Systems@ETH Zürich

Database Engines on Multicores
Why Parallelize When You Can Distribute

Tudor-Ioan Salomie, Ionut Subasu, Gustavo Alonso, Jana Giceva
`tsalomie,subasu,alonso,gicevaj@{inf.ethz.ch}`

Multimed is a system that scales

with the **number of cores**

with the **number of clients**

Scalability throughput for 200 clients

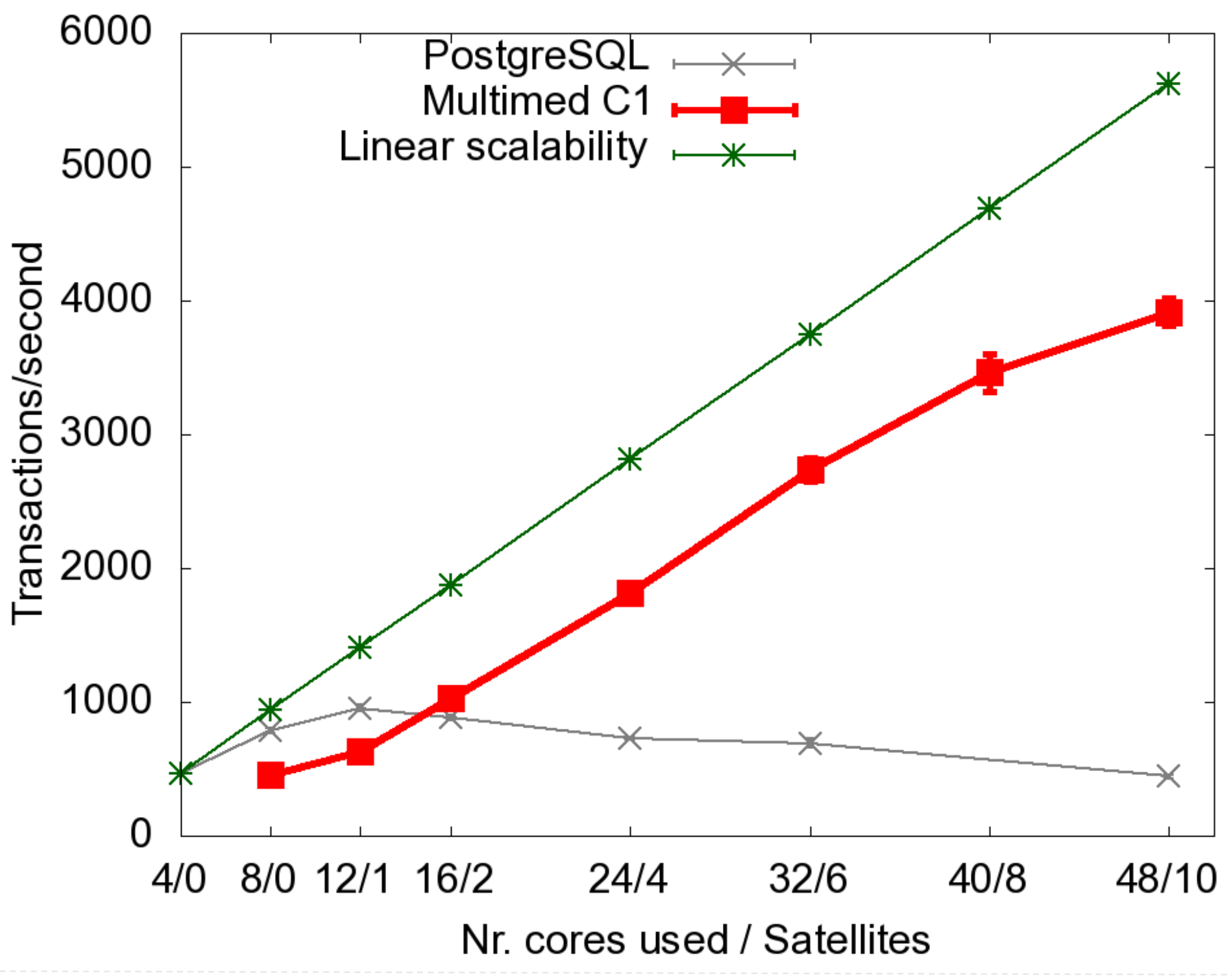
Throughput when varying the number of clients

