Tangram: Bridging Immutable and Mutable Abstractions for Distributed Data Analytics

Yuzhen Huang, Xiao Yan, Guanxian Jiang, Tatiana Jin,

James Cheng, An Xu, Zhanhao Liu, and Shuo Tu,

The Chinese University of Hong Kong

Distributed Data Analytics Systems

Existing offline data analytics frameworks can be roughly classified into two categories according to their data abstractions

- **Immutable** or **mutable**

Immutable

TEZ

hadoop
Map Reduce

Spark

Microsoft
DryadLINQ

HIVE

Mutable

Parameter Server (PS)

PETUUM

Pregel
oogle

TensorFlow

GraphLab

Immutable and Mutable Abstractions

Immutable	Mutable
+ Functional API + Fault tolerance + Load balancing	+ Stateful representation + Iterative and asynchronous execution
- Inefficient for stateful representation - Only support BSP	- Fault tolerance - Load Balancing

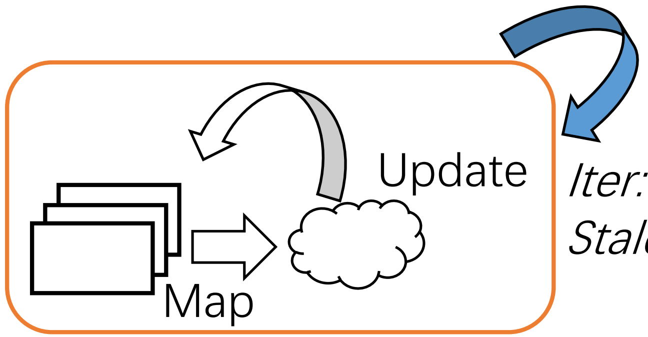
Immutable and Mutable Abstractions

Immutable	Mutable
+ Functional API + Fault tolerance + Load balancing	+ Stateful representation + Iterative and asynchronous execution
- Inefficient for stateful representation - Only support BSP	- Fault tolerance - Load Balancing

MapUpdate: Bridging immutable and mutable abstractions

MapUpdate

A.**map**(B, map_func).**update**(C, update_func)



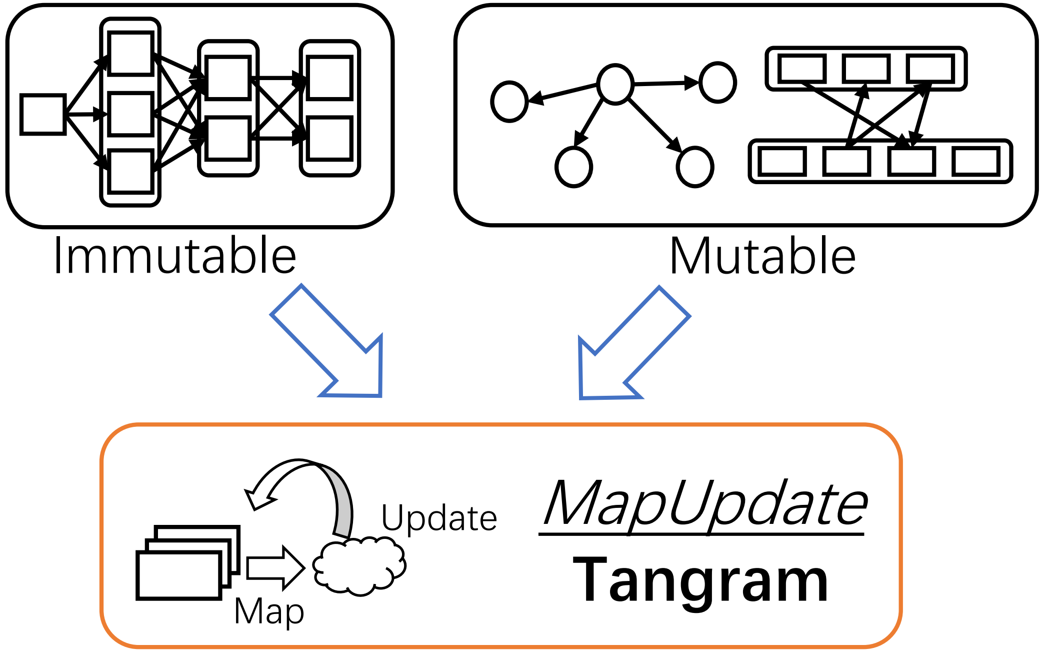
Iter: 100
Staleness: 2

- Expressive
  - Bulk processing, machine learning, graph analytics, etc.
- Enjoys the benefits of both mutable and immutable abstractions
  - Determines whether a collection is mutable automatically
  - Supports iterative and asynchronous execution naturally
  - Applies different recovery strategies adaptively according to failure scenarios

Tangram

A distributed system that implements MapUpdate

- Local Task Management
- Partition-based Progress Control
- Context-Aware Failure Recovery



Open source:
https://github.com/Yuzhen11/tangram