Sémantique et optimisation dans le langage C2QL

Santiago Bautista

juin - juillet 2017

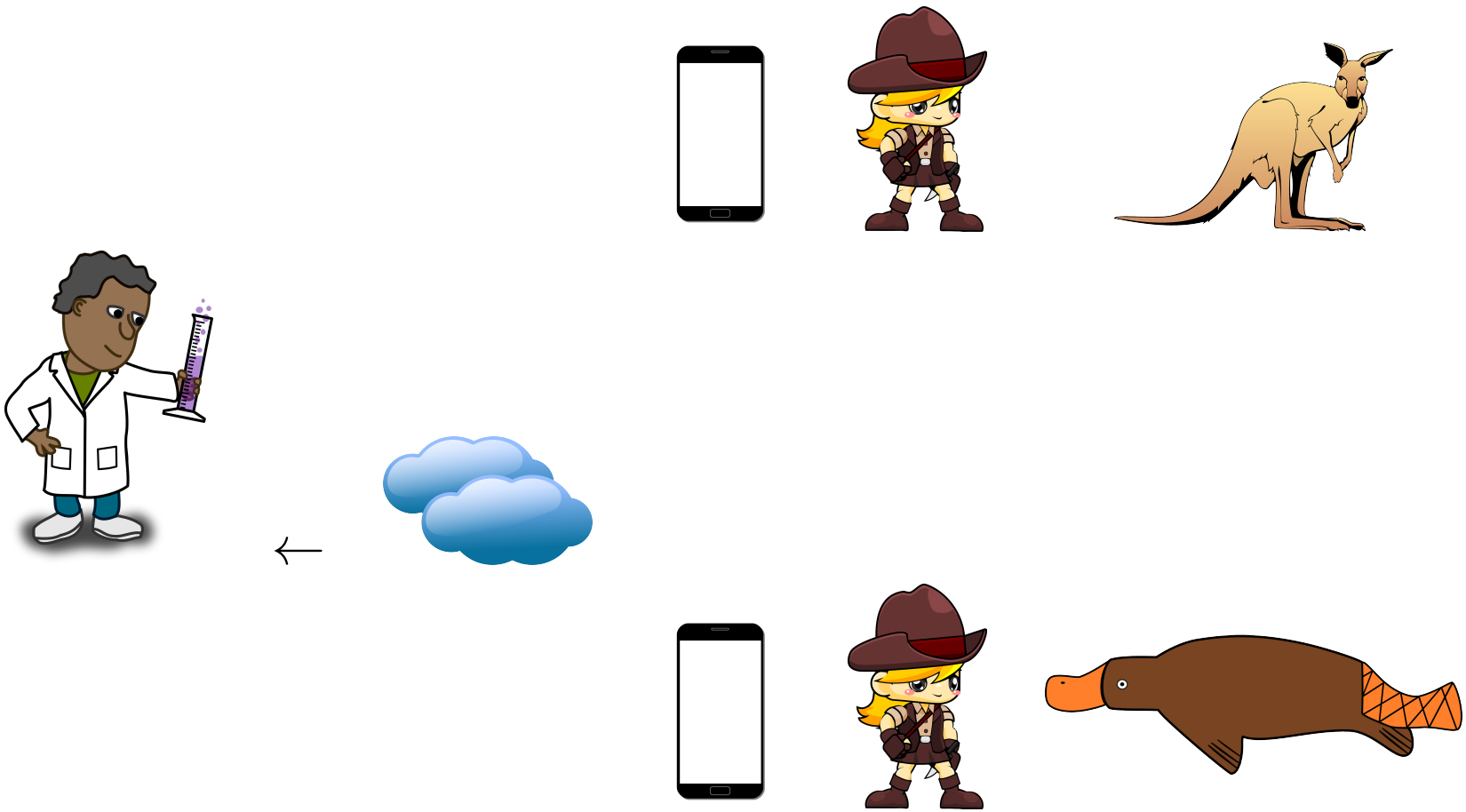
De plus en plus d'applications

$$\begin{cases} \text{Utilisent le nuage} \\ \text{Manipulent des données personnelles} \end{cases}$$

Problème

Comment protéger la confidentialité tout en privilégiant l'utilisation du nuage et les performances?

Exemple fil rouge



Exemple fil rouge

Exemple de données collectées:

- Animal vu (kangourou, ornithorynque)
- Date
- Lieu
- Nom de l'aventurier
- ...

Confidentialité

Un problème de confidentialité est défini par

- Qu'est-ce qu'on veut protéger?
- De qui veut-on le protéger?

