Why GPUs for Data Science?
Numerous hardware advantages

- Thousands of cores with up to ~20 TeraFlops of general purpose compute performance
- Up to 1.5 TB/s of memory bandwidth
- Hardware interconnects for up to 600 GB/s bidirectional GPU <--> GPU bandwidth
- Can scale up to 16x GPUs in a single node

Almost never run out of compute relative to memory bandwidth!
What is RAPIDS?
End-to-End GPU Accelerated Data Science

Data Preparation/ETL

- cuDF
  - GPU-accelerated ETL functions
  - Tracks Pandas and other common PyData APIs
  - Dask + UCX integration for scaling

Analytics/ML/Graph

- RAPIDS ML
  - GPU-native cuML library, plus XGBoost, FIL, HPO, and more
  - cuGraph
    - GPU graph analytics, including TSP, PageRank, and more

- cuGraph

- cuxfilter
  - GPU-accelerated cross-filtering
  - pyViz integration
    - Plotly Dash, Bokeh, Datashader, HoloViews, hvPlot

Visualization

Domain-Specific Libraries

- CLX + Morpheus
  - Cyber log processing + anomaly detection
- cuStreamz
  - Streaming analytics
- cuSignal
  - Signals processing
- cuCIM
  - Computer vision & image processing primitives
- cuSpatial
  - Spatial analytics
- node-RAPIDS
  - Bindings for node.js

...and more!
Overview of Changes: RAPIDS 21.10 Release

- **RAPIDS**  CUDA 11.4 now supported by RAPIDS
- **RAPIDS+Dask** Support for LocalCUDACluster with MIG; UCX 1.11.1 support
- **cuDF** Map support for ORC Reader; struct support for ORC write; Time series enhancements including support for rolling and grouped rolling variance and standard deviation, groupby first and last aggregations, and datetime manipulations like ceil and days_in_month;
- **cuML** New Categorical and Gaussian Naive Bayes models; categorical features support in FIL; improvements to ARIMA, Random Forest and HDBSCAN; new distances added to pairwise_distances; 2d kNN acceleration via RBC
- **cuGraph** New Sorensen coefficient and weighted Sorensen coefficient features; Biased Random Walks sampling for Graph Neural Networks support added to libcugraph; Improved graph primitives for better memory scaling and multi-node multi-GPU memory footprint improvements for low average vertex degree graphs; Multi-seed BFS, one seed per component, added.
- **CLX** Maintenance to existing code; DGA extended sequence handling; additional notebooks for new use cases
- **cuCIM** Support Runtime Context for CuFileDriver and Culmage; Support raw RGB tiled TIFF
cuDF Updates: Deep Dive

Features added in 21.10

- Map support for ORC Reader and struct support for ORC writer
- Struct support for `drop_list_duplicates` in libcuDF
- LibcuDF added `interleave_columns` function for struct and lists
- Rolling and grouped rolling variance and standard deviation, and groupby first and last aggregations
- Linear interpolation for filling missing values
- Computing t-digests and approximate percentiles

Planned Upcoming Features

- Expanded support for additional decimal types
- Nested type support for JSON reader
- Enhanced GPUDirect Storage integration
- Map support for ORC writer
- Refactored hash join implementation
cuML Updates: Deep Dive
Release 21.10

Features added in 21.10
- Categorical features support in FIL
- Single-GPU implementation of Categorical and Gaussian Naive Bayes algorithm
- 2-Dimensional Random Ball Cover algorithm for speeding exact Nearest Neighbors
- Added support for hamming, jensen-shannon, kl-divergence, correlation and russellrao distances for pairwise distance calculations
- Support for missing observations, padding and exogenous variables for ARIMA
- Multiple improvements to Random Forest to improve accuracy and performance.

Planned Upcoming Features
- Support for exogenous variables in ARIMA and updates for confidence intervals and Kalman filter
- Exposing KL divergence in TSNE
- Optimized Linear SVM
cuGraph Updates: Deep Dive
Release 21.10

Features added in 21.10

▸ Single-GPU implementation of Sorensen coefficient and weighted Sorensen similarity coefficient.
▸ Added Biased Random Walks for Graph Neural Networks in libcuGraph
▸ Improved graph primitives to support better memory scaling, as well multi-node multi-GPU memory footprint improvements for low average vertex degree graphs
▸ New end-to-end benchmarking scripts
▸ Resolve tech debt and enhance the library
▸ Multi-seed BFS, allow one seed per connected component, add

Planned Upcoming Features

▸ Multi-GPU graph primitives for Triangle Counting, Symmetrize, and Transpose
▸ Multi-GPU HITS
▸ SG node2vec graph sampling
Join the Conversation

GOOGLE GROUPS
https://groups.google.com/forum/#!forum/rapidsai

DOCKER HUB
https://hub.docker.com/r/rapidsai/rapidsai

SLACK CHANNEL
https://rapids-goai.slack.com/join

STACK OVERFLOW
https://stackoverflow.com/tags/rapids
THANK YOU

@RAPIDSai