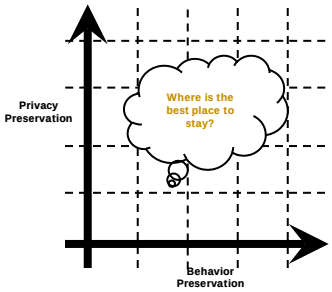
School of
Information Systems

SMU
SINGAPORE MANAGEMENT
UNIVERSITY

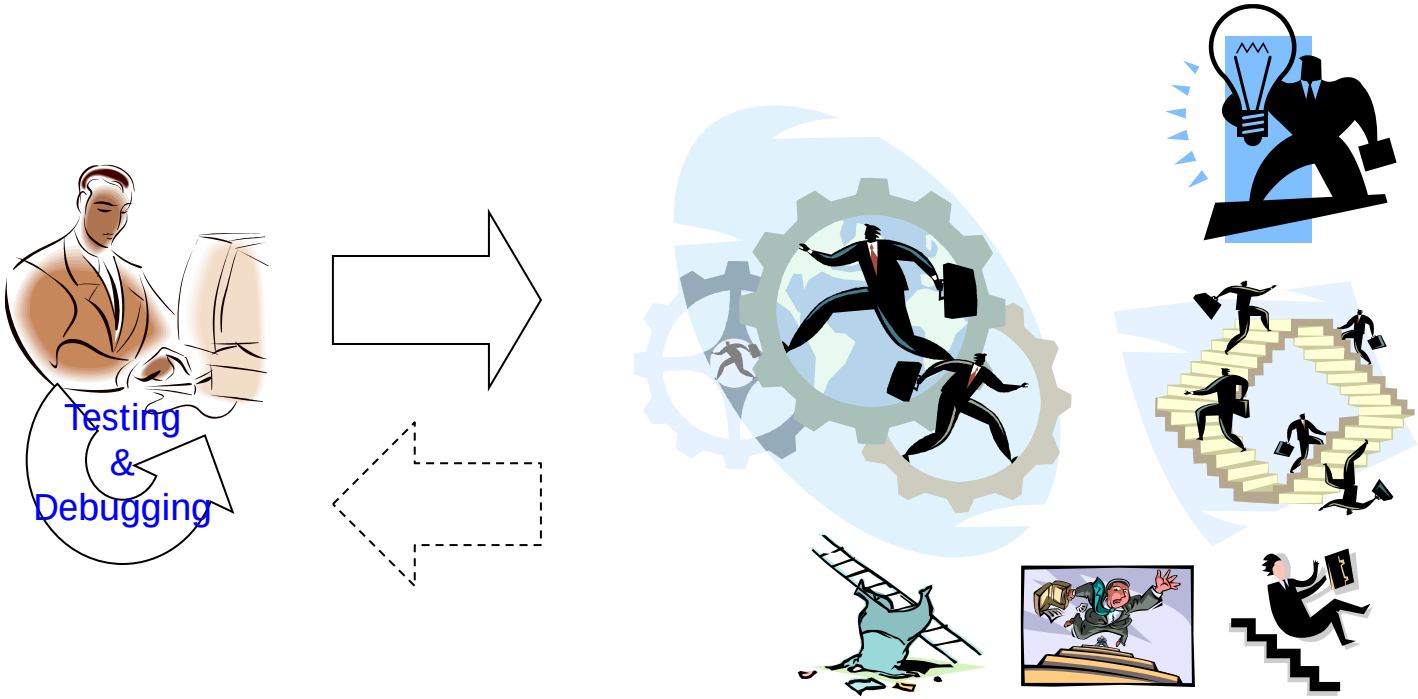
kb-Anonymity: A Model for Anonymized Behavior-Preserving Test and Debugging Data



Aditya Budi, David Lo, Lingxiao Jiang, Lucia

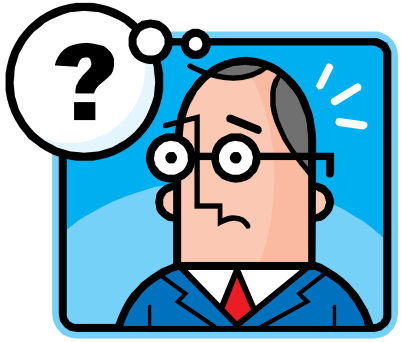
Software Testing & Debugging

- Programs may fail
  - In-house during development process
  - Post-deployment in user fields



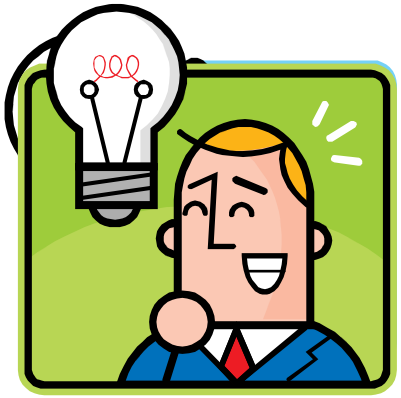
Where Come Inputs for Testing & Debugging?

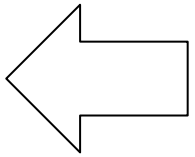
- In-house generation



Where Come Inputs for Testing & Debugging?

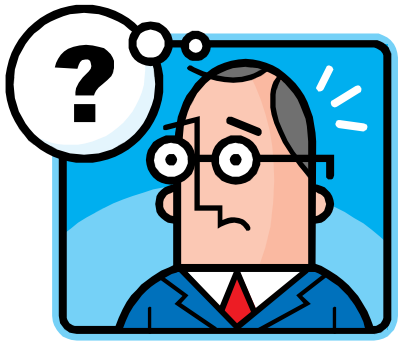
- From clients

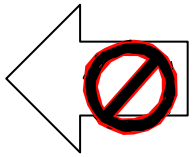




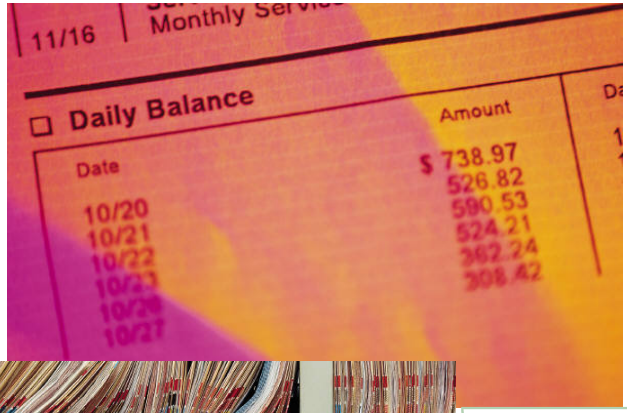
However, Privacy!

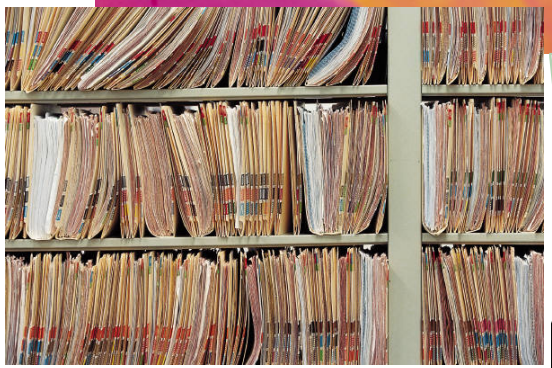
- From clients

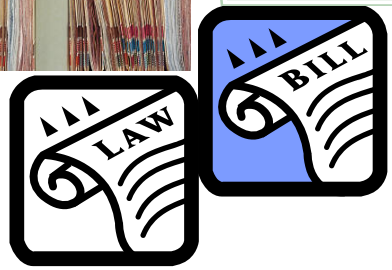




Privacy Concerns!







