Exploiting the OpenPOWER Platform for Big Data Analytics and Cognitive

Rajesh Bordawekar and Ruchir Puri
IBM T. J. Watson Research Center

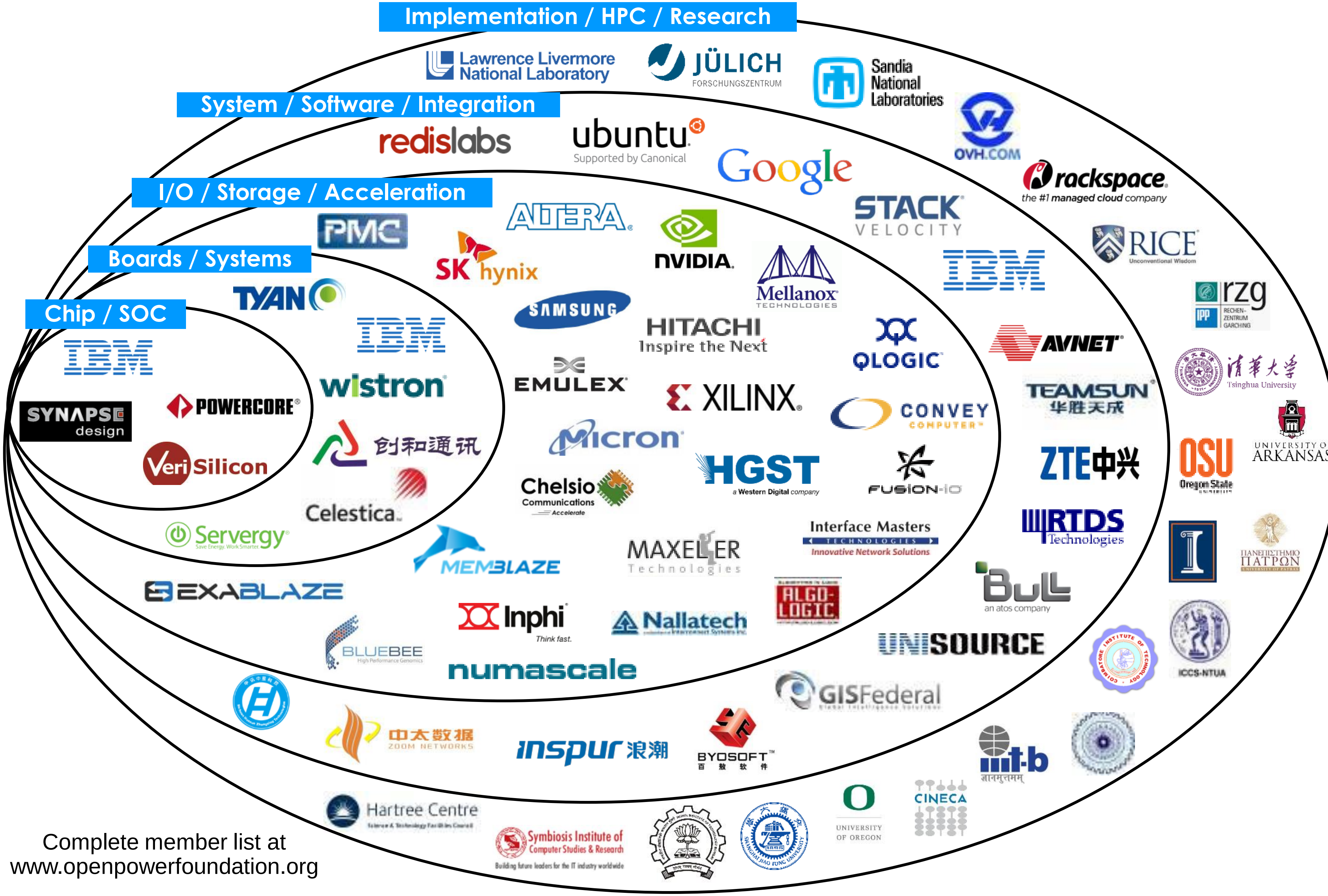
© 2014 IBM Corporation

Outline

- IBM OpenPower Platform
- Accelerating Analytics using GPUs
- Case Studies of GPU-accelerated workloads
  – Cognitive/Graph Analytics
  – In-memory OLAP
  – Deep Learning
  – Financial Modeling
- Summary

