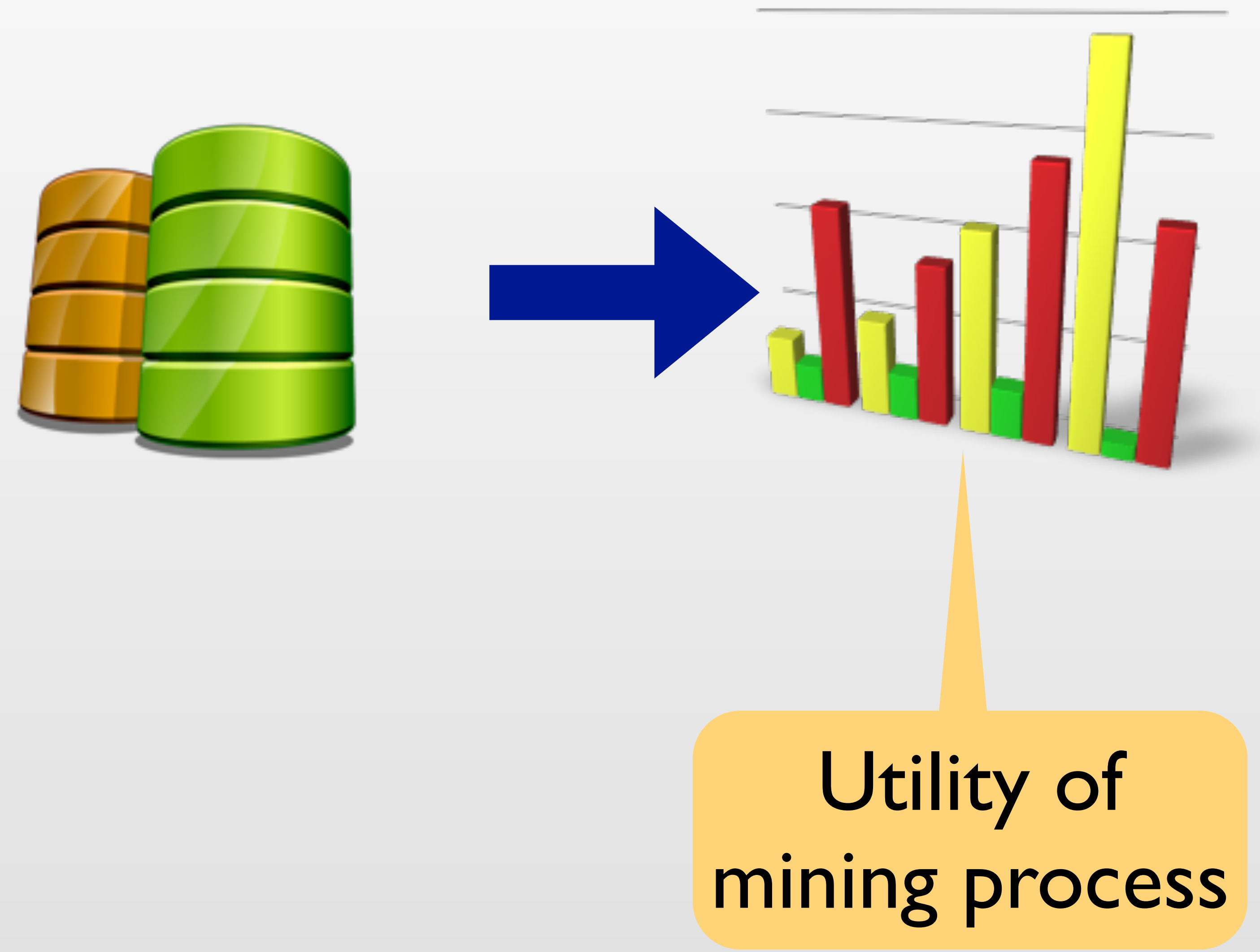
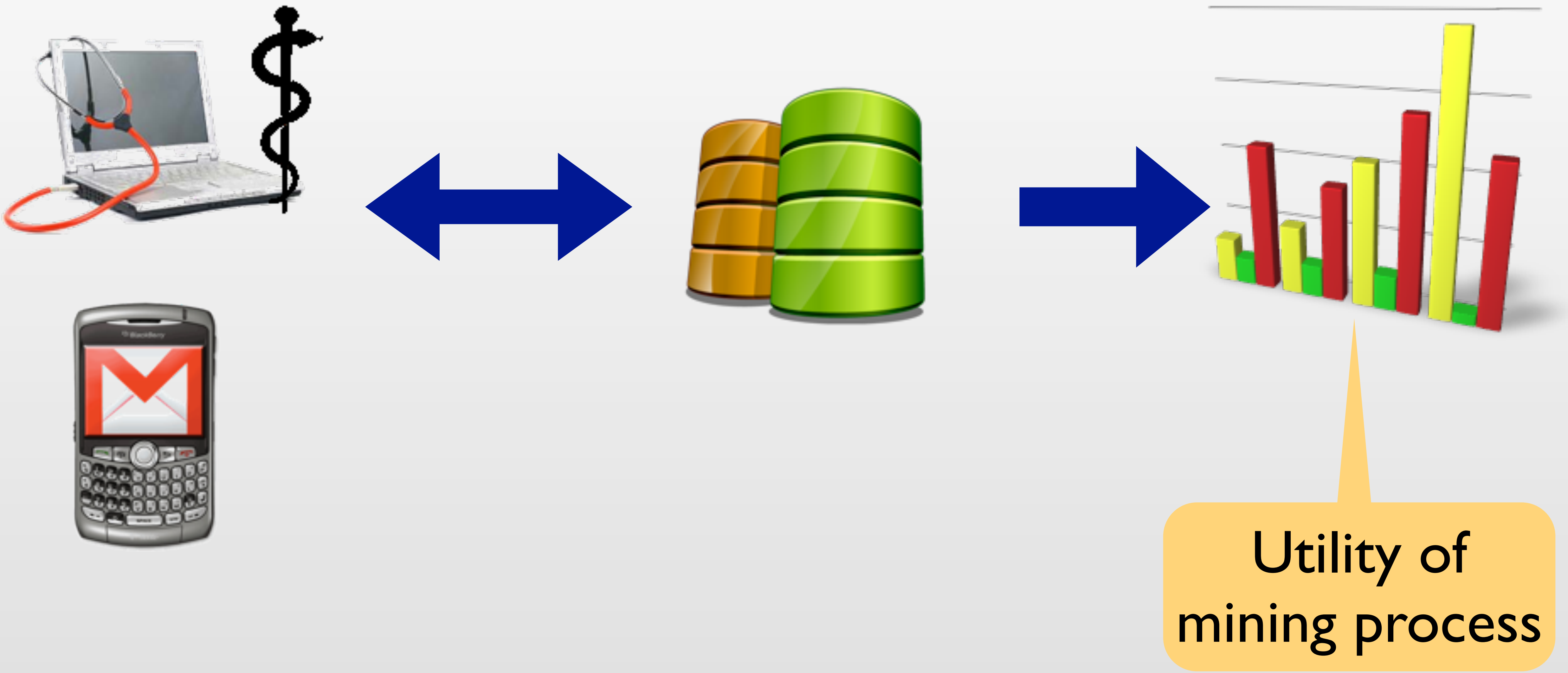
Probabilistic Relational Reasoning for Differential Privacy

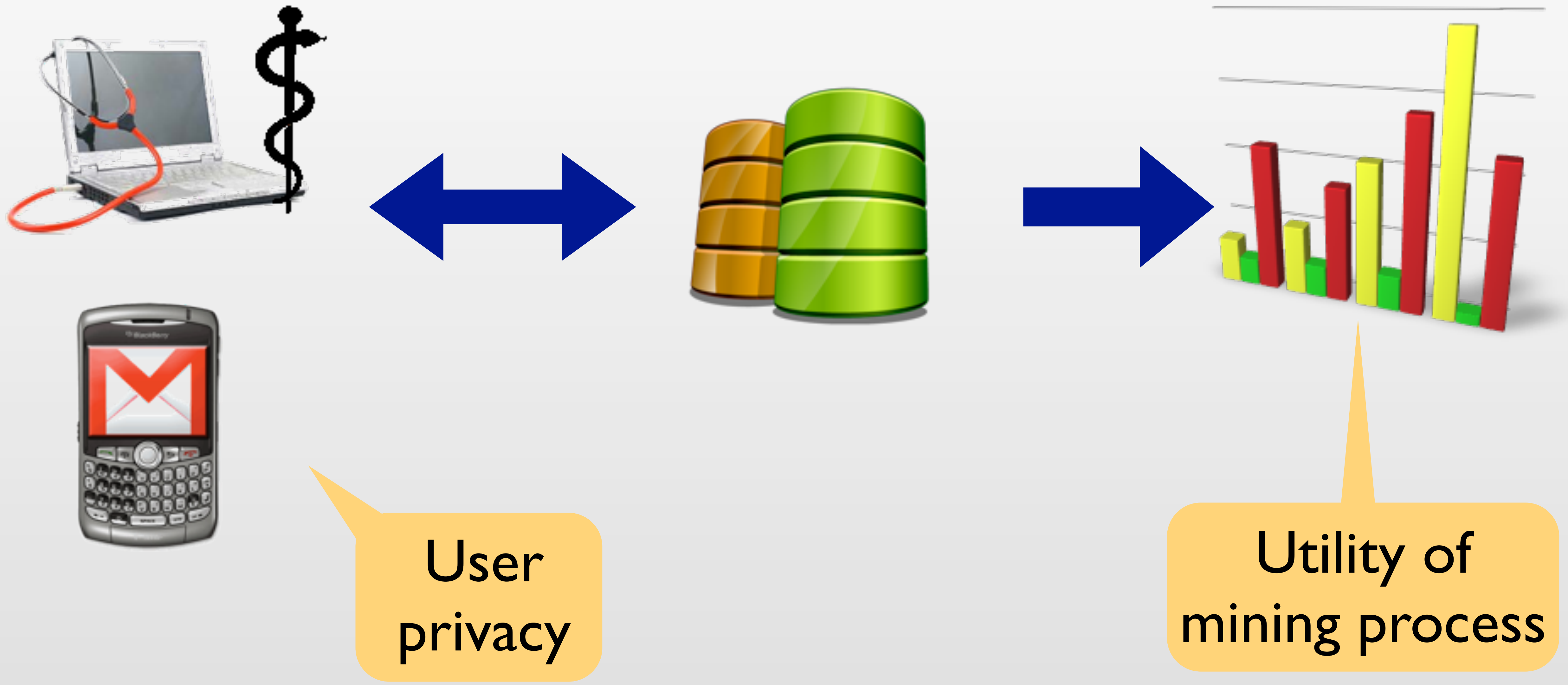
Gilles Barthe Boris Köpf
Federico Olmedo Santiago Zanella Béguelin

