What is provenance?

Principles of Provenance Workshop
June 26, 2007

James Cheney

What is it?

- Causes?
- Influence?
- Witness?
- Trace?
- Justification?
- Proof?
- Evidence?
- History?

What is it for?

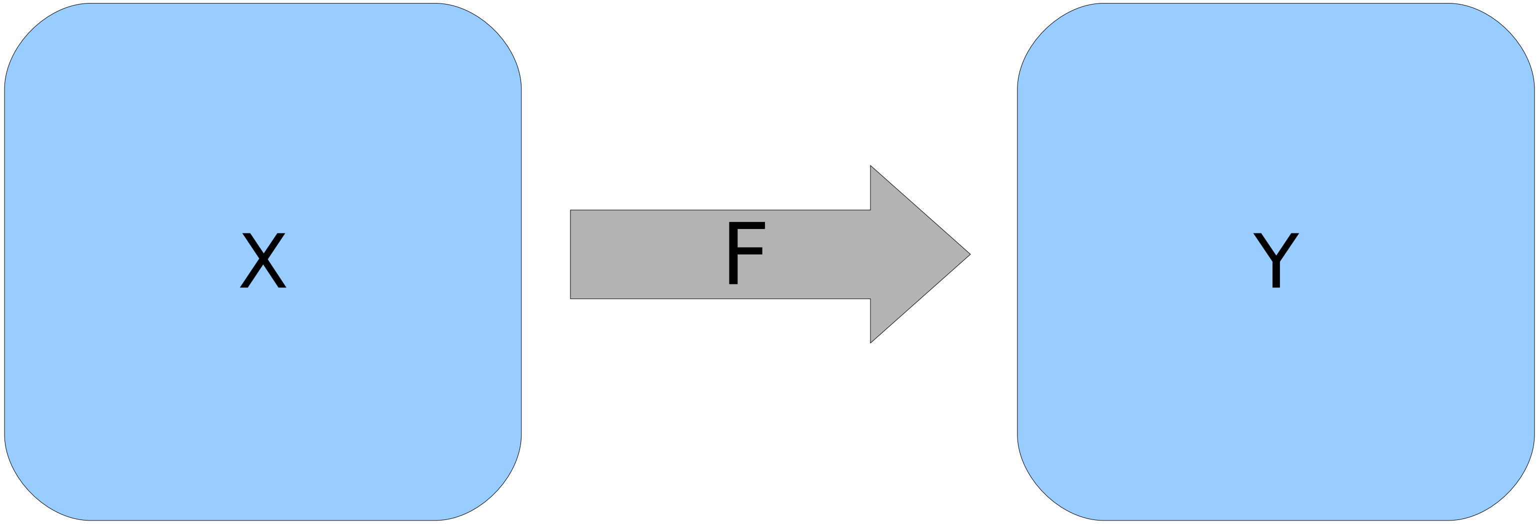
- Integrity?
- Validity?
- Quality?
- Trust?
- Validation?
- Error-correction?
- Security?
- Accountability?

Step back

- Lots of distracting details of different techniques
  - Trees, not forest
- Forget about databases, workflows, whatever.
- Try to define what we are talking about.
  - Why are we doing this?
  - How will we know when we've succeeded?

