ビッグデータのプライバシ保護技術

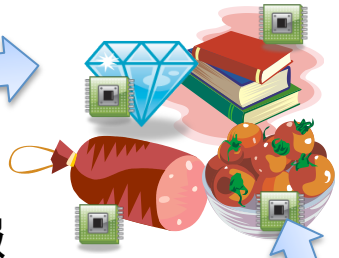
筑波大学 / JST CREST
佐久間 淳

個人情報を使った知識発見は(きっと)おもしろい

これを持っている人の平均年収はx万円です

銀行:口座情報

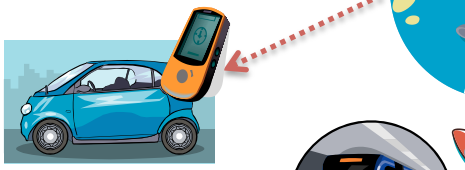
カード会社:決済情報



RFID:品物の種[illegible]

この場所にいる人はこんなものを買っています

伝染病発生源推定:この病気にかかった人はこんな路線を使っています



GPS: 移動履歴

交通機関: 移動履歴

証券会社:資産情報

携帯電話:発着信履歴, メール送受信履歴, 電話帳,

これを食べている人の平均体重はXkgです

こんな健康状態の人はこんな保険を使っています

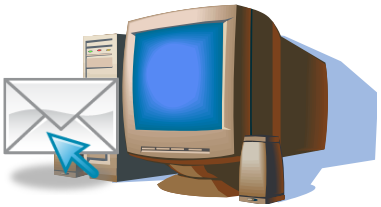
保険会社:保険情報

こんな[illegible]の人は[illegible]な薬を使っています

病院:カルテ
(疾病暦, 治療暦)

薬局:投薬暦

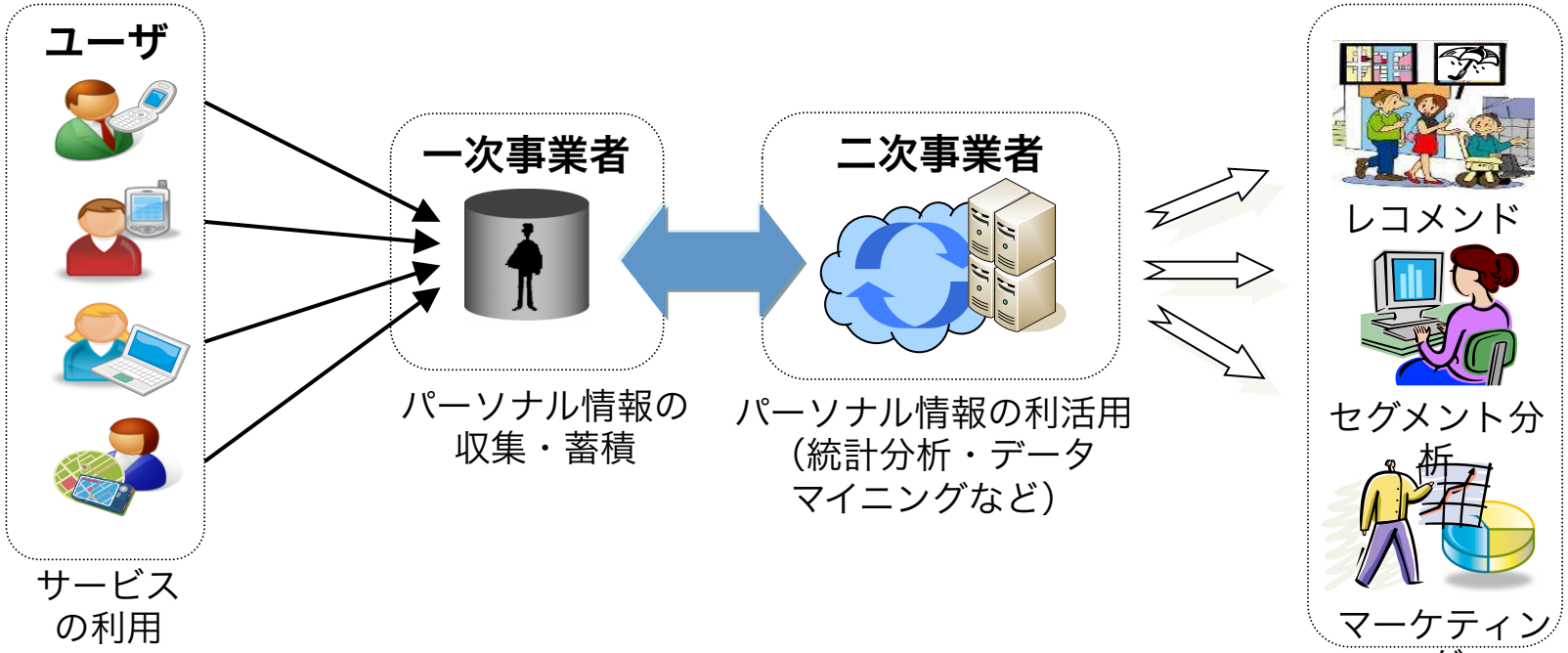
健康管理:バイタル情報
(体重, 血圧, 体脂肪率)



電子メール: 送受信履歴, アドレス帳

ビッグデータとプライバシ

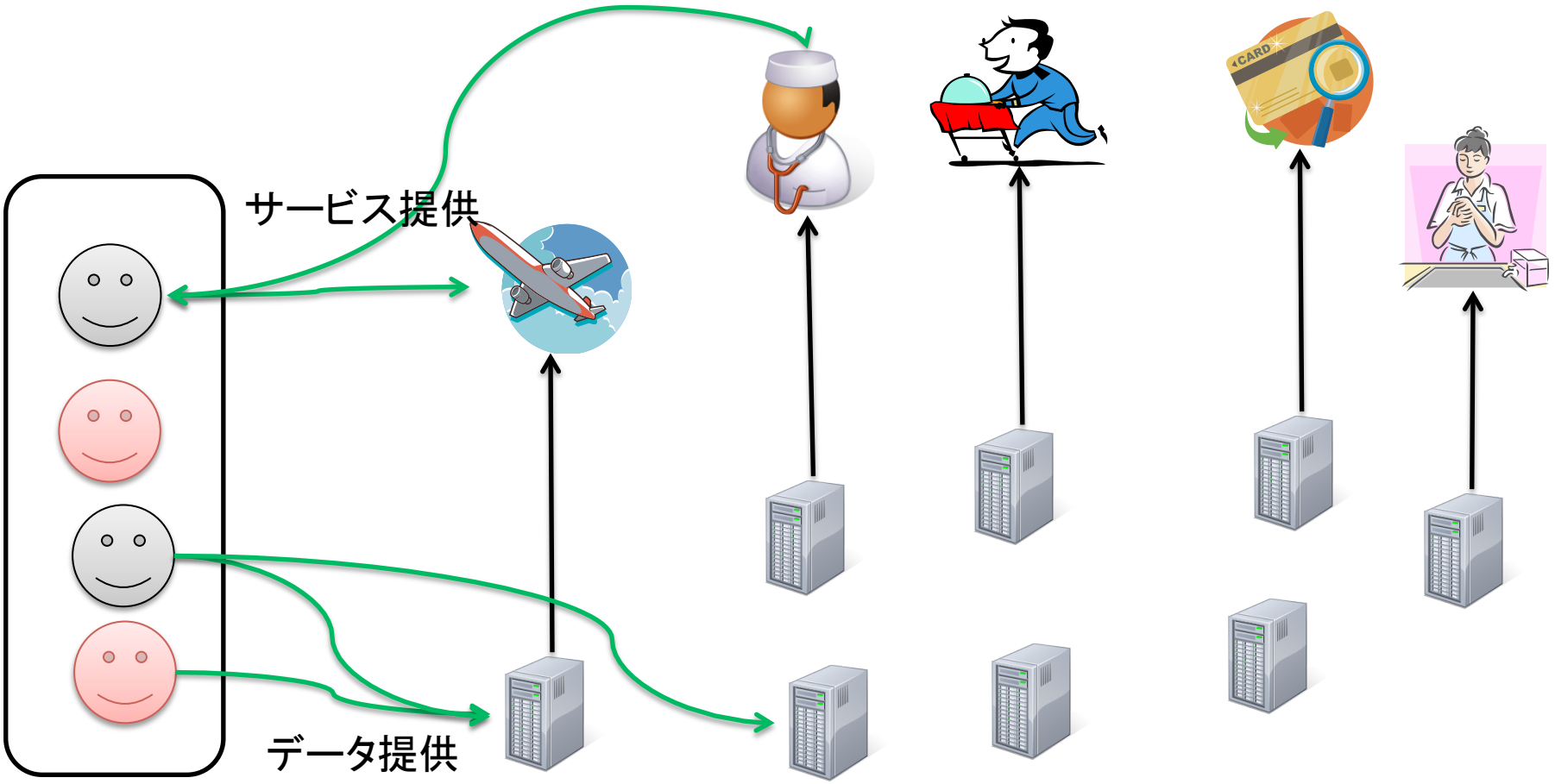
- ビッグデータに含まれる個人情報、機密情報
  - 取り扱いには注意が必要、しかし…
  - 個人情報を安全に取り扱うことができれば有用なサービスが生まれるはず