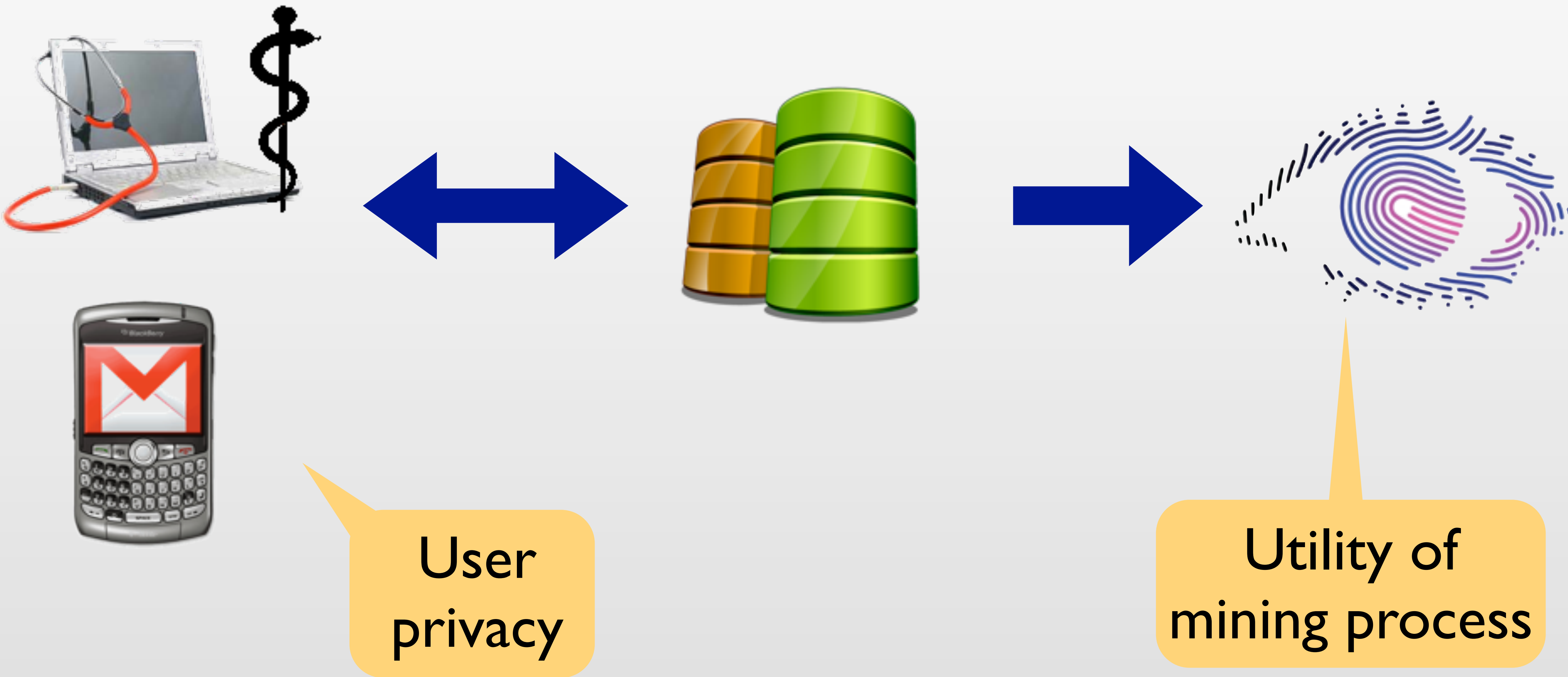
IMDEA Software Institute, Madrid

POPL 2012

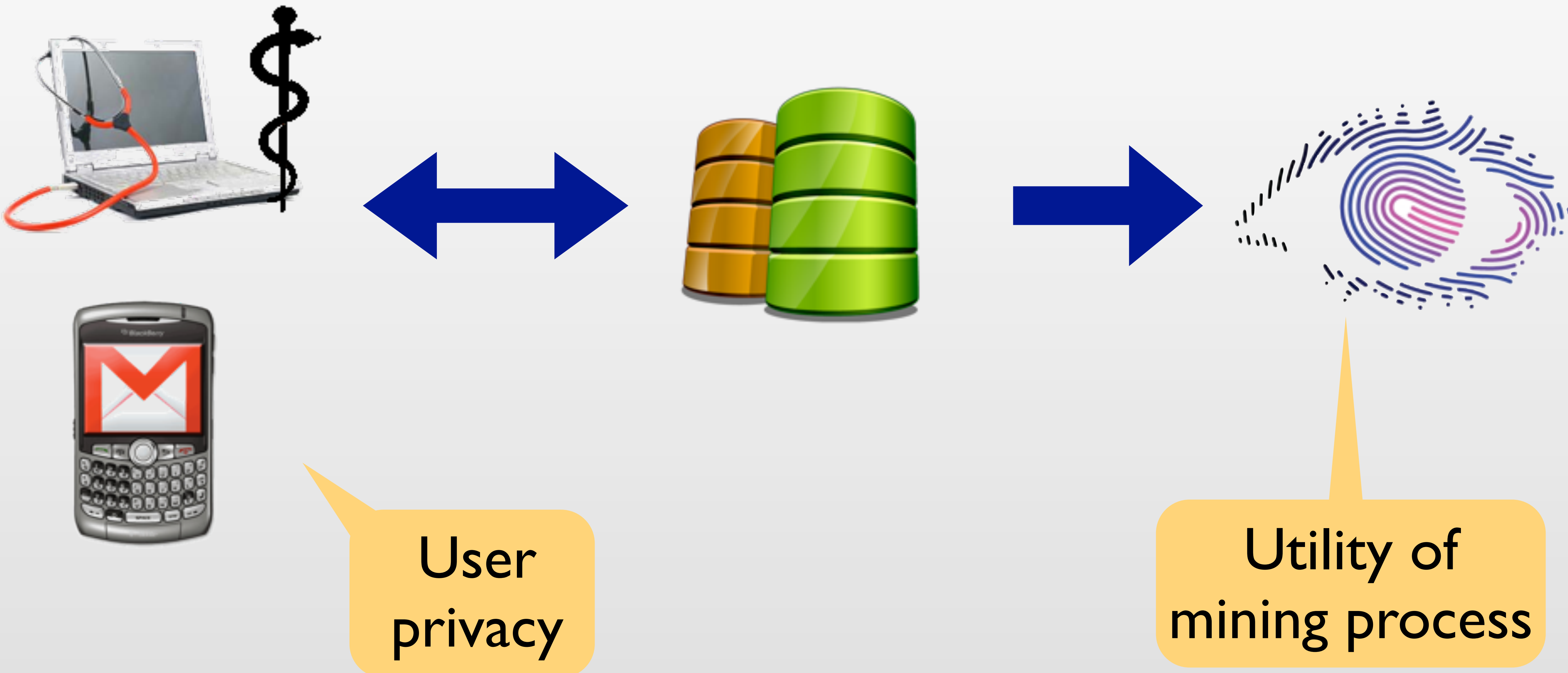

Utility of
mining process


Utility of
mining process

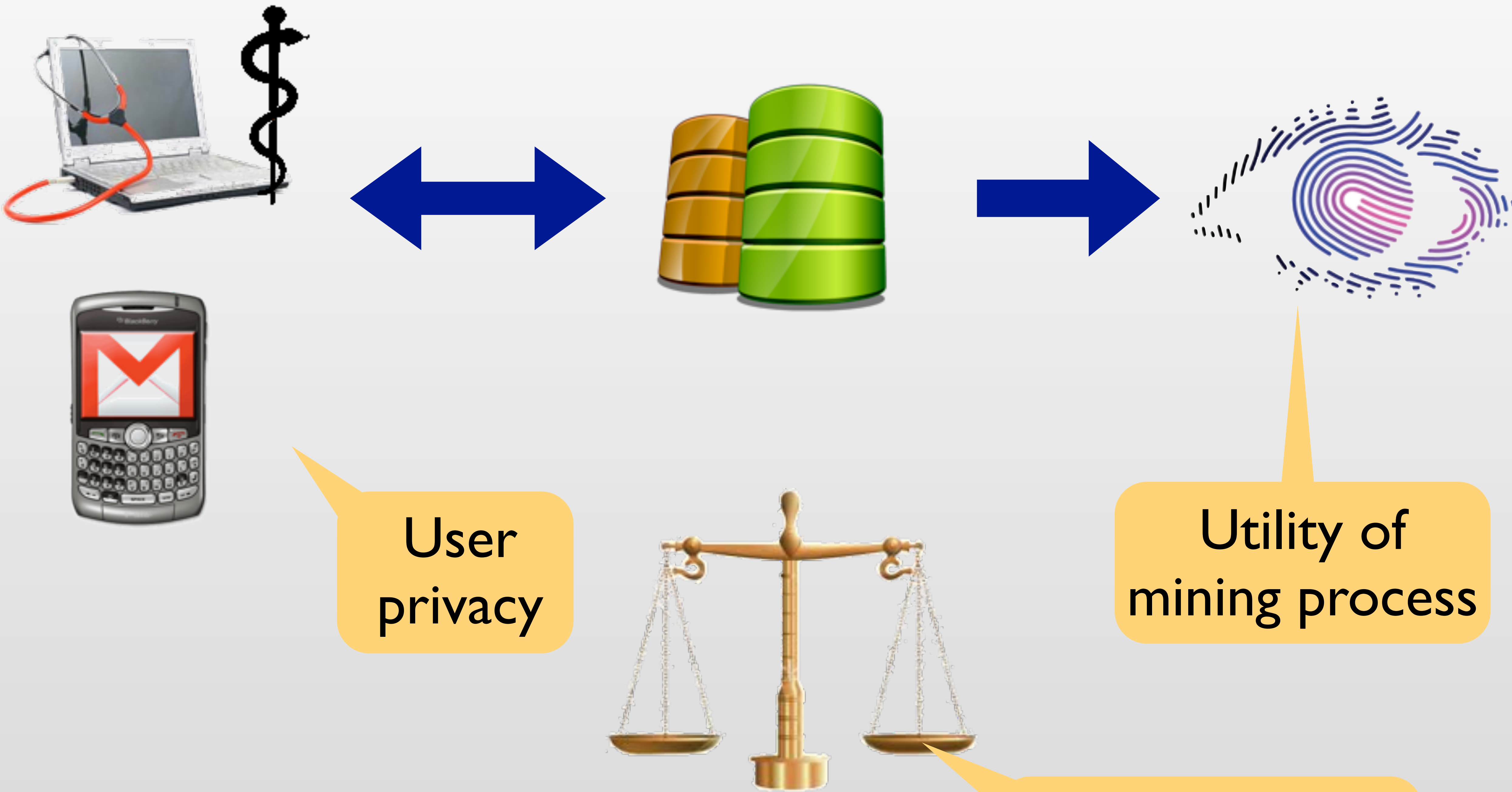

User privacy
Utility of mining process


User privacy
Utility of mining process

Conflicting requirements!



Conflicting requirements!



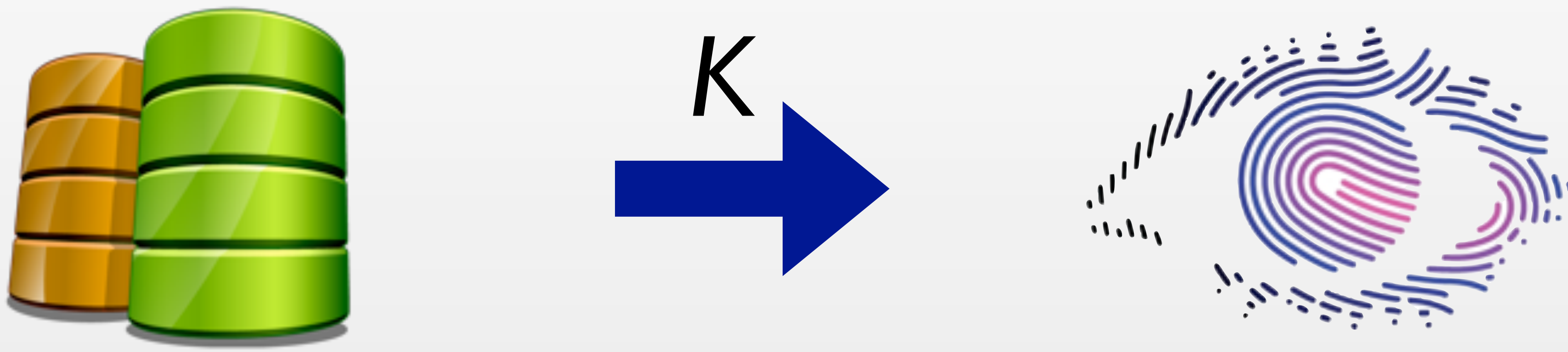
User privacy

Utility of mining process

Need to achieve flexible balance

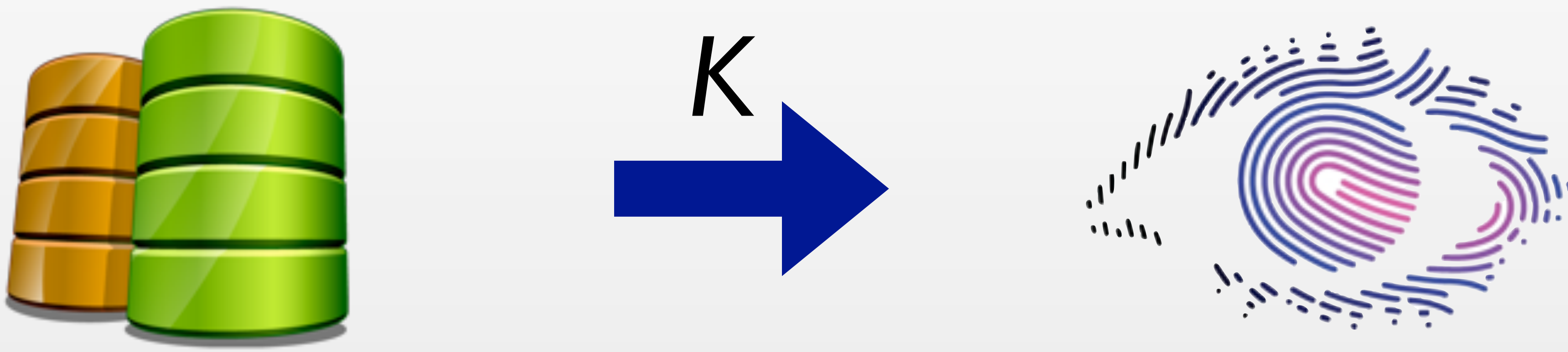
Differential Privacy

Dwork [ICALP'06]