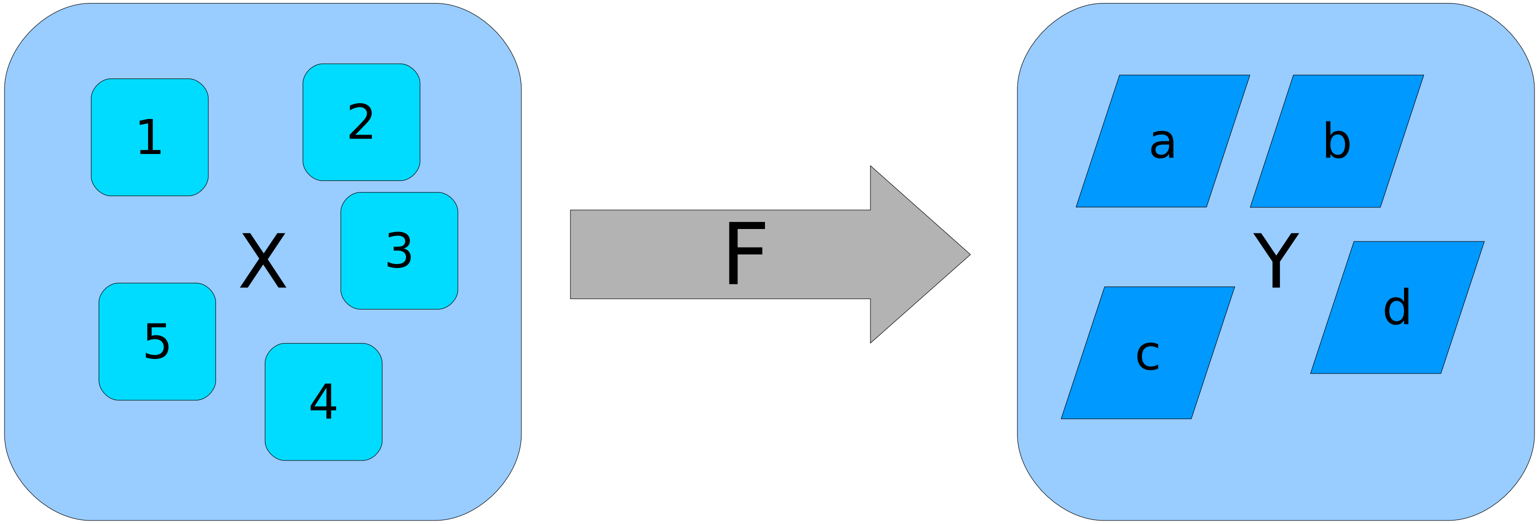
Model

- You have a function F mapping inputs in X to outputs in Y.



