07. – 10.12.2020

#ittage

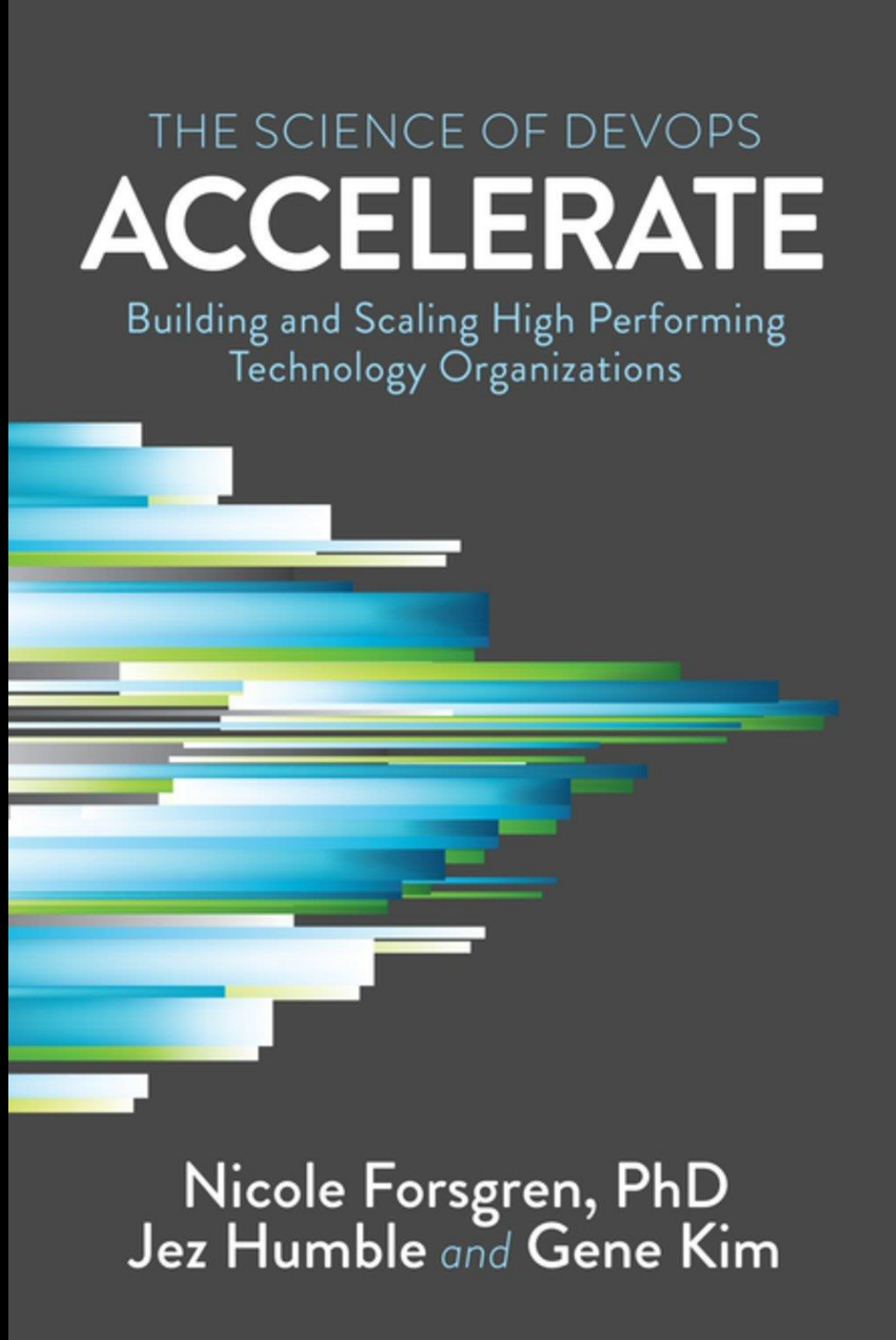
IT-Tage 2020

REMOTE KONFERENZ

Samuel Nitsche

Continuous Improvement auf der Todesstern-Datenbank

Software Delivery Performance



Samuel Nitsche

- Senior Software Developer @ Smenso
- Blogger und Star Wars Fan
- Seit 2017 utPLSQL Maintainer
- Seit 2019 Oracle ACE
- Unhöflich gegenüber Nazis + Faschisten

