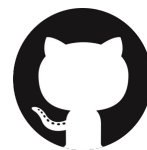


# RAPIDS

21.06 Release



[@RAPIDSai](https://twitter.com/RAPIDSai)



<https://github.com/rapidsai>



<https://rapids-goai.slack.com/join>

**RAPIDS**

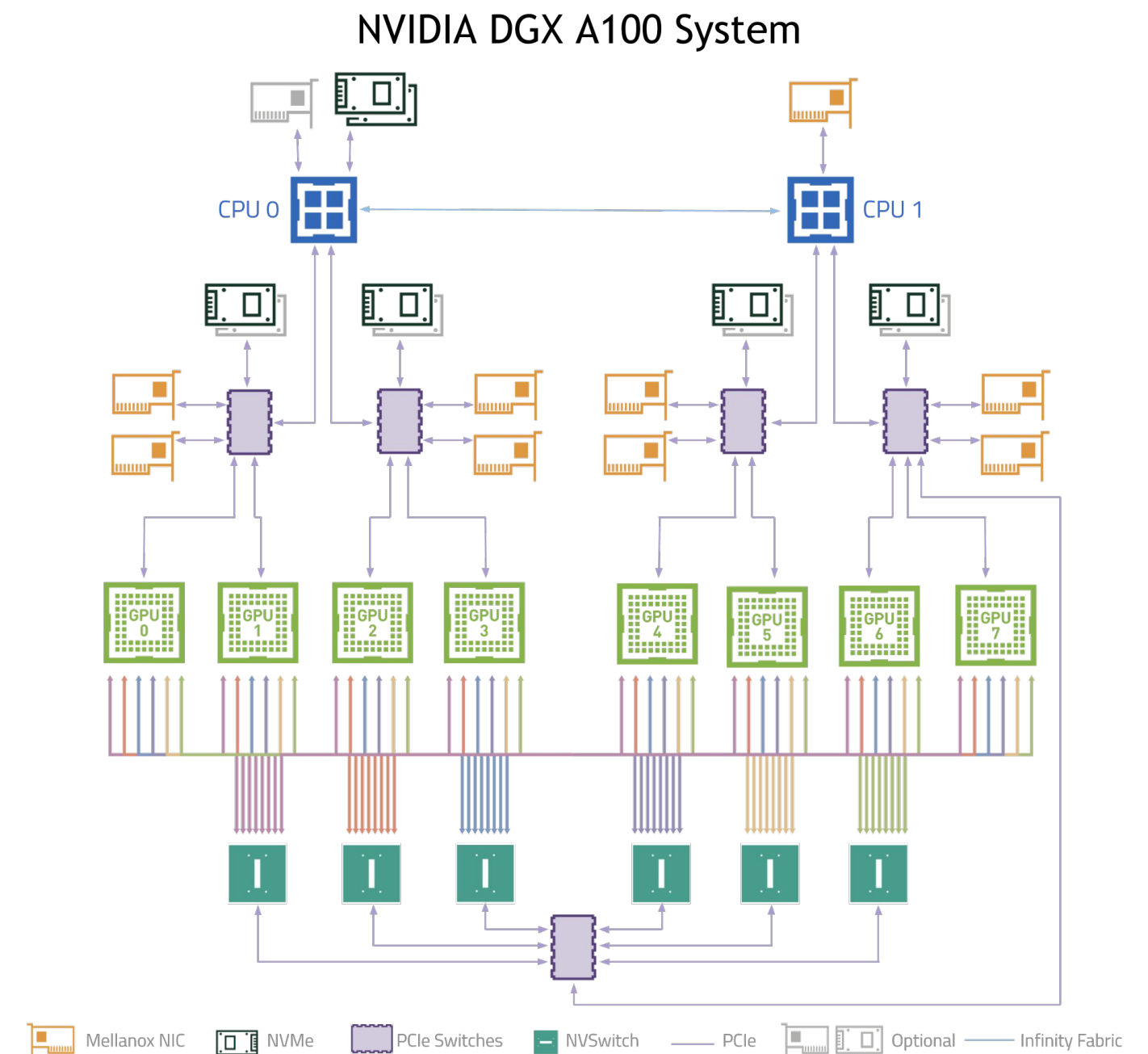
<https://rapids.ai>

# Why GPUs for Data Science?

## Numerous hardware advantages

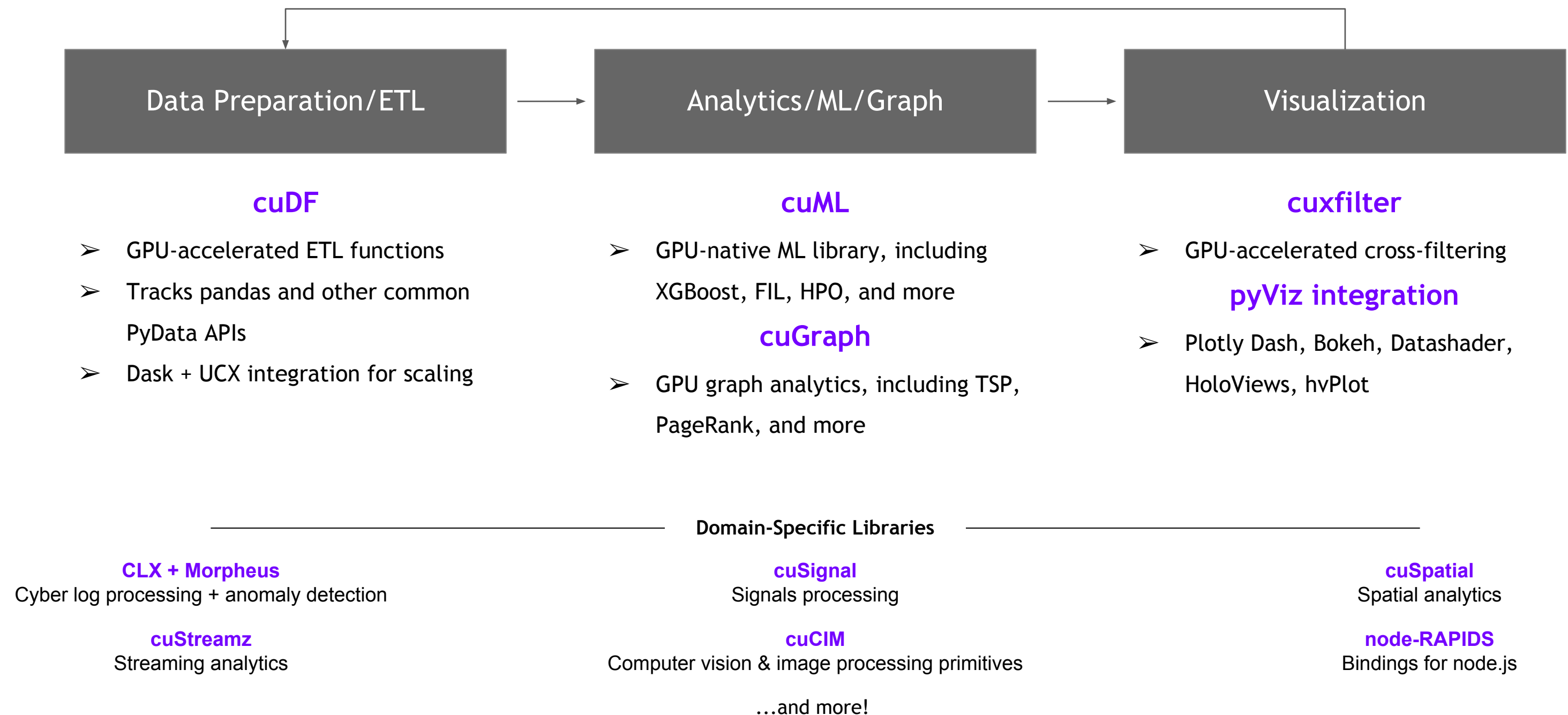
- ▶ Thousands of cores with up to ~20 TeraFlops of general purpose compute performance
- ▶ Up to 1.5 TB/s of memory bandwidth
- ▶ Hardware interconnects for up to 600 GB/s bidirectional GPU <--> GPU bandwidth
- ▶ Can scale up to 16x GPUs in a single node

Almost never run out of compute relative to memory bandwidth!



