Visualization for Hackers: Why It's Tricky, and Where to Start

Tamara Munzner
Department of Computer Science
University of British Columbia

Hackers on Planet Earth (HOPE) X
19 July 2014, New York NY

http://www.cs.ubc.ca/~tmm/talks.html#hope14

Outline

- introduction
  - what's vis anyway?
- LiveRAC
  - server logs: managed web hosting (with AT&T)
- Overview
  - text: visual document mining for journalists (with Associated Press)
- big picture and wrapup



Defining visualization (vis)

Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.

Why?...

Why have a human in the loop?

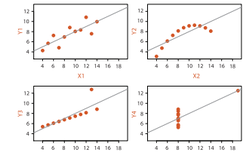
Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.

Visualization is suitable when there is a need to augment human capabilities rather than replace people with computational decision-making methods.

- many analysis problems ill-specified, not clear what questions to ask in advance
  - don't need vis when fully automatic solution exists and is trusted

Anscombe's Quartet

Identical statistics	
x mean	9
x variance	10
y mean	8
y variance	4
x/y correlation	1

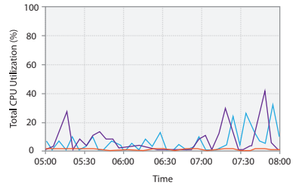


Why use an external representation?

Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.

- external representation: replace cognition with perception

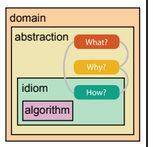
time	db01	appserver	app404
1187668800	0.256711	0.423000	0.750000
1187669100	0.169109	0.348000	0.450000
1187669400	0.236612	0.423000	0.700000
1187669700	0.178449	0.498000	0.800000
1187670000	0.215384	0.175000	2.850000
1187670300	0.198862	0.498000	2.800000
1187670600	0.221656	0.449000	1.050000
1187670900	0.171979	0.496000	1.050000
1187671200	0.236523	0.300000	0.500000
1187671500	0.167673	0.441000	0.800000
1187671800	0.214481	0.225000	0.700000
1187672100	0.180708	0.325000	1.100000
1187672400	0.245111	0.473000	0.700000
1187672700	0.185600	0.522000	0.450000
1187673000	0.206176	0.574000	0.750000
1187673300	0.181770	0.175000	0.850000
1187673600	0.213992	0.399000	0.600000
1187673900	0.179262	0.522000	3.300000
1187674200	0.231237	0.447000	0.550000



Analysis framework: Four levels, three questions

- *domain* situation
  - who are the target users?
- *abstraction*
  - translate from specifics of domain to vocabulary of vis
    - **what** is shown? **data abstraction**
    - **why** is the user looking at it? **task abstraction**
- *idiom*
  - **how** is it shown?
    - **visual encoding idiom**: how to draw
    - **interaction idiom**: how to manipulate
- *algorithm*
  - efficient computation

[A Nested Model of Visualization Design and Validation. Munzner. IEEE TVCG 15(6):921-928, 2009 (Proc. InfoVis 2009).]



[A Multi-Level Typology of Abstract Visualization Tasks Brehmer and Munzner. IEEE TVCG 19(12):2376-2385, 2013 (Proc. InfoVis 2013).]

Why analyze?

- huge design space
  - visual encoding: combinatorial explosion of choices
  - add interaction: even bigger
  - add data abstraction transformation: truly enormous
- most possibilities ineffective for particular task/data combination
  - implication: avoid random walk, be guided by principles
- analysis framework: scaffold to think systematically about design space
  - ensure that consideration space encompasses full scope of possibilities
  - improve chances that selected solution is good not mediocre
  - today's focus: abstractions and idioms, what-why-how

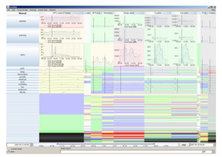
Outline

- introduction
  - what's vis anyway?
- LiveRAC
  - server logs: managed web hosting (with AT&T)
- Overview
  - text: visual document mining for journalists (with Associated Press)
- big picture and wrapup



LiveRAC

Interactive Visual Exploration of System Management Time-Series Data

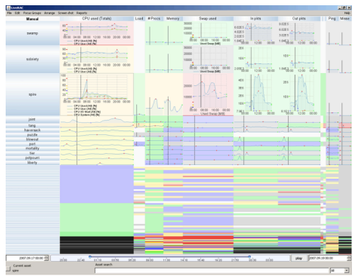


joint work with:
Peter McLachlan, Eleftherios Koutsofios, Stephen North.

http://www.cs.ubc.ca/labs/imager/tr/2008/liverac

LiveRAC - Interactive Visual Exploration of System Management Time-Series Data.
McLachlan, Munzner, Koutsofios, North. Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI'08), p 1483-1492, 2008.