Sample Privacy Leak

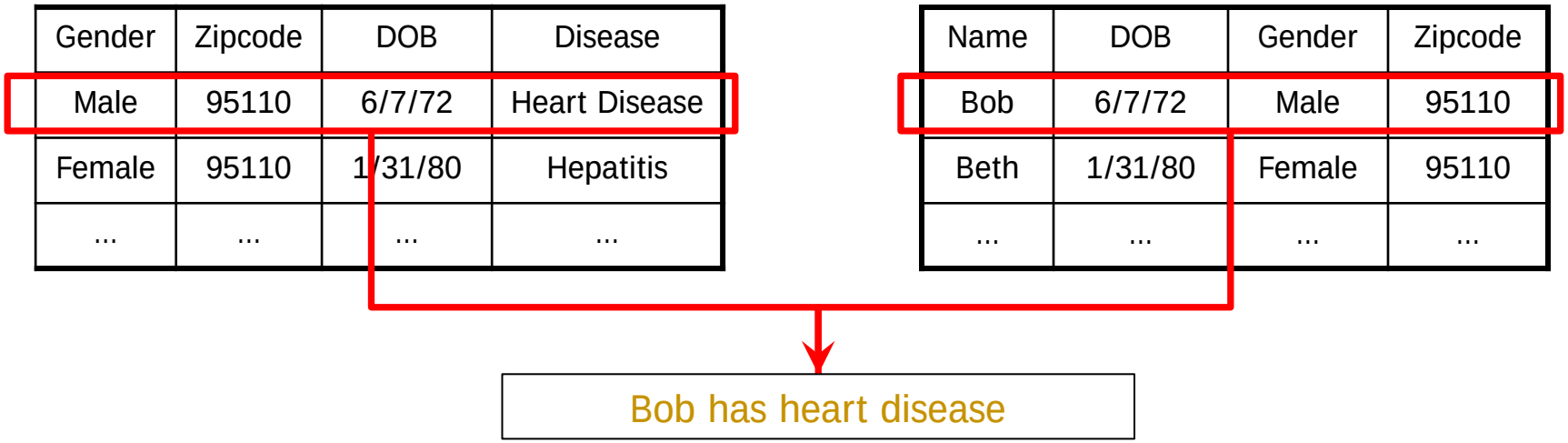
- Linking attack

Patient Records (private)

Gender	Zipcode	DOB	Disease
Male	95110	6/7/72	Heart Disease
Female	95110	1/31/80	Hepatitis
...	...	...	...

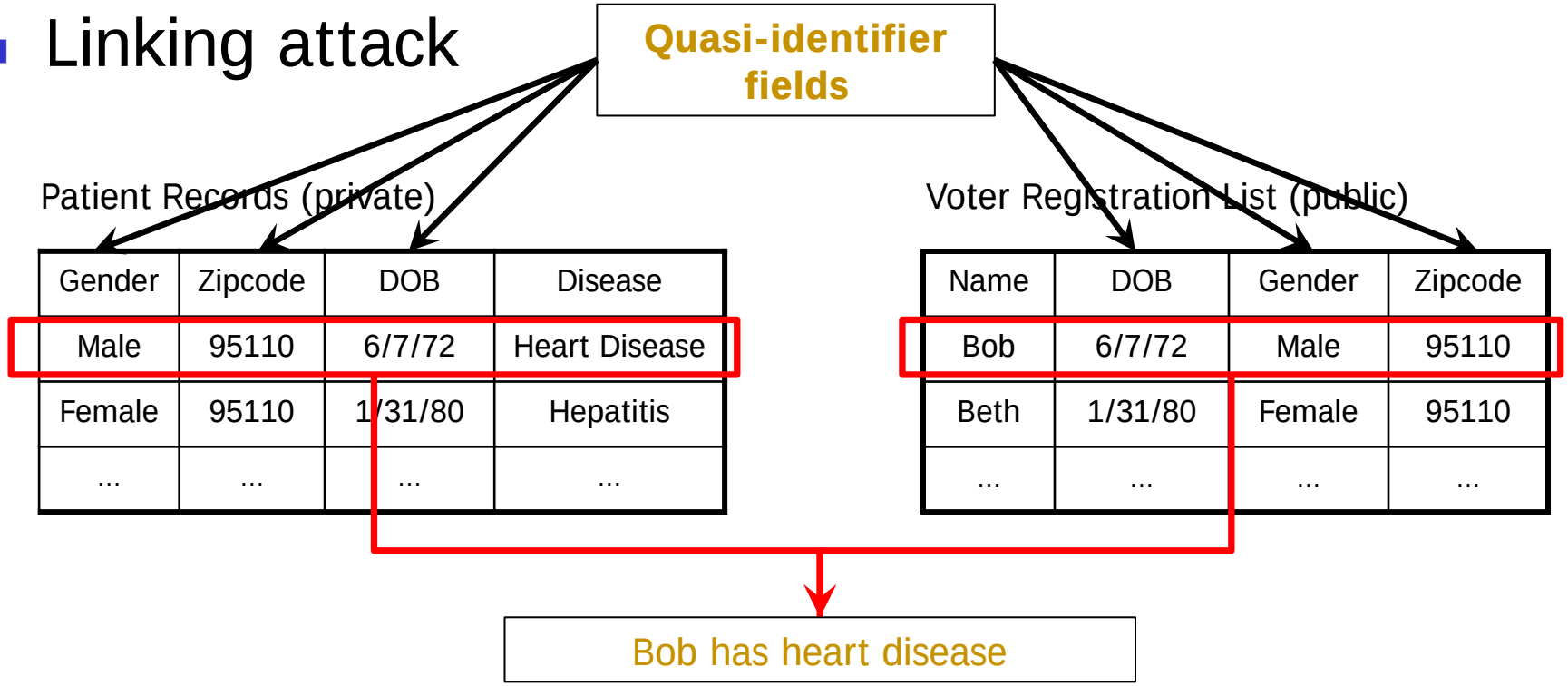
Voter Registration List (public)

Name	DOB	Gender	Zipcode
Bob	6/7/72	Male	95110
Beth	1/31/80	Female	95110
...	...	...	...



Sample Privacy Leak

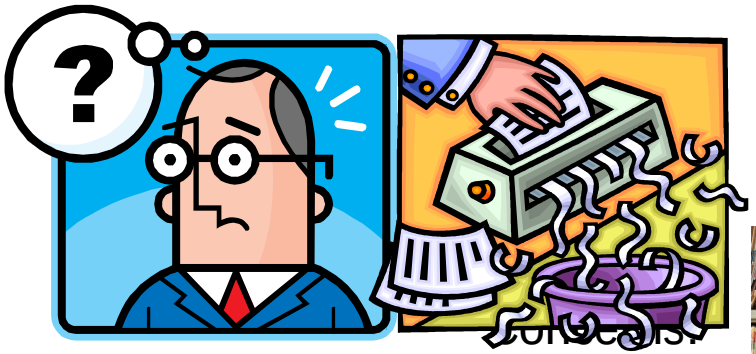
- Linking attack



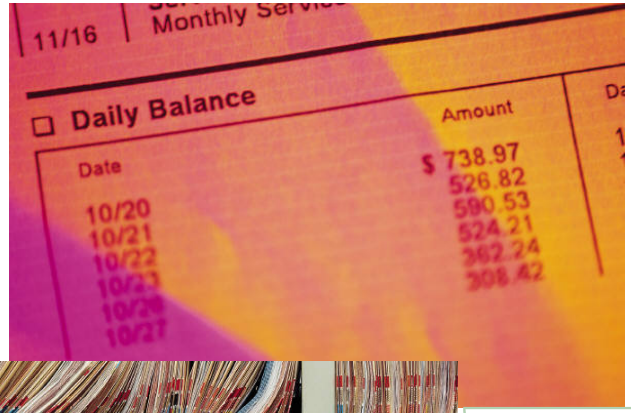
Gender	Zipcode	DOB	Disease
Male	*	*	Heart Disease
Female	*	*	Hepatitis
...	...	...	...

