\Orchestrating a brighter world

NEC

Machine learning and data analytics middleware for SX-Aurora TSUBASA

NEC Data Science Research Laboratories
Senior Principal Researcher
Takuya Araki, Ph.D.

Machine leaning in Big Data analytics

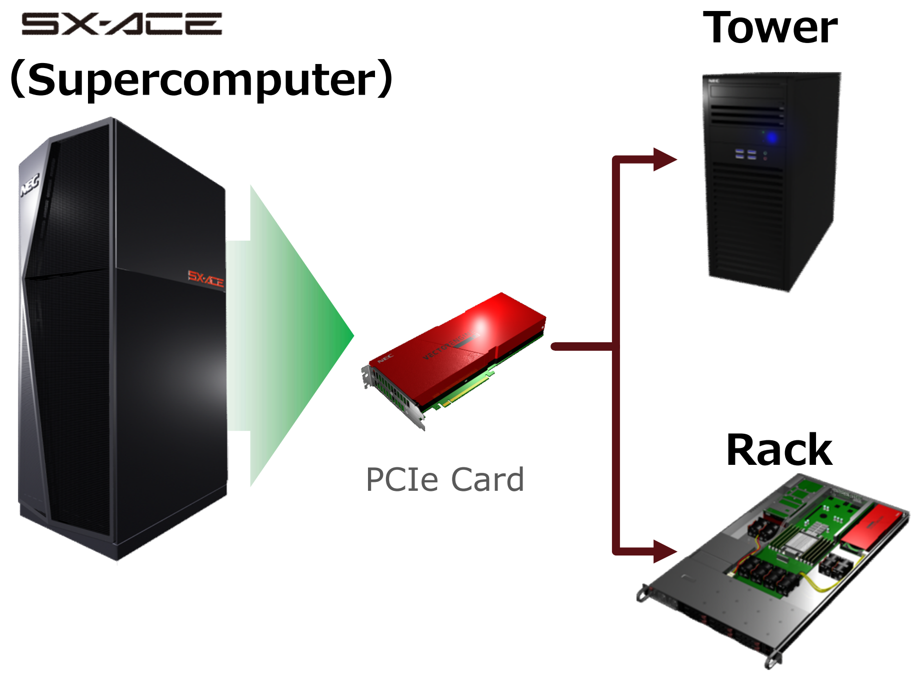
- Recently, machine learning (ML) is becoming important in Big Data analytics

- Most ML algorithms can be written as “matrix operation”; Large scale ML tends to use “sparse matrix”, which is memory intensive
  - Vector architecture is promising

- We created middleware for ML that runs on vector architecture
- In addition, we made middleware that can be seamlessly called from Apache Spark and Python
  - More than 50x performance improvement
  - Users of Spark/Python can easily utilize high performance of vector without special programming

SX-Aurora TSUBASA as the new generation of AI/BD accelerator

SX-Aurora TSUBASA



POINT 1 High Performance

Enables to achieve a high performance on memory intensive applications

POINT 2 Easy to use

Can program with standard C and C++, and our compiler generates an optimized code.