Differential Privacy

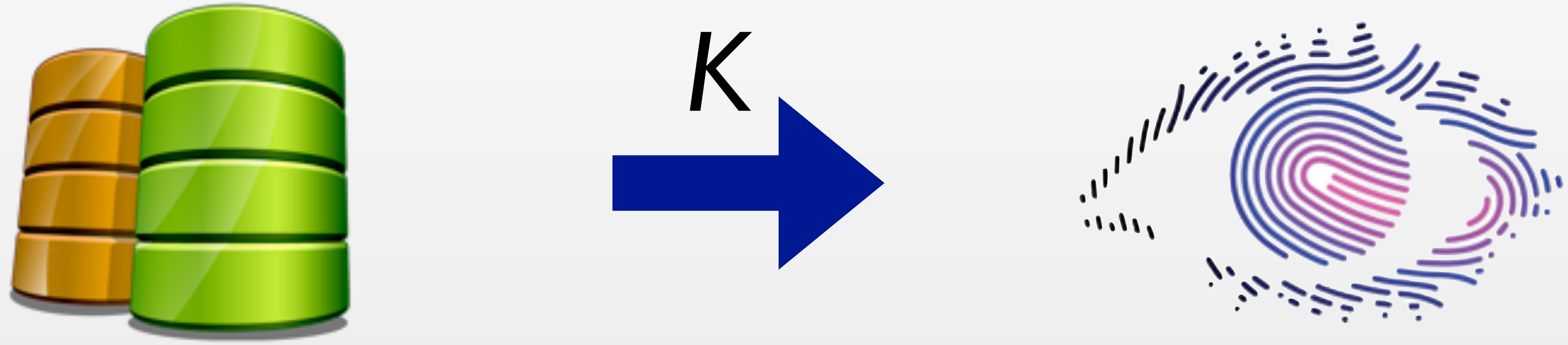
Dwork [ICALP'06]



- Fix a (symmetric) adjacency relation Φ on databases

Differential Privacy

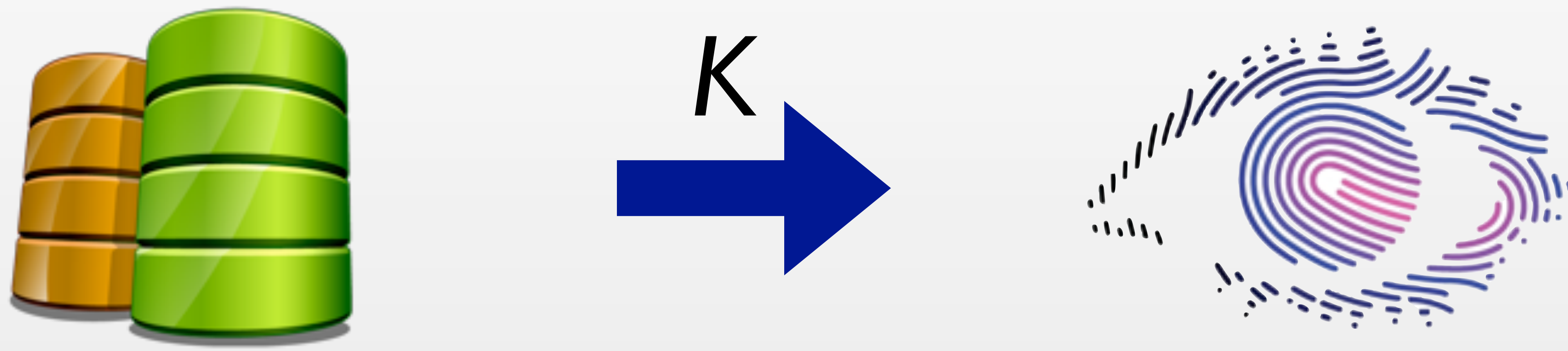
Dwork [ICALP'06]



- Fix a (symmetric) adjacency relation Φ on databases
- Fix a privacy budget ε

Differential Privacy

Dwork [ICALP'06]



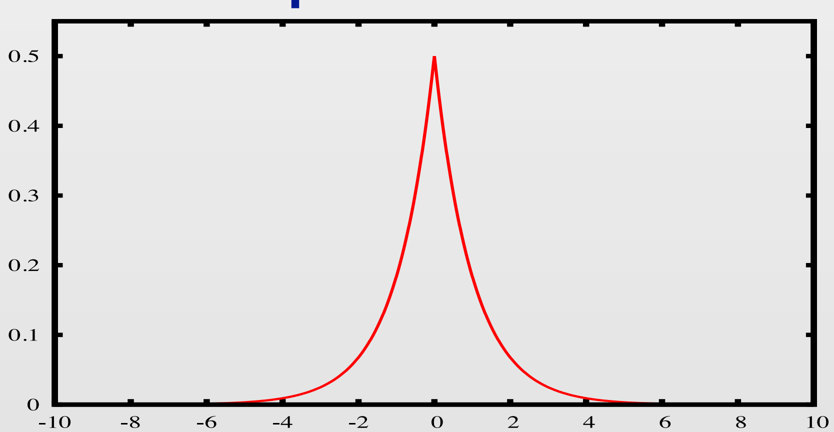
- Fix a (symmetric) adjacency relation Φ on databases
- Fix a privacy budget ε

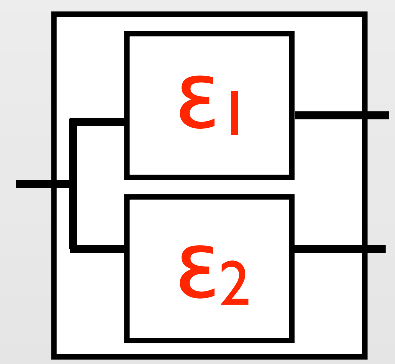
A randomized algorithm *K* is ε-differentially private w.r.t. Φ iff, for all databases D_1 and D_2, and events S

$$\Phi(D_1, D_2) \Longrightarrow \Pr[K(D_1) \in S] \leq \exp(\epsilon) \times \Pr[K(D_2) \in S]$$

Differential Privacy Primer

- Fundamentals
  - Laplacian mechanism
  - Composition theorems

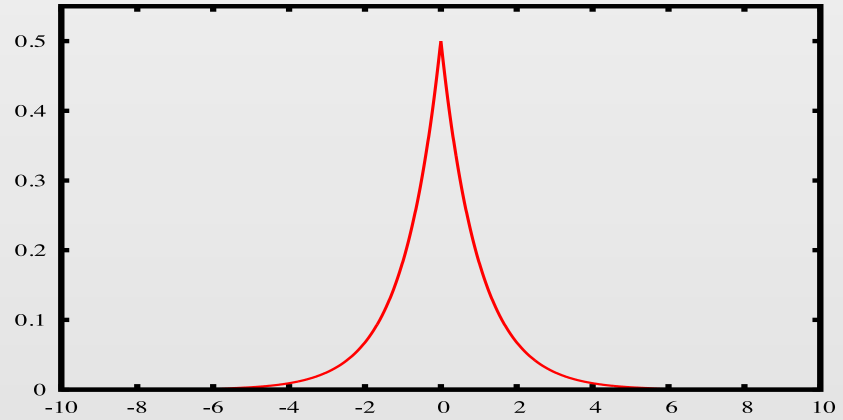


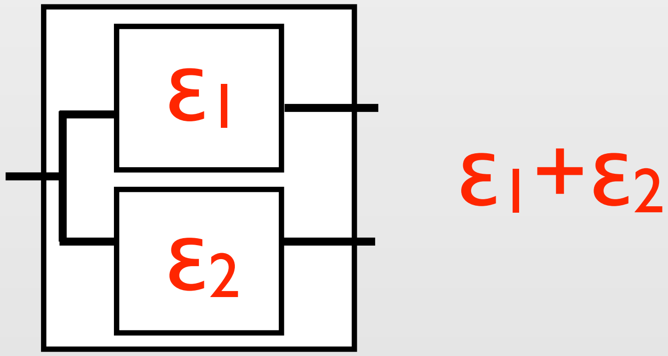


$\varepsilon_1 + \varepsilon_2$

Differential Privacy Primer

- Fundamentals
  - Laplacian mechanism
  - Composition theorems

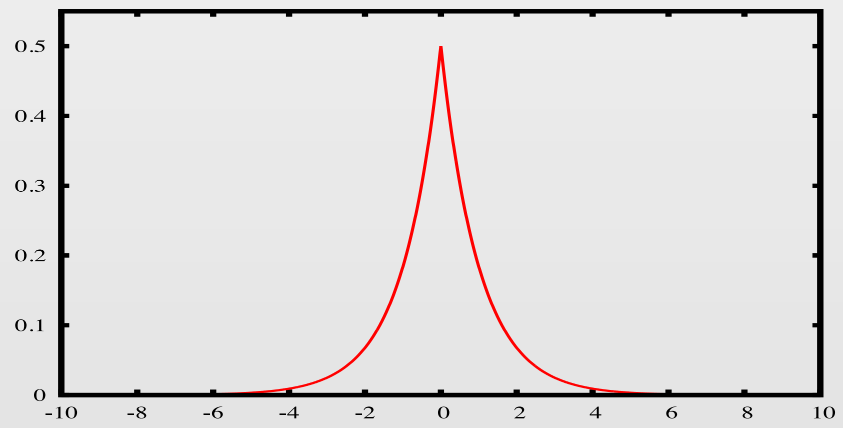


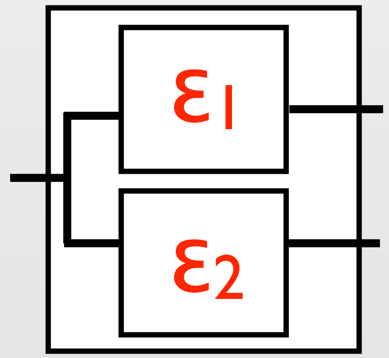


Differential Privacy Primer

- Fundamentals
  - Laplacian mechanism
  - Composition theorems

Language-based tool support available

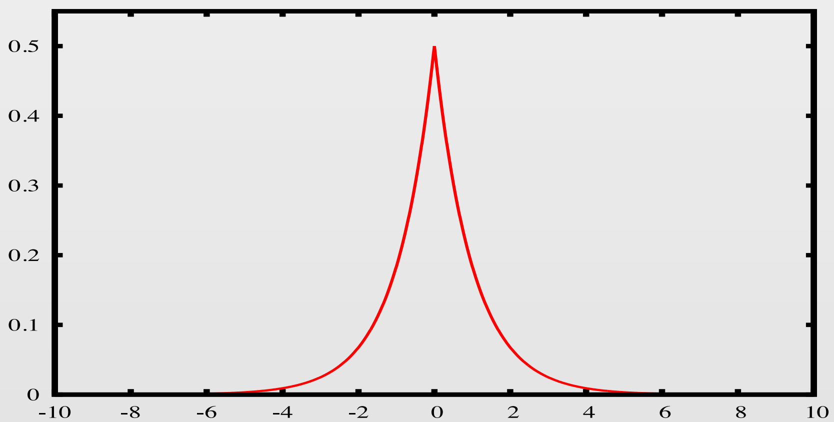


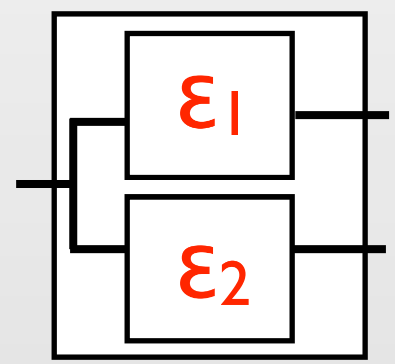


$\varepsilon_1+\varepsilon_2$

Differential Privacy Primer

- Fundamentals
  - Laplacian mechanism
  - Composition theorems





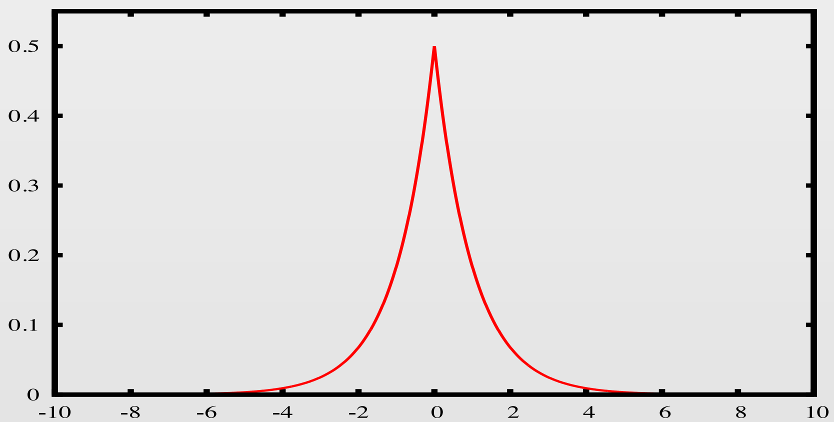
$\varepsilon_1+\varepsilon_2$

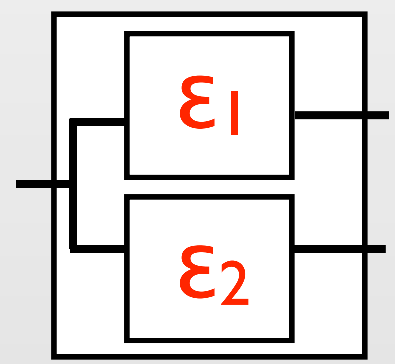
Language-based tool support available

- Expanding frontiers
  - Mechanisms: exponential, median...
  - Algorithms: streaming/graph/... algorithms
  - Definitions: approximate differential privacy, pan privacy...

Differential Privacy Primer

- Fundamentals
  - Laplacian mechanism
  - Composition theorems





$\varepsilon_1+\varepsilon_2$

Language-based tool support available

- Expanding frontiers
  - Mechanisms: exponential, median...
  - Algorithms: streaming/graph/... algorithms
  - Definitions: approximate differential privacy, pan privacy...

Increasingly complex, but not supported by existing tools!