@Der_Pesse

pesse

@pesse

Samuel_Nitsche

smenso Cloud

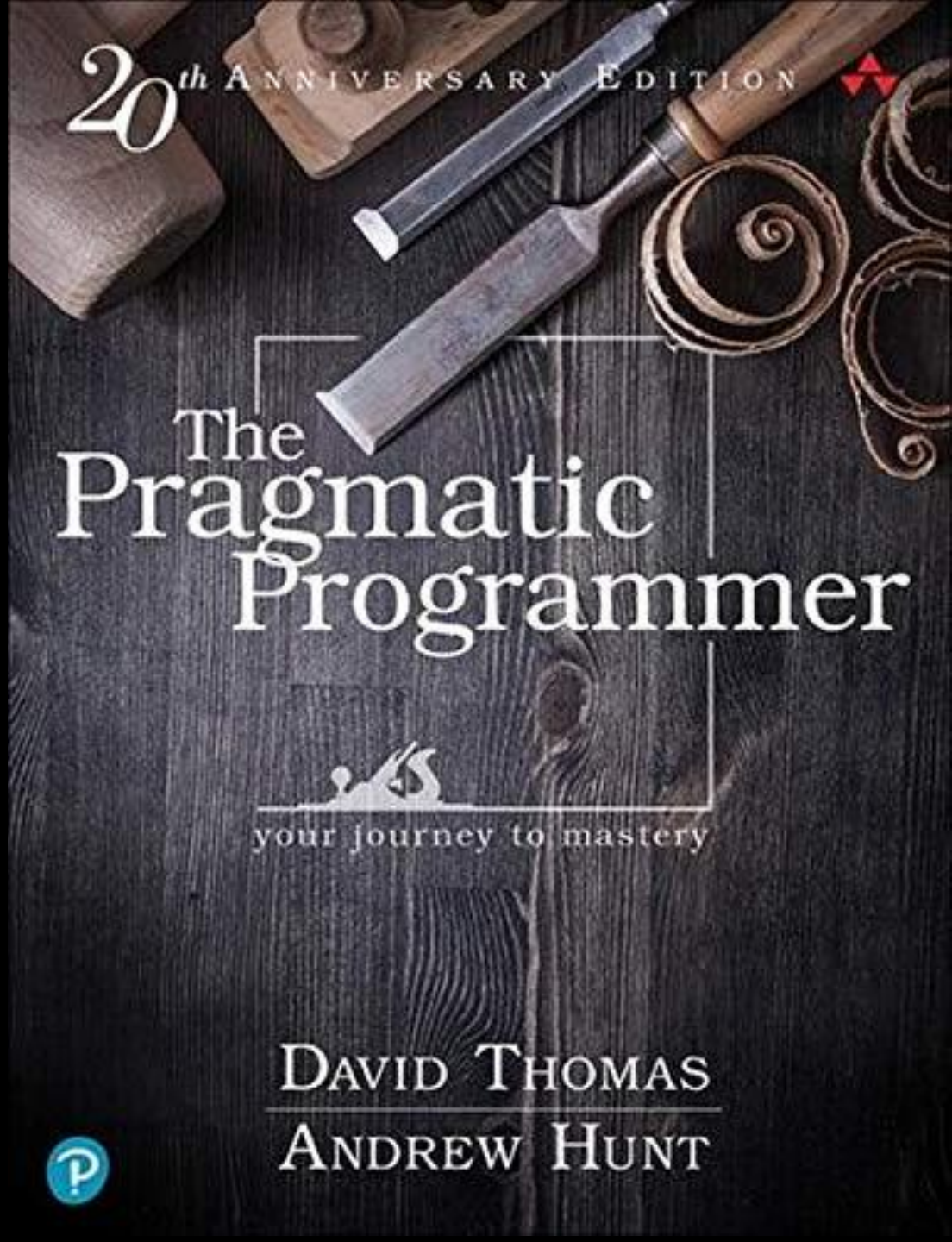
Alle Beispiele funktionieren auf einer Oracle 18 XE Datenbank.

https://github.com/pesse/deathstar-continuous-improvement

Die zugrunde liegenden Konzepte und Methoden können jedoch für jede andere relationale Datenbank angewendet werden.

Das Pragmatische Starter Kit

- Versionskontrolle
- Regression Testing
- Komplette Automatisierung



Versionskontrolle

Versionskontrolle – Datenbank-Artefakte

Schema	Daten	Tooling
• Tabellen, Constraints • View Definitionen • Stored procedures • Trigger • Synonyme • Indizes	• Metadaten • Referenz-/Testdaten • Projektdaten	• Datenbank Setup • Scripts zum Laden/Extrahieren/ Migrieren von Daten • Scripts zur Erstellung von Testdaten • Test-Scripts

Ideal: Selbes Repository wie die Applikation

Wie bekomme ich die Artefakte in Versionskontrolle?

