

## Automatisierungs-Tools der dritten Generation

Die formalen Grenzen klassischer Data-Warehouses sprengen



Lange Zeit nutzten Unternehmen für ihre datenbasierten, strategischen Entscheidungen ausschließlich Daten, die in Data-Warehouse-Architekturen lagerten. In den letzten Jahren hat jedoch eine neue Architektur-Generation die Bühne betreten. Datenspeicher wie Data Lakes, Data Hubs oder Data Lakehouses speichern und nutzen die Daten in völlig anderer Weise. Konventionelle Methoden der Automatisierung, wie sie in Data-Warehouses durchgeführt werden, greifen hier nicht. Dazu bedarf es eines neuen Typus von Code-Generator. Dieser sollte über die Automatisierung etablierter Data-Warehouse-Plattformen hinausgehen können und auch Datenarchitekturen digitalisieren und automatisieren, die nicht auf Basis traditioneller Prinzipien erstellt wurden.



Autor:  
Thomas Heuer,  
Sales Director DACH  
WhereScape  
[www.wherescape.com](http://www.wherescape.com)

### Plattform als komplexe digitale Architektur

Die meisten modernen Data-Warehouses verwenden heutzutage für ihre Automatisierung Code-Generatoren, die die gesamte Plattform als komplexe digitale Architektur erzeugen, einschließlich der Staging-Bereiche, Enterprise Data Warehouses, physischen Data Marts, ETL-Lösungen, die Daten von einer Datenbank in eine andere kopieren oder Metadaten. Die Speicherung, Pflege und Erweiterung der unternehmensweiten Daten lässt sich so mit wenigen Klicks umsetzen, was ansonsten einen Großteil der Entwicklungszeit in Anspruch nehmen würde. Die Automatisierungswerkzeuge speichern alle Metadaten-Spezifikationen einmal ab und verwenden sie erneut, wenn sie beispiels-

weise Data-Warehouse- oder Data-Mart-Tabellen und die ETL-Logik zum Kopieren der Daten erzeugen.

### Nicht jede Aufgabe eignet sich für eine Automatisierung

Allerdings lässt sich nicht jede Aufgabe auf Basis von Code-Generatoren sinnvoll automatisieren. Geeignet sind vor allem repetitive Prozesse mit vielen kleinteiligen Arbeitsschritten, die sich als „Wenn-Dann-Logik“ algorithmisch darstellen lassen. Sie sollten genaue Vorgaben enthalten, welche Schritte auszuführen sind, was bei Sonderfällen zu tun ist und wie Anwender reagieren müssen, falls etwas nicht läuft wie geplant. Kurz: Automatisierbare Aufgaben und Prozesse müssen leicht zu standardisieren sein.

Daten aus Datenspeichern wie Data Hubs, Data Lakes oder Data Lakehouses, die Daten in einer wesentlich umfangreicheren und flexibleren Form nutzen als Data-Warehouses. So kommt es beispielsweise vor, dass eine neue Datenarchitektur Daten in einen Data Hub kopiert und von dort in ein Data Warehouse migriert. Andere moderne Datenarchitekturen könnten einen Data Lake enthalten, der Daten von einem Data Warehouse, transaktionalen Datenbanken und weiteren externen Datenquellen speichert.

## „Out-of-the-box-Architekturen“

Um eine große Bandbreite von Datenarchitekturen unterstützen zu können, bedarf es Code-Generatoren, für die neue, nennen wir sie „Out-of-the-box-Architekturen“, kein Hindernis darstellen. Die dritte Generator-Generation, wie sie es WhereScape bietet, wird die formalen Grenzen klassischer Data-Warehouses sprengen und eine Vielzahl an Architekturen erzeugen und automatisieren können – selbst zukünftige und solche, die bislang noch unbekannt sind. Sie wird weit über das hinausgehen, was wir heute als Data-Warehouse-Automatisierungs-Software kennen. Die dritte Generator-Generation sollte man deshalb besser auch als Daten-Architektur-Automatisierungs-Tool bezeichnen. Das drückt es präziser aus. ◀

## Über WhereScape

WhereScape ist einer der Marktführer im Bereich der Automatisierung von Data Warehouses und hilft IT-Unternehmen aller Größenordnungen die Automatisierung zu nutzen, um Dateninfrastrukturen schneller zu entwerfen, zu entwickeln, zu implementieren und zu betreiben.

