Data Abstraction - Summary

- Data abstraction: Three operations
- translate from domain-specific language to generic visualization language

- identify dataset type(s), attribute types
- identify cardinality
- how many items in the dataset?
- what is cardinality of each attribute?
- number of levels for categorical data
- range for quantitative data

- consider whether to transform data
- guided by understanding of task

Dataset types

Tables

Items

Attributes

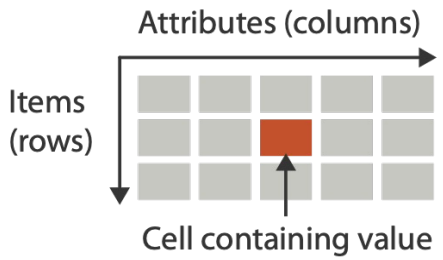
- flat table
  - –one item per row
  - –each column is attribute
  - –cell holds value for item-attribute pair
  - –unique key (could be implicit)

attributes: name, age, shirt size, fave fruit

ID	Name	Age	Shirt Size	Favorite Fruit
1	Amy	8	S	Apple
2	Basil	7	S	Pear
3	Clara	9	M	Durian
4	Desmond	13	L	Elderberry
5	Ernest	12	L	Peach
6	Fanny	10	S	Lychee
7	George	9	M	Orange
8	Hector	8	L	Loquat
9	Ida	10	M	Pear
10	Amy	12	M	Orange

item: person

➔ Tables



x

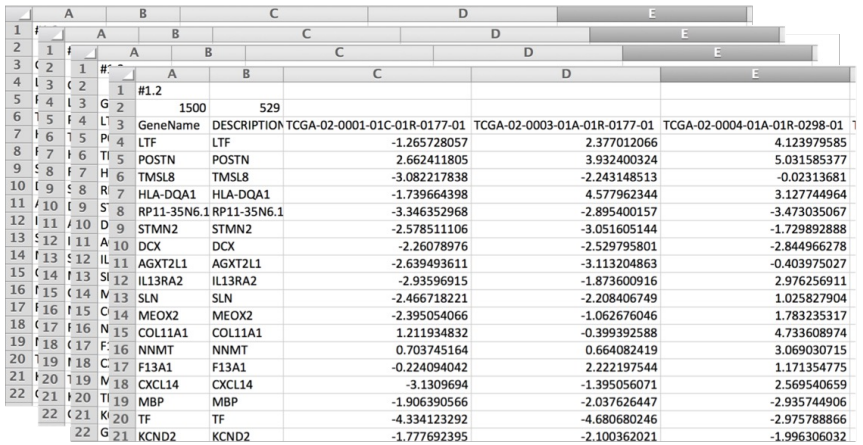
Dataset types

Tables

Items

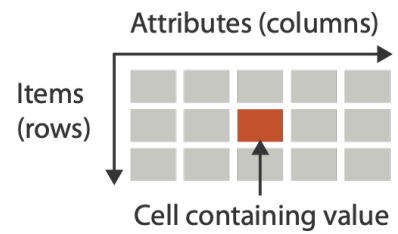
Attributes

- multidimensional tables
  - –indexing based on multiple keys
    - •eg genes, patients

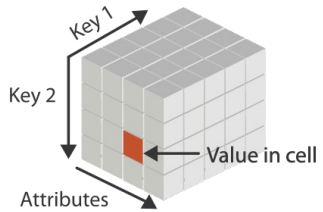


	A	B	C	D	E
1	#1.2				
2	1500	529			
3	GeneName	DESCRIPTION	TCGA-02-0001-01C-01R-0177-01	TCGA-02-0003-01A-01R-0177-01	TCGA-02-0004-01A-01R-0298-01
4	LTF	LTF	-1.265728057	2.377012066	4.123979585
5	POSTN	POSTN	2.662411805	3.932400324	5.031585377
6	TMSL8	TMSL8	-3.082217838	-2.243148513	-0.02313681
7	HLA-DQA1	HLA-DQA1	-1.739664398	4.577962344	3.127744964
8	RP11-35N6.1	RP11-35N6.1	-3.346352968	-2.895400157	-3.473035067
9	STMN2	STMN2	-2.578511106	-3.051605144	-1.729892888
10	DCX	DCX	-2.26078976	-2.529795801	-2.844966278
11	AGXT2L1	AGXT2L1	-2.639493611	-3.113204863	-0.403975027
12	IL13RA2	IL13RA2	-2.93596915	-1.873600916	2.976256911
13	SLN	SLN	-2.466718221	-2.208406749	1.025827904
14	MEOX2	MEOX2	-2.395054066	-1.062676046	1.783235317
15	COL11A1	COL11A1	1.211934832	-0.399392588	4.733608974
16	NNMT	NNMT	0.703745164	0.664082419	3.069030715
17	F13A1	F13A1	-0.224094042	2.222197544	1.171354775
18	CXCL14	CXCL14	-3.1309694	-1.395056071	2.569540659
19	MBP	MBP	-1.906390566	-2.037626447	-2.935744906
20	TF	TF	-4.334123292	-4.680680246	-2.975788866
21	KCND2	KCND2	-1.777692395	-2.100362021	-1.996306032

