addition I stopped following that path and continued to study the [https://www.mediawiki.org/wiki/API:Main\\_page](https://www.mediawiki.org/wiki/API:Main_page). This is the introduction page for the Wikipedia API.

Back to the Generative Design program M\_6\_2\_01. This program was not working at that time. When I run it I got a [Fatal Error]: 1: 1: Premature end of file in my console. I suspect that this error is caused because the start of the URL in the program begins with http://. And I think the right URL should be: https://. It needs the 's'. Https is 'Hyper Text Transfer Protocol' with Secure Sockets Layer (SSL). This is another protocol primarily developed with secure, safe Internet transactions in mind. So when I correct that and let the program run then I get a neat list of 27 numbered links from Wikipedia Superegg page in my Processing console. So I think this problem is now solved. I also tried other pages by requesting Superegg replacing Superegg with Generative%20Design (The % 20 comes from percent-encoding a space). I have done the same with New%20York, Steve%20Jobs, Dada and Kurt%20Schwitters. And these URL's all work fine. All links on those pages are nicely sorted in alphabetical order and displayed in my Processing console. I think that this exercise is over. Furthermore, there are no graphics or variations of the program involved.

en.wikipedia.org/wiki/Superegg

Not logged in Talk Contributions Create account Log in

Article Talk Read Edit View history Search Wikipedia

**Superegg**

From Wikipedia, the free encyclopedia

In geometry, a **superegg** is a solid of revolution obtained by rotating an elongated superellipsoid with exponent greater than 2 around its longest axis. It is a special case of superellipsoids.

An elongated ellipsoid, an elongated superegg can stand upright on a flat surface, or top of another superegg.<sup>[1]</sup> This is due to its curvature being zero at the tips.

The shape was popularized by Danish poet and scientist Piet Hein (1905–1996). Supereggs of various materials, like the one shown, were sold as novelties or "executive toys" in the 1960s. A 1-ton superegg made of steel and aluminum was placed outside Kelvin Hall in Glasgow in 1971, on occasion of a lecture by Piet Hein.<sup>[2]</sup>

**Mathematical description** [edit]

The superegg is a super-ellipsoid whose horizontal cross-sections are circles. It is defined by the implicit equation

$$\left| \frac{x^2 + y^2}{r} \right|^p + \left| \frac{z^p}{h} \right| \leq 1$$


Bass superegg by Piet Hein.

MediaWiki API result - Wikipedia

Wijzig schijf aan geopende pagina, meer gevat / NOS

The MediaWiki API returns JSON. HTML is good for debugging, but is unsuitable for application use. To see the non-HTML representation of the JSON format, set `format=json`.

Specify the `format` parameter to change the output format. To see the non-HTML representation of the JSON format, set `format=json`.

See the complete documentation, or the API help for more information.

```
{
  "continue": {
    "pcontinue": "10405346|International_Standard_Book_Number",
    "continue": "|"
  },
  "query": {
    "pages": [
      {
        "ns": 0,
        "title": "Aluminum"
      },
      {
        "ns": 0,
        "title": "Curvature"
      },
      {
        "ns": 0,
        "title": "Egg of Columbus"
      },
      {
        "ns": 0,
        "title": "Eric W. Weisstein"
      },
      {
        "ns": 0,
        "title": "Executive toy"
      },
      {
        "ns": 0,
        "title": "Exponent"
      },
      {
        "ns": 0,
        "title": "Exponentiation"
      },
      {
        "ns": 0,
        "title": "Geometry"
      },
      {
        "ns": 0,
        "title": "Glasgow"
      },
      {
        "ns": 0,
        "title": "Implicit function"
      }
    ]
  }
}
```

M\_6\_2\_01\_GDV\_01

```

43 XML myXML;
44 XML[] links;
45 String query;
46
47 void setup() {
48   // In the URL for the database query, various parameters are given-e.g., that the result should be in XML
49   // format or that a maximum of 500 links should be returned.
50
51   // It seems that this link gives a [Fatal Error] :1:1: Premature end of file. I think its because the
52   // url uses a http request.
53   // query = "http://en.wikipedia.org/w/api.php?titles=Superegg&format=xml&action=query&prop=links&plimit=500";
54
55   // These urls do work because they use an https request.
56   // query = "https://en.wikipedia.org/w/api.php?titles=Superegg&format=xml&action=query&prop=links&plimit=500";
57   // query = "https://en.wikipedia.org/w/api.php?titles=New%20York&format=xml&action=query&prop=links&plimit=500";
58   // query = "https://en.wikipedia.org/w/api.php?titles=Dataformat&format=xml&action=query&prop=links&plimit=500";
59   // query = "https://en.wikipedia.org/w/api.php?titles=Kurt%20Schwitters&format=xml&action=query&prop=links&plimit=500";
60
61   // query = "https://en.wikipedia.org/w/api.php?titles=Superegg&format=xml&action=query&prop=links&plimit=500";
62
63   try {
64     // The XML is loaded by creating a new XML with this query.
65     myXML = loadXML(query);
66
67     // A list of XML elements can be created using the getChildren () function, in this case all the entries
68     // are links, so we can loop through them with the tag <link>.
69     links = myXML.getChildren("query/page/links/child");
70
71     for (int i = 0; i < links.length; i++) {
72       // The list of XML elements obtained this way is processed and the attribute title read and displayed.
73       String title = links[i].getString("title");
74       println("Link " + i + " : " + title);
75     }
76   } catch (Exception exception) {
77     // catch the exception
78   }
79 }
```