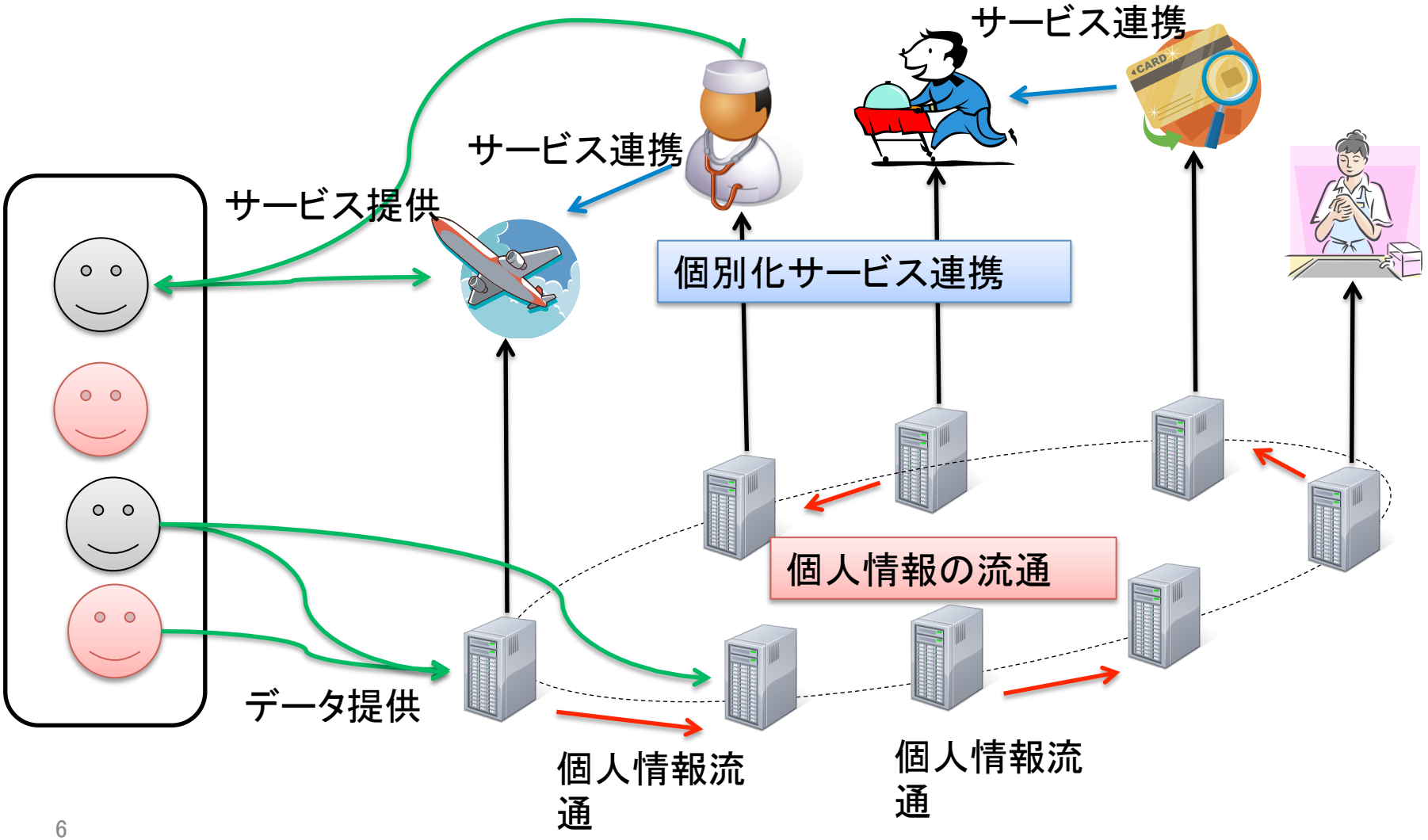
個別化サービスとプライバシ

- 位置ベースサービス
  - 個人の居場所をサーバに送信
  - サーバは場所に応じた情報提供
  - 逐一サーバに居場所を知られるのはイヤ
- ターゲティング広告
  - 自分が何を検索・購入したかをサーバに蓄積
  - サーバは個別化された広告を配信
  - 逐一サーバに行動を読まれるのはイヤ
- 通信の**安全性**: security issue
- データ利用に対する**安心感**: privacy issue
- 問題は安全性だけじゃなくて安心感