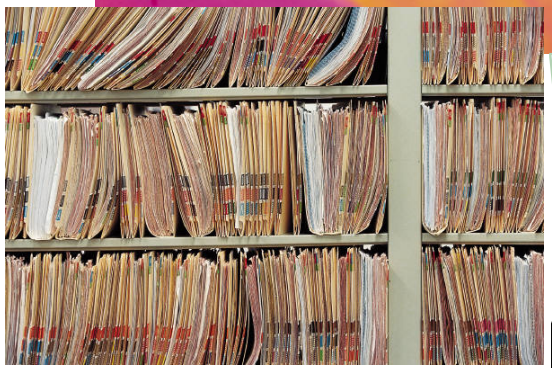
Data Anonymization

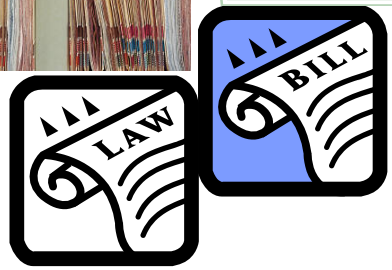
- From clients



Anonymization Function







Data Anonymization Questions

- What to anonymize?

Patient Records (private)

Sex	Zipcode	DOB	Disease
Male	95110	6/7/72	Heart Disease
Female	95110	1/31/80	Hepatitis
...	...	...	...

Sex

Zipcode

DOB

Disease

Data Anonymization Questions

- What to anonymize?
- How to anonymize?

Patient Records (private)

Sex	Zipcode	DOB	Disease
Male	95110	6/7/72	Heart Disease
Female	95110	1/31/80	Hepatitis
...	...	...	...

Sex
Zipcode
DOB
Disease

"Unknown"

Masking 95***, 1972

Generic USA CA, USA San Jose

Random

Data Anonymization Questions

- What to anonymize?
- How to anonymize?
- How useful is the anonymized data for testing and debugging?

Patient Records (private)

Sex	Zipcode	DOB	Disease
Male	95110	6/7/72	Heart Disease
Female	95110	1/31/80	Hepatitis
...	...	...	...

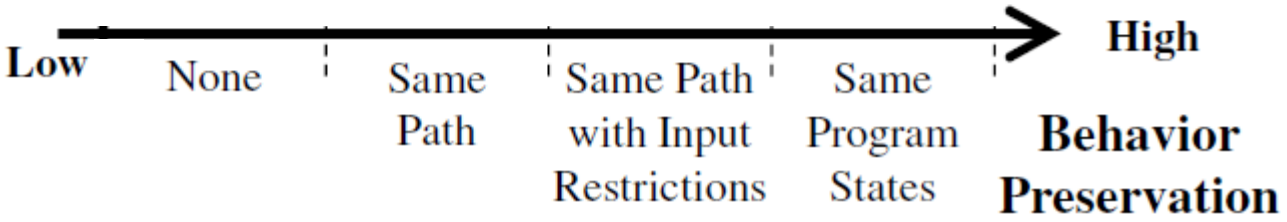
Sex	"Unknown"
Zipcode	Masking 95***, 1972
DOB	Generic USA CA, USA San Jose
Disease	Random

Our Solution

- *kb*-Anonymity: A model that provides guidance on the anonymization questions
  - How to anonymize
    - Follow guidance provided by the ***k*-anonymity** privacy model
      - Each tuple has at least k-1 indistinguishable peers
    - Generate concrete values always
    - Remove indistinguishable tuples
  - How useful is the anonymized data
    - Preserve utility for testing and debugging
    - Each anonymized tuple exhibits certain kinds of **behavior** exhibited by original tuples