M\_6\_2\_01\_GDV\_01

```

43 XML myXML;
44 XML[] links;
45 String query;
46
47 void setup() {
48   // In the URL for the database query, various parameters are given-e.g., that the result should be in XML
49   // format or that a maximum of 500 links should be returned.
50
51   Link 0: :design
52   Link 1: Activity-centered design
53   Link 2: Adaptive web design
54   Link 3: Adaptive design
55   Link 4: Aesthetic design
56   Link 5: Aesthetic design
57   Link 6: Aesthetic design
58   Link 7: Aesthetic design
59   Link 8: Algorithms-Aided Design (AAD)
60   Link 9: Application
61   Link 10: Architect-led design-build
62   Link 11: Architectural lighting design
63   Link 12: Architectural model
64   Link 13: Architecture
65   Link 14: AutoCAD
66   Link 15: Automotive design
67   Link 16: Automotive suspension design
68   Link 17: Building design
69   Link 18: Building design
70   Link 19: Book design
71   Link 20: Brand design
72   Link 21: C-K theory
73   Link 22: CMF design
74   Link 23: Circuit design
75   Link 24: Chartered Society of Designers
76   Link 25: Circuit board design
77   Link 26: Communication design
78   Link 27: Communication design
79   Link 28: Computer design
80   Link 29: Comprehensive layout
81   Link 30: Computer-aided design
82   Link 31: Computer-aided engineering design
83   Link 32: Computer-aided industrial design
84   Link 33: Computer-automated design
85   Link 34: Concept design
86   Link 35: Concept-oriented design
87   Link 36: Conceptual design
88   Link 37: Conceptual design
89   Link 38: Configuration design
90   Link 39: Corporate design
91   Link 40: Continuous design
92   Link 41: Corporate design
93   Link 42: Corporate design
94   Link 43: Costume design
95   Link 44: Creative industries
96   Link 45: Creative industries
97   Link 46: Creative problem-solving
98   Link 47: Creative problem-solving techniques
99   Link 48: Critical design
100  Link 49: Cultural icon
101  Link 50: Design
102  Link 51: Defensive design
103  Link 52: Design
104  Link 53: Design Council
105  Link 54: Design Research Society
106  Link 55: Design for Sustainable Association
107  Link 56: Design around
108  Link 57: Design brief
109  Link 58: Design committee
110  Link 59: Design by contract
111  Link 60: Design choice
112  Link 61: Design choice
113  Link 62: Design classic
114  Link 63: Design contest
115  Link 64: Design computing
116  Link 65: Design controls
117  Link 66: Design courses
118  Link 67: Design education
119  Link 68: Design ethics and principles
120  Link 69: Design engineer
121  Link 70: Design fiction
```

M\_6\_2\_01\_GDV\_01

```

43 XML myXML;
44 XML[] links;
45 String query;
46
47 void setup() {
48   // In the URL for the database query, various parameters are given-e.g., that the result should be in XML
49   // format or that a maximum of 500 links should be returned.
50
51   // It seems that this link gives a [Fatal Error] :1:1: Premature end of file. I think its because the
52   // url uses a http request.
53   // query = "http://en.wikipedia.org/w/api.php?titles=Superegg&format=xml&action=query&prop=links&plimit=500";
54
55   // These urls do work because they use an https request.
56   // query = "https://en.wikipedia.org/w/api.php?titles=Superegg&format=xml&action=query&prop=links&plimit=500";
57   // query = "https://en.wikipedia.org/w/api.php?titles=New%20York&format=xml&action=query&prop=links&plimit=500";
58   // query = "https://en.wikipedia.org/w/api.php?titles=Dataformat&format=xml&action=query&prop=links&plimit=500";
59   // query = "https://en.wikipedia.org/w/api.php?titles=Kurt%20Schwitters&format=xml&action=query&prop=links&plimit=500";
60
61   // query = "https://en.wikipedia.org/w/api.php?titles=Superegg&format=xml&action=query&prop=links&plimit=500";
62
63   try {
64     // The XML is loaded by creating a new XML with this query.
65     myXML = loadXML(query);
66
67     // A list of XML elements can be created using the getChildren () function, in this case all the entries
68     // are links, so we can loop through them with the tag <link>.
69     links = myXML.getChildren("query/page/links/child");
70
71     for (int i = 0; i < links.length; i++) {
72       // The list of XML elements obtained this way is processed and the attribute title read and displayed.
73       String title = links[i].getString("title");
74       println("Link " + i + " : " + title);
75     }
76   } catch (Exception exception) {
77     // catch the exception
78   }
79 }
```

Link 0: 1790 United States Census  
Link 1: 1800 United States Census  
Link 2: 1810 United States Census  
Link 3: 1820 United States Census  
Link 4: 1830 United States Census  
Link 5: 1840 United States Census  
Link 6: 1850 United States Census  
Link 7: 1860 United States Census  
Link 8: 1870 United States Census  
Link 9: 1880 United States Census  
Link 10: 1890 United States Census  
Link 11: 1900 United States Census  
Link 12: 1910 United States Census  
Link 13: 1920 United States Census  
Link 14: 1930 United States Census  
Link 15: 1932 Winter Olympics  
Link 16: 1940 United States Census  
Link 17: 1950 United States Census  
Link 18: 1960 United States Census  
Link 19: 1970 United States Census  
Link 20: 1980 United States Census  
Link 21: 1980 Winter Olympics  
Link 22: 1990 United States Census  
Link 23: 2000 United States Census  
Link 24: 2010 United States Census  
Link 25: 2020 United States Census  
Link 26: 21st century  
Link 27: 40 Wall Street Center  
Link 28: 42 Park Avenue  
Link 29: 500 Madison Avenue  
Link 30: 50 State Quarters  
Link 32: 70 Pine Street  
Link 33: 707 Madison Avenue  
Link 34: Abolitionism  
Link 35: African American  
Link 36: Academia  
Link 37: Adirondack High Peaks  
Link 38: Adirondack Mountain  
Link 39: Adirondack Northway  
Link 40: Adirondack Park  
Link 41: Adirondack Park divisions of New York  
Link 42: Admission to the Union  
Link 43: Advertising  
Link 44: Africa  
Link 45: African American  
Link 46: African American Civil Rights Movement  
Link 47: Agnosticism  
Link 48: AirTraffic JTF  
Link 49: Airport terminal  
Link 50: All Gore  
Link 51: All India Anna Dravida Munnetra Kazhagam  
Link 52: All India Anna Dravida Munnetra Kazhagam

M\_6\_2\_01\_GDV\_01

```

43 XML myXML;
44 XML[] links;
45 String query;
46
47 void setup() {
48   // In the URL for the database query, various parameters are given-e.g., that the result should be in XML
49   // format or that a maximum of 500 links should be returned.
50
51   // It seems that this link gives a [Fatal Error] :1:1: Premature end of file. I think its because the
52   // url uses a http request.
53   // query = "http://en.wikipedia.org/w/api.php?titles=Superegg&format=xml&action=query&prop=links&plimit=500";
54
55   // These urls do work because they use an https request.
56   // query = "https://en.wikipedia.org/w/api.php?titles=Superegg&format=xml&action=query&prop=links&plimit=500";
57   // query = "https://en.wikipedia.org/w/api.php?titles=New%20York&format=xml&action=query&prop=links&plimit=500";
58   // query = "https://en.wikipedia.org/w/api.php?titles=Dataformat&format=xml&action=query&prop=links&plimit=500";
59   // query = "https://en.wikipedia.org/w/api.php?titles=Kurt%20Schwitters&format=xml&action=query&prop=links&plimit=500";
60
61   // query = "https://en.wikipedia.org/w/api.php?titles=Superegg&format=xml&action=query&prop=links&plimit=500";
62
63   try {
64     // The XML is loaded by creating a new XML with this query.
65     myXML = loadXML(query);
66
67     // A list of XML elements can be created using the getChildren () function, in this case all the entries
68     // are links, so we can loop through them with the tag <link>.
69     links = myXML.getChildren("query/page/links/child");
70
71     for (int i = 0; i < links.length; i++) {
72       // The list of XML elements obtained this way is processed and the attribute title read and displayed.
73       String title = links[i].getString("title");
74       println("Link " + i + " : " + title);
75     }
76   } catch (Exception exception) {
77     // catch the exception
78   }
79 }
```