➔ Tables



➔ *Multidimensional Table*

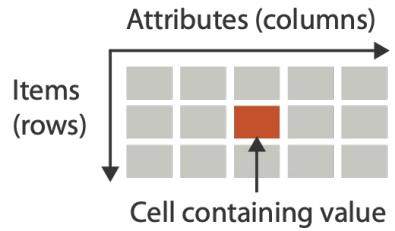


x

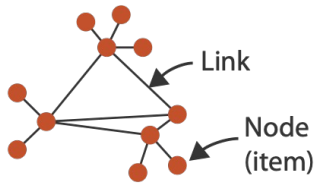
Dataset types

Tables	Networks & Trees	Fields
Items	Items (nodes)	Grids
Attributes	Links	Positions
	Attributes	Attributes

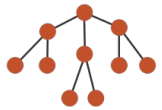
➔ Tables



➔ Networks

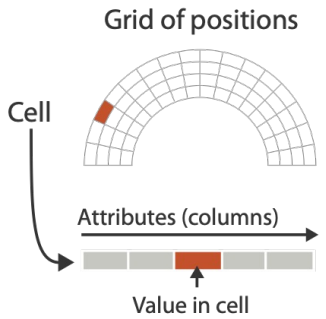


➔ *Trees*



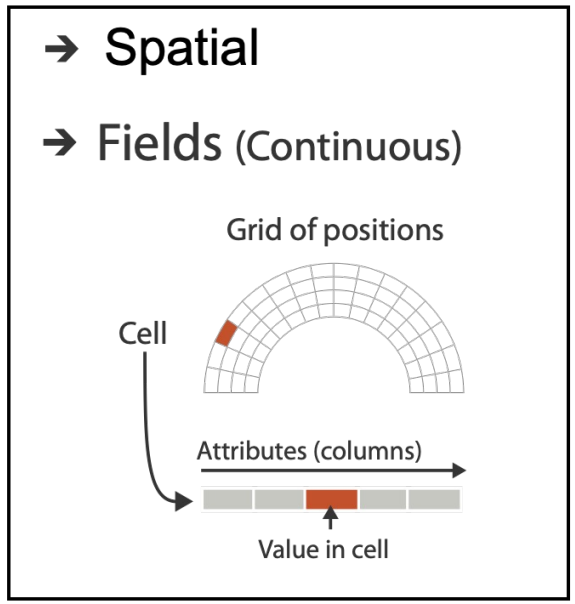
➔ Spatial

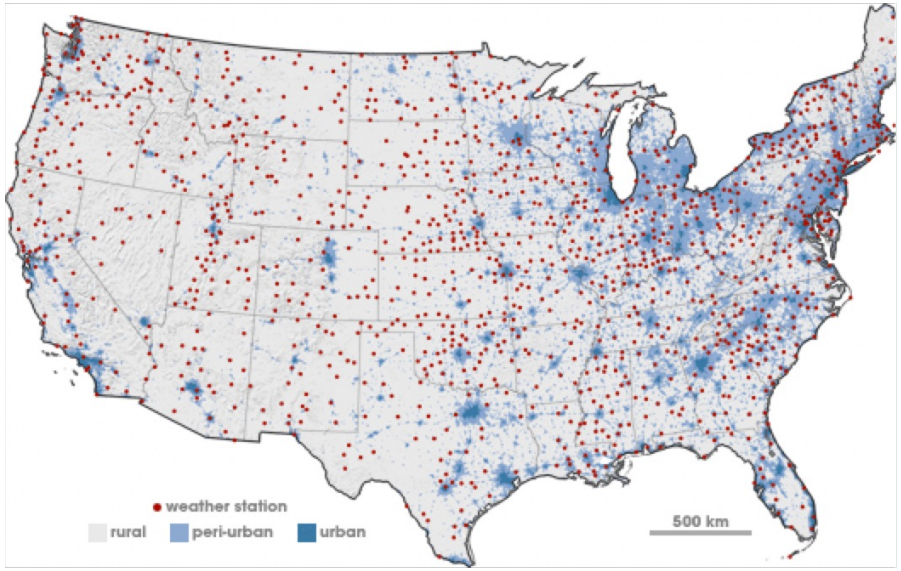
➔ Fields (Continuous)