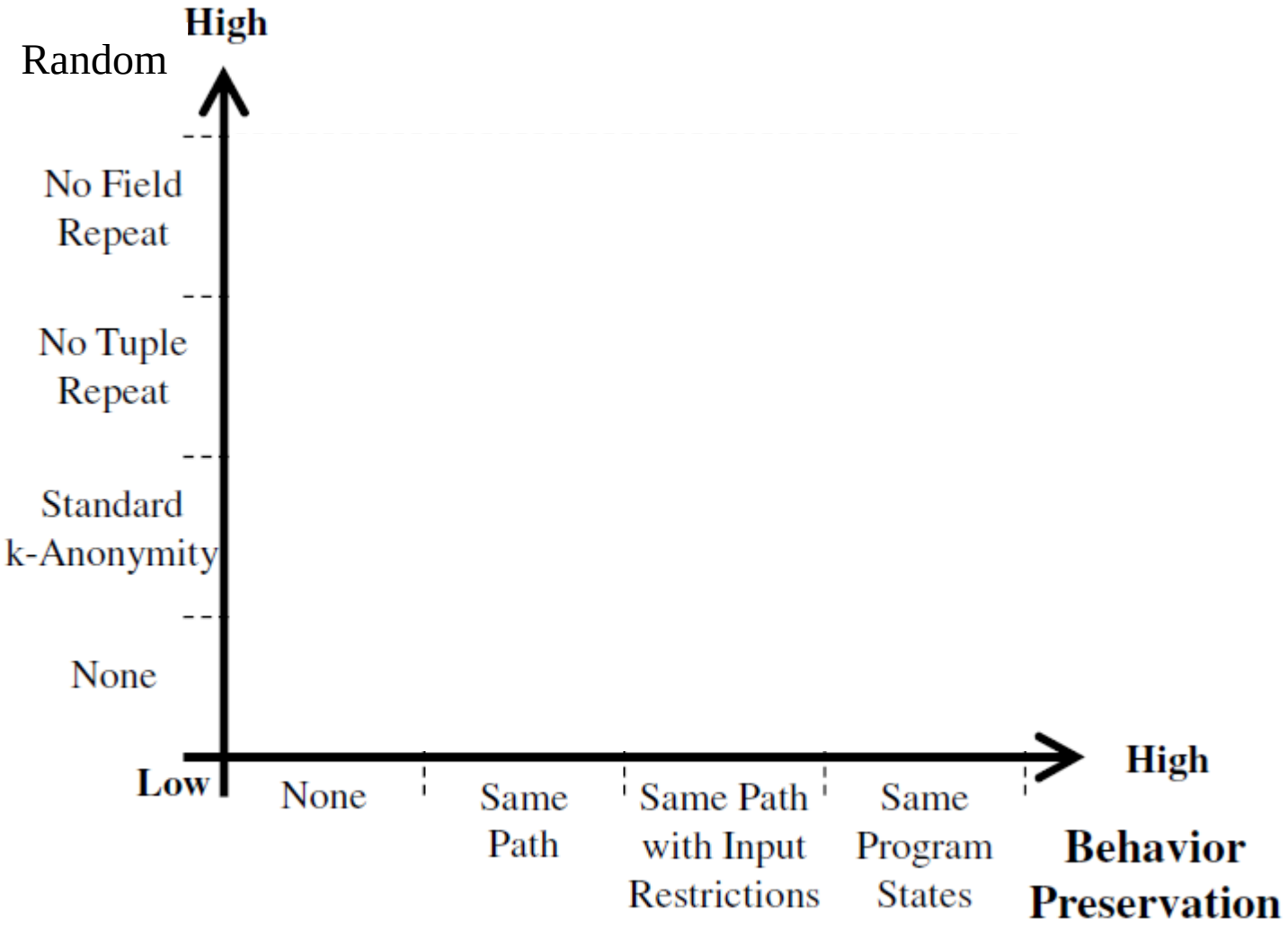
kb-Anonymity

- Behavior preservation



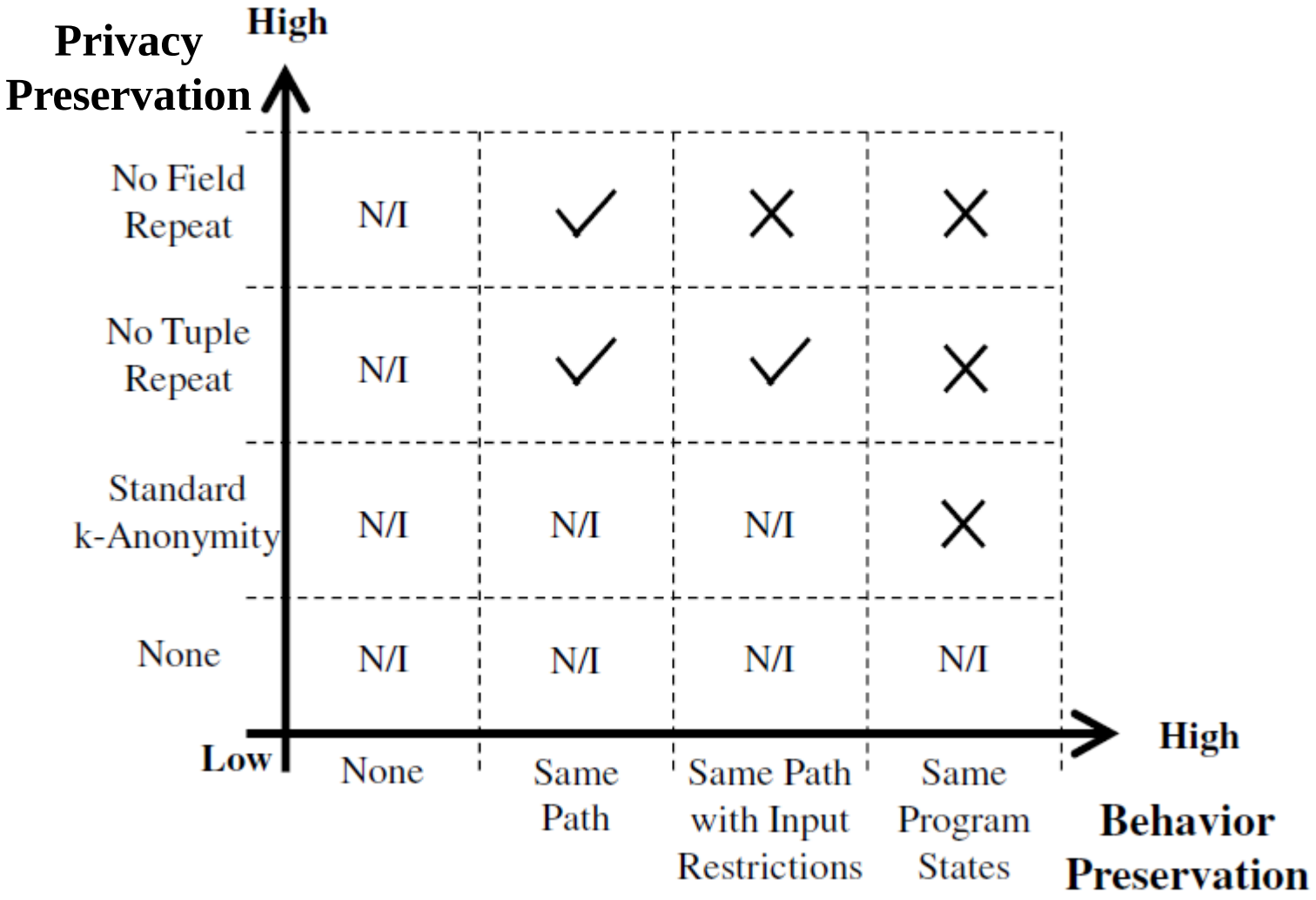
kb-Anonymity

- Privacy preservation



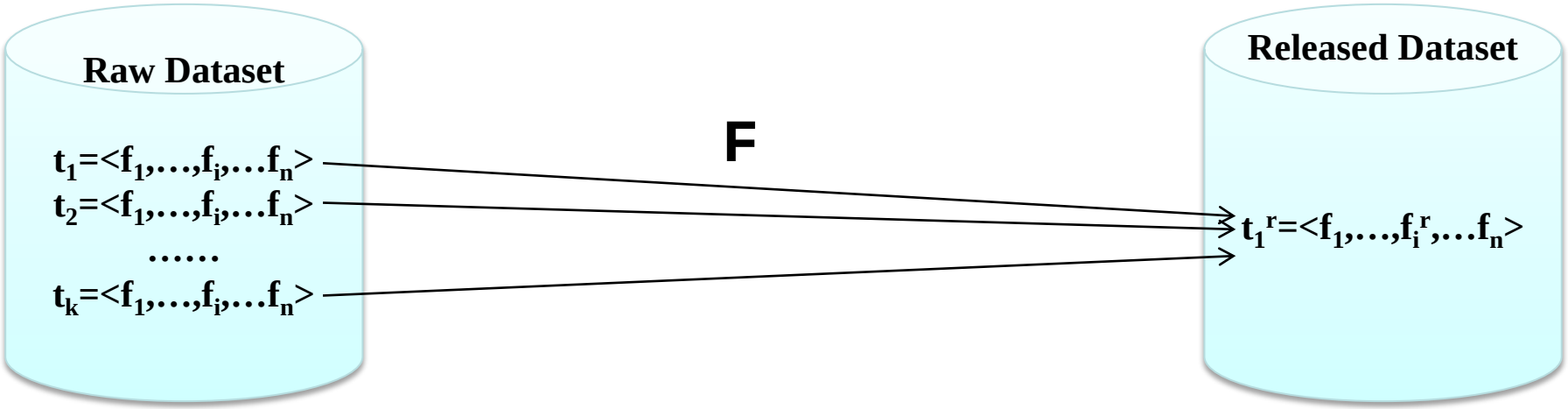
kb-Anonymity

- Behavior and Privacy preservation



kb-Anonymity - Another View

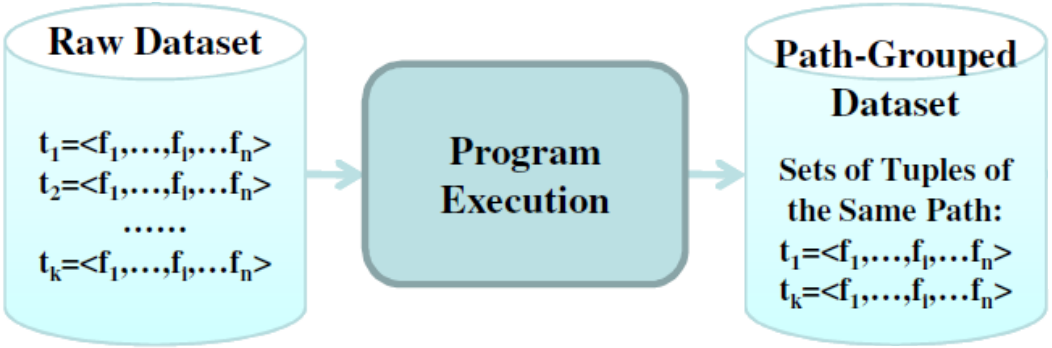
- Anonymization function (i.e., value replacement function) **F**: R → R

