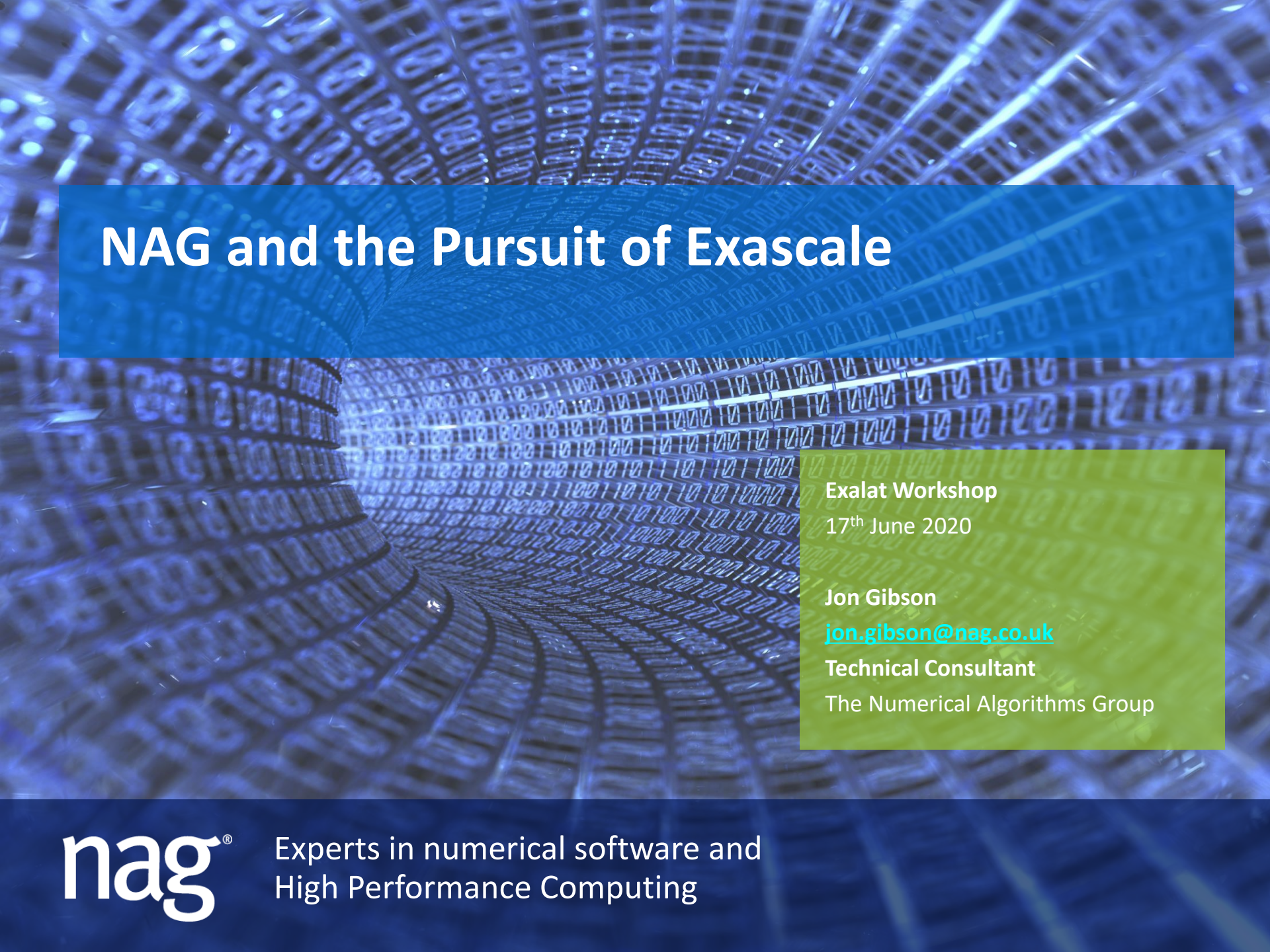


# NAG and the Pursuit of Exascale

**Exalat Workshop**

17th June 2020

**Jon Gibson**

jon.gibson@nag.co.uk

**Technical Consultant**

The Numerical Algorithms Group

nag®

Experts in numerical software and
High Performance Computing

# NAG and the Pursuit of Exascale

- What do NAG do?
- Highly Scalable Linear Algebra
- Helping the European scientific community achieve exascale

# NAG's Products, Services and Users

## What we do

- Produce **numerical libraries** and other software for Python, C/C++ & Fortran programmers, Matlab, R, Excel etc. Mark 27.1 this year.
- Software for laptops, GPUs through to supercomputers.
- Fortran compiler, the best checking compiler in the world!
- Algorithmic Differentiation tools.
- **Software consultancy**, again from desktop to High Performance Computers (HPC) including training.
- Multiple projects with universities, the EU, etc.