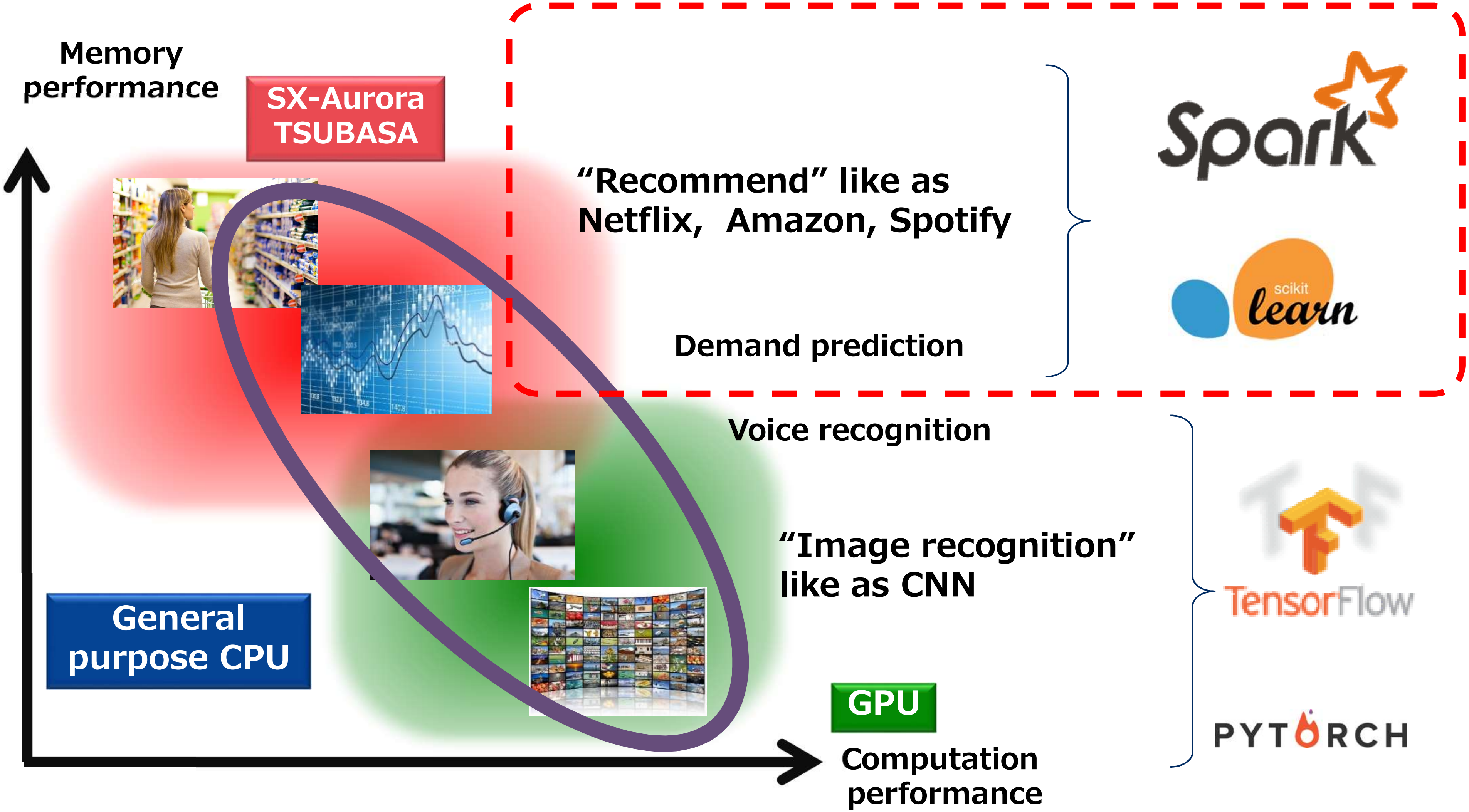
POINT 3 Flexible

Support several form factor; tower, rack, and HPC models.

Downsized super computer:
it can be used as an accelerator for Big Data and AI

Position of SX-Aurora TSUBASA

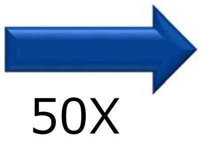
We target accelerating memory intensive workloads for Big Data/AI