LiveRAC video

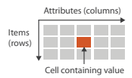


http://youtu.be/ld0c3H0YSkw

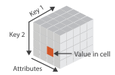
What: Data abstraction

- multidimensional table: time series data
  - key attributes
    - time
      - 50,000: 5-minute intervals over 6 months
      - multiscale levels of interest
    - devices
      - 4000
    - parameters
      - 20
      - ex: CPU usage, memory load, network traffic, alarms, ...
  - value attributes
    - parameter value for device at time point
      - quantitative
    - device groups
      - categorical

→ Tables



→ *Multidimensional Table*



Attribute Types
→ Categorical → Ordered
→ *Quantitative*

Why: Tasks in domain language

- interpret network environment status
- report generation
- capacity planning
- event investigation/forensics
- coordination
  - between customers, engineering, ops

Why: Task abstraction

- browse and correlate across combinations of parameter, device, time
  - correlate alarm attribute with other parameter attribs
  - find trends across groups of devices
  - summarize over different time intervals
  - identify devices at or beyond parameter thresholds
  - identify critical parameter values
  - compare device behavior at specific event times

→ *Discover*

Targets

All Data
→ Trends → Outliers → Features

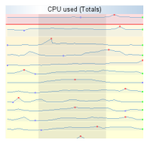
Actions

Query
→ Identify → Compare → Summarise

Attributes
→ One → Many
→ *Distribution* → *Dependency* → *Correlation* → *Similarity*
→ *Extremes*

How: Facet

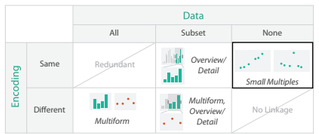
- facet: partition data into multiple views
  - juxtapose views side by side
    - same encoding, different data: *small multiples*



Facet

Juxtapose

Partition



How: Juxtapose

- juxtapose linked views
  - *linked highlighting*
    - marker line tracks across views

Facet

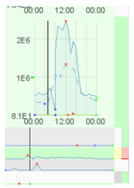
Juxtapose and Coordinate Views

→ Share Encoding: Same/Different

→ *Linked Highlighting*

How: Navigate

- semantic zooming
  - representation adapts to pixels available for object
    - many: superimposed line charts with full labeling
    - some: iconic line chart (sparkline)
    - few: color-coded box (heatmap)



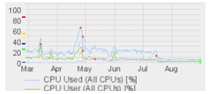
Manipulate

Change View Over Time

Navigate
→ Item Reduction
→ *Zoom*
Geometric or Semantic
→ *Pan/Translate*
→ *Constrained*

How: Superimpose

- superimpose layers
  - vs juxtapose side by side



Facet

→ Juxtapose

→ Partition

→ Superimpose

How: Reduce

- reduce data shown with complex combination of filtering and aggregation
  - embed focus+context in single view
  - distort geometry
    - metaphor: stretch and squish navigation
    - shape: rectilinear
    - foci: multiple
    - impact: global

Reduce

→ Filter

→ Aggregate

→ Embed

→ Distort Geometry

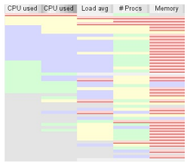
Manipulate

→ Navigate

→ Constrained

How: Reordering

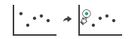
- change spatial arrangement
  - resort by selected attribute
  - check for correlations between aligned attribute columns
    - ex: high load without high CPU, maybe I/O bound