Link 0: Anton Chekhov  
Link 47: Anton Giulio Bragaglia  
Link 48: Antoni Gaudí  
Link 50: Antonin Artaud  
Link 51: Antoni Gaudí  
Link 52: Aristarch Lebedev  
Link 53: Amory Show  
Link 54: Amorphous  
Link 55: Arnold Schoenberg  
Link 56: Arnold Subbili  
Link 57: Arnold  
Link 58: Art Deco  
Link 59: Art Nouveau  
Link 60: Art Intervention  
Link 61: Art intervention  
Link 62: Art interventionism  
Link 63: Art pop  
Link 64: Artwork  
Link 65: Arthur Homer  
Link 66: Arthur Koestler  
Link 67: Arthur Miller  
Link 68: Arthur Rimbaud  
Link 69: Arthur Segal (painter)  
Link 70: Arthurian legend  
Link 71: Aesthetic writing  
Link 72: Aesthetic stage (art)  
Link 73: Aesthetic stage  
Link 74: Altas Press  
Link 75: Alcatraz  
Link 76: Amedeo Modigliani  
Link 77: August Strindberg  
Link 78: Auguste Rodin  
Link 79: Avant-garde  
Link 80: Avant-garde art  
Link 81: Avant-garde metal  
Link 82: Avant-garde music  
Link 83: Avant-garde  
Link 84: Avant-prog  
Link 85: Avant-punk  
Link 86: Avant-rock  
Link 87: BBC News  
Link 88: Beets Russes  
Link 89: Beethoven  
Link 90: Barbican school  
Link 91: Baroque  
Link 92: Battleship Potemkin  
Link 93: Bahá'u'lláh  
Link 94: Bahá'í religion  
Link 95: Beatrice Wood  
Link 96: Bengal School of Art  
Link 97: Bengal Renaissance  
Link 98: Bengal Renaissance

MyCodeHistory: 2 August 2015

This is the second more or less theoretical exercise about the chapter 'Data from the Internet.' Basically, this exercise is not much different from the previous one. You download XML data from Wikipedia's Superegg page. The XML text is displayed in the Processing Console. The difference is that during that download procedure Processing is running an animation. In addition to that, the draw block must constantly check whether the download process has already started and finished. If all the XML is downloaded, Processing displays a text in the console that the XML text is available.

Now when I run the program (M\_6\_2\_02) it shows that it is not working. Nothing is displayed in the Processing console. Although the animation is running in Processing's display window. When I'm clicking in the display window I get a comment in the console which says: "XML not loaded yet. Plus a remark [Fatal Error]: 1: 1: Premature end of file. So I suspect that this is the same mistake as in the previous program (M\_6\_2\_01). And it is! The URL in the program begins with `http://`. The correct URL begins with `https://`. When I change that, the program works fine. First it begins with the animation. After the display window is clicked, the XML text is displayed in the console.

That's basically all there is to say about this exercise. I've considered for a moment making a number of animations that are different than the one which is generated by this program. But that's a bit crap because the next program does just that. So it's better to make variations on that program. I understand the principle behind this one so I leave it for what it is and I am going to concentrate on the next exercise.

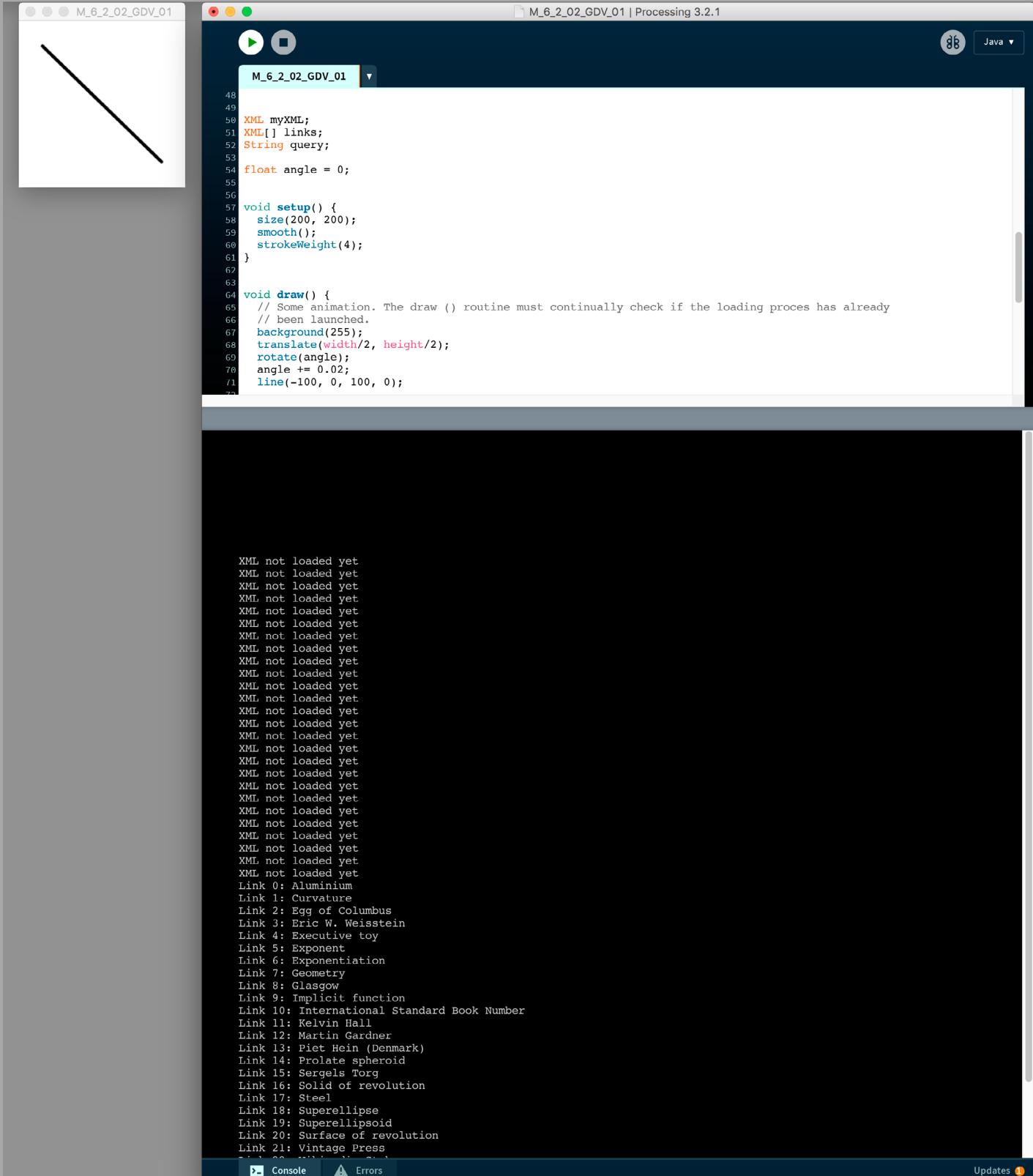
And while I am writing this text in December 2016 I noticed that the program of August 2015 did not work anymore. I replaced the `http:` with `https:` but that didn't make any difference. I get an error message: The function "`loadXMLAsync2(M6_2_02_GDV_01, String)`" does not exist. I checked the Generative Design library. And the function `loadXMLAsync2` does still exist. I checked the example file for `loadXMLAsync2`. In the header of the file is mentioned: This function needed to be implemented for Processing 2.X because the previously used class `XMLElement` was renamed to `XML`. Ah... I am running Processing 3 so I left out the 2 (from `loadXMLAsync2`). And then the program worked fine.