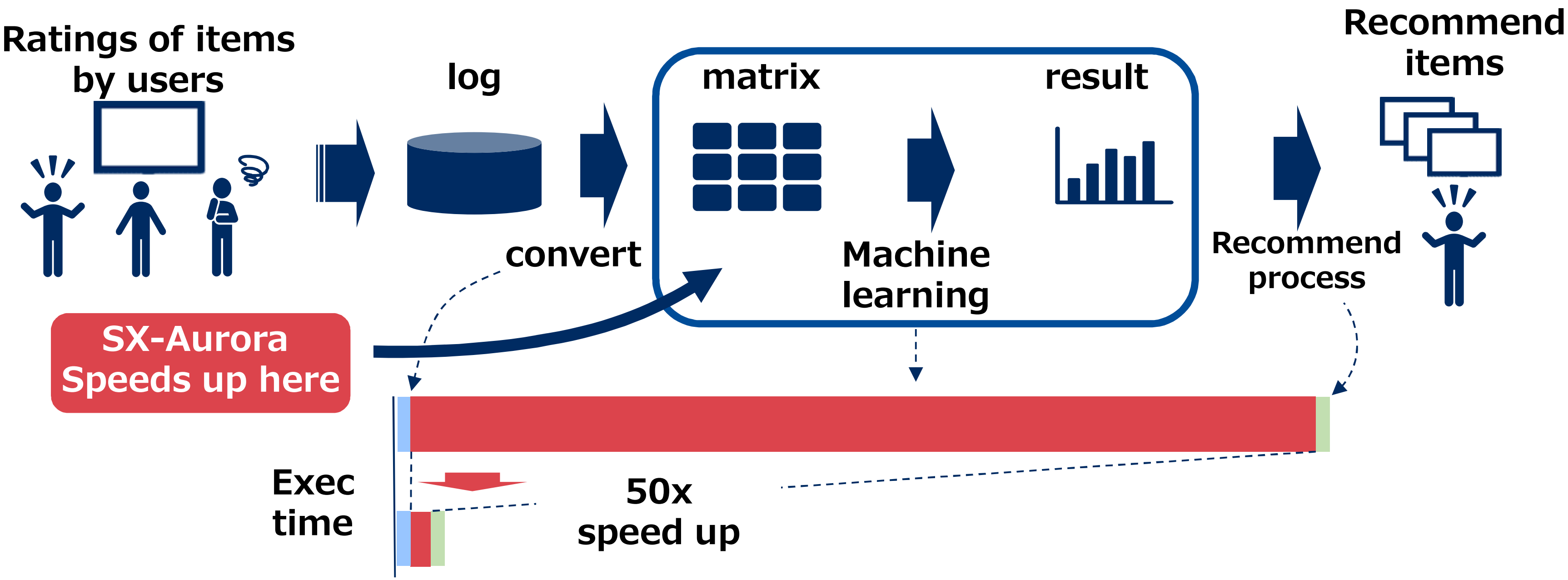
Example application: recommender system

SX-Aurora TSUBASA can reduce the number of servers by 1/50

- 35% sales of Amazon, 75% sales of Netflix is from recommendation
- More than 95% of the execution time is spent on machine learning
  - SX-Aurora TSUBASA showed 50x speed up with small data