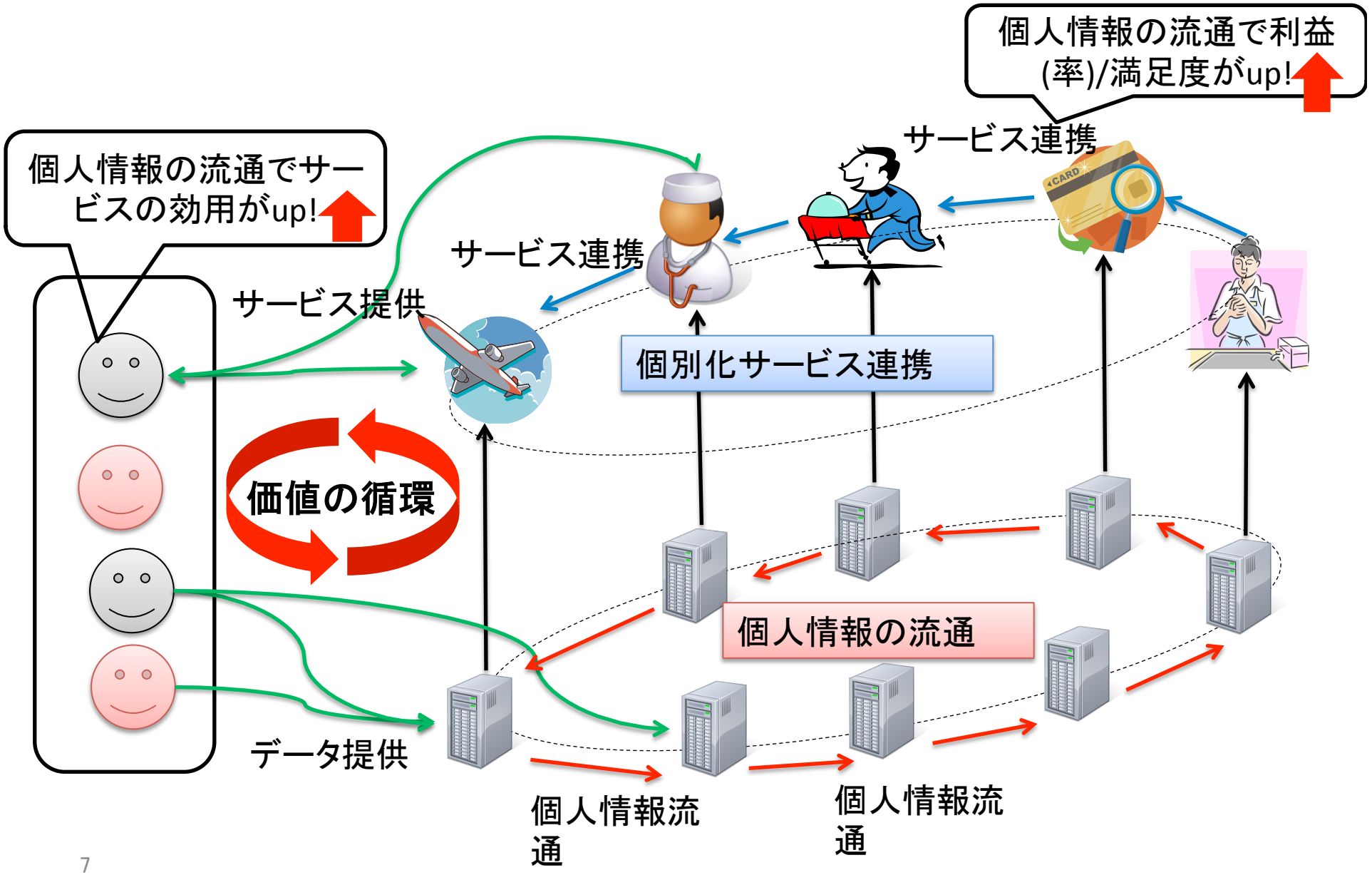
パーソナライズドサービスと個人情報: 一次利用



パーソナライズドサービスと個人情報: 二次利用



個別化サービスと個人情報:個人情報エコシステム



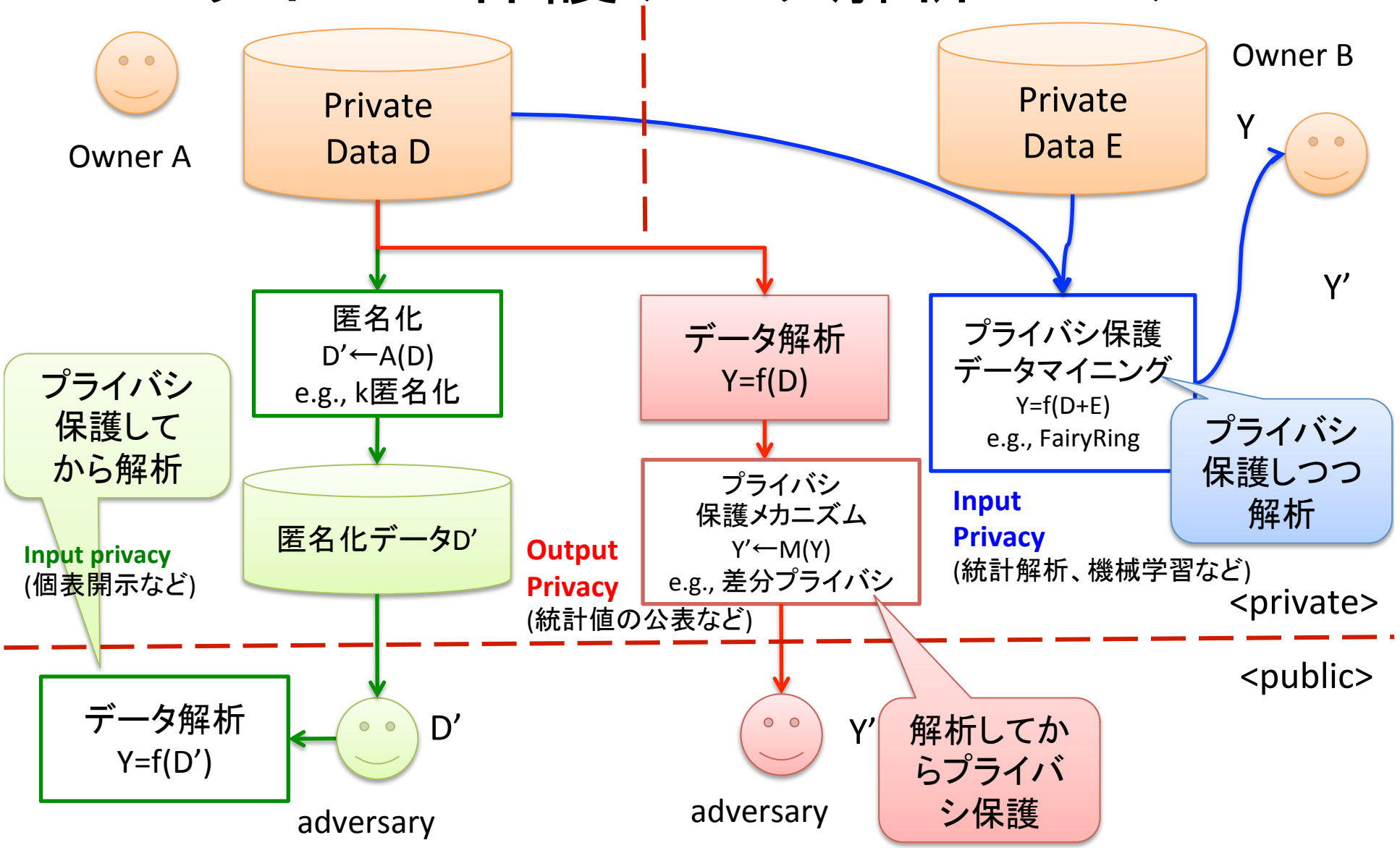
個人情報利用サービスの方向性

- 個人情報の二次利用
  - T-point
    - 各種消費行動・利用者情報の統合と分析
    - クーポンによる価値の還元
- 利用者同意の問題
  - データのオーナシップの尊重
  - Don’t be evil原則 by google
- Privacy by design
  - ビッグデータ時代のプライバシ概念
  - はじめからプライバシーのことを考えてサービス設計
- サービスにおけるプライバシとはなにか，技術的に健全にモデル化することが重要

プライバシ保護データ解析のモデル

- 秘密の情報Xが存在する
  - 秘密Xを持っている人: データ保有者
  - 秘密を暴きたい人: 攻撃者 (=データ解析者であることも、そうでないことも)
- やりたいこと
  - 秘密情報Xをつかってサービスしたい
    - y=f(x)
    - E.g., x=位置情報, f: 位置→広告への変換, y:打つべき広告
  - しかしadversaryにxを知られたくない
- モデル
  - 入力プライバシ (e.g., 匿名化)
  - 秘密計算 (e.g., プライバシ保護データマイニング)
  - 出力プライバシ (e.g., 差分プライバシ)

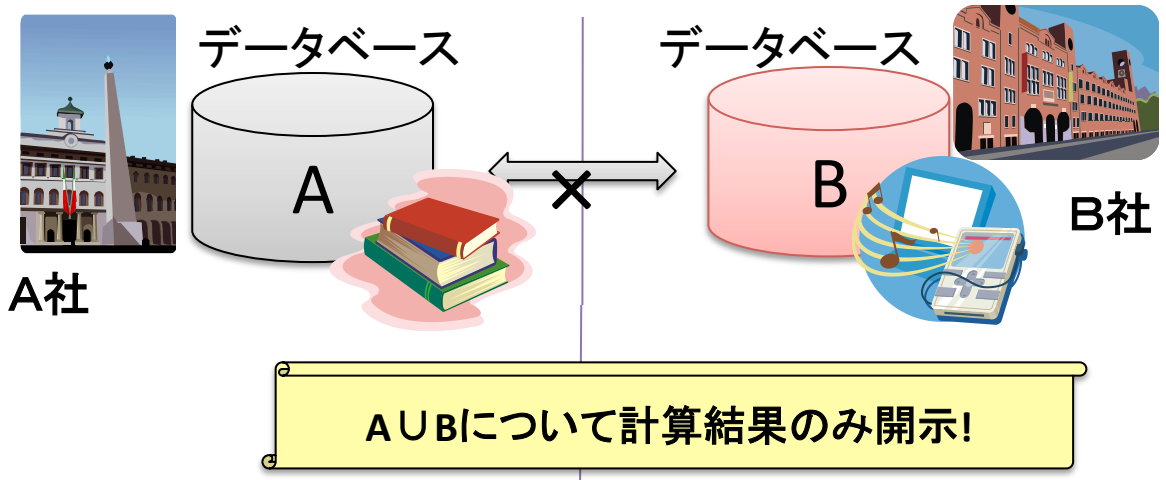
プライバシ保護データ解析のモデル



Beyond anonymization

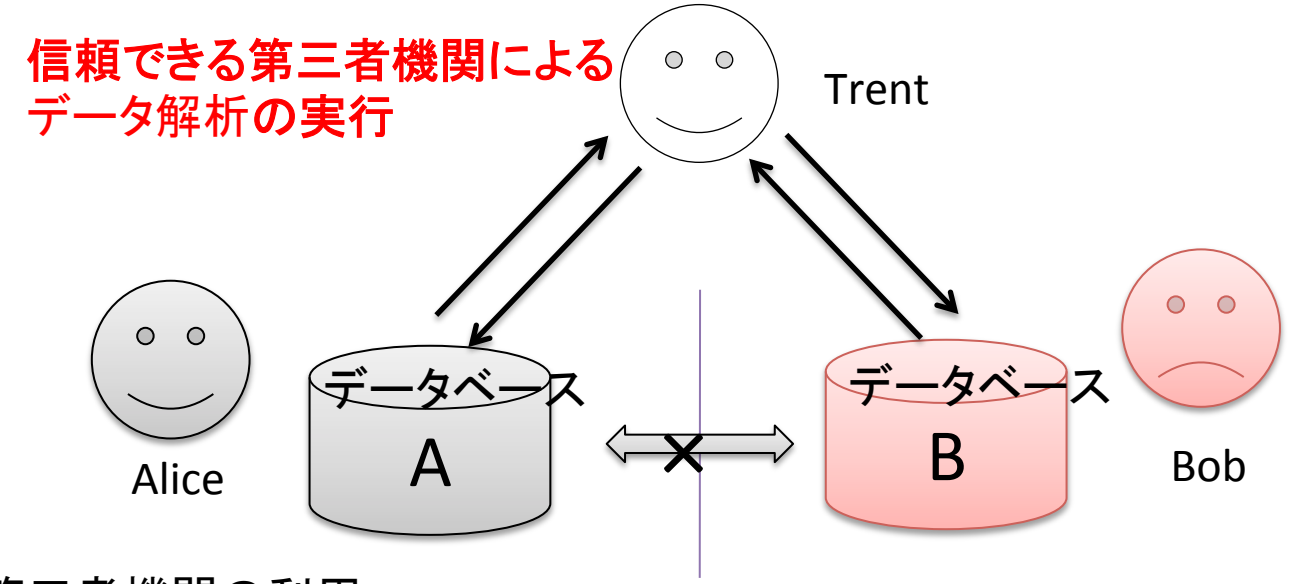
- 現段階で考慮されているプライバシ保護技術はほとんどが匿名化
  - プライバシは守られるが、データの質は落ちる
  - できることは限られてくる
- 秘密計算
  - プライバシ保護とデータ利用を両立する技術
  - 実証実験レベルでは注目されている
- 差分プライバシ
  - データベース問い合わせにおけるプライバシ保護
  - 強い攻撃者に対して頑健なプライバシ定義としてアカデミアでは広く注目を集めている
- この講演では「匿名化」の後に続く、次世代のプライバシ保護技術を紹介します

