ZEITGEIST

A project at the intersection of computation, contemporary history and the arts.

Synopsis

The ZEITGEIST engine analyses a corpus of contemporary news articles and visualises them as a moving picture. This talk explains the conception, architecture, and design decisions in ZEITGEIST, and gives some thoughts on its interpretation.

Sections within this Talk

- Conception and Motivation
- Demonstration
- Architecture
- Computational Features
- Example Run
- Artistic Design Decisions
- Interpretation of the results
- Outlook & Future Experiments
- Credits and Sources

The foundation for ZEITGEIST originated at 21C3

- I attended a talk by the Austrian group Quintessenz on “Datamining the NSA”.
- While Quintessenz performed the analysis by hand, I wanted to use computational techniques.
- I believed computational techniques can produce a more objective and more comprehensive view.
- I was also keen to use visualisation to capture the audience's attention and imagination.

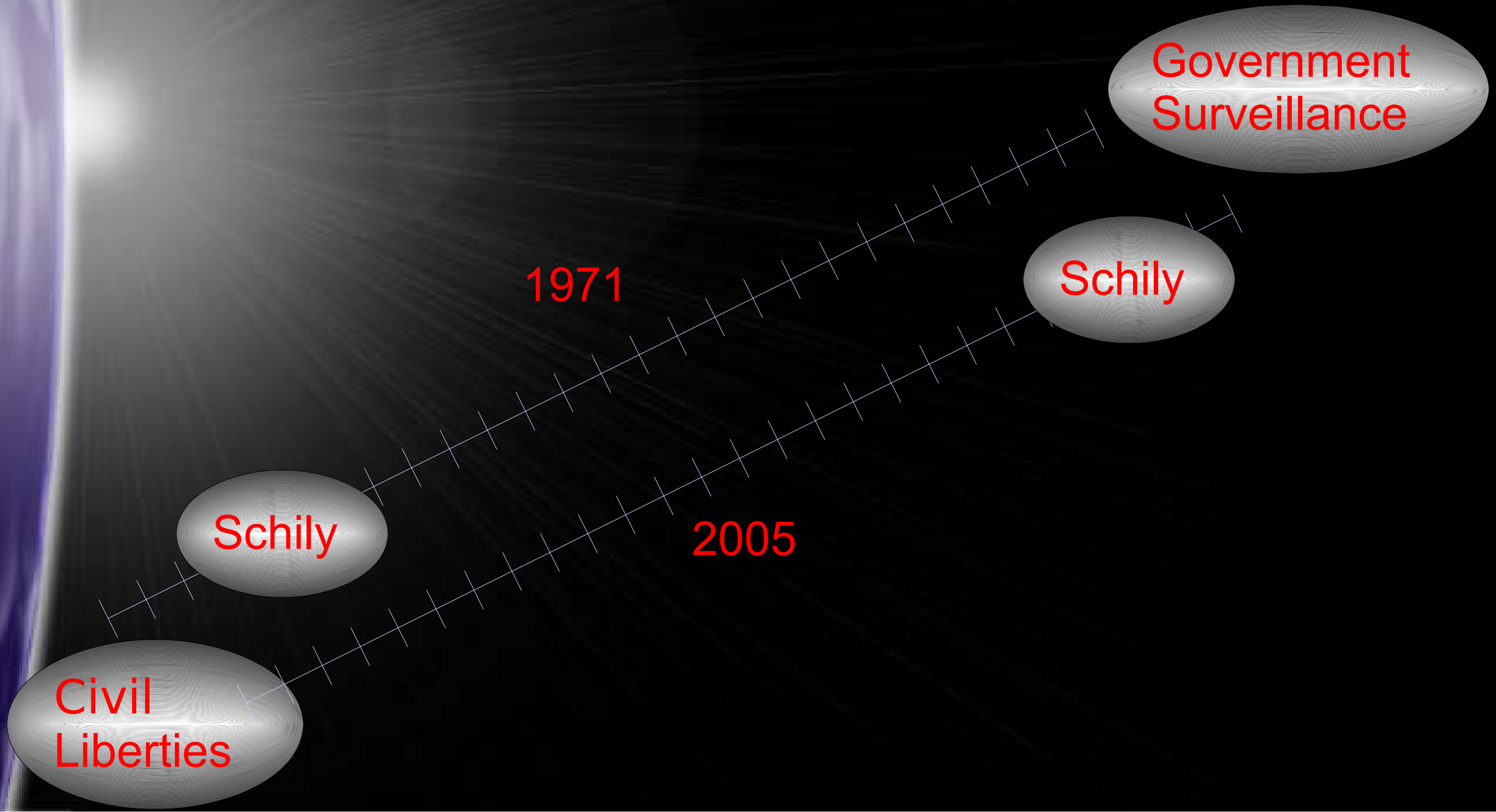
Is consistency an ethically “good thing”?

- I was intrigued by the career of german politician Otto Schily.
- Mr Schily came to fame as defendand attorney for the leftist extremists RAF (red army fraction terrorist group).

- During our last legislative cycle, Mr Schily headed the Ministry for Interior Affairs -- superficially observed -- the opposite of his former persuasions.
- This change of mind has happened over a long period of time. Other individuals are much more fickle.

How can text be visualised (1/3)?

- The spectrum of Mr Schily's aforementioned activitys can be characterized along the dimension “control”:



Government Surveillance

1971

Schily

Schily

2005

Civil Liberties

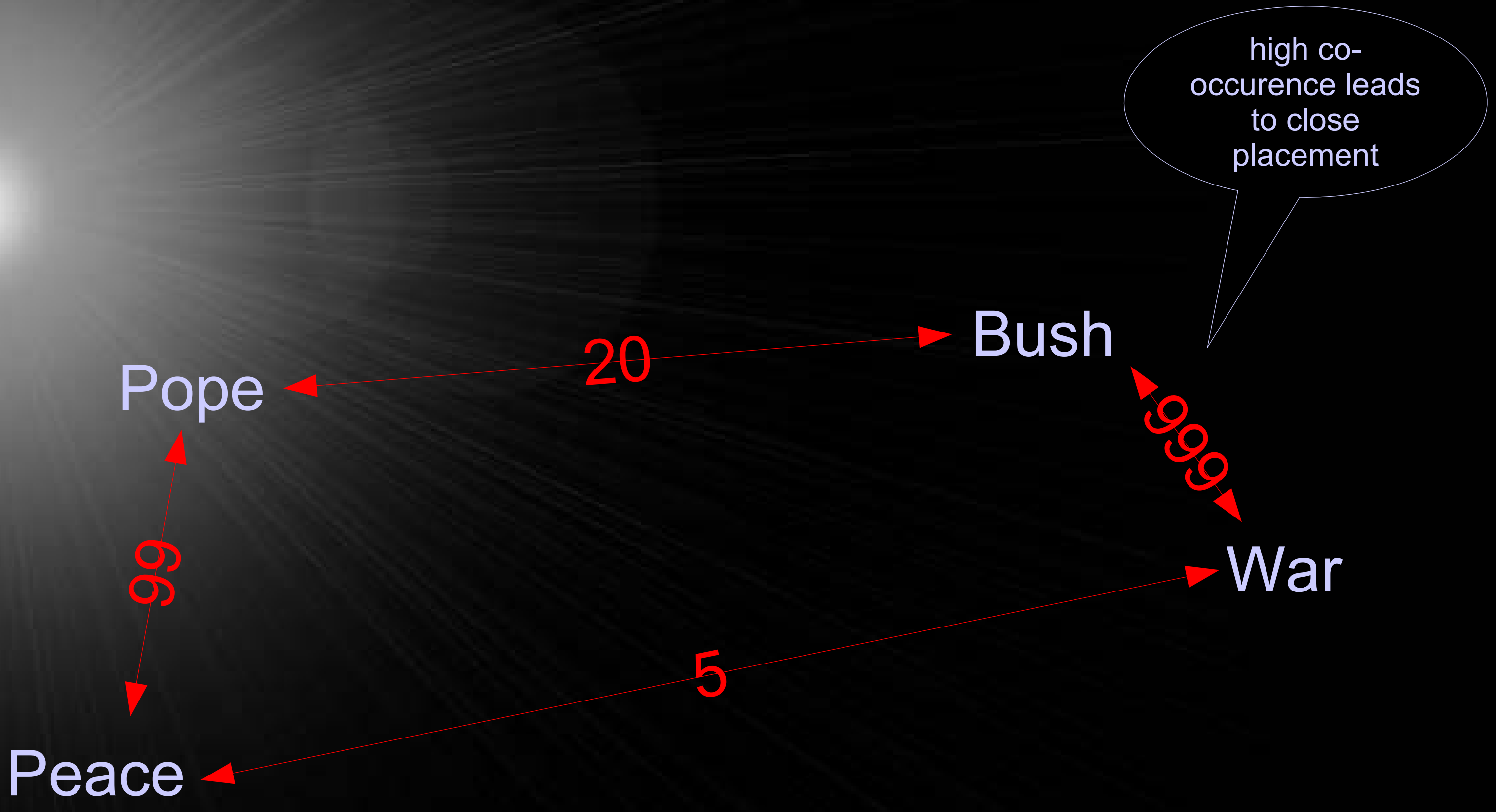
How can text be visualised (2/3)?

- If words occur frequently together ...

	Bush	Peace	Pope	War
Bush	∞	0	20	999
Peace	-	∞	99	5
Pope	-	-	∞	2
War	-	-	-	∞

How can text be visualised (3/3)?

- ... they get placed more closely to each other:



high co-occurence leads to close placement

Bush

Pope

20

999

War

99

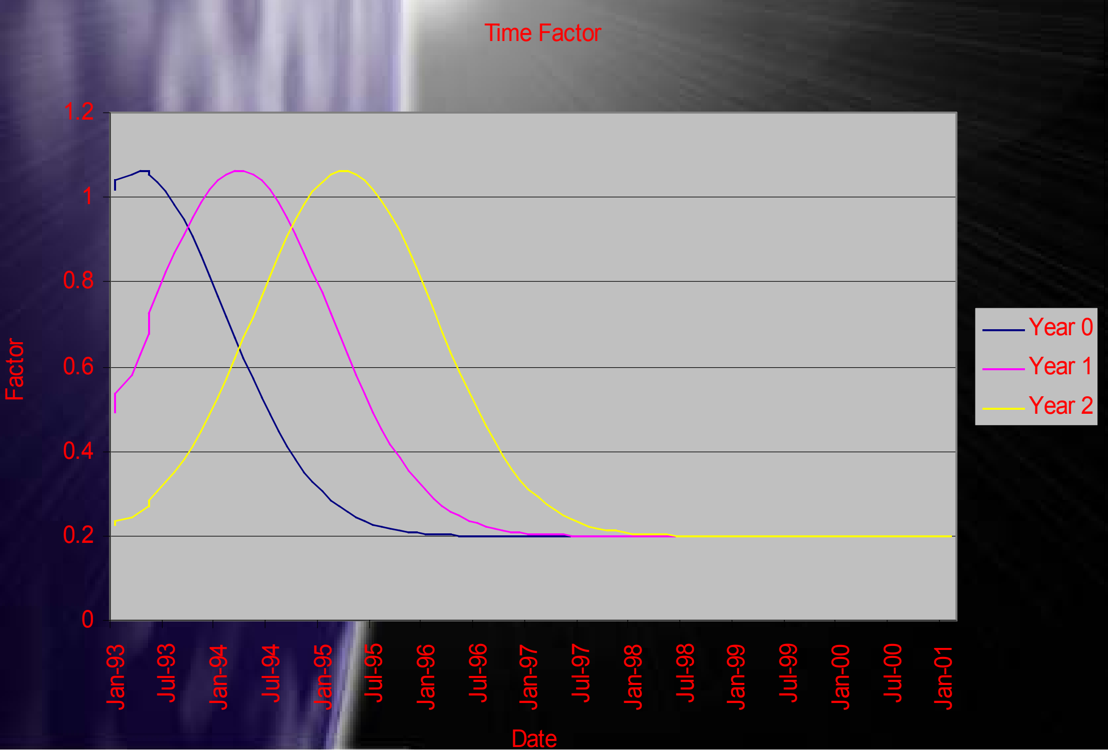
5

Peace

How can the visualisation be animated?

- The previous example considered the co-occurrence counts across the complete corpus equally.

	Bush	Peace	Pope	War
Bush	∞	0	20	999
Peace	-	∞	99	5
Pope	-	-	∞	2
War	-	-	-	∞



- The time effect is obtained by scaling the co-occurrences with a bell-shaped (width σ=300) factor.
- Many movie-frames are computed with varying μ (three differing μ are shown on the left, 4000 are used).

$$Factor(\mu,\varsigma)=e^{-(x-\mu)^2/2\varsigma^2}/\varsigma\sqrt{2\pi}$$

Sections within this Talk

- Conception and Motivation
- Demonstration
- Architecture
- Computational Features
- Example Run
- Artistic Design Decisions
- Interpretation of the results
- Outlook & Future Experiments
- Credits and Sources

Demonstration

- Movie – Words only visualisation

- Movie – Image icon visualisation

Sections within this Talk

- Conception and Motivation
- Demonstration
- Architecture
- Computational Features
- Example Run
- Artistic Design Decisions
- Interpretation of the results
- Outlook & Future Experiments
- Credits and Sources

Program Architecture

- ZEITGEIST is written in JAVA.
- The engine consists of a set of batch programs that must be run in sequence.
- Communication between the processing steps happens through files.
- All configuration happens through a JAVA properties file.
- All parts of the system are optimised for utilisation of SMP machines.
- Component libraries have been employed where possible.

Sections within this Talk

- Conception and Motivation
- Demonstration
- Architecture
- Computational Features
- Example Run
- Artistic Design Decisions
- Interpretation of the results
- Outlook & Future Experiments
- Credits and Sources

Layout Algorithm (1/3)

- A distance matrix M is produced from the Co-Occurence matrix C through the rule:

$$M_{i,j}=1/C_{i,j}$$

- The algorithm „Graph Drawing by Stress Majorization″ by Gansner, Koren & North from the AT&T research laboratories – as seen in the „graphviz″ utility – constructs a computational model of steel springs adjusted in length to the distances in the matrix M.
- The stress in this model is reduced via a computational minimisation method.

Layout Algorithm (2/3)

- The algorithm starts with a list of n random 2D vectors describing the positions of the vertices and at each step reduces the stress function:

derived from M

$$stress(M, X) = \sum_{i<j} \omega_{ij} (\|X_i - X_j\| - M_{ij})^2$$

- The algorithm stops when the reduction in stress is a factor less than 1e-5:

$$stress(M, X_{n+1}) / stress(M, X_n) \le 10^{-5}$$

- The reduction step itself is absolutely brilliant and took me 9 weeks of reading to comprehend. Your math may be better.

Layout Algorithm (3/3)

- ... or it may not be
- The reduction step works as follows:

$$L^{\omega} X = L^{Z} Z$$

$L^{?} = graph\ Laplacian$

$Z = approximation\ for\ X$

- For the construction of the movie, the distance matrix M changes slightly between frames.
- With a very similar distance matrix, the 2D vectors X from the previous frame -- describing the coordinates of the words -- are a very good approximation to the 2D vectors of this frame.

⇒ Very few iterations are needed.

Solving Matrix Equations (1/3)

- The reduction step really solves the system:

$$Ax=b$$

- My particular system is dense (memory representation), and positive definite (mathematical property).
- My problem chooses 2,000 representative words, I compute 4,000 frames with 8 iterations each (plus some startup).
- The worst approach is to invert the matrix and multiply by the result vector ...
- ... with a naïve approach, it takes 3800s (more than an hour) per iteration. I tried.

Solving Matrix Equations (2/3)

- I also tried:
  - JAMA – java matrix package
  - MTJ – matrix toolkit for java
  - MTJ+ATLAS – not on my architecture
  - OR – opsresearch.com
  - COLT – CERN research labs
- None really was using SMP satisfactorily.
- Problem sizes up to 500 words can be computed in a day on my laptop.

- MTJ 360s/8 iterations, JAMA 120s/8 iter.s

Solving Matrix Equations (3/3)

- I broke the CompSci mantra of „don't build it yourself!".
- My matrix library uses the Java 5 „concurrency" classes (and a backport to Java 1.4).
- It does not solve general systems of equations, but uses „Conjugate Gradient" to solve „positive definite" systems.
- It uses an accuracy of 1e-10, not 1e-1000, which is good enough for my needs.
- It runs 8 iterations in a total of 5 seconds.

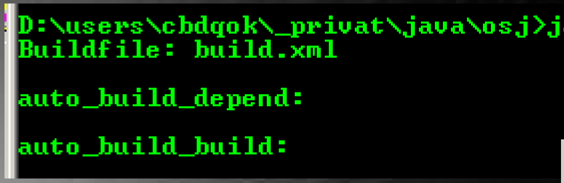
➡ Speedup by factor of 6080 !!!

Sections within this Talk

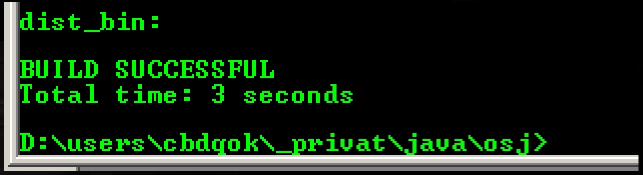
- Conception and Motivation
- Demonstration
- Architecture
- Computational Features
- Example Run
- Artistic Design Decisions
- Interpretation of the results
- Outlook & Future Experiments
- Credits and Sources

Compiling and Configuration

- Compiling the system is easy:

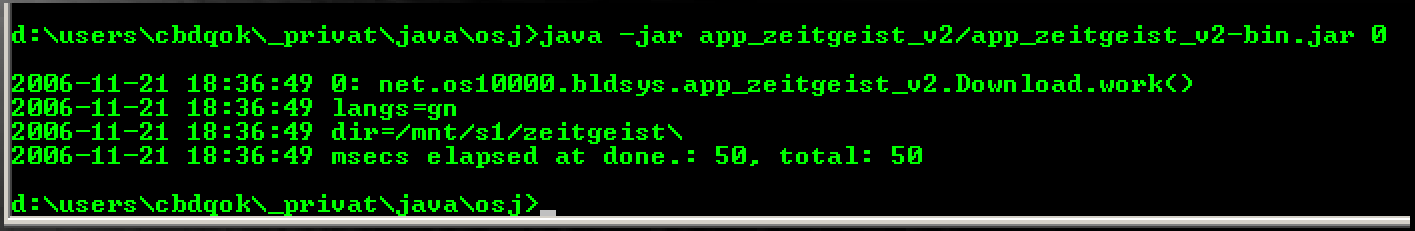


```
D:\users\cbdqok\_privat\java\osj>java -jar ant14.jar dist_bin
Buildfile: build.xml

auto_build_depend:

auto_build_build:
```



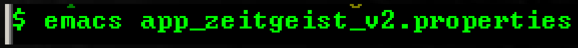
```
dist_bin:

BUILD SUCCESSFUL
Total time: 3 seconds

D:\users\cbdqok\_privat\java\osj>
```

- Running it creates the config file:

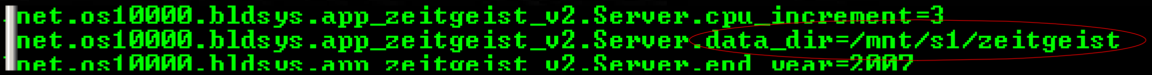


```
d:\users\cbdqok\_privat\java\osj>java -jar app_zeitgeist_v2/app_zeitgeist_v2-bin.jar 0

2006-11-21 18:36:49 0: net.os10000.bldsys.app_zeitgeist_v2.Download.work()
2006-11-21 18:36:49 langs=gn
2006-11-21 18:36:49 dir=/mnt/s1/zeitgeist\
2006-11-21 18:36:49 msecs elapsed at done.: 50, total: 50

d:\users\cbdqok\_privat\java\osj>_
```

- Now set up the path to the datafiles:



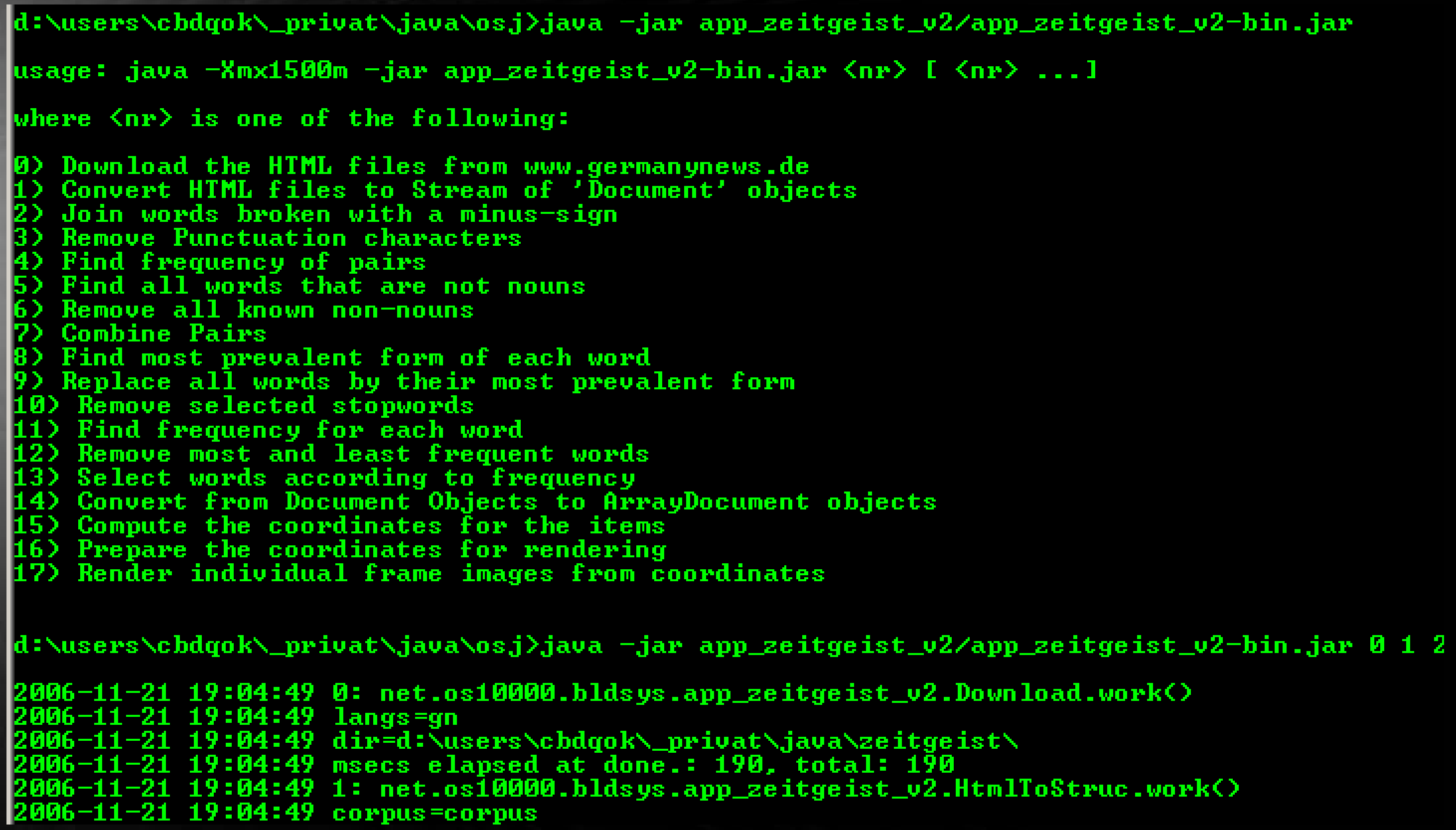
```
$ emacs app_zeitgeist_v2.properties
```



```
net.os10000.bldsys.app_zeitgeist_v2.Server.cpu_increment=3
net.os10000.bldsys.app_zeitgeist_v2.Server.data_dir=/mnt/s1/zeitgeist
net.os10000.bldsys.app_zeitgeist_v2.Server.end_year=2007
```

Operating the System

- Invocation is fairly easy ...

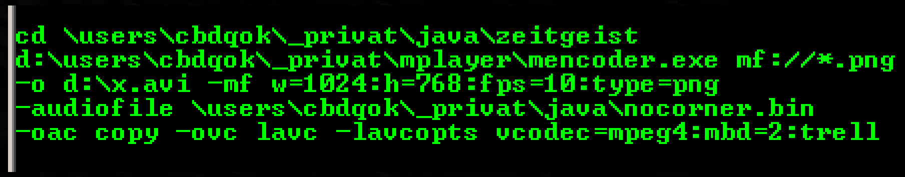


```
d:\users\cbdqok\_privat\java\osj>java -jar app_zeitgeist_v2/app_zeitgeist_v2-bin.jar

usage: java -Xmx1500m -jar app_zeitgeist_v2-bin.jar <nr> [ <nr> ...]

where <nr> is one of the following:

0) Download the HTML files from www.germanynews.de
1) Convert HTML files to Stream of 'Document' objects
2) Join words broken with a minus-sign
3) Remove Punctuation characters
4) Find frequency of pairs
5) Find all words that are not nouns
6) Remove all known non-nouns
7) Combine Pairs
8) Find most prevalent form of each word
9) Replace all words by their most prevalent form
10) Remove selected stopwords
11) Find frequency for each word
12) Remove most and least frequent words
13) Select words according to frequency
14) Convert from Document Objects to ArrayDocument objects
15) Compute the coordinates for the items
16) Prepare the coordinates for rendering
17) Render individual frame images from coordinates


d:\users\cbdqok\_privat\java\osj>java -jar app_zeitgeist_v2/app_zeitgeist_v2-bin.jar 0 1 2

2006-11-21 19:04:49 0: net.os10000.bldsys.app_zeitgeist_v2.Download.work()
2006-11-21 19:04:49 langs=gn
2006-11-21 19:04:49 dir=d:\users\cbdqok\_privat\java\zeitgeist\
2006-11-21 19:04:49 msecs elapsed at done.: 190, total: 190
2006-11-21 19:04:49 1: net.os10000.bldsys.app_zeitgeist_v2.HtmlToStruc.work()
2006-11-21 19:04:49 corpus=corpus
```

- ... you may want to give some of the steps enough memory (especially step 16)

Building the movie file

- After step 15 has produced a 700mb text file with the coordinates ...
- And after the machine has managed to have step 16 load this file into RAM ...
- And if the disk has not filled up under the 4500 .png images ...
- I use the dependable „mplayer" package to combine the frames and the mp3-soundtrack into an avi-container:

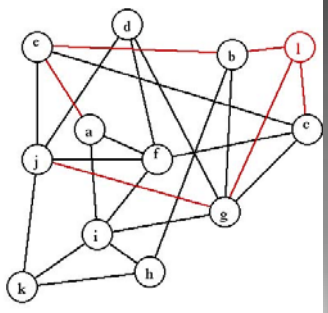


```
cd \users\cbdqok\_privat\java\zeitgeist
d:\users\cbdqok\_privat\mplayer\mencoder.exe mf://*.png
-o d:\x.avi -mf w=1024:h=768:fps=10:type=png
-audiofile \users\cbdqok\_privat\java\nocorner.bin
-oac copy -ovc lavc -lavcopts vcodec=mpeg4:mbd=2:trell
```

Sections within this Talk

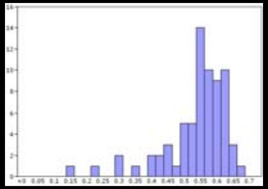
- Conception and Motivation
- Demonstration
- Architecture
- Computational Features
- Example Run
- Artistic Design Decisions
- Interpretation of the results
- Outlook & Future Experiments
- Credits and Sources

Artistic Decisions: Algorithm

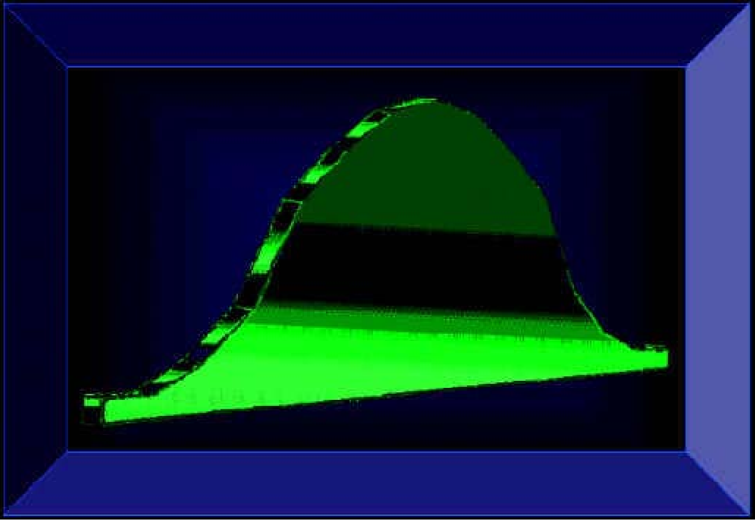


- Visualisation of co-occurence frequencies as weighted graph

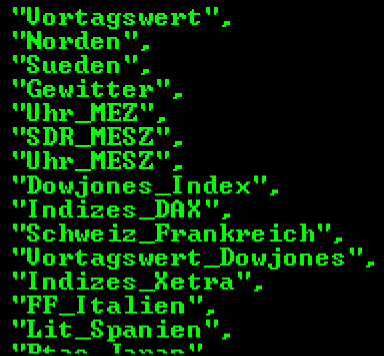
- Choice of word focus through weighted individual occurrence count



- Time indication through scaling with a moving Gauss-curve



Artistic Decisions: Linguistics



```
"Vortagswert",
"Norden",
"Sueden",
"Gewitter",
"Uhr_MEZ",
"SDR_MESZ",
"Uhr_MESZ",
"Dowjones_Index",
"Indizes_DAX",
"Schweiz_Frankreich",
"Vortagswert_Dowjones",
"Indizes_Xetra",
"FF_Italien",
"Lit_Spanien",
"Ptas_Japan"
```

- Choice of Nouns only
- Choice of Stop Words
- Pairing threshold of 30%:
  $P(A|B)>30\%$ or $P(B|A)>30\%$
- Ordering of Steps
- Choice of range to remove from most frequent side (200)
- Choice of range to use for layout (2000)

Artistic Decisions: Layout

- Bell width is 300 days, less is too hectic, more ignores short-lived events.
- The distance matrix is tamed so that pow(dist_max,f)/pow(dist_min,f)<2.
- Coordinates are distorted through a rectangular fish-eye lens.

- Importance is shown in font-colour and font-size, icon scale and transparency are chosen quadratically.

Artistic Decisions: Candy

- Eye Candy
  - Choice of Typeface
  - Choice of Background
  - Choice of Framing
  - Choice of Image search engine

picsearch™

- Ear Candy (you don't want to try this)
  - Choice of soundtrack

Sections within this Talk

- Conception and Motivation
- Demonstration
- Architecture
- Computational Features
- Example Run
- Artistic Design Decisions
- Interpretation of the results
- Outlook & Future Experiments
- Credits and Sources

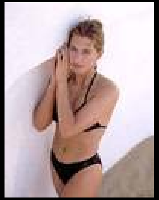
Aesthetics & Ontology (1/3): Words

- The representation is faithful in the sense that the frequency distribution matches what we got in the media.
- Both, positive and negative association brings a subject close to its object, so the word „labour" is associated with both the words „union" and „employer", but in quite different ways.
- A proper solution to this problem is „AI-hard".

Aesthetics & Ontology (2/3): Images

- The problem of interpretation is even more apparent when representing words through images
- Choosing the first image a search engine presents not always yields desirable results:
  - beef – shows a „fricassee" of sorts

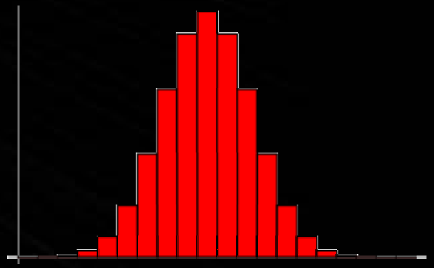
  - labour strike – „pink T" logo
  - Steffi Graf – lightly clothed model



  - Tour de France – split in two tire frame („tour") & map of France
- Getting a cabbage for „Kohl" prompted the inclusion of pairing („Bundeskanzler Kohl")

Aesthetics & Ontology (3/3): Recognition

- An interesting insight was the comparison between expected and measured co-occurrence frequencies:
  - „Terror" less frequent than expected
  - „Taxes" less frequently than expected
  - „Iraq" less frequently than expected
  - Football-related terminology less frequently than expected

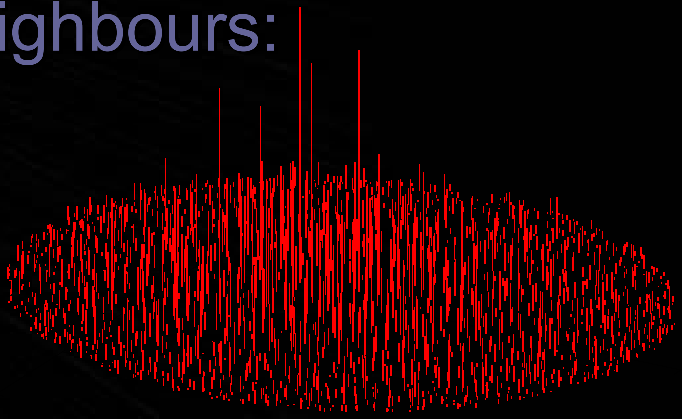


Sections within this Talk

- Conception and Motivation
- Demonstration
- Architecture
- Computational Features
- Example Run
- Artistic Design Decisions
- Interpretation of the results
- Outlook & Future Experiments
- Credits and Sources

Future Experiments

- Use of different corpus of documents.
- Use of majority voting between search engines on pictures.
- First steps in semantic representation through facilities like WORDNET, or search engine statistics.
- Boosting of particular words, -- and through the connections their neighbours:



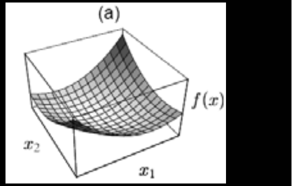
Sections within this Talk

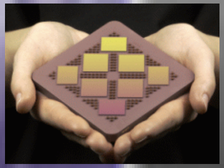
- Conception and Motivation
- Demonstration
- Architecture
- Computational Features
- Example Run
- Artistic Design Decisions
- Interpretation of the results
- Outlook & Future Experiments
- Credits and Sources

A great many thanks to ...

- My wife's patience with my arcane hobby

- AT&T research: the excellent algorithm
- J.R.Shewchuk „An Introduction to the Conjugate Gradient Method Without the Agonizing Pain"





- IBM: letting me use an OpenPower 8cpu
- Ebay: selling me an IBM Netfinity (4x .7GHz 2mb cache, 2gb RAM, €151,--)

- Countless java package authors
- William Gibson for inspiring visions of the matrix

Downloads

- You can find the software at:

  http://www.os1000.net/fs/java/app_zeitgeist_v2/index.html

- I kindly ask that you download the corpus pack from my site and only fetch an update from the News server.

  http://www.os10000.net/fs/java/app_zeitgeist_v2/corpus.zip

- You can find the movie at:

  http://www.os10000.net/fs/java/app_zeitgeist_v2/movie_wds.avi
  http://www.os10000.net/fs/java/app_zeitgeist_v2/movie_ico.avi

The End

- Thank you for your time and attention.