1/50 computing power consumption
Every 30 min updates from 24hours updates



Apache Spark for Big Data analytics

Spark is de facto standard of statistical machine learning middleware

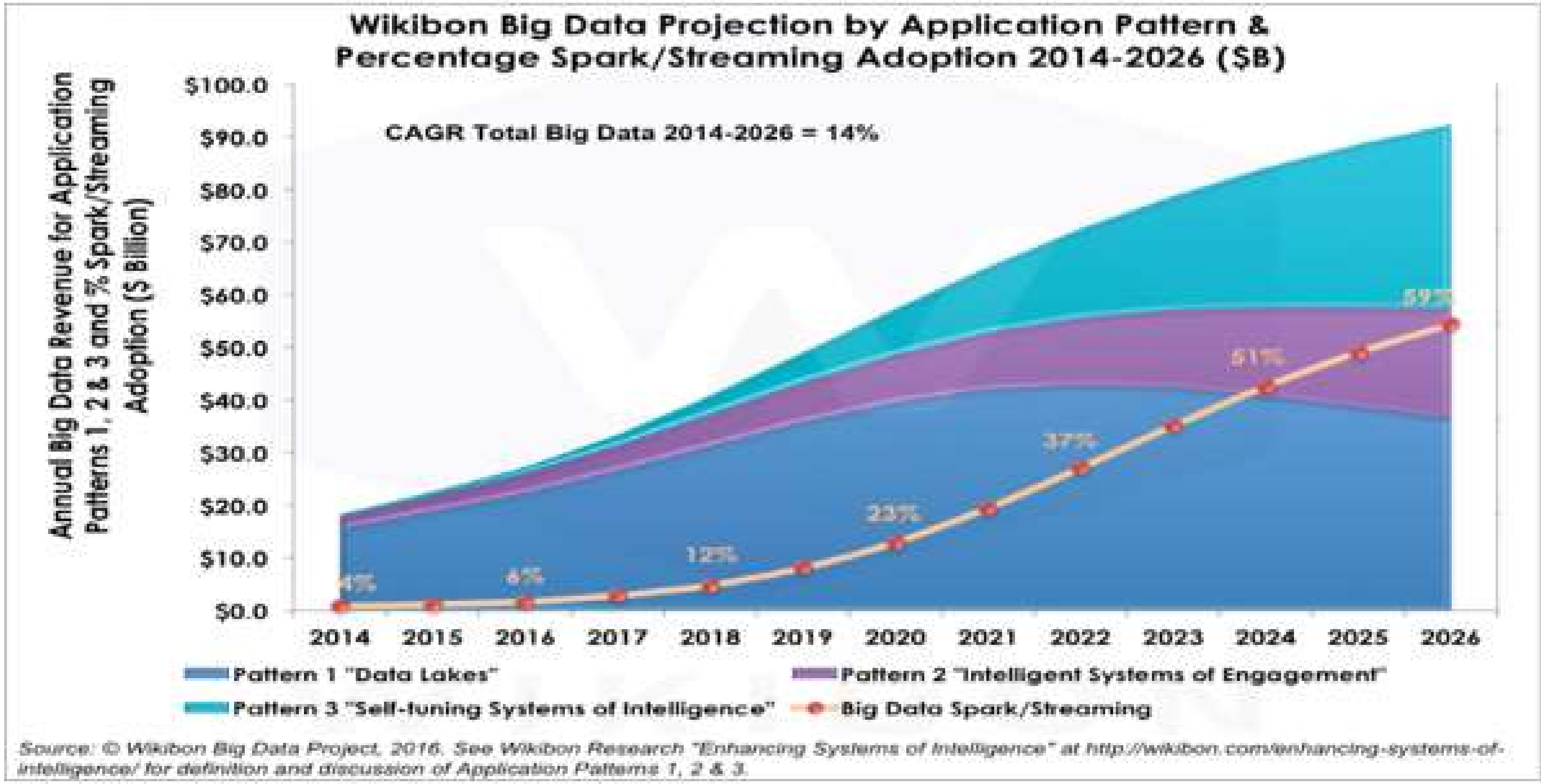


Figure: Wikibon Big Data Size and Projections

Frovedis: middleware for statistical machine learning

Middleware that provides interface like Spark

- Written in C++
- Internally uses MPI to implement distributed processing
- Users need not be aware of MPI to write distributed processing
- Support seamless interface from Spark/Python

C++ Example: double each element of distributed variable

```
int two_times(int i) {return i * 2;}
int main(...) {
  ...
  dvector<int> r = d1.map(two_times);
}
```