Spatial fields

- attribute values associated w/ cells
- cell contains value from continuous domain
  - eg temperature, pressure, wind velocity
- measured or simulated

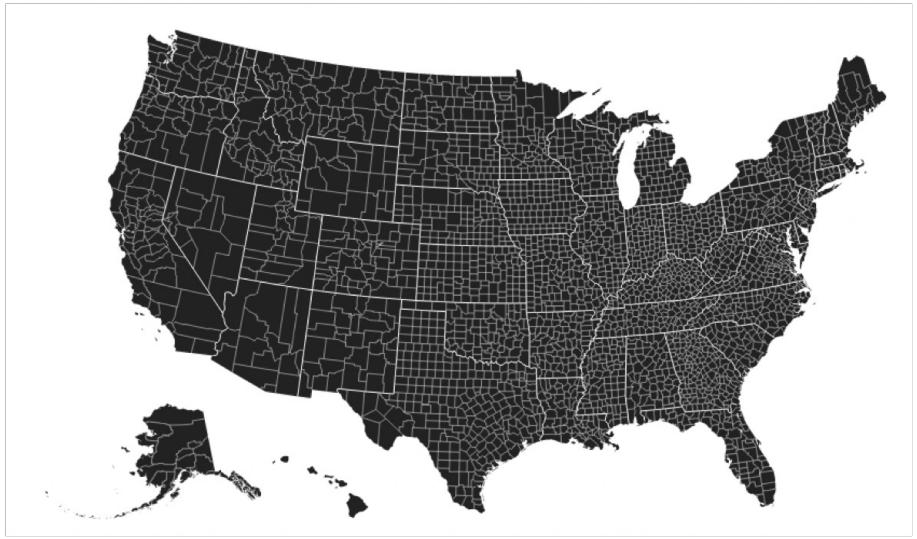




x

Geometry

- shape of items
- explicit spatial positions / regions
  - points, lines, curves, surfaces, volumes
- boundary between computer graphics and visualization
  - graphics: geometry taken as given
  - vis: geometry is result of a design decision

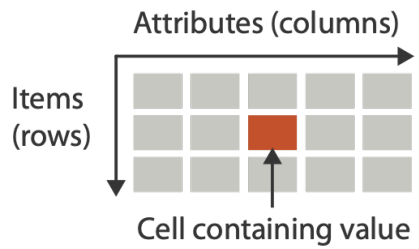


x

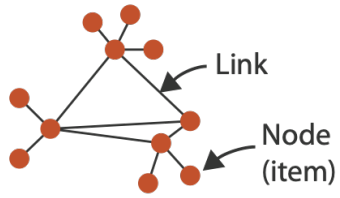
Dataset types

Tables	Networks & Trees	Fields	Geometry
Items	Items (nodes)	Grids	Items
Attributes	Links	Positions	Positions
	Attributes	Attributes	

➔ Tables



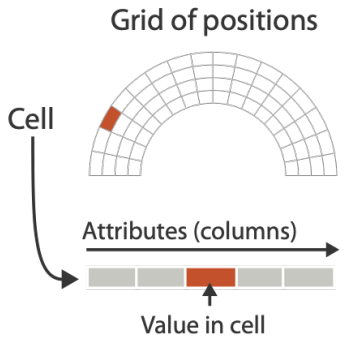
➔ Networks



➔ *Trees*

➔ Spatial

➔ Fields (Continuous)



➔ Geometry (Spatial)

x

Data Semantics

- Start with Data Semantics – what is the meaning of each of the data elements?
- For nominal and ordinal elements what are the unique levels (values)?
- For discrete and continuous elements what are the units and ranges?