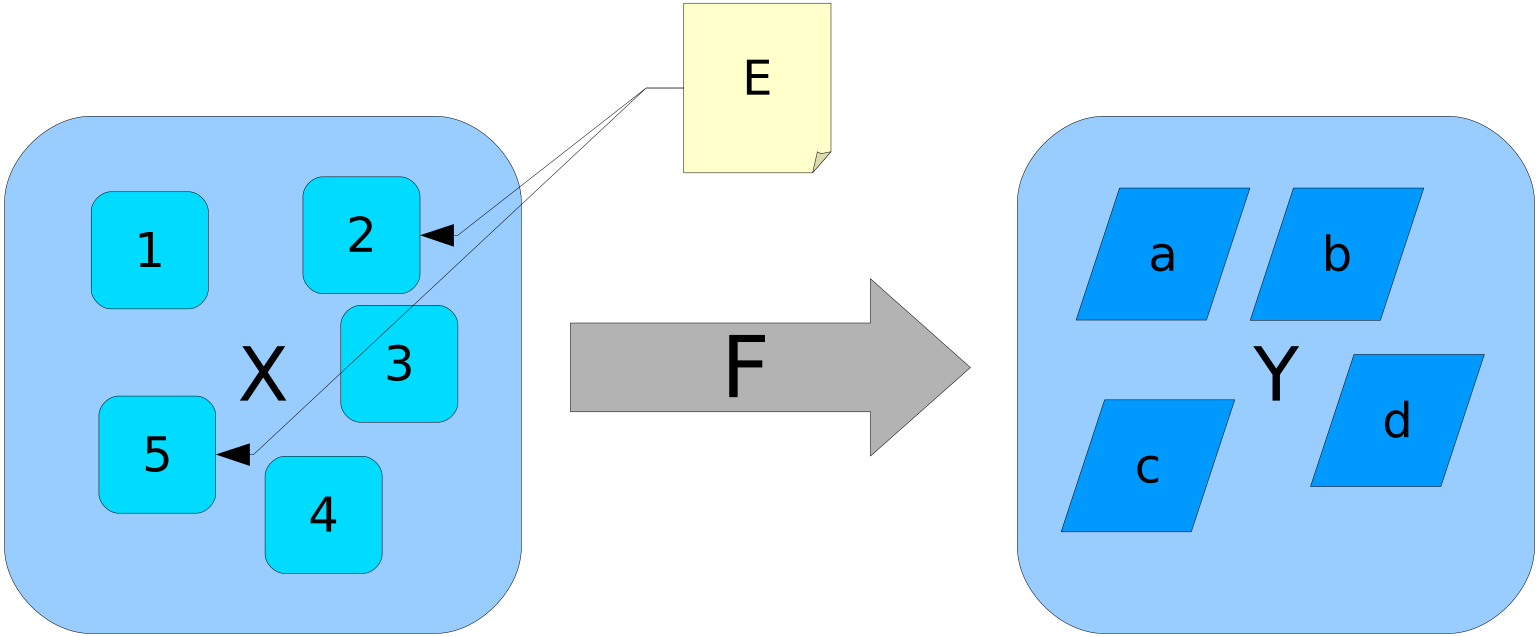
Model

- X and Y have parts that can be addressed by *locations*



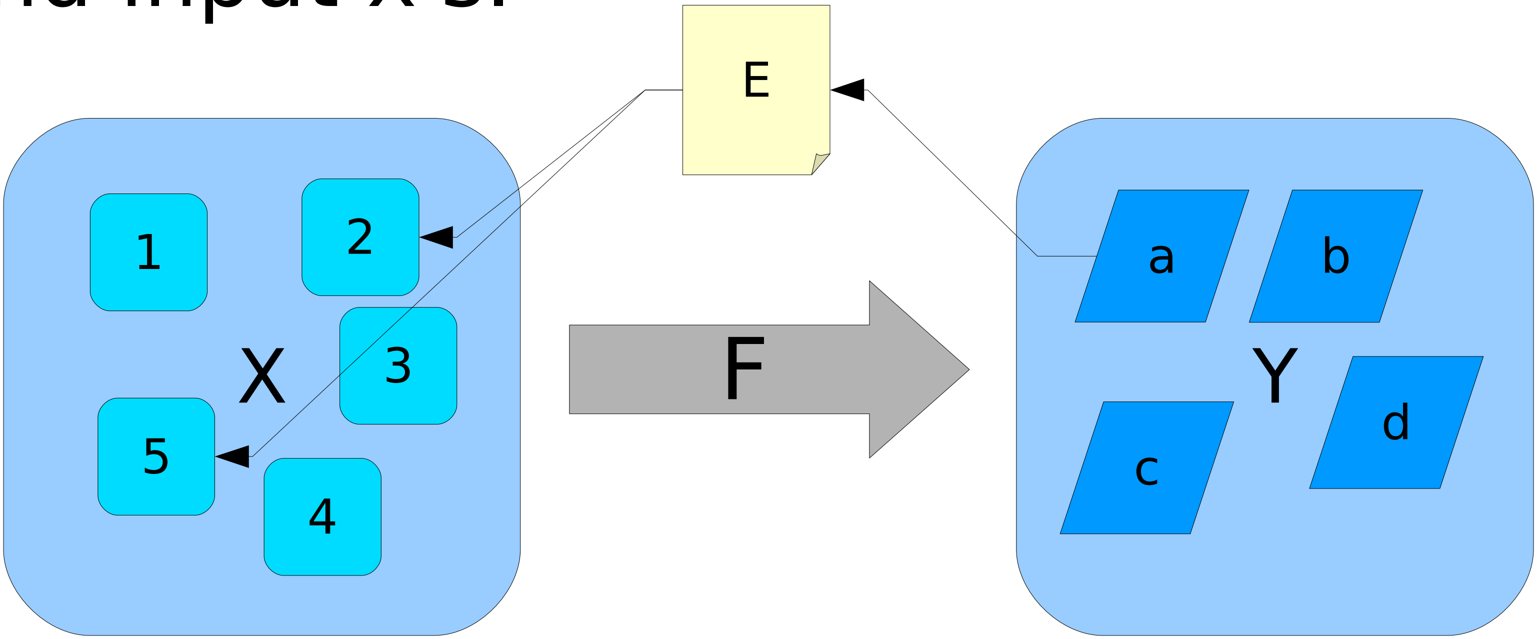
Model

- There is a notion of *explanations* (that can refer to locations in X)