---

Progress is impossible without change,  
and those who cannot change their  
minds cannot change anything.

George Bernard Shaw, 1856–1950  
British Playwright, critic and political  
activist.

```
not loaded yet
Link 0: Aluminum
Link 1: Curvature
Link 2: Egg of Columbus
Link 3: Executive
Link 4: Exponent
Link 5: Geometry
Link 6: Glasgow
Link 7: Implicit function
Link 8: Kelvin Hall
Link 9: Martin Gardner
Link 10: Piet Hein (Denmark)
Link 11: Power (mathematics)
Link 12: Prolate spheroid
Link 13: Sergels Torg
Link 14: Solid of revolution
Link 15: Steel
Link 16: Superellipse
Link 17: Superellipsoid
Link 18: Toy
Link 19: Vintage Press
```



```
M_6_2_02_GDV_01 | Processing 3.2.1
Java ▾
```

```
48
49
50 XML myXML;
51 XML[] links;
52 String query;
53
54 float angle = 0;
55
56
57 void setup() {
58   size(200, 200);
59   smooth();
60   strokeWeight(4);
61 }
62
63
64 void draw() {
65   // Some animation. The draw () routine must continually check if the loading proces has already
66   // been launched.
67   background(255);
68   translate(width/2, height/2);
69   rotate(angle);
70   angle += 0.02;
71   line(-100, 0, 100, 0);
72 }
```

```
XML not loaded yet
Link 0: Aluminium
Link 1: Curvature
Link 2: Egg of Columbus
Link 3: Eric W. Weisstein
Link 4: Executive toy
Link 5: Exponentiation
Link 6: Exponentiation
Link 7: Geometry
Link 8: Glasgow
Link 9: Implicit function
Link 10: International Standard Book Number
Link 11: Kelvin Hall
Link 12: Martin Gardner
Link 13: Piet Hein (Denmark)
Link 14: Prolate spheroid
Link 15: Sergels Torg
Link 16: Solid of revolution
Link 17: Steel
Link 18: Superellipse
Link 19: Superellipsoid
Link 20: Surface of revolution
Link 21: Vintage Press
```

Console Errors Updates 1

MyCodeHistory: 6 August 2015

The first thing I tried is to take a different keyword for each exercise and not the keyword 'Superegg'. This time I have chosen the name 'Steve Jobs'. And that does not work. Except when I bring back the plimits in Wikipedia node from 500 to 100. The program also keeps on displaying [Fatal Error]: 1: 1: Premature end of file in the console. But it continues to work. So I do not worry too much about it. Commented out the frame. setResizable (true). I'm not going to increase or decrease the display window. I keep it always at 800 x 800 pixels. I made the background black. But then it seems that all rectangles below the text are displayed. That does not seem right to me. Which function is responsible for displaying those rectangles? Ah... some of the rectangles dissolve in the background over time. The backgrounds are black now but it would be better if they were completely transparent. I find it unnecessary to display them. Another thing is that the nodes are getting much more attention than the texts. While the text (for me) is more important. And there are a few more things I cannot fix at this time. That is something that I have to do in a later stage.

There is a problem with the color scheme. Two color modes (RGB and HSB) are mixed. Why? And who uses what? Let me first start filling in another keyword. I choose 'Processing (programming language)'. I am trying to figure out why I have two different shades of red. Ok... everything is changed and the place to do that can be found in the draw label block in the Wikipedia Node class. All top-nodes are now red. The sub-nodes are blue. Not everything is quite clear to me. I have extended the display time of the sub-nodes. I think they're too short on the screen. With the start of the next paragraphs I will mention the keyword first.

**Keyword:** Data Visualisation. It seems better to make smaller nodes and fill them with white. Probably it is also better to use a larger point size for the top-nodes. The point size of the sub-nodes is fine.

**Keyword:** Color. I guess that the point size should stay lower than 1.0. Zooming in the keyword related top-nodes increases their point size. And that's actually a good thing. Because you eventually want to know where the top-nodes are located. But maybe this is not such a good idea after all. I think this will become clear later.

**Keyword:** Interpolation. Actually I would also be able to zoom in and out much deeper. And maybe the length of the springs needs to be shortened. So just for fun, I have increased the node radius to ten. That means I get very short springs. When I clicked on a large number of links it gives me total chaos.

**Keyword:** Complexity. I have no idea why the nodes are drawn in three steps. So I started with commenting out the last dot. This results in one circle less in the node. When the second block of code is commented out the program displays no node at all. I check the code in the original program. The first circle is drawn to the first node. But apparently I do not see that. The second circle is a white ring between the center circle and the link circle. And the last circle is the point at the center of the node. I exaggerated the second circle in size with a very low alpha channel.