Contraintes de confidentialité

Dans l'exemple on veut protéger

- L'association (date, lieu)
- Les noms des aventuriers

Hypothèses sur l'attaquant

On ne fait confiance

- Ni aux nuages utilisés
- Ni au réseau utilisé

On fait confiance

- A la machine de l'utilisateur.

Chiffrement

Définition

Transformer l'information intelligible en inintelligible de façon réversible pour les destinataires.

Chiffrement

Définition

Transformer l'information intelligible en inintelligible de façon réversible pour les destinataires.

Exemple

$\rightarrow$

Ana	Melton	4/8	LD6h736B	Melton	4/8
Fred	Tamworth	5/8	3ghjKrE	Tamworth	5/8
Ana	Orange	6/8	LD6h736B	Orange	6/8

$\leftarrow$

Fragmentation verticale

Définition

Séparer (géographiquement, par exemple) deux informations pour rendre inintelligible leur association.

Fragmentation verticale

Définition

Séparer (géographiquement, par exemple) deux informations pour rendre inintelligible leur association.

Exemple

Ana	Melton	9 août
Fred	Tamworth	10 août
Ana	Orange	11 août

Melton	Ana 9 août
Tamworth	Fred 10 août
Orange	Ana 11 août

Calculs côté utilisateur

On fait confiance à la machine de l'utilisateur

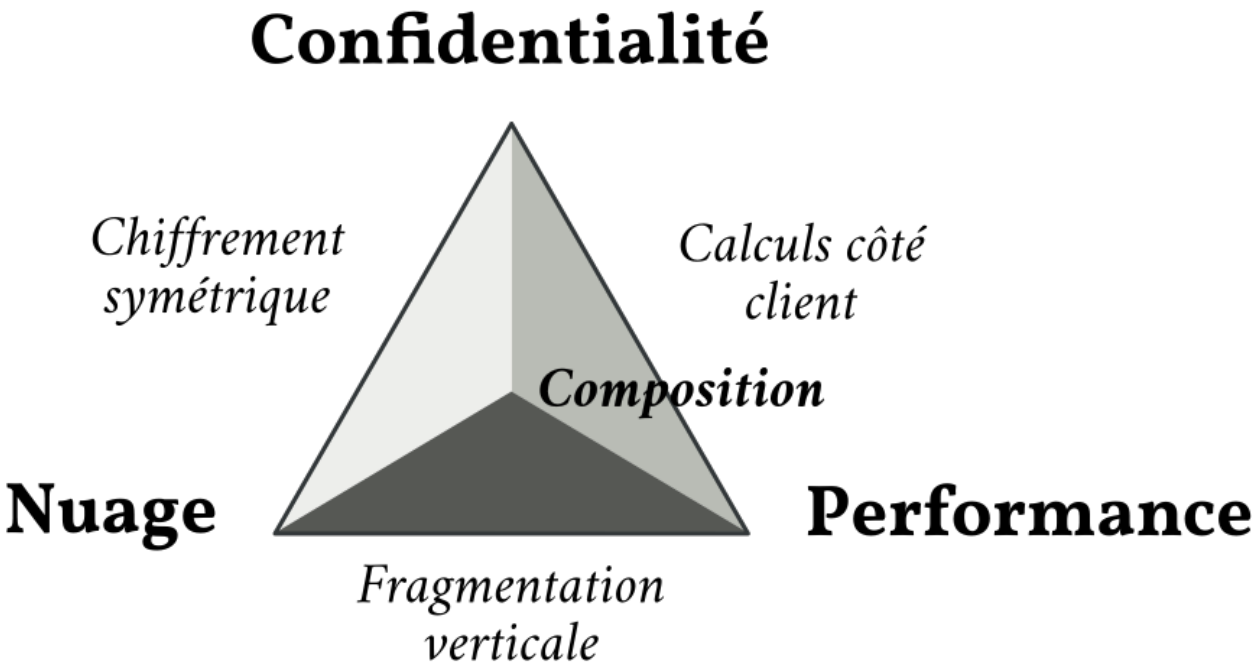
$\Downarrow$

Les informations traitées et stockées *localement* sont supposées en sécurité

Enjeux d'une application utilisant le nuage

- Utilisation du nuage
  - Disponibilité
  - Passage à l'échelle automatisable
- Protection de la confidentialité
- Performances

