

どんな研究？

アジャイル開発やリーン開発など、小さく作り段階的に同時並行的にソフトウェア発展させていく手法が広まっていますが、データベースにおいては、そのデータベーススキーマを柔軟に変更し、複数のバージョンを同時利用することが難しくなっています。そこでビューを用いてデータベーススキーマを設計し、更新可能にしたビューの間でデータ更新を共有することで、スキーマの共存を実現する研究を進めています。

何がわかる？

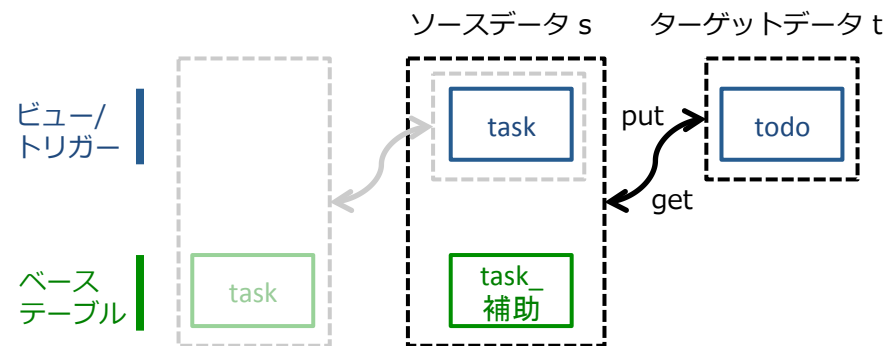
- データベース移行ツールやデータベースリファクタリングとは何が違うか？
- データベーススキーマを発展させていくとは？
- 更新可能なビューとは？
- データベーススキーマ間でデータを共有するとは？
- 双方向変換とは？

データベーススキーマの共存とは

元のスキーマ			新しいスキーマ			
task			todo			
auth	task	prio	auth	task	prio	
Roy	go dentist	2	Ann	write paper	1	
+Ann	write paper	1	← prio=1	+Ada	clean room	1
Ada	clean room	1	← prio=1	-Tom	go dentist	+
Tom	go dentist	+	← prio=1	+Ron	buy book	4

- 元のスキーマから新しいスキーマへ発展が可能 (selectionによるテーブルの分割やカラム追加、など)
- それぞれのスキーマにおいて、任意の更新が可能
- 設計者の意図に応じたデータの共有のルールが設計可能

研究内容



- 更新されたデータの共有ルールをput関数として記述します
 $put(s, t) = s'$
- 更新に際して副作用を及ぼさないように、双方向変換の特性を満たすget関数を自動的に出力します
 $get(s) = t$