**Keyword:** Population genetics. I changed the color blue. The text is more readable now. Perhaps I should give the larger circle a bit more transparency. Finally opted for a line version. But that made the image very unclear and therefore less legible. So I go back to the earlier version. Brought the nodeRadius back to its original setting.

**Keyword:** Greek history. In one way or another, the program does not always start. Sometimes it does just nothing when you let it run. Sometimes it works. I have no idea why. The readability of the blue text is no longer optimal when the text is on light grey rectangles. I could replace the rectangles by horizontal or vertical lines. I now uncomment the original white rectangles. But I think the overall picture did not improve. I have applied a gradient in the background. But I still think that those rectangles behind the text are not an improvement.

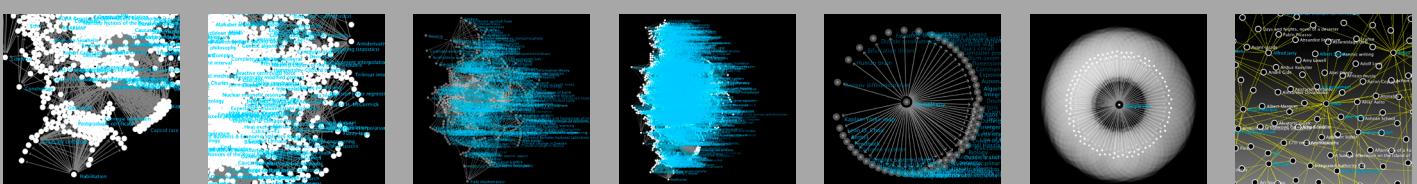
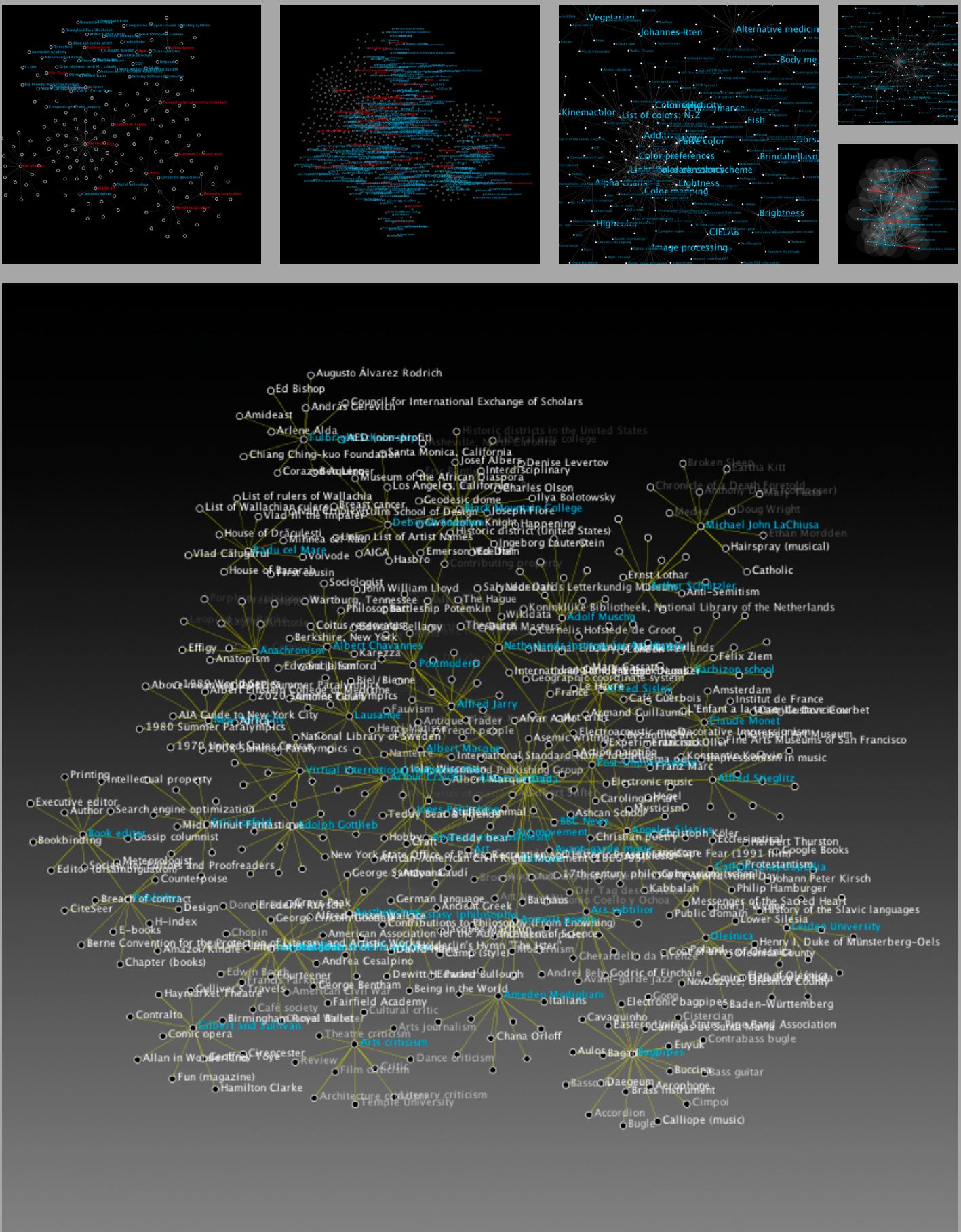
**Keyword:** Typography. Time to add some sophistication. I'm going to remove the rectangles behind the text. And I think that the gradient must begin darker at the bottom otherwise the texts are no longer legible.

**Keyword:** Dada. A few more optimisations. But I have not changed a lot when you compare it with the earlier version. The focus was on the details. And details make (or break) the image.

---

I never guess. It is a capital mistake to theorise before one has data. Insensibly one begins to twist facts to suit theories, instead of theories to suit facts.

Sir Arthur Conan Doyle, 1859–1930, Scottish novelist, short story writer, poet and physician.



MyCodeHistory: 10 August 2015

Extremely annoying to say but this tool does not work. It displays the keyword: Design. And you can add a new node. But that's all. I started the program several times. But I have figured out three different ways I can still use this tool. The first way is to import the changes I made in M.6.3.1 (previous program) and just wait and see what happens. The second way is to build on the previous program (M.6.3.1). And then forget about the tool. And the third way is to try to make my version of the tool. That program would have absolutely nothing to do with what is described in the Generative Design book. But that would obviously be sad. Since I have been able to use all the programs so far.