Manipulate

→ Change View Over Time

→ Select



Encode

→ Arrange

→ Order

→ Align

Importance of arranging space: Underlying definitions

- marks
  - geometric primitives
- channels
  - control appearance of marks

→ Points → Lines → Areas

→ Position → Color

→ Horizontal → Vertical → Both

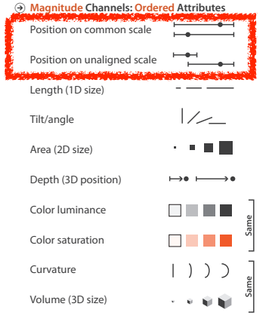
→ Shape → Tilt

→ Size

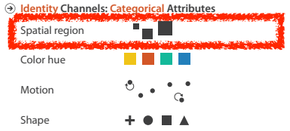
→ Length → Area → Volume

Channels: Expressiveness types and effectiveness rankings

→ Magnitude Channels: Ordered Attributes



→ Identity Channels: Categorical Attributes

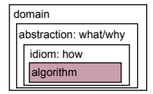


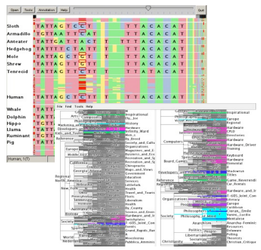
- spatial position channels best in both cases
  - high accuracy
- more on channel rankings: hour-long talk *Visualization Principles*
  http://www.cs.ubc.ca/~tmm/talks.html#networkbio12

Algorithms

- back end: SWIFT server
- front end: PRISAD rendering
  - separate threads for render vs server update
  - guaranteed visibility of semantically important marks even when squished small
    - sublinear rendering: O(p) where p = pixel count
      - scalable for n of millions
      - generic framework
        - » time series charts, gene sequences, trees

[Partitioned Rendering Infrastructure for Scalable Accordion Drawing (Extended Version). Slack, Hildebrand, and Munzner. Information Visualization, 5(2), p. 137-151, 2006.]





Outline

- introduction
  - what's vis anyway?
- LiveRAC
  - server logs: managed web hosting (with AT&T)
- Overview
  - text: visual document mining for journalists (with Associated Press)
- big picture and wrapup

Overview

The Design, Adoption, and Analysis of a Visual Document Mining Tool For Investigative Journalists

joint work with:
Matt Brehmer, Stephen Ingram, Jonathan Stray

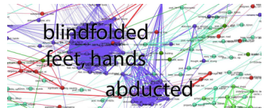
http://www.cs.ubc.ca/labs/imager/tr/2014/Overview/

https://www.overviewproject.org/

Overview: The Design, Adoption, and Analysis of a Visual Document Mining Tool For Investigative Journalists.
Brehmer, Ingram, Stray, and Munzner. IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis 2014), to appear.

Origin story: WikiLeaks meets Glimmer

- WikiLeaks: hacker-journalist Jonathan Stray analyzing Iraq warlogs
  - conjecture that existing label classification falls short of showing all meaningful structure in data
    - friendly action, criminal incident, ...
  - had some NLP, needed better vis tools



- Glimmer: multilevel dimensionality reduction algorithm
  - scalability to 30K documents and terms

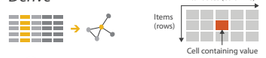
[Glimmer: Multilevel MDS on the GPU.
Ingram, Munzner, Olano. IEEE TVCG 15(2):249-261, 2009.]

What: Data and task abstraction

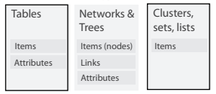
- derive data to transform text into visualizable dataset
  - from documents to high-dimensional table
    - bag of words model
      - attribute: any word that appears across entire collection
      - document/item: word counts (sparse)
  - from high-dimensional table to low-dimensional table
    - synthesize new dimensions that capture most of high-dim proximity structure
    - find clusters of items in lowD space
      - discover: generate or verify