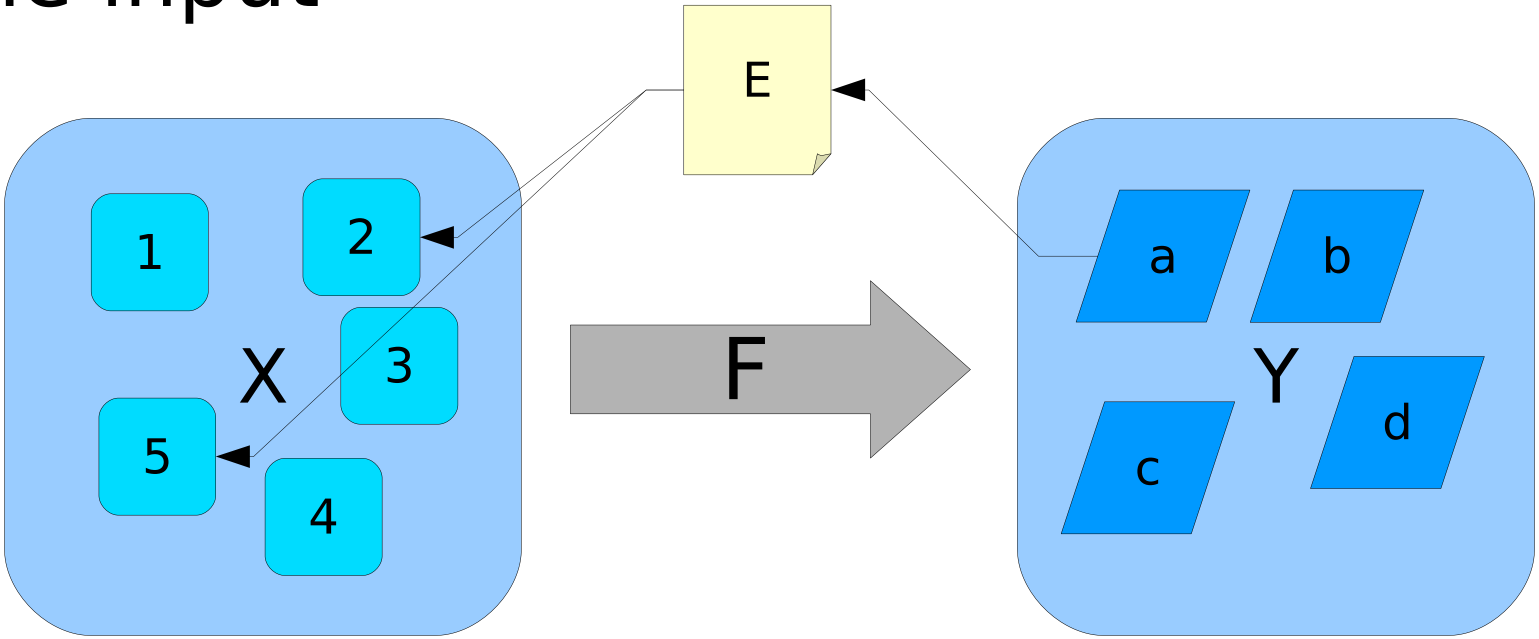
Model

- Explanations need to be *accurate* and *concise* with respect to the F, output y, and input x's.



Model

- *Provenance* is an accurate explanation of a part of the output in terms of parts of the input



Loose ends

- What are sensible X's, Y's and locations?
- What are sensible notions of “accuracy” for explanations?
- Is there a “most general” prov semantics for a given language/model?
  - Semirings may be for relations...

What fits?

- Polygen
- Why, where
- Lineage
- Update (where-)provenance
- Workflow prov
- Provenance semirings
- Line numbers in compilers/interpreters

What doesn't fit?