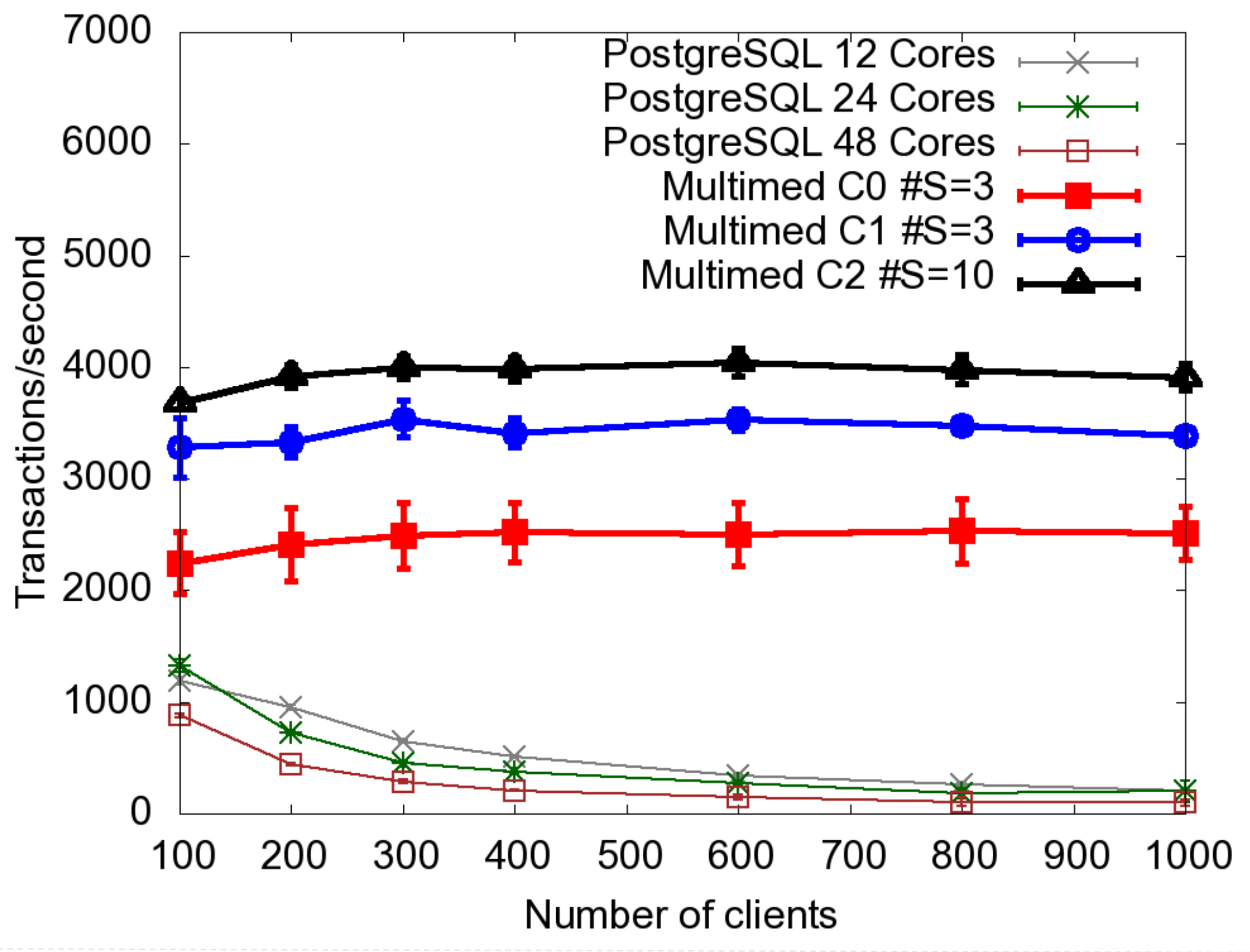
TPC-WB , database size **2GB**

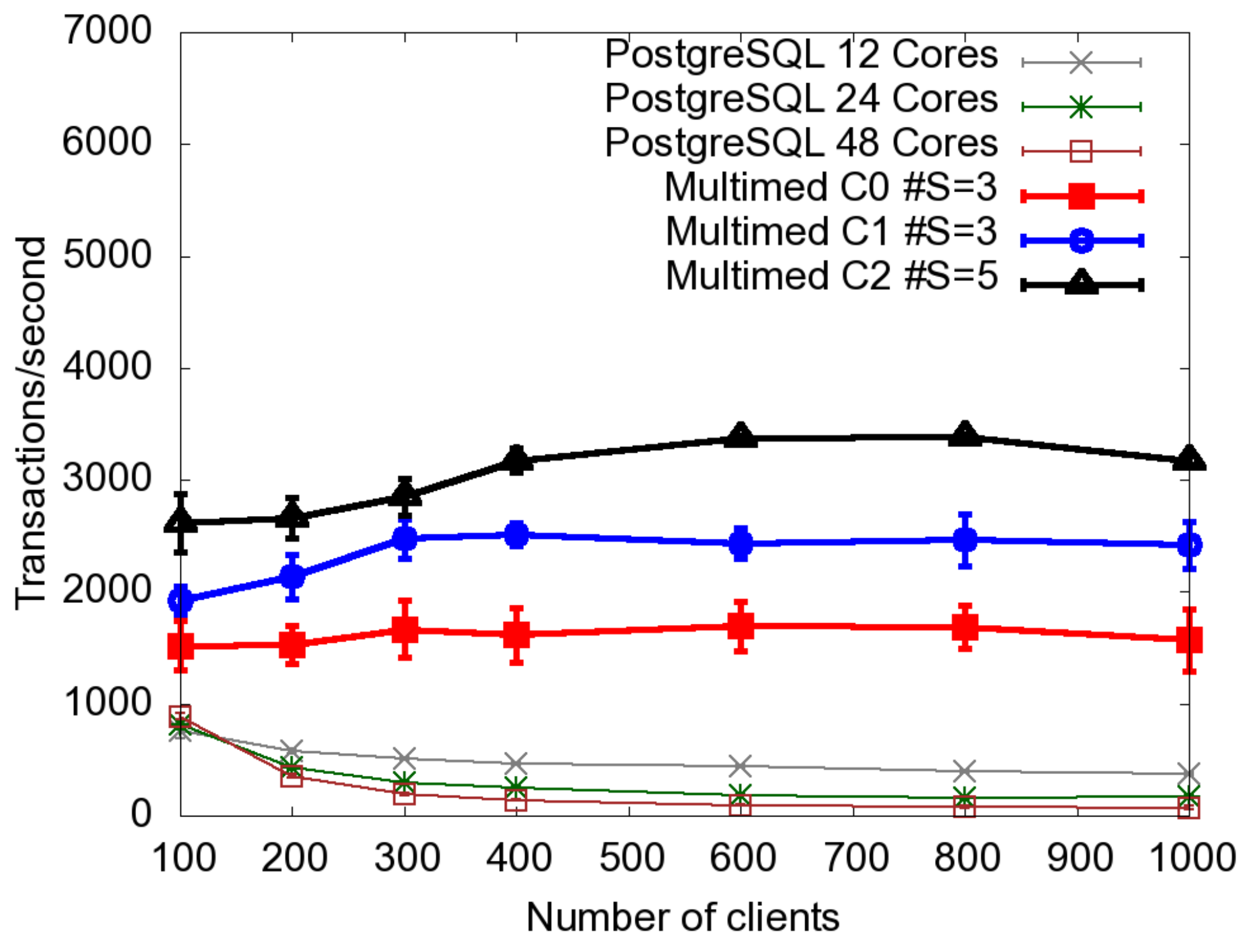
TPC-WB , Database size **2GB**

Database size **20GB**

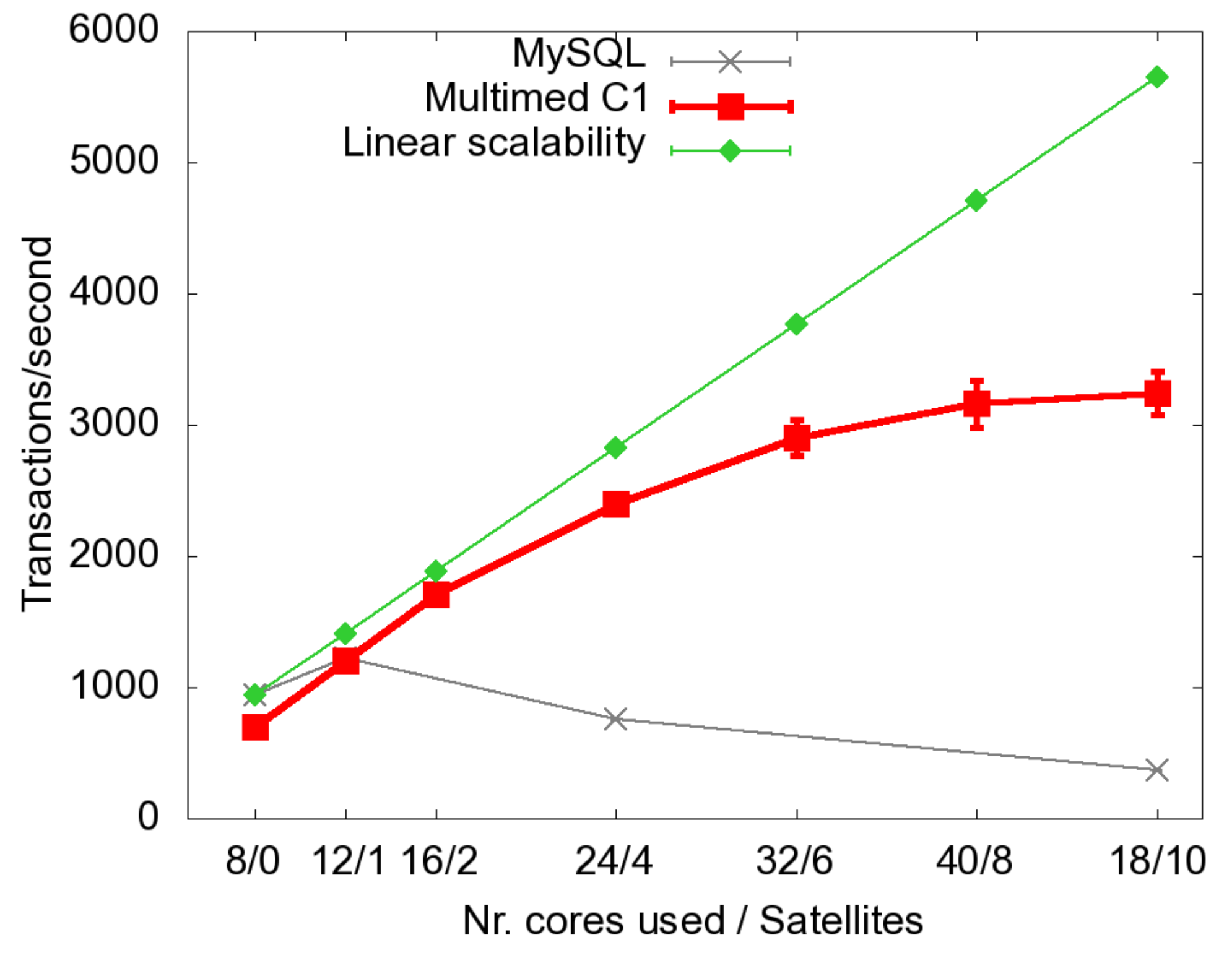
PostgreSQL