Our Contribution: CertiPriv

- Allows reasoning about approximate quantitative properties of randomized computations
  - Built from first principles and fully formalized in Coq
  - Machine-checked proofs of differential privacy
    - Correctness of Laplacian and Exponential mechanisms
    - State-of-art graph and streaming algorithms
- Generalizes CertiCrypt and opens new applications to crypto

Differential privacy as quantitative 2-safety

- K is (ε, δ)-diff. private w.r.t. Φ iff for all D_1 and D_2 and S

$$\Phi(D_1, D_2) \Longrightarrow \Pr[K(D_1) \in S] \leq \exp(\epsilon) \times \Pr[K(D_2) \in S] + \delta$$

Relational pre-condition

(Quantitative) relational post-condition

Differential privacy as quantitative 2-safety

- K is (ε, δ)-diff. private w.r.t. Φ iff for all D_1 and D_2 and S

$$\Phi(D_1, D_2) \Longrightarrow \Pr[K(D_1) \in S] \leq \exp(\epsilon) \times \Pr[K(D_2) \in S] + \delta$$

Relational pre-condition

(Quantitative) relational post-condition

- We propose a quantitative probabilistic relational Hoare Logic

$$c_1 \sim_{\alpha,\delta} c_2 : \Phi \Longrightarrow \Psi$$

such that c is (ε, δ)-diff. private w.r.t. Φ iff

$$c \sim_{\exp(\epsilon),\delta} c : \Phi \Longrightarrow \equiv$$

Differential privacy as quantitative 2-safety

- K is (ε, δ)-diff. private w.r.t. Φ iff for all D_1 and D_2 and S

$$\Phi(D_1, D_2) \implies \Pr[K(D_1) \in S] \leq \exp(\epsilon) \times \Pr[K(D_2) \in S] + \delta$$

Relational pre-condition

(Quantitative) relational post-condition

- We propose a quantitative probabilistic relational Hoare Logic

$$c_1 \sim_{\alpha,\delta} c_2 : \Phi \implies \Psi$$

Needs to be lifted to distributions

such that c is (ε, δ)-diff. private w.r.t. Φ iff

$$c \sim_{\exp(\epsilon),\delta} c : \Phi \implies \equiv$$

Characterizing differential privacy

$c_1 \sim_{\alpha,\delta} c_2 : \Phi \Rightarrow \Psi$ is valid iff for all D_1 and D_2

$$\Phi(D_1, D_2) \Longrightarrow \mathrm{lift}_{\alpha,\delta}\ \Psi\ (\llbracket c_1 \rrbracket\ D_1)\ (\llbracket c_2 \rrbracket\ D_2)$$

We define α-distance such that:

Characterizing differential privacy

$c_1 \sim_{\alpha,\delta} c_2 : \Phi \Rightarrow \Psi$ is valid iff for all D_1 and D_2

$$\Phi(D_1, D_2) \Longrightarrow \mathrm{lift}_{\alpha,\delta}\ \Psi\ (\llbracket c_1 \rrbracket\ D_1)\ (\llbracket c_2 \rrbracket\ D_2)$$

We define α-distance such that:

- c is (ε, δ)-diff. private w.r.t. Φ iff for all D_1 and D_2

Characterizing differential privacy

$c_1 \sim_{\alpha,\delta} c_2 : \Phi \Rightarrow \Psi$ is valid iff for all D_1 and D_2

$$\Phi(D_1, D_2) \Longrightarrow \mathrm{lift}_{\alpha,\delta}\ \Psi\ (\llbracket c_1 \rrbracket\ D_1)\ (\llbracket c_2 \rrbracket\ D_2)$$

We define α-distance such that:

- c is (ε, δ)-diff. private w.r.t. Φ iff for all D_1 and D_2

$$\Phi(D_1, D_2) \Longrightarrow \Delta_\alpha(\llbracket c_1 \rrbracket\ D_1, \llbracket c_2 \rrbracket\ D_2) \leq \delta$$

Characterizing differential privacy

$c_1 \sim_{\alpha,\delta} c_2 : \Phi \Rightarrow \Psi$ is valid iff for all D_1 and D_2

$$\Phi(D_1, D_2) \Longrightarrow \mathrm{lift}_{\alpha,\delta}\ \Psi\ (\llbracket c_1 \rrbracket\ D_1)\ (\llbracket c_2 \rrbracket\ D_2)$$

We define α-distance such that:

- c is (ε, δ)-diff. private w.r.t. Φ iff for all D_1 and D_2

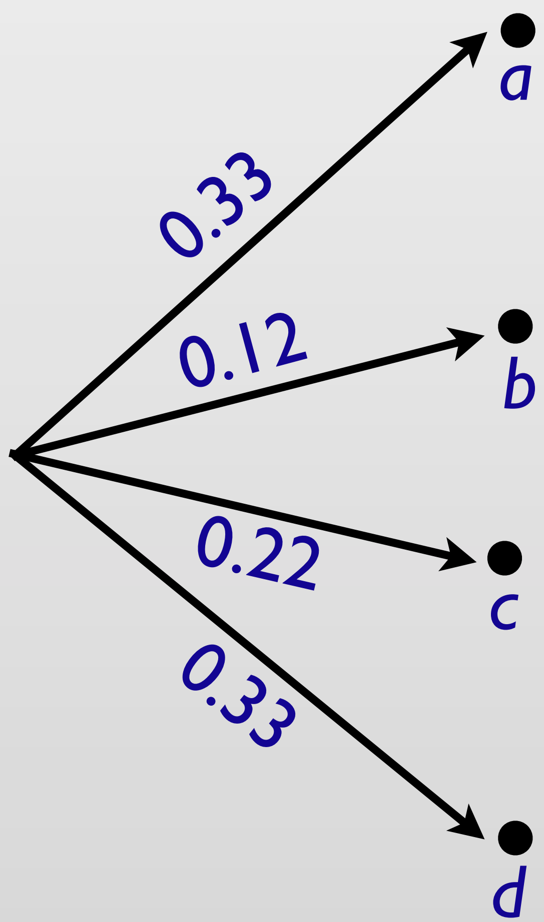
$$\Phi(D_1, D_2) \Longrightarrow \Delta_\alpha(\llbracket c_1 \rrbracket\ D_1, \llbracket c_2 \rrbracket\ D_2) \leq \delta$$

- Fundamental property of lifting

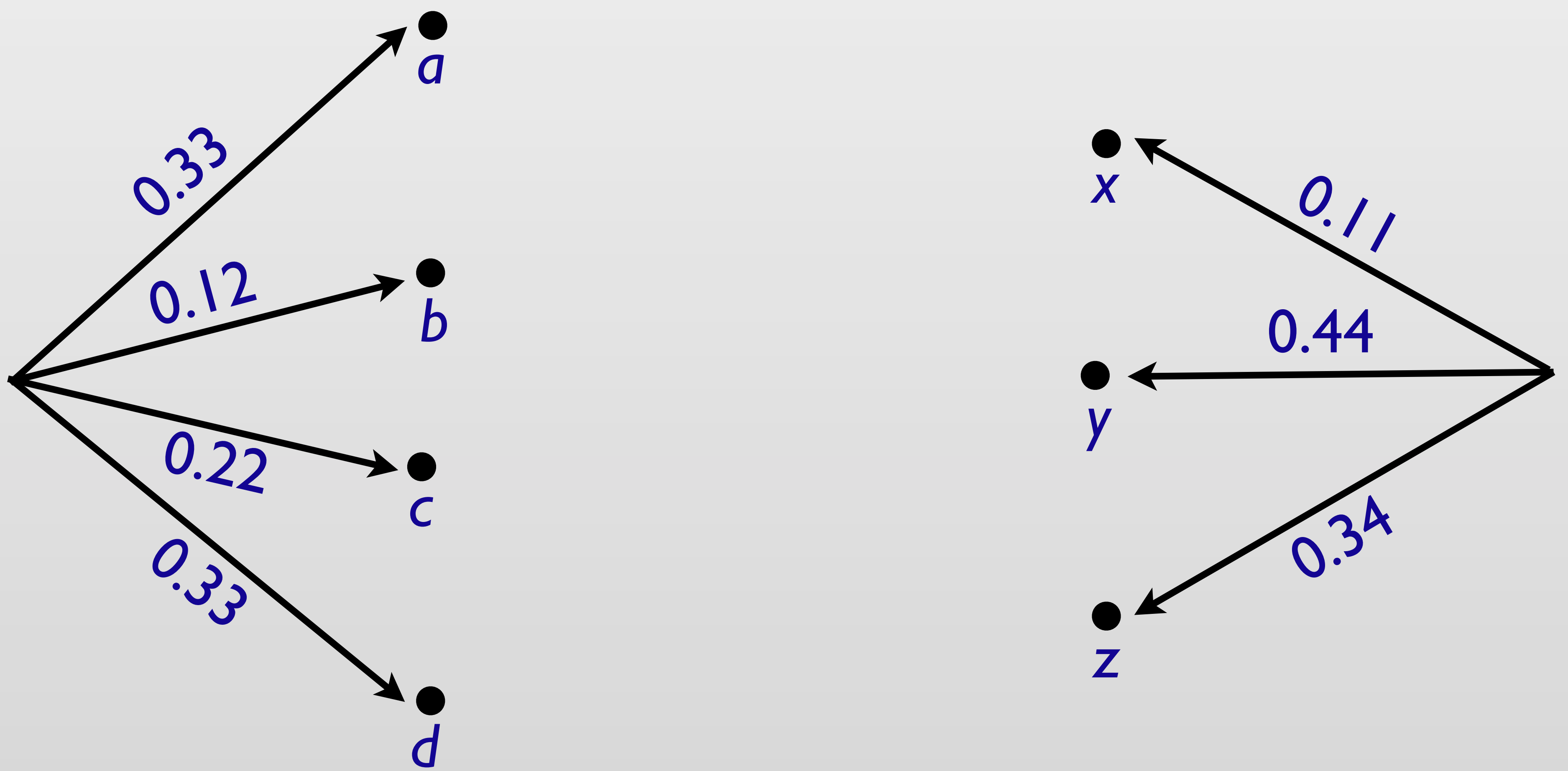
$$\Delta_\alpha(\mu_1, \mu_2) \leq \delta \iff \mathrm{lift}_{\alpha,\delta} \equiv \mu_1\ \mu_2$$