I first replaced in the WikipediaNode class all `http://` links by `https://`. And then I was really surprised! The program worked normal. I got no more errors in the console. So the problem seems to be solved easily. I reported that on the Generative Design website. There could be another fact that causes errors. I work with a Wacom tablet and a pen. When I start the program with the pen it does not work in some cases. When I start the program with the mouse on the Wacom tablet then it starts most of the time. Sounds very vague. But it is what it is.

As usual I started to type all the book-comments into the code. I also downloaded the complete font family MISO. MISO is an architectural lettering font completed in 2006 by Märten Nettelbladt. I wanted to test whether any of these fonts (or a combination of them) is also suitable for use in this program. But unfortunately this did not lead to better results. I did not choose a different keyword. That will happen in the following exercises. The keyword for this exercise stays 'Design'. I am now going to set the first variables. Auto Zoom is true. Invert background is true. I do not know what the variable `resultCount` represents. So I'm going to increase it from 10 to 50.

Ah... `resultCount` is the number of nodes with which the program starts. I put this back to 1 for now. Then it is much easier to track where all nodes are going to be linked. I think a `resultCount` of 500 is the limit for this program. That number is also stated in the API lines of the WikipediaNode class. There also occurs a strange side effect in the graphics of this program. The design node begins to vibrate violently when I use higher numbers of nodes. I have no idea why. I also removed all color information from the image. I will bring color in the graphics in the next program.

Let's use design related keywords for all of these variations. I will use the keyword 'Designer' for this exercise. That's a tiny difference from the previous keyword: Design. But there is a big difference in the end-result. Specially when you put the spring length at 10 or 500. I can influence that amount via the GUI. So these differences are even getting bigger. I reduced the minimum setting. And the doubled maximum setting in the GUI. The disadvantage is that the texts in many nodes can no longer be read. So you may have to zoom in. Actually, this is typographically not very wise. Perhaps the texts had to be decreased more slowly when creating more nodes. Or switch off automatic zoom.

**Keyword:** Industrial Designer. I have put the spring stiffness very low. This way you can make beautiful organic structures. And if you want to go that way the tool can also create unexpected node clusters.

**Keyword:** Design Research. What happens when I double the slider for the spring stiffness again? Than everything becomes even more interesting. Only the readability goes steps backwards. In this case, I do not care but my clients will appreciate it less.

**Keyword:** Graphic design. Spring strength and spring stiffness are at their maximum. Spring damping is on its minimum. This results in very fancy circles which are formed by the nodes. As you change the settings the circles are getting more organic again.

**Keyword:** Industrial design. I Increased the node radius to its maximum. Once to its minimum and one time half of the slider. I also increased the spring length. At a certain moment all nodes are piling themselves up. Therefore it seems that there are fewer nodes present. But when you adjust the settings the nodes are getting visible again.

**Keyword:** Designer drug. Brought every setting to its maximum. But wait! That's not a reassuring warning in my console: Error during `asyncHTMLLoad` – but it seems not to be a problem. Everything is still working fine.

**Keyword:** Design Patent. Brought the line weight to an extreme setting. This results graphically in nice images. But I really should do something about the size of the texts. I'm going to try to solve that problem when I will work with the next program.

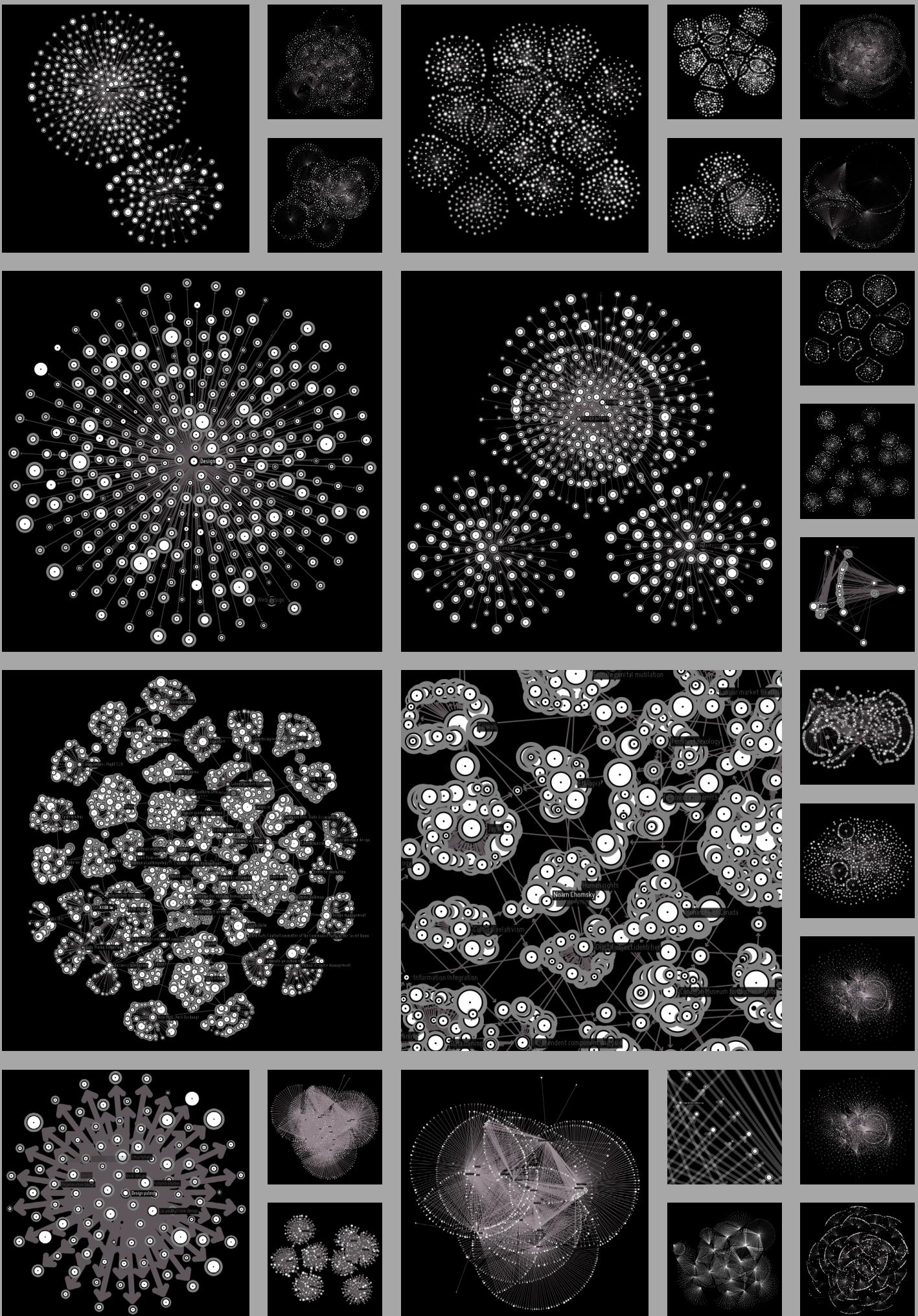
**Keyword:** Software design. Just a variation in which I made the line thickness excessively thick. It is less sophisticated but not really wrong. Plus some screen dumps of the moments that I've zoomed in really deep on a constellation of nodes and springs.

