Get Back in Control of your SQL

™ Why your developers need jOOQ.

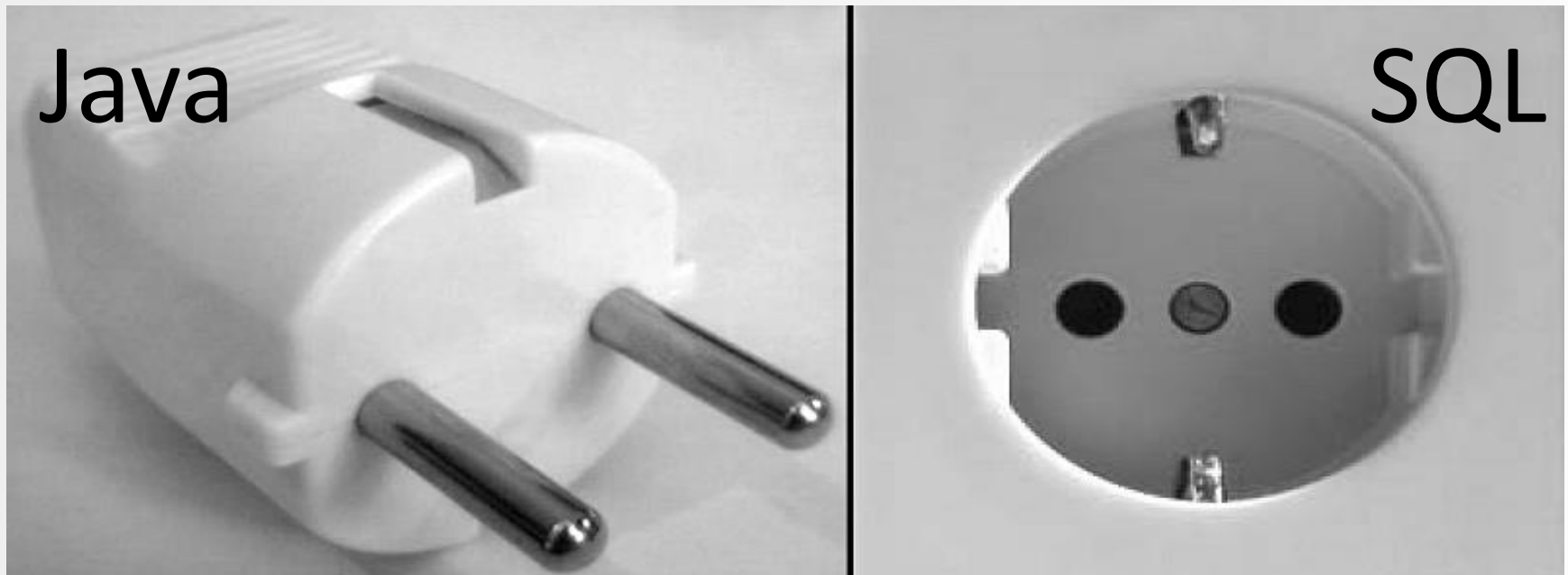SQL and Java could work together so much better if we only let them.

jOOQ is SQL written in Java

- SQL is from 1974: Mature and robust
- Java is from 1995: Mature and robust

- Up to date, it is hard to write SQL in Java

(«hard» means inefficient, error-prone. developers focus on infrastructure instead of business-logic)
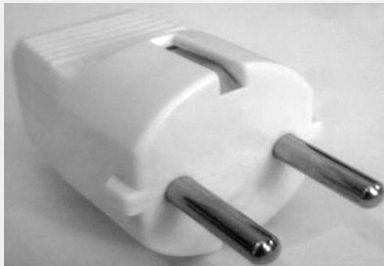
SQL and Java – in theory

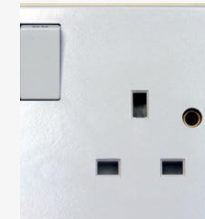Java   SQL

In this metaphor, electricity is the data (SQL) that flows into your appliance / application (Java)

SQL and Java – in practice – a bad fit

Java

SQL

Images from: http://en.wikipedia.org/wiki/AC_power_plugs_and_sockets. License: public domain

What JDBC means for your developers

With JDBC, your developers have to do a lot of manual, error-prone (dangerous) and inefficient work
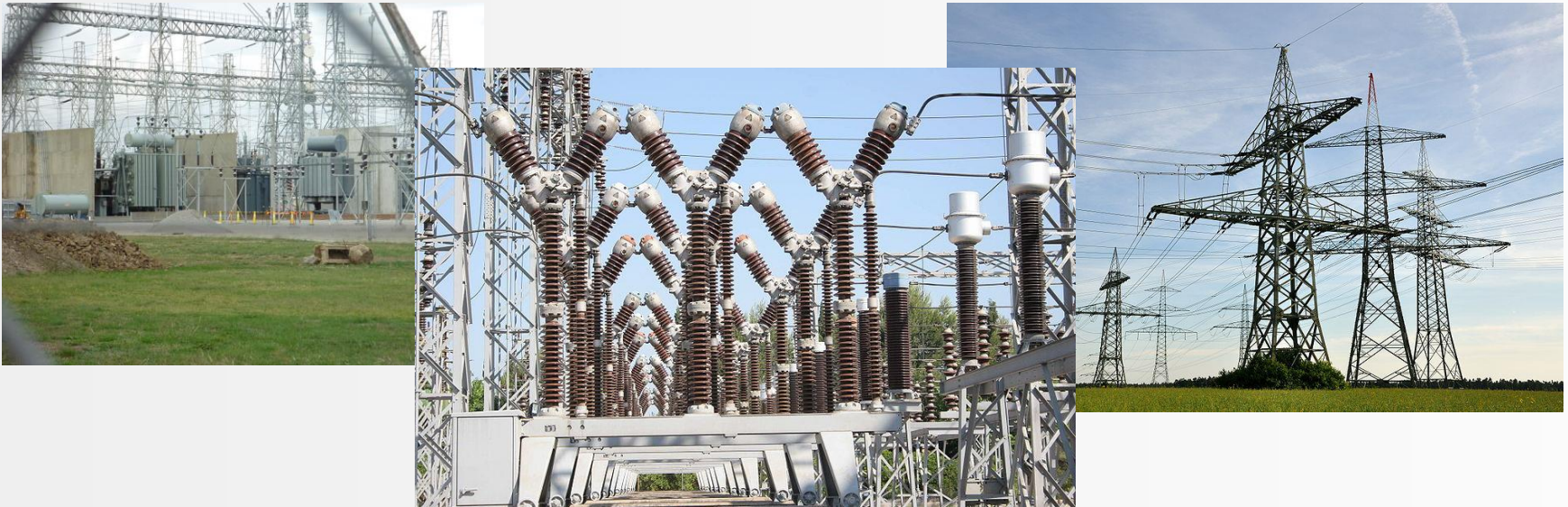
How JDBC compares with jOOQ

JDBC	jOOQ
Time consuming	Time saving
Manual work	Automated work (code generation)
Error prone (no Java typesafety)	Error safe (Java typesafety)
Errors detected at run time (production)	Errors detected at compile time (development)
Errors cost XX,XXX.XX € or more	Errors cost XXX.XX € or less
Databases are all different	jOOQ handles database differences
Very basic SQL	Highly extensible SQL

With JDBC, your developers will build something like jOOQ themselves instead of writing business-logic

What JPA means for your developers

With JPA, your developers use a huge framework with lots of complexity that can get hard to manage

Images from Wikimedia. License: public domain. High voltage power lines by Simon Koopmann. License: CC-BY SA 3.0

How JPA compares with jOOQ

JPA	jOOQ
Time consuming	Time saving
Complex	Simple
Performance is difficult	Performance is easy
Writing SQL is hard	Writing SQL is easy
You need a DBA *and* a JPA expert	You only need a DBA
JPA defines your whole architecture	jOOQ works with any architecture
JPA costs much in development efforts	jOOQ costs little in development efforts and little in licensing

With JPA, your developers will spend a lot of time tuning JPA and a lot of time circumventing JPA
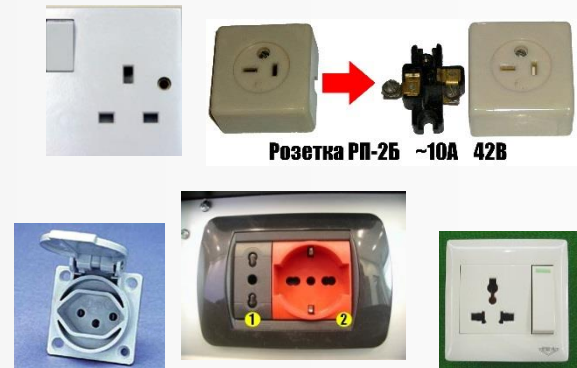
What jOOQ means for your developers

Java   jOOQ   SQL

With jOOQ, Java plugs into SQL intuitively, letting your developers focus on business-logic again.

Your ROI with jOOQ

REDUCE YOUR OVERALL PRODUCT COSTS

▬ Money spent on jOOQ licenses   ▬ Money spent on development

Without jOOQ   With jOOQ

Why use jOOQ

- Your developers save time

- Your developers increase quality

- Your customers get better software

- And: You save money

Testimonials

I love your licensing! It let us truly try you out and then when we were satisfied (security, functionality, etc) that things were good and that this would help us with some of our less SQL-knowledged developers who kept giving us bad SQL, SQL injection, etc, that would have to be fixed prior to release, we bought it and began telling people throughout our company about it so that others might buy it for their projects. The ability to trivially add SQL functions (like the Oracle-specific substring call, concat, and to_number) in a single line of a single method is fantastic! Honestly, we've gotten away using your documentation, examples, and source and never had to bug you guys with "How do I ...?" You blog, manual, etc, all combine together with a simple library (simple for us) to make a great product that let's us do some insane SQL ***really really easily***.

Daniel Owens - DanielSecurities

Testimonials

Using jOOQ has been a joy, it's made life much easier for us.

Ilkka Halila - Boomlagoon Ltd.

The level of support from the jOOQ team was nothing short of excellent. They were extremely easy to talk to, and had my problem solved in less than an hour on Skype. I would recommend jOOQ support services to anyone.

Paul Woodland - ABC Information Solutions Pty Ltd

Thanks to jOOQ we are in control of our SQL which helps to be in control of our project.

Marco Dubacher - Ergon Informatik AG