Lifting relations to distributions

Lifting relations to distributions

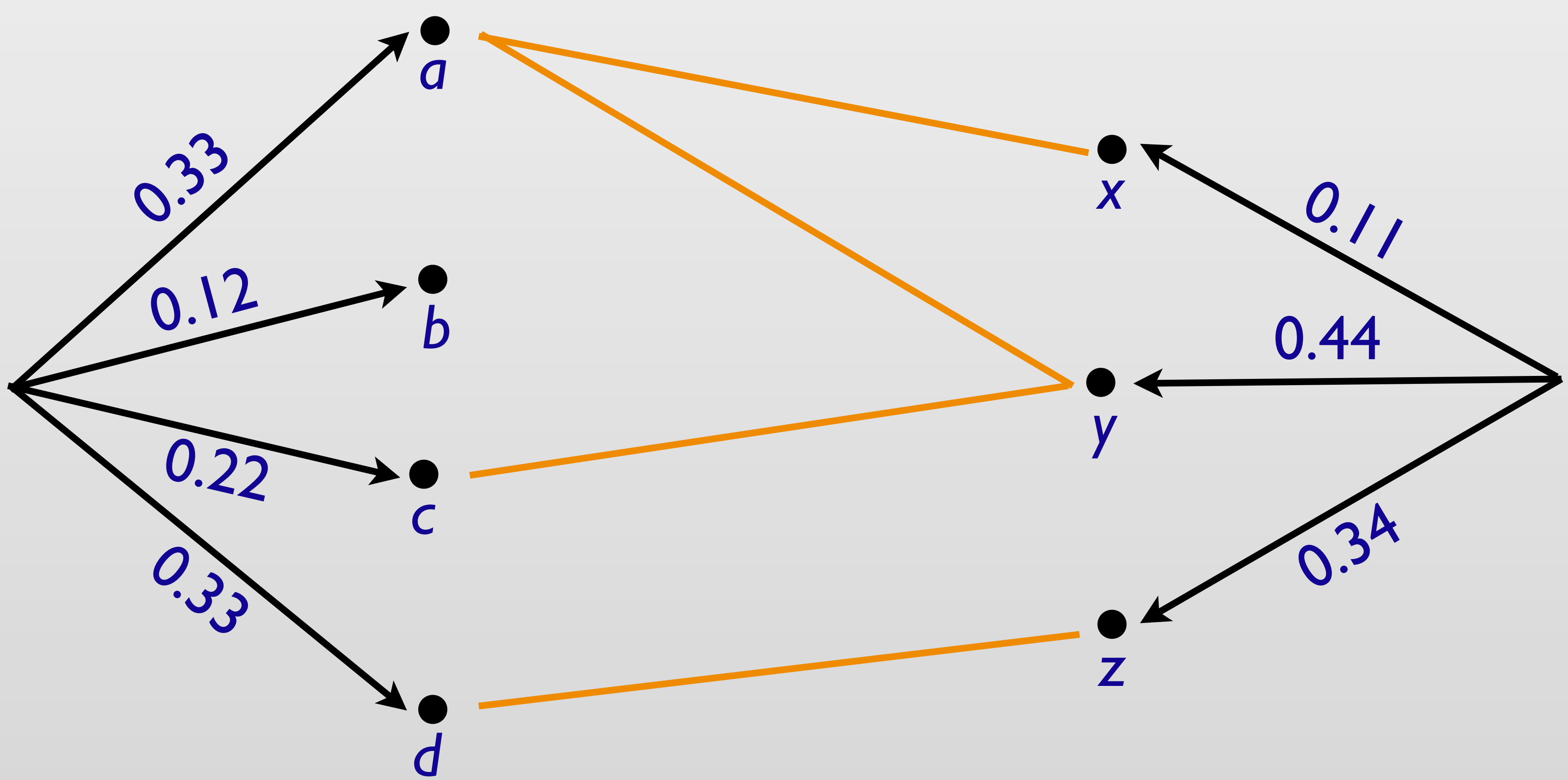


Lifting relations to distributions



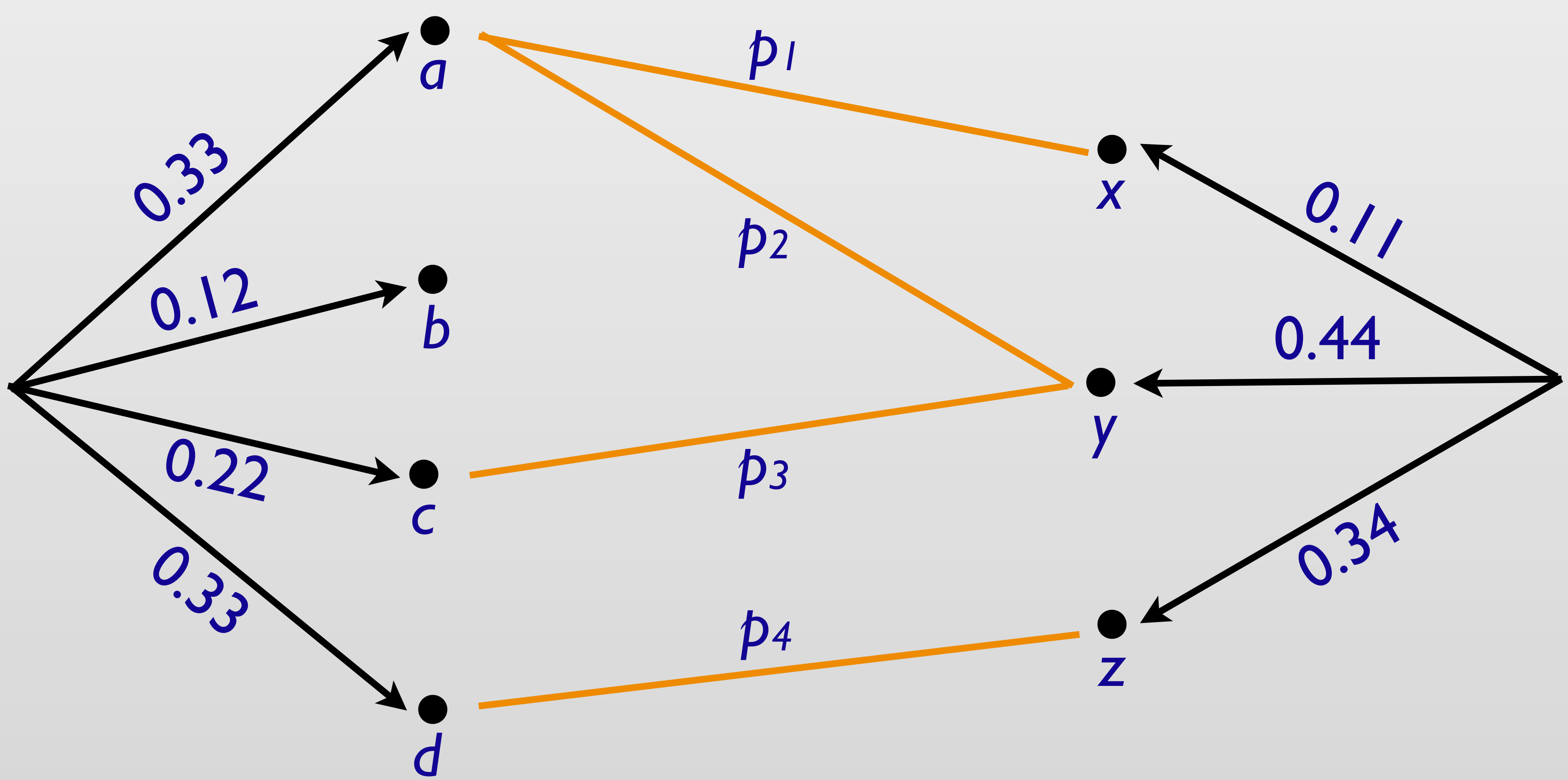
Lifting relations to distributions

Given R = {(a,x), (a,y), (c,y), (d,z)}



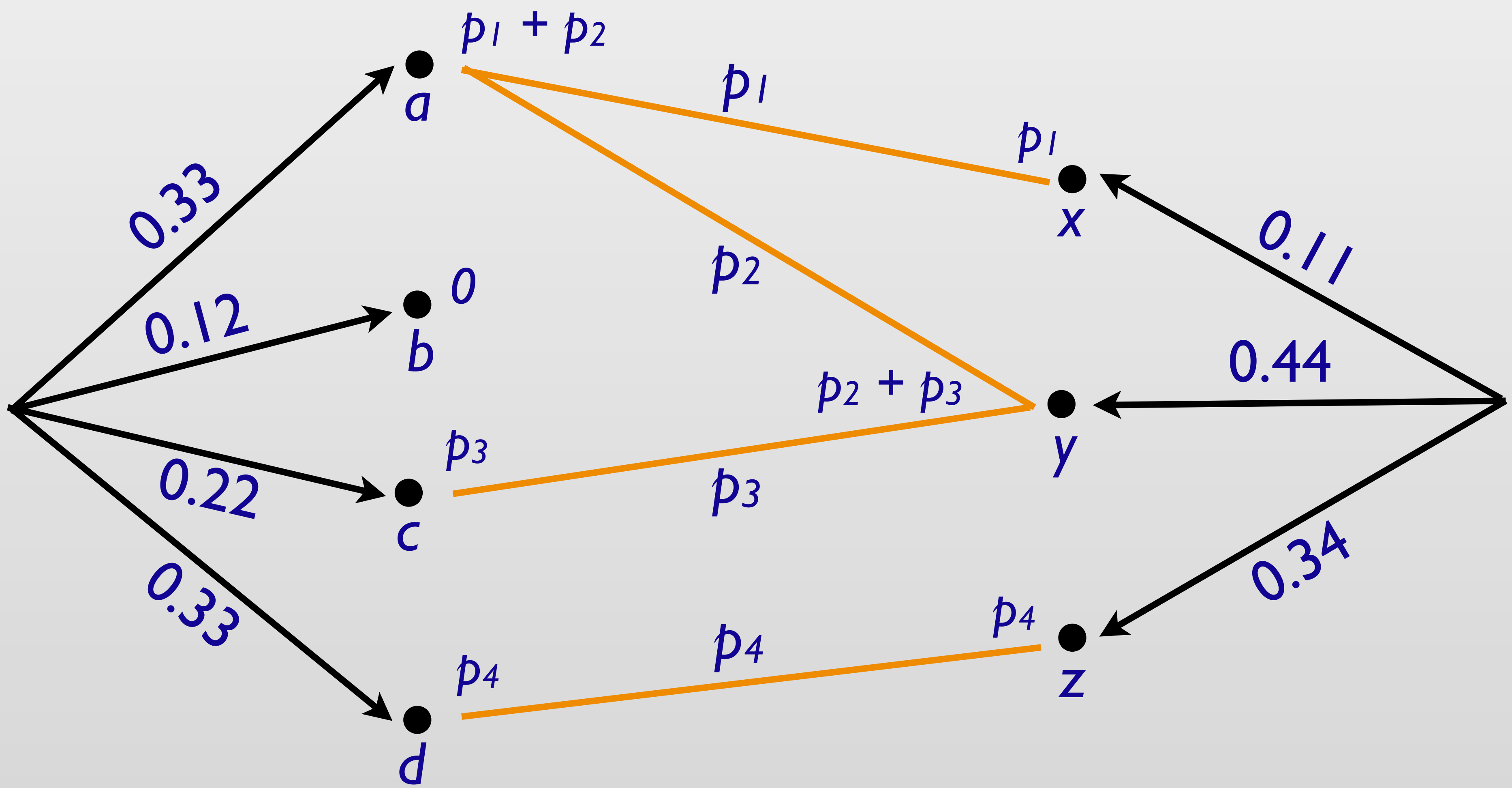
Lifting relations to distributions

Given R = {(a,x), (a,y), (c,y), (d,z)}, α=1.1 and δ=0.01



Lifting relations to distributions

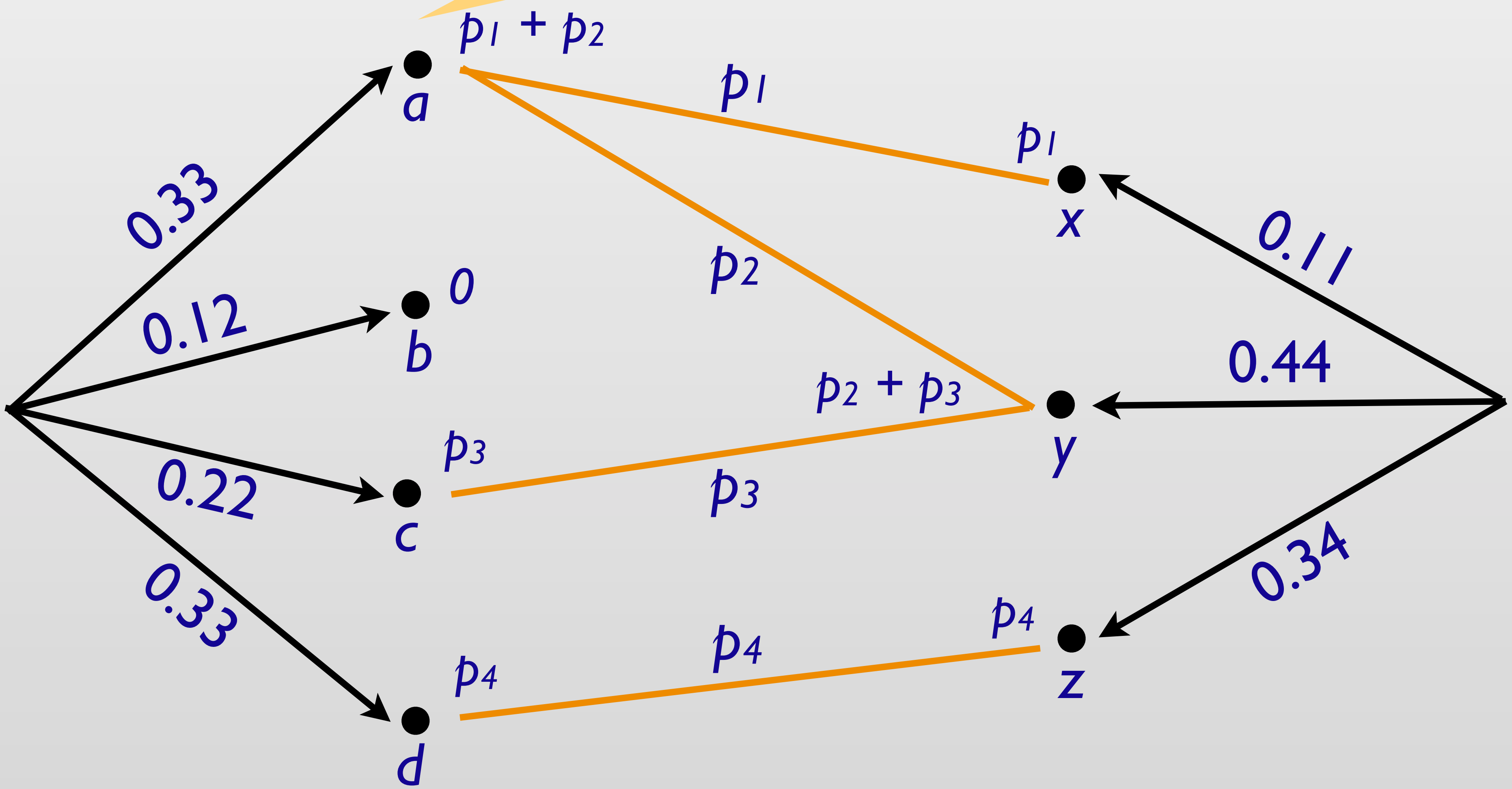
Given R = {(a,x), (a,y), (c,y), (d,z)}, α=1.1 and δ=0.01