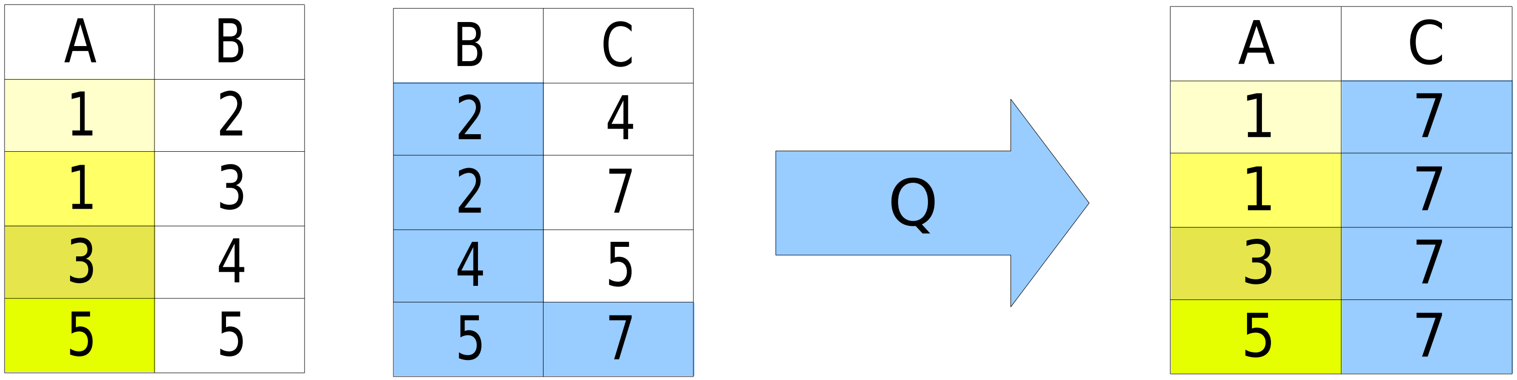
- Probably lots of things...
- Not meant to be “last word”
    - point of definition is partly so that we can pick it apart
- try to understand what it *doesn't* describe
        - & why

One new idea

- Can we apply well-understood techniques from PL/program analysis?
  - Program slicing: identify subset of *program* relevant to a particular output
  - “Data slicing”: identify subset of *input data* relevant to a particular output
- “Why-provenance as data slicing”
  - Can re-use existing machinery for dependence analysis

“Data slicing”

- Provenance of output is data “relevant” in input



A	B
1	2
1	3
3	4
5	5

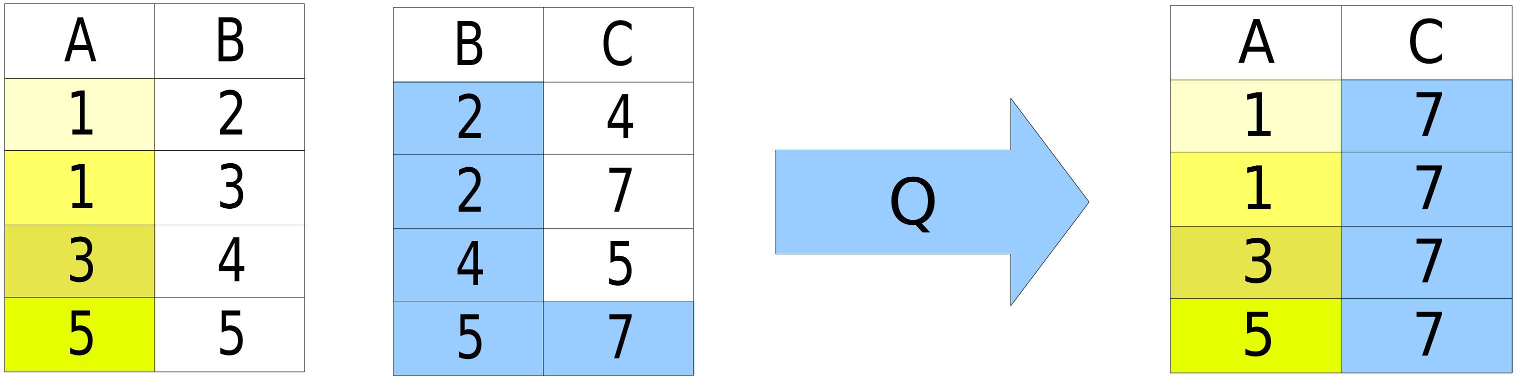
B	C
2	4
2	7
4	5
5	7

A	C
1	7
1	7
3	7
5	7

- What does this tell us about Q?