© 2015 IBM Corporation

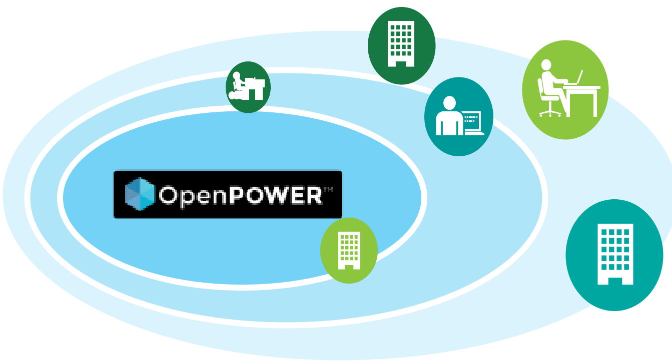
OpenPOWER: An Open Development Community

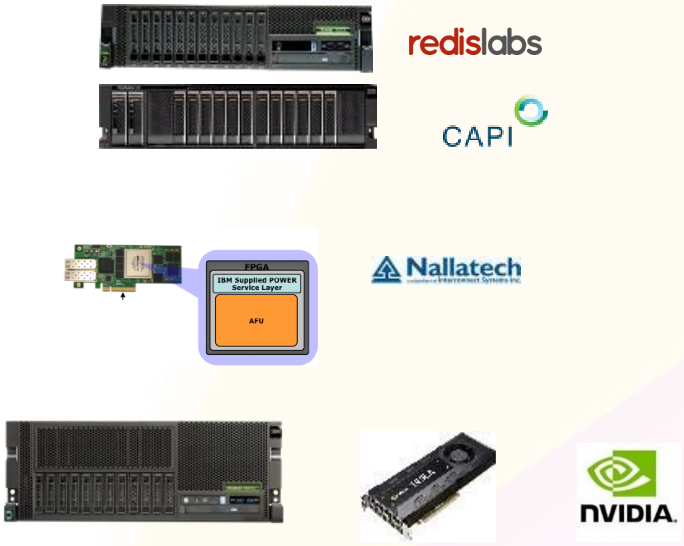


Complete member list at
www.openpowerfoundation.org

The OpenPOWER Foundation creates a pipeline of continued innovation and extends POWER8 capabilities

- Opening the architecture and innovating across the full hardware & software stack
- Driving an expansion of enterprise-class hardware and software
- Building a complete sever ecosystem delivering maximum client flexibility





More information at the OpenPower summit today and tomorrow

© 2014 IBM Corporation

IBM and NVIDIA deliver new acceleration capabilities for analytics, big data, and Java

- ✓ Runs pattern extraction analytic workloads 8x faster
- ✓ Provides new acceleration capability for analytics, big data, Java, and other technical computing workloads
- ✓ Delivers faster results and lower energy costs by accelerating processor intensive applications

