SystemML

SystemML

Abstract

SystemML provides declarative large-scale machine learning (ML) that aims at flexible specification of ML algorithms and automatic generation of hybrid runtime plans ranging from single node, in-memory computations, to distributed computations on Apache Hadoop MapReduce and Apache Spark. ML algorithms are expressed in an R-like syntax, that includes linear algebra primitives, statistical functions, and ML-specific constructs. This high-level language significantly increases the productivity of data scientists as it provides (1) full flexibility in expressing custom analytics, and (2) data independence from the underlying input formats and physical data representations. Automatic optimization according to data characteristics such as distribution on the disk file system, and sparsity as well as processing characteristics in the distributed environment like number of nodes, CPU, memory per node, ensures both efficiency and scalability.

Proposal

The goal of SystemML is to create a commercial friendly, scalable and extensible machine learning framework for data scientists to create or extend machine learning algorithms using a declarative syntax. The machine learning framework enables data scientists to develop algorithms locally without the need of a distributed cluster, and scale up and scale out the execution of these algorithms to distributed Apache Hadoop MapReduce or Apache Spark clusters.

Background

SystemML started as a research project in the IBM Almaden Research Center around 2007 aiming to enable data scientists to develop machine learning algorithms independent of data and cluster characteristics.

Rationale

SystemML enables the specification of machine learning algorithms using a declarative machine learning (DML) language. DML includes linear algebra primitives, statistical functions, and additional constructs. This high-level language significantly increases the productivity of data scientists as it provides (1) full flexibility in expressing custom analytics and (2) data independence from the underlying input formats and physical data representations.

SystemML computations can be executed in a variety of different modes. It supports single node in-memory computations and large-scale distributed cluster computations. This allows the user to quickly prototype new algorithms in local environments but automatically scale to large data sizes as well without changing the algorithm implementation.

Algorithms specified in DML are dynamically compiled and optimized based on data and cluster characteristics using rule-based and cost-based optimization techniques. The optimizer automatically generates hybrid runtime execution plans ranging from in-memory single-node execution to distributed computations on Apache Spark or Apache Hadoop MapReduce. This ensures both efficiency and scalability. Automatic optimization reduces or eliminates the need to hand-tune distributed runtime execution plans and system configurations.

Initial Goals

The initial goals to move SystemML to the Apache Incubator is to broaden the community foster the contributions from data scientists to develop new machine learning algorithms and enhance the existing ones. Ultimately, this may lead to the creation of an industry standard in specifying machine learning algorithms.

Current Status

The initial code has been developed at the IBM Almaden Research Center in California and has recently been made available in GitHub under the Apache Software License 2.0. The project currently supports a single node (in memory computation) as well as distributed computations utilizing Apache Hadoop MapReduce or Apache Spark clusters.

Meritocracy

We plan to invest in supporting a meritocracy. We will discuss the requirements in an open forum. Several companies have already expressed interest in this project, and we intend to invite additional developers to participate. We will encourage and monitor community participation so that privileges can be extended to those that contribute operating to the standard of meritocracy that Apache emphasizes.

Community

The need for a generic scalable and declarative machine learning approach in the open source is tremendous, so there is a potential for a very large community. We believe that SystemML’s extensible architecture, declarative syntax, cost based optimizer and its alignment with Spark will further encourage community participation not only in enhancing the infrastructure but also speed up the creation of algorithms for a wide range of use cases. We expect that over time SystemML will attract a large community.

Alignment

The initial committers strongly believe that a generic scalable and declarative machine learning approach for machine learning will gain broader adoption as an open source, community driven project, where the community can contribute not only to the core components, but also to a growing collection of algorithms which will leverage the optimizations and ease of scaling in SystemML. Our hope is that the Apache Spark, Apache Hadoop and other communities will find tremendous value in SystemML and this will foster further collaboration between these projects furthering the already existing integration points.

Known Risks

To-date, development has been sponsored by IBM and coordinated mostly by the core team of researchers at the IBM Almaden Research Center.

For SystemML to fully transition to an "Apache Way" governance model, it needs to start embracing the meritocracy-centric way of growing the community of contributors.