秘密計算



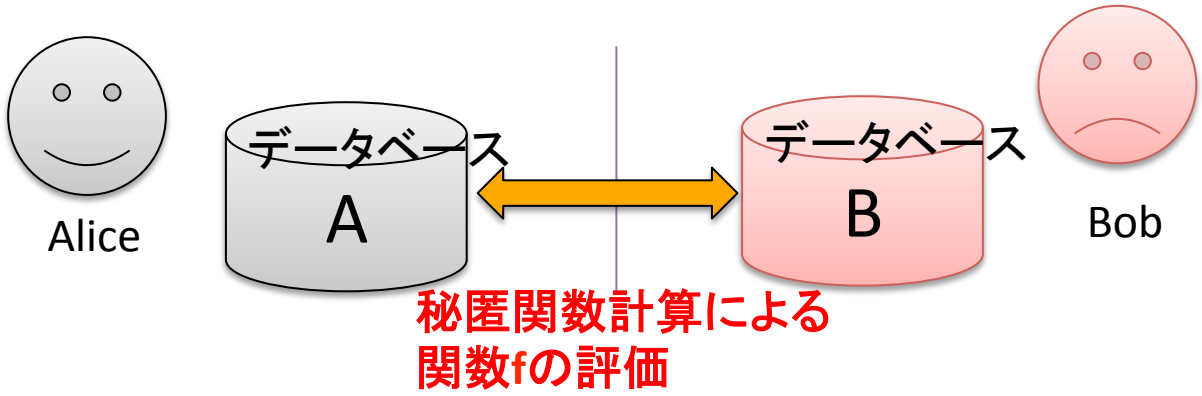
- 秘密計算
  - A社はデータベースAをB社に開示したくない
  - B社もデータベースBをA社に開示したくない
  - ただし結合されたデータベースAUBについてデータマイニングや統計処理を実行し、その結果のみを知りたい

秘密計算：第三者機関の利用



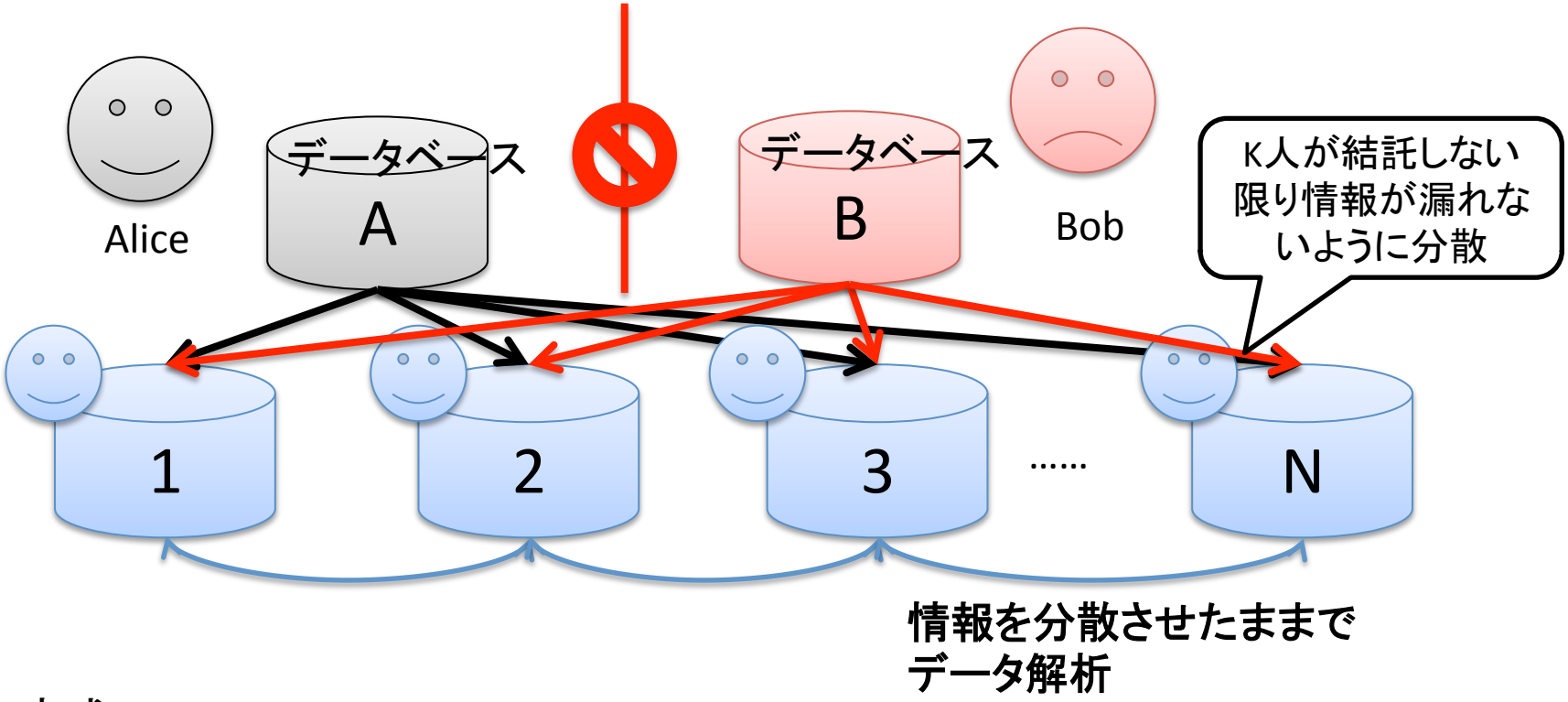
- 第三者機関の利用
  - 信頼できる第三者にデータを委託
  - 第三者がデータ解析を実行し、計算結果を返す
- これがいつでも実行可能ならば苦労はない
- もうすこし緩やかな条件で秘密計算を実行できないか?
  - 第三者機関を前提としない
  - 通常のネットワーク環境 (e.g., TCP/IP)

秘密計算：秘匿関数評価の利用



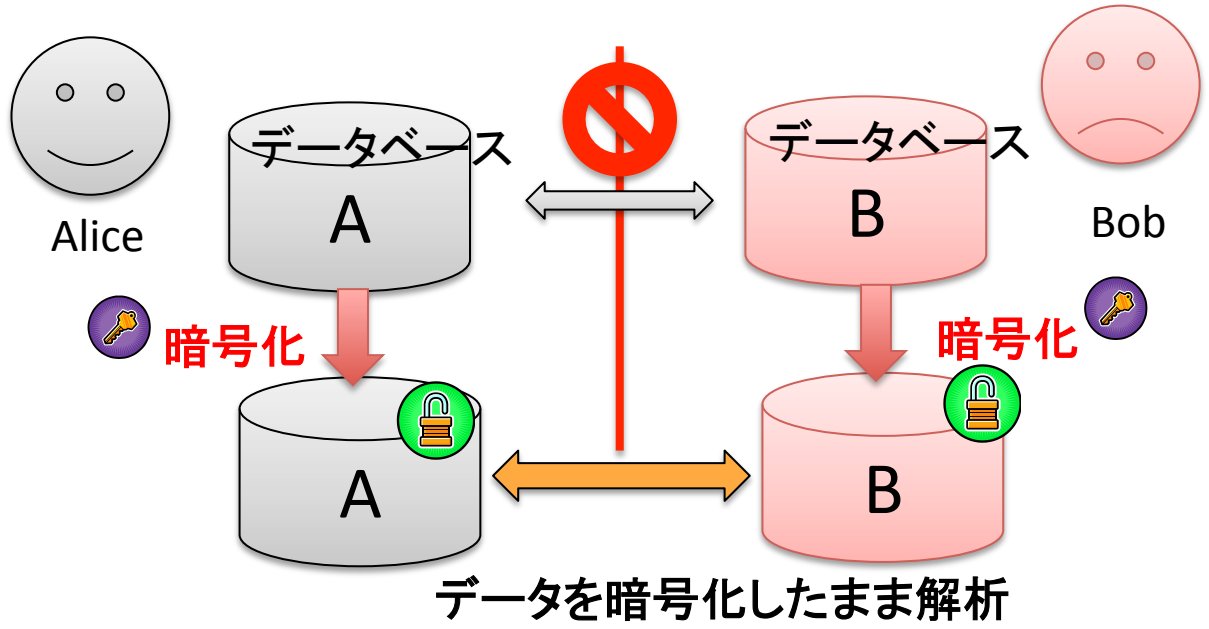
- 秘匿関数評価 [Yao86]他
  - Alice, Bobが持つ秘密入力に対し、
    - その入力を明かさずに、特定の関数fを多項式時間で実行
    - Alice, Bobはfの評価結果を取得
  - 関数fをデータ解析と捉えれば秘密計算が実行できる
- 残念ながら, 入力サイズが大きく, 評価対象の関数が複雑な場合、Yaoは遅い
  - ビッグデータ時代には不向き
  - 近年の技術発展で高速化も

秘密計算：秘密分散による方法



- NTT方式、Sharemind
  - K人以上が結託しない限り情報が漏れないように分散
  - **情報を分散させたままでデータ解析**
- 計算速度は速いが、複数の信頼出来る機関が必要

秘密計算：暗号プロトコルによる方法



- 暗号プロトコル [Lindel et al. *CRYPTO2000*]他
  - オリジナルデータを準同系性公開鍵により暗号化
    - 暗号化されたデータ同士の加算や乗算が可能
  - 暗号化されたデータ上でのデータ解析を実行
  - 解析結果を復号

準同型性公開鍵暗号

- m ∈ Z_N をメッセージ, r ∈ Z_N を乱数とする
- (pk, sk): 公開鍵と秘密鍵のペア
  - 暗号化: $c \leftarrow \mathrm{Enc}_{p_k}(m_0; r_0)$
  - 複号化: $m_0 \leftarrow \mathrm{Dec}_{s_k}(c)$
- m0,m1, r1,r2 ∈ Z_N
- 暗号系が(加法的)準同系性を持つとき:
  - 暗号文の和

$$\mathrm{Enc}_{p_k}(m_0; r_0) \cdot \mathrm{Enc}_{p_k}(m_1; r_1) = \mathrm{Enc}_{p_k}(m_0 + m_1; r_1 \cdot r_2)$$

