

# Elefant unter Strom

von OldSql  
über NoSQL  
zu NewSQL?



# Motivation

**Moore's Law is Broken**

processing can't scale-up rather out

**Data is Growing**



**Relationale Datenbanken  
sind groß, alt, schwer und  
langsam wie ein Elefant.**

**Michael  
Stonebreaker**  
Richtig gut können die  
eigentlich gar nichts.

**Michael Stonebreaker**



# Relationale Datenbanken

## Features

**ACID**

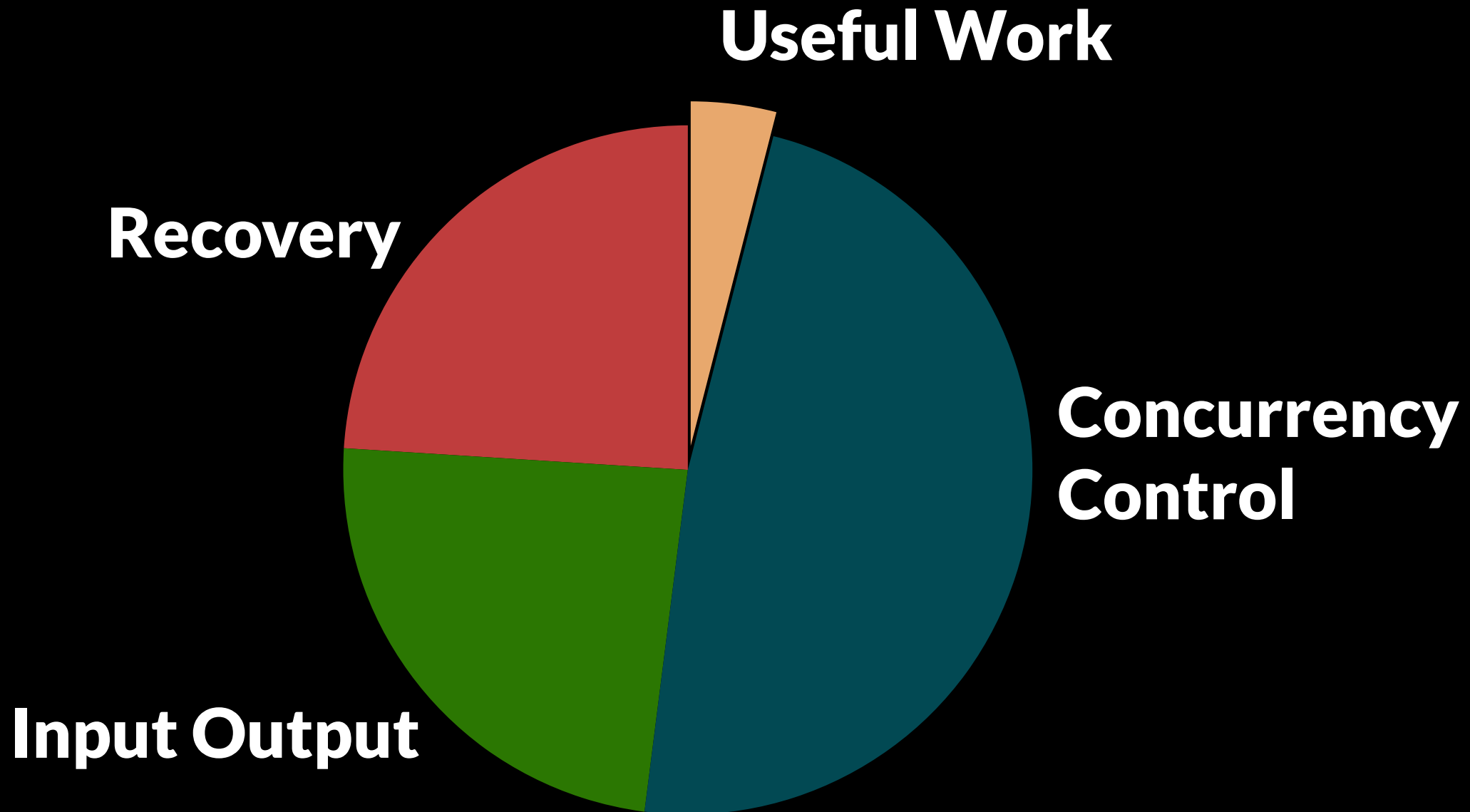
**SQL**

**Scale-up**

**Fixed Schema**

# Relationale Datenbanken

## Performance





# NoSQL to the rescue?



# NoSQL Datenbanken

## Features

**BASE**

**low-level Query Language**

**Scale-out**

**Flexible Schema**



**B**asically  
**A**vailable

**S**oft state

**E**ventually consistent





# Why ACID?

**It's better to have programmers deal with performance problems of transactions, than coding around the lack of transactions.**

