Synthetische Datengenerierung aus PostgreSQL für PostgreSQL

Studenten

Jari Elmer

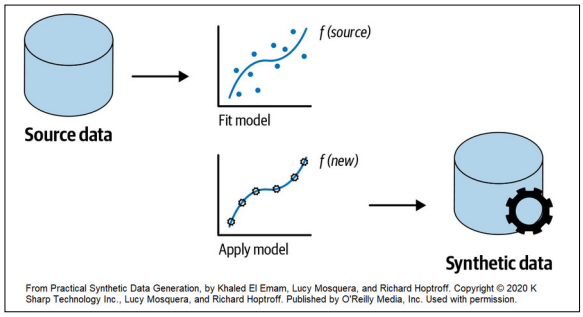
Timon Erhart

Ausgangslage: In den Bereichen Software- und Data-Engineering sowie beim Machine-Learning besteht eine grosse Nachfrage nach umfangreichen Datensätzen. Dabei ist der Datenschutz oftmals ein Grund, keine Real-World-Daten zu verwenden. Eine Pseudonymisierung bietet keinen vollständigen Schutz vor De-Anonymisierungs-Attacken. Eine komplette Anonymisierung wäre eine mögliche Lösung. Doch diese ist aufwändig und zeitintensiv. Am vielversprechendsten sind rein synthetisch generierte Daten mit ähnlichen statistischen Eigenschaften wie die Originaldaten. Das hat den zusätzlichen Vorteil, dass beliebige Datenmengen erzeugt werden können.
Ein an der OST entwickelter Softwareprototyp "pgsynthdata" setzt darum auf das Prinzip der rein synthetischen Datengeneratoren. Das Kommandozeilen-Tool ist in der Lage, fast beliebige PostgreSQL-Datenbanken zu synthetisieren und in eine generierte Datenbank mit gleicher Struktur und vergleichbaren statistischen Eigenschaften abzufüllen. Als Grundlage dienen der Katalog von PostgreSQL für funktionale Abhängigkeiten sowie Statistiken, die PostgreSQL zum Zwecke der Anfrageoptimierung erstellt. Ein Alleinstellungsmerkmal des Tools ist, dass es ohne aufwändige manuelle Konfiguration auskommt.

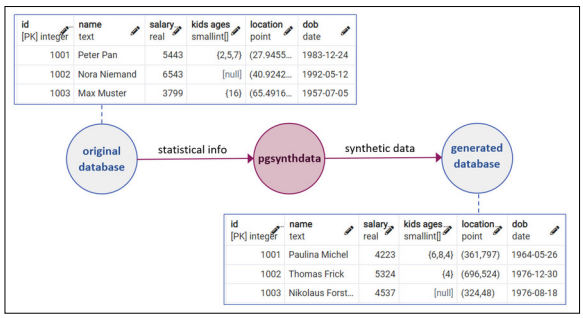
Vorgehen: Das Ziel dieser Arbeit war es, die bestehende Software pgsynthdata in einen wartbaren, erweiterbaren und einfach zu nutzenden Zustand zu überführen. Zudem wurde die Funktionalität erweitert um Generatoren für Datentypen wie enumerated Types (Enum), Arrays und Geometrien. Diese Erweiterungen wurden getestet und dokumentiert.
Dazu war zu Beginn ein umfassendes Refactoring des gesamten Tools nötig. Der Fokus lag dabei auf einem Plugin-System, das es ermöglicht, neue Generatoren einfacher zu integrieren. Nach der Übernahme der bestehenden Generatoren ins neue System, wurde ein Array-Generator für numerische Typen, ein Generator für PostGIS-Geometrie-Typen und ein Enum-Generator implementiert. Aus dem Bedürfnis heraus, statt zufälliger Zeichenketten realistische Eigennamen generieren zu können, ist eine Möglichkeit entstanden, per Spalten-Kommentar im Schema spezifische Generatoren zu konfigurieren. Die implementierten Generatoren für Eigennamen und Postadressen verdeutlichen dieses Konzept.

Fazit: Entstanden ist ein wartbares und erweiterbares Programm mit einem flexiblen Plugin-System und mit Generatoren für eine Vielzahl von Datentypen. Constraints wie Unique- oder Foreign-Keys werden unterstützt. Bei nicht unterstützten Datentypen wird dem User eine entsprechende Warnung angezeigt. Durch spezielle Konfigurationen lassen sich solche Spalten eins-zu-eins kopieren oder die Warnung ignorieren.
Das Tool ist im Stande aus verschiedenen PostgreSQL-Datenbanken synthetische Daten zu generieren. Durch die Entkopplung der Softwaremodule konnte ausserdem die Testabdeckung stark verbessert werden. Das Konfigurieren der spezifischen Generatoren per Spalten-Kommentar hat sich als benutzerfreundlich und unkompliziert bewährt.
Weiterhin gibt es viele denkbare Erweiterungen, um die Funktionalität des Tools zu erhöhen, sowie weitere Datentypen zu unterstützen.

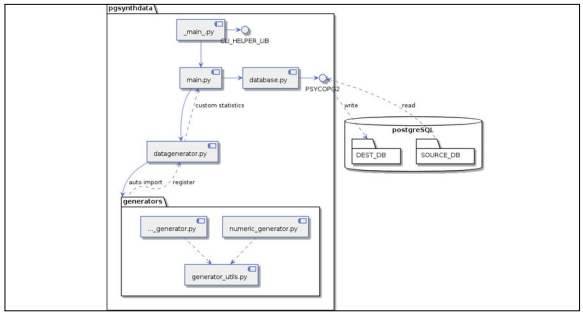
Generierung von synthetischen Daten (Quelle: El Emam et al. 2020: Practical Synthetic Data Generators)
Mit freundlicher Genehmigung: © O'Reilly Media



Funktionsweise des synthetischen Datengenerators pgsynthdata
Eigene Darstellung



Software-Komponentendiagram nach Refactoring (UML-Notation)
Eigene Darstellung



Examinatoren
Prof. Stefan F. Keller, Nicola Jordan

Themengebiet
Software, Software Engineering - Core Systems, Application Design