**Keyword:** User interface design. Actually, the grey text on black rectangles is not readable. Perhaps there is also room for improvement for that issue in the next program.

---

'I love to be a graphic designer, but could we get rid of clients somehow please?'

Erik Spiekermann, 1947–, German typographer and designer



MyCodeHistory: 17 August 2015

This program is about the functional use of color. Furthermore, I'm going to do something about the readability of the text. In the earlier program (M.6.4) I had chosen to take design related keywords. Let me choose this time for computer-related words. I begin with the keyword: Computer. I switched on colorise nodes. And than some strange things happen. I ran the program and it does not work. Only after three restarts (or re-runs), it works. I try it again. Now the program works only after starting twice. That's one thing. Another thing is that I have reduced the nodes (resultCode) from fifty to five. The first time I get the following nodes from the keyword 'Computer': Kermit (protocol), Computer Monitor, EGB, fighter aircraft and Alan Turing. The second time I get the following nodes: Digital object identifier, Interactive fiction, Floppy disk, first-person shooter, Lisp (programming language) and Digital object identifier. The third time I run the program: Computer data storage, Mac OS, BUNCH, BASIC, and Nano Engineering. So I keep getting different results (links / nodes) with the same keyword. I'm not sure if I like that a lot. Concerning color, we have three groups: Science is blue. And geography and politics are yellow. Culture and art is purple. The colours are fine. I've just made them a bit brighter. And it is still true that I have to start the program at least three times before it works (or runs).

Keyword: Software. I have to think of something that can improve the readability of the text. I think the grey text is not (or hardly) legible. So I removed the black rectangle below the text. Just like I did in the previous exercise. With regard to the size of the font you simply turn off auto zoom. And than it reads all just fine. With auto zoom turned on text will automatically get larger or smaller. On a certain moment (very large or ver small) the texts are unreadable. To increase the size for all texts (to allow them always be legible) is tricky because at a certain stage you cannot see the graphics anymore.

Keyword: Hardware. I want to see if I can get rid of those circles. I do a search in the program itself and actually all circles are generated in the Wikipedia Node class. I swapped the circles with rectangles. But there's still a circle inside the rectangles.

Keyword: Malware. I've changed the color scheme. Science is now white, geography and politics are now red. And culture and art is now blue. All shapes now consist solely of squares. Changed the colours again. Science is now blue, geography and politics are now purple and culture and art is now red.

Keyword: Algorithm. I have changed the squares into rectangles. That creates a less chaotic image than the previous images. But is less chaotic any better?

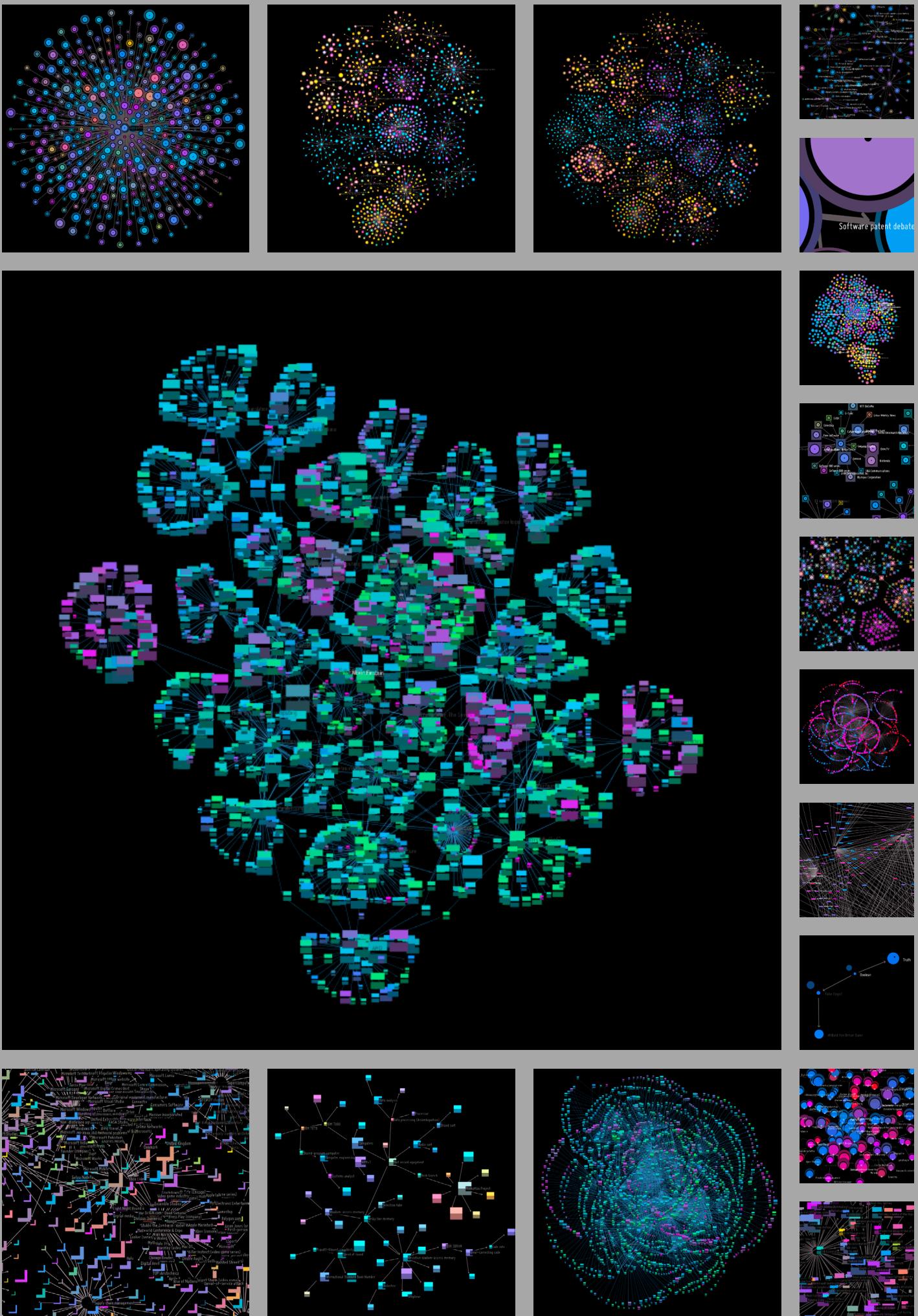
Keyword: Boolean. Maybe it's good to go back to the original version with circles. And maybe those circles do not have to sit on each other but close to each other. Or maybe side by side. This also means that some colours will flow into the color of their neighbour. And I think an outline version also could be an option. But for now I only will give the center dot (for back links) a white outline. There is still one problem with the color of some text. Some stay grey. Others remain white. I also think that the node's circles should be a little closer to each other.