Die Automatisierung ermöglicht es, manuelle Codierung und andere sich wiederholende, zeitintensive Aspekte von Dateninfrastrukturprojekten zu eliminieren und Data Warehouses, Data Vaults, Data Lakes und Data Marts in Tagen oder Wochen statt in Monaten oder Jahren bereitzustellen.



Ein Enterprise Data Warehouse arbeitet beispielsweise standardmäßig mit Data Vault-Modellierungen und nutzt dafür physische Data Mart-Sternschemata. Ein geeignetes Automatisierungs-Tool sollte imstande sein, beide Modelle aus einem einzigen zentralen Datenmodell zu generieren, einschließlich des ETL-Codes, der nötig ist, um die Daten aus dem Warehouse in die Data Marts zu kopieren. Weitere repetitive Aufgaben, die sich für eine Automatisierung eignen sind beispielsweise Änderungen bei der Länge eines Feldes, das Ersetzen eines SQL-Datenbankservers durch einen anderen oder das Ändern einer Datenstruktur von einem Schneeflockenmuster in ein Sternschema.

## Generation Eins: Erzeugen einzelner Data Warehouse-Komponenten

Die Erstellung von Daten-Plattformen erfolgt, wie schon erwähnt, auf Basis von Code-Generatoren, die Spezifikationen von einer höheren Ebene in die Programmiersprache der gewählten Zielpattform übersetzen und so die Digitalisierung von Prozessen ermöglichen. Bereits die erste Generator-Generation hat mächtige Werkzeuge wie beispielsweise ETL-, BI- oder Datenmodellierungstools hervorgebracht. So besteht die Aufgabe von ETL-Tools vorwiegend darin, Spezifikationen von einer höheren Ebene auf eine

niedrigere Code-Ebene zu transformieren, um die eigentlich ETL-Aufgaben auszuführen. Auch viele BI-Tools übernehmen die Rolle von Code-Generatoren. Erstellen sie doch SQL-Abfragen, die je nach Anforderung Daten aus einer Datenbank selektieren, eintragen, aktualisieren oder löschen können. Data-Science-Werkzeuge ermöglichen Datenwissenschaftlern wiederum die Arbeit auf einer hohen konzeptionellen Ebene, deren Resultate sie anschließend in Code umsetzen.

## Die erste Generation von Code-Generatoren

erignet sich hervorragend, um Entwicklungs- und Wartungsprozesse eines Data-Warehouse zu beschleunigen und zu vereinfachen. Ihr Nachteil: Die Generatoren übersetzen jeweils nur eine einzelne Komponente der Datenarchitektur in Code. Um eine gesamte Data-Warehouse-Architektur zu erstellen, ist demzufolge der Einsatz einer Vielzahl von Code-Generatoren notwendig. Da sie alle ähnliche Spezifikationen erfordern, ist es möglich sie mehrfach zu definieren oder einfach zu duplizieren. Die Herausforderung für das Datenteam besteht nun darin, das Sammelsurium an Spezifikationen kontinuierlich aktuell zu halten und dafür zu sorgen, dass die Generatoren perfekt zusammenarbeiten. Bei einer Änderung müssen natürlich auch alle Duplikate angepasst werden.

## Generation Zwei: Erzeugen der gesamten Data Warehouse-Architektur

Aufgrund der ähnlichen Spezifikation lassen sich die Regeln mit denen ein Code-Generator eine einzelne Plattformkomponente erzeugt auch auf die Erstellung ganzer Datenarchitekturen übertragen. Daher war es nur eine Frage der Zeit, bis auch Code-Generatoren entwickelt wurden, die eine Data-Warehouse-Plattform als Ganzes erzeugen konnten. Diese zweite Generator-Generation hat sich mittlerweile als Data-Warehouse-Automatisierungs-Tool im Markt etabliert und sorgt seit Jahren dafür, dass die Prozesse in den Data-Warehouses schneller und effizienter werden.

## Generation Drei: Erzeugen einer Vielzahl von Architekturen

Der nächste logische Entwicklungsschritt wären Automatisierungstools, die in der Lage sind alle Datenarchitekturen - nicht bloß Data-Warehouses - zu automatisieren. Die dritte Generation von Code-Generatoren könnte im Zeitalter von Big Data die Entscheidungsprozesse von Unternehmen strategisch auf einer sehr viel breiteren Datenbasis unterstützen, als es bisher der Fall ist. Denn Unternehmen und Organisationen arbeiten heutzutage zunehmend auch mit