Overview of BBDC and Related Research

Volker Markl
http://www.user.tu-berlin.de/marklv/
http://www.dima.tu-berlin.de
http://www.dfki.de/web/forschung/iam
http://bbdc.berlin

volker.markl@tu-berlin.de
Machine Learning + Data Management = X

Think ML-algorithms in a scalable way

Declarative Languages
Automatic Adaption
Scalable processing

Relational Algebra/SQL
Data Warehouse/OLAP
NF²/XQuery
Scalability
Hardware adaption
Fault Tolerance
Resource Management

Technology X

Goal: Data Analysis without System Programming!
X = Big Data Analytics – System Programming! („What“, not „How“)

Data Analyst

Description of „What“?
(declarative specification)
Technology X

Think ML-algorithms in a scalable way
Analysis of “data in motion”
Multimodal analysis
Numerical stability

Description of „How“?
(State of the art in scalable data analysis)

Hadoop, MPI

Larger human base of „data scientists“
Reduction of „human“ latencies
Cost reduction

Declarative specification
Automatic optimization, parallelization and hardware adaption of dataflow and control flow with user-defined functions, iterations and distributed state
Consistent intermediate results
Software-defined networking

Scalable algorithms and debugging
Algorithmic fault tolerance
Iterative algorithms in a scalable way

Machine
Application Examples: Technology Drivers and Validation

Technology X

Application example:
Marketplace for information
economics-based

Application Example:
Health
society-based

Application Example:
Material science
science-based

Think ML-algorithms in a scalable way
Declarative
Process iterative algorithms in a scalable way

text data flows
multiple data
integration: video, images, text
numerical stability
hierarchical numerical simulation data

© Volker Markl
Data & Analysis: Increasingly Complex!

Data
- Volume
- Velocity
- Veracity

Analysis
- Reporting
- Ad-Hoc Queries
- ETL/ELT
- Data Mining
- Predictive/Prescriptive

Scalability
- scalability
- algorithms

- data volume too large
- data rate too fast
- data too heterogeneous
- data too uncertain

MATLAB, R, Python
MATLAB, R, Python
“Data Scientist” – “Jack of All Trades!”

Domain Expertise (e.g., Industry 4.0, Medicine, Physics, Engineering, Energy, Logistics)
Mathematical Programming
Linear Algebra
Stochastic Gradient Descent
Error Estimation
Active Sampling
Regression
Monte Carlo
Statistics
Sketches
Hashing
Convergence
Decoupling
Iterative Algorithms
Curse of Dimensionality
Accuracy

Application

Data Science

Data Analysis

Scalable Data Management

Relational Algebra / SQL
Data Warehouse/OLAP
NF²/XQuery
Resource Management
Hardware Adaptation
Fault Tolerance
Memory Management
Parallelization
Scalability
Memory Hierarchy
Compression
Compiler
Query Optimization
Indexing
Data Flow
Control Flow
Real-Time
Performance

New Technology to the Rescue!
A Zoo of Technologies!

Big Data Landscape 2016 (Version 3.0)

Infrastructure
- Hadoop
- Apache
- IBM
- Pivotal
- Amazon
- AWS
- Azure
- Google
- Microsoft
- Oracle
- SAP
- IBM
- Pivotal
- Amazon
- AWS
- Azure
- Google
- Microsoft
- Oracle
- SAP

Analytics
- Hadoop
- Cloudera
- Apache
- IBM
- Pivotal
- Amazon
- AWS
- Azure
- Google
- Microsoft
- Oracle
- SAP

Applications
- Sales
- Marketing
- RA羊
- Data
- Social
- AI

Security
- AI
- Cloud
- Storage
- App
- Security
- Management
- Monitoring
- New Relic
- Log
- Social
- Search
- Cross-Infrastructure

Open Source
- Apache
- Spark
- Framework
- Query
- Data Access
- Coordination
- Real-Time
- Data

Data Sources & APIs
- Health
- IoT
- Financial & Economic
- Air
- Location
- Other

Last Updated 3/23/2016
© Matt Turck (@mattturck), Jim Hao (@jimhao), & FirstMark Capital (@firstmarkcap)
Apache Flink – Big Data Batch and Stream Processing

http://flink.apache.org
http://www.stratosphere.eu
Stratosphere: General Purpose Programming + Database Execution

Draws on Database Technology:
- Relational Algebra
- Declarativity
- Query Optimization
- Robust Out-of-core

Adds:
- Iterations
- Advanced Dataflows
- General APIs
- Native Streaming

Draws on MapReduce Technology:
- Scalability
- User-defined Functions
- Complex Data Types
- Schema on Read
What is Apache Flink?

Apache Flink is an open source platform for scalable batch and stream data processing.