→ Produce

→ Derive



→ Data and Dataset Types

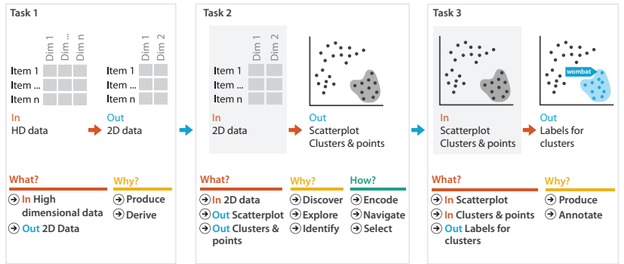


→ Consume

→ Discover

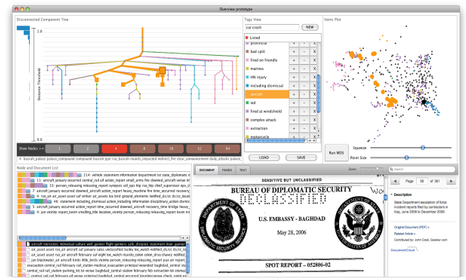
→ Tables

Dimensionality reduction for document datasets



- more on DR: hour-long talk *Dimensionality Reduction from Several Angles*
  http://www.cs.ubc.ca/~tmm/talks.html#linz14

Overview video (version 1)



http://www.cs.ubc.ca/labs/imager/tr/2012/modiscotag

What/Why/How interplay

- why: understand clusters
- what: derive data of full cluster hierarchy
  - explore space of possible clusterings
- how: show cluster hierarchy
  - arrange space: node-link
- how: support tagging clusters/docs
  - following or cross-cutting hierarchy!
    - simple annotation
    - progress tracking
    - user-defined semantics

→ Dataset Types

→ Networks

→ Trees

Targets

→ Network Data

→ Topology

→ Paths

Arrange Networks And Trees

→ Node-link Diagrams
Connections and Marks

→ Produce

→ Annotate

How: Idiom design decisions

- facet: juxtapose linked views
  - linked color coding
    - cluster hierarchy tree
    - DR scatterplot
    - tags
  - reading text/keywords
    - cluster list
    - doc reader

→ Juxtapose and Coordinate Views

→ Share Encoding: Same/Different

→ Linked Highlighting

→ Identity Channels: Categorical Attributes

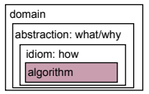
Algorithm

- version 1
  - fast cluster hierarchy construction for sparse data
  - research prototype by PhD student
  - positive initial assessment from AP Caracas bureau chief
    - barrier to adoption: difficult install/load process

Algorithm

- version 1
  - fast cluster hierarchy construction for sparse data
  - research prototype by PhD student
  - positive initial assessment from AP Caracas bureau chief
    - barrier to adoption: difficult install/load process
- version 2
  - web deployment, DocumentCloud integration, usability
    - many months of engineering
      - Knight Foundation funding to the rescue!
    - published story by unaffiliated reporter: police corruption in Tulsa

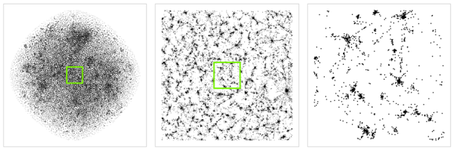
Algorithm: Spinoff series



- dimensionality reduction for huge text collections
  - great algorithm problem in its own right!
  - QSNE: fast and high-quality DR for millions of documents
    - key feature: handle sparseness appropriately

[Dimensionality Reduction for Documents with Nearest Neighbor Queries. Ingram and Munzner. Neurocomputing (Special Issue on Visual Analytics using Multidimensional Projections), to appear 2014.]

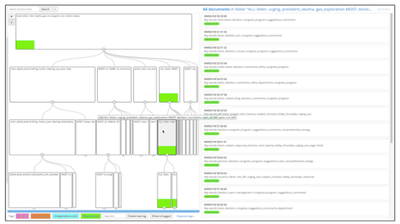
http://www.cs.ubc.ca/labs/imager/tr/2014/QSNE/



Path to adoption

- even more rounds of what/why/how interplay
  - which views needed? what should they show? how should they show it?
  - usability and utility
- version 3
  - published story: VP candidate Ryan asked for federal help even as championed cuts
  - published story: gun control debate
- version 4
  - followup investigation: government corruption in Texas
  - published story: police corruption in New York *(Pulitzer prize finalist!)*

Overview v4 video



http://vimeo.com/71483614

- versions 3 and 4
  - no DR scatterplot
  - tree arrangement emphasizing nodes not links
  - combined doc/cluster viewer