**Google Spanner**



# Why SQL?

**no SQL no Standard  
high-level Language**



NoSQL

over

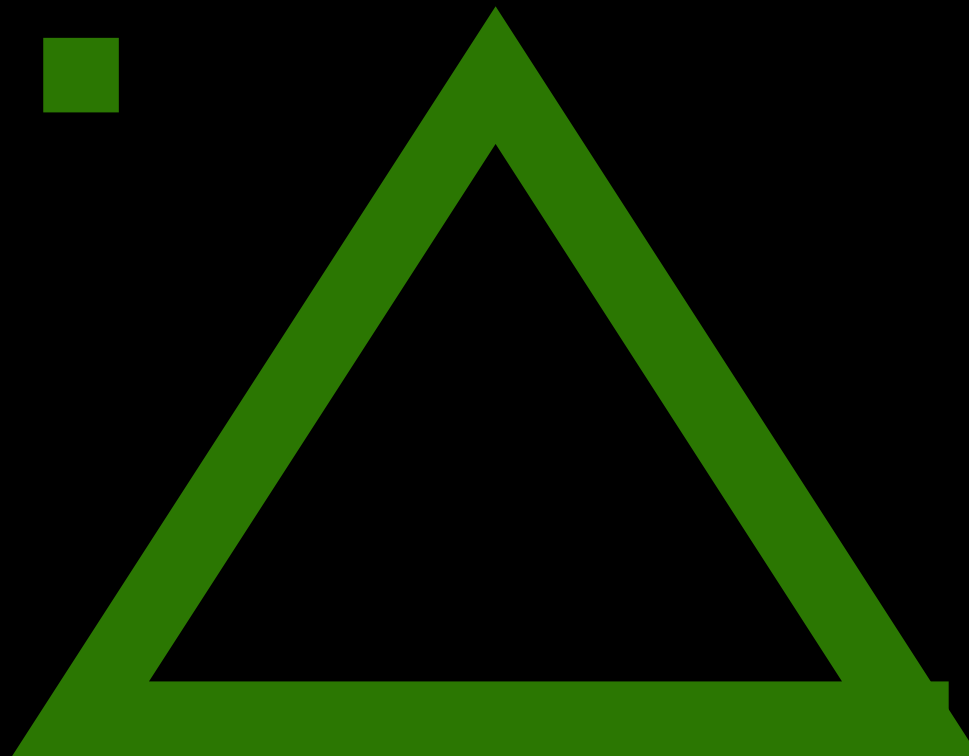
OldSQL ?

# Database World

Stonebreaker



others



Datawarehouse

OLTP



# Introducing NewSQL

**SQL**

**ACID**

**Scale-out**

**Fixed Schema**



# NewSQL = new Architecture

Single Thread, MVCC

Main Memory

Redundance

over

Locking

Disk

Recovery



# NewSql Players





# Introducing VoltDB





# **VoltDB** Architecture

**Main Memory**

**Shared Nothing**

**ACID**

**HA and Recovery**



# Getting Started

## Create Database

```
$ voltdb create -d deployment.xml
```

## Load Database Definition

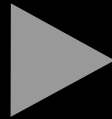
```
$ sqlcmd  
1> load classes procedures.jar;  
2> file schema.sql
```

## SQL and Stored Procedures



# Schema Changes

**DDL Statements**



**Live Schema Update**

**Unique Index or  
Remove Column**



**Save and Restore**



# SQL Stored Procedure

```
CREATE PROCEDURE lastX AS  
    SELECT TOP ? * FROM ACTIVITY  
    ORDER BY end ASC
```



# Java Stored Procedure

```
LastXActivities extends VoltProcedure {  
  
    VoltTable[] run(int x) {  
        voltQueueSQL(  
            "SELECT TOP ? * FROM ACTIVITY  
            ORDER BY act_end ASC",  
            x  
        );  
        return voltExecuteSQL();  
    }  
}
```



# Async Stored Procedure

```
client.callProcedure(  
    new Callback() { ... },  
    "AddActivity",  
    "Franz",  
    "Ferkel",  
    "uuid-3"  
)
```