Data Wrangling

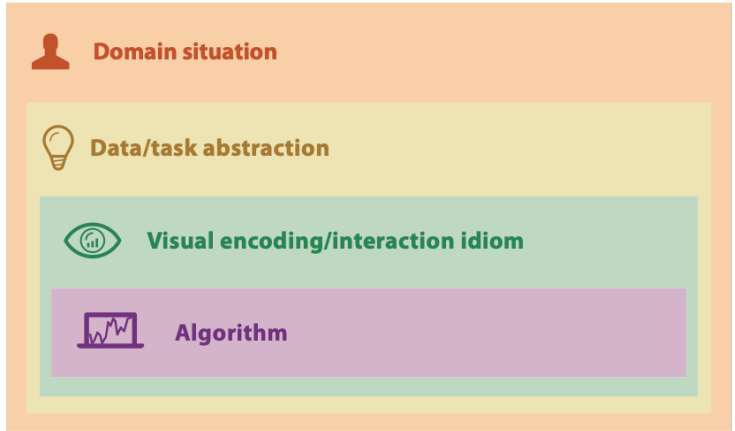
- Missing Data
  - Quantitative Data substitute mean or median
  - Qualitative Data substitute mode
  - Use domain specific function for substitution
- Data in Wrong Format
  - Change format
- Wrong Data
  - Delete data
- Duplicate Data
  - Remove duplicates

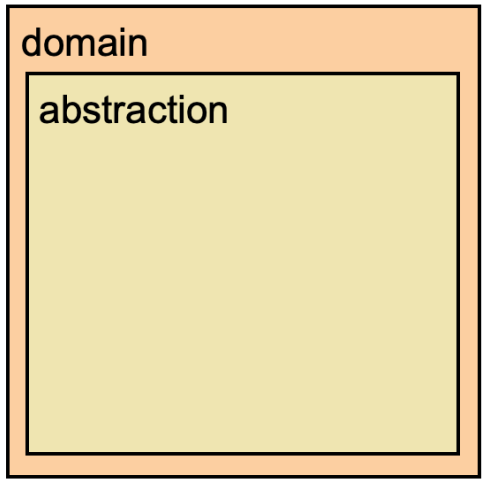
Data Preprocessing

- Data transformation
  - convert from one unit to another (Fahrenheit to Celsius)
- Data binning
  - bin years into decades
- Assigning numerical values to categorical elements
  - grades in class – A:4, B:3, C:2, D:1, F:0

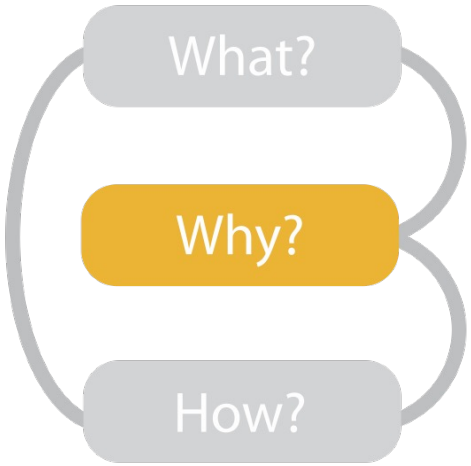
From domain to abstraction

- domain characterization:
  details of application domain
  - group of users, target domain, their questions & data
    - varies wildly by domain
    - must be specific enough to get traction
  - domain questions/problems
    - break down into simpler abstract tasks
- abstraction: data & task
  - map *what* and *why* into generalized terms
    - identify tasks that users wish to perform, or already do
    - find data types that will support those tasks
      - possibly transform /derive if need be





x



- {action, target} pairs
  - *discover distribution*
  - *compare trends*
  - *locate outliers*
  - *browse topology*

Why?

Actions

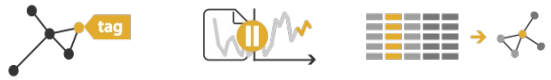
Analyze

→ Consume

→ *Discover* → *Present* → *Enjoy*

→ Produce

→ *Annotate* → *Record* → *Derive*

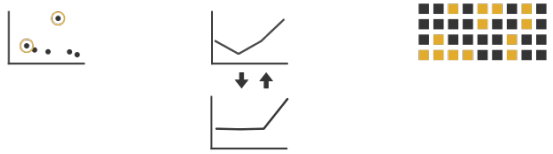


Search

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

Query

→ Identify → Compare → Summarize



Targets

All Data

→ Trends → Outliers → Features

Attributes

→ One → Many

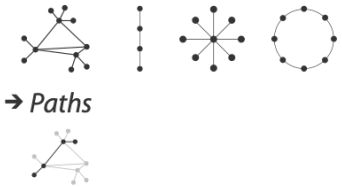
→ *Distribution* → *Dependency* → *Correlation* → *Similarity*