Enjeux et protection de la confidentialité



Relations

Nom	Lieu	Date	Animal
Ana	Melton	9 août	Kangourou
Fred	Tamworth	10 août	Ornithorynque
Ana	Orange	11 août	Ornithorynque

Ensemble des fonctions

- Projection, π
- Sélection, σ
- Jonction, $\bowtie$
- Agrégation et réduction, group et fold

Ensemble des fonctions

- Projection, π , π_{Animal}
- Sélection, σ , $\sigma_{|Aujourd'hui-date|<8jours}$
- Jonction, $\bowtie$, *Observations* $\bowtie$ *Aventuriers*
- Agrégation et réduction, group et fold
  $\text{fold}_{Animal,[nbK,esp \mapsto \text{if}(esp==Kang)(nbK+1) \text{ else } nbK],0} \circ \text{group}_{date}$

Ensemble des fonctions étendu

- Projection, π , π_{Animal}
- Sélection, σ , $\sigma_{|Aujourd'hui-date|<8jours}$
- Jonction, $\bowtie$, *Observations* $\bowtie$ *Aventuriers*
- Agrégation et réduction, group et fold
  $\text{fold}_{Animal,[nbK,esp \mapsto \text{if}(esp==Kang)(nbK+1)\ \text{else}\ nbK],0} \circ \text{group}_{date}$
- Fragmentation et défragmentation, frag et defrag

- Chiffrement et déchiffrement, crypt et decrypt

Ensemble des fonctions étendu

- Projection, π , π_{Animal}
- Sélection, σ , $\sigma_{|Aujourd'hui-date|<8jours}$
- Jonction, $\bowtie$, *Observations* $\bowtie$ *Aventuriers*
- Agrégation et réduction, group et fold
  $\text{fold}_{Animal,[nbK,esp \mapsto \text{if}(esp==Kang)(nbK+1)\ \text{else}\ nbK],0} \circ \text{group}_{date}$
- Fragmentation et défragmentation, frag et defrag
  $\text{defrag} \circ \text{frag}_{Lieu}$
- Chiffrement et déchiffrement, crypt et decrypt
  $\text{decrypt}_{AES,\texttt{Nom}}\ \text{crypt}_{AES,\texttt{Nom}}$

Le langage C2QL

En algèbre relationnelle

$$\#KangF = \mathsf{countK} \circ \pi_{date,Animal} \circ \sigma_{Nom=Fred \wedge |Aujourd'hui - date| < 8}$$

En C2QL

$$\#KangF = \mathsf{defrag} \circ (\pi_{\emptyset}, \mathsf{countK} \circ \pi_{date,Animal} \circ \sigma_{\mathsf{AES}(Nom)=3ghjKrE \wedge |Aujourd'hui - date| < 8}) \circ \mathsf{frag}_{\{Lieu\}} \circ \mathsf{crypt}_{AES,\mathtt{Nom}}$$

Développer un programme C2QL

D'abord, écrire version locale

$\mathsf{countK} \circ \pi_{date,Animal} \circ \sigma_{Nom=Fred \wedge |Aujourd'hui - date| < 8}$

Développer un programme C2QL

D'abord, écrire version locale

$$\mathsf{countK} \circ \pi_{date,Animal} \circ \sigma_{Nom=Fred \wedge |Aujourd'hui - date| < 8}$$

Ensuite, composer les protections/déprotection à droite

$$\mathsf{countK} \circ \pi_{date,Animal} \circ \sigma_{Nom=Fred \wedge |Aujourd'hui - date| < 8} \circ \mathsf{defrag} \circ \mathsf{frag}_{Lieu} \circ \mathsf{decrypt}_{AES,\mathtt{Nom}} \circ \mathsf{crypt}_{AES,\mathtt{Nom}}$$

Développer un programme C2QL

Ensuite, composer les protections/déprotection à droite

$$\text{countK} \circ \pi_{date,Animal} \circ \sigma_{Nom=Fred \wedge |Aujourd'hui - date| < 8} \circ \text{defrag} \circ \text{frag}_{Lieu} \circ \text{decrypt}_{AES,\texttt{Nom}} \circ \text{crypt}_{AES,\texttt{Nom}}$$

Développer un programme C2QL

Ensuite, composer les protections/déprotection à droite

$$\begin{aligned}\text{countK} \circ \pi_{date,Animal} \circ \sigma_{Nom=Fred \wedge |Aujourd'hui - date| < 8} \\ \circ \text{defrag} \circ \text{frag}_{Lieu} \circ \text{decrypt}_{AES,\text{Nom}} \circ \text{crypt}_{AES,\text{Nom}}\end{aligned}$$

Finalement, faire commuter les opérateurs

$$\begin{aligned}\#KangF = \text{defrag} \circ (\pi_{\emptyset}, \text{countK} \circ \pi_{date,Animal} \circ \\ \sigma_{\text{AES}(Nom)=3ghjKrE \wedge |Aujourd'hui - date| < 8}) \\ \circ \text{frag}_{\{Lieu\}} \circ \text{crypt}_{AES,\text{Nom}}\end{aligned}$$

Choix à faire: Tuples ou fonctions?