  - 暗号文と平文の積

$$\mathrm{Enc}_{p_k}(m_0; r_0)^{m_1} = \mathrm{Enc}_{p_k}(m_0 m_1; r')$$

-
- e.g. Paillier暗号

値を他人に見せないまま、データ解析が可能

どこでも秘密計算: FairyRing

- カジュアルな用途でも気軽に使える暗号プロトコルベースの秘密計算のフレームワーク
- 誰でも簡単に秘密計算を実装できる
  - 準同型性暗号ベースの秘密計算用のjavaフレームワーク
  - 少数のプリミティブを組み合わせて開発
  - 信頼できる第三者の排除(信用できないサーバ」の導入)
  - マルチパーティー対応
    - メッセージング・セッション管理等はフレームワークが提供
  - 出力プライバシー対応(差分プライバシー)
- どこでも簡単に秘密計算を利用できる
  - Androidを利用

http://www.mdl.cs.tsukuba.ac.jp/fairyring/

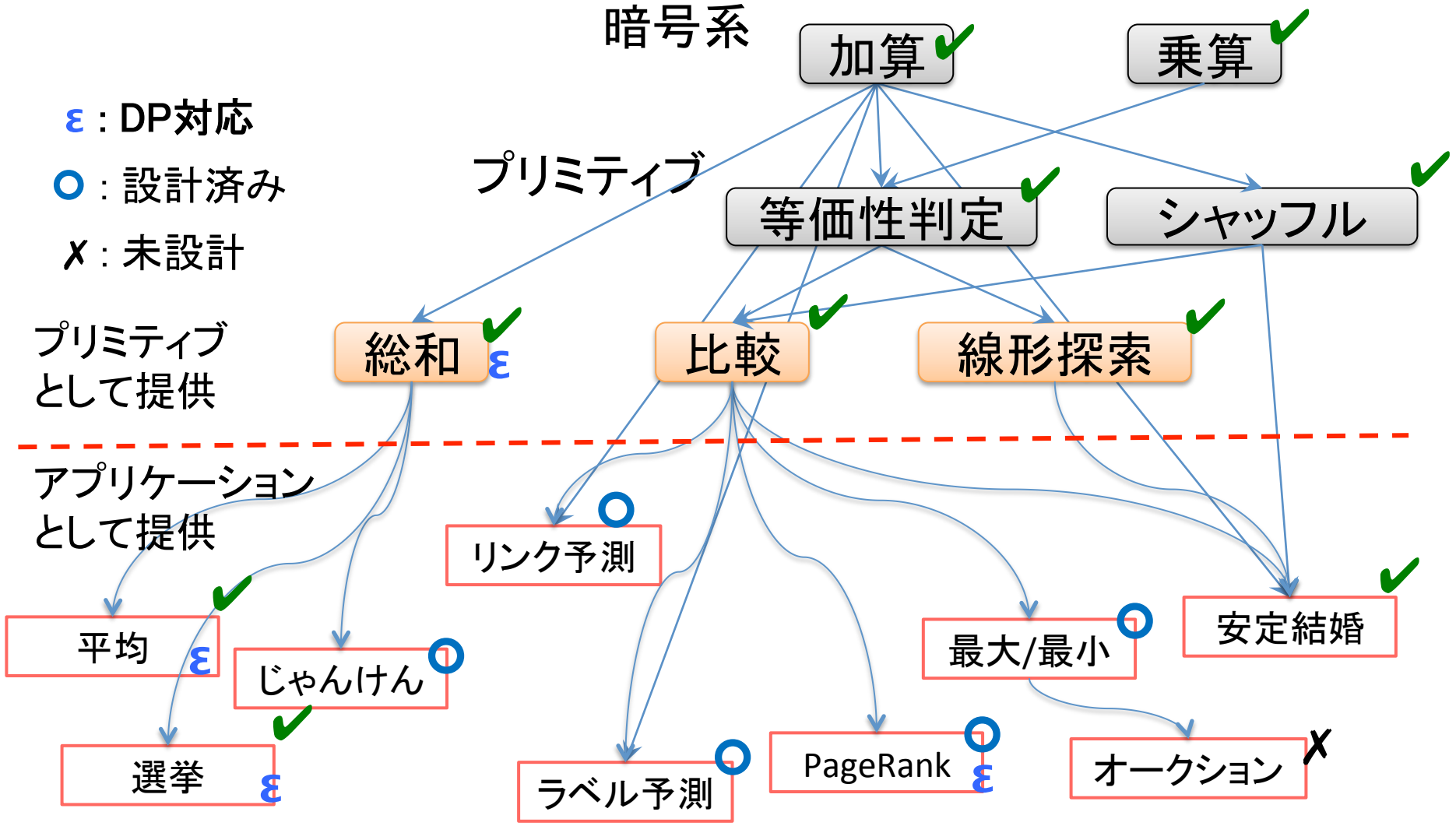
Fairy Ringのプリミティブとプロトコル

✔：実装済み

ε：DP対応

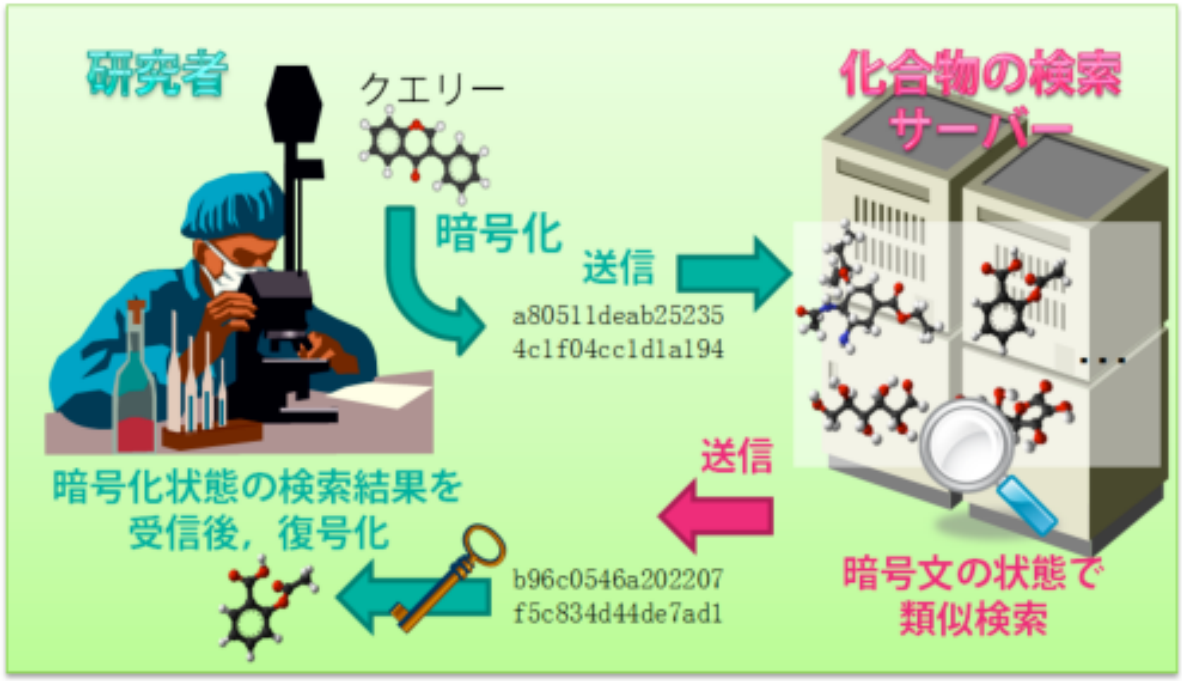
○：設計済み

✗：未設計



秘密計算応用:
化合物データベースに対するプライバシ保護検索

- 準同型性暗号を用いた検索プロトコル
- クエリと合致した化合物の数を返す
- サーバの管理者にはクエリが漏えいしない
- 検索者にはサーバが保持するデータが漏洩しない



特許を取得＆
プレスリリース


（第三種郵便物認可）

類似化合物

暗号化したまま検索

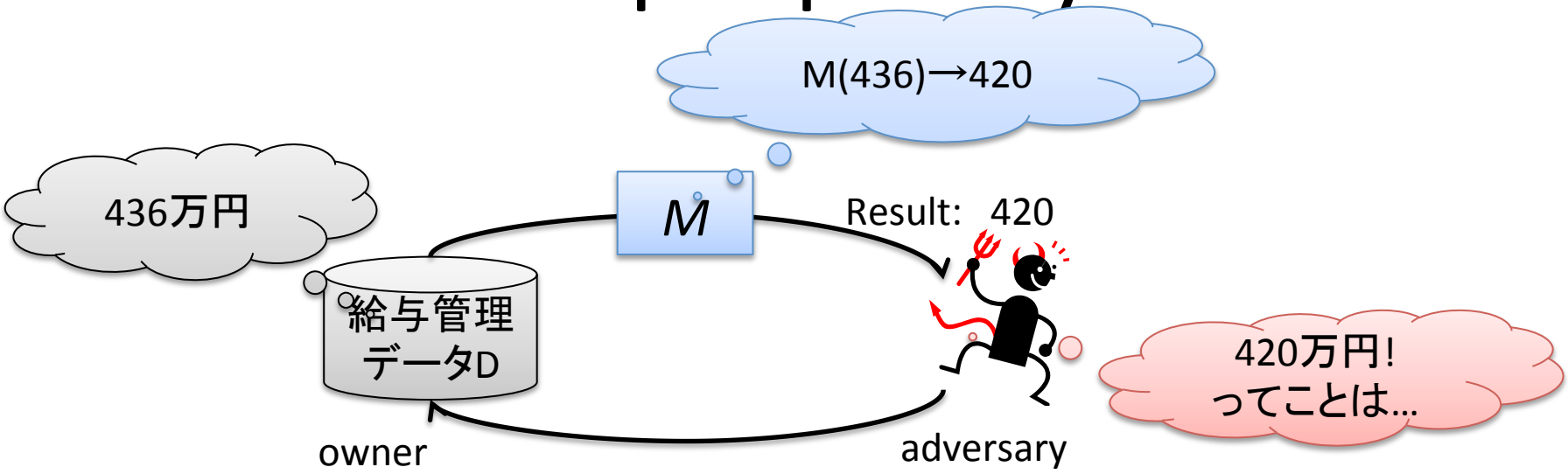
産総研―筑波大―東大

創薬研究で秘密保持

産業技術総合研究所生…功した。製薬企業などの…物検索サービスとして事…命情報工学研究センタ…間でも安全に情報交換可…業化につなげる計画。…、筑波大学、東京大学の…能で、おたがいの秘密を…開発した技術は、検索…ばれる方法で化合物の類…