“Data slicing”

- Provenance of output is data “relevant” in input



A	B
1	2
1	3
3	4
5	5

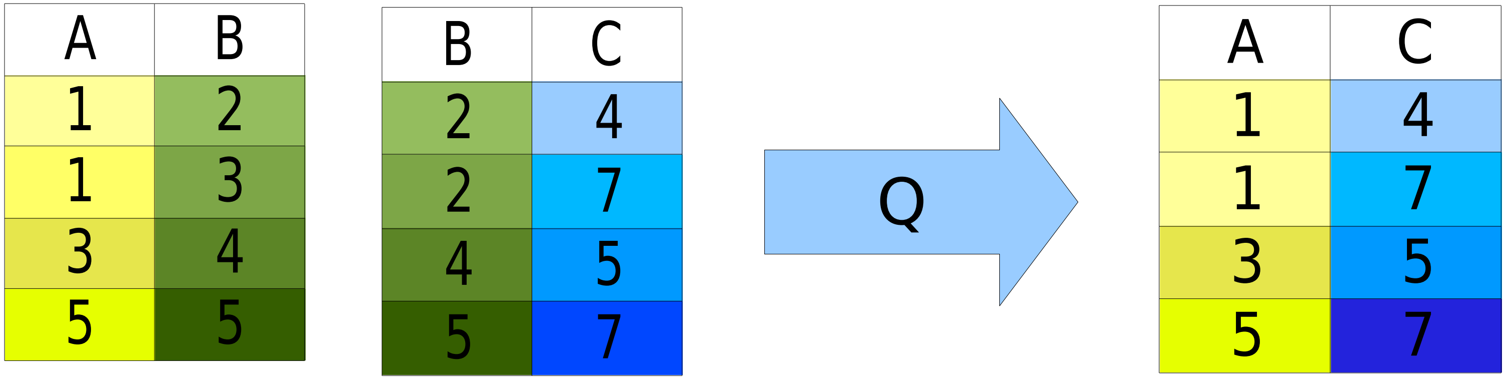
B	C
2	4
2	7
4	5
5	7

A	C
1	7
1	7
3	7
5	7

- What does this tell us about Q?
- SELECT A,C FROM R,S WHERE S.B = 5

“Data slicing”

- Provenance of output is data “relevant” in input



A	B
1	2
1	3
3	4
5	5

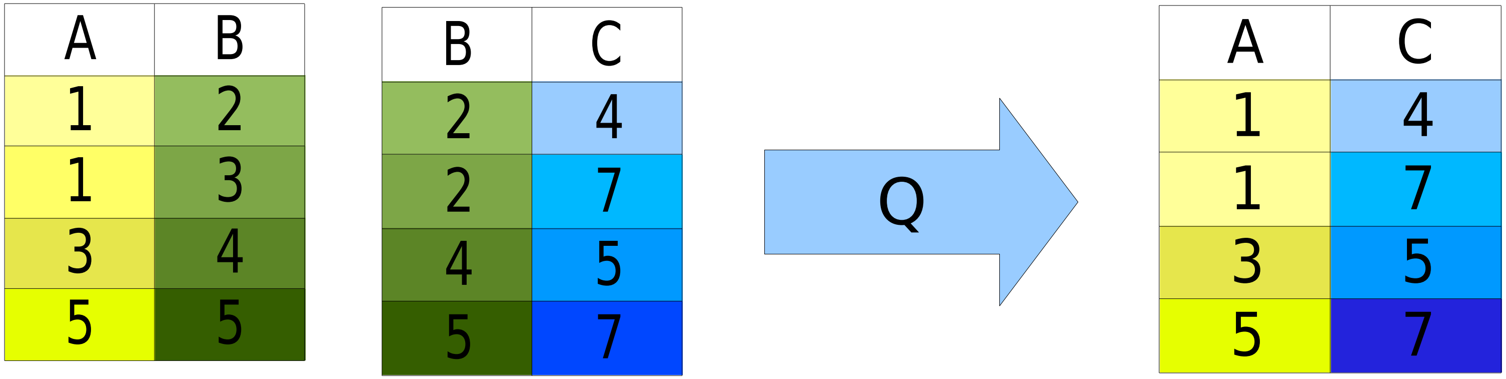
B	C
2	4
2	7
4	5
5	7

A	C
1	4
1	7
3	5
5	7

- What does this tell us about Q?
- Green data relevant to both yellow & blue

“Data slicing”

- Provenance of output is data “relevant” in input



A	B
1	2
1	3
3	4
5	5

B	C
2	4
2	7
4	5
5	7

A	C
1	4
1	7
3	5
5	7

- What does this tell us about Q?
- Green data relevant to both yellow & blue
- SELECT A,C FROM R,S WHERE R.B=S.B

Information flow security

- Dependency analysis techniques also underly research on *information flow security*
  - implicit, explicit flows, “taintedness” tracking