## Library Users:

- Academics
- Finance
- Engineers, etc …

## Consultancy:

- Oil & Gas
- Finance
- Etc, …

# NAG Libraries

- Our **numerical algorithms** are contributed by some of the world's most renowned mathematicians and computer scientists.
- Our **code development** is mainly in house with some collaboration with academia and industry.
- In short, NAG apply state-of-the-art software techniques to state-of-the-art algorithms.
- Collaborations at multiple institutions including PhDs, MSc projects and Knowledge Transfer Partnerships (KTPs).
- Currently a KTP on parallel performant numerical linear algebra.

# NAG Libraries and Toolboxes, 1800+ routines

- Root Finding
- Summation of Series
- Quadrature
- Ordinary Differential Equations
- Partial Differential Equations
- Numerical Differentiation
- Integral Equations
- Mesh Generation
- Interpolation
- Curve and Surface Fitting
- Optimization
- Approximations of Special Functions
- Option Pricing
- Dense Linear Algebra
- Sparse Linear Algebra
- Correlation & Regression Analysis
- Multivariate Methods
- Analysis of Variance
- Random Number Generators
- Univariate Estimation
- Nonparametric Statistics
- Smoothing in Statistics
- Contingency Table Analysis
- Survival Analysis
- Time Series Analysis
- Operations Research

# Numerical Algorithms

- Leave the scientist to do the science.
- Numerical computation is difficult to do accurately.
- Problems of :
  - Overflow / underflow
  - Condition
  - Stability
- Importance of error analysis and bounds on errors.
- So use of libraries or consultancy.
- Accuracy and then speed!

# Optimised BLAS and LAPACK for Modern CPU Architectures

# The NAG NPC library

- "NAG Performance Component"
- Targets AMD and ARM architectures, where an alternative to MKL is required for good performance.
- Uses the OpenMP task-based programming model to efficiently exploit modern multi/many-core architectures.