# Transaktionen

**ein SQL Statement**

**eine Stored Procedure**



# Partitionen

## replicated vs. partitioned Tables

```
PARTITION TABLE activity ON COLUMN project_id
```

## partitioned Stored Procedures

```
PARTITION PROCEDURE lastX
```

```
ON TABLE activity COLUMN project_id
```





# **Durability** and **Recovery**

**Snapshots**

**K-Safety**

**Replication**

**Command Logging**



# VoltDB

## Demo

[DB Monitor](#)[Admin](#)[Schema](#)[SQL Query](#)[Help](#)[Cluster](#) > ↑ Active (1) ↓ Missing (0)Monitoring > [dev-jan](#) << [Server](#)[Show/Hide Graph](#)View Seconds[Display Preference](#)[Show/Hide Data](#)

### Stored Procedures (Cluster)

[Prev](#) [Page 1 of 1](#) [Next](#)

Stored Procedure	Invocations	Min Latency (ms)	Max Latency (ms)	Avg Latency (ms)	% Time of Execution
org.voltdb.sysprocs.SnapshotSave	1	70.73	70.73	70.73	57
org.voltdb.sysprocs.AdHoc_RW_MP	2	25.49	25.49	25.49	41
org.voltdb.sysprocs.AdHoc_RO_SP	1	3.25	3.25	3.25	3



### Database Tables

[Prev](#) [Page 1 of 1](#) [Next](#)

Table	Row Count	Max Rows (per partition)	Min Rows (per partition)	Avg Rows (per partition)	Type
ACTIVITY	0	0	0	0	REPLICATED
PROJECT	2	2	2	2	REPLICATED

[DB Monitor](#)[Admin](#)[Schema](#)[SQL Query](#)[Help](#)[Overview](#)[Schema](#)[Procedures & SQL](#)[Size Worksheet](#)[DDL Source](#)[Refresh](#)

## Schema

☐ Expand All

Name	Type	Partitioning	Columns	Indexes	PKey	TupleLimit
▼ <a href="#">ACTIVITY</a>	Table	Replicated	5	1 (1 unused)	Has-PKey	No-limit

```
CREATE TABLE ACTIVITY (  
  ID varchar(36) NOT NULL,  
  DESCRIPTION varchar(4000),  
  ACT_START timestamp,  
  ACT_END timestamp,  
  PROJECT_ID integer NOT NULL,  
  UNIQUE (ID),  
  PRIMARY KEY (ID)  
);
```

Read-only by procedures: LastXActivitiesJava, lastXactivities

Index Name	Type	Columns	Attributes
▶ <a href="#">VOLTDB_AUTOGEN_IDX_PK_ACTIVITY_ID</a>	BALANCED_TREE	ID	Unique Unused

▶ <a href="#">PROJECT</a>	Table	Replicated	4	1 (1 unused)	Has-PKey	No-limit
---------------------------	-------	------------	---	--------------	----------	----------

[DB Monitor](#)[Admin](#)[Schema](#)[SQL Query](#)[Help](#)[Refresh](#)

Query

[Run](#)[Clear](#)[Tables](#)[Views](#)[Stored Procedures](#)

▼ ACTIVITY

ID (varchar)

DESCRIPTION (varchar)

ACT\_START (timestamp)

ACT\_END (timestamp)

PROJECT\_ID (integer)

```
select * from project order by id desc
```



Query Result

HTML



ID	TITLE	DESCRIPTION	ACTIVE
uuid-2	LEO	null	null
uuid-1	Humboldt	null	null

[DB Monitor](#)[Admin](#)[Schema](#)[SQL Query](#)[Help](#)[Cluster](#)[Pause](#)[Promote](#)[Save](#)[Restore](#)[Shutdown](#)[Download Configuration](#)[Server !\[\]\(111c5272ee3f91361f0d2e3665dd6ad0\_img.jpg\)](#)

## Overview

Sites Per Host	8
----------------	---

K-Safety	0
----------	---

Partition Detection	<input checked="" type="checkbox"/> On
---------------------	--

▶ Security	<input type="checkbox"/> Off 
------------	--

▶ HTTP Access	<input checked="" type="checkbox"/> On
---------------	--

▶ Auto Snapshots	<input type="checkbox"/> Off 
------------------	--

▶ Command Logging	<input checked="" type="checkbox"/> On
-------------------	--

▶ Export	<a href="#">+</a>
----------	-------------------

▶ Advanced
------------

## Network Interfaces

Port Name	Server Settings	Cluster Settings
Client Port		21212
Admin Port		21211
HTTP Port		8080
Internal Port		3021
Zookeeper Port	127.0.0.1	7181
Replication Port		0