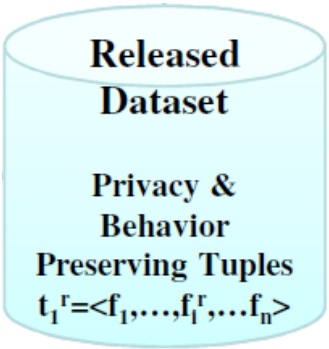


- **Each original tuple is mapped by F to at most one released tuple**
- **At least k original tuples are mapped to the same released tuple**

kb-Anonymity Implementation

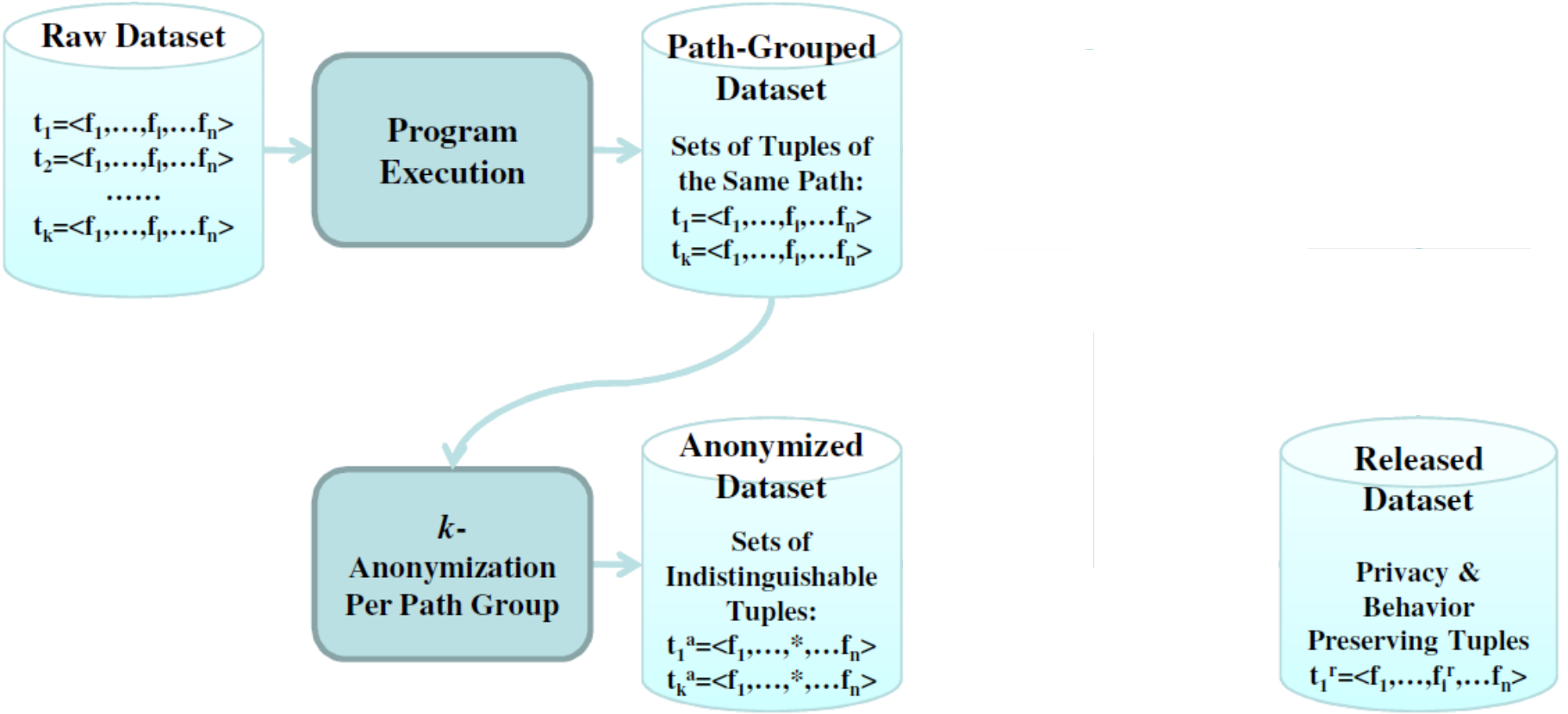
- Dynamic symbolic (a.k.a. concolic) execution with controlled constraint generation and solving





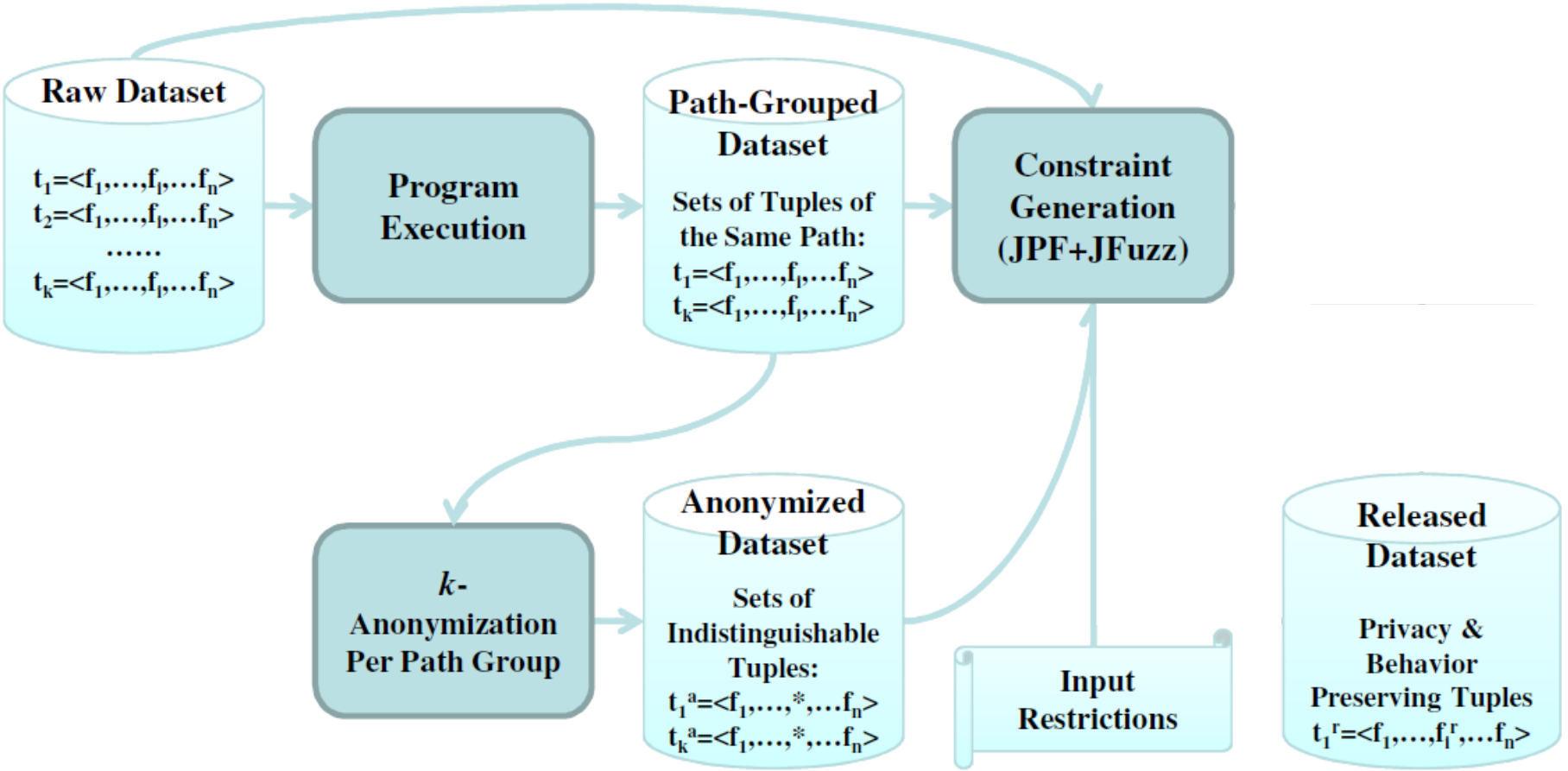
kb-Anonymity Implementation

- Dynamic symbolic (a.k.a. concolic) execution with controlled constraint generation and solving