産総研、筑波大、東大による共同研究開発

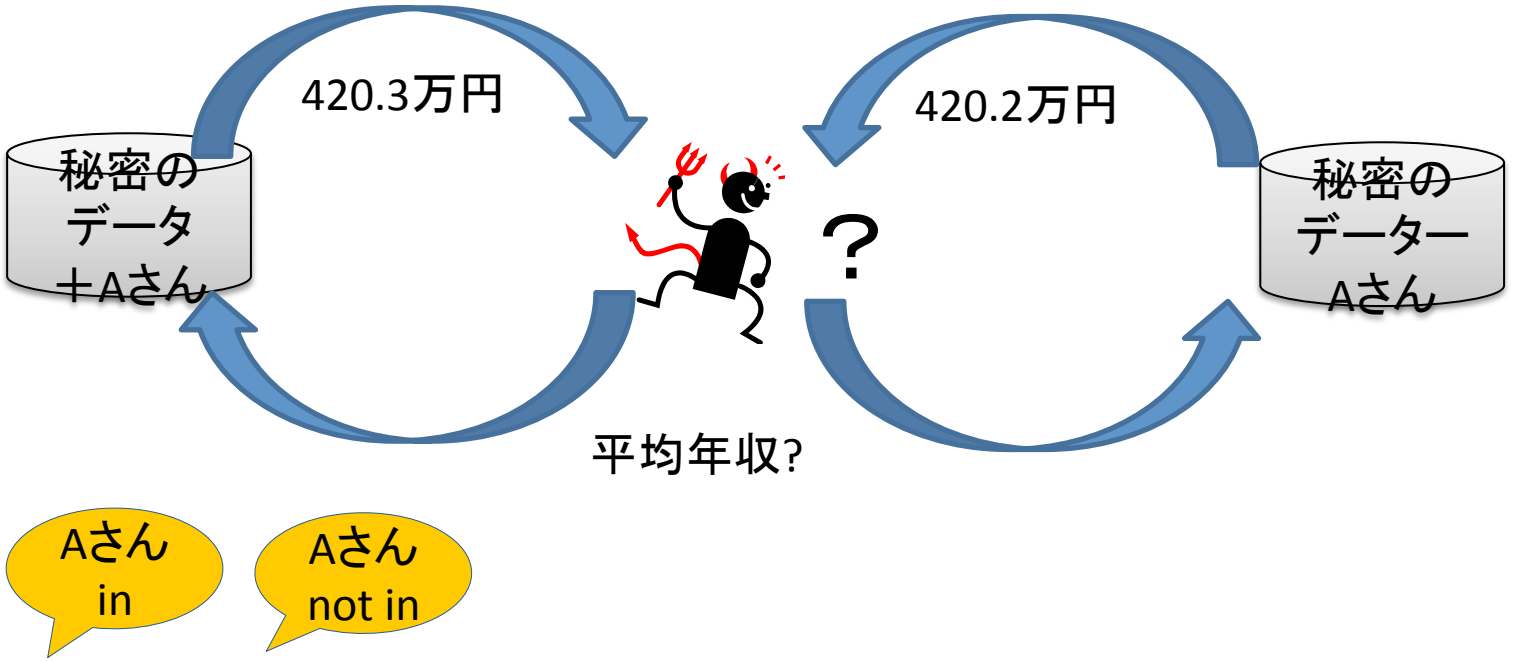
Output privacy



Query:入社3年目の平均年収?

- 攻撃者はクエリを発行
- オーナーはf(D)を計算
- Dの秘密を守るためにf(D)に細工Mをしてから開示
- 攻撃者はM(f(D))からDを推測
- 問題:どんなMなら安全なのか?

Output privacyにおける Semantic security

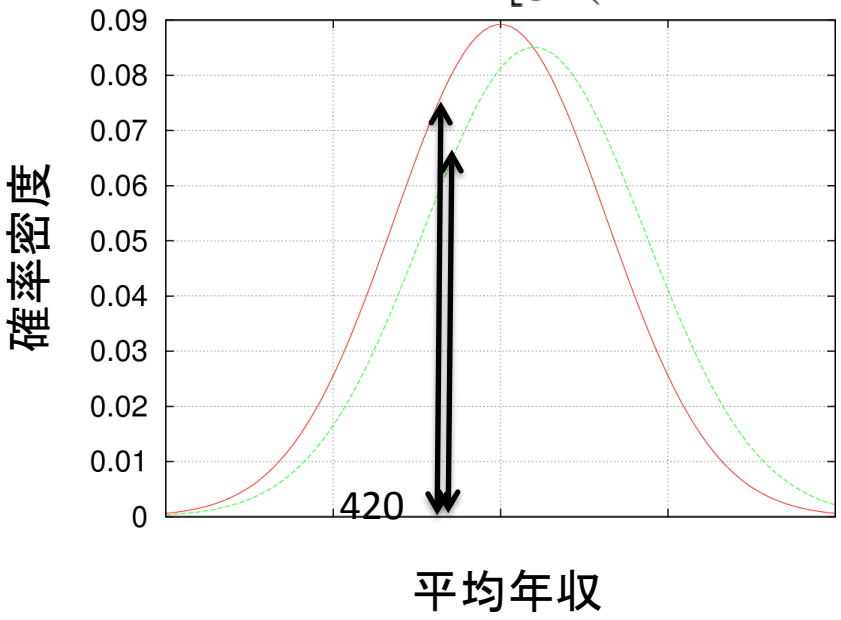


- f(D+A)とf(D)がそんなに変わらなければ、f(D+A)の開示はAさんのプライバシを侵害していない (ことにしよう！)

Differential Privacy

- **そんなに変わらないとはどう定義できるか…?**

$$\frac{\Pr[f(\text{DB-You}) \in S]}{\Pr[f(\text{DB+You}) \in S]} \leq e^{\epsilon}$$



あなたがDBに入っている場合と、そうでない場合で、平均年収は420万円という結果が返ってくる確率の比がe^{ε}以下ならオッケー

fの返す結果が, 正の小さいεについて, 上の条件を満たすならば, あなたがいようがいまいが、攻撃者がDBから推測できることは (ほぼ)一緒

Sensitivity

- 関数 $f : \mathcal{D} \mapsto \mathbb{R}^d$について、$f$ の感度(sensitivity)を定義
- *Definition: Sensitivity* $\Delta f = \max_{D_1, D_2} ||f(D_1) - f(D_2)||_1$
- 例1: f が平均の場合
  - データ数n, データの定義域を[0,N]とすると$\Delta f = N/n$
  - データ数が多ければsensitivityは低い
- 例2: f がmaxの場合
  - データ数n, データの定義域を[0,N]とすると$\Delta f = N$
  - データ数にかかわらずsensitivityは高い

どうやってDPを実現するか?
Laplaceメカニズム

- f(DB)に加法的ノイズを加えて開示
  - どんなノイズ?
- 定理: $R = \Delta f / \epsilon$ としたときに $\mathrm{Lap}(|x|/R)$ に従うノイズrを加えればε-differential privacyが達成される
- 証明:

$$\frac{\Pr[\mathcal{M}(f(\mathrm{DB} - \mathrm{You})) = t]}{\Pr[\mathcal{M}(f(\mathrm{DB} + \mathrm{You})) = t]} = \exp(-(|t - f^{-}| - |t - f^{+}|)/R) \leq \exp(-\Delta f/R)$$