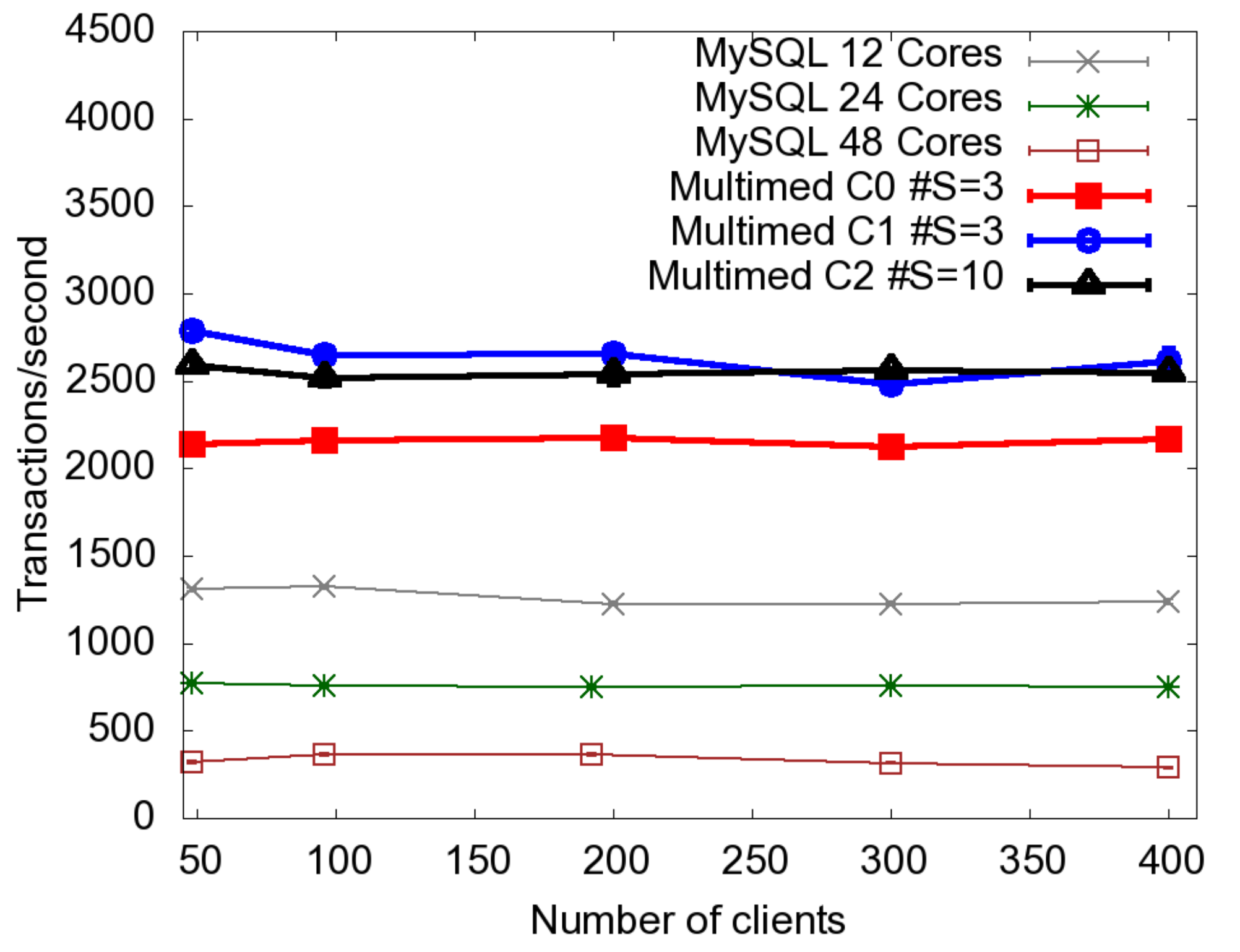


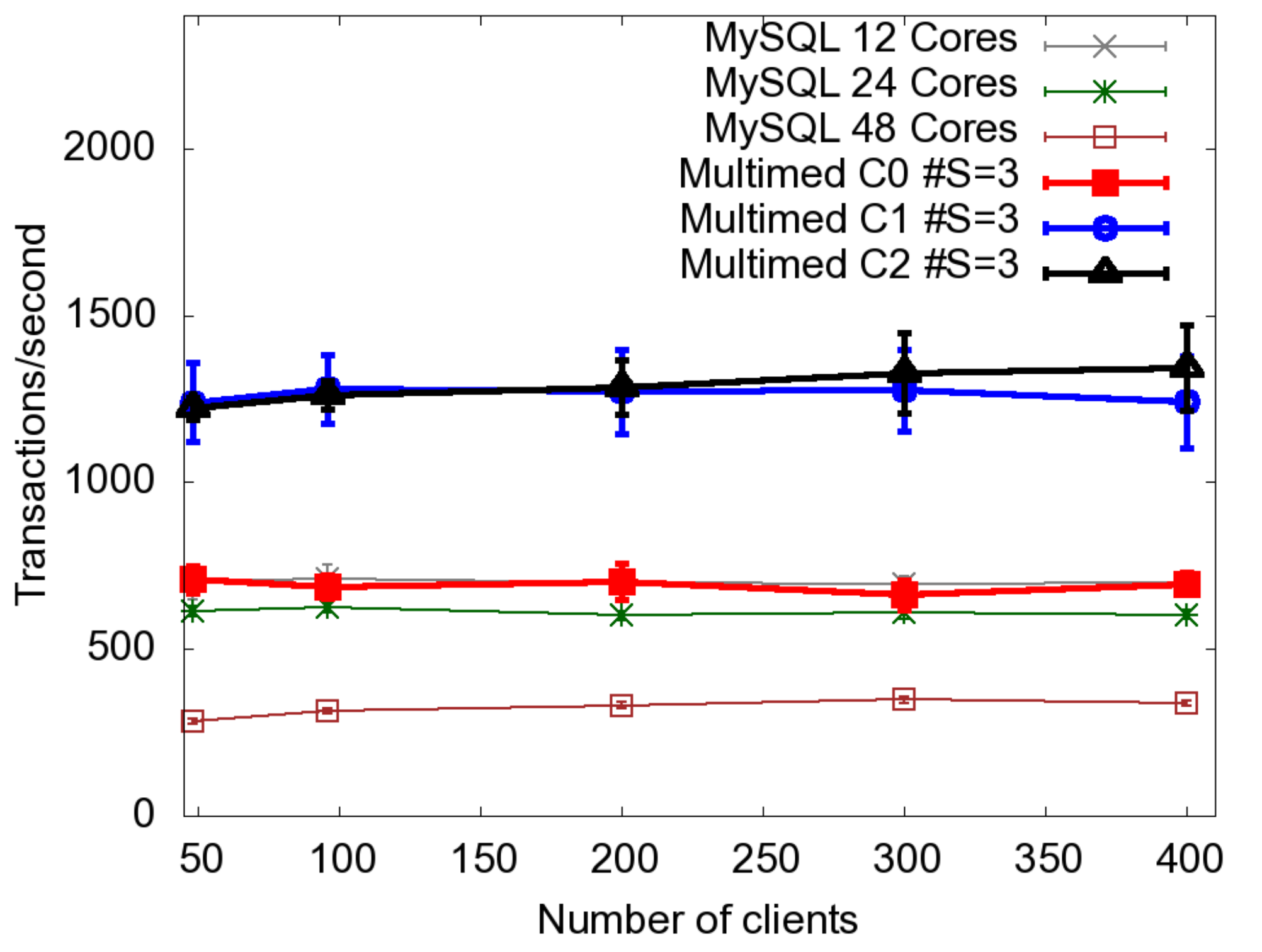




MySQL







Multimed adapts techniques used in database clusters to multicores.

The system **scales** with increasing number of cores and satellites and **separates loads** across databases

Different **optimizations** can be used to improve performance:

- **C0** full replication on disk
- **C1** full replication in main memory
- **C2** partial or full replication in main memory

Evaluation setup

Hardware
4 way AMD Magny Cours (6174), 48 cores, 128GB of RAM
Two dies per CPU, 6 cores per die. Each core has a local L1 (128KB) and L2 cache (512KB). Each die has a shared L3 cache (12MB).