distributed variable

run "two_times" in parallel

Open Sourced
https://github.com/frovedis

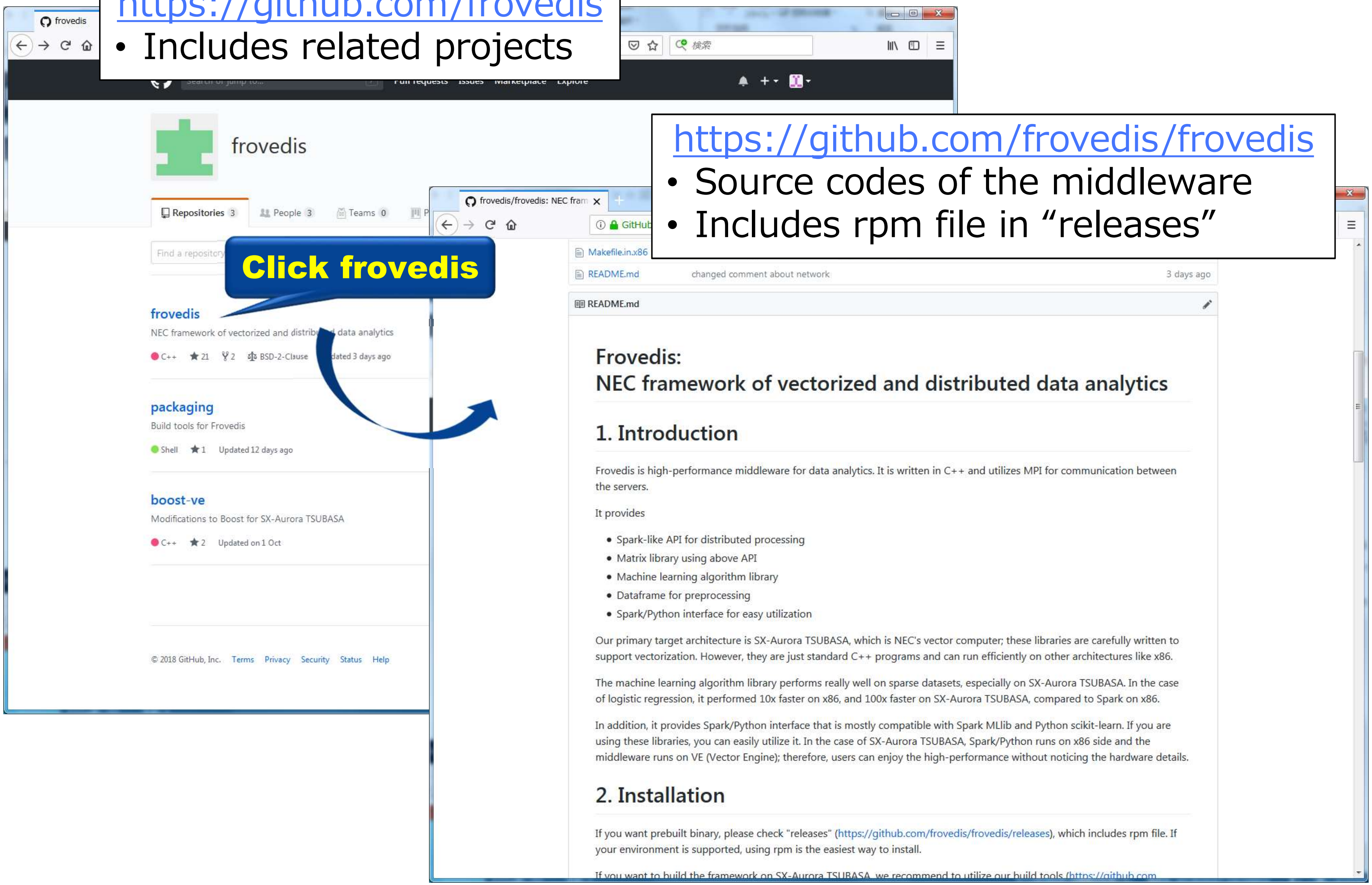
https://github.com/frovedis

- Includes related projects

https://github.com/frovedis/frovedis

- Source codes of the middleware
- Includes rpm file in "releases"

Click frovedis



Complete sample program

- Scatter a vector; double each element; then gather

```
#include <frovedis.hpp>
using namespace frovedis;

int two_times(int i) {return i*2;}

int main(int argc, char* argv[]) {
  use_frovedis use(argc, argv);

  std::vector<int> v = {1,2,3,4,5,6,7,8};
  dvector<int> d1 = make_dvector_scatter(v);
  dvector<int> d2 = d1.map(two_times);
  std::vector<int> r = d2.gather();
}
```

initialization

scatter to create dvector

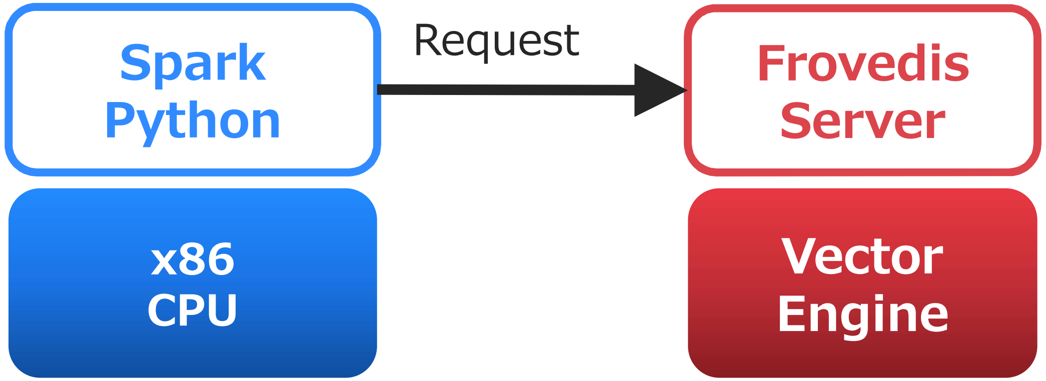
gather to std::vector

- Do not have to be aware of MPI (SPMD programming style)
  - Looks more like sequential program

Seamless interface from Spark/Python

- Writing C++ program is sometimes tedious, so we support seamless Spark/Python interface to utilize vector computer
  - Call the middleware through the same API (Spark MLlib / Python scikit-learn)
  - Users do not have to be aware of vector hardware

- Implementation: created a server with the functionalities
  - Receives RPC request from Spark and executes ML algorithm, etc.