- Sensitivityが小さい→分散の小さいノイズですむ

$$\mathrm{Lap}(x; R) = \frac{1}{2R} e^{-\frac{|x|}{R}}$$

Aさんがいるといないとで結果がどれくらい変わるか、で決まる定数
＝sensitivity

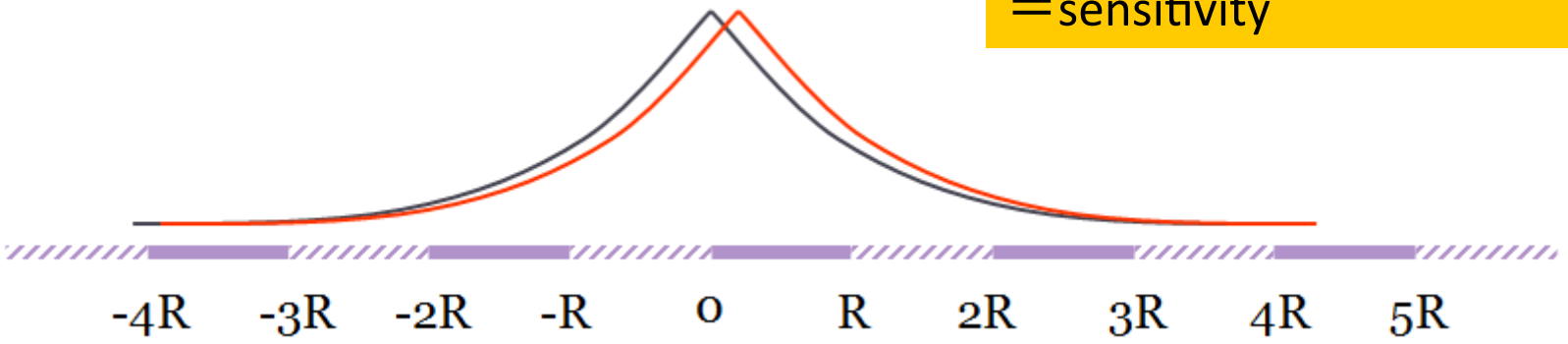
- 結局, $f(\mathrm{DB}) + r$ ただし $\mathrm{Lap}(x; \Delta f/\epsilon)$ を返せばよい

どうやってDPを実現するか?
Laplaceメカニズム

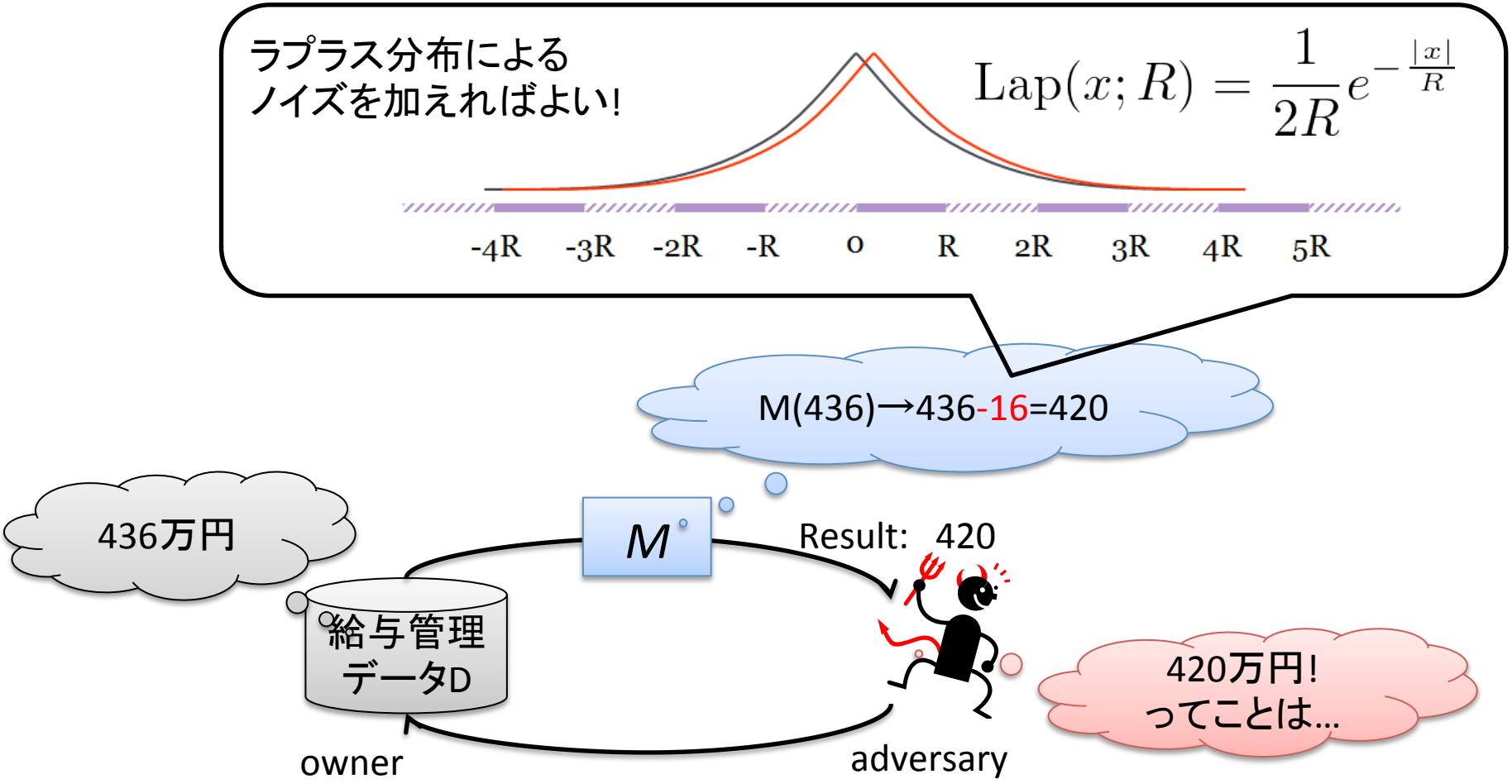
- f(DB)に加法的ノイズを加えて開示
- 定理: R = Δf/εとしたときにLap(|x|/R)に従うノイズrを加えればε-differential privacy
- Δf : fのデータの変化に対する感度

$$\mathrm{Lap}(x; R) = \frac{1}{2R} e^{-\frac{|x|}{R}}$$

Aさんがいるといないとで結果がどれくらい変わるか、で決まる定数
＝sensitivity

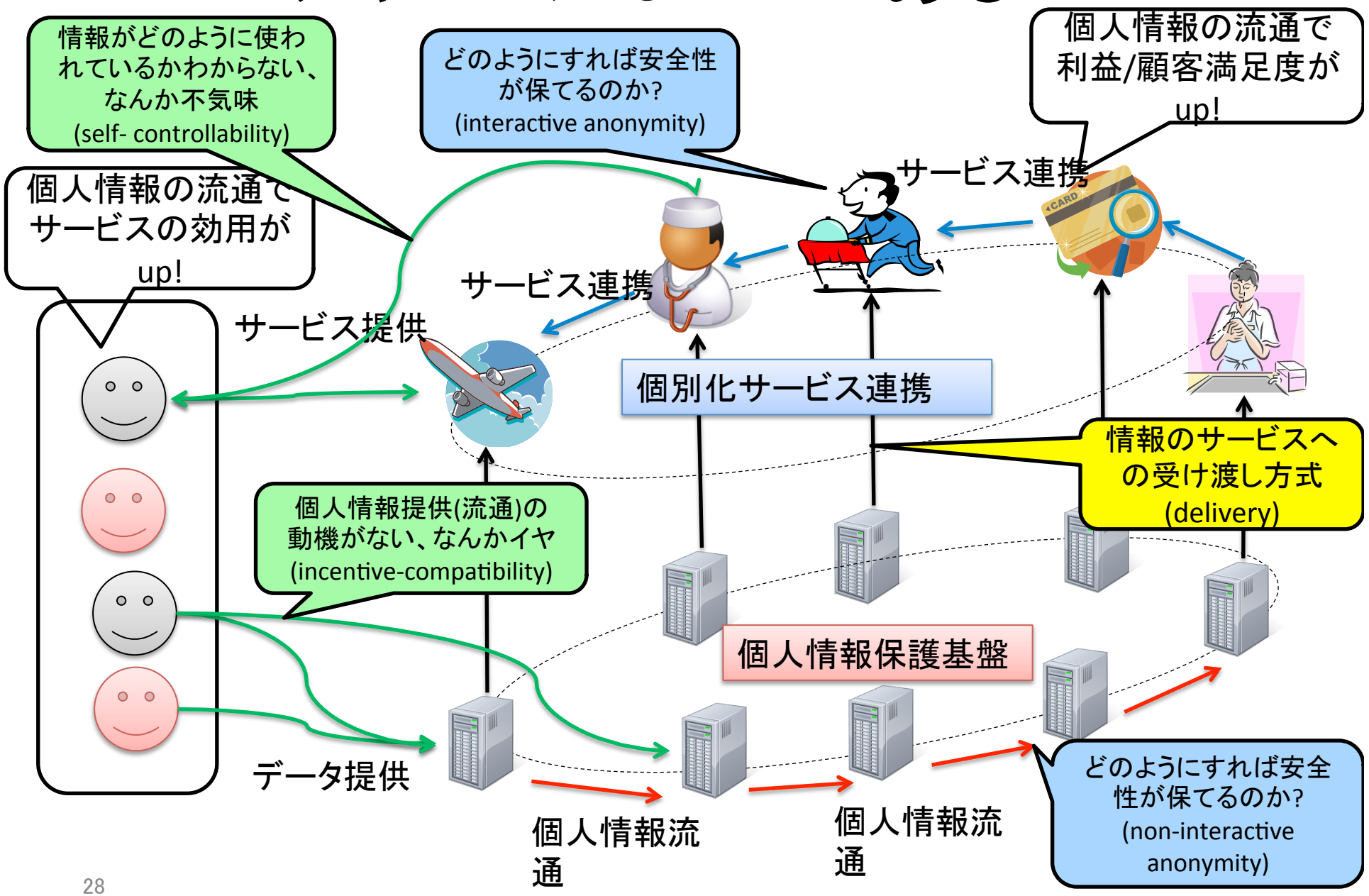


Output privacy



Query:入社3年目の平均年収?