Why: Task abstractions revisited

- what's in this collection? (of leaked docs)
  - *generate* hypothesis
  - *summarize* clusters
  - *explore* clusters
- locate evidence (within FOIA dump)
  - *verify* hypothesis
  - *identify* clusters/documents
  - *locate* clusters/documents
- prove non-existence of evidence
  - even harder!
  - exhaustive reading vs filtering out irrelevant

	Target known	Target unknown
Location known	Lookup	Browse
Location unknown	Locate	Explore

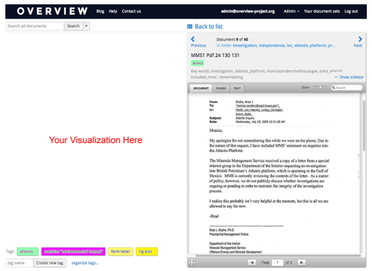
[A Multi-Level Typology of Abstract Visualization Tasks. Brehmer and Munzner. IEEE TVCG 19(12):2376-2385, 2013 (Proc. InfoVis 2013).]

Now what?

- continuing adoption
  - food stamp distribution delays in North Carolina
  - credit card agreements allow repossession
  - this week
    - The Brilliance of Louis C.K.'s Emails: He Writes Like a Politician
- continuing development
  - Knight Foundation funds v5
    - named entity recognition
    - plugin API

https://www.overviewproject.org/

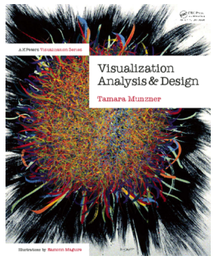
http://overview.ap.org/



Outline

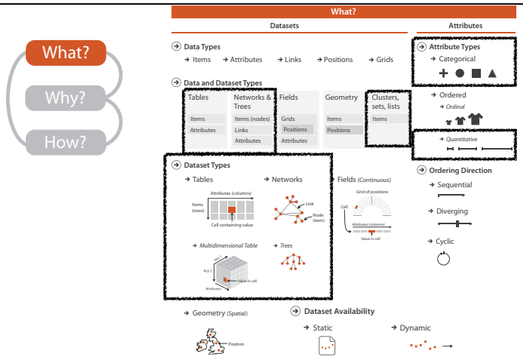
- introduction
  - what's vis anyway?
- LiveRAC
  - server logs: managed web hosting (with AT&T)
- Overview
  - text: visual document mining for journalists (with Associated Press)
- big picture and wrapup