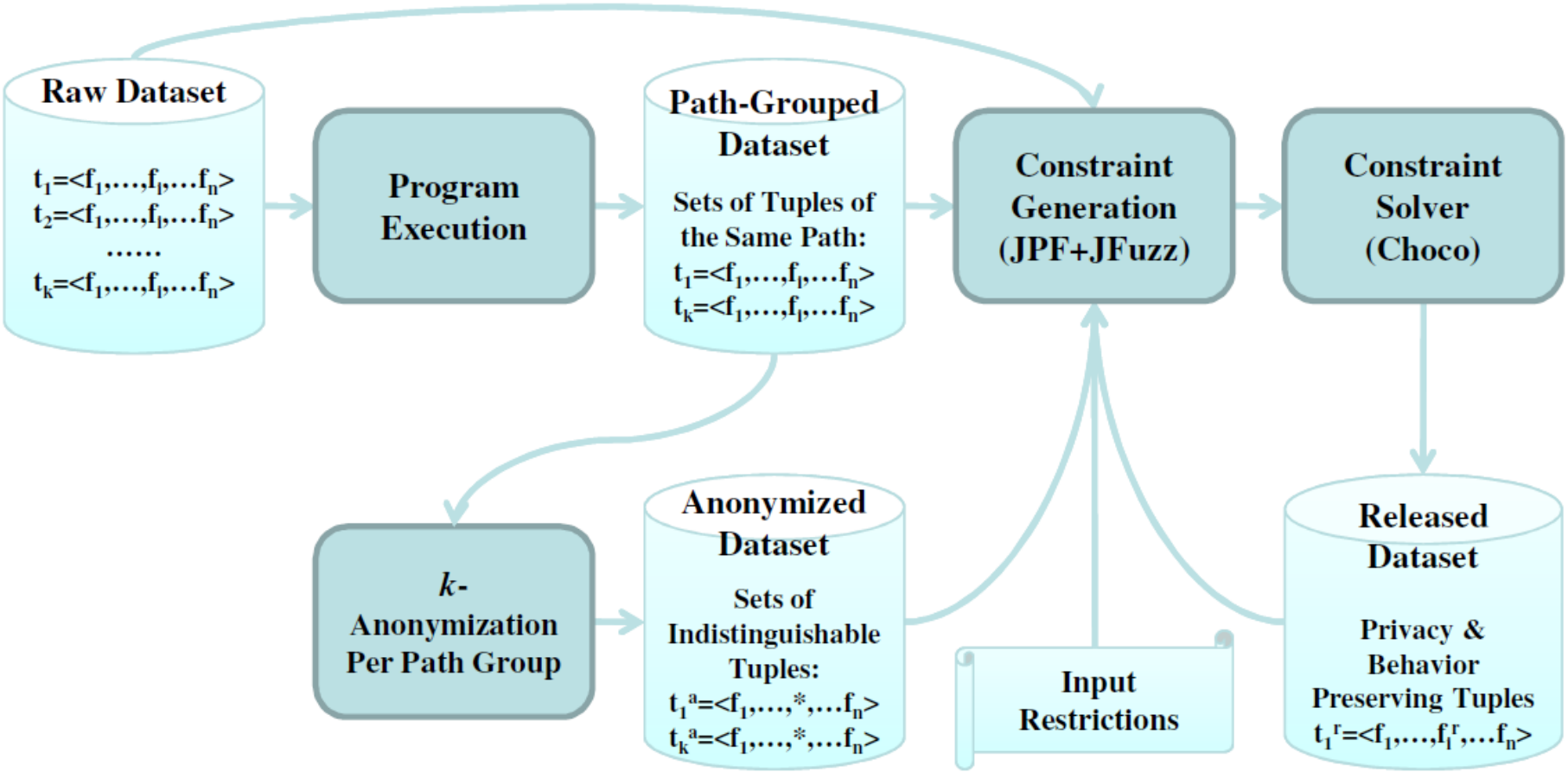
kb-Anonymity Implementation

- Dynamic symbolic (a.k.a. concolic) execution with controlled constraint generation and solving



kb-Anonymity Implementation

- Dynamic symbolic (a.k.a. concolic) execution with controlled constraint generation and solving



Empirical Evaluation

- On slices of open source programs
  - **OpenHospital**, **iTrust**, **PDManager**
  - From sourceforge
  - Modified to deal with integers only
  - Randomly generated test data for anonymization