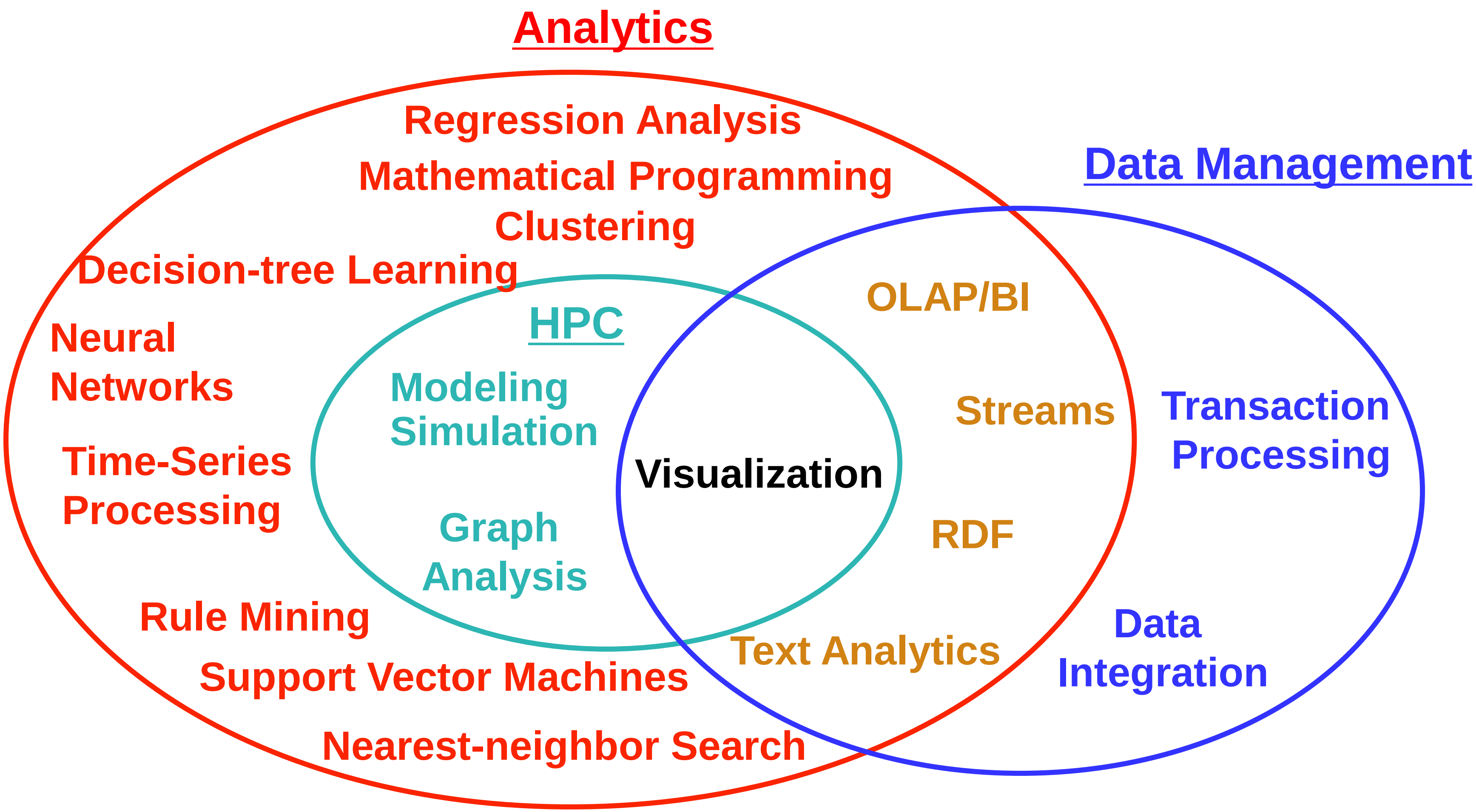
Power System S824L

- Up to 24 POWER8 cores
- Up to 1 TB of memory
- Up to 2 NVIDIA K40 GPU Accelerators
- Ubuntu Linux running bare metal

- **Power 8 provides high memory bandwidth and a large number of concurrent threads. Ideal for scaling memory-bound workloads such as OLTP and OLAP.**
- **GPUs provide high memory and compute bandwidth.**
- **Nvlink to provide very high host-to-GPU bandwidth leading to further performance Efficiency and simplified multi-GPU programming**
- **Together, Power 8+GPUs ideal of accelerating analytics, HPC and data Management workloads.**