チャレンジはどこにある？



個人情報提供(流通)の動機がない、なんかイヤ
incentive-compatibility (ユーザ視点)

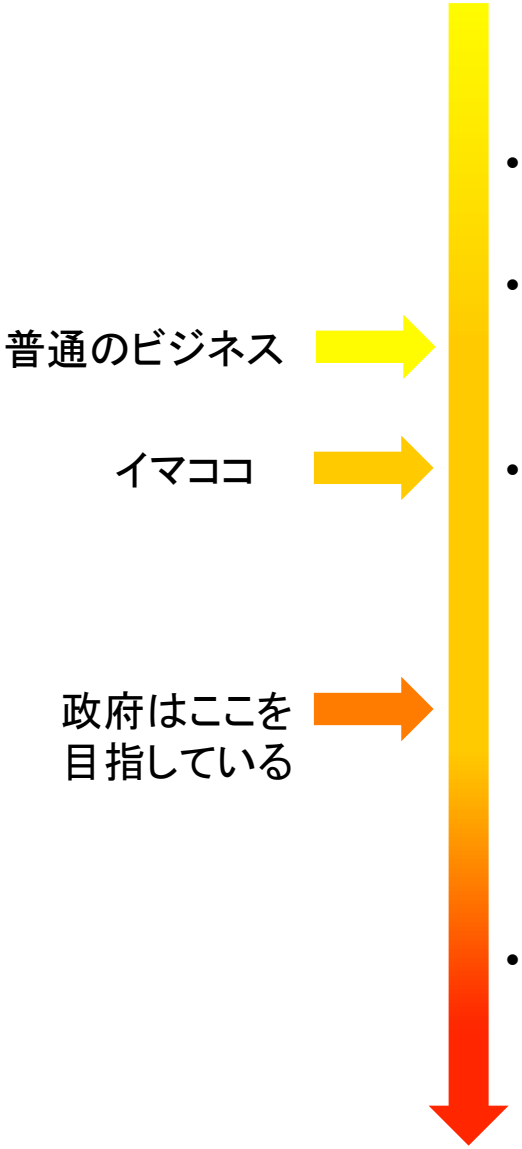
- Googleのフリーサービスとユーザ
  - Google検索, Gmail
    - 検索/メール情報と引き換えにサービスのpersonalization
  - Googleの”Don’t be evil”原則の信頼
    - ユーザ視点:サービスの便益＞情報提供のイヤさ
    - Google視点:個人情報取得が生む利益>サービスのコスト
    - Incentive compatibilityが保たれている
- 個人情報の流通(個人情報保護基盤)
  - ”Don’t be evil”原則をどうやって担保するか？
  - Incentive compatibilityの確保
    - ユーザ視点:サービスの便益＞情報提供のイヤさ
    - サービサ視点: 個人情報取得が生む利益>サービスのコスト
    - その実現は簡単ではない
- 制度設計(mechanism design)からのアプローチ
  - ユーザの便益の定量化、個人情報の流通財化, ユーザとサービサをプレイヤとしたゲーム理論的解析
  - 秘密計算とインセンティブ→semi-honest, malicious modelからrational modelへ

情報がどのように使われているかわからない、なんか不気味
self- controllability　(ユーザ視点)

- Data is the next intel inside (Tim O'Reilly) : コアデータをめぐる争いはすでに始まっている。こうしたデータの例としては、位置情報、アイデンティティ（個人識別）情報、公共行事の日程、製品の識別番号、名前空間などがある。作成に多額の資金が必要となるデータを所有している企業は、そのデータの唯一の供給元として、インテル・インサイド型のビジネスを行うことができるだろう。そうでない場合は、最初にクリティカルマスのユーザーを確保し、そのデータをシステムサービスに転換することのできた企業が市場を制する。

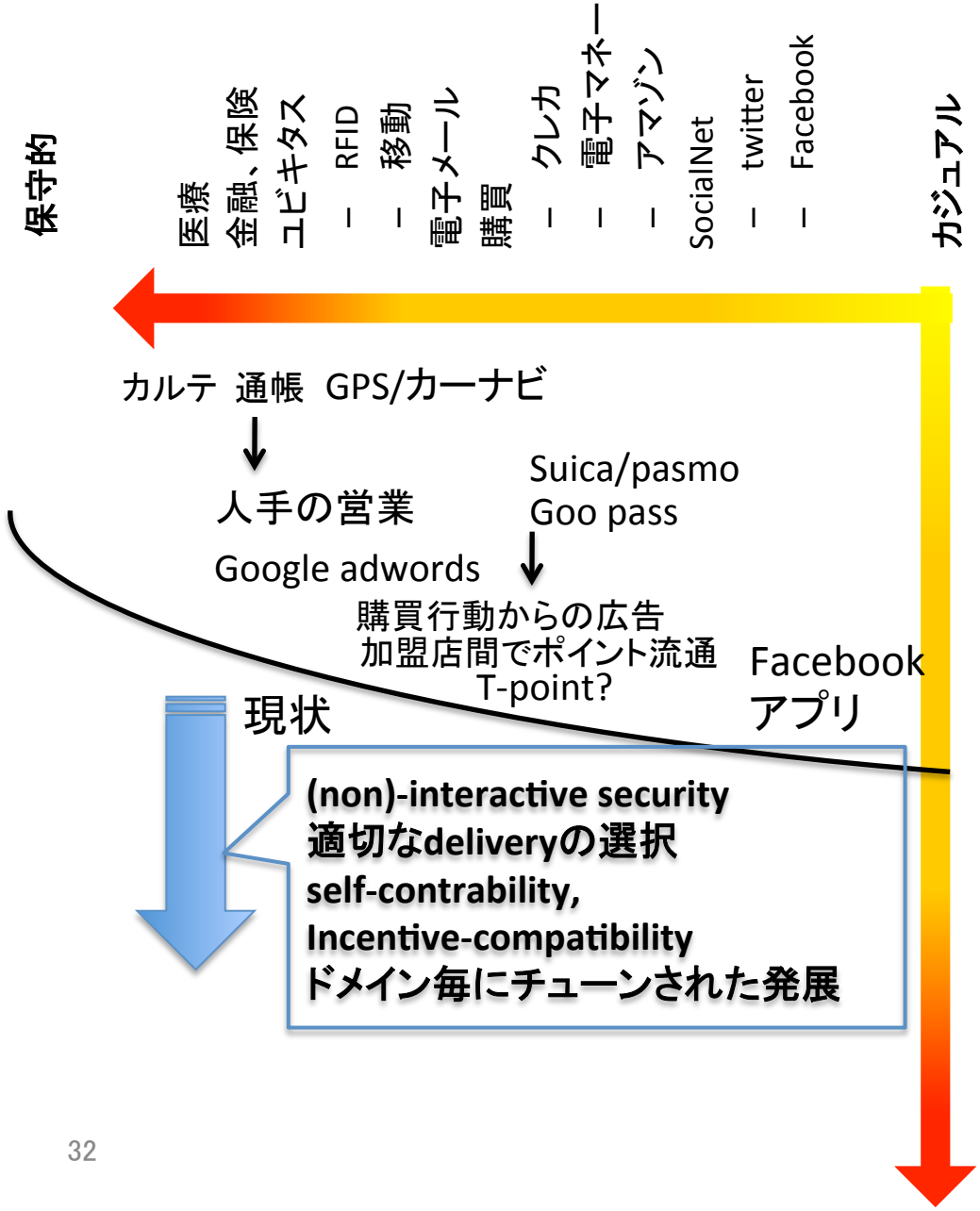
- Your data is your data: In some sense, open data is more important than open source. Your *data* is more valuable than the tools you use--you can always find new tools, but if you lose access to your data, no tool in the world will give you access to it. (とあるブロガー)
- ユーザデータのオーナシップをきちんと尊重し 、集めたデータで生み出した価値をユーザに還元する
- ロジックとデータの分離、ユーザへのデータ編集権限の委譲

個人情報の蓄積・流通の発展



- 蓄積のみ・未活用
  - サーバ側に蓄積・だれも活用していない
- 単体サーバでの活用
  - 個人と紐付けなし(スーパーなど)
  - 個人と紐付けあり(アマゾンなど)
- サーバ間流通(二次利用)
  - サーバからサーバへの個人情報の流通
    - 流通毎に契約・詳細設計
    - オプトイン, 事前許諾
  - 二次利用からエコシステムへ
    - 流通ルールの確立
    - オプトアウト, 事前許諾無し
- 発展形: 仮想化(情報銀行)、分散化、オープン化(api)

個人情報のサービスドメインと活用の進化



蓄積のみ・未活用

- サーバ側に蓄積・だれも活用していない

単体サーバでの活用

- 個人と紐付けなし(普通のビジネス)
- 個人と紐付けあり(アマゾンなど)

サーバ間流通(二次利用)

- サーバからサーバへの個人情報の流通
  - 流通毎に契約・詳細設計
  - オプトイン, 事前許諾
- 二次利用からエコシステムへ
  - 流通ルールの確立
  - オプトアウト, 事前許諾無し

発展形: クラウド、peer-to-peer、オープン化?