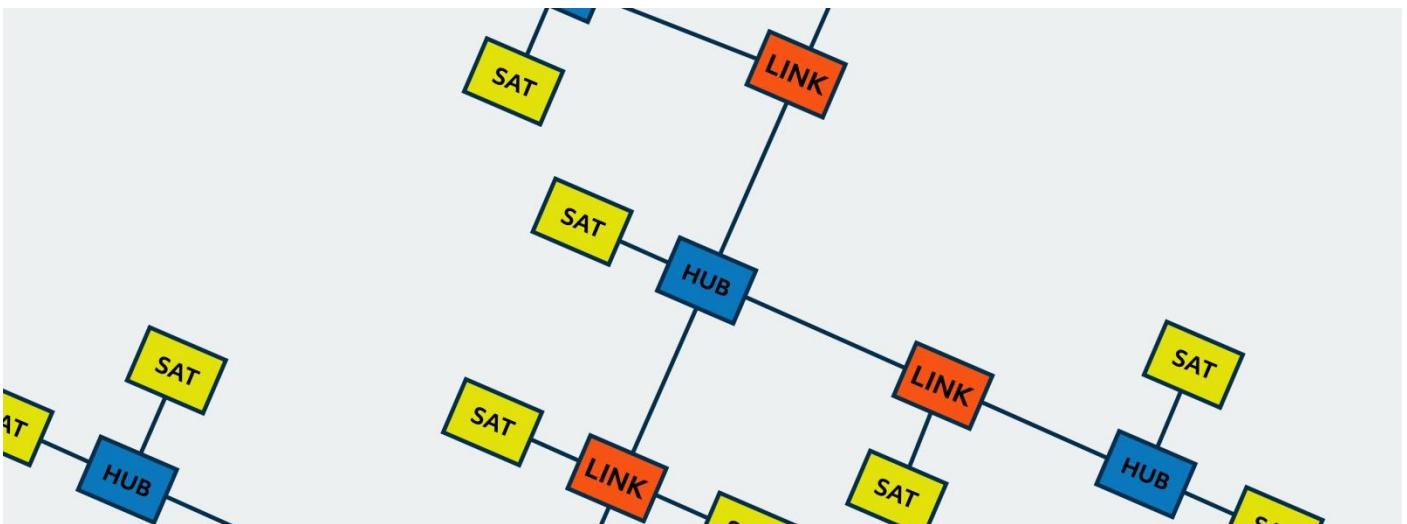


Data Warehouse mit Data Vault 2.0

Implementierung mit dem UDG von PPI – Generierung Ihrer DWH Komponenten



Kennen Sie diese Probleme?

- Ihr DWH hat vielfältige Abhängigkeiten zwischen den verschiedenen Schichten, Liefersystemen, Abnehmern und der implementierten Logik
- „Kleine“ neue Anforderungen wirken sich kaskadierend auf viele Systeme aus
- Es entstehen hohe Analyse- und Testaufwände
- Die Kosten und Umsetzungszeiten steigen
- Sie kämpfen zusätzlich mit Performanceproblemen

Sowohl die fachliche Analyse, die Entwicklung und der Test, als auch die Beladung der einzelnen Komponenten werden dadurch bewusst entkoppelt. Insbesondere werden kaskadierende Abhängigkeiten vermieden.

Data Vault 2.0 ist damit die ideale Umsetzung eines agilen Vorgehensmodells. Neue Anforderungen werden schneller und kostengünstiger produktiv gesetzt.

Gleichzeitig wird die DWH Beladung massiv parallelisiert und die Performance signifikant erhöht.

DataVault 2.0

Das Vorgehen nach Data Vault 2.0 adressiert genau diese Probleme.

Unter Data Vault 2.0 wird ein stringentes Datenmodell verwendet. Dieses Modell besteht aus:

- (blauen) Hubs, die Ihre Geschäftsobjekte abbilden
- (roten) Links, die Beziehungen zwischen Geschäftsobjekten darstellen
- (gelben) Satelliten, die Eigenschaften von Hubs und Links enthalten

Durch das standardisierte Datenmodell bietet Data Vault 2.0 die Möglichkeit, DWH Komponenten zu generieren.

Data Vault verfolgt das Grundprinzip, komplexe Aufgabenstellungen in möglichst kleine und sehr einfache Schritte zu zerlegen.

Diese große Menge von einfachen „Problemen“ erfordert den Einsatz von automatisierten Vorgehensweisen.

Ein sinnvoller Einsatz von Data Vault und effiziente Generatoren müssen daher Hand in Hand gehen.

Universal Datamodell Generator (UDG)

Data Warehouse mit Data Vault 2.0

Implementierung mit dem UDG von PPI – Generierung Ihrer DWH Komponenten

Die Lösung: UDG

PPI hat daher den Universal Datamodel Generator (UDG) entwickelt, der Data Vault 2.0 extrem effizient umsetzt.

UDG ist ein Eclipse Plugin, deren Funktionalität über eine offene API (Application Programming Interface) genau an Ihre Projektbedürfnisse angepasst ist. UDG kann daher immer genau das, was Sie in Ihrem Projekt benötigen.

UDG erfordert wenig technisches Know-How. Technische Fehlbedienungen sind durch implementierte Validierungen nicht möglich. Die Datenmodellierung kann daher durch Mitarbeiter aus den Fachbereichen erfolgen.

UDG generiert automatisch:

- SQL-Skripte zur Erstellung und Migration des Datenmodells
- Views für den Zugriff auf das Data Vault Modell. Das Data Vault Modell wird logisch, wie gewohnt als relationales Modell präsentiert.
- PL/SQL-Packages und Steuerungsskripte für eine parallele Beladung der Oracle-DB
- Dokumentation des Datenmodells in Excel, Word sowie in grafischen Übersichten
- technische Komponenten für die Implementierung von Geschäftslogiken
- weitere projektspezifische Komponenten

UDG ermöglicht:

- die Konzentration auf die Erstellung des Datenmodells
- die Implementierung der fachlichen Logik.

Den gesamten „technischen Rahmen“ erstellt der Generator.

Unser Angebot umfasst:

- die Analyse, ob und welche Data Vault Modellierung für Ihr Data Warehouse geeignet ist
- die kostenlose Bereitstellung des UDG in einer voll funktionsfähigen Variante als Open Source-Komponente
- die Anpassung des UDG an Ihre konkreten Projektanforderungen
- die Migration Ihres bestehenden Data Warehouse auf Data Vault 2.0

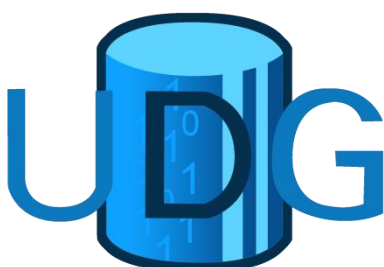
Sie erhalten:

- eine auf Ihr Projekt zugeschnittene Data Vault 2.0 – Architektur
- einen Generator zur effizienten Umsetzung dieser Architektur.

derzeitige Systemvoraussetzungen

- Eclipse
- Java
- Oracle
- Unix Server

Falls diese Voraussetzungen nicht erfüllt sind können projektspezifische Anpassungen erfolgen (z.B. Einsatz DB2, MS SQL, Windows Server u.a.)



Fragen und weitere Informationen:



Jörg Stahnke
Principal Technology
Consultant Datenbanken
T +49 40 227433-1329
M +49 1603841771
Joerg.Stahnke@ppi.de

PPI AG
Moorfuhrweg 13
22301 Hamburg
Germany
www.ppi.de