cs6964 | February 14 2012

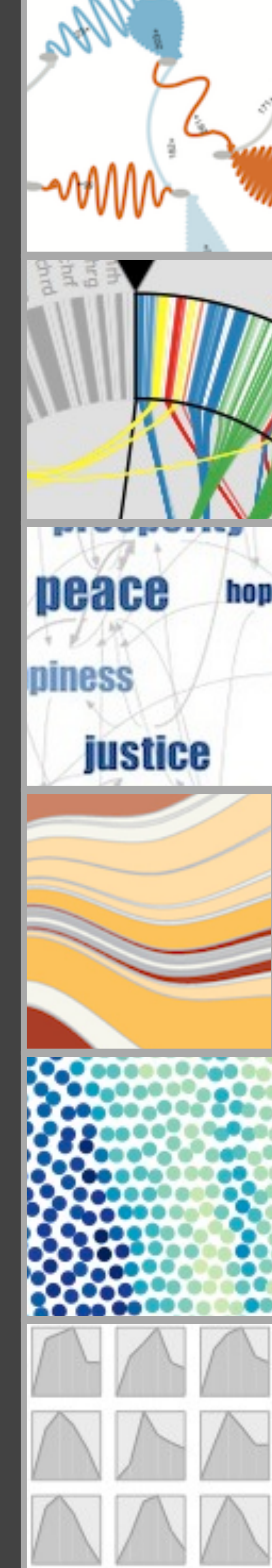
FILTERING & AGGREGATION

Miriah Meyer

University of Utah

slide acknowledgements:

Tamara Munzner, University of British Columbia



administrivia

feb 14-23 : proposal meetings

march 7 : presentation topics due

march 9 : proposals due

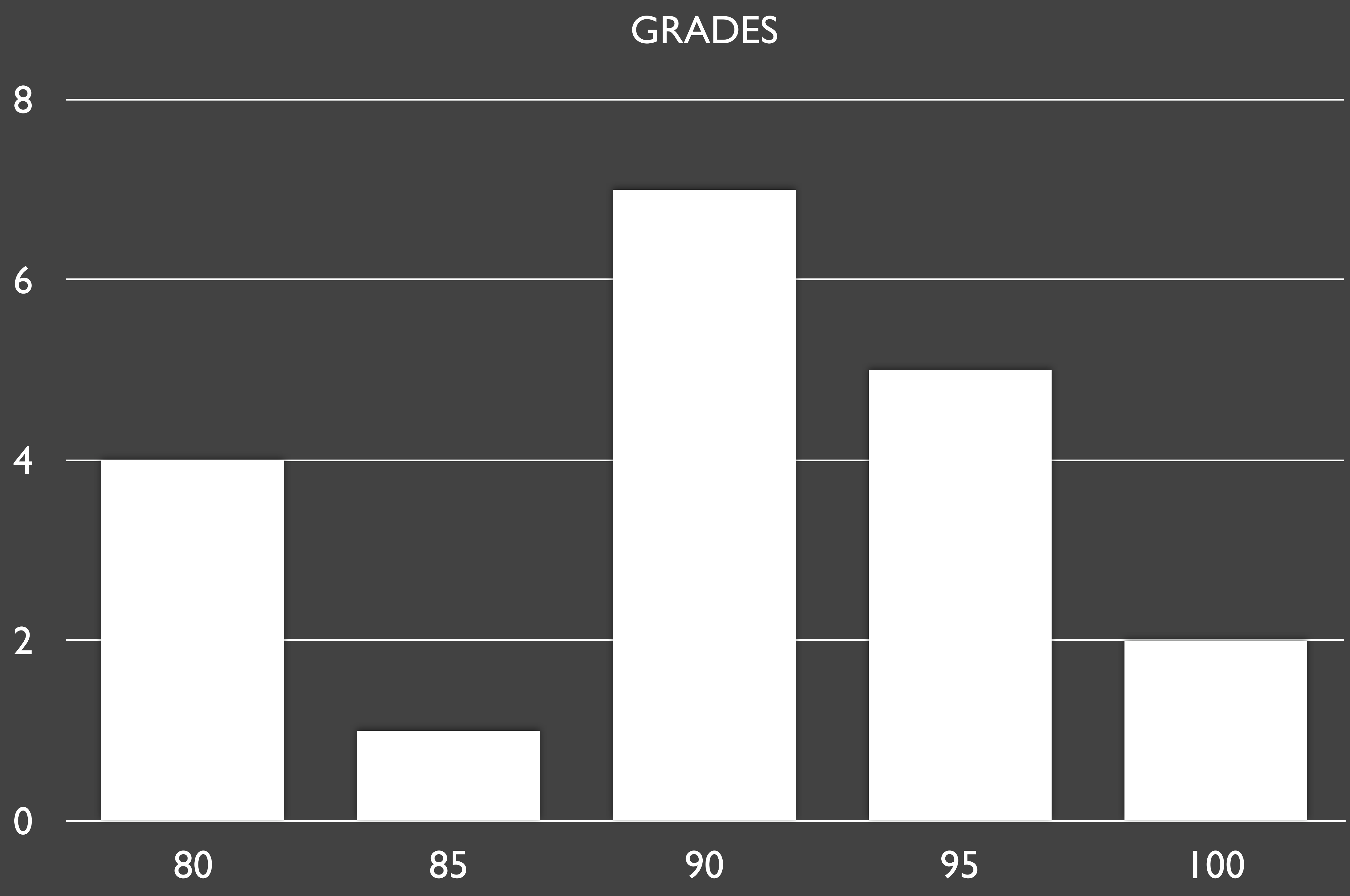
march 27-april 3 : project updates

april 5-24 : paper presentations

may 1 : final project presentations

may 3 : process books due

THE PANCAKE RECIPE CHALLENGE


GRADES
8
6
4
2
0
80
85
90
95
100

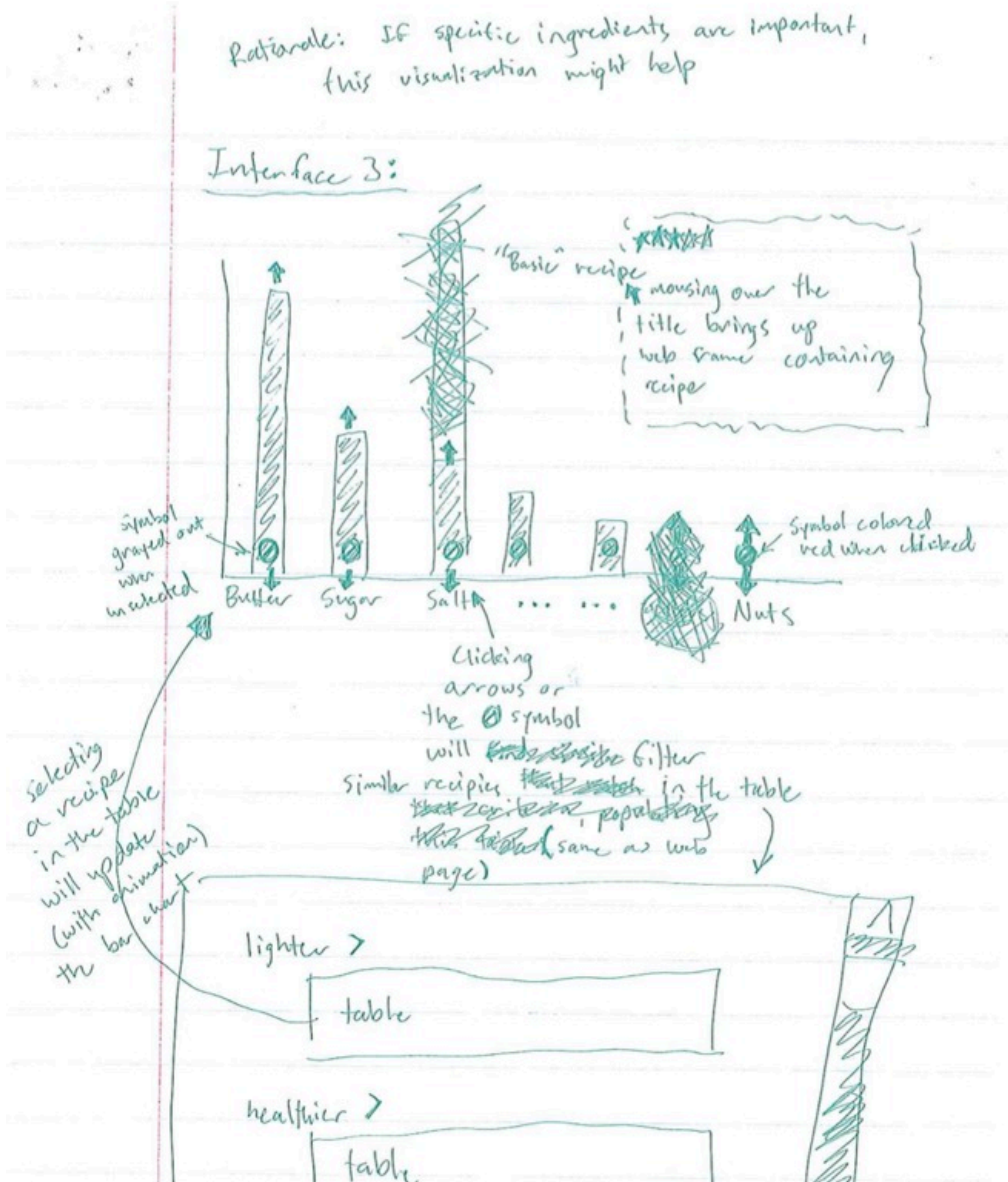
Question: What are the most common extensions to the basic pancake recipe?

Alex Bigelow

Data:	Automatic (non-visual) Transformations:	Derived Data:
- Web Text		
- ratings →	Normalization (eg 5 stars vs scale 1-10) →	Ordinal table of ratings
- descriptions/ comments/ reviews →	Natural Language Processing (eg find words like "low-calorie", "dry", "thick", "moist", "light") →	Categorical table of tags, Quantitative table of # of people that tried it
- ingredients →	Natural Language Processing, Normalization for # of servings →	Quantitative table of ingredients
	Machine Learning, PCA, etc. to find clusters of recipes with similar ingredient distributions →	Categorical table indicating cluster
	Select most popular recipe within each cluster - these are the "basic" recipes, Also select most common tags for each cluster →	Categorical table indicating a recipe's "root" recipe, and its descriptive tags

network?

Alex Bigelow



Jeffrey Jestes

The data types we will work with include tabular and text data. The rows in the table will correspond to a particular type of pancake. The columns will include:

1. Name of the pancake	categorical data attribute
2. Type of ingredient	categorical data attribute
3. Quantity of ingredient	sequential quantitative, abstract, static, distinct
4. Unit of measure	categorical data attribute
5. URL to recipe	text/label data

Since ingredients for pancakes are finite, and probably sufficiently small, maintaining a column per possible ingredient will probably not be an issue. However, we can also store in a single column a compound data list type which includes all ingredients as well as the quantity of each.

In order to answer the challenge question, we must be able to ask if a pancake has a subset of the ingredients in our pantry and if for each ingredient if we have enough in our pantry to meet the ingredient quantity requirements of the recipe. The primary visualization task necessary is the filter task. We apply the first filtering technique which first checks if a pancake recipe only lists types of ingredients which we have in our pantry, and add these candidate recipes to a set S. The second filtering looks at each pancake in set S and determines if for each required ingredient, if we have enough of that ingredient in our pantry, and if so adds the pancake to result set R. After obtaining R we may wish to sort the pancakes in R by total number of ingredients, or total quantity of ingredients, or some other user specified function.

Potential data transformations include the construction of an inverted index over ingredients. Treating the type of ingredient as a token, each token in the inverted index points to the set of pancake names which use the ingredient. In addition each pancake name could also be associated with the quantity necessary for the ingredient. To obtain candidate pancakes, we query the inverted index for each ingredient we have in our pantry. After finding candidate pancakes the problem boils down to determining if all ingredients in the candidate pancakes appear in the pantry. Since (homemade) pancakes do not typically have many ingredients this can be accomplished in a linear scan of ingredients for each pancake. Efficient string matching techniques using q-grams might also be considered to prune some pancakes from consideration. After finding pancakes with a subset of the ingredients in the pantry, the unit of measure of the required ingredient might be transformed to match that of the unit provided by the user. A simple comparison is used on the quantity of each ingredient to determine if we have enough.

The main classes of change which could be used for this analysis task include changing selection, changing highlighting, and changing view. The changing selection can be used to select and deselect pancake results. Changing highlighting is useful to show pancakes which satisfy the ingredient requirements, the pancakes which do not may be presented but not highlighted.