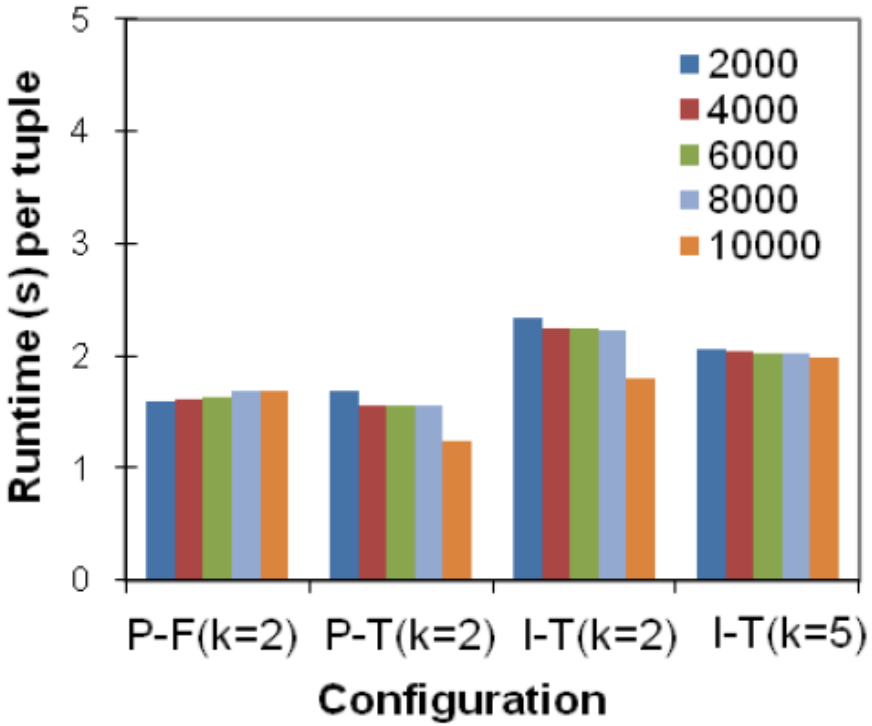
Empirical Evaluation - Utility

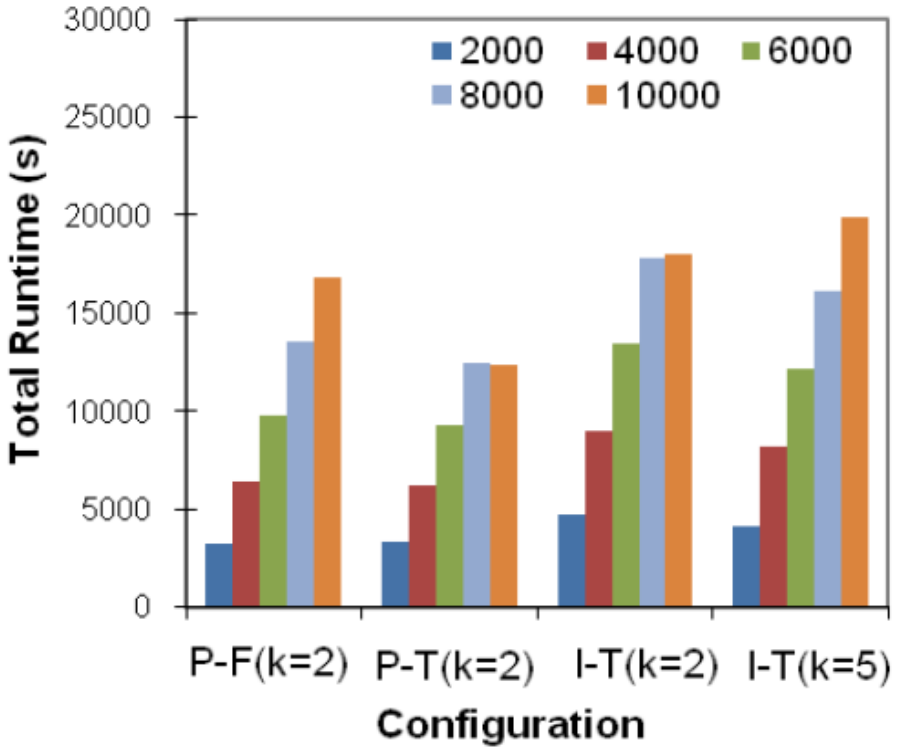
- 16 fields: first name, last name, age, gender, address, city, number of siblings, telephone number, birth date, blood type, mother's name, mother's deceased status, father's name, father's deceased status, insurance status, and whether parents live together.

No	Raw Data Point	Released Tuple
1	⟨90207, 10125, 2, -1, 16261, 22549, 69883, 914, 8201, -2, 68353, -1, -53, -1, -1, -2⟩	⟨-9999, 10000, 0, -10000, 16261, 22549, 69883, 914, 8201, -2, 68353, -1, -53, -1, -1, -2⟩
2	⟨19892, 16536, 78, 1, 36688, 88797, 172, 7519, 50896, -1, 44500, 1, 7452, -2, -1, 1⟩	
3	⟨35778, 21908, 89, -1, 89965, 41493, 35861, 50182, 79181, 1, 30668, -1, 34926, -2, -1, 1 ⟩	
4	⟨9543, 23693, 48, 1, 18133, 75043, -173, 38100, 14912, 1, 69504, 0, 14969, -1, -2, 1⟩	
5	⟨42164, 40607, -6, 1, 46920, 21328, 15089, 42147, 81975, 1, 24382, -2, -252, -2, -1, -1⟩	Error Message

Empirical Evaluation - Scalability

- Running time is proportional to the size of the original data set, and almost constant per tuple.





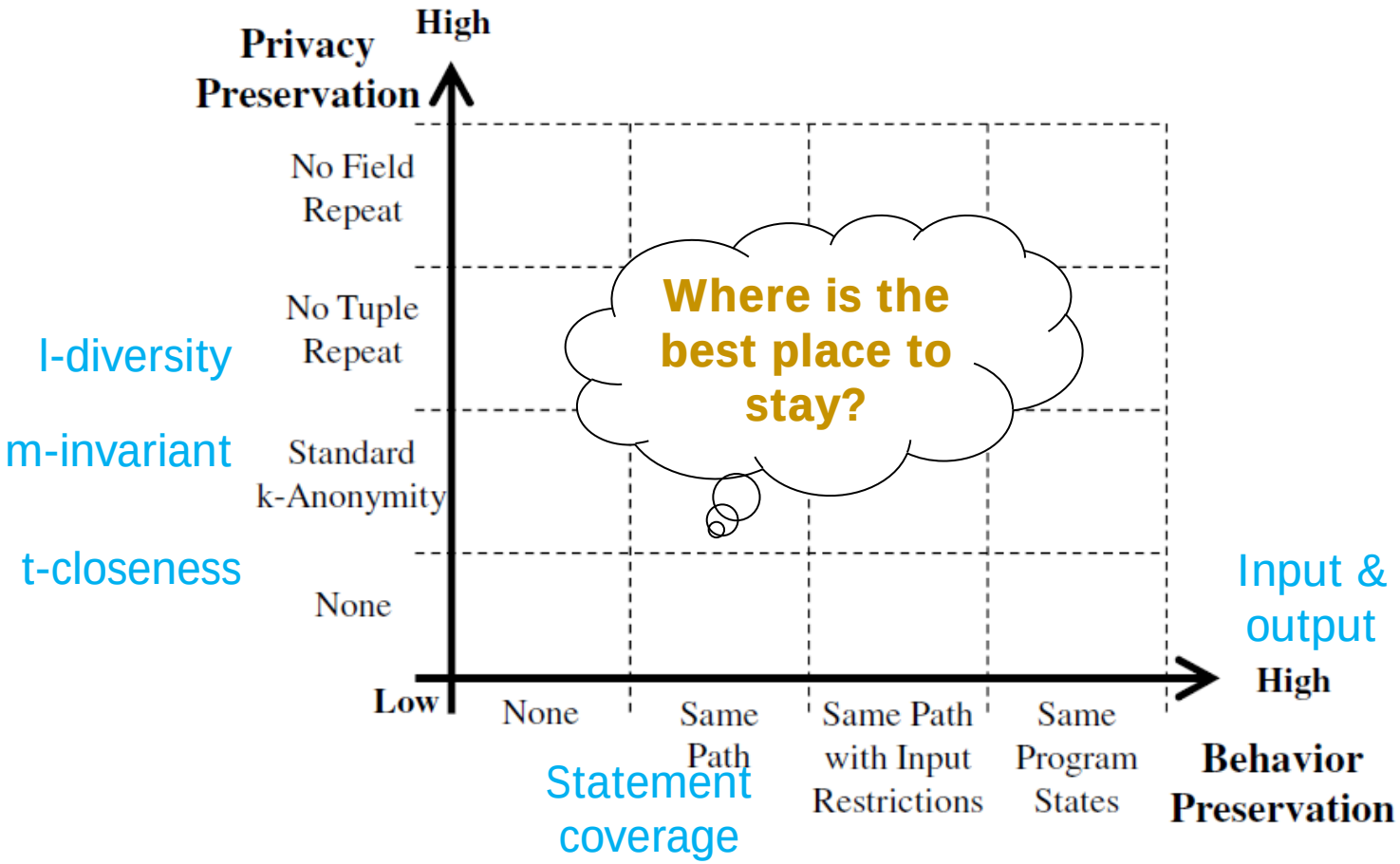
x-axis: different configurations; y-axis: running time in seconds;
Different colors represent the sizes of different original data sets

Limitations

- Selection of quasi-identifiers
  - Reply on data owners to choose appropriate QIs
- Assume each tuple is used independently from other tuples by a program
- Data distortion
  - Do not maintain data statistics, and thus not suitable for data mining or epidemiological studies
- Integer constraints only
  - May handle string constraints based on JPF+jFuzz