## Directories

Root (Destination)	voltdbroot
Snapshot	snapshots
Export Overflow	export_overflow



# Wrap Up

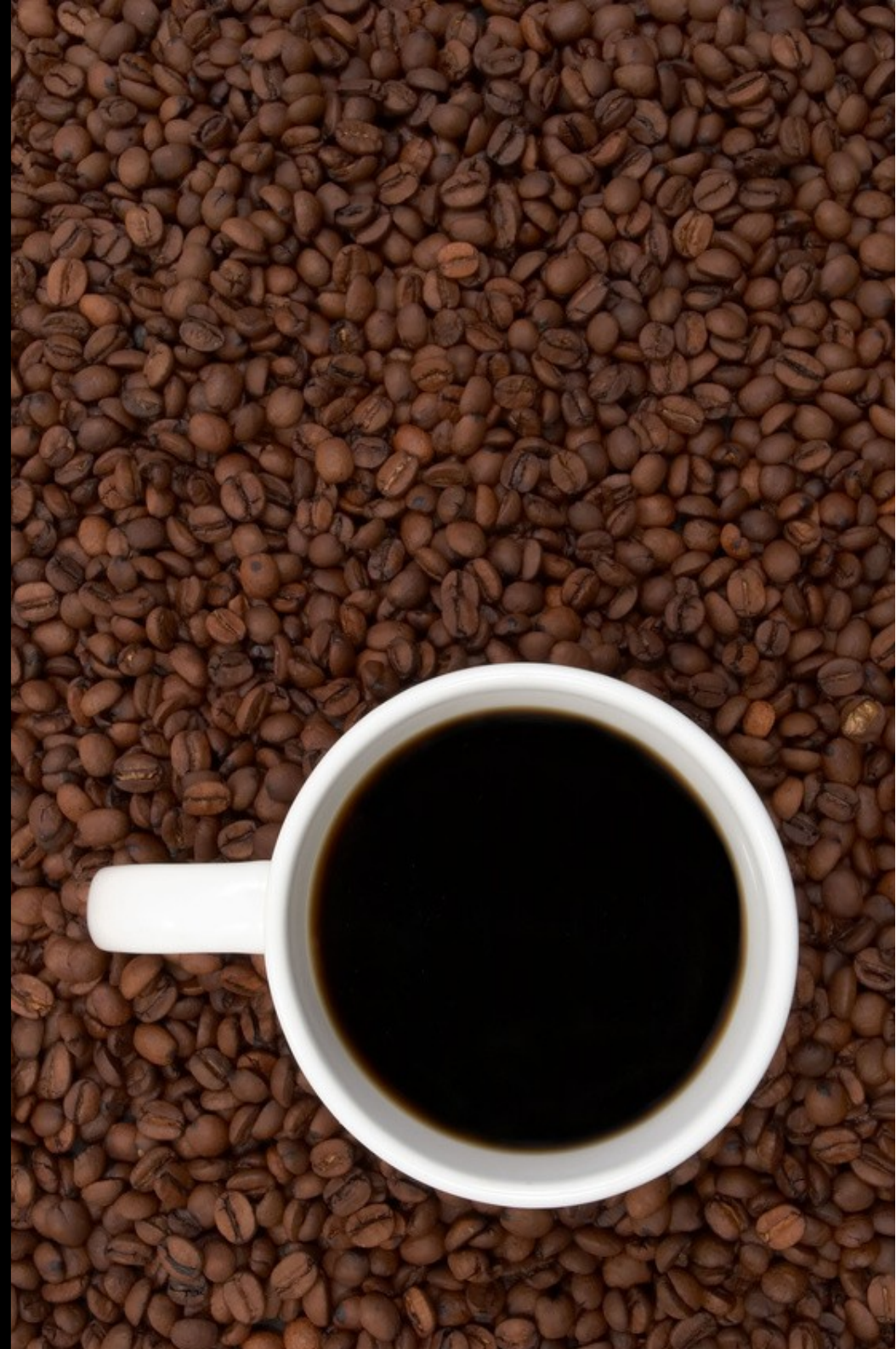
**NewSQL als dritter Weg**

**ACID and SQL can be fast**

**Choose your DB carefully**

**Jan Stamer**

jan.stamer@red6-es.de







# Farben (siehe Farbschema)

Theme 1

Theme 2

Theme 3

Theme 4

Theme 5

Theme 6



# CAP Theorem