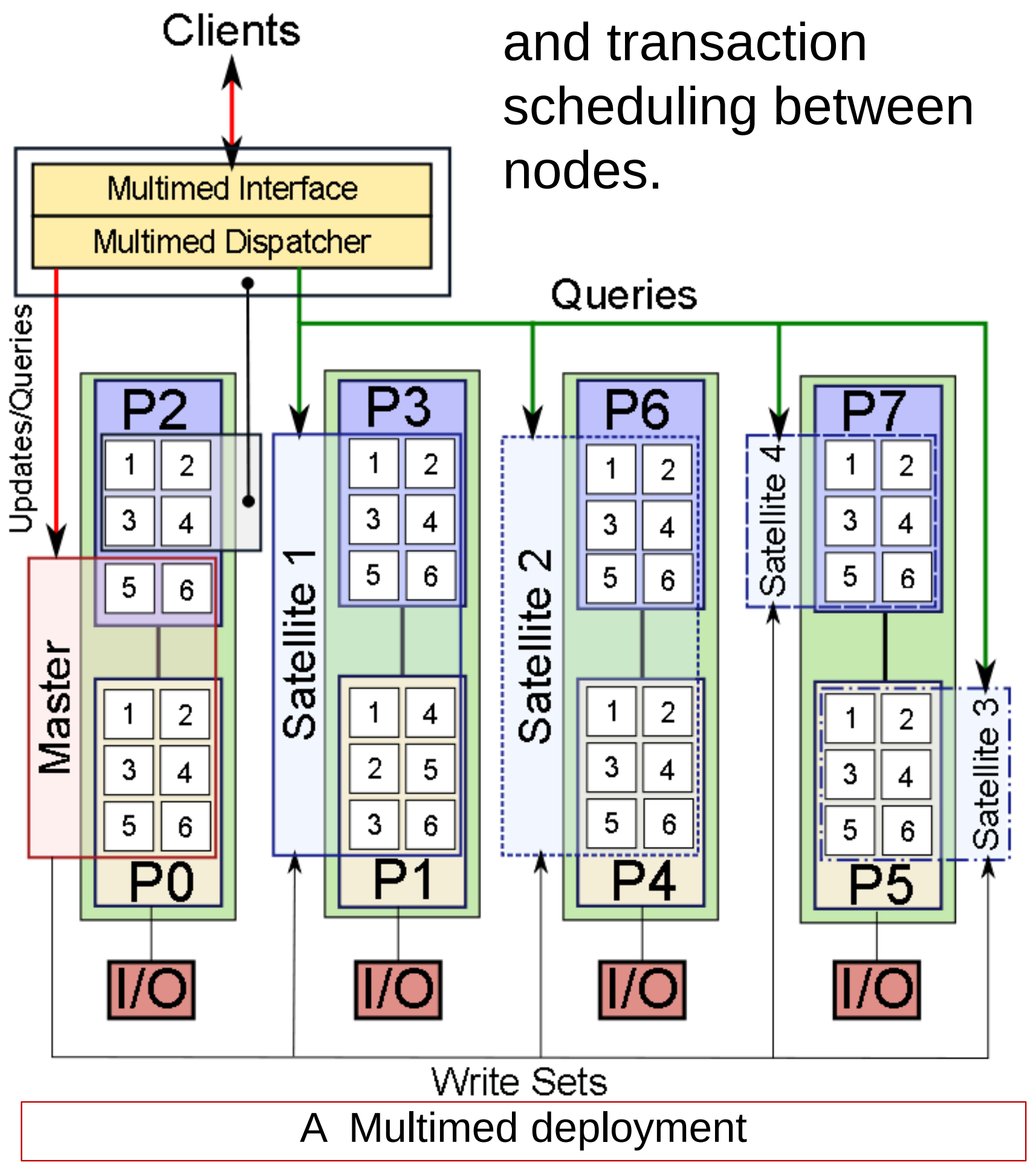
TPC-W Benchmark mixes:
TPC-W Browsing (10% Upd), **TPC-WB**
TPC-W Shopping (20% Upd), **TPC-WS**
TPC-W Ordering (50% Upd), **TPC-WO**

What is Multimed?

Multimed is a **replication** system designed **for multicore architectures** using **single-master**, multi-satellite replication.

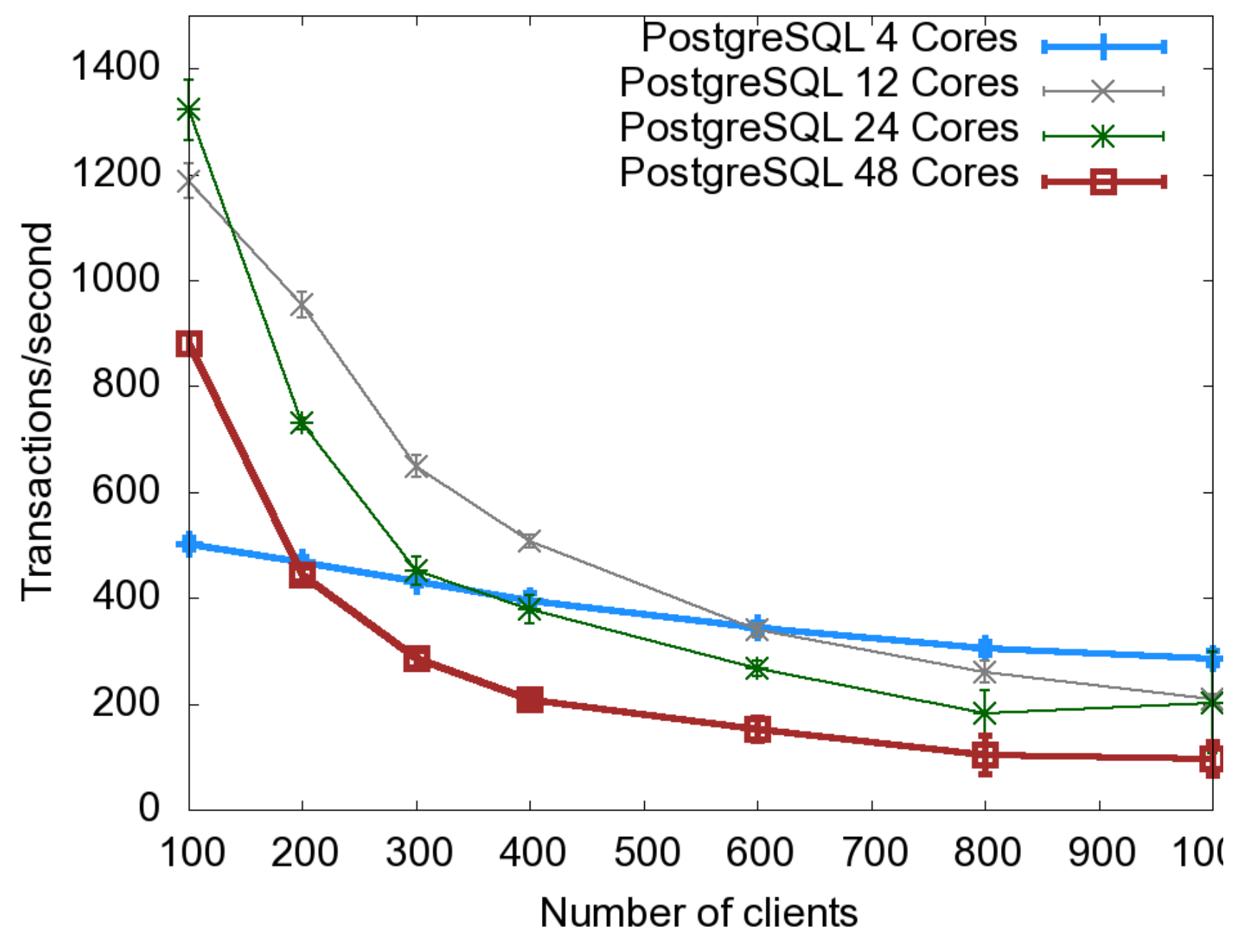
Supports **full and partial replication**. Using RSI-PC replication protocol.