Keyword: Programming language. Think it is time to sort out a few things. I feel that I have strayed away a bit. In the original program, each circle reflects a Wikipedia article. The arrows between the articles show whether an article is linked to another article. Then there are three themes. Blue is science, nature and society is yellow and purple is art and culture. The size of the circle indicates the length of the article. And the thickness of the outer ring represents the number of links in the article. The dot in the center indicates if there are multiple items. So I think I should go back to these principles.

Keyword: Virtual machine. I have to change the names of some variables in the program. As 's' and 'b' and 'd' tell me so little about their functionality. I now have some circles commented out. The circles (who are on), are replaced by rectangles. Eventually I've replace the last circle to a rectangle. And now all the information is translated from circle to rectangle. While this does not give me an image which is wrong. It is also not entirely true. I do have the impression that it is an improvement on M\_6\_5\_01\_GDV\_04. But I really want to try something else.

Keyword: Processor. Actually, I want to get rid of the circles because it does not give a proper interpretation of the data. Why? I suggest to buy Alberto Cairo's book: 'The functional art'. He explains the circle problem better than I can. What I want to try is not to make a square but a horizontal line. I now have two horizontal rectangles. The lower rectangle has a color which is a percentage of the upper rectangle color. If there is no bottom rectangle, then there is no link. The quantity of links I cannot measure now. And the texts have to be moved out-of-the-way.

Keyword: Microprocessor. I was not able to figure out how I could make the visual translation to the quantity of links in the lower rectangle. But you can clearly see if there are any links. So this exercise was not 100 percent successful. But perhaps 75 percent. Let's see if I succeed to 100% in the next exercise. I'll also try to bring more structure to the layout of the nodes (as I've tried in M\_6\_4\_01\_GDV\_06). That has to be better in the next program.



MyCodeHistory: 21 August 2015

This is still the same tool I used in the previous program. And as I said in the previous exercise it is now my intention to use all functionality of the tool to create images. I have made a few screen dumps from the arrangement as it was in the earlier exercise (M.6.5). So the colours I have used in this previous set-up are changed and do not correspond to the above list. Any node reflects a Wikipedia article. And the keywords that I would like to use are coming from people who are working in the science, nature, society, art and the cultural world. I start with: Louis B Mayer (1885–1957), US film producer.

Keyword: Maria Callas (1923–1977), Greek operatic soprano. In this graphic are three theme colours. But I want to modify those because I think there are more possibilities. Science is now red, nature and society are yellow and art and culture is blue. And looking a bit deeper into this linking mechanism it seems that Maria Callas was little concerned with science. That is why there is only a bit red present in the image. The majority is art, which is blue-ish. Of course it depends on how many nodes you click. But there also are interesting relationships noticeable. For example, what has Maria Callas to do with L-Dopa? I continue to change colours during the following exercises.

Keyword: Ralph Waldo Emerson (1803–1882), US writer. Because the colours are interpolated it seemed better to me to choose three very different colours. I have made science red, nature and society white, art and culture is blue. The disadvantage of white is that it brings back all bright colours to a shade or a tint of that color. So I replaced white with magenta. Probably this is not a good choice but I think yellow is still good one. Although I think it's a terrible color combination. I experiment now by exaggerating the shapes. Which also leads to unpleasant results.

Keyword: Alma Mahler Werfel (1879–1964), Wife of composer Gustav Mahler, then architect Walter Gropius, and finally writer Franz Werfel. There are now three bars above each other. The white bar is displayed only if there are additional items. The middle bar represents the themes and the lower bar is displayed when there are links.

Keyword: Arthur Wellesley 1st Duke of Wellington (1769–1852), British general and statesman. What will happen if I drop all those rectangles and just start with text and lines. Maybe I should also cut the arrow points for a moment. Finally, the main thing that I am interested in is the article's name. So I'm going to bring this chart all the way back to its essence.

Keyword: George Sand (1804–1876), French novelist and memoirist. I didn't use the function that lets you specify coloured lines. That gives a completely different picture.

Keyword: Ludwig Mies van der Rohe (1886–1969), German-American architect. Color was added to the central point that indicates that there are back links. I think the point maybe slightly larger. Or maybe I should emphasise the arrows. Actually, I make the same mistake again. The emphasis is now on the nodes. While I would like the emphasis to be on the text.

Keyword: Eva Duarte de Perón (1919–1952), Actress and First Lady of Argentina. I have associated the colours with the text. So now you see directly see to which group the text belongs. Over time the coloured texts disappear but the context of the categories is taken over by the coloured dots in the nodes.

Keyword: Groucho Marx (1895–1977), Comedian. I tried to find out how many links there are possible. But only so that it all remains manageable. I also think that those black circles in the background do a great job. I leave them in.

And let's spend the last keyword to the woman to whom we owe all this programming: Ada, Countess of Lovelace (1815–1852), English mathematician, writer and pioneer of computing. And the Wikipedia link is not working. What is wrong? Nothing! Just kidding!

---

This was the last program of the Generative Design book. This is also the last publication about the Generative Design Variations project. Technically I have modified 1219 programs. The programs generated 5957 images. And I have written 83 articles about this project which lasted from November 10, 2013 until August 21, 2015.

---

Don't cry because its over. Smile because it happened.

Theodor Seuss Geisel aka Dr Seuss, 1904–1991, American writer and illustrator.