→ *Extremes*

Network Data

→ Topology

→ *Paths*

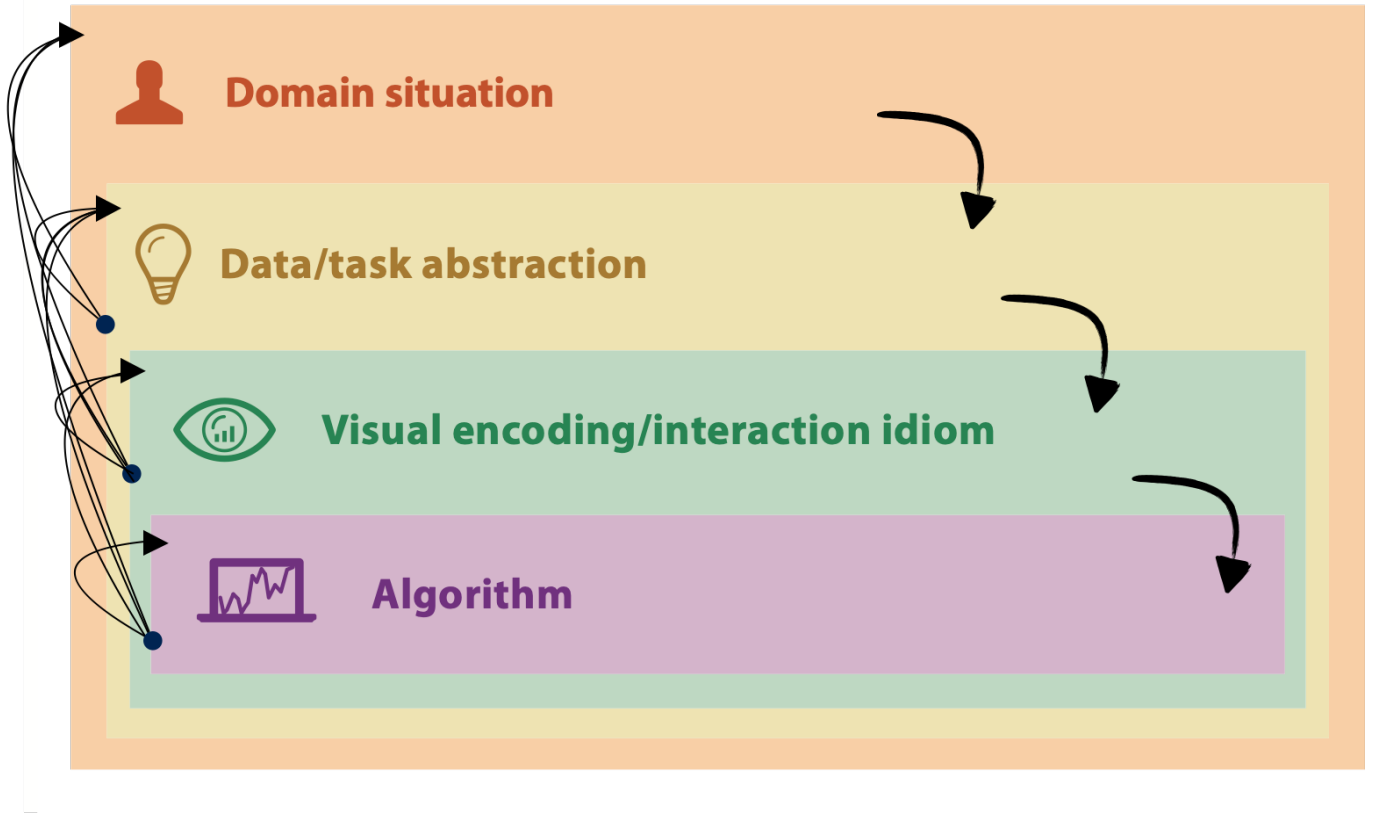


Spatial Data

→ Shape

Nested model

- downstream: cascading effects
- upstream: iterative refinement



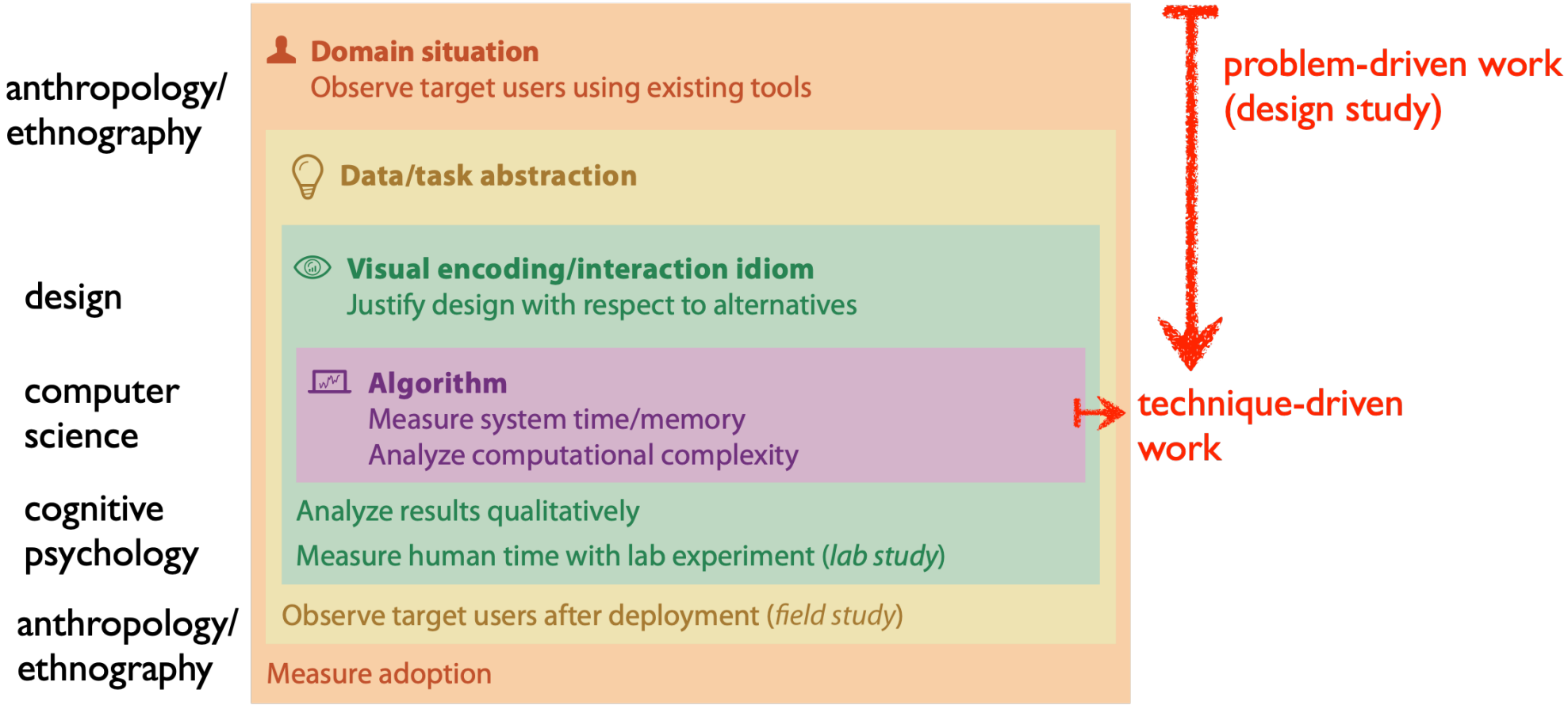
[A Nested Model of Visualization Design and Validation. Munzner. IEEE TVCG 15(6):921-928, 2009 (Proc. InfoVis 2009).]

Validation Process

- Start thinking about validation from the very beginning of the design process rather than at the end
- Design process is iterative – a better understanding at one level will refine blocks at other levels
- Domain is used to mean a particular field of interest – microbiology, astrophysics, economics
- Target users are identified and the designer understands the needs of the user

Why is validation difficult?

- solution: use methods from different fields at each level



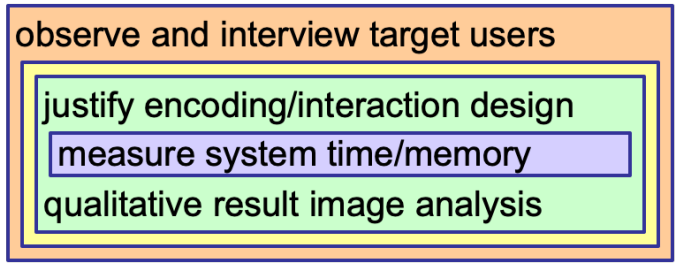
[A Nested Model of Visualization Design and Validation. Munzner. IEEE TVCG 15(6):921-928, 2009 (Proc. InfoVis 2009).]