© 2014 IBM Corporation

Analytics, HPC, and Data Management Landscape



© 2015 IBM Corporation

Acceleration Opportunities for GPUs

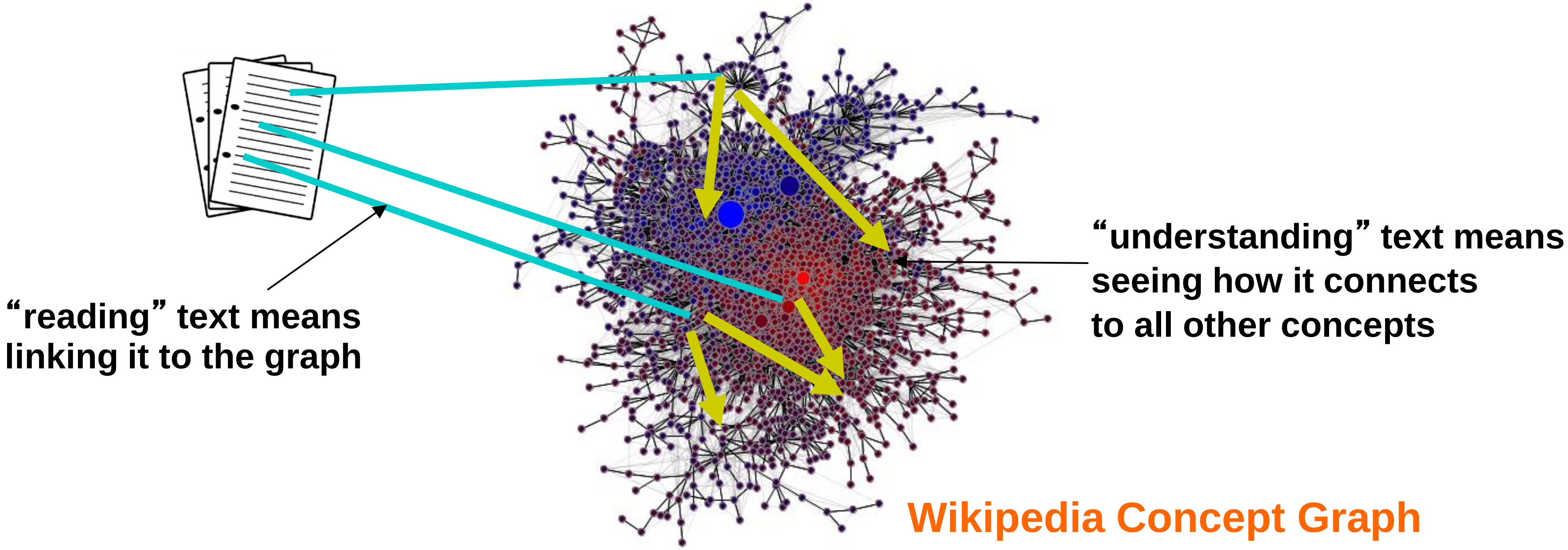
Analytics Model	Computational Patterns suitable for GPU Acceleration
Regression Analysis	Cholesky Factorization, Matrix Inversion, Transpose
Clustering	Cost-based iterative convergence
Nearest-neighbor Search	Distance calculations, Singular Value Decomposition, Hashing
Neural Networks	Matrix Multiplications, Convolutions, FFTs, Pair-wise dot-products
Support Vector Machines	Linear Solvers, Dot-product
Association Rule Mining	Set Operations: Intersection, union
Recommender Systems	Matrix Factorizations, Dot-product
Time-series Processing	FFT, Distance and Smoothing functions
Text Analytics	Matrix multiplication, factorization, Set operations, String computations, Distance functions, Pattern Matching
Monte Carlo Methods	Random number generators, Probability distribution generators
Mathematical Programming	Linear solvers, Dynamic Programming
OLAP/BI	Aggregation, Sorting, Hash-based grouping, User-defined functions
Graph Analytics	Matrix multiplications, Path traversals

© 2015 IBM Corporation

Case Study: Concept Graph Analytics

Human-like natural language understanding *via identifying* and *ranking related concepts* from massively crowd-sourced knowledge sources like Wikipedia.

A query for "GPU" would return documents first on GPUs, then on "multi-core" and "FPGAs" (GPUs, multi-cores, and FPGAs are related concepts).