How to use Spark Interface

Only need to modify importing module and add start/stop server

Original Spark program: logistic regression

```
…
import org.apache.spark.mllib.classification.LogisticRegressionWithSGD
…
val model = LogisticRegressionWithSGD.train(data)
…
```

Change to call Frovedis implementation

```
…
import com.nec.frovedis.mllib.glm.LogisticRegressionWithSGD  // change import
…
FrovedisServer.initialize(…)                          // invoke Frovedis Server
val model = LogisticRegressionWithSGD.train(data)     // no change: same API
FrovedisServer.shut_down()                            // stop Frovedis Server
…
```

Specify command to invoke Frovedis

Supported functionalities

- Matrix operations: both dense and sparse
- Data Frame (= table operation) for preprocessing
- Many statistical machine learning algorithms: still extending them

Machine Learning

Logistic Regression
Linear Regression
Lasso Regression
Ridge Regression
Linear SVM
SVD
EVD
PCA
Collaborative filtering(ALS)
Naive Bayes
K-means
Decision Tree
word2vec
Factorization Machines

Data Frame

Filter
Sort
Join
Group by

Basic matrix operations

Solve
Inverse
Gemv
SpMV
LU
Least square
Gemm
…

Backed by ScaLAPACK, LAPACK, BLAS

Collaboration with Hortonworks

- We have started collaboration with Hortonworks, which is one of the leading Hadoop/Spark distributors

HORTONWORKS®

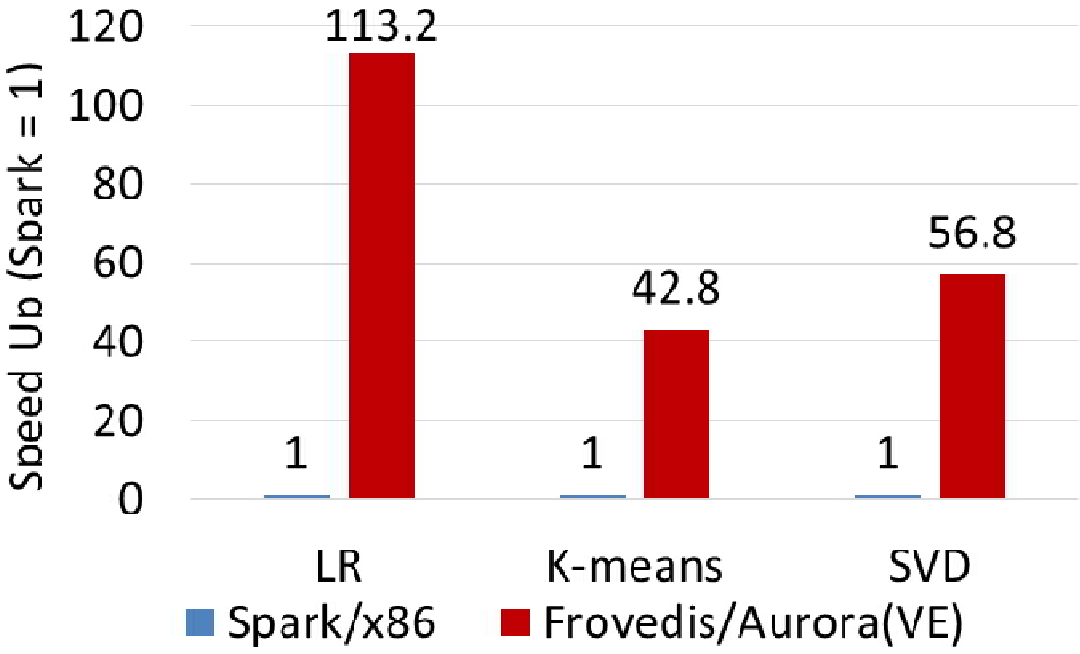
- YARN will be extended to support SX-Aurora TSUBASA
  - YARN is the resource manager of Hadoop cluster