Future Work

- Model Refinement
  - Various definitions of behavior preservation
  - Various privacy models



Related Work

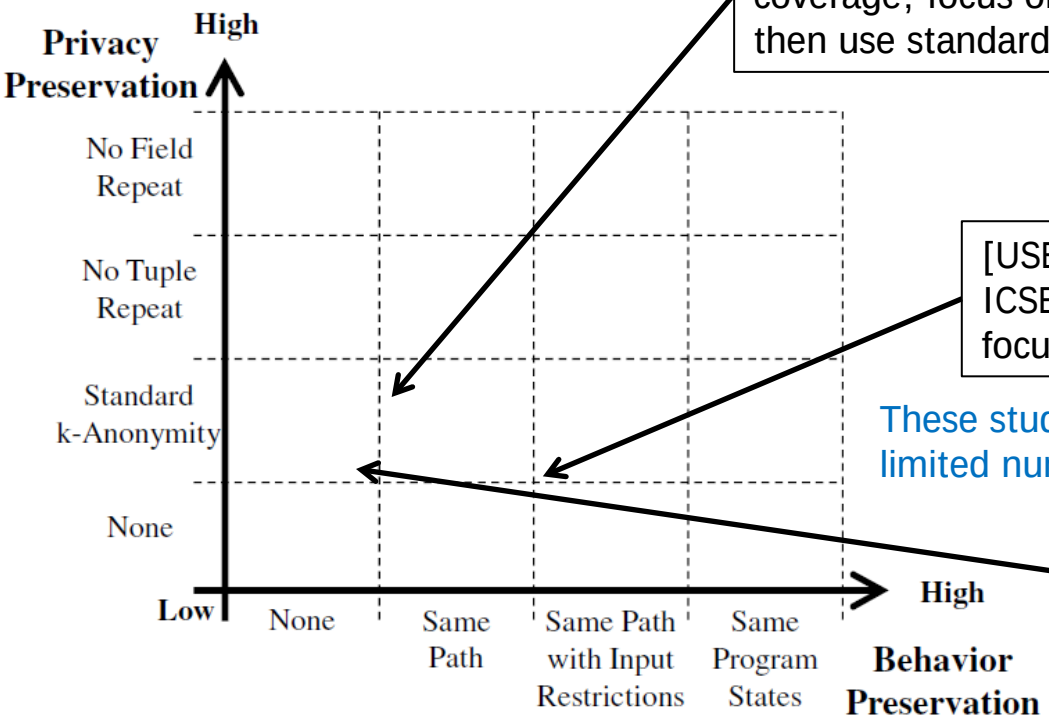
- On concolic execution
  - S. Anand, C. Pasareanu, andW. Visser. **JPF-SE: A symbolic execution extenion to Java PathFinder**. In TACAS, 2007.
  - C. Cadar, D. Dunbar, and D. R. Engler. **KLEE: Unassisted and automatic generation of high-coverage tests for complex systems programs**. In OSDI, pages 209–224, 2008.
  - P. Godefroid, N. Klarlund, and K. Sen. **DART: Directed automated random testing**. In PLDI, pages 213–223. ACM, 2005.
  - K. Jayaraman, D. Harvison, V. Ganesh, and A. Kiezun. **jFuzz: A concolic tester for NASA Java**. In NASA Formal Methods Workshop, 2009.
  - K. Sen, D. Marinov, and G. Agha. **CUTE: A concolic unit testing engine for C**. In FSE, pages 263–272, 2005.

Related Work

- On privacy-preserving testing & debugging
  - Pete Broadwell, Matt Harren, and Naveen Sastry. **Scrash: A system for generating secure crash information**. In USENIX Security 2003.
  - Miguel Castro, Manuel Costa, and Jean-Philippe Martin. **Better Bug Reporting With Better Privacy**. In ASPLOS 2008
  - James Clause and Alessandro Orso. **Camouflage: Automated Anonymization of Field Data**. In ICSE 2011.
  - Mark Grechanik, Christoph Csallner, Chen Fu, and Qing Xie. **Is Data Privacy Always Good For Software Testing?** In ISSRE 2010.
  - Rui Wang, Xiaofeng Wang, and Zhuowei Li. **Panalyst: Privacy-aware remote error analysis on commodity software**. In USENIX Security 2008.

Related Work

- On privacy-preserving testing & debugging



[ISSRE 2010] consider same statement coverage; focus on choosing better QIs, then use standard *k*-anonymity algorithm

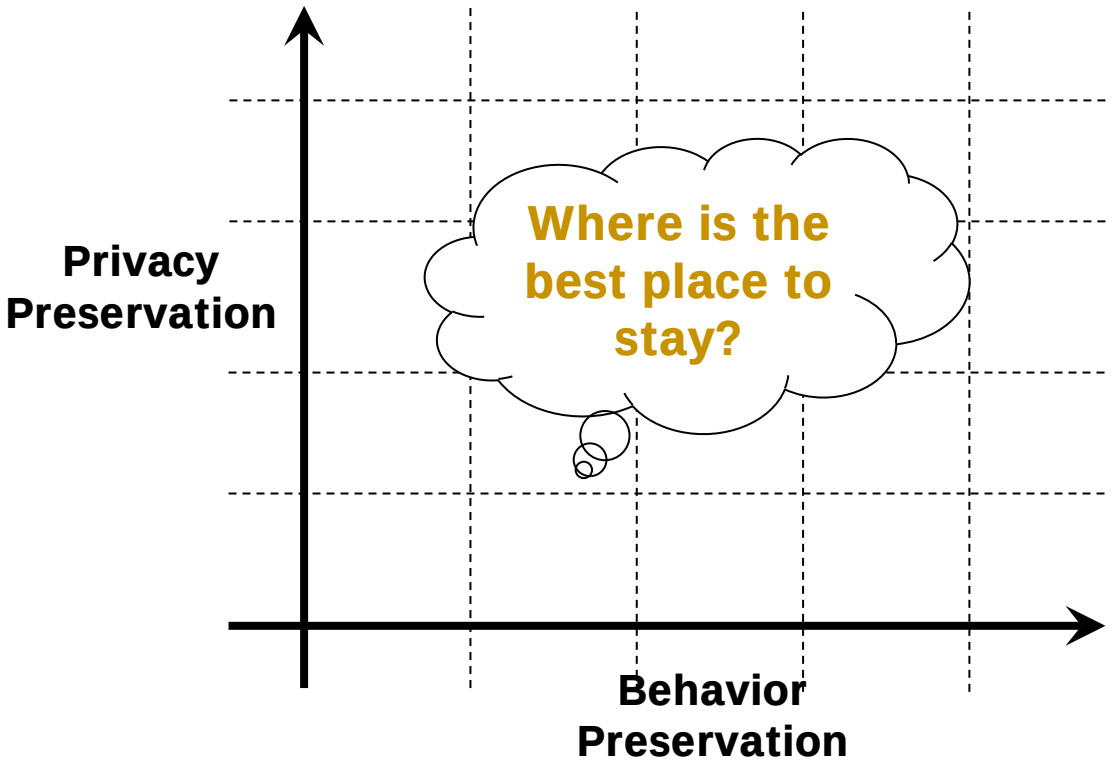
[USENIX Security 2008, ASPLOS 2008, ICSE 2011] consider path conditions; focus on anonymizing a single tuple

These studies complement ours in cases when only a limited number of failed test inputs are considered.

[USENIX Security 2003] focus on anonymizing a single tuple only

Conclusion

- *kb*-Anonymity: A model that guides data anonymization for software testing and debugging purposes.



Thank you!

Questions?

{adityabudi, davidlo, lxjiang, lucia.2009}@smu.edu.sg