Introduction
Research on "Small Data". Dealing with data management challenges in personal and per-device interactions. Fuelled by smart devices and low-cost embedded computing platforms. At small scales, the law of large numbers is inapplicable and there is insufficient noise to absorb all of the outliers.

Mobile Databases
- Most mobile apps use embedded databases.* Operate in heterogeneous environments (varying battery, RAM, CPU, storage).
- Focus more on efficiency and share resources with other apps on the phone.

Whereas Database Servers:
- Are tuned for continuous high-throughput query processing
- Have exclusive access to all resources of a machine
- Focus on performance (throughout, latency), not efficiency.

POCKETDATA Toolchain:
- Nexus 6 phones. Custom Android ROM with instrumentation in the SQLite native layer
- YCSB Benchmark as an app
- Log collected about different events - DB connections, schemas, statement compilations, and queries into Android buffer
- Experiments to emulate different kinds of workloads

POCKETDATA Toolchain:
- Nexus 6 phones. Custom Android ROM with instrumentation in the SQLite native layer
- YCSB Benchmark as an app
- Log collected about different events - DB connections, schemas, statement compilations, and queries into Android buffer
- Experiments to emulate different kinds of workloads

Importance:
- Enabling research into cross-cutting communities working on data management, real-time and embedded devices, programming languages and operating systems.
- Helping app developers choose the best database
- Identifying performance bottlenecks in a database

Our Focus
Unlike traditional database research, we focus on low-throughput, bursty workloads under resource constrained environments.

POCKETDATA Toolchain:
- Nexus 6 phones. Custom Android ROM with instrumentation in the SQLite native layer
- YCSB Benchmark as an app
- Log collected about different events - DB connections, schemas, statement compilations, and queries into Android buffer
- Experiments to emulate different kinds of workloads

Comparing mobile databases
Comparisons among SQLite, BDB and BDB100 [See Figure below]:
A. SQLite is particularly tuned for scan-heavy workloads (YCSB E).
B. BDB wins for write-heavy workloads (YCSB A).
C. SQLite performs better for read-heavy workloads. (YCSB B)

Session Identification
- Traditional approaches to session identification fail in our scenario
- Connection time, Timeout and Semantic segmentation
- User tasks keep switching between background and foreground : Multi-tasking
- Database Sessions as a subset of repetitive logical user tasks
- Automatic session detection

Session Similarity
- Modular approach for session similarity.
- Considers both query features and activity distribution within the sessions
- Reveals shared activities
- Helps identifying common and unusual behavior patterns
- Variety of application areas such as predicting incoming queries to improving database performance

Applications
- Methodology for automatic benchmark generation from query logs.
- Identification of representative samples from query logs
- Guidance for on-the-fly DB performance tuning.

* Curino Et. Al. http://hdl.handle.net/1721.1/62241

The ODIn Lab @
odin.cse.buffalo.edu/research/pocketdata/index.html