Lifting relations to distributions

Given R = {(a,x), (a,y), (c,y), (d,z)}, α=1.1 and δ=0.01

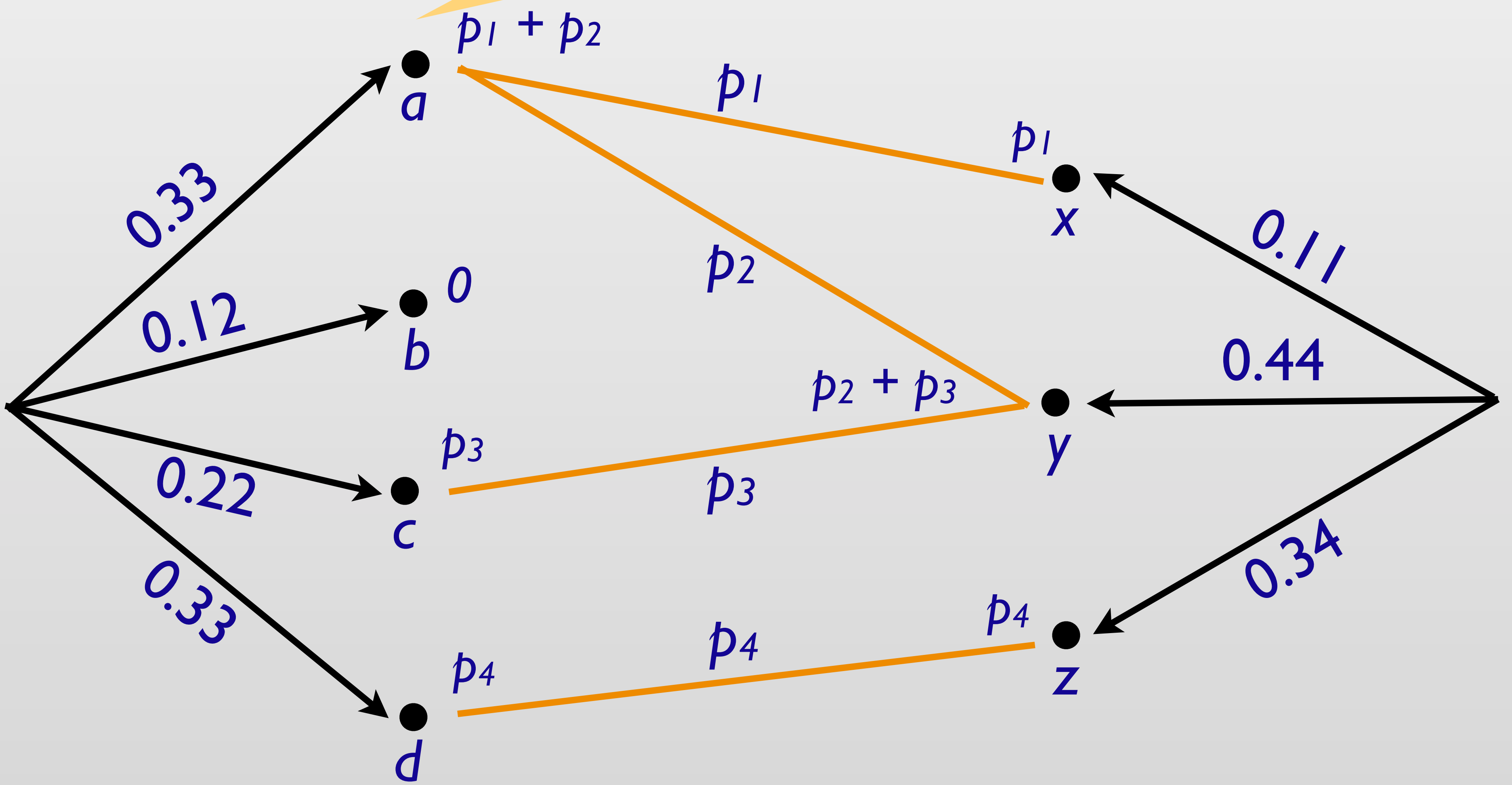
$$p_1 + p_2 \leq 0.33$$
$$\delta_a = \max\{0, 0.33 - \alpha\,(p_1 + p_2)\}$$
$$\delta_a + \delta_b + \delta_c + \delta_d \leq \delta$$



Lifting relations to distributions

Given R = {(a,x), (a,y), (c,y), (d,z)}, α=1.1 and δ=0.01

$$p_1 + p_2 \leq 0.33$$
$$\delta_a = \max\{0, 0.33 - \alpha\,(p_1 + p_2)\}$$
$$\delta_a + \delta_b + \delta_c + \delta_d \leq \delta$$



Witness distribution

$X \times Y$	$\mu(\cdot,\cdot)$
(a, x)	0.10
(a, y)	0.20
(c, y)	0.20
(d, z)	0.30
…	0

Selected rules

Sequential composition

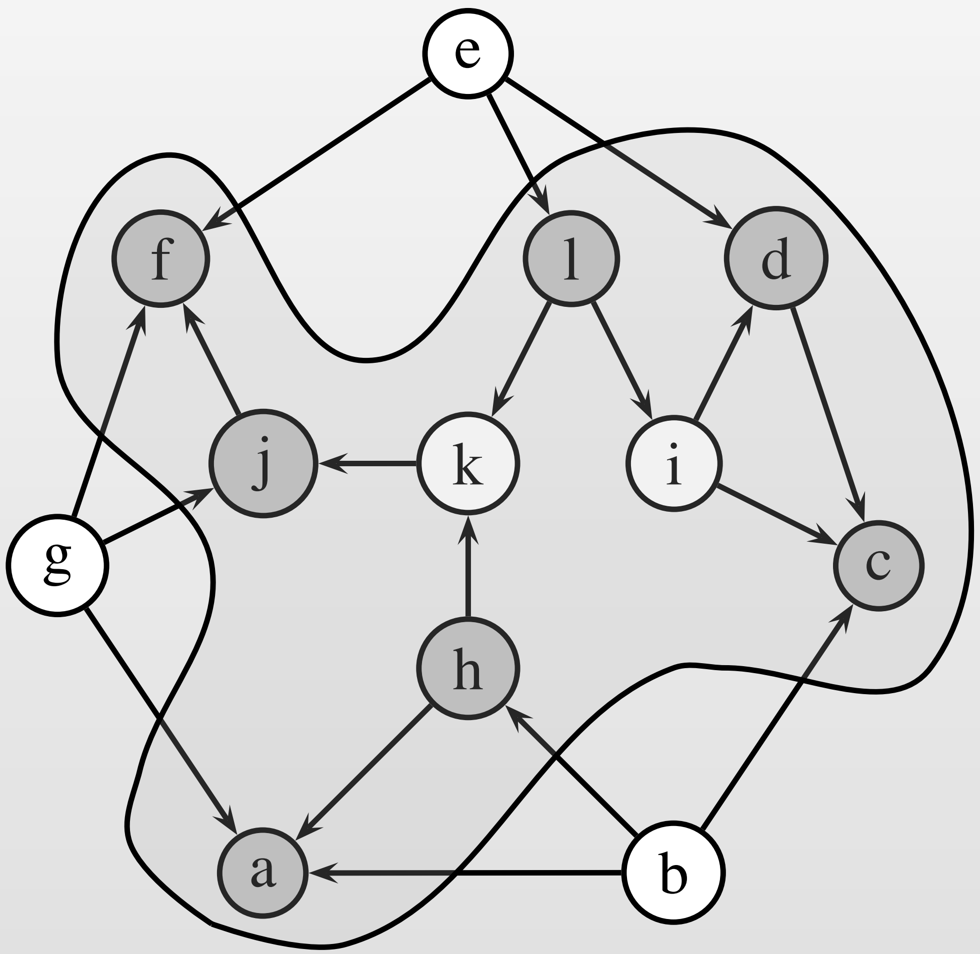
$$\frac{\models c_1 \sim_{\alpha,\delta} c_2 : \Psi \Rightarrow \Phi' \qquad \models c_1' \sim_{\alpha',\delta'} c_2' : \Phi' \Rightarrow \Phi}{\models c_1; c_1' \sim_{\alpha\alpha',\delta+\delta'} c_2; c_2' : \Psi \Rightarrow \Phi}$$

Laplacian Mechanism

$$\frac{}{\models x \xleftarrow{\$} \mathcal{L}_\lambda(r) \sim_{\exp(\epsilon),0} y \xleftarrow{\$} \mathcal{L}_\lambda(s) : |r\langle 1\rangle - s\langle 2\rangle| \leq \lambda\epsilon \Rightarrow x\langle 1\rangle = y\langle 2\rangle}$$

Application: Vertex Cover

Gupta et al. [SODA '10]



VertexCover(V, E)

$\pi \leftarrow \text{nil};$

while $E \neq \emptyset$ do

$\quad v \xleftarrow{\$} \text{pick}(V, E);$

$\quad \pi \leftarrow v :: \pi;$

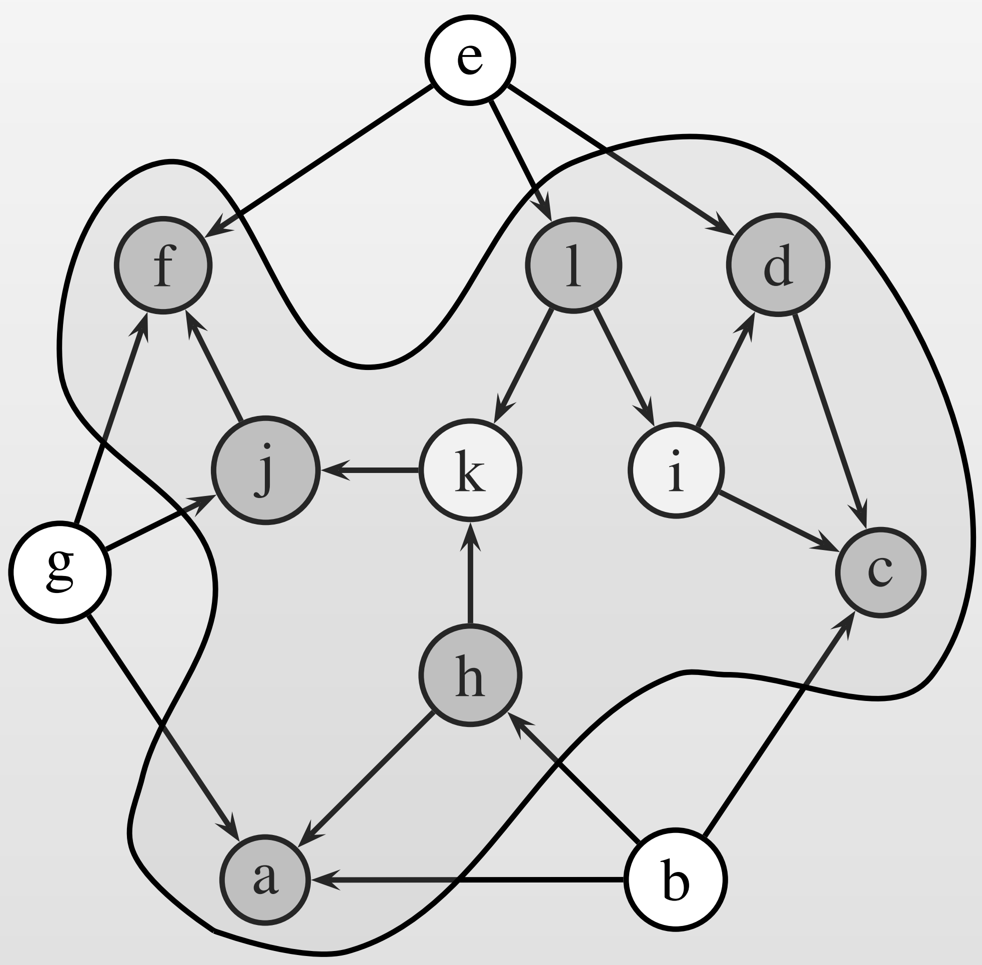
$\quad V \leftarrow V \setminus \{v\};\ \ E \leftarrow E \setminus (\{v\} \times V);$

end

$$\text{pick}(V, E) \;\propto\; deg_E(v)$$

Application: Vertex Cover

Gupta et al. [SODA '10]



VertexCover(V, E)

1 $\pi \leftarrow$ nil;

2 while $E \neq \emptyset$ do

3 $\quad v \xleftarrow{\$}$ pick(V, E);

4 $\quad \pi \leftarrow v :: \pi$;

5 $\quad V \leftarrow V \backslash \{v\};\ E \leftarrow E \setminus (\{v\} \times V)$;

6 end

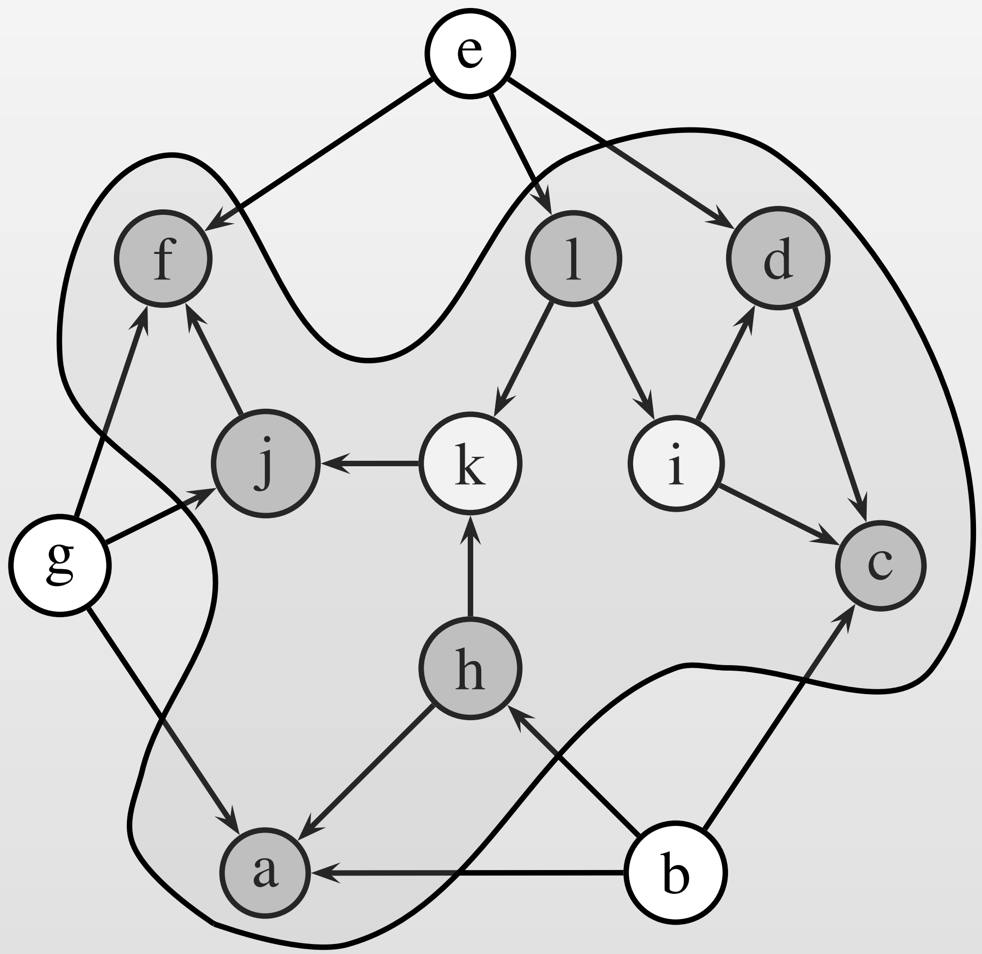
pick$(V, E) \;\propto\; deg_E(v)$

$$\text{VertexCover}(V, E) \sim_{\exp(\epsilon),0} \text{VertexCover}(V, E) :$$

$$V\langle 1\rangle = V\langle 2\rangle \wedge E\langle 1\rangle = E\langle 2\rangle \cup \{(t, u)\} \implies \pi\langle 1\rangle = \pi\langle 2\rangle$$

Application: Vertex Cover

Gupta et al. [SODA '10]



$\mathsf{VertexCover}(V, E)$

```
π ← nil;
while E ≠ ∅ do
    v ←$ pick(V, E);
    π ← v :: π;
    V ← V\{v};  E ← E \ ({v} × V);
end
```

$\mathsf{pick}(V, E) \;\propto\; deg_E(v)$

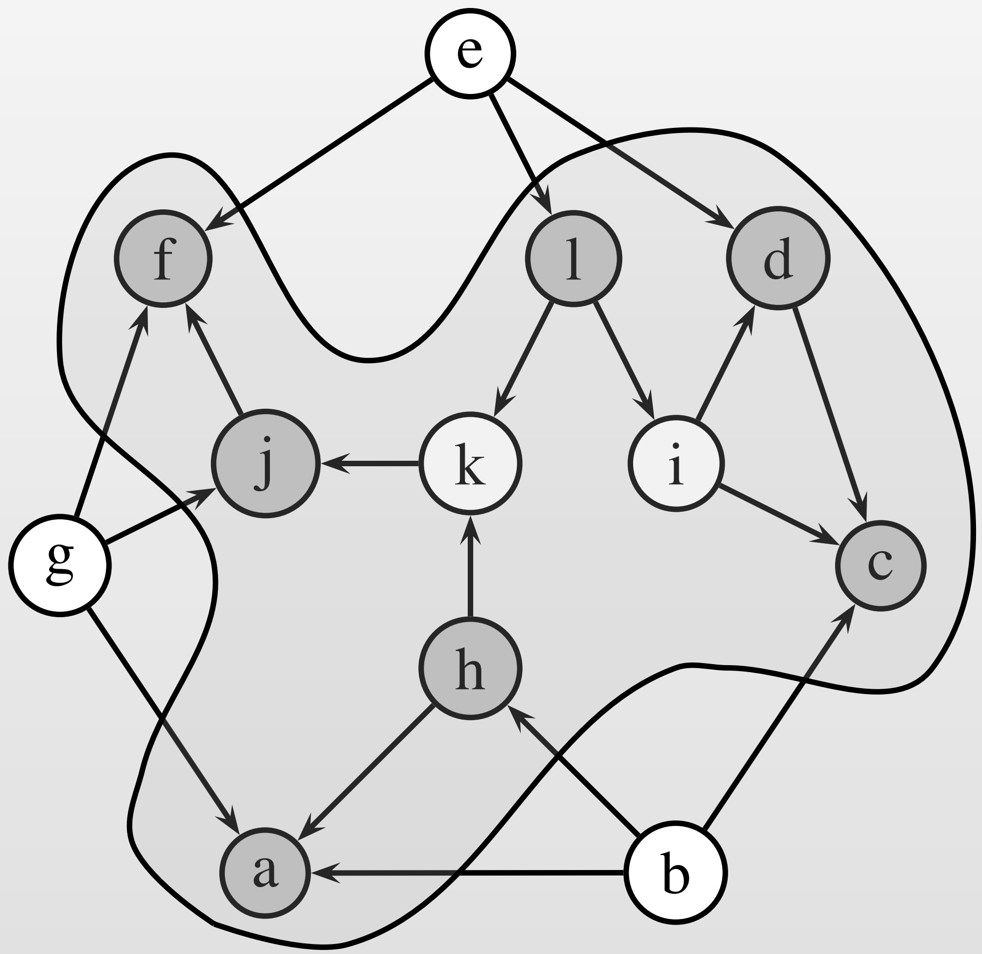
Not satisfied!

$$\mathsf{VertexCover}(V, E) \sim_{\exp(\epsilon),0} \mathsf{VertexCover}(V, E) :$$

$$V\langle 1\rangle = V\langle 2\rangle \wedge E\langle 1\rangle = E\langle 2\rangle \cup \{(t, u)\} \Longrightarrow \pi\langle 1\rangle = \pi\langle 2\rangle$$

Application: Vertex Cover

Gupta et al. [SODA '10]



VertexCover(V, E)

$\pi \leftarrow \text{nil};$

while $E \neq \emptyset$ do

  $v \xleftarrow{\$} \text{pick}(V, E);$

  $\pi \leftarrow v :: \pi;$

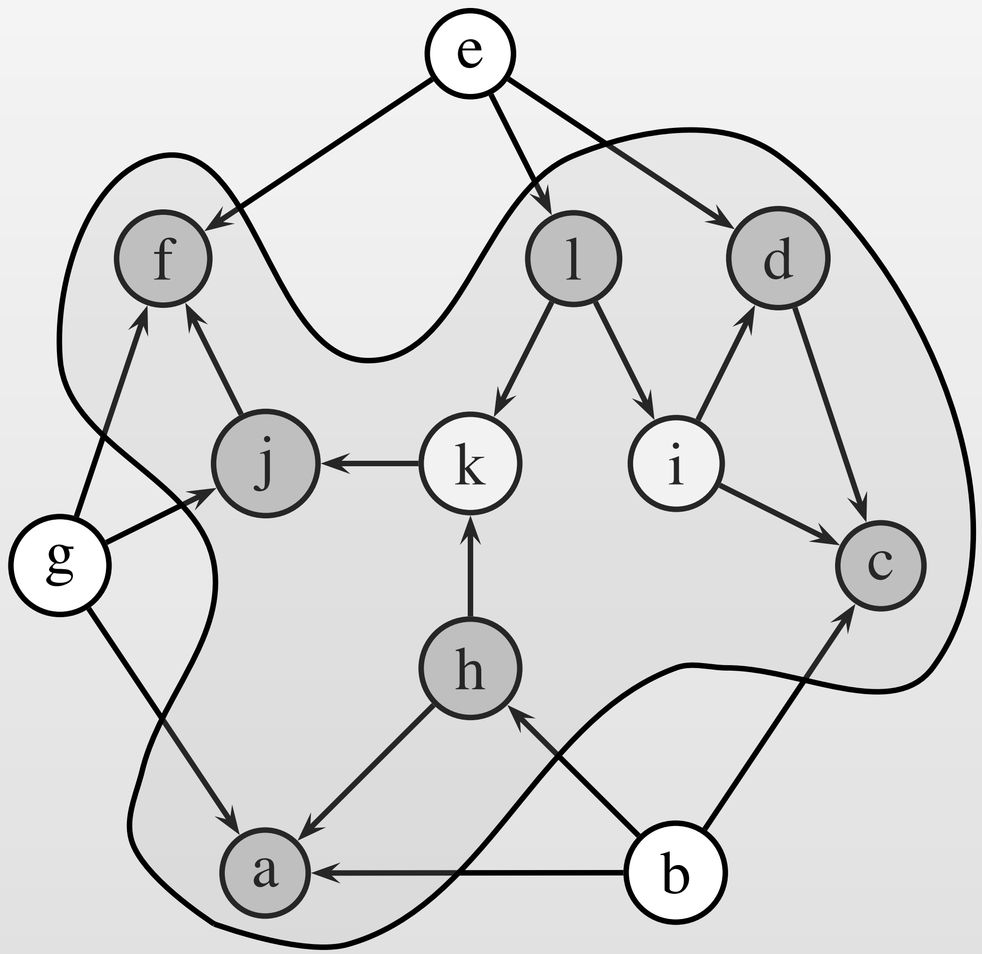
  $V \leftarrow V \setminus \{v\};\ E \leftarrow E \setminus (\{v\} \times V);$

end

$\text{pick}(V, E) \ \propto \ deg_E(v)$

Application: Vertex Cover

Gupta et al. [SODA '10]



$\pi = [b, g, e, h, l, k,$
$j, i, f, d, c, a]$

VertexCover(V, E)

1 $\pi \leftarrow$ nil;

2 while $E \neq \emptyset$ do

3 $\quad v \xleftarrow{\$}$ pick(V, E);

4 $\quad \pi \leftarrow v :: \pi$;

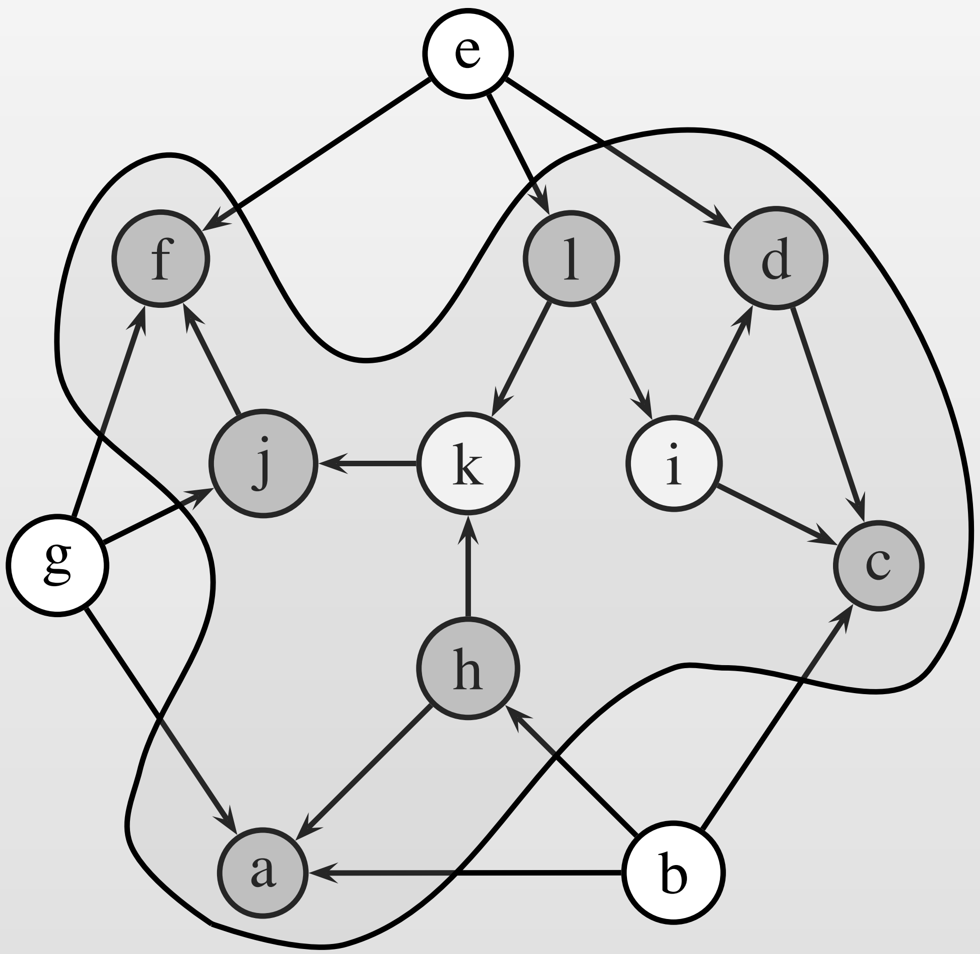
5 $\quad V \leftarrow V \setminus \{v\};\ E \leftarrow E \setminus (\{v\} \times V)$;

6 end

pick$(V, E) \;\propto\; deg_E(v)$

Application: Vertex Cover

Gupta et al. [SODA '10]



$\pi = [b, g, e, h, l, k, j, i, f, d, c, a]$

VertexCover(V, E, ϵ)

$\pi \leftarrow$ nil; $n \leftarrow |V|$; $i \leftarrow 0$;

while $i < n$ do

  $v \xleftarrow{\$}$ pick(V, E, ϵ, n, i);

  $\pi \leftarrow v :: \pi$;

  $V \leftarrow V \setminus \{v\}$; $E \leftarrow E \setminus (\{v\} \times V)$;

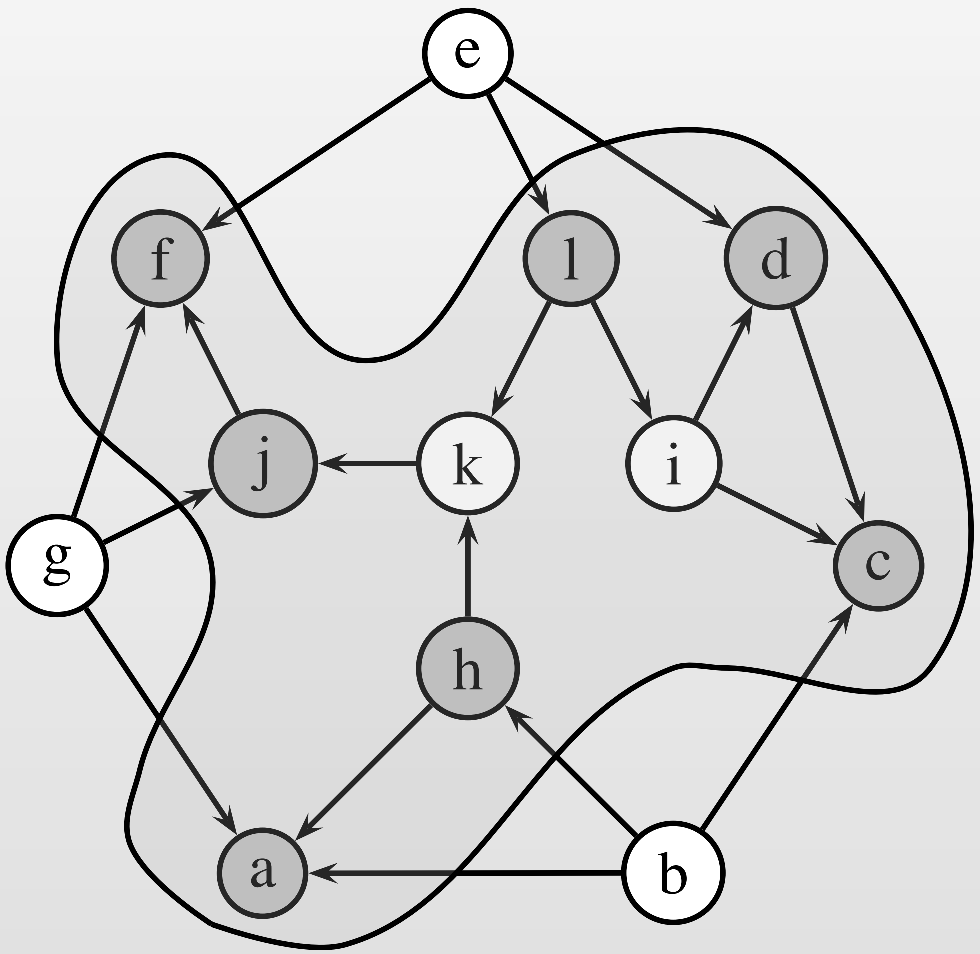
  $i \leftarrow i + 1$

end

$$\text{pick}(V, E, \epsilon, n, i) \;\propto\; deg_E(v) + \frac{4}{\epsilon}\sqrt{\frac{n}{n-i}}$$

Application: Vertex Cover

Gupta et al. [SODA '10]



$\mathrm{VertexCover}(V, E, \epsilon)$

$\pi \leftarrow \mathrm{nil};\ n \leftarrow |V|;\ i \leftarrow 0;$

$\quad$ while $i < n$ do

$\quad\quad v \xleftarrow{\$} \mathrm{pick}(V, E, \epsilon, n, i);$

$\quad\quad \pi \leftarrow v :: \pi;$

$\quad\quad V \leftarrow V \backslash \{v\};\ E \leftarrow E \setminus (\{v\} \times V);$

$\quad\quad i \leftarrow i + 1$

$\quad$ end

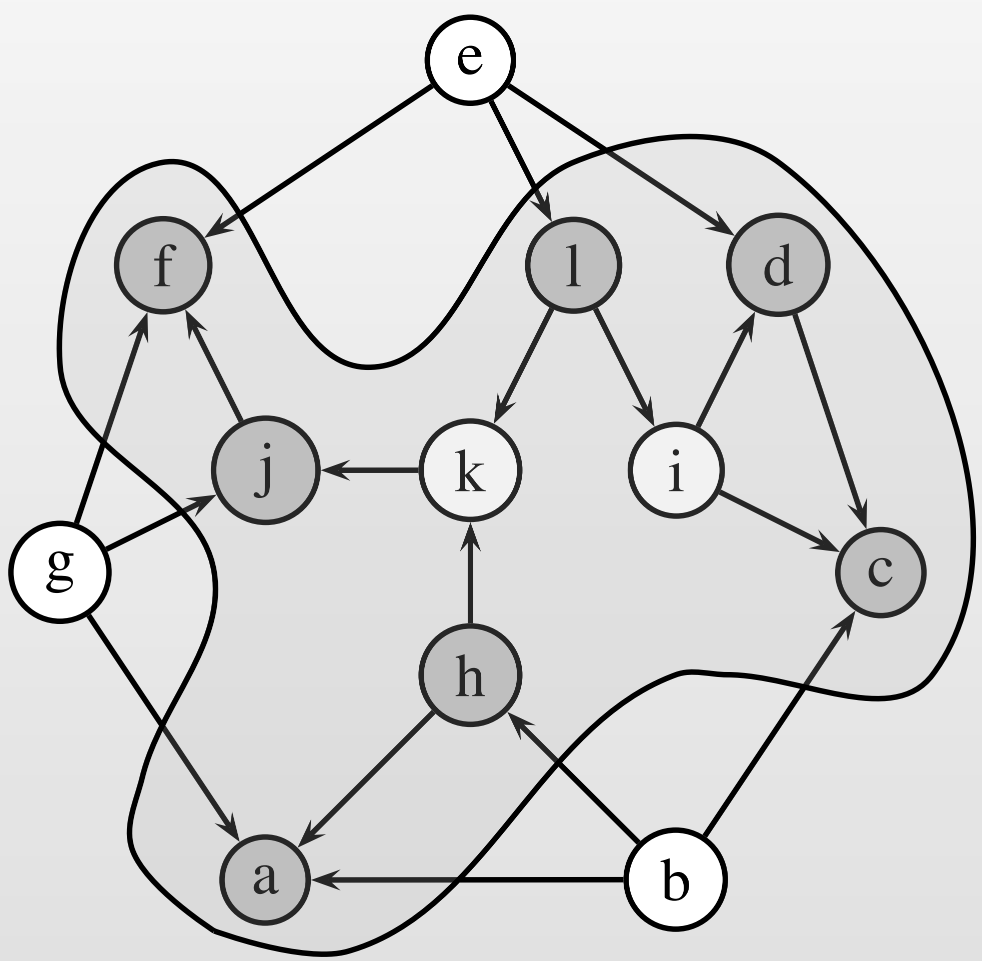
$$\mathrm{pick}(V, E, \epsilon, n, i) \ \propto \ deg_E(v) + \frac{4}{\epsilon}\sqrt{\frac{n}{n-i}}$$

$$\mathrm{VertexCover}(V, E, \epsilon) \sim_{\exp(\epsilon),0} \mathrm{VertexCover}(V, E, \epsilon) :$$

$$V\langle 1\rangle = V\langle 2\rangle \wedge E\langle 1\rangle = E\langle 2\rangle \cup \{(t, u)\} \implies \pi\langle 1\rangle = \pi\langle 2\rangle$$

Application: Vertex Cover

Gupta et al. [SODA '10]



$\mathsf{VertexCover}(V, E, \epsilon)$

$$
\begin{array}{l}
\pi \leftarrow \mathsf{nil};\ \ n \leftarrow |V|;\ \ i \leftarrow 0; \\
\mathsf{while}\ i < n\ \mathsf{do} \\
\quad v \xleftarrow{\$} \mathsf{pick}(V, E, \epsilon, n, i); \\
\quad \pi \leftarrow v :: \pi; \\
\quad V \leftarrow V \backslash \{v\};\ \ E \leftarrow E \setminus (\{v\} \times V); \\
\quad i \leftarrow i + 1 \\
\mathsf{end}
\end{array}
$$

$\mathsf{pick}(V, E, \epsilon, n, i) \ \propto \ a$

Proven correct using CertiPriv

$$\mathsf{VertexCover}(V, E, \epsilon) \sim_{\exp(\epsilon),0} \mathsf{VertexCover}(V, E, \epsilon) :$$

$$V\langle 1\rangle = V\langle 2\rangle \wedge E\langle 1\rangle = E\langle 2\rangle \cup \{(t, u)\} \implies \pi\langle 1\rangle = \pi\langle 2\rangle$$

Conclusions

- Framework for reasoning about quantitative relational properties of randomized computations
  - Laplacian and Exponential mechanisms
  - Differential privacy for streaming and graph algorithms
  - Asymmetric logic
- Further work:
  - Computational differential privacy
  - Hash functions unto elliptic curves and statistical zero-knowledge
- Challenge: logic for arbitrary quantitative relational properties

Thanks for your attention!

Define α-distance as:

$$\Delta_\alpha(d_1, d_2) = \max_A(\max(d_1\ 1_A - \alpha\ (d_2\ 1_A), d_2\ 1_A - \alpha\ (d_1\ 1_A)))$$

(α,δ)-lifting of relations to distributions:

$$\begin{aligned}\text{lift}_{\alpha,\delta}\ R\ (d_1 : \mathcal{D}_A)\ (d_2 : \mathcal{D}_B) = \exists (d : \mathcal{D}_{A*B}),\\ \pi_1(d) \leq d_1 \wedge \Delta_\alpha(\pi_1(d), d_1) \leq \delta \wedge\\ \pi_2(d) \leq d_2 \wedge \Delta_\alpha(\pi_2(d), d_2) \leq \delta \wedge\ \text{range}\ R\ d\end{aligned}$$

Output perturbation makes numerical queries ε-diff. private

- The Φ-sensitivity of a query $f : \mathcal{D} \to \mathbb{R}$ is defined as:

$$\Delta(f) = \max\{f(D_1) - f(D_2) \mid \Phi(D_1, D_2)\}$$

- The randomized computation

$$K(D) = f(D) + \mathsf{Lap}(\Delta(f)/\epsilon)$$

is ε-differentially private

Density proportional to $\exp(-\epsilon/\Delta(f))$

Composition theorems

If K_1 is $(\varepsilon_1, \delta_1)$-diff. private and K_2 is $(\varepsilon_2, \delta_2)$-diff. private

- Sequential composition

$(\varepsilon_1+\varepsilon_2, \delta_1+\delta_2)$-diff. private

- Parallel composition

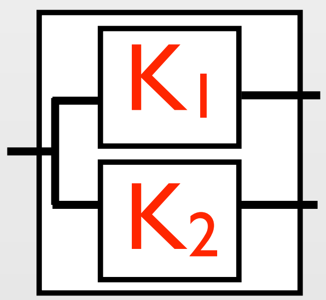
K_1 and K_2 depend on disjoint parts of the database

$(\max\{\varepsilon_1, \varepsilon_2\}, \max\{\delta_1, \delta_2\})$-diff. private

Composition theorems

If K_1 is $(\varepsilon_1, \delta_1)$-diff. private and K_2 is $(\varepsilon_2, \delta_2)$-diff. private

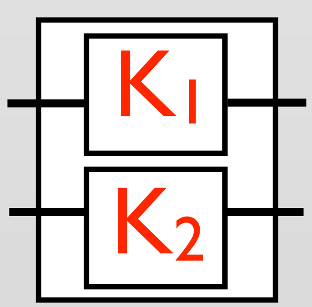
- Sequential composition



$(\varepsilon_1+\varepsilon_2, \delta_1+\delta_2)$-diff. private

- Parallel composition

K_1 and K_2 depend on disjoint parts of the database



$(\max\{\varepsilon_1, \varepsilon_2\}, \max\{\delta_1,\delta_2\})$-diff. private