

Die StarData GmbH stellt sich vor...

Die StarData GmbH ist seit 12 Jahren erfolgreich am IT-Markt vertreten und auf die Beratung sowie Durchführung von IT-Projekten spezialisiert.

Zentrale Aufgaben sind die Entwicklung von verteilten Systemen (internetfähig und objektorientiert), Prozess- und Datenmanagement sowie Code-Generierung.

Die Qualität unserer Arbeit hat immer die höchste Priorität mit dem Ziel unseren Kunden Zeit- und Kostenreduzierung zu ermöglichen. Es wird stets auf den Einsatz modernster Technologien geachtet, die mit eigenen Produktentwicklungen kombiniert werden.

Als Erfinder und Entwickler sind wir am **TLGen**-Projekt beteiligt und besitzen daher die Möglichkeit Ihr Projekt mit diesem neuen Produkt der Persistence Layer Erstellung zu vervollständigen.

www.stardata.de

Eine Auswahl unserer Kunden...



Adalbert-Stifter-Str. 9
82110 Germering

Telefon 089/244 007 42
Telefax 089/244 007 39

info@stardata.de
www.stardata.de

Testen Sie uns...

Sind Sie neugierig geworden?

Dann testen Sie uns:

Lassen Sie uns aus einem laufenden oder abgeschlossenen Projekt mit Ihrem Domain Model den Code erstellen.

Lassen Sie sich diese neue Möglichkeit in einer Live-Präsentation vorführen, Sie werden eine neue Art der Persistence Layer Erstellung erfahren.

TLGEN

www.tlgen.de



Persistence
Layer
Generator **TLGEN**

Die StarData GmbH unterstützt Ihr IT-Entwicklungsvorhaben in Kombination mit neuesten Technologien und eigenen Produktentwicklungen.

In Verbindung mit dem Generator **TLGen** bieten wir eine neue Form der Persistence Layer Erstellung an, ohne Framework, durch ein Domain Model gesteuert. **TLGen** generiert die Persistence Layer nach Ihren Wünschen.

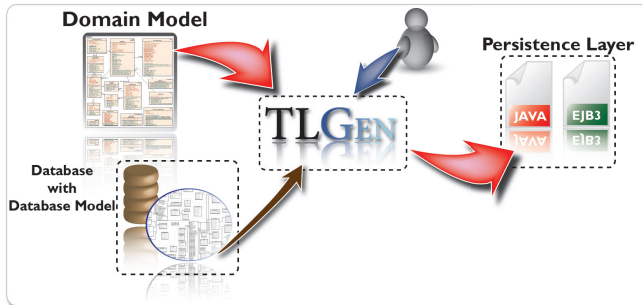


Das Produkt TLGen...

TLGen ist ein Generator, der die Vorteile des MDA (Model Driven Architecture) sowie CASE (Computer Aided Softwareengineering) Ansatzes verbindet.

Mit **TLGen** ist es möglich den JAVA Code (EJB 3-Standard) der Persistence Layer automatisch aus einem Domain oder Database Model zu generieren.

In Verbindung mit einem Application Server Ihrer Wahl ermöglicht der Code eine noch nie dagewesene Performance in der Persistence Layer.



Die Vorteile:

- Trennung von Funktionalität und Technik
- Ausgangspunkt: Eindeutiges Domain Model
- Projektsteuerung über Domain Model
- Logik steht im Vordergrund
- Kein überflüssiger Code mehr
- Code-Generierung nach neuestem J2EE-Standard (EJB 3) - kein Framework-Zwang
- Keine Beschränkung bei Bedarf von spezialisierten Frameworks (z.B. Hibernate)
- Reduzierung der Projektzeit
- Abschaffung von „wild hacking“ im Code
- Kein Auseinanderdriften mehr von Modell und entwickeltem Code
- Sehr hohe Flexibilität im Projektablauf
- Minimierung des Teamaufwands
- Refactoring von Legacy-Systemen (Alt-Systeme)

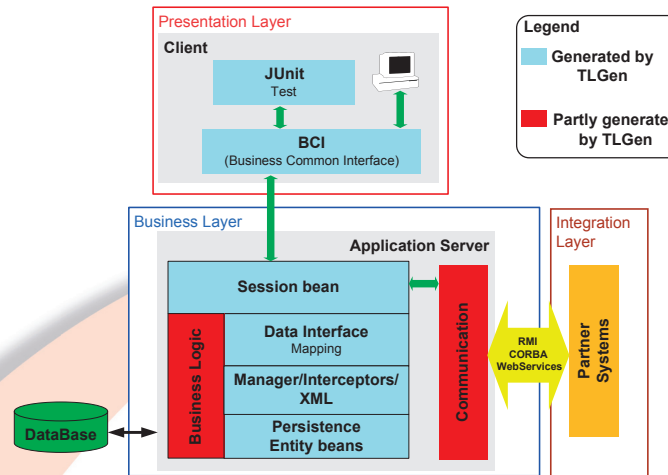
Unser Angebot...

Wir bieten Ihnen in Verbindung mit dem Produkt **TLGen** an, Ihre Projekte für die Persistence Layer Entwicklung zu unterstützen, angefangen mit der Architektur des Domain Models bis hin zum vollendeten Code.

Sie können die Entwicklung vollständig zu uns outsourcen oder eine nach Ihren Wünschen gegliederte Zusammenarbeit mit uns eingehen.

Unsere Leistungen im Überblick:

- Komplette Persistence Layer Erstellung
- Domain Model Architektur
- Teilprojektdurchführung
- Refactoring der Legacy-Systeme
- Projektbegleitende Beratung
- **TLGen**-Einarbeitung



Projektablauf mit TLGen...

Projektdurchführungsaufwand minimieren

In IT-Projektablaufen werden heutzutage die Teilprozesse „Design-Architecture“, „Program Coding“, „Database Generation“ sowie „Erstellung von Test-Klassen“ nacheinander erstellt. Durch Nutzung von **TLGen** lassen sich diese Teilschritte in einem einzigen Prozess generieren.

Logik-Optimierung

Beim Auftreten von logischen Fehlern muss nicht mehr jeder einzelne Schritt angepasst werden, um die Fehler in aufwendiger Teamarbeit zu korrigieren. Allein die Anpassung des Domain Models reicht aus, damit **TLGen** den Code der einzelnen Schritte automatisch neu generiert.

Projektverlauf in einem Guss

Aufwendige Teamgrößen und damit verbundene Koordination fällt weg; die Code-Erstellung muss nicht mehr aufgeteilt werden, was einen durchgängigen Entwicklungsfluss ermöglicht.