Nutzung von Software, die die Datenbank kontrolliert

Einführung von dateibasierter DB-Entwicklung

Aber – wie bekomme ich sie INITIAL?

- IDE Export (SQL Developer, SSMS)
- DB-interne Tools (DBMS_METADATA)
- Open Source Tools (z.B. PLEX)
- Visual Studio SSDT

Datenbank Testen

utPLSQL
TESTING FRAMEWORK FOR PL/SQL

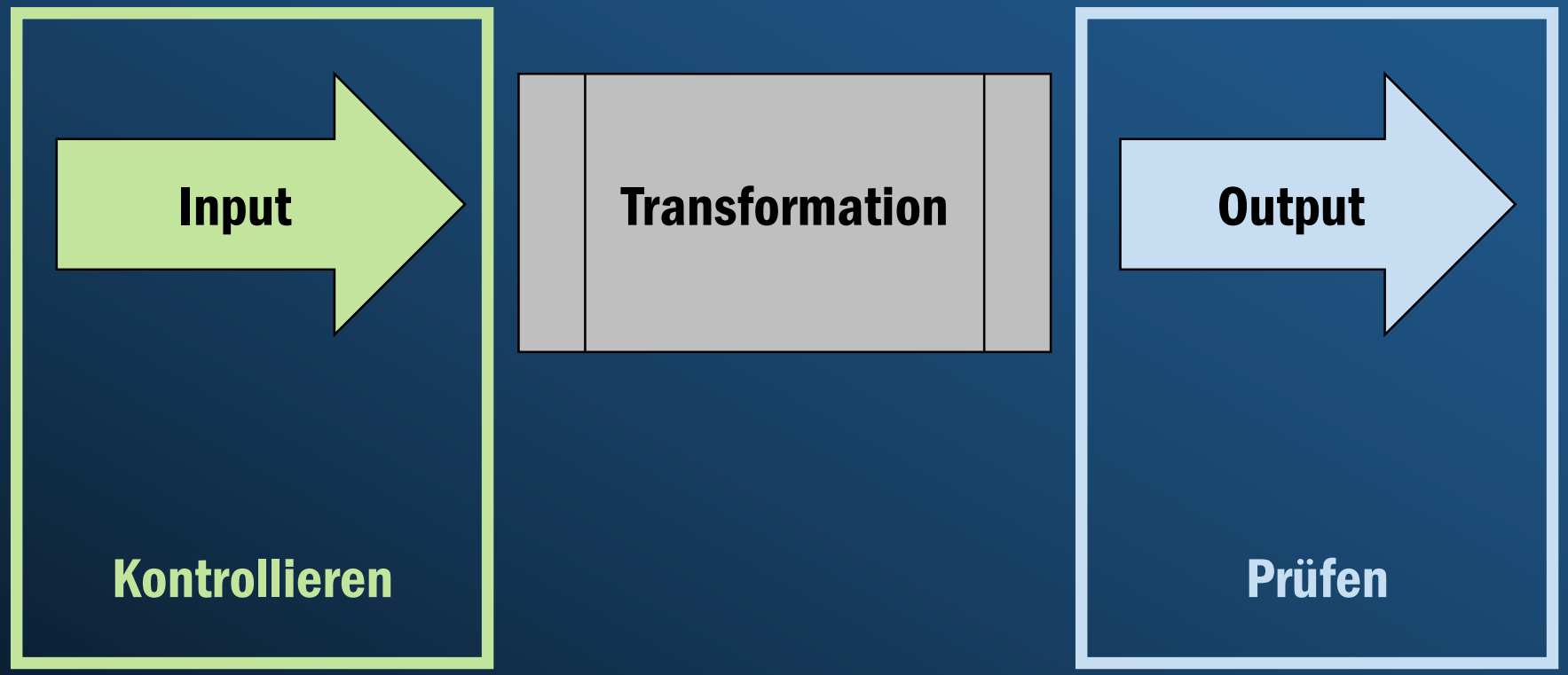
tSQLt
Test Driven Design for SQL Server

dbUnit

DBFit

JUnit 5

Was ist ein Test?



Parameter
Tabelleninhalte
Cursors

Function-Rückgaben
Tabellenwerte
Vorhandensein von Zeilen

Test von Insert und Select mit utPLSQL

```
procedure insert_select_sith_persons as
    l_row sith_persons%rowtype;
begin
    insert into sith_persons (name, alive) values ('Darth Sam', 0);

    select * into l_row from sith_persons
        where name = 'Darth Sam';
    ut.expect(l_row.id).to_be_greater_than(0);
    ut.expect(l_row.alive).to_equal(0);
end;
```

Test von Insert und Select mit utPLSQL

```
procedure insert_select_sith_persons as
   l_row sith_persons%rowtype;
```

Heute 17:00 Uhr – Track 2

Oracle Database – utPLSQL: Unit Testing für PL/SQL in der Praxis

Sabine Heimsath

```
   ut.expect(l_row.alive).to_equal(0);
end;
```

Wir brauchen eine Sandbox

Photo by Daniel Cheung on Unsplash

Lokaler SQL Server 2019 in Docker

```
> docker pull mcr.microsoft.com/mssql/server:2019-CU3-ubuntu-18.04

> docker run -e "ACCEPT_EULA=Y" -e "SA_PASSWORD=Strong@Passw0rd" \
  -p 1433:1433 --name sql1 \
  -d mcr.microsoft.com/mssql/server:2019-CU3-ubuntu-18.04
```

Done

Lokale Postgres in Docker

```
> docker pull postgres

> docker run -e "POSTGRES_PASSWORD=docker" \
 -p 5432:5432 --name postgres1 \
 -d postgres
```

Done

Lokale Oracle 18 XE in Docker

```
> Download git repo https://github.com/oracle/docker-images/blob/master/OracleDatabase/SingleInstance

> ./buildDockerImage.sh –v 18.4.0 –x

Sehr lange Zeit warten

> docker run --name orcl18xe -p 1521:1521 -p 5500:5500 \
  -e ORACLE_PWD=oracle oracle/database:18.4.0-xe

Sehr lange Zeit warten
```

Aber wenigstens ist es möglich

Benutzung von

Docker Volumes

zum Erstellen von

Snapshots

Export und Import von Docker Volumes

Docker Volume -> Archive

```
> docker run --rm -v my_volume:/u02 -v "C:\localFolder":/backup \
  ubuntu tar czvf /backup/backup.tar.gz /u02
```

Archive -> Docker Volume

```
> docker run --rm -v my_volume:/u02 -v "C:\localFolder":/backup \
  ubuntu tar xvpfz /backup/backup.tar.gz -C /
```

Continuous Improvement: Ändern von Sith-Namen

SITH_CHARACTERS_T
ID
NAME
ALIVE

SITH_CHARACTERS_T
ID
ALIVE
PRENAME
SURNAME

SITH_PERSONS
ID
NAME
ALIVE

Schritt 1 (V2.2): Neue Spalten in der Tabelle

```
alter table sith_characters_t
  add prename varchar2(100)
  add surname varchar2(100);
```

Schritt 1.1 (V2.2): Werte aktualisieren

```
update sith_characters_t set
  prename = trim(substr(name, 1, instr(name, ' '))),
  surname = trim(substr(name, instr(name, ' ')));
```

Schritt 1.2 (V2.2): Kontinuierliche Aktualisierung

```
create or replace trigger sith_persons_trg_save
  instead of insert or update or delete on sith_persons
  begin
    if inserting then
      insert into sith_characters_t (name, alive, prename, surname)
        values (
          :new.name,
          :new.alive,
          trim(substr(:new.name, 1, instr(:new.name, ' '))),
          trim(substr(:new.name, instr(:new.name, ' ')))
        );
    ...
```

Schritt 1.2 (V2.2): Kontinuierliche Aktualisierung

```
  ...
  elsif updating then
    update sith_characters_t set
      name = :new.name,
      alive = :new.alive,
      prename = trim(substr(:new.name, 1, instr(:new.name, ' '))),
      surname = trim(substr(:new.name, instr(:new.name, ' ')))
    where id = :new.id;
   end if;
 end;
/
```

Schritt 2 (V2.3): Constraints

```
alter table sith_characters_t
  modify prename not null
  modify surname not null;
```

```
alter table sith_characters_t
  add constraint sith_characters_uq_pre_surname
    unique (prename, surname);
```

Schritt 3 (V2.4): View anpassen

```
create or replace view sith_persons as
  select
    id,
    prename || ' ' || surname as name,
    alive
    from sith_characters_t;
```

Schritt 3.1 (V2.4): Trigger anpassen

```
create or replace trigger sith_persons_trg_save
  instead of insert or update or delete on sith_persons begin
    if inserting then
      insert into sith_characters_t (alive, prename, surname)
        values (
          :new.alive,
          trim(substr(:new.name, 1, instr(:new.name, ' '))),
          trim(substr(:new.name, instr(:new.name, ' ')))
        );
      ...
  end;
/
```

Schritt 4 (V2.5): Nicht mehr benötigte Spalte löschen

```
alter table sith_characters_t
  drop column name;
```

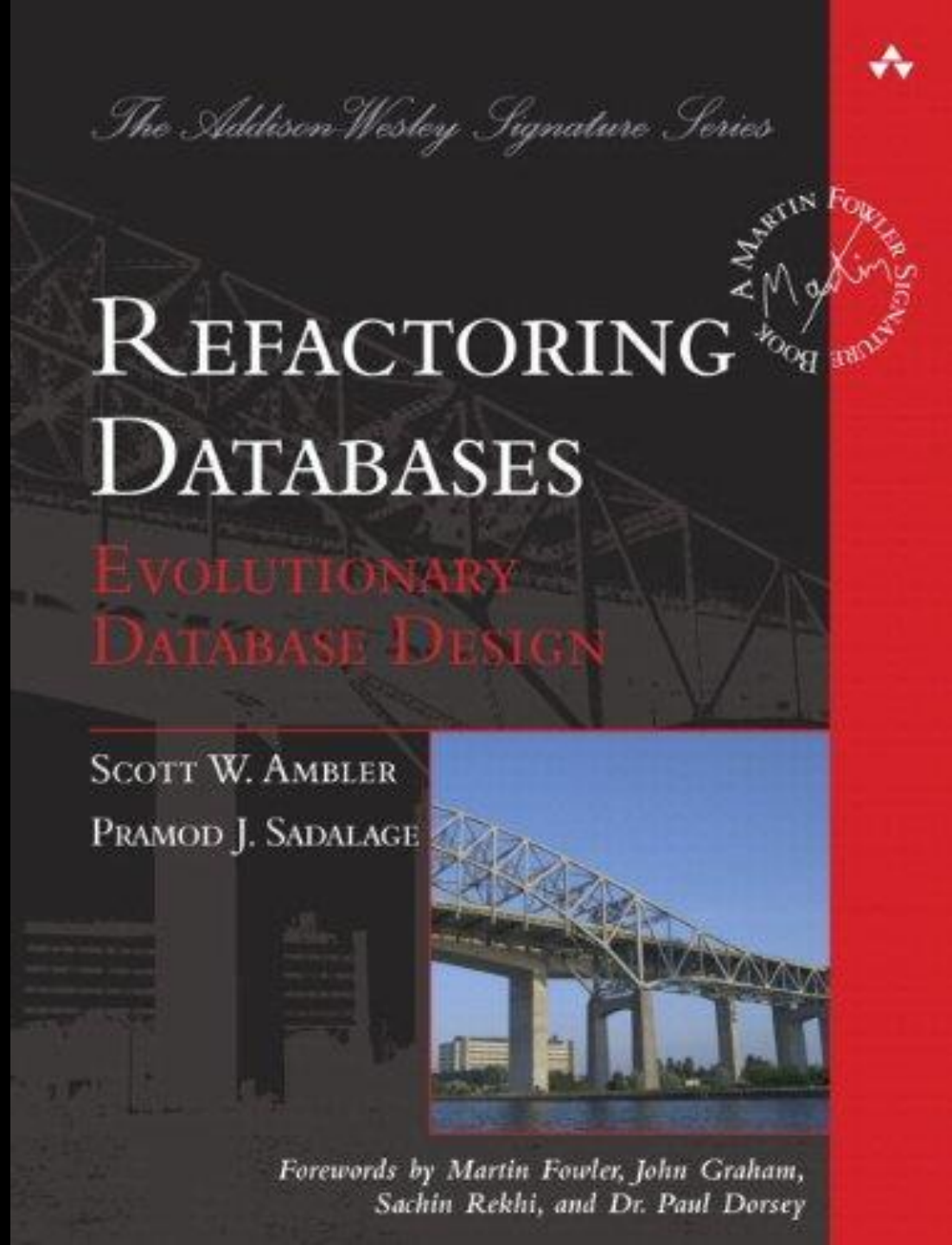
Let’s get serious

https://github.com/pesse/deathstar-continuous-improvement

Photo by Hello I'm Nik on Unsplash

„The first impediment,
and the hardest one to overcome,
is cultural.“

Scott Ambler & Pramod Sadalage, 2006



Gemeinsam Voran!

...moderne RDBMS haben viel zu bieten!

Photo by Daniel Cheung on Unsplash

„Ich wusste gar nicht, dass so etwas mit SQL geht!“

Advent Of Code 2020 – in SQL

cleandatabase.wordpress.com/tag/adventofcode2020