Goal: To use GPUs to implement instantaneous concept discovery system that can scale to millions of documents

Concept Graph Analysis Basics

- Operates on a corpora of documents (e.g., webpages)
- Relationships in the external knowledge base represented as values in a N*N sparse matrix, where N is the total number of concepts
- Each document has a set of concepts that get mapped to sparse N-element vectors, one vector per concept
- Two core operations:
  - **Indexing** to relate concepts from current document corpus to the knowledge base (Throughput-oriented operation)
  - **Real-time querying** to relate few concepts generated by the queries (e.g., gpu, multi-core, FPGA) to documents (Latency-sensitive operation)

 © 2015 IBM Corporation

Implementation of Concept Graph Analysis

- Core computation: Markov-chain based iterative personalized page rank algorithm to calculate relationships between concepts of a knowledge graph
  - Implemented as a sparse matrix-dense matrix multiplication operation (**cuSPARSE csrmm2**) over a sparse matrix representing the concept graph and a dense matrix representing the concepts from document
  - Usually requires 5 iterations to converge
- Indexing involves multiplication over a large number of concept vectors, querying involves multiplication over a small number of concept vectors. **GPUs suitable for accelerating indexing.**
- On a wikipedia knowledge graph, the sequential indexing execution requires around **59 sec**. The GPU implementation takes around **2.31 sec**

  © 2015 IBM Corporation

Case Studies: Deep Learning

- Deep learning usually exploits neural network architectures such as multiple layers of fully-connected networks (called deep neural networks or DNNs) or convolution neural networks (CNNs)
  - DNNs very good at classification
  - CNNs very good at translation-invariant feature extraction
- Both approaches heavily compute-bound and use extensive use of matrix multiplications
- Our focus on exploiting GPUs for accelerating speech-to-text processing using a joint CNN-DNN architecture
  - Our system uses a "native" implementation of CNN and DNN operations using only cuBLAS functions
  - **Performance competitive to (sometimes better than) cuDNN**
- Talk on this work on Friday 9 am, 210A (S5231)

© 2015 IBM Corporation

Case Studies: In-memory Relational OLAP

- Relational OLAP characterized by both compute-bound and memory-bound operations
  - Aggregation operations compute-bound
  - Join, Grouping, and Ordering memory-bound
    - Query optimizers use either sort or hash based approaches
- Both sorting and hashing can exploit GPU's high memory bandwidth
  - Data needs to fit in the device memory
- Two projects
  - GPU-optimized new hash table (GTC 2014)
  - Exploiting GPU-accelerated hashing for OLAP group-by operations in DB2 BLU (Talk and demo on Tuesday S5835 and S5229)

© 2015 IBM Corporation

Case Studies: In-memory Multi-dimensional OLAP

- Sparse data, representing values (measures) of multi-dimensional attributes (dimensions) with complex hierarchies (e.g., IBM Cognos TM1)
- Logically viewed as a multi-dimensional cube (MOLAP) accesses by a non-SQL query language
  - Unlike relational OLAP, MOLAP characterized by non-contiguous memory accesses similar to accessing multi-dimensional arrays
  - Most queries involve aggregation computations
- GPU-accelerated MOLAP prototype implemented
  - Up to 4X improvement over multi-threaded SIMDized execution

© 2015 IBM Corporation

Case Studies: Financial Modeling via Monte Carlo Simulation

- Monte Carlo simulation extensively used in financial modeling
  - Used for pricing esoteric options, when there is no analytical solution. Typically 10-20% of pricing functions in a portfolio
- Low I/O- High Compute Workload: suitable for accelerators such as FPGA and GPUs
- Key computational functions
  - Random number generators (e.g., Sobol)
  - Generating probability distribution (e.g., Inverse Normal)
  - Pricing functions
- Current focus on exploiting low-power GPUs
  - Talk on this work on Thursday 10.30 am, 210C (S5227)

© 2015 IBM Corporation

Other Analytics Opportunities

- Sparse Matrix Factorization
  - Watson Sparse Matrix Package on GPU (Presentation S5232 on Thursday, 9 am, 210G)
- Streaming Analytics
  - GPUs to accelerate functions on data streams (e.g., aggregation, Join)
- Graph Analytics
  - Identification of Concurrent Cycles in a fMRI image graph
- Mathematical Programming
  - Accelerating Linear Programming Libraries

© 2015 IBM Corporation

Summary

- OpenPower consortium creates an open community to spur innovation using Power-based platforms
- Demonstrated advantages of GPU via different analytics workloads
  – cognitive computing, OLAP, deep learning, financial modeling
- Power 8 and GPU system ideal for accelerating analytics, HPC, and data management workloads

© 2015 IBM Corporation