- We will enhance the capabilities of NEC's distributed processing platform "Data Platform for Hadoop" by integrating SX-Aurora TSUBASA and Frovedis

https://www.nec.com/en/press/201810/global_20181015_01.html

Performance evaluation (1) Machine Learning

Frovedis on SX-Aurora TSUBASA shows 42x to 113x performance improvement from Spark MLlib

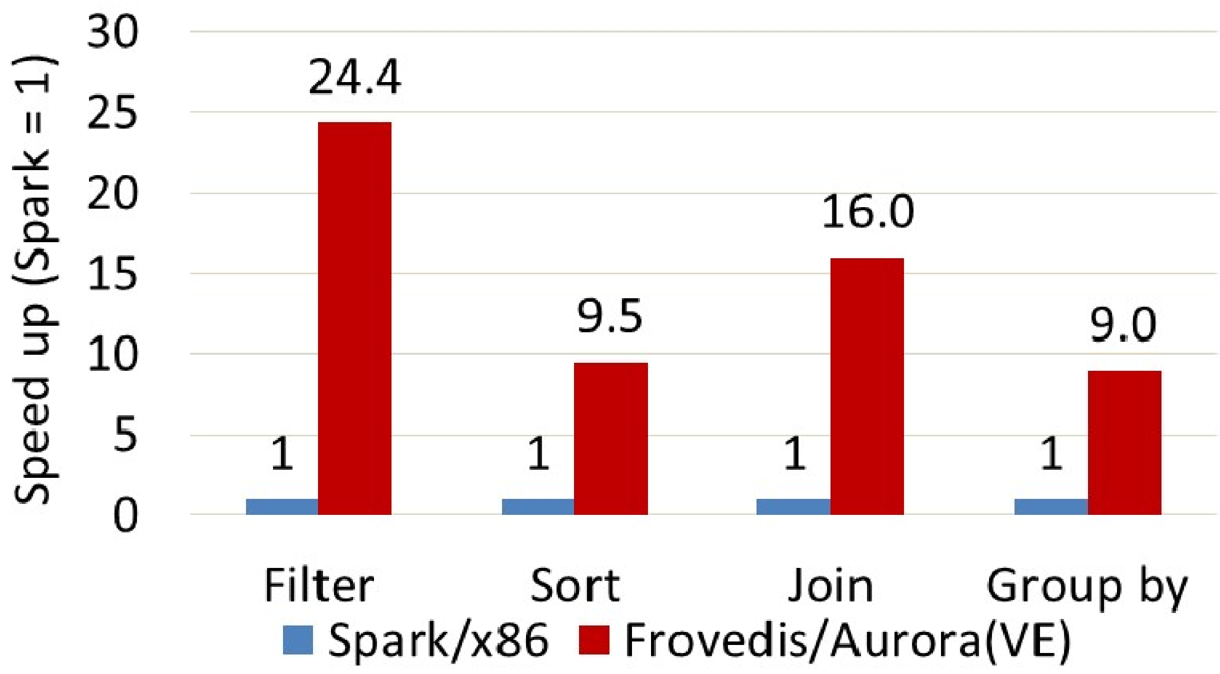


Xeon (Gold 6126) 1 socket vs 1 VE, with sparse data (w/o I/O)

- LR uses CTR data provided by Criteo (1/4 of the original, 6GB)
- K-means and SVD used Wikipedia doc-term matrix (10GB)

Performance evaluation (2) Data Frame

Frovedis on SX-Aurora TSUBASA shows 9x to 24x performance improvement from Spark Data Frame