# What is RAPIDS?

## End-to-End GPU Accelerated Data Science



# Overview of Changes: RAPIDS 21.06 Release

- ▶ **RAPIDS+Dask** Updated memory spilling (JIT\_UNSPILL) to use compatibility mode; added capability to benchmarks; added support for CUDA Async Memory Resource; UCX usability improvements including clean endpoint shutdown and documentation;
- ▶ **cuDF** Additional support for fixed-point decimal types in Python; more groupby and join aggregations; more functionalities for list and string type operations in Python; expanded dictionary type operations in C++;
- ▶ **cuML** New HDBSCAN model; major improvements to Random Forest; single-linkage hierarchical clustering; sample weights for logistic regression; major improvements to build system;
- ▶ **XGBoost** 1.4.2 ships with 21.06, including improvements to Dask integration and prediction functions;
- ▶ **cuGraph** Single and Multi GPU Weakly Connected Components; graph batching for C++; multi-column support; Bipartite graph structures; continued improving graph primitives for performance
- ▶ **CLX** URL processing for DGA detection; added Multiclass Sequence Classifier; enhanced Lightweight Online Detector of Anomalies
- ▶ **cuCIM** Multiple GPU accelerated computer vision and image processing primitives for N dimensional images
- ▶ **Morpheus** using RAPIDS libraries; early access available on June 15<sup>th</sup> 2021

# cuDF Updates: Deep Dive

## Release 21.06

### Features added in 21.06

- ▶ Decimal data type is now supported for *csv* and *orc writer* functions in Python
- ▶ *Shift* and multiple cumulative operations for *groupby* aggregation are now available
- ▶ Added *getitem* and *concatenate\_list\_elements* functions to list in C++
- ▶ Continuing refactoring and improving string support

### Planned Upcoming Features

- ▶ Conditional Joins
- ▶ Decimal support for additional cuIO functions
- ▶ ORC GDS and struct support
- ▶ Update to Arrow 4.0.1
- ▶ Support multiple inputs in json reader

# cuML Updates: Deep Dive

## Release 21.06

### Features added in 21.06

- ▶ New *Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN)* algorithm
- ▶ *Fast Fourier Transform* accelerated *t*-Stochastic Neighborhood Embedding (*Flt-SNE*)
- ▶ New backend to use cuML's *Forest Inference Library (FIL)* in the NVIDIA Triton Inference Server.
- ▶ Added *ColumnTransformer* and *FunctionTransformer* functions to cuML's preprocessing module
- ▶ New *Random Forest* backend is the default both classification and regression models - provides better performance and accuracy
- ▶ Many model-specific improvements! *ARIMA* memory improvements, deterministic mode for UMAP, sample weights for logistic regression, ....

### Planned Upcoming Features

- ▶ Further enhancements for *Random Forest*, *Forest Inference Library* and *HDBSCAN*
- ▶ *Sparse Logistic Regression*
- ▶ Weighted multi-node multi-gpu *KMeans*

# cuGraph Updates: Deep Dive

## Release 21.06

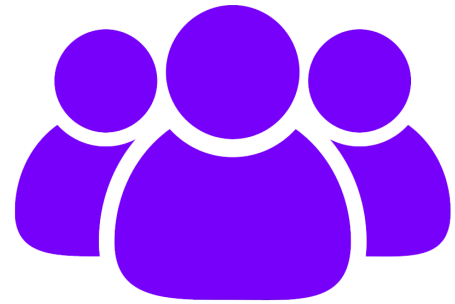
### Features added in 21.06

- ▶ New Multi GPU implementation of [Weakly Connected Components](#)
- ▶ *Graph Batching* functionality added to libcugraph
- ▶ Multi-column support for all cuGraph algorithms
- ▶ Bipartite graph structures are now supported in cuGraph
- ▶ New padded result set for Random Walk

### Planned Upcoming Features

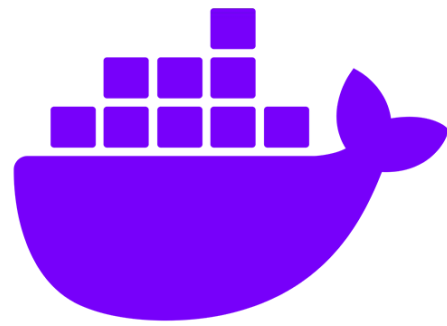
- ▶ Depth limit functionality on traversal algorithms
- ▶ *Batch Random Walk*
- ▶ *Multi-Seed Breadth First Search*
- ▶ *Aggregated Traversal Graph Prim*

# Join the Conversation



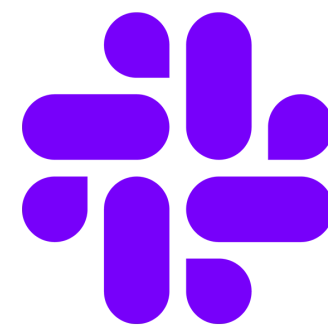
## GOOGLE GROUPS

<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



RAPIDS