Threats to Validity

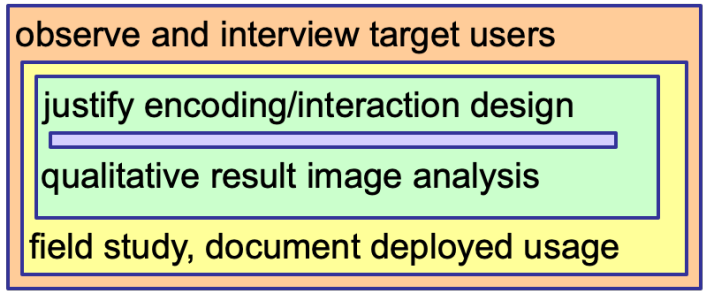
- Domain situation: Wrong problem – you misunderstood their needs
  - field study – observe and interview users in situ
- Data/task abstraction: Wrong abstraction – you are showing them the wrong thing
  - assess how the user deploys the viz tool
- Visual encoding/interaction idiom: Wrong idiom – the way you show does not work
  - justify the design using known perceptual and cognitive principles
- Algorithm: Wrong algorithm – your code is too slow
  - analyze the computational complexity of the algorithm

Analysis examples: Single paper includes only subset of methods

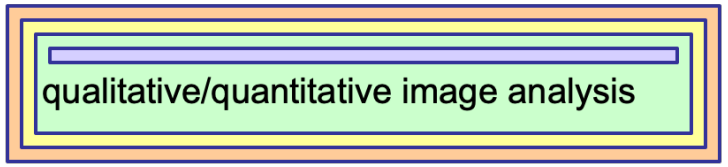
MatrixExplorer. Henry and Fekete. InfoVis 2006.



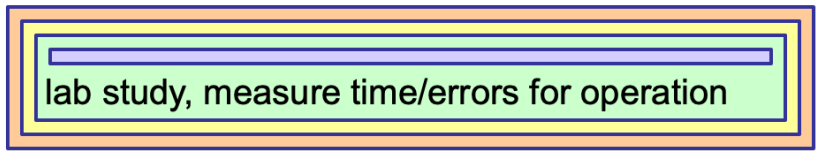
LiveRAC. McLachlan, Munzner, Koutsofios, and North. CHI 2008.



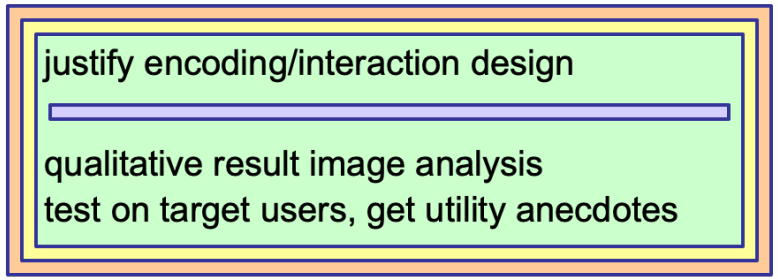
An energy model for visual graph clustering. (LinLog) Noack. Graph Drawing 2003



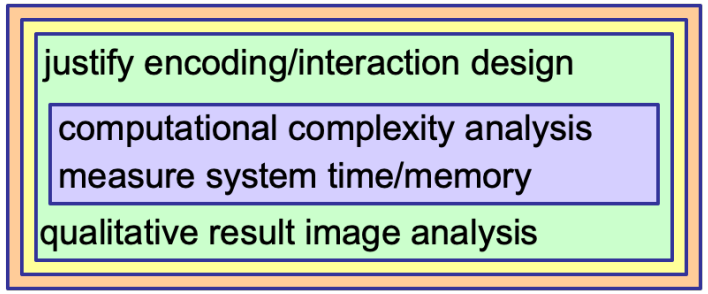
Effectiveness of animation in trend visualization. Robertson et al. InfoVis 2008.



Interactive visualization of genealogical graphs. McGuffin and Balakrishnan. InfoVis 2005.



Flow map layout. Phan et al. InfoVis 2005.



x