Xeon (Gold 6126) 1 socket vs 1 VE

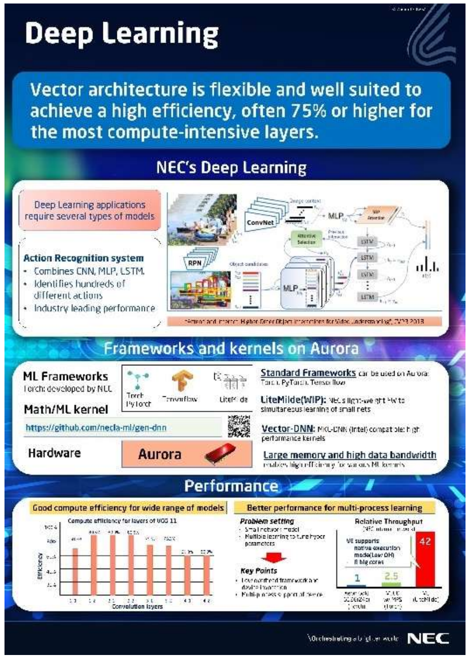
- Evaluated with part of TPC-H SF-100 data

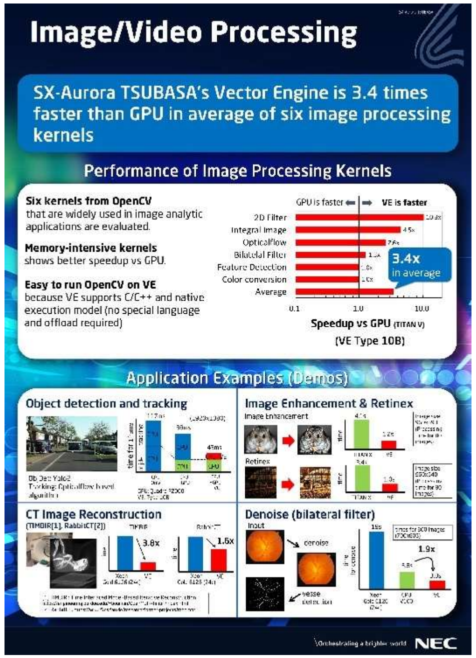
Conclusion

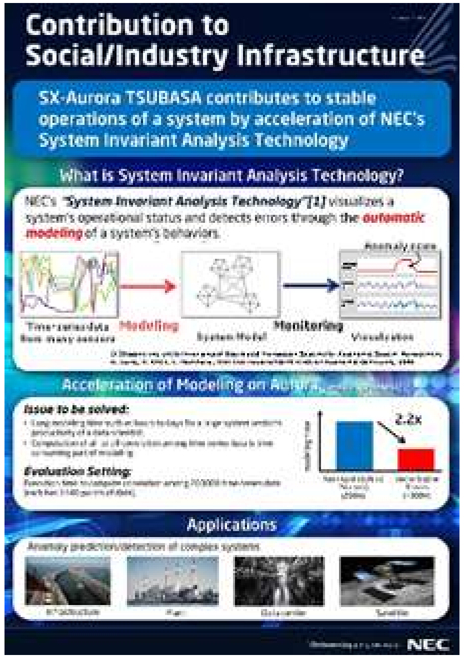
- We created middleware called Frovedis that speeds up statistical machine learning on SX-Aurora TSUBASA

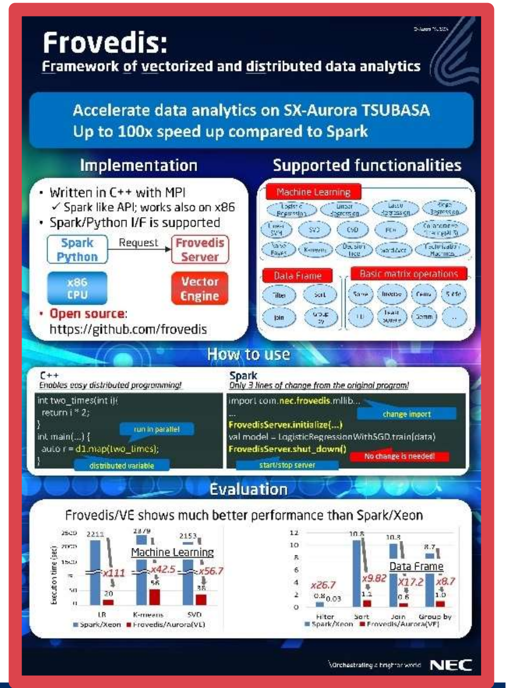
- Please visit our GitHub page! **https://github.com/frovedis**

- We are showing demos of AI applications of SX-Aurora TSUBASA in our booth. Please visit them also!









\Orchestrating a brighter world NEC

\Orchestrating a brighter world

NEC