Jeffrey Jestes

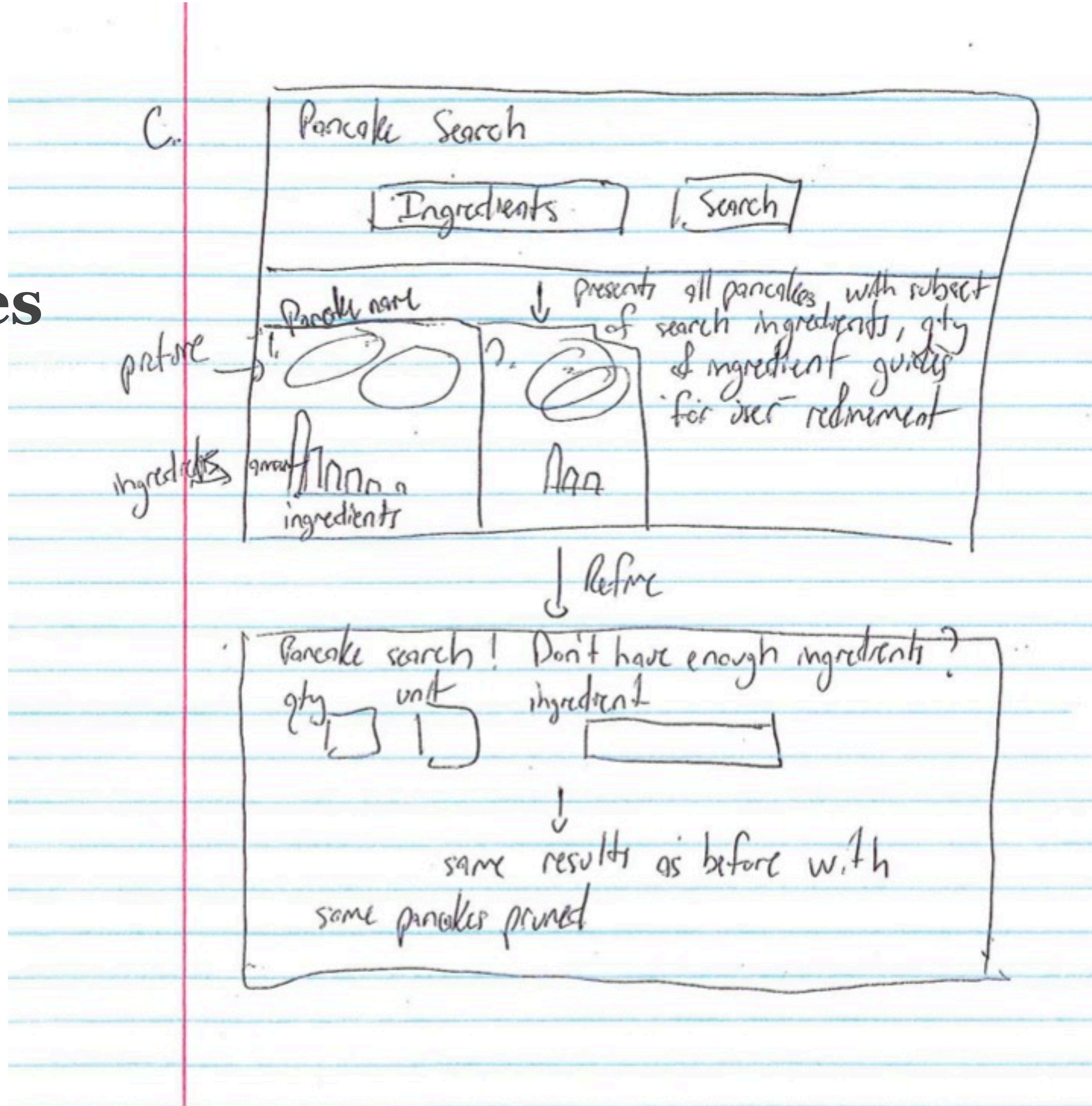
The primary mark types I have considered for my design are lines and bars, for line and bar style charts indicating quantity of ingredients. The primary visual channels are length and color highlighting for indicating quantity of ingredient as well as pancakes which contain all ingredients respectively. The primary mechanism interactions are: 1. user input forms, which allows users to change selection by inputting ingredients; 2. Selection of pancakes in results through mouse-over and clicking; 3. Scrolling through the results with mouse-over to obtain a synopsis of information before selecting a pancake; 4. Sorting results based on some user specified function, e.g. quantity of ingredients, total distinct ingredients, etc.

My favorite design is C on the attached sketch paper. The primary markers are bars to indicate quantity of each distinct ingredient appearing in a pancake. There are also images of the pancake to drive user interaction and images and the ingredient bar charts are grouped. This design requires users to specify all ingredients and no quantities. This comes from reasoning that a user may what ingredients are in the pantry, but not precisely how much. After finding an appealing pancake(s) they can further refine and filter by entering quantities. Otherwise they can manually inspect the ingredient list.

In brief:

1. Bar charts — allow users to quickly check quantities of ingredients
2. Grouped image — groups image of pancake with name and bar chart to appeal to user
3. Interactive ingredient search — specify ingredient without quantity
4. Refined search — specify quantities or maximum amount of ingredient
5. Scrolling capability — user can scroll through results
6. Sorting — allow user to sort based on user defined function
7. Highlighting — potential after refined search, highlight pancakes which match

Jeffrey Jestes



Nikhil Mishrikoti

Abstraction :

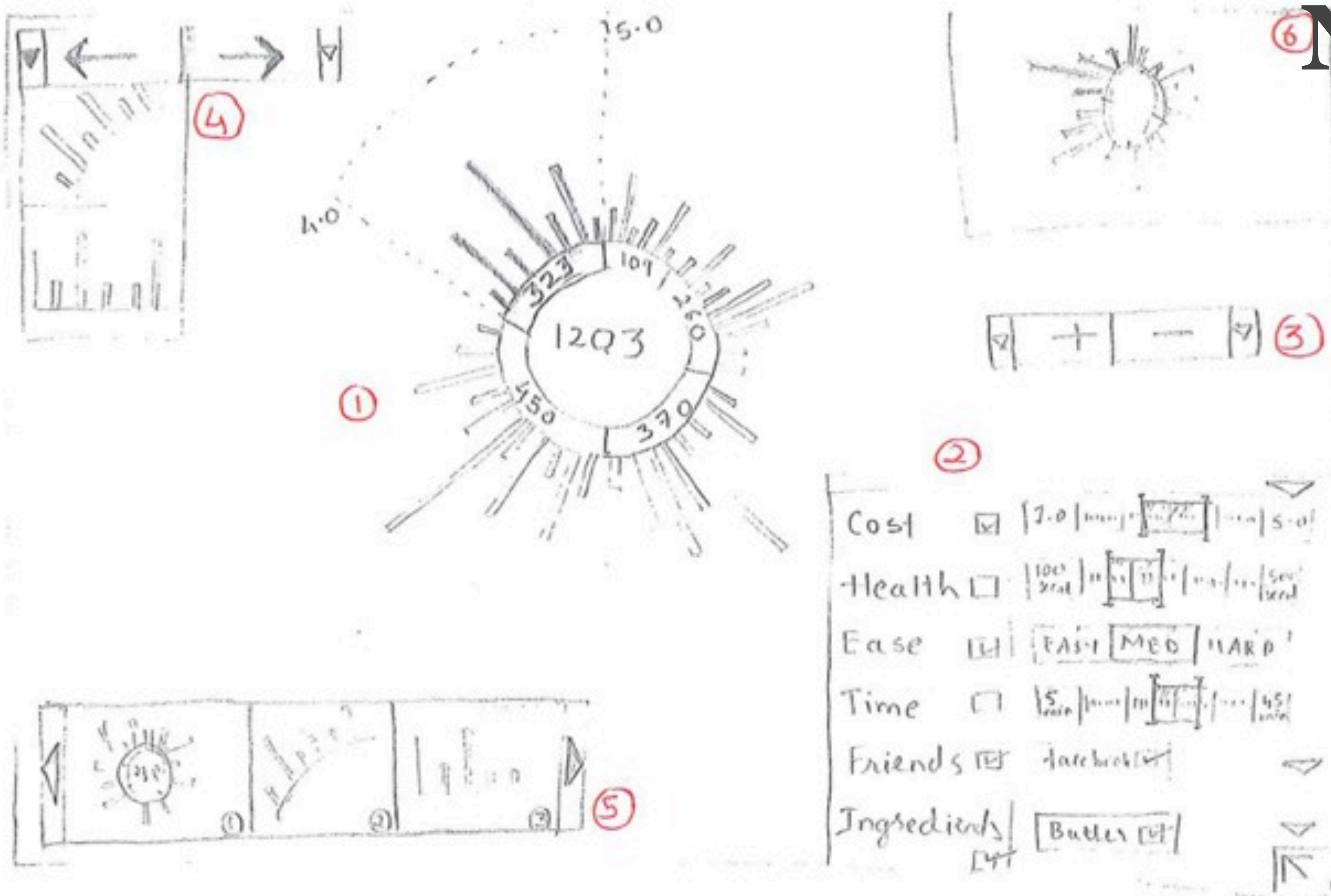
* Data types and their attributes
  - Ingredients - Tabular type

    ex [2 tsp sugar , salt]
    → categorical
    → quantitative

  - Recipe Instructions - Ordinal type ex : 1, step 2, step ...]
  - Overal Rating - Ordinal
    : 3/5
  - Cost : Ordinal [if rating . ex : ... : 2/5]
    Quantitative [if $: ex $30 , $40 ...]
  - Health : Quantitative [in kcal ex 190 kcal / serving]
  - Ease of Preparation : Ordinal [if rating or classes
    ex : Easy , Medium , Hard]
  - Time : Quantitative [ex: 15 mins , 45 mins]
  - Recommendations : Categorical [ex facebook, twitter]
  - User tags : Categorical [ex: lowfat , vegan ... etc].

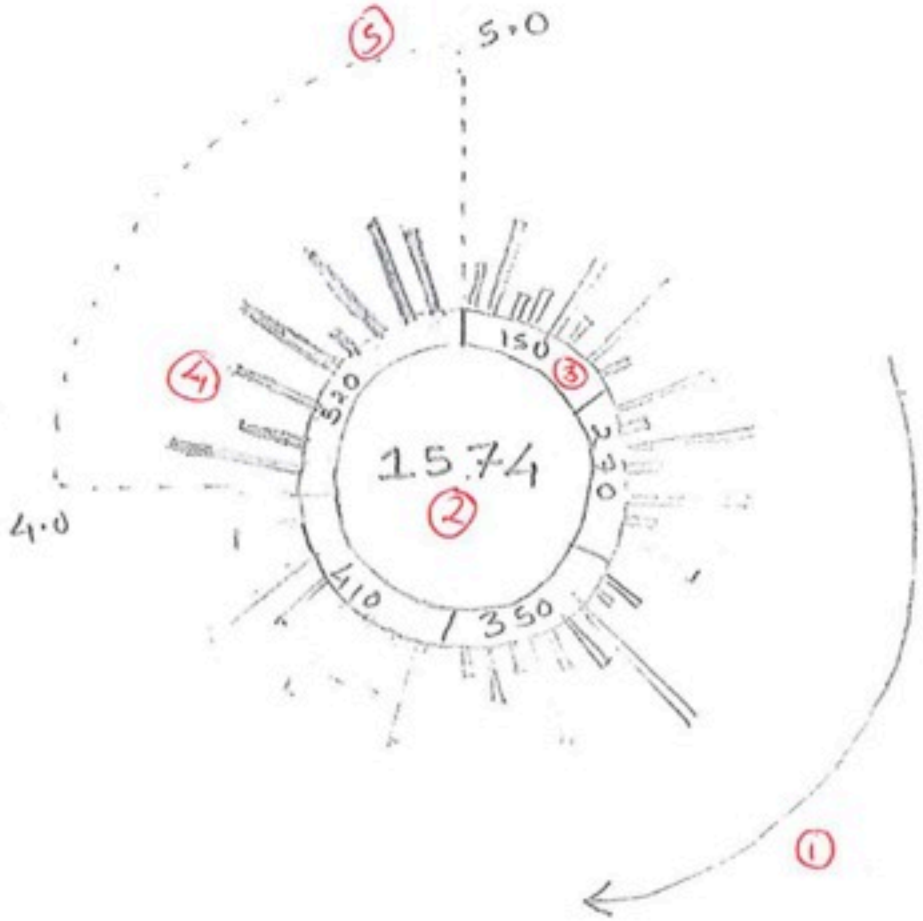
SUN*: Ordinal data based exploratory tool

Nikhil Mishrikoti



1. SUN - Pie-Bar graph visualization of Recipes
2. Ordinal data Selection / Filtering menu
3. Zoom level adjustment
4. View Selection - Prev / Next [Past / Latest]
5. View Selection - Up / down [stages in hierarchy]
6. Overview of filtered data in SUN

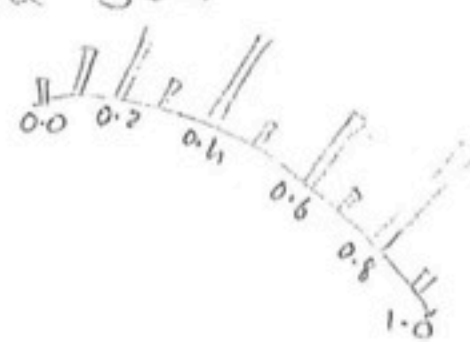
Nikhil Mishrikoti



All recipes available to us are arranged by their overall rating, from 0 → 5.00 ex: 3.9/5, 4.5/5 etc

1. Ratings increase in clockwise direction along the circumference of the SUN.

ex:



2. Indicates the total amount of recipes being viewed across all ratings.

Nikhil Mishrikoti

(3) - Number Indicates the number of recipes in that rating range.

ex: 150 recipes with rating between 0.0 & 1.0

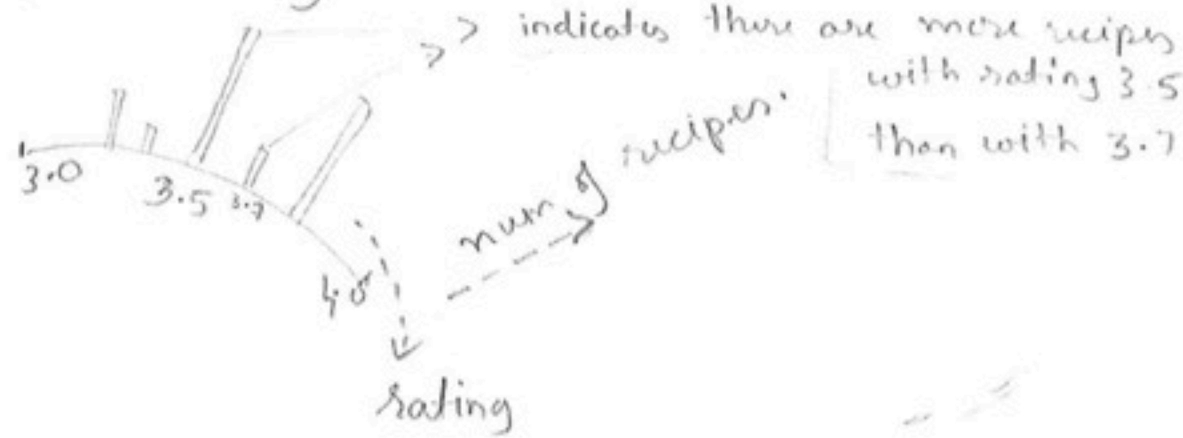
- Length of arc proportional to this number

ex: Length $\propto$ 150/1574

(4) - Length of the bar graph indicates the number of recipes with rating indicated on the circumference.

- Does not represent accuracy, indicates shape to find patterns of interest

ex:



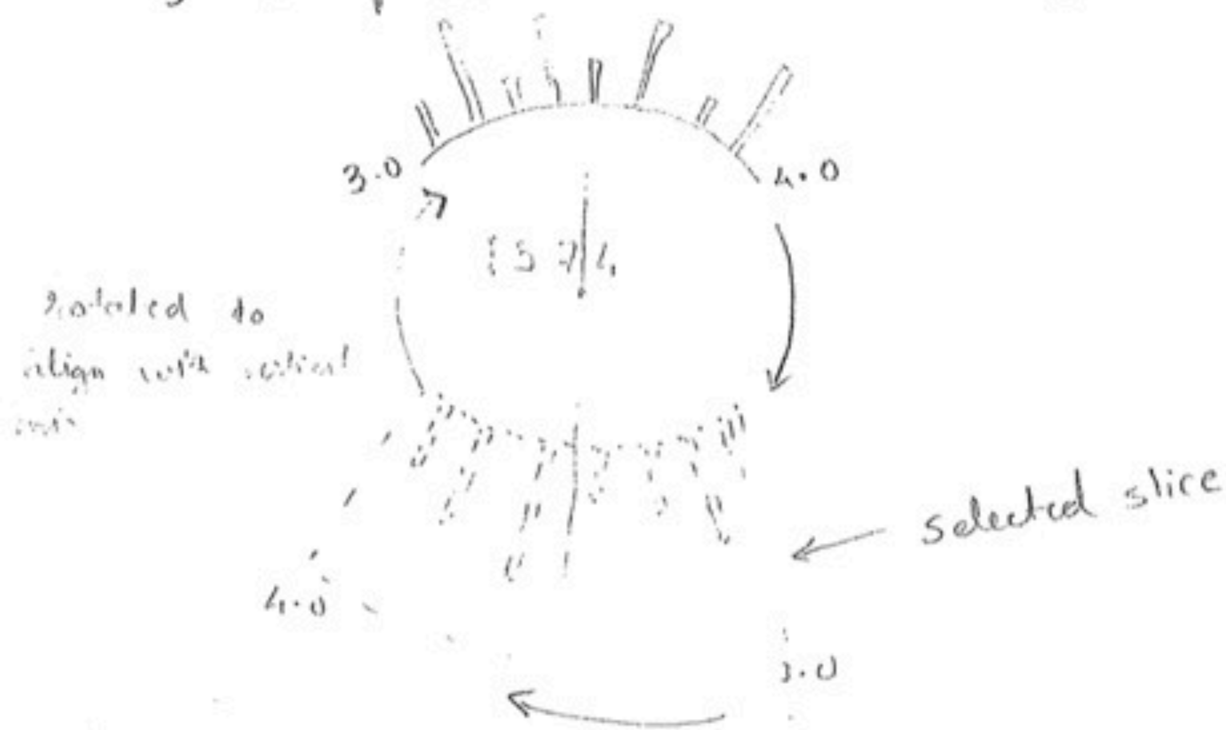
(5) - Shaded pie-bar graph indicates selection during mouse hover.

- Clicking on the region deepens or proceeds

Nikhil Mishrikoti

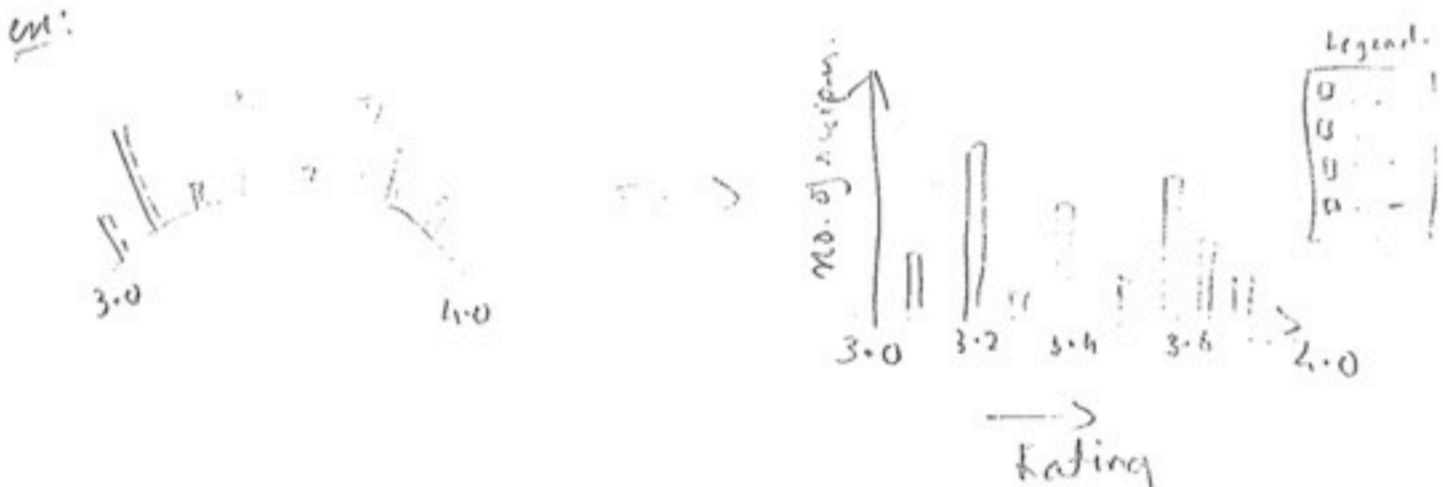
(1.1) Clicking on the pie-bar slice in the SUN

- overview: the slice is rotated to align with vertical axis (animation to preserve context) & then proceeds to the zoom step.

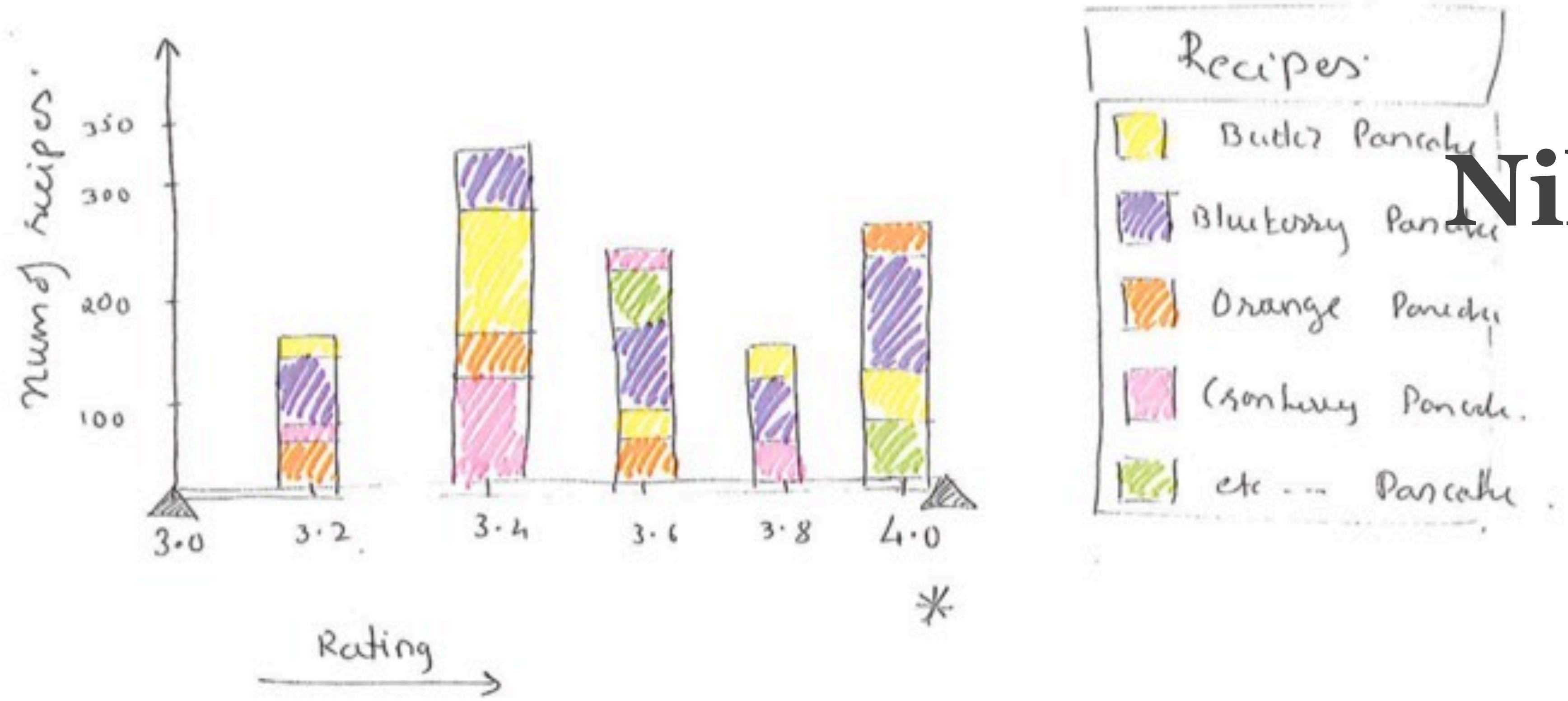


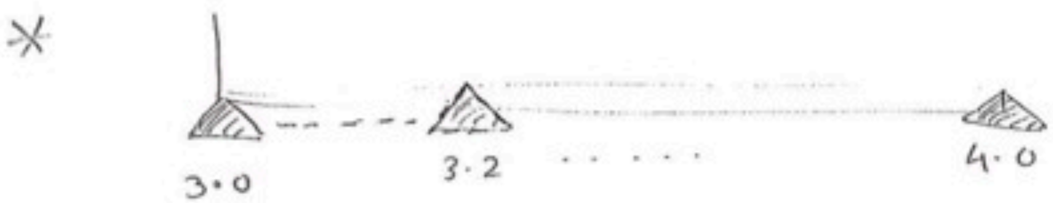
- zoom: The bar graph is then zoomed into slowly (animation) & the axes are straightened out, with a legend menu showing recipes.

ex:

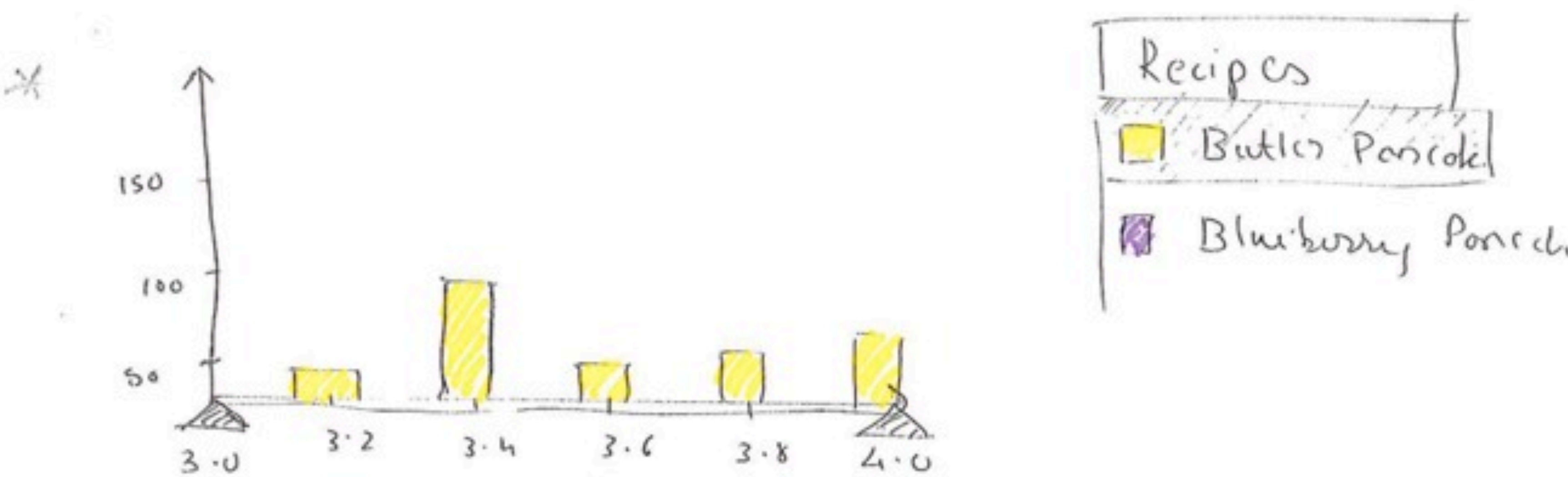


Nikhil Mishrikoti



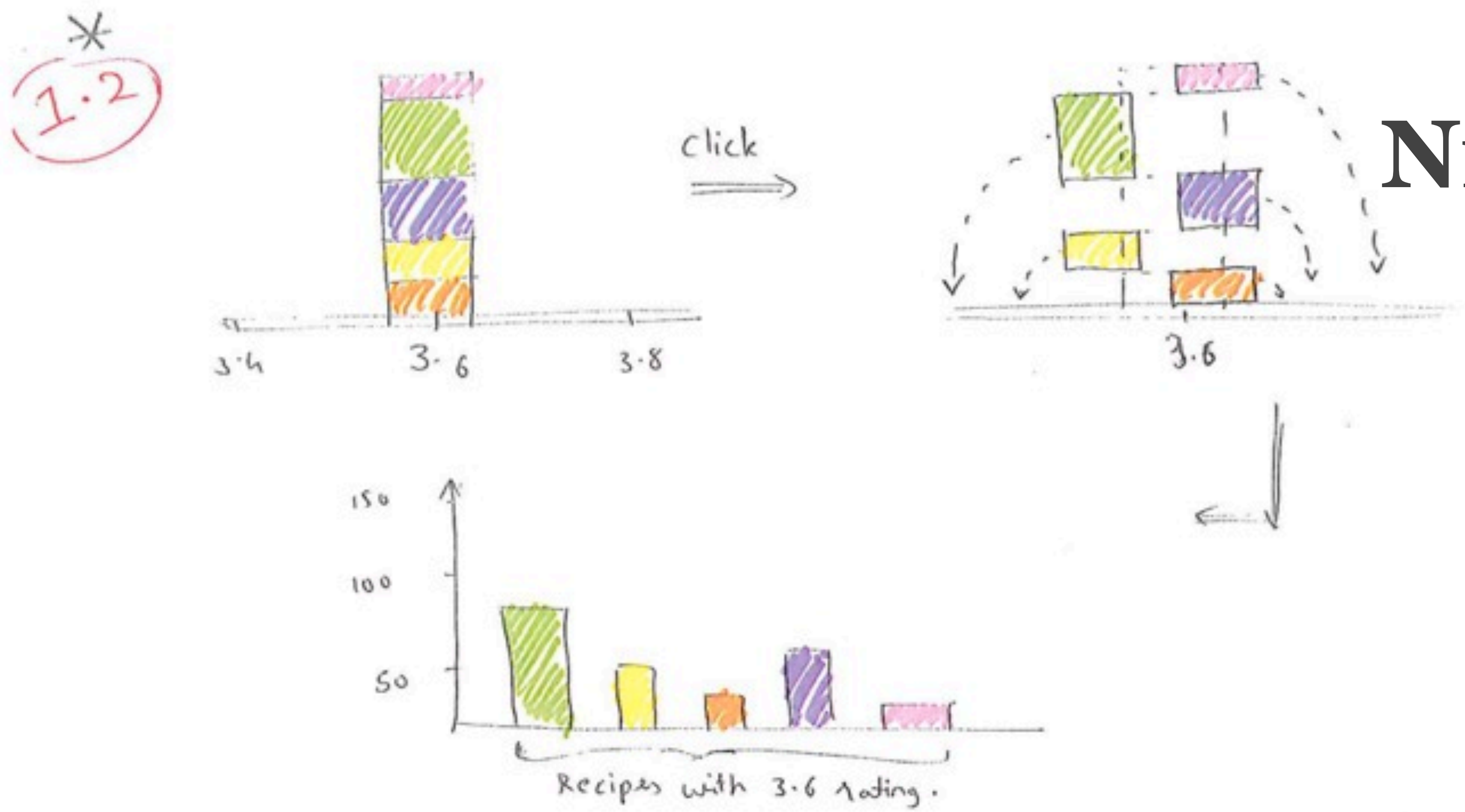


arrows used to change ratings range on x-axis if graphs are too crowded.



if a particular recipe chosen from menu, then data

Nikhil Mishrikoti



- Clicking on a bar, again follow the overview of zoom & filter procedure.
- Overview & zoom: The colored blocks slowly move away (animation to preserve context) & form a new bar graph.
- filter: The bars are arranged by recipe names/colors shown in menu.

LAST TIME

LINKED VIEWS

multiple views that are simultaneously visible and linked together such that actions in one view affect the others

- **linking choices**
  - linked highlighting
  - linked navigation

- **view choices**
  - encoding: same or multiform
  - dataset: same or small multiple
  - data: all or subset (overview/detail)
  - conditioning

-today

comments on readings?

DATA REDUCTION

- **how to reduce the amount of stuff to draw?**
  - crosscuts view composition and interaction considerations
- **item reduction**
  - today
  - rows of table
- **attribute reduction**
  - Thursday
  - columns of table

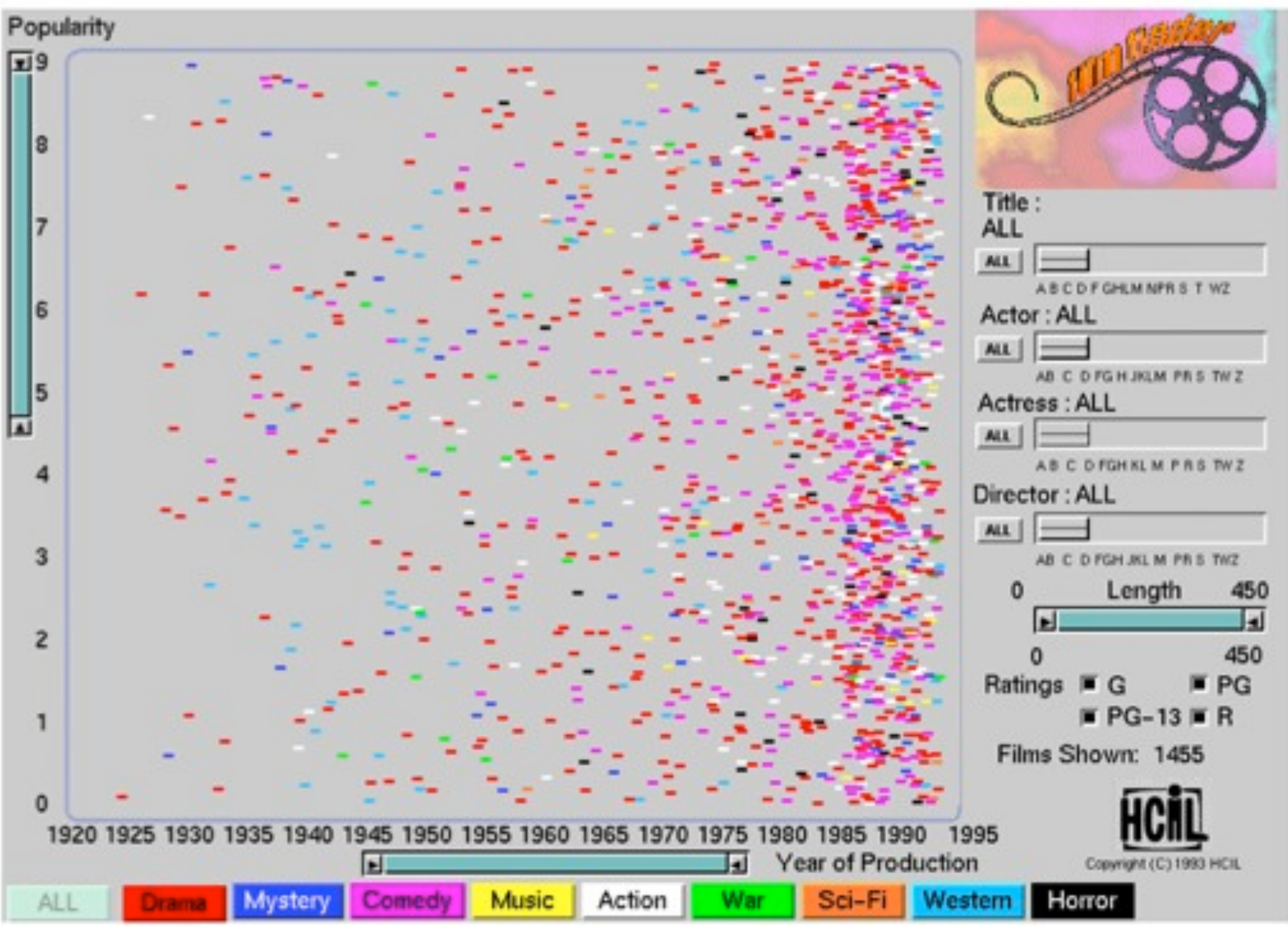
ITEM REDUCTION METHODS

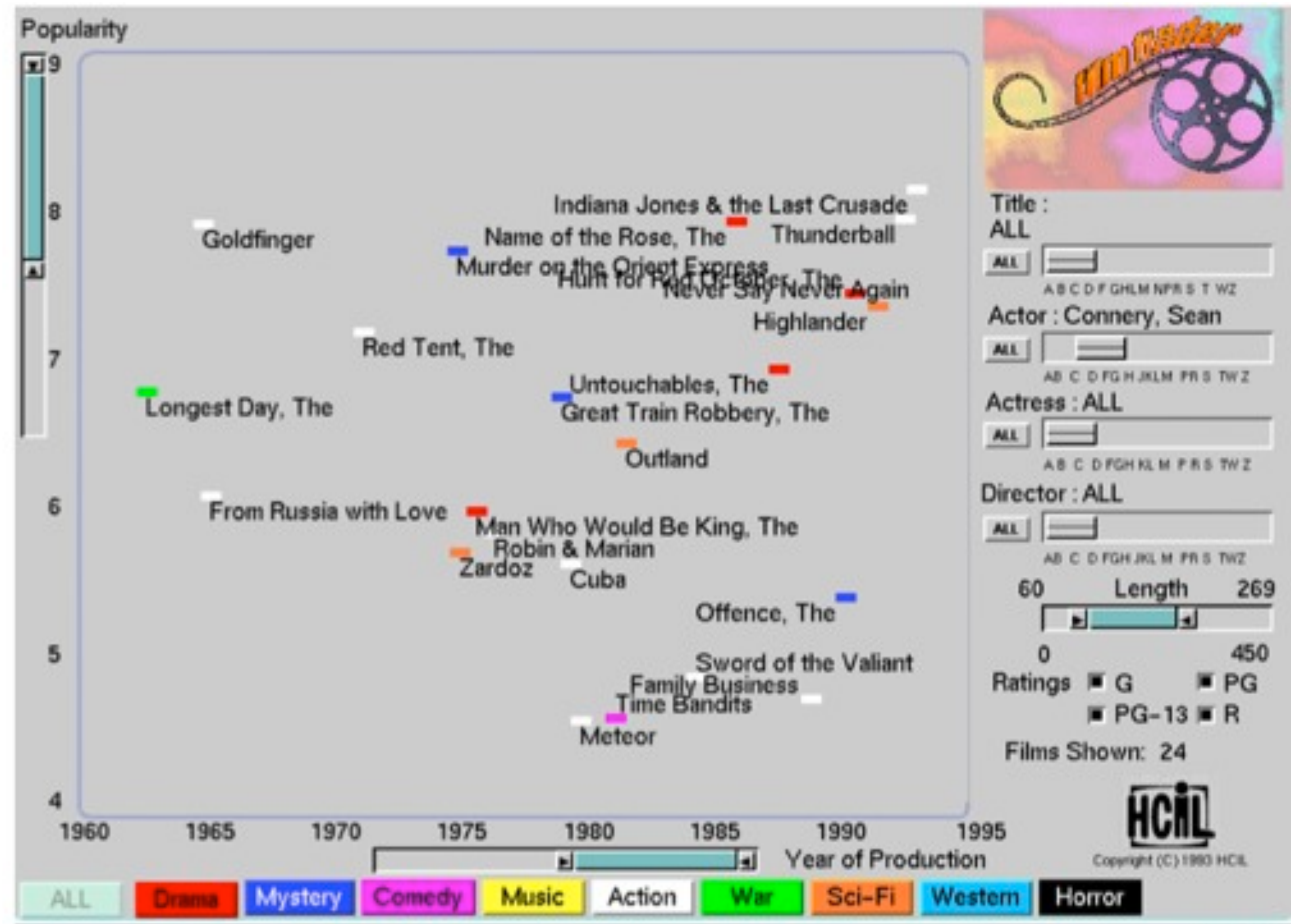
- **filtering and navigation**
  - leave some things out
- **aggregation**
  - merge things together
- **overviews**
  - temporal through navigation
  - separate dedicated view
  - focus + context
    - *selective filtering*
    - *geometric distortion*
    - *distortion costs/benefits*

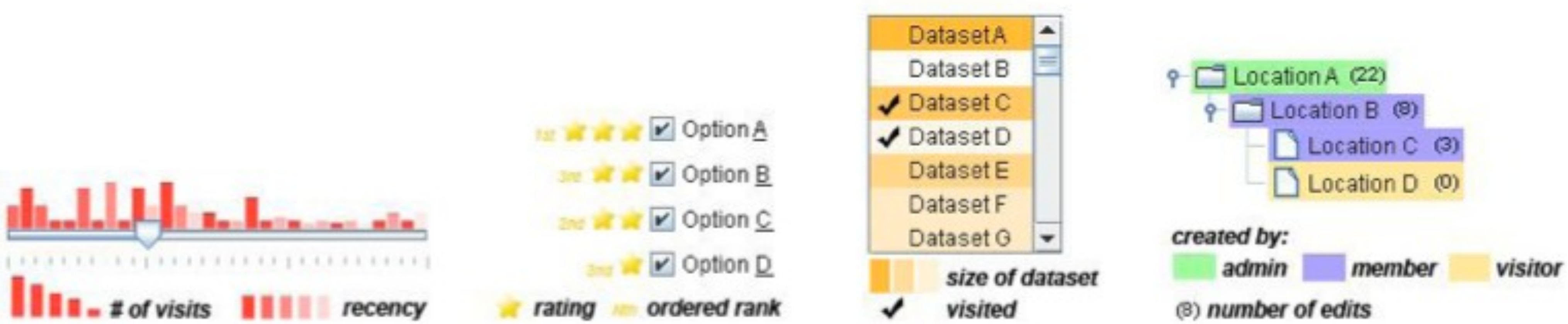
FILTERING

NONSPATIAL FILTERING

Ahlberg 1994







Willett 2007

The New York Times - Breaking

www.nytimes.com

the 6 of 106

HOME PAGE | TODAY'S PAPER | VIDEO | MOST POPULAR | TIMES TOPICS

Subscribe: Digital / Hom

The New York Times

Tuesday, February 14, 2012 Last Update: 2:58 PM ET

Search | ING DIRECT | Follow Us | Subscribe to Home Delivery | Personalize Your Weather

Switch to Global Edition »

JOBS
REAL ESTATE
AUTOS
ALL CLASSIFIEDS

WORLD
U.S.
POLITICS
NEW YORK
BUSINESS
DEALBOOK
TECHNOLOGY
SPORTS
SCIENCE
HEALTH
OPINION
ARTS
Books
Movies
Music
Television
Theater
STYLE
Dining & Wine
Fashion & Style
Home & Garden
Weddings/ Celebrations
TRAVEL

All Blogs
Cartoons
Classifieds
Corrections
Crossword / Games
Education
Event Guide
First Look
Learning Network

U.S. Seeks to Size Up China's Heir Apparent During Visit

By MARK LANDLER and EDWARD WONG 19 minutes ago

China's vice president, Xi Jinping, began a visit to the United States on Tuesday that amounts to a get-to-know-you exercise for the man who is widely seen as the next leader of China.

Post a Comment | Read (48)

Obama Keeps Up Pressure on G.O.P. Over Payroll Tax

By HELENE COOPER and JENNIFER STEINHAUER 12:22 PM ET

The president's remarks on Tuesday came a day after House Republicans said that they were ready to extend the payroll tax cut without offsetting it with other cuts.

Post a Comment | Read (43)

Syria Resumes Shelling, Rejecting U.N. Rebuke

By NEIL MacFARQUHAR and RICK GLADSTONE 10:50 AM ET

Syrian forces brushed aside a castigation from a United Nations official, resuming what one activist called "brutal

TimesCast

Nicolas Asfouri/Agence France-Presse — Getty Images

TimesCast: Bombings in Bangkok

Explosions in Bangkok | Santorum ties Romney in latest New York Times/CBS News poll | Underage models at New York Fashion Week.

· TimesCast Archive

Blasts in Bangkok Add to Suspicions About Iran

By THOMAS FULLER 40 minutes ago

A series of explosions on Tuesday led to the capture of two men identified in Thai media reports as Iranians.

Post a Comment | Read (223)

Flaw Found in an Online Encryption Method

By JOHN MARKOFF 1 minute ago

Researchers have discovered an unexpected weakness in the encryption system widely used for online shopping.

Post a Comment | Read (1)

Portugal's Debt Efforts May Be Warning for Greece

By LANDON THOMAS Jr. 12:29 PM ET

By a broad measure of a country's ability to repay its debts, Portugal is going deeper into the hole, because its economy is shrinking.

· Tracking Europe's Debt Crisis

Post a Comment | Read (8)

OPINION »

CAMPAIGN STOPS

Egan: A Widow's Wisdom

Same-sex marriage finds an unlikely G.O.P. champion.

Why Don't Americans Elect Scientists?

Scientists have never gotten a political foothold in the United States.

- Brooks: Materialist Fallacy
- Bruni: The Do-Over Derby
- Cohen: Israeli Power
- Nocera: N.C.A.A. Rules
- Editorial: 2013 Budget
- Emanuel: Let's Move
- The Loyal Opposition: The Sant

MARKETS » At 3:08 PM ET

S.&P. 500	Dow	Nasdaq
1,344.75	12,822.91	2,918.59
-7.02	-51.13	-12.80
-0.52%	-0.40%	-0.44%

GET QUOTES My Portfolios »

Stock, ETFs, Funds Go

MOVIES »

Cast Your Votes

Who will take home an Oscar this year? Start your ballot below with the Best Picture category.

Nominees...

THE CARPETBAGGER

Defending 'The Help'

Viola Davis speaks out on a mind-set that she says harms black actors.

'70s Sex Star Fascinates a New Era

"Lovelace" depicts the life of Linda Lovelace, the star of a 1972 pornographic film.

REAL ESTATE | AUTOS | JOBS | ALL CLASSIFIEDS

TELEVISION

The British Are Coming

NEW CARS SEARCH

SPATIAL FILTERING: NAVIGATION

- **filter based on spatial position**
  - unconstrained: camera can move anywhere
  - constrained: limit on possible motion
- **panning | translating**
- **rotating (in 3D)**
- **zooming**
  - geometric: analogous to real-world
    - *appearance fixed, viewpoint changes sizes of objects*
  - semantic: representation adopts to screen space

AGGREGATION

- combine items (vs. eliminate them with filtering)
- derived attributes
  - min | max | avg | sum
  - many many many other ways!
- challenge: avoid averaging out signal

OVERVIEWS

- **strategies: both filter and aggregate**
  - simple: geometric zoomout
  - complex: aggregation
- **methods**
  - temporal through navigation
  - separate dedicated view
  - embedded/integrated focus + context

- **taxonomy**
  - overview + detail: spatial separation
  - zooming: temporal separation
  - focus + context: integrated/embedded

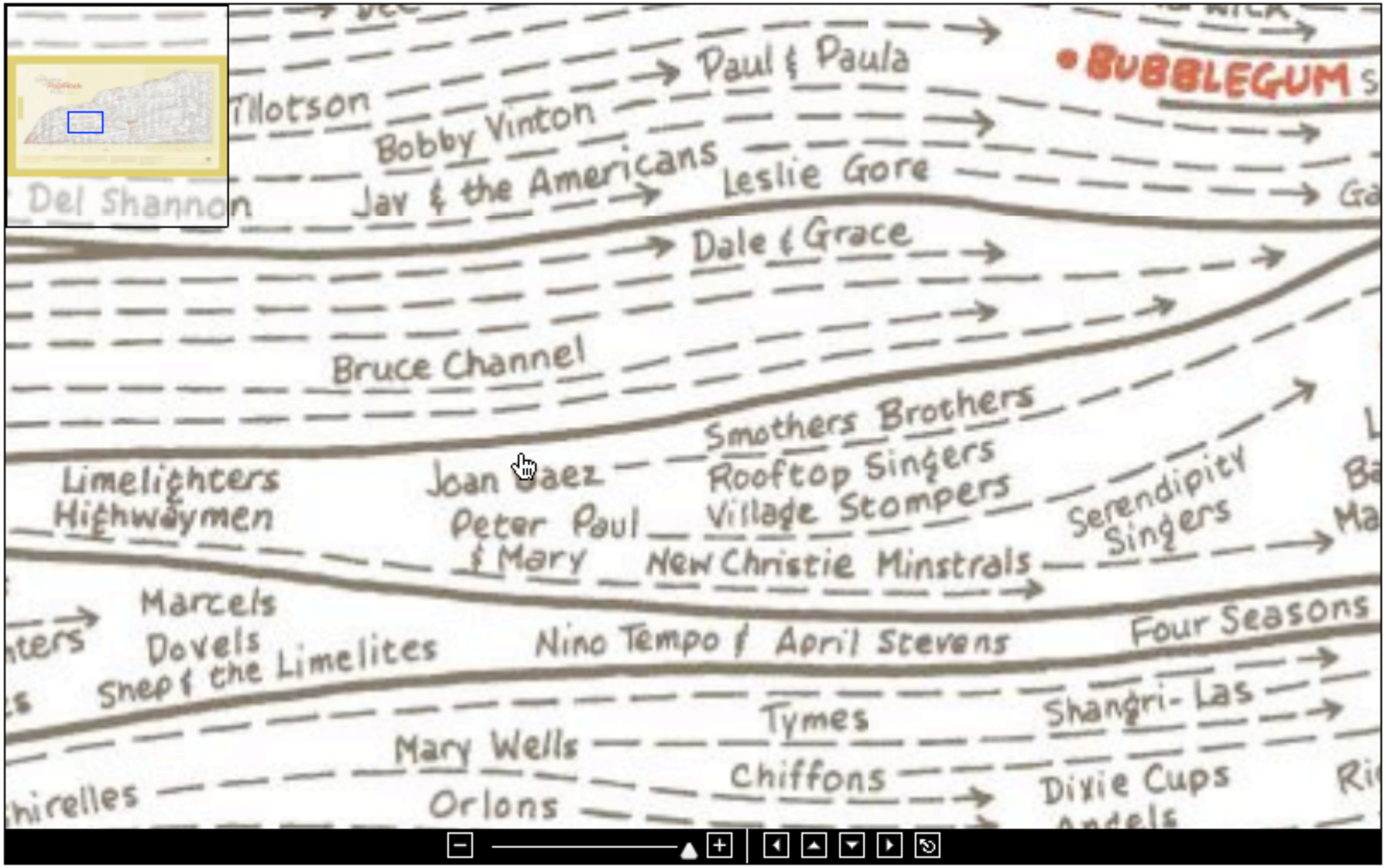
  - cue-based: selectively highlight/suppress
    - *crosscuts other techniques*

OVERVIEWS
overview + detail

OVERVIEW + DETAIL

Cockburn 2008

OVERVIEW + DETAIL



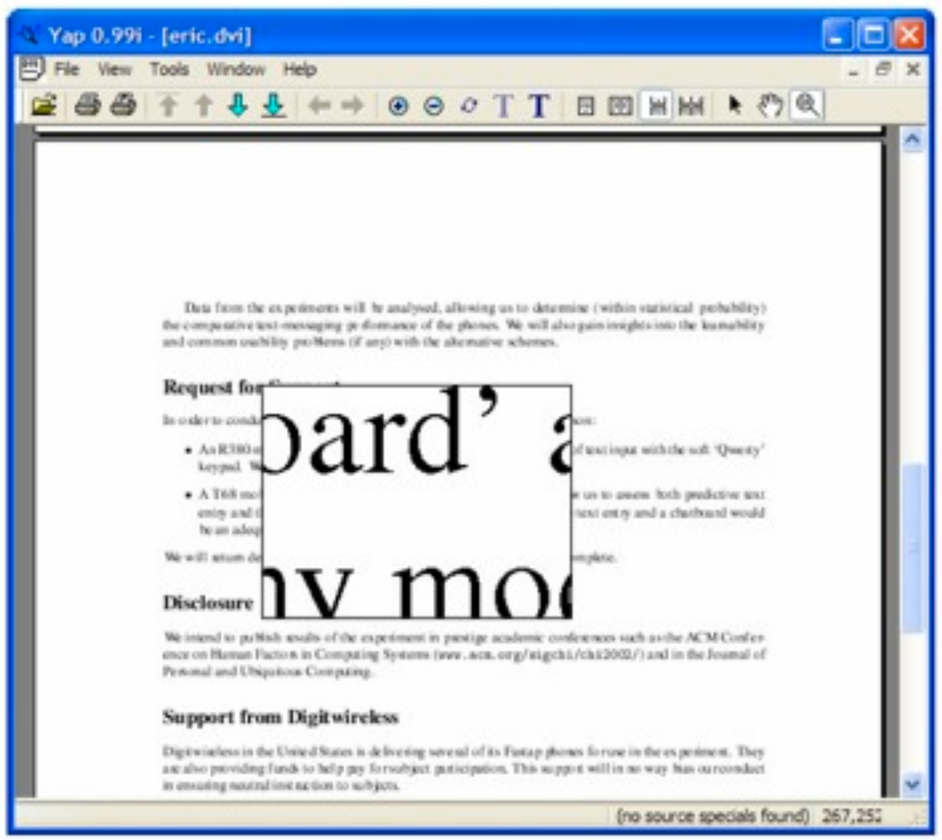
http://www.historyshots.com/rockmusic/

OVERVIEW + DETAIL ISSUES

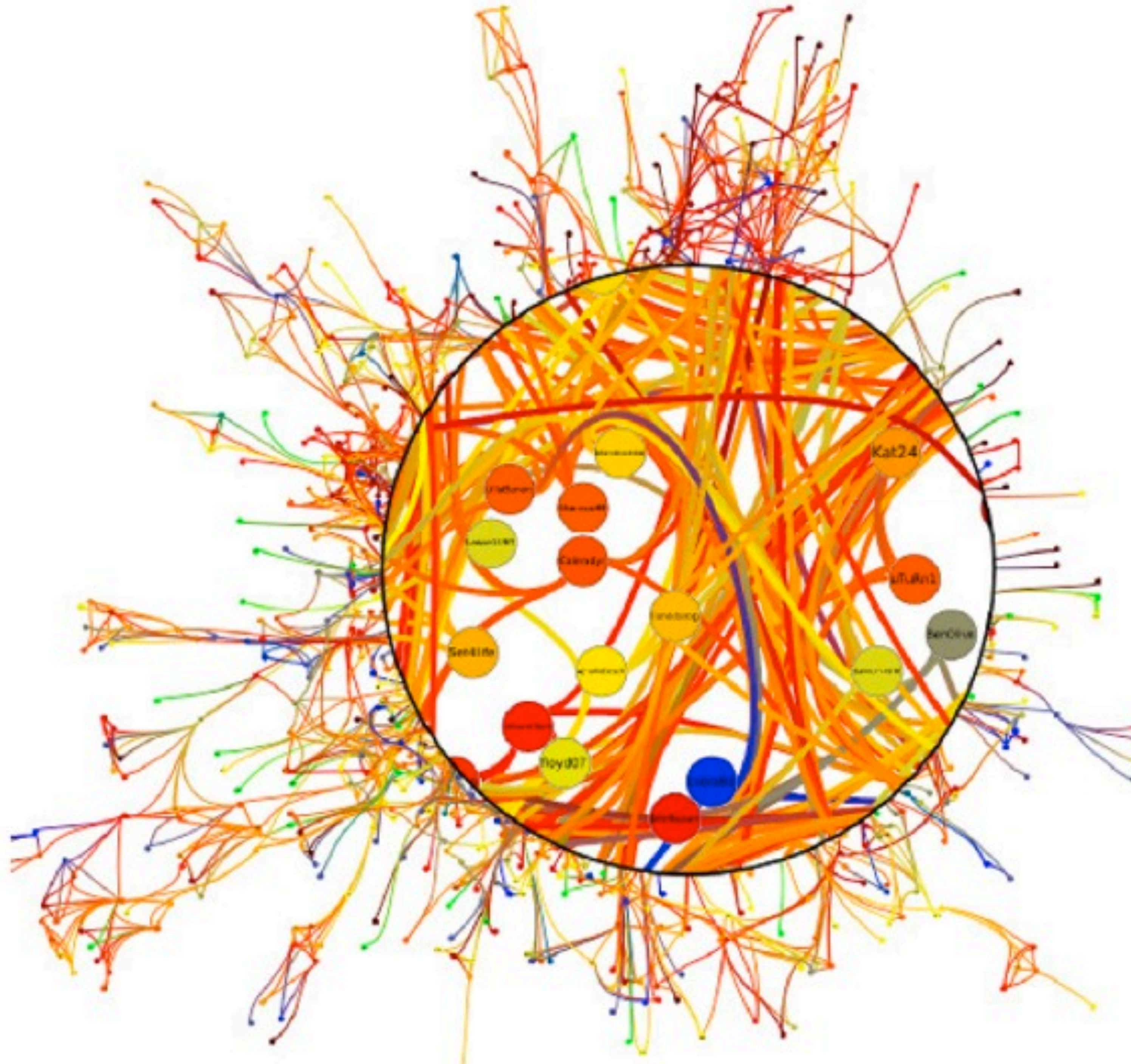
- **linked navigation**
  - shortcut navigation: thumbnail to detail
    - *leaping to far-away regions*
  - explore overview without changing detail (or vice versa)
    - *full synchronize does not support*
  - detail changes immediately shown in overview
- **their definition: lens at O+D**
  - O and D separated in z-depth (layered)
  - Munzner's usage: focus + context

Cockburn 2008

LENSES



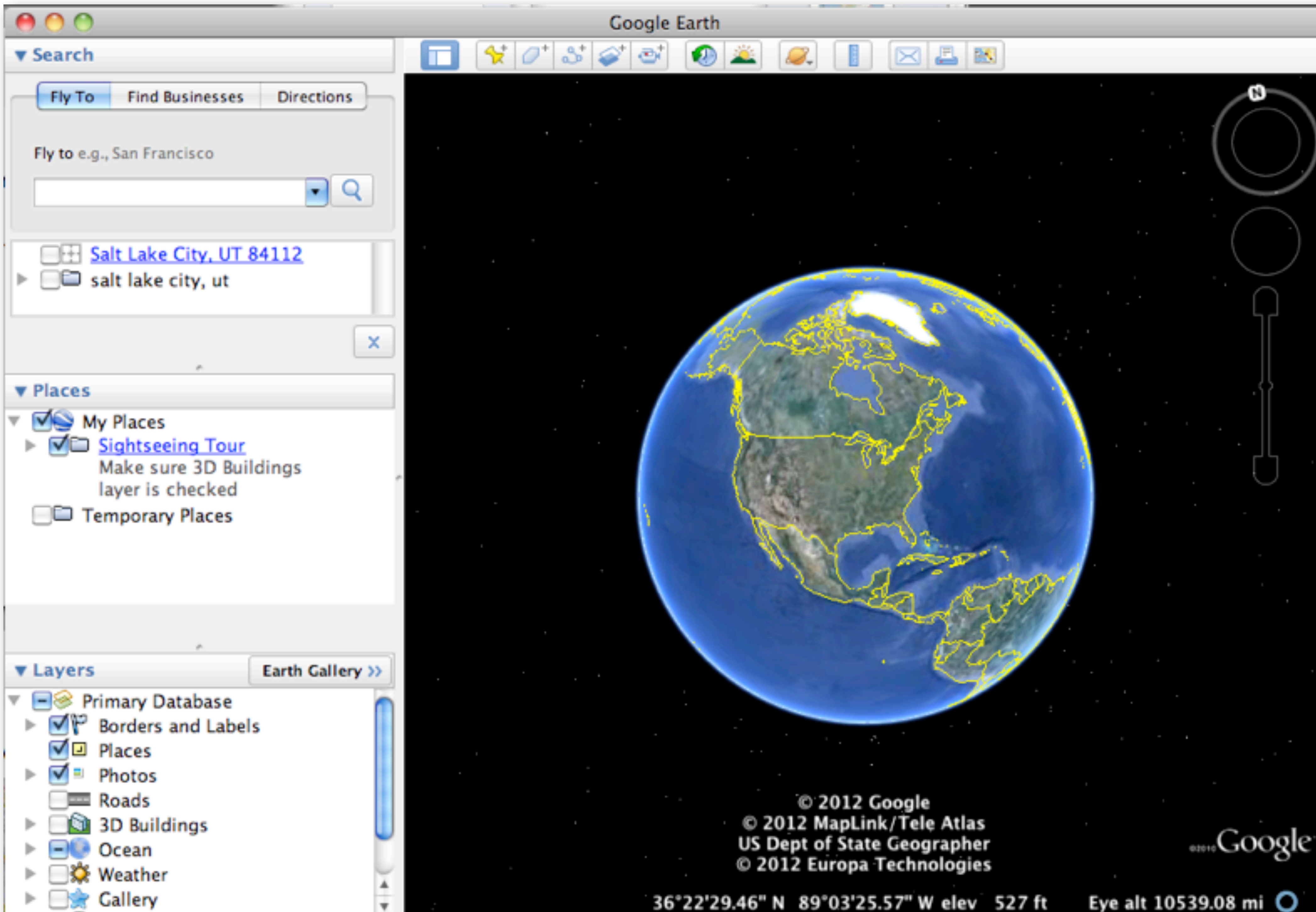
Cockburn 2008



Lambert 2010

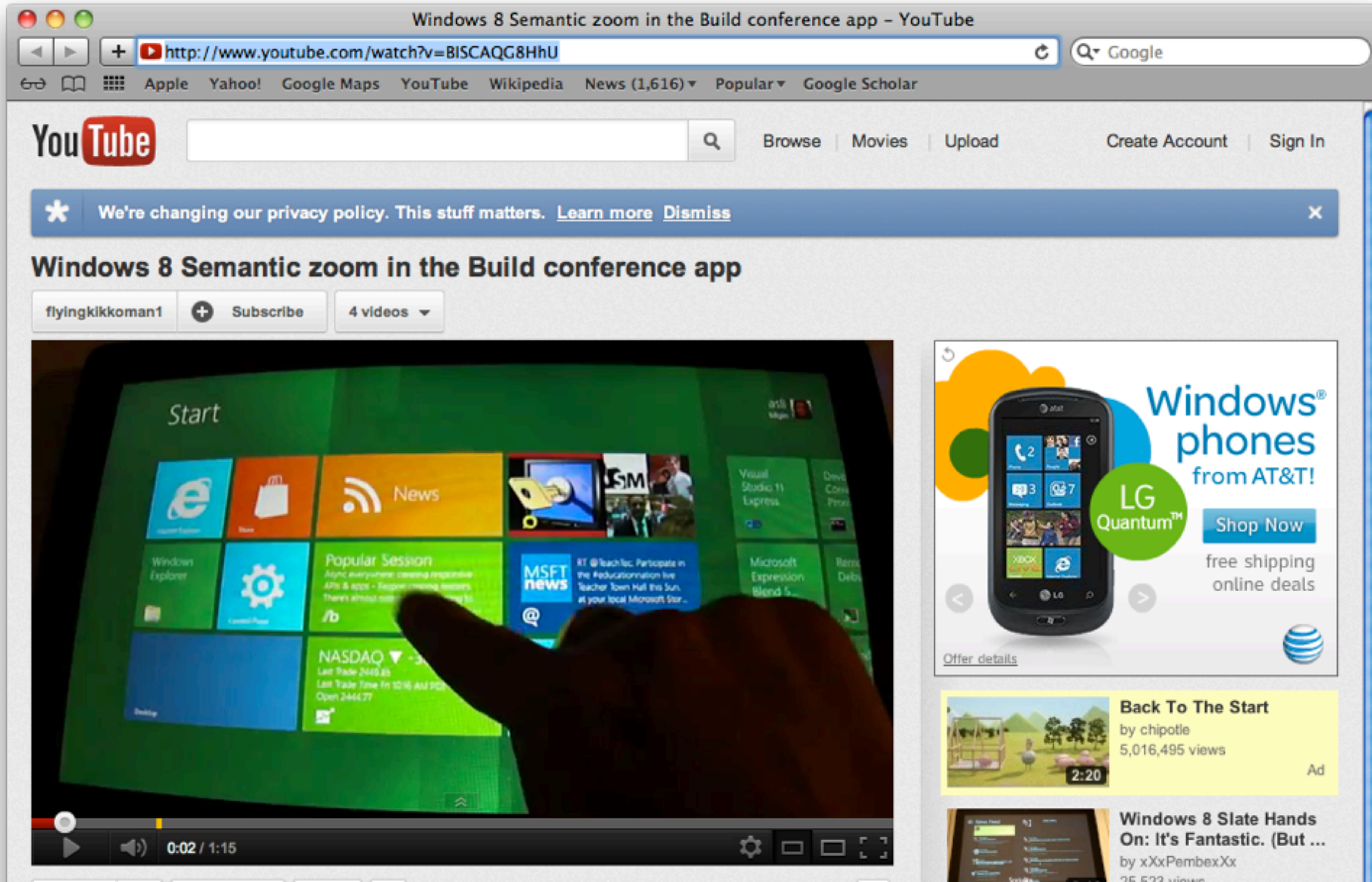
OVERVIEWS
zooming

GEOMETRIC ZOOMING



ZOOMABLE USER INTERFACES

SEMANTIC ZOOMING



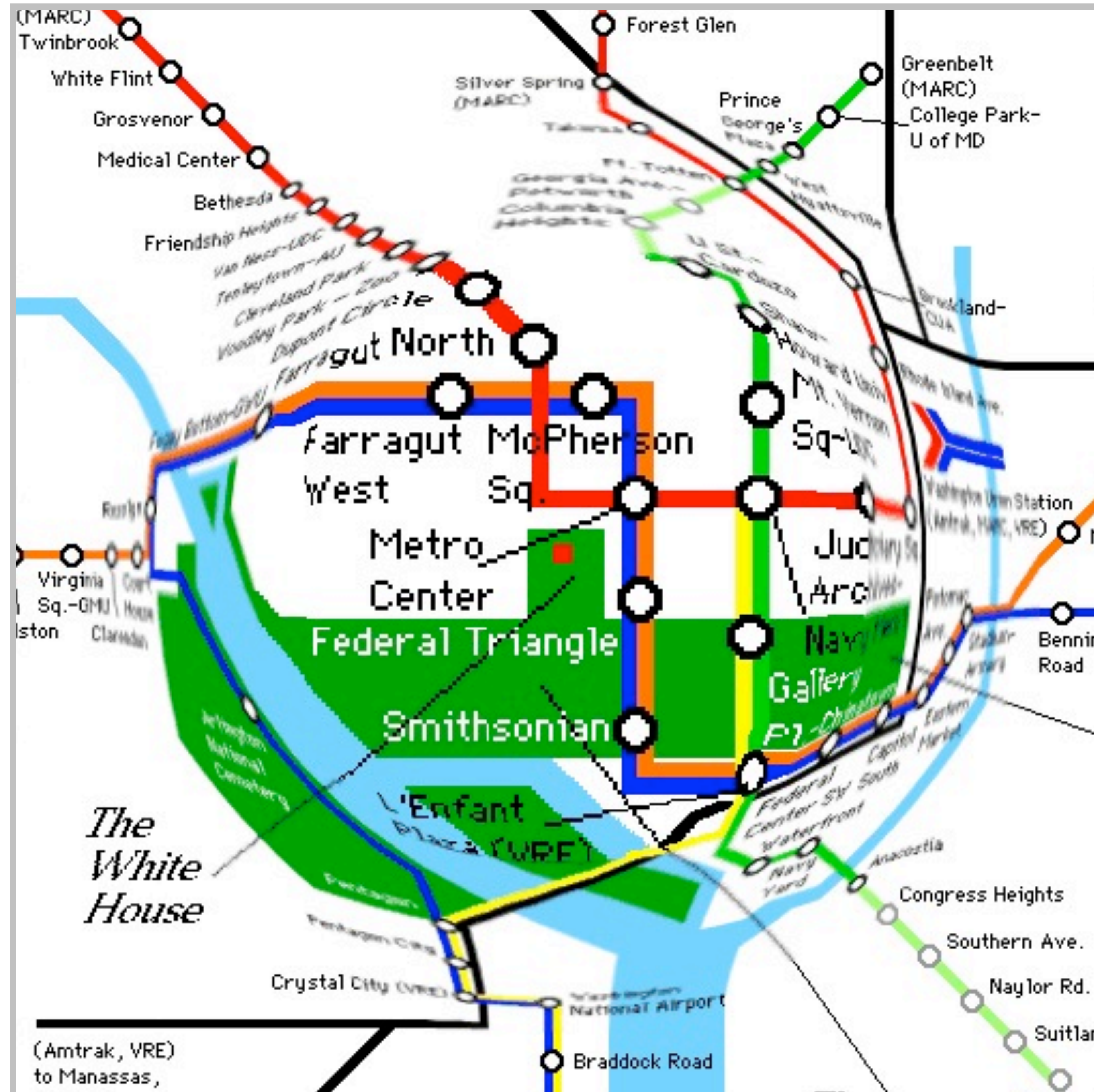
ZOOMING ISSUES

- **geometric zooming**
  - hard to make intuitive zoomout control
  - zoomable user interfaces (ZUIs)
- **semantic zooming**
  - different representations at different scales
- **challenge: stability**

OVERVIEWS
focus + context

FOCUS + CONTEXT

- **integrate focus and context in a single view**



http://www.cs.umd.edu/class/fall2002/cmsc838s/tichi/fisheye.html

F+C FORMALISM

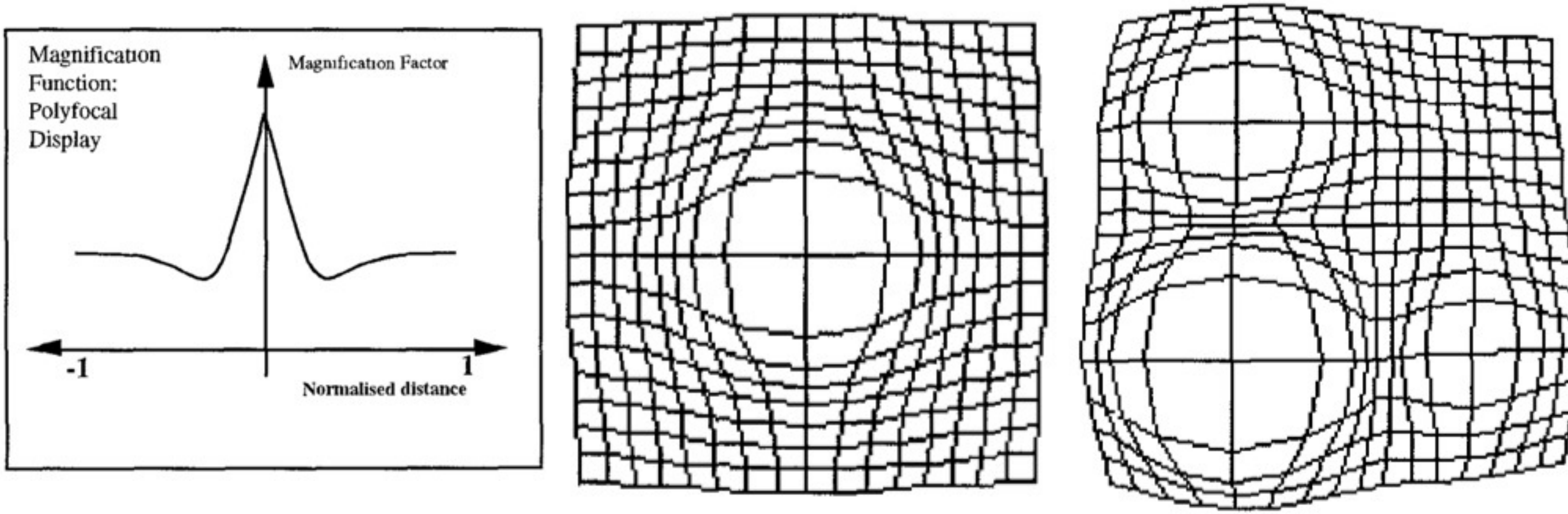
$$\mathbf{DOI}_{fisheye}(\mathbf{a}|. = \mathbf{b}) = \mathbf{API}(\mathbf{a}) - \mathbf{D}(\mathbf{a}, \mathbf{b}),$$

where

(1) $\mathbf{DOI}_{fisheye}(\mathbf{a}|. = \mathbf{b})$ is the degree of interest in **a**, given that the current point of focus is **b**.
(2) **API(a)** is a static global value called *a priori importance* at point **a**; API values are preassigned to each point in the structure under consideration, and
(3) **D(a, b)** is the distance between point **a** and the point of focus **b**.

- **DOI for selective presentation vs distortion**
- **infer DOI through interaction vs explicit selection**
- **single vs multiple foci**

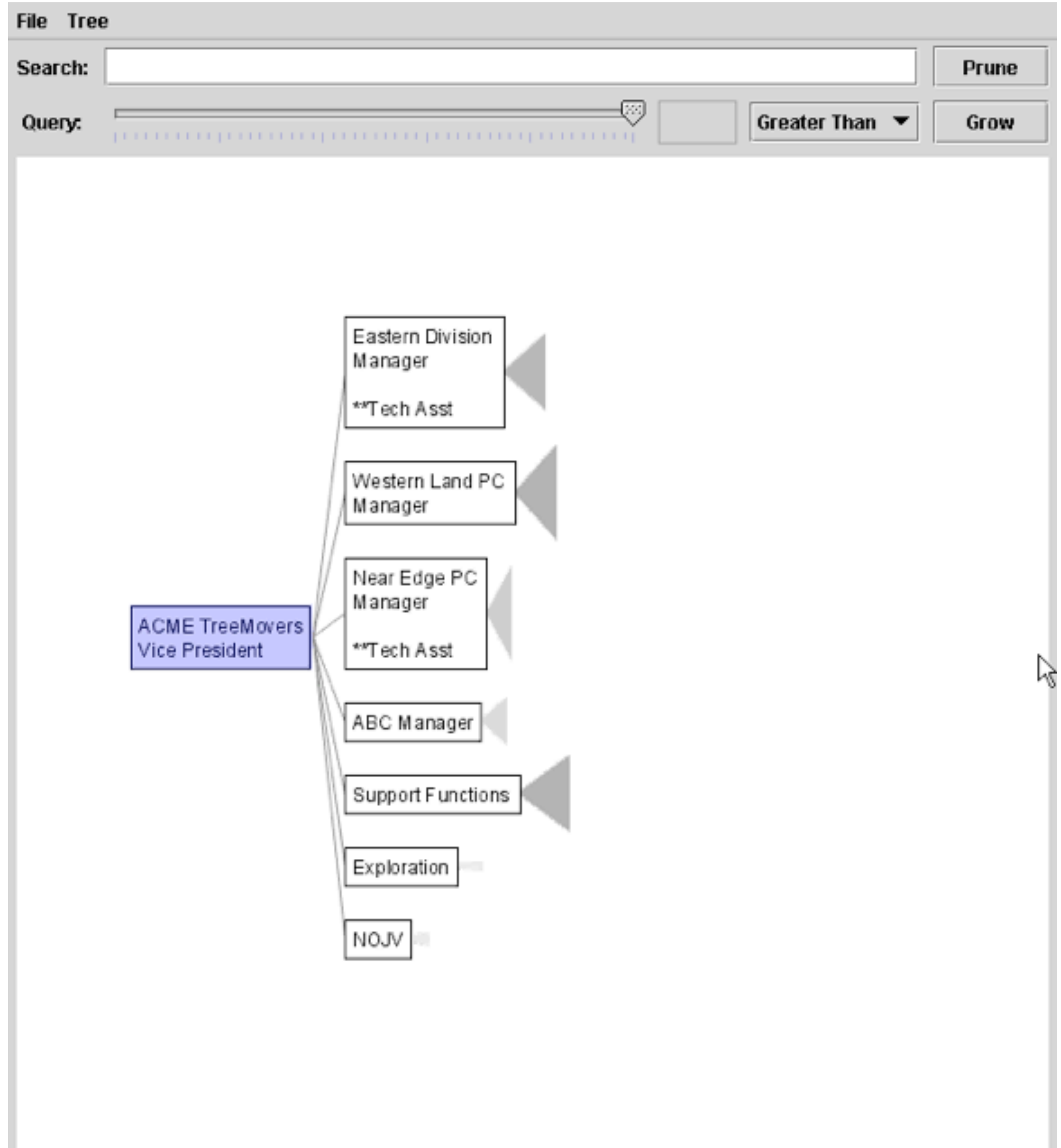
Leung 1994


Magnification
Function:
Polyfocal
Display
Magnification Factor
-1
1
Normalised distance

Leung 1994

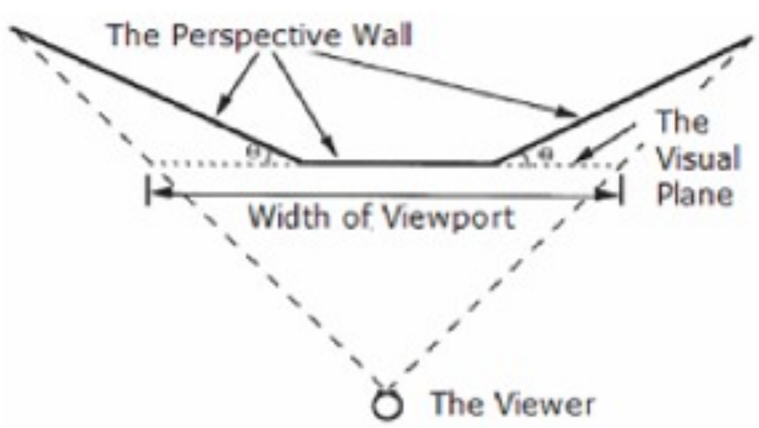
F+C ELISION: SpaceTree

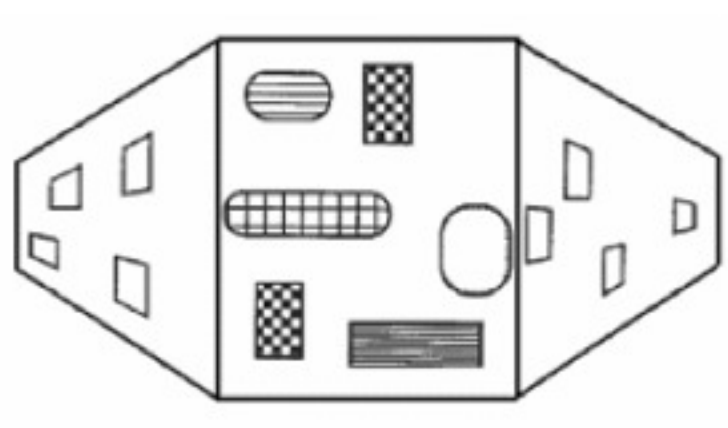
selective filtering with elision



Grosjean 2002

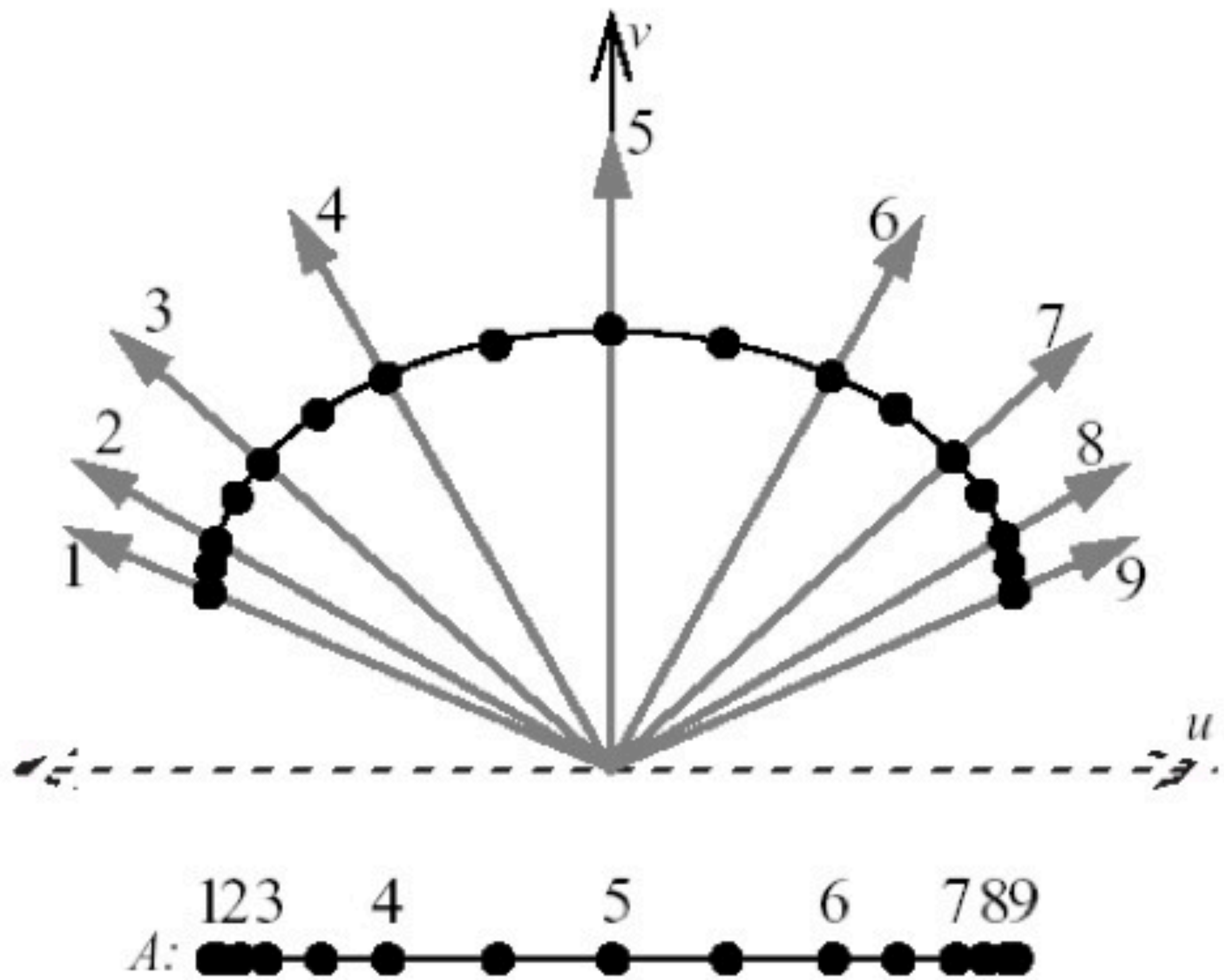
F+C DISTORTION: 3D





Cockburn 2008

F+C DISTORTION: FISHEYE



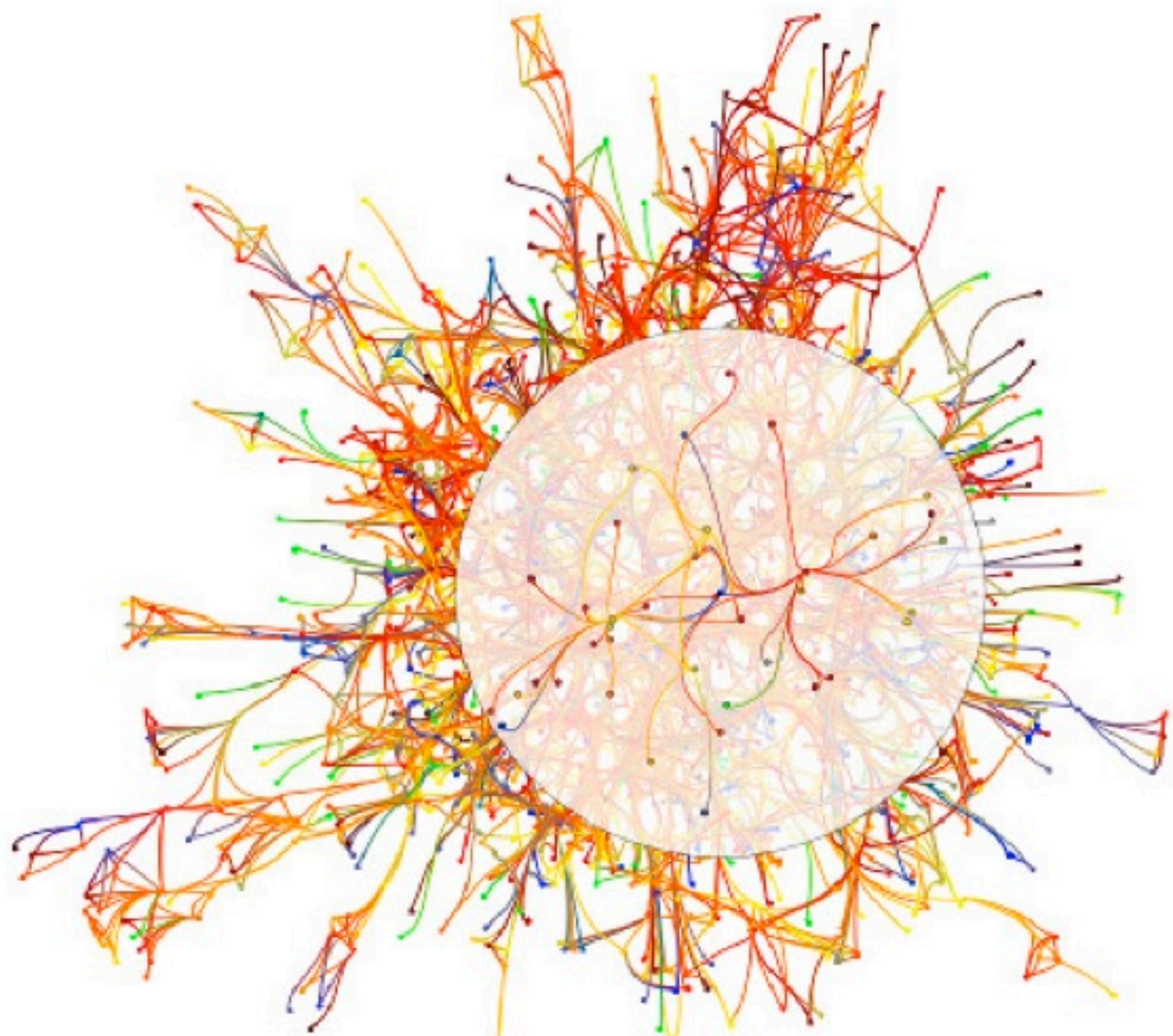
Furnas 1995

DISTORTION CHALLENGES

- **unsuitable for relative spatial judgements**
- **overhead of tracking distortion**
- **visual communication of distortion**
  - gridlines, shading
- **target acquisition problem**
  - lens displacing items away from screen location
- **mixed results compared to separate views and temporal navigation**
- **fisheye follow-up: concern with enthusiasm over distortion**
  - *what* is being shown: selective filtering
  - *how* it is being shown: distortion as one possibility

F+C WITHOUT DISTORTION

magnification



highlight | suppress

Lambert 2010

L11: Dimensionality Reduction

REQUIRED READING

8

Attribute Reduction Methods

The previous chapter approached data reduction in terms of reducing the number of data items to show. This chapter covers **attribute reduction**, where the goal is to reduct the number of attributes that are displayed by eliminating or aggregating them. The distinction is easiest to understand when we consider tabular data. Item reduction corresponds to eliminating rows, whereas attribute reduction means eliminating columns. Of course, these two approaches are not mutually exclusive: both item and attribute reduction could be used in conjunction. Some attribute reduction methods also implicitly eliminate items, as discussed below.

There are many ways to reduce the number of displayed attributes. Camera metaphors naturally lead to slicing, cutting, and projection. Unwanted attributes can simply be filtered away so that only a subset of the attributes are shown, or aggregated so that one representative attribute is a stand-in for all those in its group. A large body of techniques attempt to preserve most of the information in the dataset originally specified through many attributes with a more compact representation using far fewer attributes. These are usually called *dimensionality reduction* techniques; while this book has used the term *attribute* thus far, in this chapter the term *dimension* will also be used in keeping with this terminology. Both linear and the more complex nonlinear mappings can be used to create new synthetic dimensions that capture the important content of the dataset more concisely. Common visual encodings for dimensionally reduced data include scatterplots, scatterplot matrices, and spatializations.

8.1 Camera Metaphors

One set of methods for reducing the number of attributes is based on geometric intuitions about manipulating a virtual camera. Given this foun-

Glimmer: Multilevel MDS on the GPU

Stephen Ingram, Tamara Munzner, *Member, IEEE,* and Marc Olano, *Member, IEEE*

***Abstract*— We present Glimmer, a new multilevel algorithm for multidimensional scaling designed to exploit modern graphics processing unit (GPU) hardware. We also present GPU-SF, a parallel, force-based subsystem used by Glimmer. Glimmer organizes input into a hierarchy of levels and recursively applies GPU-SF to combine and refine the levels. The multilevel nature of the algorithm makes local minima less likely while the GPU parallelism improves speed of computation. We propose a robust termination condition for GPU-SF based on a filtered approximation of the normalized stress function. We demonstrate the benefits of Glimmer in terms of speed, normalized stress, and visual quality against several previous algorithms for a range of synthetic and real benchmark datasets. We also show that the performance of Glimmer on GPUs is substantially faster than a CPU implementation of the same algorithm.**

***Index Terms*— Multidimensional scaling, multilevel algorithms, optimization, GPGPU.**

I. Introduction

MULTIDIMENSIONAL scaling, or MDS, is a technique for dimensionality reduction, where data in a measured high-dimensional space is mapped into some lower-dimensional target space while minimizing spatial distortion. MDS is used when the dimensionality of the dataset is conjectured to be smaller than dimensionality of the measurements. When dimensionality reduction is used for information visualization applications, the low- the spatial distortion between the embedding and the original data increases.

MDS algorithms vary in precisely what form the stress function takes and in how they minimize the stress function. Some are approximate while others are exact, some are iterative while others are completely analytical. Such diversity in algorithms leads to diversity in the quality of the results and the speed at which they are computed. Section II gives a brief overview of various relevant classes of existing MDS algorithms and their underlying characteristics.

One class of MDS algorithms that has had significant influence in information visualization is the class of iterative, force directed algorithms. In such algorithms, data points are modeled as particles in space attached to other particles with springs with an ideal length proportional to the original distance δ. The algorithm computes a simulation by integrating forces until the physical system settles down into a state of minimal energy. At this point computation halts and the final positions of the particles are assigned the resulting coordinates of the data. Naïve implementations of such algorithms can be computationally expensive and prone to converge to local minima.

We present three substantial improvements to the iterative class of MDS algorithms based on simulated forces. First, we improve algorithm speed by exploiting the modern PC graphics processing unit (GPU) as a computational engine. Second, we introduce a