Orphaned Products

The SystemML developers and previous sponsor have a long-term interest in use and maintenance of the code and there is also hope that growing a diverse community around the project will become a guarantee against the project becoming orphaned. We feel that it is also important to put formal governance in place both for the project and the contributors as the project expands. We feel ASF is the best location for this.

Inexperience with Open Source

The current SystemML set of contributors are very diverse regarding participation in Open Source. While some initial members are experiencing an open source project for the first time, others have been contributing and mentoring various Apache and non-Apache open source projects.

Reliance on Salaried Developers

SystemML currently receives substantial support from salaried developers. However, they are all passionate about the project, and we are confident that the project will continue even if no salaried developers contribute to the project. We are committed to recruiting additional committers including non-salaried developers.

Relationships with Other Apache Products

Currently, SystemML integrates with Apache Hadoop MapReduce and Apache Spark as underlying computational distributed runtimes.

An Excessive Fascination with the Apache Brand

SystemML solves a real need for generic scalable and declarative machine learning approach for machine learning in the Apache Hadoop and Spark ecosystems, something that has been addressed in a very ad hoc manner so far by multiple Apache projects. Our rationale for developing SystemML as an Apache project is detailed in the Rationale section. We believe that the Apache brand and community process will help us attract more contributors to this project, and help establish ubiquitous APIs.

Documentation

Documentation regarding SystemML is available in the current GitHub repository https://github.com/SparkTC/systemml/tree/master/system-ml/docs.

Initial Source

Initial source is available on GitHub under the Apache License 2.0

https://github.com/SparkTC/systemml

Source and Intellectual Property Submission Plan

We know of no legal encumbrances in the transfer of source code and rights to Apache. In fact, given the internal IBM due diligence performed on the source code during open sourcing, we expect the code base to be free from any IP issues.

External Dependencies

SystemML is written in Java and currently supports Apache Hadoop MapReduce and Apache Spark runtimes.

To the best of our knowledge, all dependencies of SystemML are distributed under Apache compatible licenses. Upon acceptance to the incubator, we would begin a thorough analysis of all transitive dependencies to verify this fact and introduce license checking into the build and release process (for instance integrating Apache Rat).

Cryptography N/A

Required Resources

Mailing lists

- private@sysml.incubator.apache.org (moderated subscriptions)
- commits@sysml.incubator.apache.org
- dev@sysml.incubator.apache.org

Git Repository

- https://git-wip-us.apache.org/repos/asf/incubator-sysml.git

Issue Tracking

- JIRA (SYSML)

Initial Committers

- Luciano Resende (lresende AT apache DOT org)
- Berthold Reinwald (reinwald AT us DOT ibm DOT com)
- Matthias Boehm (mboehm AT us DOT ibm DOT com)
- Shirish Tatikonda (statiko AT us DOT ibm DOT com)
- Niketan Pansare (npansar AT us DOT ibm DOT com)
- Prithviraj Sen (senp AT us DOT ibm DOT com)
- Alexandre V Evfimievski (evfimi AT us DOT ibm DOT com)
- Fred Reiss (frreiss AT us DOT ibm DOT com)
- Deron Eriksson (deron AT us DOT ibm DOT com)
- Arvind Surve (asurve AT us DOT ibm DOT com)
- Mike Dusenberry (mwdusenb AT us DOT ibm DOT com)
- Reynold Xin (rxin AT apache DOT org)
- Xiangrui Meng (meng AT apache DOT org)
- Joseph Bradley (jkbradley AT apache DOT org)
- Patrick Wendell (pwendell AT apache DOT org)
- Holden Karau (holden AT apache DOT org)
- DB Tsai (dbtsai AT apache DOT org)

Affiliations

- DataBricks: Reynold Xin, Xiangrui Meng, Joseph Bradley, Patrick Wendell
- Netflix: DB Tsai
- IBM: Luciano Resende, Berthold Reinwald, Matthias Boehm, Shirish Tatikonda, Niketan Pansare, Prithviraj Sen, Alexandre V Evfimievski, Fred Reiss, Deron Eriksson, Arvind Surve, Mike Dusenberry and Holden Karau.

Sponsors

Champion

- Luciano Resende

Nominated Mentors

- Luciano Resende
- Reynold Xin
- Patrick Wendell
- Rich Bowen

Sponsoring Entity

We would like to propose the Apache Incubator to sponsor this project.