Does load distribution and transaction scheduling between nodes.

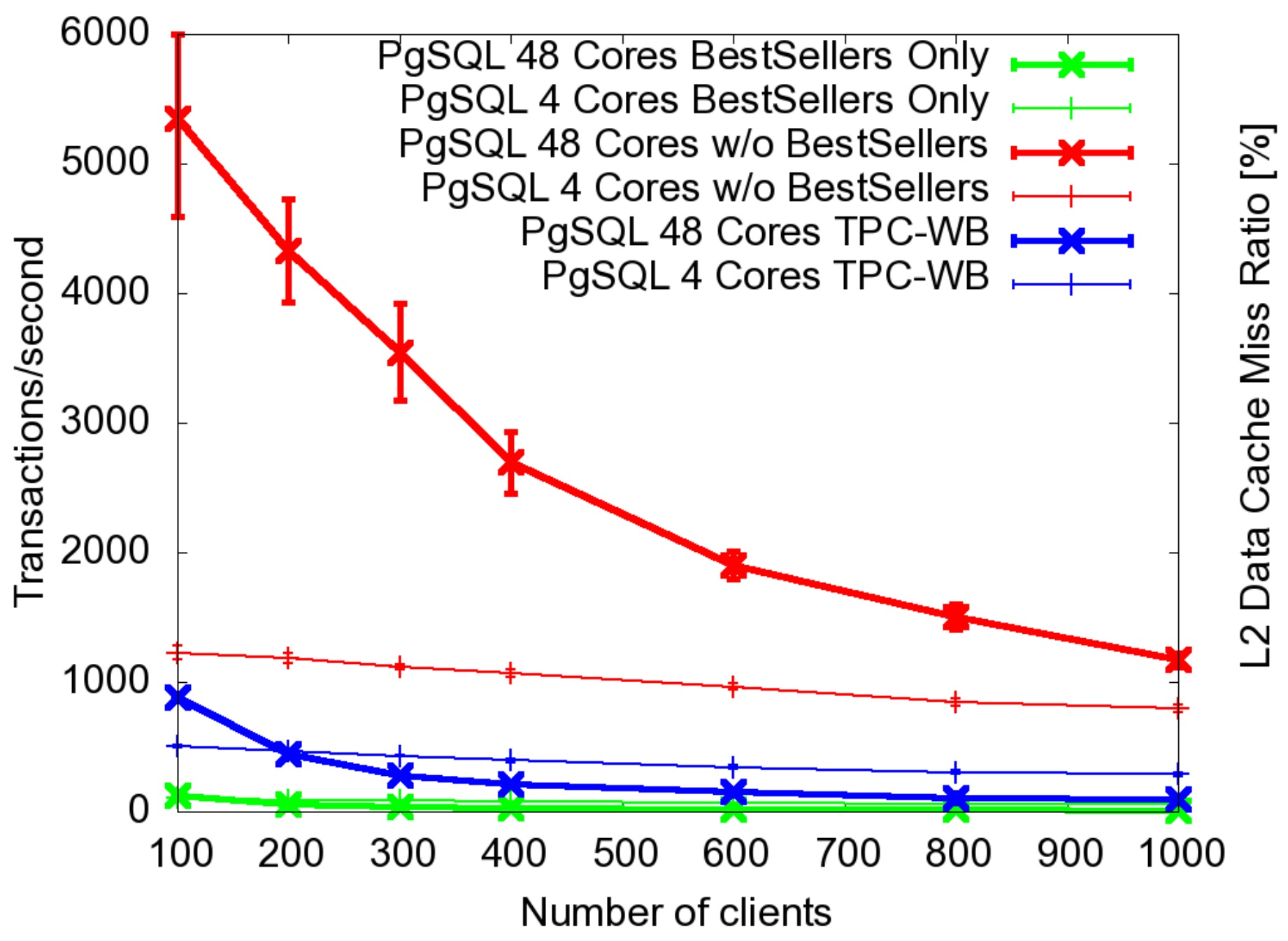


A Multimed deployment

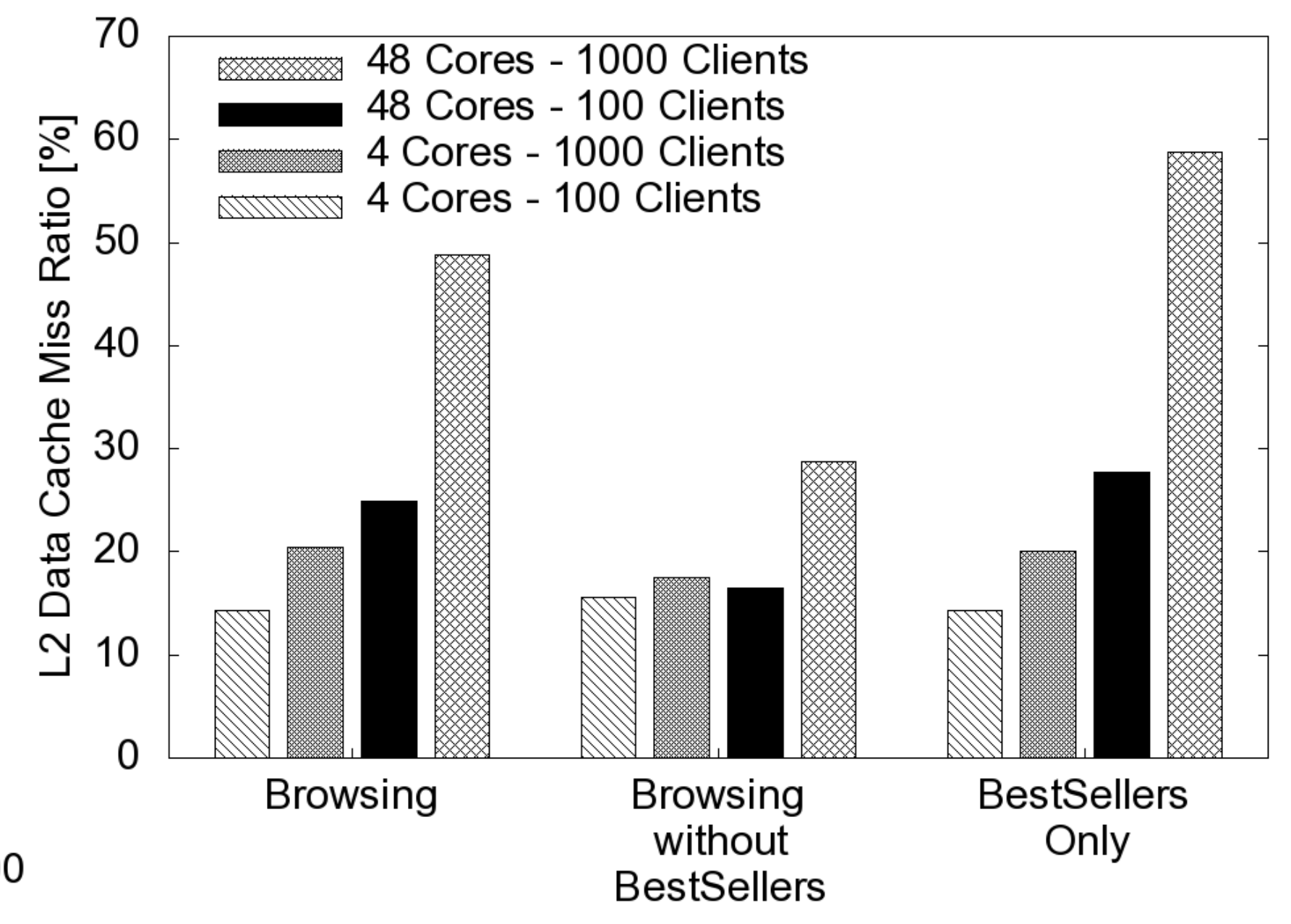
Database Engines on Multicores



PostgreSQL: TPC-WB Throughput



PostgreSQL: Load Interaction



PostgreSQL: L2 data cache miss ratio

Multimed is **independent of the database engine** running underneath.

Write set extraction is done using triggers for portability.

In our paper we have benchmarked PostgreSQL and MySQL.

Load Interaction leads to performance and scalability degradation due to **concurrent transactions that significantly interfere** with each other.

Multicores amplify it due to increased number of hardware contexts.

Contention due to **concurrent access to locks and synchronization primitives**, increases with number of cores and clients.

The L2 data cache miss rations is computed using the formula :

$$L2DC\_Miss\_Ratio = \frac{100 \times L2Cache\_Misses}{(L2Cache\_Fills + L2Requests)}$$

The values were measured using CPU event counters.

FNSNF
FONDS NATIONAL SUISSE
SCHWEIZERISCHER NATIONALFONDS
FONDO NAZIONALE SVIZZERO
SWISS NATIONAL SCIENCE FOUNDATION

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich