Then and Now – Growing as a child of ECP

EXASCALE COMPUTING PROJECT

Approved for public release

Hartwig Anzt, University of Tennessee

National Nuclear Security Administration

U.S. DEPARTMENT OF ENERGY | Office of Science

“Then” - Before ECP

- 2013-16, I am PostDoc in Jack Dongarra’s ICL at UTK and lucky to be part of a collaboration project with SNL exploring the development of sparse linear algebra for NVIDIA GPUs.

INNOVATIVE COMPUTING LABORATORY

MAGMA SPARSE

MAGMA-sparse as a “child” of MAGMA explores the development of sparse linear algebra for NVIDIA GPUs.

+ Plenty of research freedom
+ Plenty of opportunities to publish
+ No deliverables or deadlines

- Uncertain funding future
- Writing software is not the main scope
- *“who cares what I am doing”*

Extreme-scale Algorithms & Solver Resilience (EASIR)

Architecture-aware Algorithms for Scalable Performance and Resilience on Heterogeneous Architectures

“Then” - Before ECP

- 2013-16, I am PostDoc in Jack Dongarra’s ICL at UTK and lucky to be part of a collaboration project with SNL exploring the development of sparse linear algebra for NVIDIA GPUs.

- In 2015/16, funding for software development is uncertain and I apply for an early career grant in Germany *and* participate in a proposal for the US Exascale Computing Project.

Extreme-scale Algorithms & Solver Resilience (EASIR)

Architecture-aware Algorithms for Scalable Performance and Resilience on Heterogeneous Architectures

“Then” - Before ECP

- 2013-16, I am PostDoc in Jack Dongarra’s ICL at UTK and lucky to be part of a collaboration project with SNL exploring the development of sparse linear algebra for NVIDIA GPUs.
- In 2015/16, funding for software development is uncertain and I apply for an early career grant in Germany *and* participate in a proposal for the US Exascale Computing Project.
- In 2016, I receive the early career award *and* ECP becomes reality.

I am torn between the options and chose: Both.

INNOVATIVE COMPUTING LABORATORY

Extreme-scale Algorithms & Solver Resilience (EASIR)

Architecture-aware Algorithms for Scalable Performance and Resilience on Heterogeneous Architectures

EXASCALE COMPUTING PROJECT

"Then" - Before ECP

- 2013-16, I am PostDoc in Jack Dongarra's ICL at UTK and lucky to be part of a collaboration project with SNL exploring the development of sparse linear algebra for NVIDIA GPUs.

- In 2015/16, funding for software development is uncertain and I apply for an early career grant in Germany *and* participate in a proposal for the US Exascale Computing Project.

- In 2016, I receive the early career award *and* ECP becomes reality.

I am torn between the options and chose: Both.

And I start a new sparse linear algebra library:

Ginkgo

Extreme-scale Algorithms & Solver Resilience (EASIR)

Architecture-aware Algorithms for Scalable Performance and Resilience on Heterogeneous Architectures

EXASCALE COMPUTING PROJECT

The Design of an ECP Math Library

MAGMA SPARSE

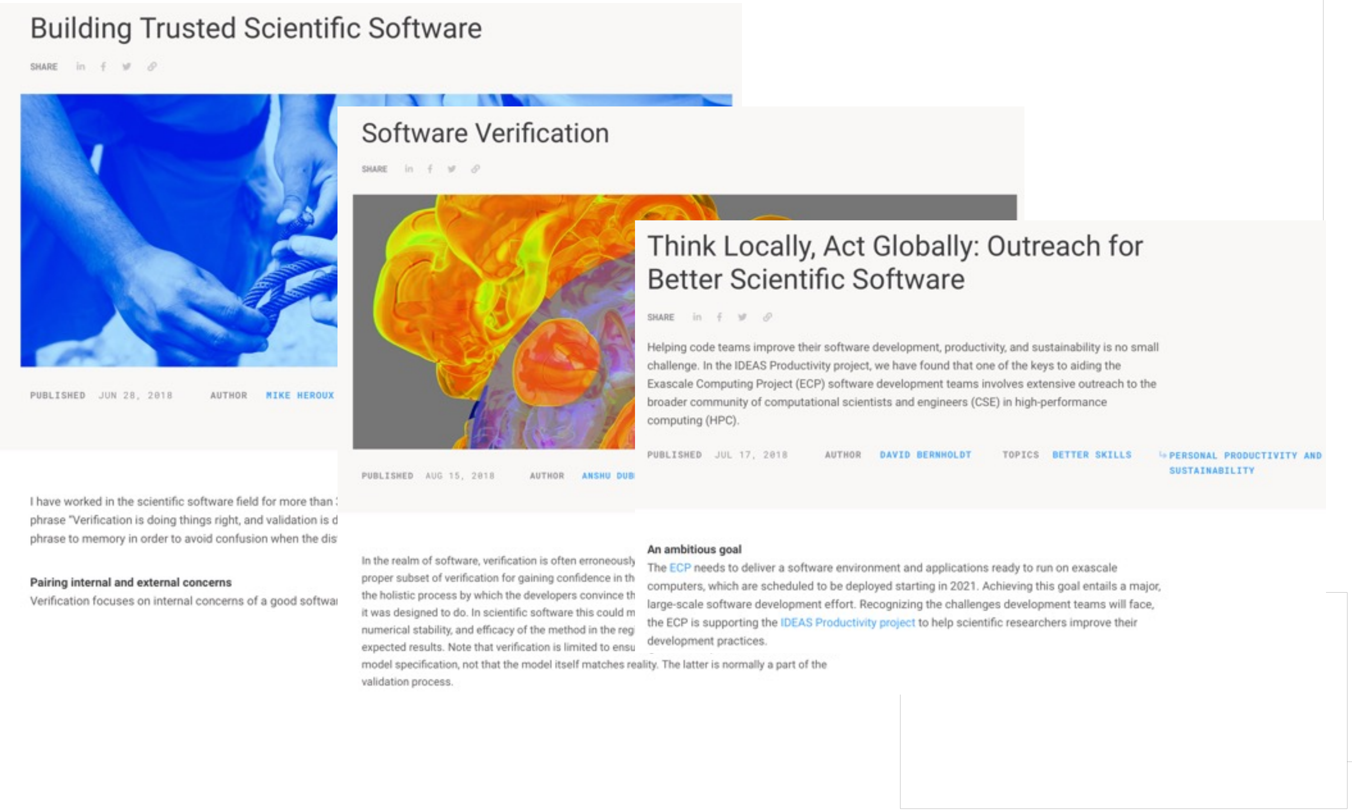
MAGMA-sparse as a “child” of MAGMA explores the development of sparse linear algebra for NVIDIA GPUs.

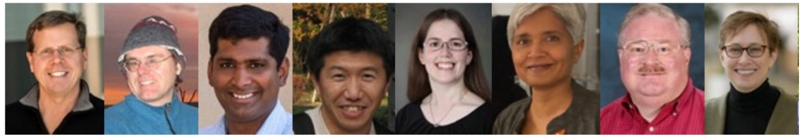
Design considerations for Ginkgo

- Platform Portability
- Performance
- Rapid integration of new algorithms
- xSDK / E4S Community Policies
- BSSw expertise / experience
- Modern C++
- CI/CD and unit testing
- Open source & permissive licensing

IDEAS productivity

better scientific software





The Design of an ECP Math Library

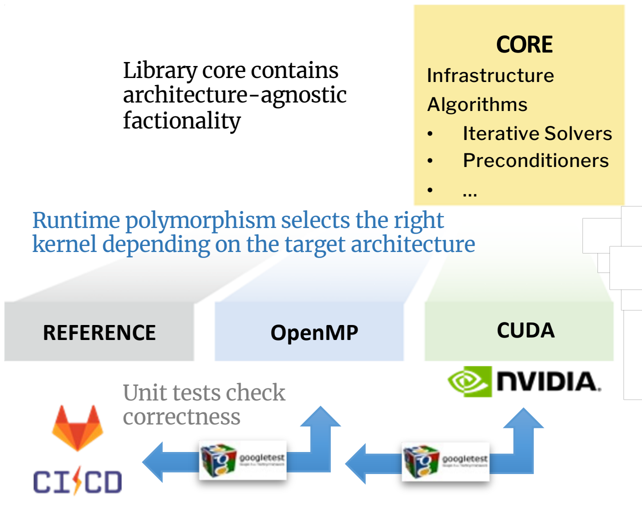
MAGMA SPARSE

MAGMA-sparse as a “child” of MAGMA explores the development of sparse linear algebra for NVIDIA GPUs.

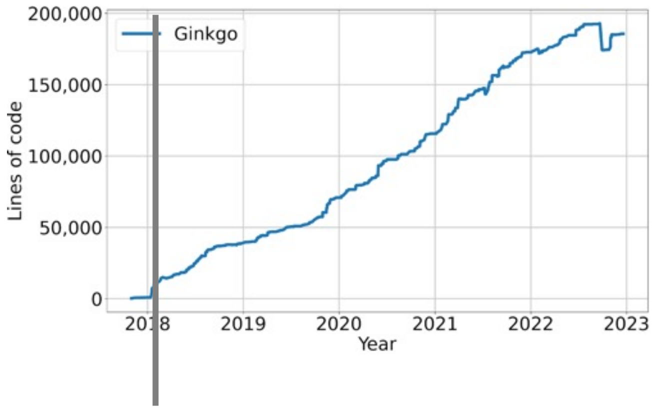
Design considerations for Ginkgo

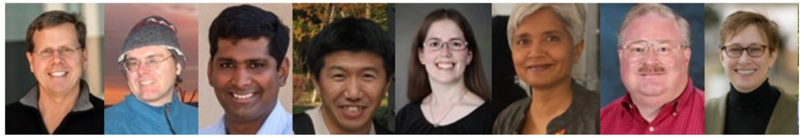
- Platform Portability
- Performance
- Rapid integration of new algorithms
- xSDK / E4S Community Policies
- BSSw expertise / experience
- Modern C++
- CI/CD and unit testing
- Open source & permissive licensing

Before the first line of code is written, we spend a year on whiteboard discussions.



Ginkgo





The Design of an ECP Math Library

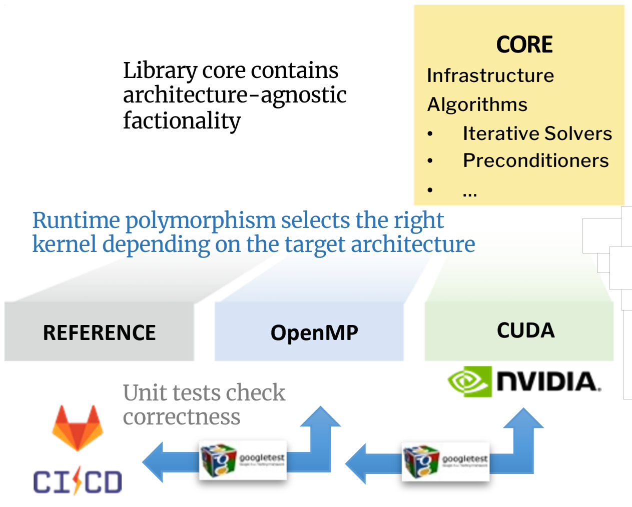
MAGMA SPARSE

MAGMA-sparse as a “child” of MAGMA explores the development of sparse linear algebra for NVIDIA GPUs.

Design considerations for Ginkgo

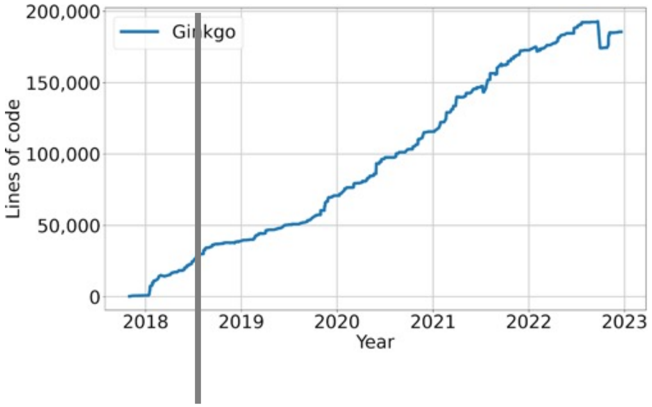
- Platform Portability
- Performance
- Rapid integration of new algorithms
- xSDK / E4S Community Policies
- BSSw expertise / experience
- Modern C++
- CI/CD and unit testing
- Open source & permissive licensing

Before the first line of code is written, we spend a year on whiteboard discussions.

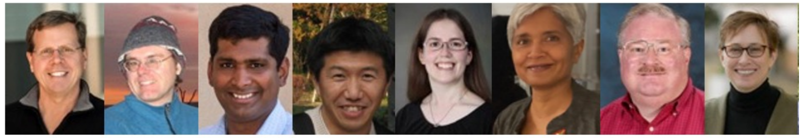


Ginkgo

	Functionality	OMP	CUDA
Basic	SpMV	✓	✓
	SpMM	✓	✓
	SpGeMM	✓	✓
Krylov solvers	BiCG	✓	✓
	BiCGSTAB	✓	✓
	CG	✓	✓
	CGS	✓	✓
	GMRES	✓	✓
	IDR	✓	✓
Preconditioners	(Block-)Jacobi	✓	✓
	ILU/IC		✓
	Parallel ILU/IC	✓	✓
	Parallel ILUT/ICT	✓	✓
	Sparse Approximate Inverse	✓	✓



EXASCALE COMPUTING PROJECT



The Design of an ECP Math Library

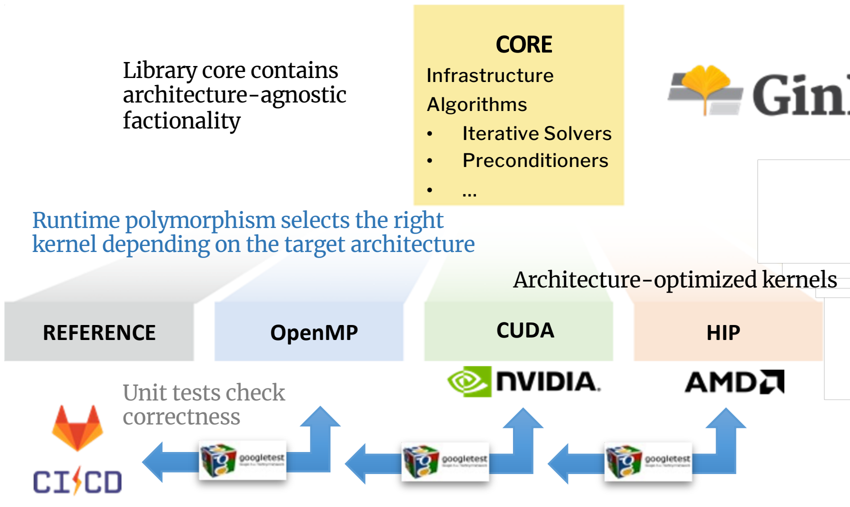
MAGMA SPARSE

MAGMA-sparse as a “child” of MAGMA explores the development of sparse linear algebra for NVIDIA GPUs.

Design considerations for Ginkgo

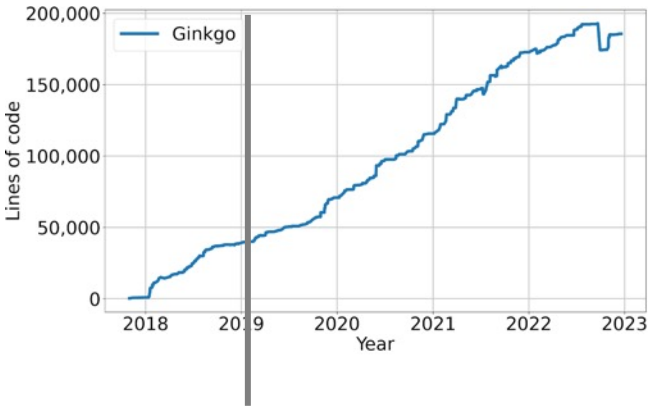
- Platform Portability
- Performance
- Rapid integration of new algorithms
- xSDK / E4S Community Policies
- BSSw expertise / experience
- Modern C++
- CI/CD and unit testing
- Open source & permissive licensing

Before the first line of code is written, we spend a year on whiteboard discussions.



Ginkgo

	Functionality	OMP	CUDA	HIP
Basic	SpMV	✓	✓	✓
	SpMM	✓	✓	✓
	SpGeMM	✓	✓	✓
Krylov solvers	BiCG	✓	✓	✓
	BiCGSTAB	✓	✓	✓
	CG	✓	✓	✓
	CGS	✓	✓	✓
	GMRES	✓	✓	✓
	IDR	✓	✓	✓
Preconditioners	(Block-)Jacobi	✓	✓	✓
	ILU/IC		✓	✓
	Parallel ILU/IC	✓	✓	✓
	Parallel ILUT/ICT	✓	✓	✓
	Sparse Approximate Inverse	✓	✓	✓



Spack xSDK E4S

The Design of an ECP Math Library

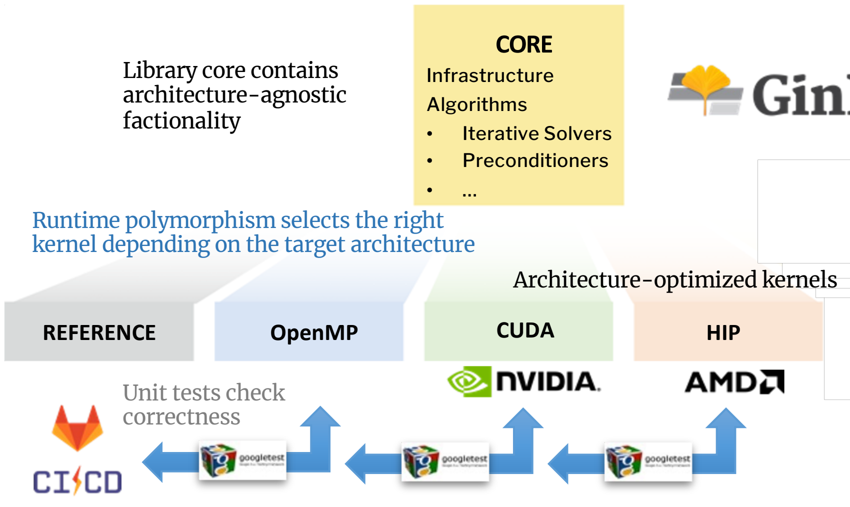
MAGMA SPARSE

MAGMA-sparse as a "child" of MAGMA explores the development of sparse linear algebra for NVIDIA GPUs.

Design considerations for Ginkgo

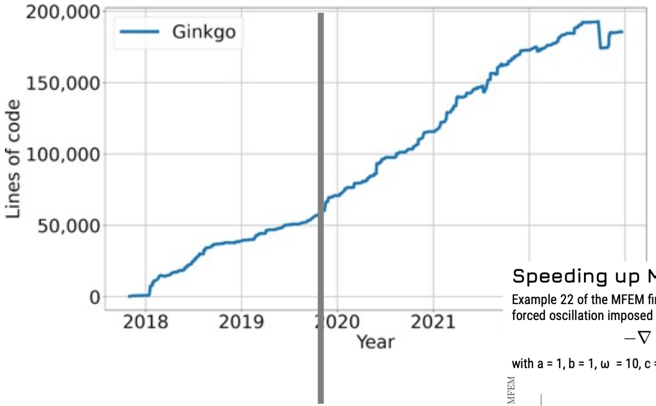
- Platform Portability
- Performance
- Rapid integration of new algorithms
- xSDK / E4S Community Policies
- BSSw expertise / experience
- Modern C++
- CI/CD and unit testing
- Open source & permissive licensing

Before the first line of code is written, we spend a year on whiteboard discussions.



Ginkgo

	Functionality	OMP	CUDA	HIP
Basic	SpMV	✓	✓	✓
	SpMM	✓	✓	✓
	SpGeMM	✓	✓	✓
Krylov solvers	BiCG	✓	✓	✓
	BiCGSTAB	✓	✓	✓
	CG	✓	✓	✓
	CGS	✓	✓	✓
	GMRES	✓	✓	✓
	IDR	✓	✓	✓
Preconditioners	(Block-)Jacobi	✓	✓	✓
	ILU/IC		✓	✓
	Parallel ILU/IC	✓	✓	✓
	Parallel ILUT/ICT	✓	✓	✓
	Sparse Approximate Inverse	✓	✓	✓
Utilities	On-Device Matrix Assembly	✓	✓	✓
	MC64/RCM reordering	✓		
	Wrapping user data		✓	



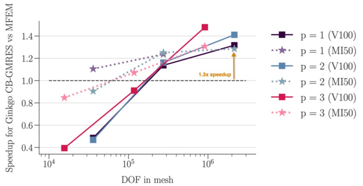
Spack xSDK E4S

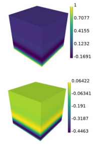
Speeding up MFEM's "example 22" on GPUs

Example 22 of the MFEM finite element library solves harmonic oscillation problems, with a forced oscillation imposed at the boundary. In this test, we use variant 1:

$$-\nabla \cdot (a\nabla u) - \omega^2 bu + i\omega cu = 0$$

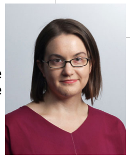
with a = 1, b = 1, ω = 10, c = 20





Real part of solution (top), imaginary part of solution

Speedup of Ginkgo's Compressed Basis-GMRES solver vs MFEM's GMRES solver for three different orders of basis functions (p), using MFEM matrix-free operators and the Ginkgo-MFEM integration wrappers in MFEM. CUDA 10.1/V100 and ROcm 4.0/MI50.



EXASCALE COMPUTING PROJECT

The Design of an ECP Math Library

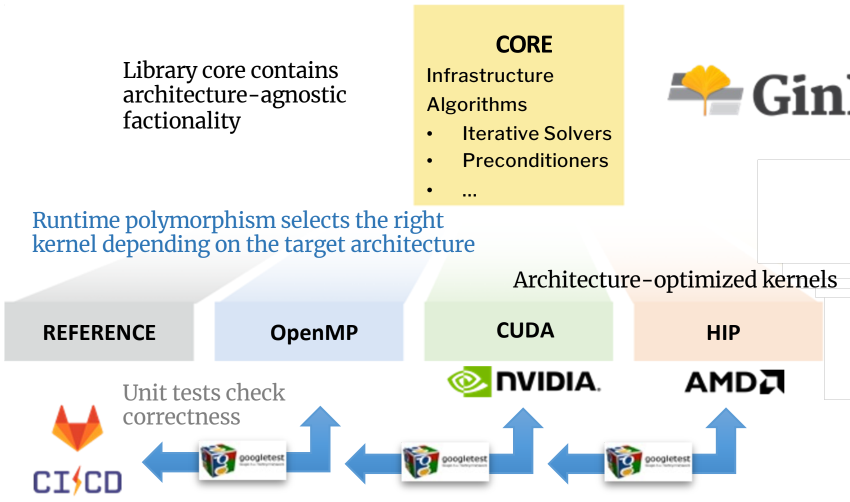
MAGMA SPARSE

MAGMA-sparse as a “child” of MAGMA explores the development of sparse linear algebra for NVIDIA GPUs.

Design considerations for Ginkgo

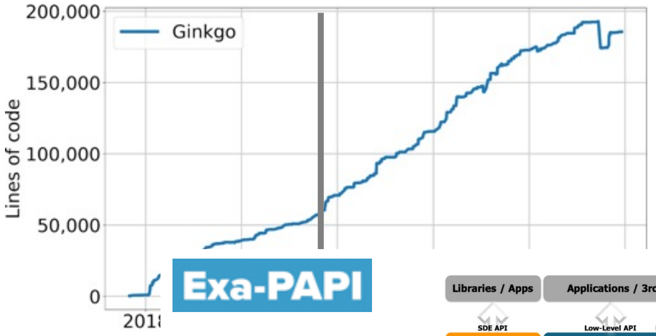
- Platform Portability
- Performance
- Rapid integration of new algorithms
- xSDK / E4S Community Policies
- BSSw expertise / experience
- Modern C++
- CI/CD and unit testing
- Open source & permissive licensing

Before the first line of code is written, we spend a year on whiteboard discussions.

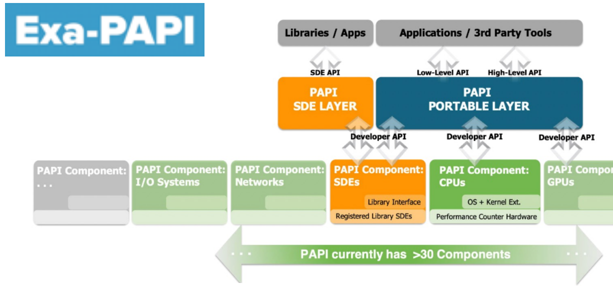


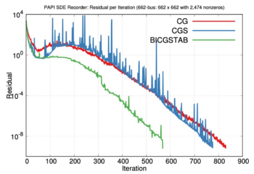
Ginkgo

	Functionality	OMP	CUDA	HIP
Basic	SpMV	✔	✔	✔
	SpMM	✔	✔	✔
	SpGeMM	✔	✔	✔
Krylov solvers	BiCG	✔	✔	✔
	BiCGSTAB	✔	✔	✔
	CG	✔	✔	✔
	CGS	✔	✔	✔
	GMRES	✔	✔	✔
	IDR	✔	✔	✔
Preconditioners	(Block-)Jacobi	✔	✔	✔
	ILU/IC		✔	✔
	Parallel ILU/IC	✔	✔	✔
	Parallel ILUT/ICT	✔	✔	✔
	Sparse Approximate Inverse	✔	✔	✔
Utilities	On-Device Matrix Assembly	✔	✔	✔
	MC64/RCM reordering	✔		
	Wrapping user data		✔	



Spack xSDK E4S





EXASCALE COMPUTING PROJECT

The Design of an ECP Math Library

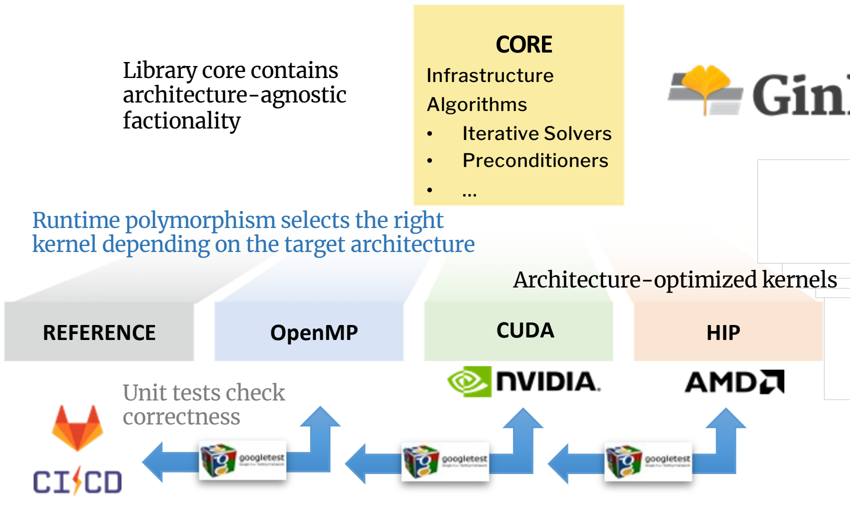
MAGMA SPARSE

MAGMA-sparse as a “child” of MAGMA explores the development of sparse linear algebra for NVIDIA GPUs.

Design considerations for Ginkgo

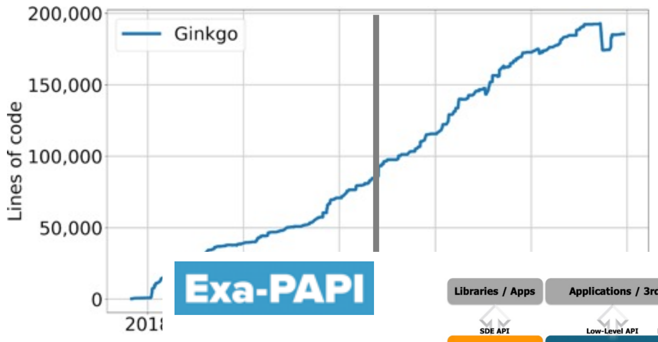
- Platform Portability
- Performance
- Rapid integration of new algorithms
- xSDK / E4S Community Policies
- BSSw expertise / experience
- Modern C++
- CI/CD and unit testing
- Open source & permissive licensing

Before the first line of code is written, we spend a year on whiteboard discussions.

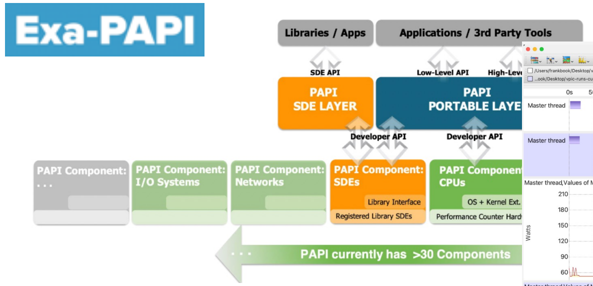


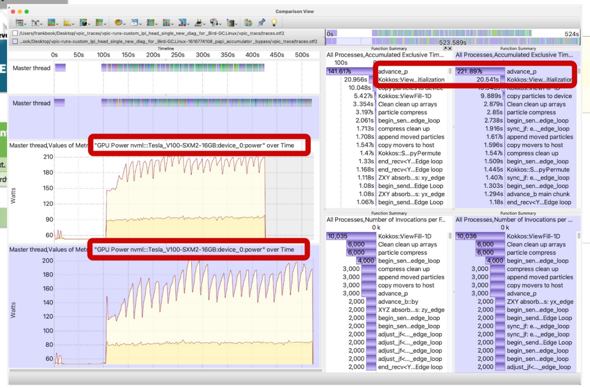
Ginkgo

	Functionality	OMP	CUDA	HIP
Basic	SpMV	✓	✓	✓
	SpMM	✓	✓	✓
	SpGeMM	✓	✓	✓
Krylov solvers	BiCG	✓	✓	✓
	BiCGSTAB	✓	✓	✓
	CG	✓	✓	✓
	CGS	✓	✓	✓
	GMRES	✓	✓	✓
	IDR	✓	✓	✓
Preconditioners	(Block-)Jacobi	✓	✓	✓
	ILU/IC		✓	✓
	Parallel ILU/IC	✓	✓	✓
	Parallel ILUT/ICT	✓	✓	✓
	Sparse Approximate Inverse	✓	✓	✓



Spack xSDK E4S





EXASCALE COMPUTING PROJECT

The Design of an ECP Math Library

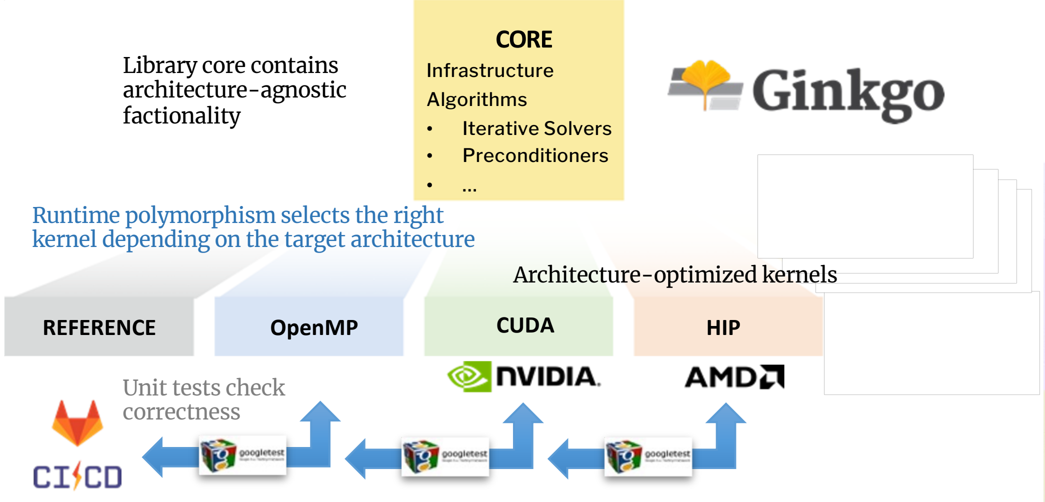
MAGMA SPARSE

MAGMA-sparse as a “child” of MAGMA explores the development of sparse linear algebra for NVIDIA GPUs.

Design considerations for Ginkgo

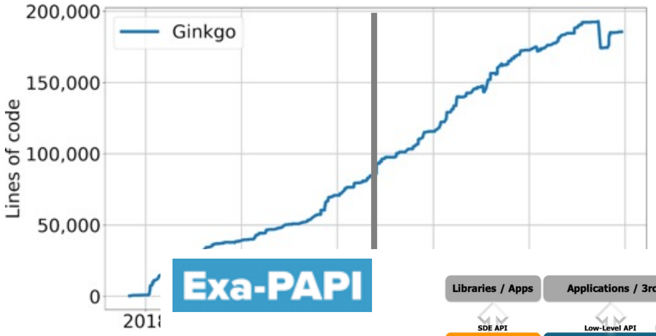
- Platform Portability
- Performance
- Rapid integration of new algorithms
- xSDK / E4S Community Policies
- BSSw expertise / experience
- Modern C++
- CI/CD and unit testing
- Open source & permissive licensing

Before the first line of code is written, we spend a year on whiteboard discussions.



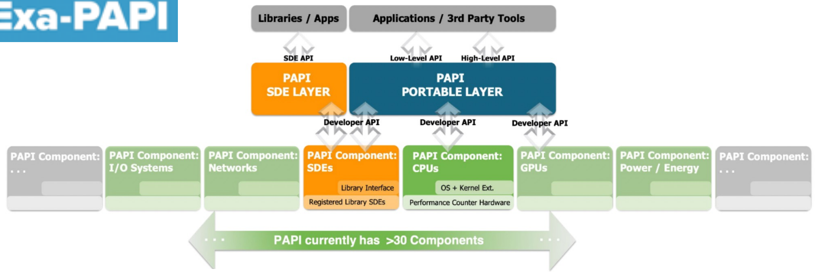
Ginkgo

	Functionality	OMP	CUDA	HIP
Basic	SpMV	✓	✓	✓
	SpMM	✓	✓	✓
	SpGeMM	✓	✓	✓
Krylov solvers	BiCG	✓	✓	✓
	BiCGSTAB	✓	✓	✓
	CG	✓	✓	✓
	CGS	✓	✓	✓
	GMRES	✓	✓	✓
	IDR	✓	✓	✓
Preconditioners	(Block-)Jacobi	✓	✓	✓
	ILU/IC		✓	✓
	Parallel ILU/IC	✓	✓	✓
	Parallel ILUT/ICT	✓	✓	✓
	Sparse Approximate Inverse	✓	✓	✓
Utilities	On-Device Matrix Assembly	✓	✓	✓
	MC64/RCM reordering	✓		
	Wrapping user data	✓	✓	✓
	Logging	✓	✓	✓
	PAPI counters	✓	✓	✓



Spack xSDK E4S

Exa-PAPI



ECP EXASCALE COMPUTING PROJECT

The Design of an ECP Math Library

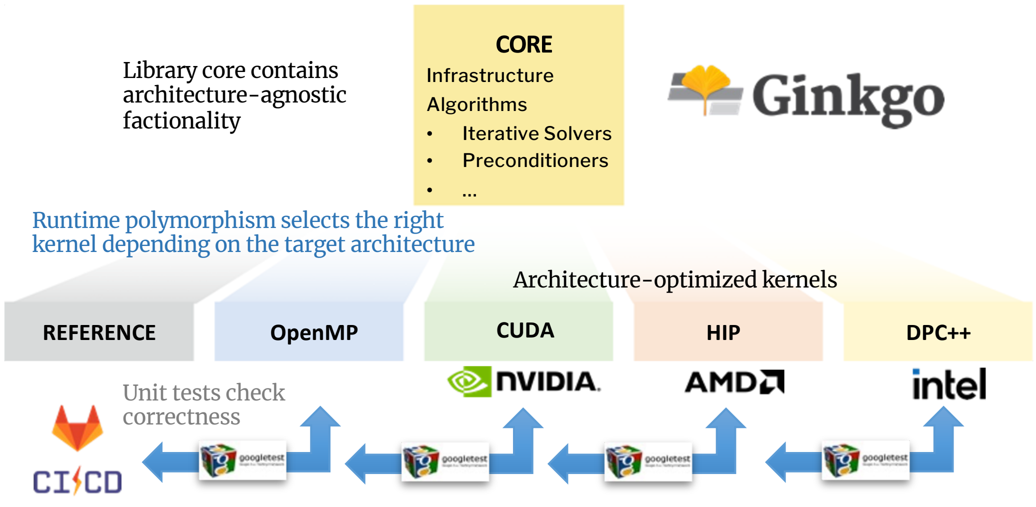
MAGMA SPARSE

MAGMA-sparse as a “child” of MAGMA explores the development of sparse linear algebra for NVIDIA GPUs.

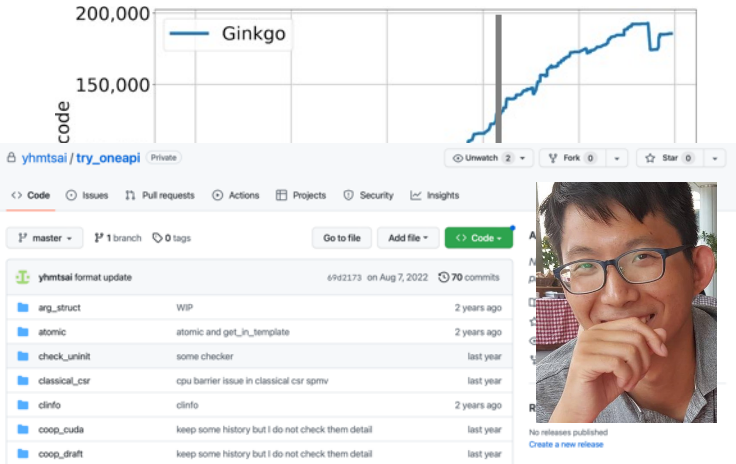
Design considerations for Ginkgo

- Platform Portability
- Performance
- Rapid integration of new algorithms
- xSDK / E4S Community Policies
- BSSw expertise / experience
- Modern C++
- CI/CD and unit testing
- Open source & permissive licensing

Before the first line of code is written, we spend a year on whiteboard discussions.



	Functionality	OMP	CUDA	HIP	DPC++
Basic	SpMV	✔	✔	✔	✔
	SpMM	✔	✔	✔	✔
	SpGeMM	✔	✔	✔	✔
Krylov solvers	BiCG	✔	✔	✔	✔
	BiCGSTAB	✔	✔	✔	✔
	CG	✔	✔	✔	✔
	CGS	✔	✔	✔	✔
	GMRES	✔	✔	✔	✔
	IDR	✔	✔	✔	✔
Preconditioners	(Block-)Jacobi	✔	✔	✔	✔
	ILU/IC		✔	✔	✔
	Parallel ILU/IC	✔	✔	✔	✔
	Parallel ILUT/ICT	✔	✔	✔	✔
	Sparse Approximate Inverse	✔	✔	✔	✔
Utilities	On-Device Matrix Assembly	✔	✔	✔	✔
	MC64/RCM reordering	✔			
	Wrapping user data	✔ (all)			
	Logging	✔ (all)			
	PAPI counters	✔ (all)			



Spack xSDK E4S Exa-PAPI oneAPI

Industry Collaboration with bi-weekly meetings

EXASCALE COMPUTING PROJECT

The Design of an ECP Math Library

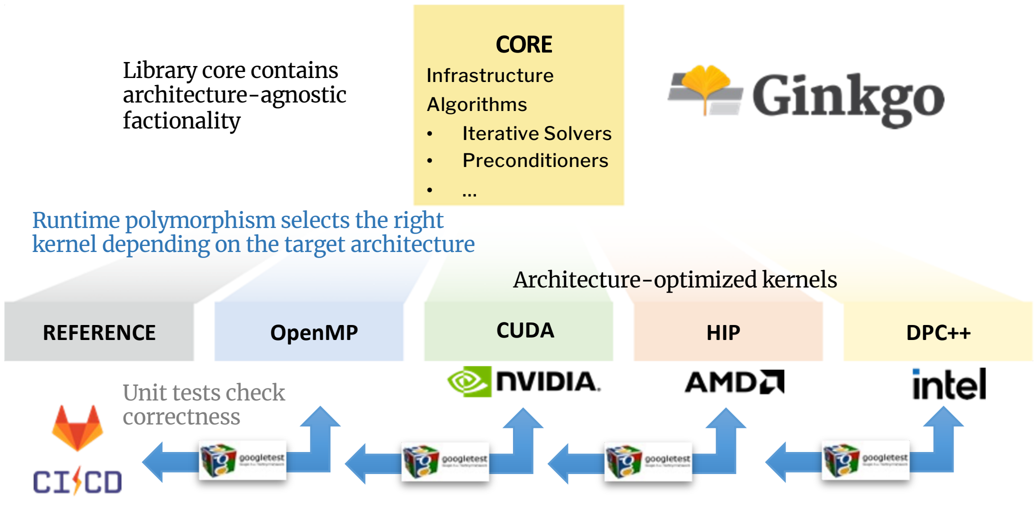
MAGMA SPARSE

MAGMA-sparse as a “child” of MAGMA explores the development of sparse linear algebra for NVIDIA GPUs.

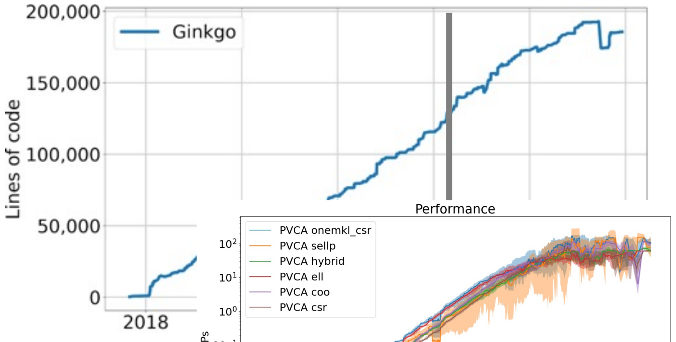
Design considerations for Ginkgo

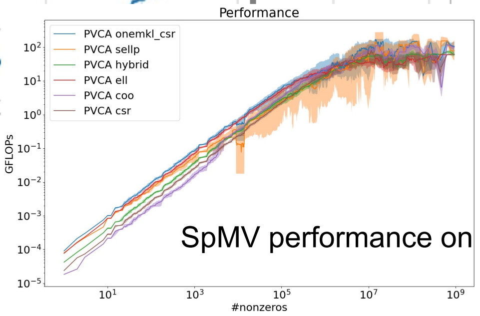
- Platform Portability
- Performance
- Rapid integration of new algorithms
- xSDK / E4S Community Policies
- BSSw expertise / experience
- Modern C++
- CI/CD and unit testing
- Open source & permissive licensing

Before the first line of code is written, we spend a year on whiteboard discussions.



	Functionality	OMP	CUDA	HIP	DPC++
Basic	SpMV	✓	✓	✓	✓
	SpMM	✓	✓	✓	✓
	SpGeMM	✓	✓	✓	✓
Krylov solvers	BiCG	✓	✓	✓	✓
	BiCGSTAB	✓	✓	✓	✓
	CG	✓	✓	✓	✓
	CGS	✓	✓	✓	✓
	GMRES	✓	✓	✓	✓
	IDR	✓	✓	✓	✓
Preconditioners	(Block-)Jacobi	✓	✓	✓	✓
	ILU/IC		✓	✓	✓
	Parallel ILU/IC	✓	✓	✓	✓
	Parallel ILUT/ICT	✓	✓	✓	✓
	Sparse Approximate Inverse	✓	✓	✓	✓
Utilities	On-Device Matrix Assembly	✓	✓	✓	✓
	MC64/RCM reordering	✓			
	Wrapping user data	✓			
	Logging	✓			
	PAPI counters	✓			





SpMV performance on Intel PVC

Spack xSDK E4S Exa-PAPI oneAPI

Industry Collaboration with bi-weekly meetings

EXASCALE COMPUTING PROJECT

The Design of an ECP Math Library

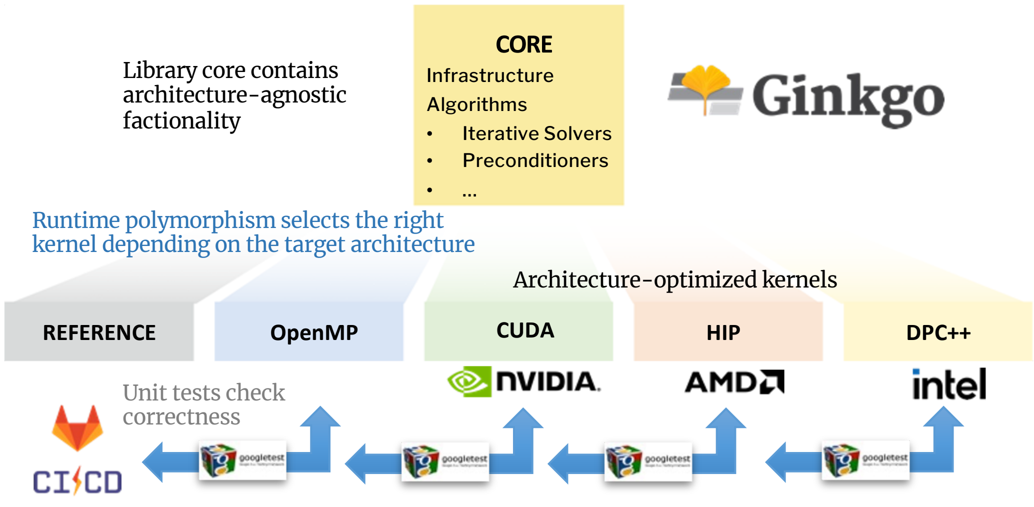
MAGMA SPARSE

MAGMA-sparse as a “child” of MAGMA explores the development of sparse linear algebra for NVIDIA GPUs.

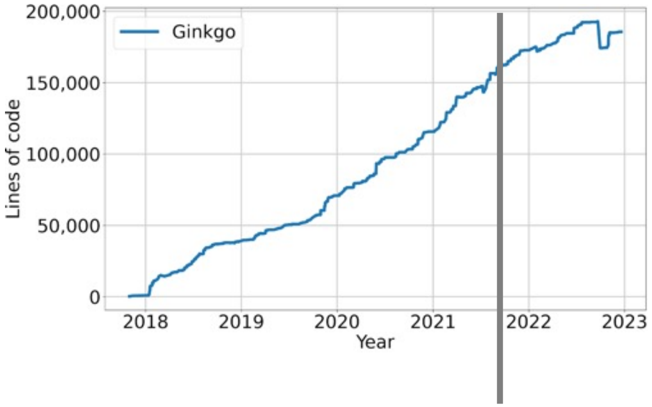
Design considerations for Ginkgo

- Platform Portability
- Performance
- Rapid integration of new algorithms
- xSDK / E4S Community Policies
- BSSw expertise / experience
- Modern C++
- CI/CD and unit testing
- Open source & permissive licensing

Before the first line of code is written, we spend a year on whiteboard discussions.



	Functionality	OMP	CUDA	HIP	DPC++
Basic	SpMV	✓	✓	✓	✓
	SpMM	✓	✓	✓	✓
	SpGeMM	✓	✓	✓	✓
Krylov solvers	BiCG	✓	✓	✓	✓
	BiCGSTAB	✓	✓	✓	✓
	CG	✓	✓	✓	✓
	CGS	✓	✓	✓	✓
	GMRES	✓	✓	✓	✓
	IDR	✓	✓	✓	✓
Preconditioners	(Block-)Jacobi	✓	✓	✓	✓
	ILU/IC		✓	✓	✓
	Parallel ILU/IC	✓	✓	✓	✓
	Parallel ILUT/ICT	✓	✓	✓	✓
	Sparse Approximate Inverse	✓	✓	✓	✓
Utilities	On-Device Matrix Assembly	✓	✓	✓	✓
	MC64/RCM reordering	✓			
	Wrapping user data	✓	✓	✓	✓
	Logging	✓	✓	✓	✓
	PAPI counters	✓	✓	✓	✓



Spack xSDK E4S Exa-PAPI

oneAPI

Industry Collaboration with bi-weekly meetings

xSDK

Focus efforts
- Mixed precision
- batched

The Design of an ECP Math Library

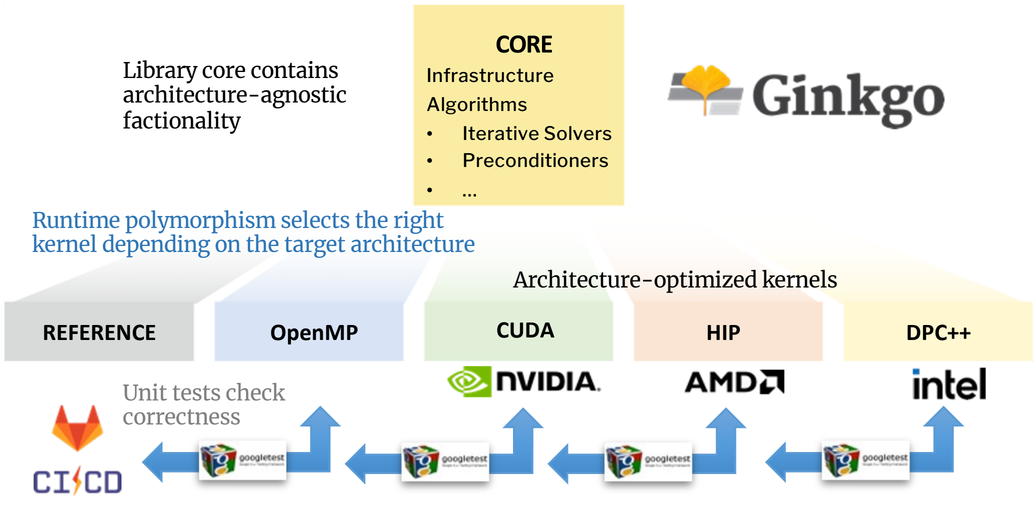
MAGMA SPARSE

MAGMA-sparse as a “child” of MAGMA explores the development of sparse linear algebra for NVIDIA GPUs.

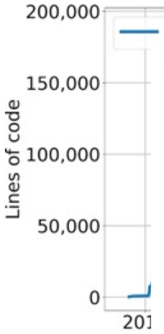
Design considerations for Ginkgo

- Platform Portability
- Performance
- Rapid integration of new algorithms
- xSDK / E4S Community Policies
- BSSw expertise / experience
- Modern C++
- CI/CD and unit testing
- Open source & permissive licensing

Before the first line of code is written, we spend a year on whiteboard discussions.



	Functionality	OMP	CUDA	HIP	DPC++
Basic	SpMV	✓	✓	✓	✓
	SpMM	✓	✓	✓	✓
	SpGeMM	✓	✓	✓	✓
Krylov solvers	BiCG	✓	✓	✓	✓
	BiCGSTAB	✓	✓	✓	✓
	CG	✓	✓	✓	✓
	CGS	✓	✓	✓	✓
	GMRES	✓	✓	✓	✓
	IDR	✓	✓	✓	✓
Preconditioners	(Block-)Jacobi	✓	✓	✓	✓
	ILU/IC		✓	✓	✓
	Parallel ILU/IC	✓	✓	✓	✓
	Parallel ILUT/ICT	✓	✓	✓	✓
	Sparse Approximate Inverse	✓	✓	✓	✓
AMG	AMG preconditioner	✓	✓	✓	✓
	AMG solver	✓	✓	✓	✓
	Parallel Graph Match	✓	✓	✓	✓
Utilities	On-Device Matrix Assembly	✓	✓	✓	✓
	MC64/RCM reordering	✓			
	Wrapping user data	✓ (all)			
	Logging	✓ (all)			
	PAPI counters	✓ (all)			



ICL UTK @ICL_UTK · Sep 13

Congratulations to Yu-Hsiang Mike Tsai from @KITKarlsruhe, in collaboration with ICL's Natalie Beams and @HartwigAnzt! Their paper "Mixed Precision Algebraic Multigrid on GPUs" took home a best paper award at PPAM2022. ppam.edu.pl

Spack xSDK E4S Exa-PAPI oneAPI

Industry Collaboration with bi-weekly meetings

xSDK

Focus efforts
- Mixed precision
- batched

EXASCALE COMPUTING PROJECT

The Design of an ECP Math Library

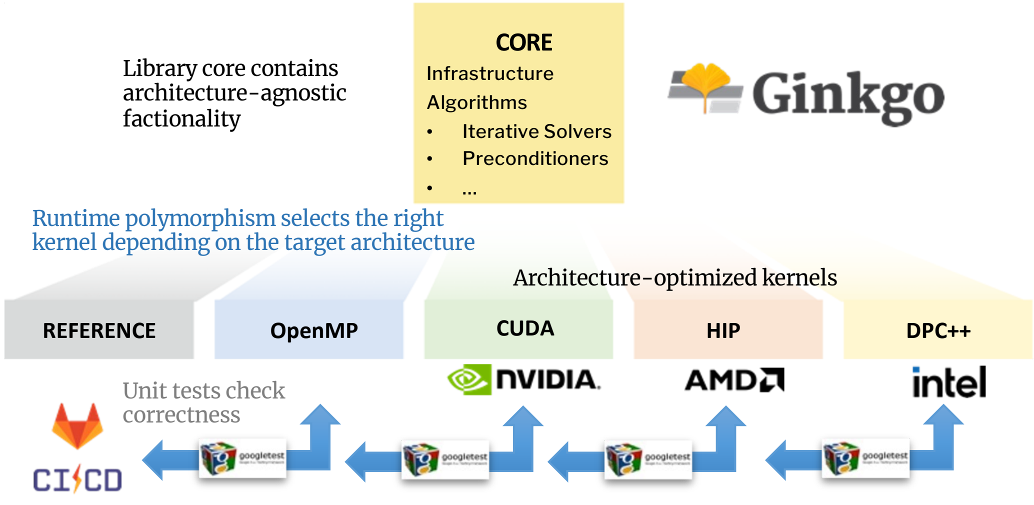
MAGMA SPARSE

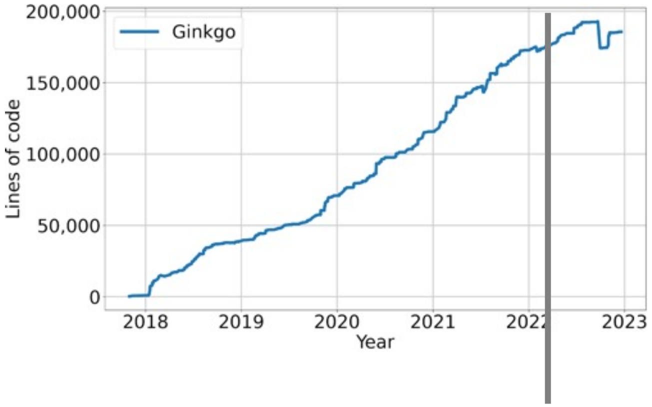
MAGMA-sparse as a "child" of MAGMA explores the development of sparse linear algebra for NVIDIA GPUs.

Design considerations for Ginkgo

- Platform Portability
- Performance
- Rapid integration of new algorithms
- xSDK / E4S Community Policies
- BSSw expertise / experience
- Modern C++
- CI/CD and unit testing
- Open source & permissive licensing

Before the first line of code is written, we spend a year on whiteboard discussions.





Spack xSDK E4S Exa-PAPI oneAPI sundials

Industry Collaboration with bi-weekly meetings

	Functionality	OMP	CUDA	HIP	DPC++
Basic	SpMV	✓	✓	✓	✓
	SpMM	✓	✓	✓	✓
	SpGeMM	✓	✓	✓	✓
Krylov solvers	BiCG	✓	✓	✓	✓
	BiCGSTAB	✓	✓	✓	✓
	CG	✓	✓	✓	✓
	CGS	✓	✓	✓	✓
	GMRES	✓	✓	✓	✓
	IDR	✓	✓	✓	✓
Preconditioners	(Block-)Jacobi	✓	✓	✓	✓
	ILU/IC		✓	✓	✓
	Parallel ILU/IC	✓	✓	✓	✓
	Parallel ILUT/ICT	✓	✓	✓	✓
	Sparse Approximate Inverse	✓	✓	✓	✓
Batched	Batched BiCGSTAB	✓	✓	✓	
	Batched CG	✓	✓	✓	
	Batched GMRES	✓	✓	✓	
	Batched ILU	✓	✓	✓	
	Batched ISAI	✓	✓	✓	
	Batched Jacobi	✓	✓	✓	
AMG	AMG preconditioner	✓	✓	✓	✓
	AMG solver	✓	✓	✓	✓
	Parallel Graph Match	✓	✓	✓	✓
Utilities	On-Device Matrix Assembly	✓	✓	✓	✓
	MC64/RCM reordering	✓			
	Wrapping user data	✓ (all)			
	Logging	✓ (all)			
	PAPI counters	✓ (all)			

ECP EXASCALE COMPUTING PROJECT

The Design of an ECP Math Library

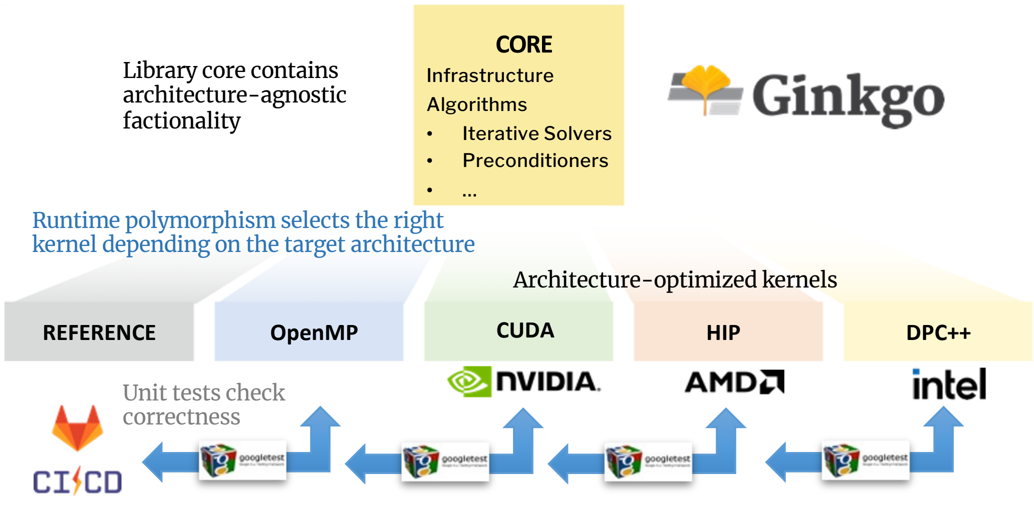
MAGMA SPARSE

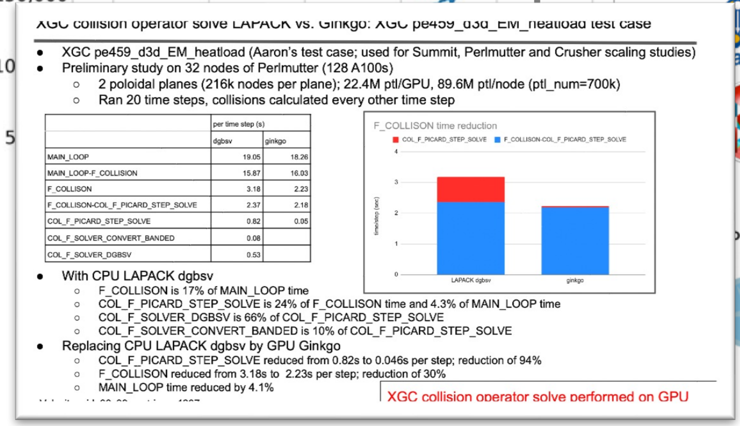
MAGMA-sparse as a “child” of MAGMA explores the development of sparse linear algebra for NVIDIA GPUs.

Design considerations for Ginkgo

- Platform Portability
- Performance
- Rapid integration of new algorithms
- xSDK / E4S Community Policies
- BSSw expertise / experience
- Modern C++
- CI/CD and unit testing
- Open source & permissive licensing

Before the first line of code is written, we spend a year on whiteboard discussions.




XGC collision operator solve LAPACK vs. Ginkgo: XGC pe459_d3d_EM_heatload test case

- XGC pe459_d3d_EM_heatload (Aaron’s test case; used for Summit, Perlmutter and Crusher scaling studies)
- Preliminary study on 32 nodes of Perlmutter (128 A100s)
  - 2 poloidal planes (216k nodes per plane); 22.4M ptl/GPU, 89.6M ptl/node (ptl_num=700k)
  - Ran 20 time steps, collisions calculated every other time step

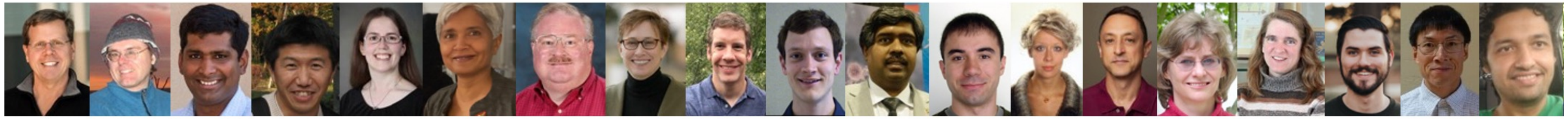
	per time step (s)	
	dgbsv	ginkgo
MAIN_LOOP	19.05	18.26
MAIN_LOOP-F_COLLISION	15.87	16.03
F_COLLISON	3.18	2.23
F_COLLISON-COL_F_PICARD_STEP_SOLVE	2.37	2.18
COL_F_PICARD_STEP_SOLVE	0.82	0.05
COL_F_SOLVER_CONVERT_BANDED	0.08	
COL_F_SOLVER_DGBSV	0.53	

- With CPU LAPACK dgbsv
  - F_COLLISON is 17% of MAIN_LOOP time
  - COL_F_PICARD_STEP_SOLVE is 24% of F_COLLISON time and 4.3% of MAIN_LOOP time
  - COL_F_SOLVER_DGBSV is 66% of COL_F_PICARD_STEP_SOLVE
  - COL_F_SOLVER_CONVERT_BANDED is 10% of COL_F_PICARD_STEP_SOLVE
- Replacing CPU LAPACK dgbsv by GPU Ginkgo
  - COL_F_PICARD_STEP_SOLVE reduced from 0.82s to 0.046s per step; reduction of 94%
  - F_COLLISON reduced from 3.18s to 2.23s per step; reduction of 30%
  - MAIN_LOOP time reduced by 4.1%

XGC collision operator solve performed on GPU

Industry Collaboration with bi-weekly meetings

	Functionality	OMP	CUDA	HIP	DPC++
Basic	SpMV	✓	✓	✓	✓
	SpMM	✓	✓	✓	✓
	SpGeMM	✓	✓	✓	✓
Krylov solvers	BiCG	✓	✓	✓	✓
	BiCGSTAB	✓	✓	✓	✓
	CG	✓	✓	✓	✓
	CGS	✓	✓	✓	✓
	GMRES	✓	✓	✓	✓
	IDR	✓	✓	✓	✓
Preconditioners	(Block-)Jacobi	✓	✓	✓	✓
	ILU/IC		✓	✓	✓
	Parallel ILU/IC	✓	✓	✓	✓
	Parallel ILUT/ICT	✓	✓	✓	✓
	Sparse Approximate Inverse	✓	✓	✓	✓
Batched	Batched BiCGSTAB	✓	✓	✓	
	Batched CG	✓	✓	✓	
	Batched GMRES	✓	✓	✓	
	Batched ILU	✓	✓	✓	
	Batched ISAI	✓	✓	✓	
	Batched Jacobi	✓	✓	✓	
AMG	AMG preconditioner	✓	✓	✓	✓
	AMG solver	✓	✓	✓	✓
	Parallel Graph Match	✓	✓	✓	✓
Utilities	On-Device Matrix Assembly	✓	✓	✓	✓
	MC64/RCM reordering	✓			
	Wrapping user data	✓ (all)			
	Logging	✓ (all)			
	PAPI counters	✓ (all)			



EXASCALE COMPUTING PROJECT

The Design of an ECP Math Library

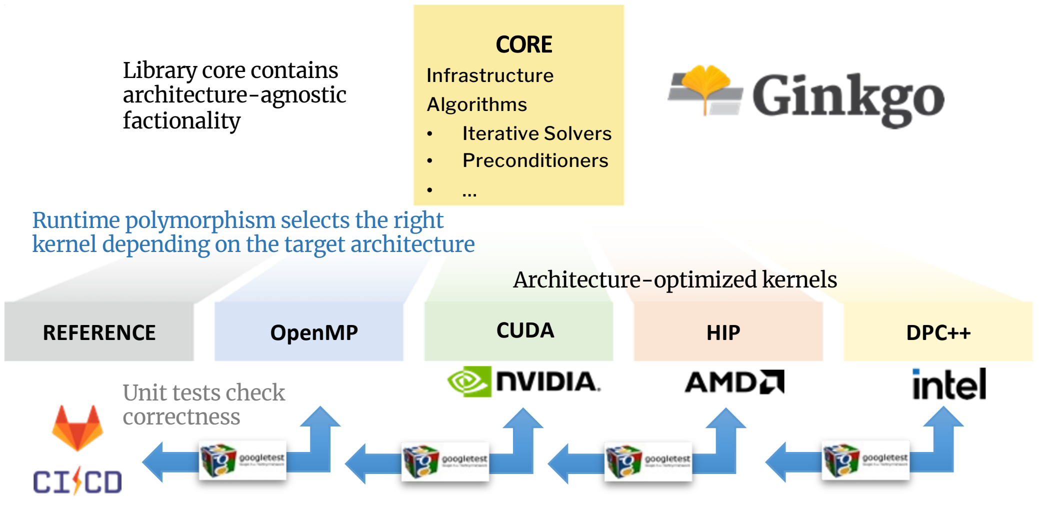
MAGMA SPARSE

MAGMA-sparse as a “child” of MAGMA explores the development of sparse linear algebra for NVIDIA GPUs.

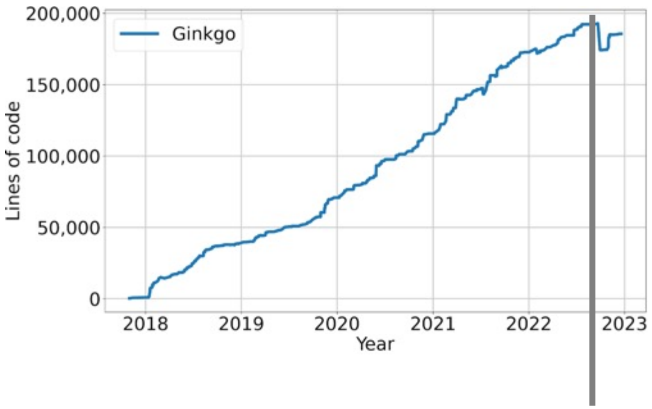
Design considerations for Ginkgo

- Platform Portability
- Performance
- Rapid integration of new algorithms
- xSDK / E4S Community Policies
- BSSw expertise / experience
- Modern C++
- CI/CD and unit testing
- Open source & permissive licensing

Before the first line of code is written, we spend a year on whiteboard discussions.



Ginkgo



Industry Collaboration with bi-weekly meetings

	Functionality	OMP	CUDA	HIP	DPC++
Basic	SpMV	✓	✓	✓	✓
	SpMM	✓	✓	✓	✓
	SpGeMM	✓	✓	✓	✓
Krylov solvers	BiCG	✓	✓	✓	✓
	BiCGSTAB	✓	✓	✓	✓
	CG	✓	✓	✓	✓
	CGS	✓	✓	✓	✓
	GMRES	✓	✓	✓	✓
	IDR	✓	✓	✓	✓
Preconditioners	(Block-)Jacobi	✓	✓	✓	✓
	ILU/IC		✓	✓	✓
	Parallel ILU/IC	✓	✓	✓	✓
	Parallel ILUT/ICT	✓	✓	✓	✓
	Sparse Approximate Inverse	✓	✓	✓	✓
Batched	Batched BiCGSTAB	✓	✓	✓	
	Batched CG	✓	✓	✓	
	Batched GMRES	✓	✓	✓	
	Batched ILU	✓	✓	✓	
	Batched ISAI	✓	✓	✓	
	Batched Jacobi	✓	✓	✓	
AMG	AMG preconditioner	✓	✓	✓	✓
	AMG solver	✓	✓	✓	✓
	Parallel Graph Match	✓	✓	✓	✓
Sparse direct	Symbolic Cholesky	✓	✓	✓	✓
	Numeric Cholesky	UNDER DEVELOPMENT			
	Symbolic LU	✓	✓	✓	✓
	Numeric LU	✓	✓	✓	
	Sparse TRSV	✓	✓	✓	
Utilities	On-Device Matrix Assembly	✓	✓	✓	✓
	MC64/RCM reordering	✓			
	Wrapping user data	✓			
	Logging	✓			
	PAPI counters	✓			

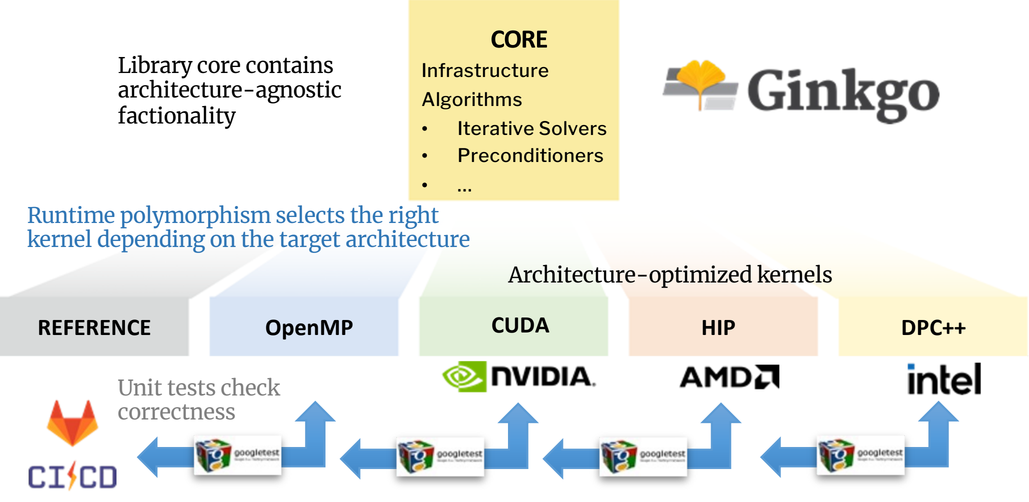
EXASCALE COMPUTING PROJECT

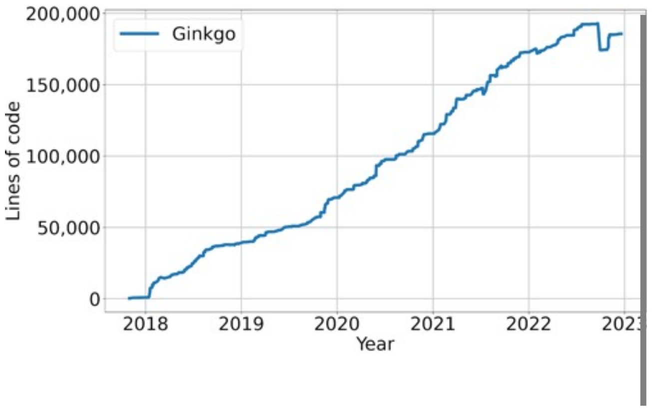
"Now" – Near completion of ECP

- Sustainable software design ready for the addition of new backends.
- Numerous collaborations with DOE partners and industry.
  - Bi-weekly meetings with Intel
  - regular meetings with AMD
- Significant interest from the computational science community outside ECP:

HYTEG deal.II OpenFOAM preCICE

The Open Source CFD Toolbox





	Functionality	OMP	CUDA	HIP	DPC++
Basic	SpMV	✓	✓	✓	✓
	SpMM	✓	✓	✓	✓
	SpGeMM	✓	✓	✓	✓
Krylov solvers	BiCG	✓	✓	✓	✓
	BiCGSTAB	✓	✓	✓	✓
	CG	✓	✓	✓	✓
	CGS	✓	✓	✓	✓
	GMRES	✓	✓	✓	✓
	IDR	✓	✓	✓	✓
Preconditioners	(Block-)Jacobi	✓	✓	✓	✓
	ILU/IC		✓	✓	✓
	Parallel ILU/IC	✓	✓	✓	✓
	Parallel ILUT/ICT	✓	✓	✓	✓
	Sparse Approximate Inverse	✓	✓	✓	✓
Batched	Batched BiCGSTAB	✓	✓	✓	
	Batched CG	✓	✓	✓	
	Batched GMRES	✓	✓	✓	
	Batched ILU	✓	✓	✓	
	Batched ISAI	✓	✓	✓	
	Batched Jacobi	✓	✓	✓	
AMG	AMG preconditioner	✓	✓	✓	✓
	AMG solver	✓	✓	✓	✓
	Parallel Graph Match	✓	✓	✓	✓
Sparse direct	Symbolic Cholesky	✓	✓	✓	✓
	Numeric Cholesky	UNDER DEVELOPMENT			
	Symbolic LU	✓	✓	✓	✓
	Numeric LU	✓	✓	✓	
	Sparse TRSV	✓	✓	✓	
Utilities	On-Device Matrix Assembly	✓	✓	✓	✓
	MC64/RCM reordering	✓			
	Wrapping user data	✓			
	Logging	✓			
	PAPI counters	✓			

Spack xSDK E4S Exa-PAPI

oneAPI Industry Collaboration with bi-weekly meetings

sundials XGC

EXASGD

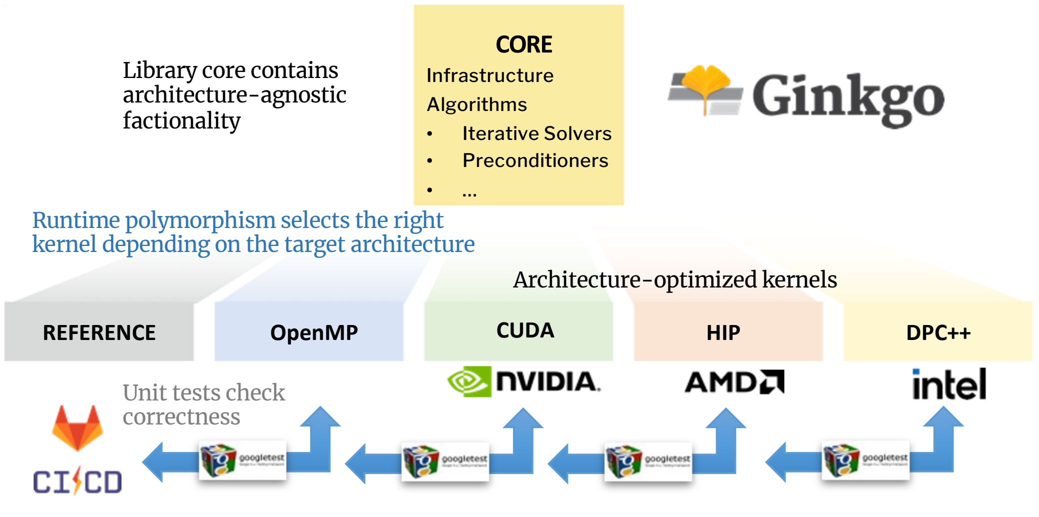
ECP EXASCALE COMPUTING PROJECT

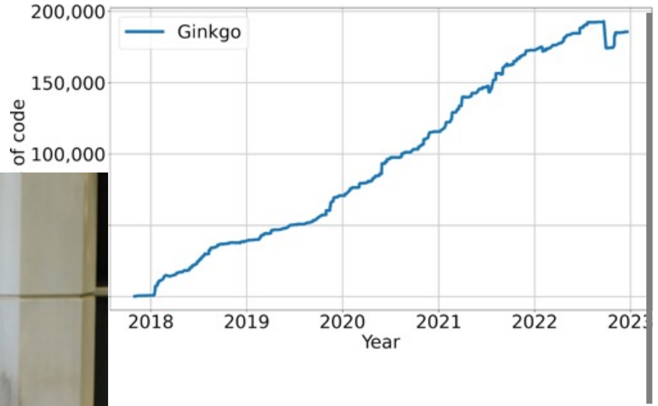
“Now” – Near completion of ECP

- Sustainable software design ready for the addition of new backends.
- Numerous collaborations with DOE partners and industry.
  - Bi-weekly meetings with Intel
  - regular meetings with AMD
- Significant interest from the computational science community outside ECP:

HYTEG | deal.II | OpenFOAM *The Open Source CFD Toolbox* | preCICE

- *Myself: Started as Director of ICL in August 2022.*





	Functionality	OMP	CUDA	HIP	DPC++
Basic	SpMV	✓	✓	✓	✓
	SpMM	✓	✓	✓	✓
	SpGeMM	✓	✓	✓	✓
Krylov solvers	BiCG	✓	✓	✓	✓
	BiCGSTAB	✓	✓	✓	✓
	CG	✓	✓	✓	✓
	CGS	✓	✓	✓	✓
	GMRES	✓	✓	✓	✓
	IDR	✓	✓	✓	✓
Preconditioners	(Block-)Jacobi	✓	✓	✓	✓
	ILU/IC		✓	✓	✓
	Parallel ILU/IC	✓	✓	✓	✓
	Parallel ILUT/ICT	✓	✓	✓	✓
	Sparse Approximate Inverse	✓	✓	✓	✓
Batched	Batched BiCGSTAB	✓	✓	✓	
	Batched CG	✓	✓	✓	
	Batched GMRES	✓	✓	✓	
	Batched ILU	✓	✓	✓	
	Batched ISAI	✓	✓	✓	
	Batched Jacobi	✓	✓	✓	
AMG	AMG preconditioner	✓	✓	✓	✓
	AMG solver	✓	✓	✓	✓
	Parallel Graph Match	✓	✓	✓	✓
Sparse direct	Symbolic Cholesky	✓	✓	✓	✓
	Numeric Cholesky	UNDER DEVELOPMENT			
	Symbolic LU	✓	✓	✓	✓
	Numeric LU	✓	✓	✓	
	Sparse TRSV	✓	✓	✓	
Utilities	On-Device Matrix Assembly	✓	✓	✓	✓
	MC64/RCM reordering	✓			
	Wrapping user data	✓ (all)			
	Logging	✓ (all)			
	PAPI counters	✓ (all)			

Spack | xSDK | E4S | Exa-PAPI | oneAPI | sundials | XGC | EXASGD

Industry Collaboration with bi-weekly meetings

Lessons learnt from ECP

- **Earmarking roughly half the budget to Software & App development is a game changer.**
  - **Central component for the success of ECP.**
  - This concept becomes the blueprint for other nations.

- **Workforce recruitment and workforce retention are the key to success in software development.**
  - Money does not write software. RSEs do. **We need to create attractive career plans.**
  - Chips act foresees workforce shortage: we need to train the future workforce for research and industry and invest in **strategic cooperation between Universities and DOE Labs.**

- **Anticipating the future in hardware development accelerates the porting process.**
  - **Blueprints** and **early access systems** both useful.
  - **Interaction with industry** is mutually beneficial.

- **Management, tools, and strategic initiatives are important.**
  - Jira milestones and deliverables give projects and collaborative interactions a structure and timeline.
  - **Strategic focus groups** bring experts together and **create collaboration**.