A distributed system that you can use to process data
Like a DBMS but not exactly a DBMS

What kind of data?
Data that comes in the form of streams

What kind of processing
Quite flexible. You can use Java/Scala APIs similar to programming with Java collections, the new SQL API, etc

Distributed: runs on many (1000s) of machines and hides this complexity from the user

http://flink.apache.org
Technology inside Flink

```scala
case class Path (from: Long, to: Long)
val tc = edges.iterate(10) {
  paths: DataSet[Path] =>
  val next = paths .join(edges) .where("to") .equalTo("from") {
    (path, edge) =>
    Path(path.from, edge.to)
  } .union(paths) .distinct()
  next
 }
```

Type extraction

Cost-based optimizer

Dataflow Graph

Pre-flight (Client)

Program

DataSourse orders.tbl
Filter
Map
DataSourse lineitem.tbl
Join
Hybrid Hash
build HT
probe hash-part [0]
hash-part [0]
GroupRed
sort forward

deploy operators

track intermediate results

Worker

Memory manager
Out-of-core algos

Batch & Streaming
State & Checkpoints

Recovery metadata
Task scheduling

Master

© Volker Markl

THE FLINK COMMUNITY
The Flink Community: Meetups By Country Concerning Flink

Apache Flink Meetups Worldwide (Data accurate as of 30.5.16)
6326 members strictly focused on Apache Flink (comprising 57%)
4771 members broader in scope, including Flink (comprising 43%)
Flink community (2)

• More than 250 people have contributed code to Flink

By courtesy of Kostas Tzoumas
Zalando, one of the largest ecommerce companies in Europe, uses Flink for real-time business process monitoring.

King, the creators of Candy Crush Saga, uses Flink to provide data science teams with real-time analytics.

Alibaba, the world's largest retailer, built a Flink-based system (Blink) to optimize search rankings in real time.

Bouygues Telecom uses Flink for real-time event processing over billions of Kafka messages per day.

By courtesy of Kostas Tzoumas
Largest job has > 20 operators, runs on > 5000 vCores in 1000-node cluster, processes millions of events per second

Complex jobs of > 30 operators running 24/7, processing 30 billion events daily, maintaining state of 100s of GB with exactly-once guarantees

30 Flink applications in production for more than one year. 10 billion events (2TB) processed daily

By courtesy of Kostas Tzoumas
> 20 Companies Using Flink

- Alibaba.com
- bouygues TELECOM
- KKAM
- ERICSSON
- Pragsis Bidoop
- treelogic
- ResearchGate
- mbttargeting
- zalando
- otto group
- EURIX NOVA
- RADICALBIT
- FIRELAYERS
- Atos
- dataArtisans
- amadeus
> 8 Software Projects Using Flink

**Google Cloud Platform**

CLOUD DATAFLOW
A fully-managed cloud service and programming model for batch and streaming big data processing.

**CASCADING**

Apache Flink is a replacement for MapReduce to support large-scale batch workloads and streaming data flows. It eliminates the concept of mapping and reducers and leverages in-memory storage, resulting in significant performance gains over MapReduce.

**Apache SAMOA**

Apache SAMOA is a distributed streaming machine learning (ML) framework that contains a programming abstraction for distributed streaming ML algorithms.

**Apache Streams™**

The Apache Mahout™ project’s goal is to build an environment for quickly creating scalable performant machine learning applications.

**Apache MRQL**

MRQL is a query processing and optimization system for large-scale, distributed data analysis, built on top of Apache Hadoop, Hama, Spark, and Flink.

**DMLC**

Apache Beam is an open source, unified programming model that you can use to create a data processing pipeline.
> 10 Research Institutions
Using Flink
Flink in the ecosystem

By courtesy of Kostas Tzoumas
Declarative Data Processing and Big Data
A Billion $$$ Mantra...

Declarative Data Processing

An effective, formal foundation based on relational algebra and calculus (Codd ’71).

A simple, high-level language for querying data (Chamberlin ’74).

An efficient, low-level execution environment tailored towards the data (Selinger ’79).
With 40+ years of success...

SQL

Relations

RDBMS

Declarative Data Processing
Declarative Data Processing

SQL

Relations

RDBMS

Second-Order Functions

Distributed Collections

Parallel Dataflow Engines

Is Being Revised
Mosaics

Frontend:

Backends:
Research

1. Unifying Modelling Across Theories
2. Cross Theory Optimization
3. Optimizing Across Engines
4. Predicting and Learning Program Runtimes
5. Optimizing Across Hardware
6. Generating Hardware-Targeted Code
Emma Compiler Overview

**Programming Abstractions**

**Common Concrete Syntax**
( realized as Scala eDSL)

**Common Abstract Syntax**

* e.g. Spark, Flink, CoGaDB
Falcon Query Compiler

SQL
Emma

Variant Generator

Pipeline variants

Pipeline programs
Evolution of Big Data Platforms

- **First Generation**
  - Data Warehouses, e.g., relational DBMS

- **Second Generation**
  - Scale-out, Map/Reduce, UDFs, e.g., Apache Hadoop

- **Third Generation**
  - In-memory Performance and Improved Programming Model, e.g., Apache Spark

- **Fourth Generation**
  - In-memory + Out of Core Performance, Declarativity, Optimization of Iterative Algorithms, True Streaming e.g., Apache Flink