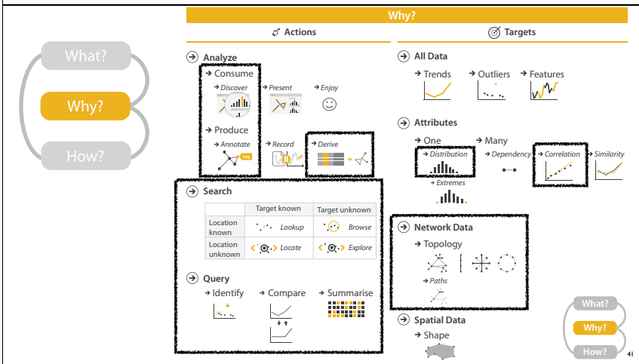
Visualization Analysis & Design

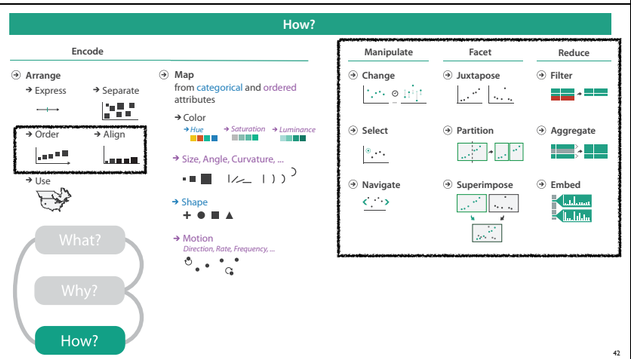


http://www.cs.ubc.ca/~tmm/vadbook

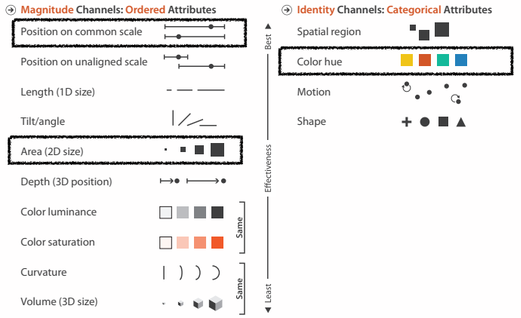
Visualization Analysis and Design.
Munzner. Taylor and Francis / CRC Press, AK Peters Visualization Series, to appear Oct 2014.





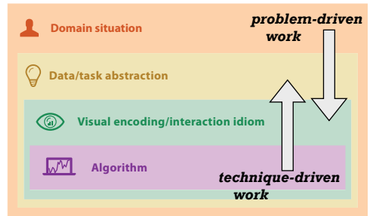


Channels: Expressiveness types and effectiveness rankings



Four levels of design

- inverse cases: technique-driven vs. problem-driven work
  - both useful, but learning curve to switch between



Design Study Methodology

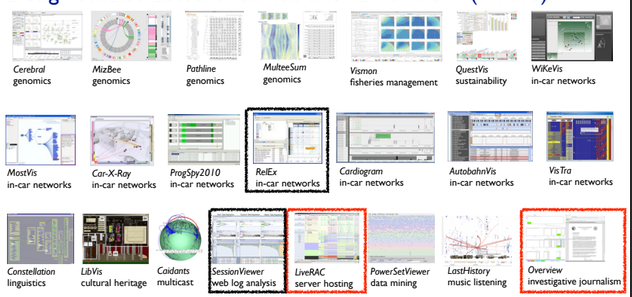
Reflections from the Trenches and from the Stacks

joint work with:
Michael Sedlmair, Miriah Meyer

http://www.cs.ubc.ca/labs/imager/tr/2012/dsm/

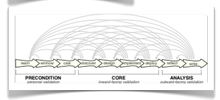
Design Study Methodology: Reflections from the Trenches and from the Stacks.
Sedlmair, Meyer, Munzner. IEEE Trans.Visualization and Computer Graphics 18(12): 2431-2440, 2012 (Proc. InfoVis 2012).

Design Studies: Lessons learned after 21 of them (+more)



Methodology for Problem-Driven Work

- definitions
- 9-stage framework
- 32 pitfalls and how to avoid them



PF-1	premature advance: jumping forward over stages	general
PF-2	premature start: insufficient knowledge of vis literature	learn
PF-3	premature commitment: collaboration with wrong people	winnow
PF-4	no real data available (yet)	winnow
PF-5	insufficient time available from potential collaborators	winnow
PF-6	no need for visualization: problem can be automated	winnow
PF-7	researcher expertise does not match domain problem	winnow
PF-8	no need for research: engineering vs. research project	winnow
PF-9	no need for change: existing tools are good enough	winnow

Wrapup

- two systems analyzed
  - LiveRAC, Overview
- analysis framework big ideas
  - what: data abstraction
    - characterize and derive data
  - why: task abstraction
    - translate from domain-specific to generic
  - how: visual encoding and interaction idioms
    - separate from questions of algorithm design
  - scaffolding for thinking systematically about full design space
    - describing existing systems helps with generating new ones

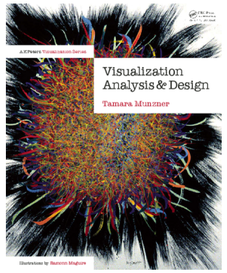
More Information

- this talk
  http://www.cs.ubc.ca/~tmm/talks.html#hope14

- papers, videos, software, talks, courses
  http://www.cs.ubc.ca/group/infovis
  http://www.cs.ubc.ca/~tmm

- book *(to appear Oct 2014)*
  http://www.cs.ubc.ca/~tmm/vadbook

- acknowledgements
  - funding: AT&T, Knight Foundation, NSERC
  - talk feedback: Matt Brehmer



Visualization Analysis and Design.
Munzner. Taylor and Francis / CRC Press, AK Peters Visualization Series, to appear Oct 2014.