Nom	Lieu	Date	Animal
Ana	Melton	9 août	Kangourou
Fred	Tamworth	10 août	Ornithorynque
Ana	Orange	11 août	Ornithorynque

$f : Nom \mapsto Fred, Lieu \mapsto Tamworth,$

$Date \mapsto 10/08, Animal \mapsto Ornithorynque$

Choix à faire: Expressivité ou restrictivité?

L'implémentation faite en Idris évite déjà les erreurs de programmation

$$\Downarrow$$

Pour les démonstrations, les expression peuvent être expressives

Choix à faire: Expressivité ou restrictivité?

L'implémentation faite en Idris évite déjà les erreurs de programmation

$\Downarrow$

Pour les démonstrations, les expression peuvent être expressives

Exemple

$$\pi_\delta : \quad r \quad \mapsto \quad \{l|_{(\delta \cap \mathsf{sch}(r)) \cup \{id\}} / l \in r\}$$

Structure des preuves

$$\begin{cases} res_1 = & (f \circ g)(r) \\ res_2 = & (g \circ f)(r) \end{cases}$$

- Égalité des schémas relationnels
- Inclusion de res_1 dans res_2
- Inclusion de res_2 dans res_1

Erreurs corrigées: projection et projection

Loi 3 page 30 de la thèse

$$\pi_{a_1} \circ \cdots \circ \pi_{a_n} \equiv \pi_{a_1,\ldots,a_n}$$

Contre-exemple

a_1	a_2
a	1
b	2

$\pi_{a_1} \circ \pi_{a_2}(r) = \emptyset$ mais $\pi_{a_1,a_2}(r) = (r)$

Correction

$$\pi_{\delta_1} \circ \cdots \circ \pi_{\delta_n} \equiv \pi_{\delta_1 \cap \cdots \cap \delta_n}$$

Erreurs corrigées : Defragmentation et projection

Loi 12 page 64 de la thèse

$$\pi_\delta \circ \mathsf{defrag} \equiv \mathsf{defrag} \circ (\pi_{\delta \cap \delta'}, \pi_{\delta \setminus \delta'})$$

Correction

$$\pi_\delta \circ \mathsf{defrag} = \mathsf{defrag} \circ (\pi_{\delta \cap \delta_1}, \pi_{\delta \cap \delta_2}) \qquad \text{si } \delta_1 \cap \delta_2 = \emptyset$$

Critère de complétude

Toutes les paires de fonctions possibles soient considérées

Tableau des lois

	proj	sel	frag	defrag	crypt	decrypt	join	group	fold
proj	M	M	-	M	-	C	M	A	A
sel		C	-	C	-	C	M	A	A
frag			-	C	C	C	-	-	-
defrag				-	A	C	A	C	A
crypt					A	C	-	-	-
decrypt						A	A	M	A
join							A	A	A
group								A	A
fold									A

Exemple

Soit δ_1 le schéma relationnel du premier argument et δ_2 le schéma relationnel du deuxième argument. En appelant (P) la propriété Soit c est injectif, soit $\alpha \notin \delta_1 \cap \delta_2$,

$$\mathsf{decrypt}_{\alpha,\mathsf{c}} \circ \bowtie \equiv \bowtie \circ (\mathsf{decrypt}_{\alpha,\mathsf{c}}, \mathsf{id}) \quad \text{si } \alpha \in \delta_1 \text{ et } (P) \quad (1)$$

$$\mathsf{decrypt}_{\alpha,\mathsf{c}} \circ \bowtie \equiv \bowtie \circ (\mathsf{id}, \mathsf{decrypt}_{\alpha,\mathsf{c}}) \quad \text{si } \alpha \in \delta_2 \text{ et } (P) \quad (2)$$

Automatiser l'optimisation

Autres pistes à explorer

- Autres propriétés de sécurité
- Autres mécanismes
- Compilateur C2QL application concrète
- Finir compilateur vers Provérif
- ...

Conclusion