# Key Features of the NAG NPC library

Serial and parallel BLAS and LAPACK

Double and single precision for both real and complex

32-bit and 64-bit integer versions

Standard Fortran and C interfaces

Optimized for a large range of CPU architectures

# Performance of NPC on AMD EPYC Naples

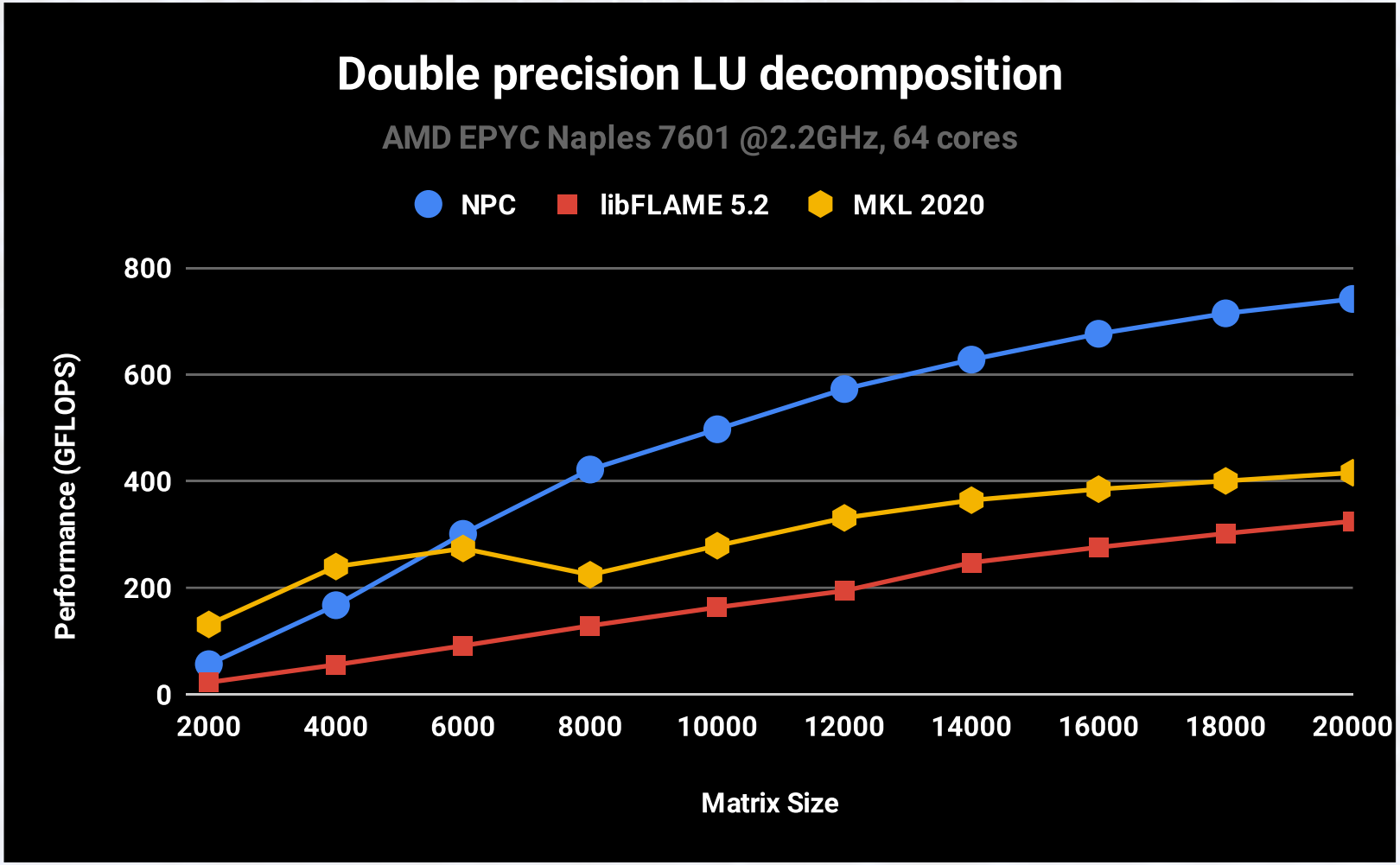
## Sofware compared in the benchmark

- LibFLAME 5.2 linked against AMD BLIS 2.1
- MKL 2020
- The NAG NPC library

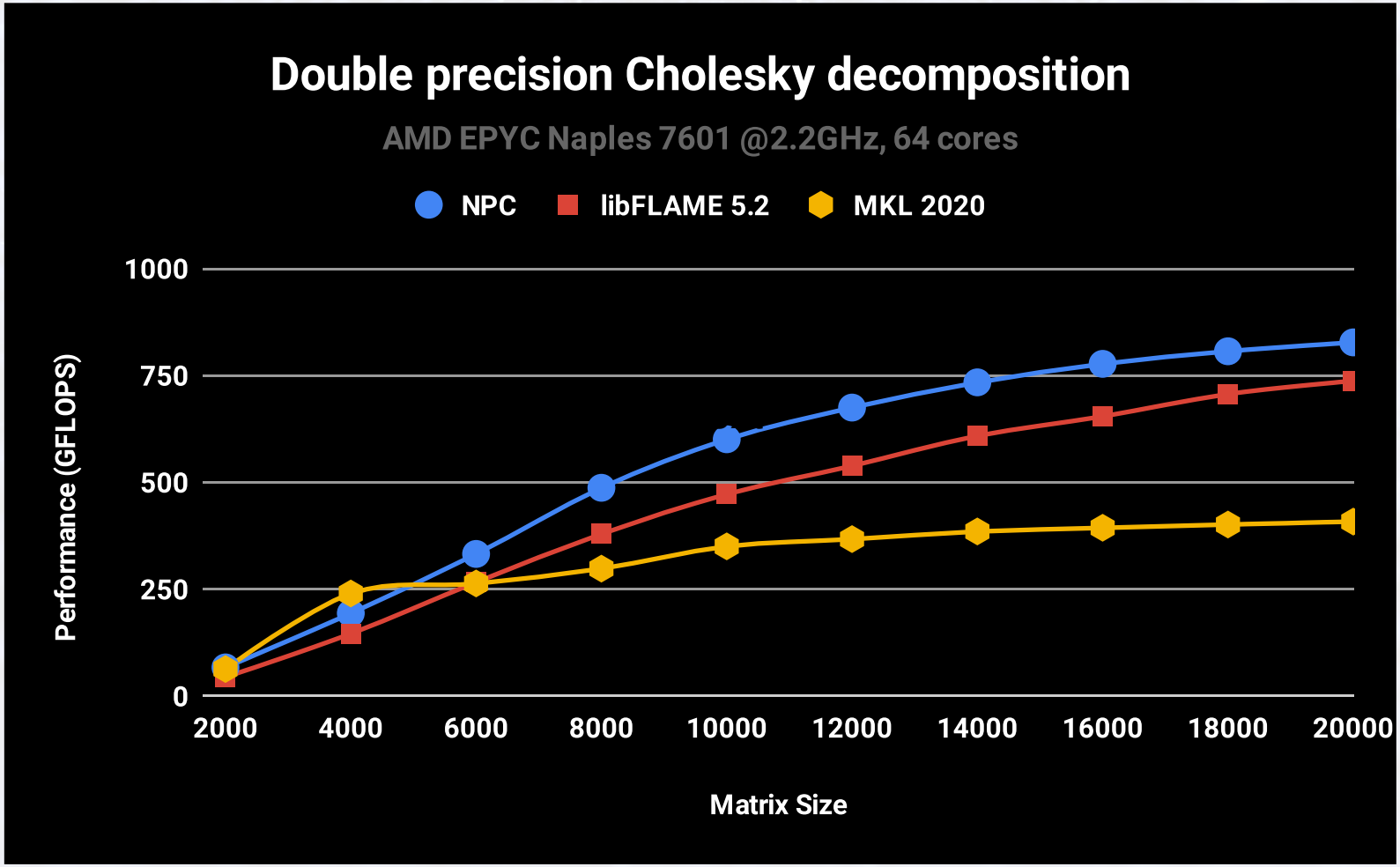
## Routines evaluated

- LAPACK Cholesky factorization
- LAPACK LU factorization
- LAPACK QR factorization
- LAPACK LDLt factorization

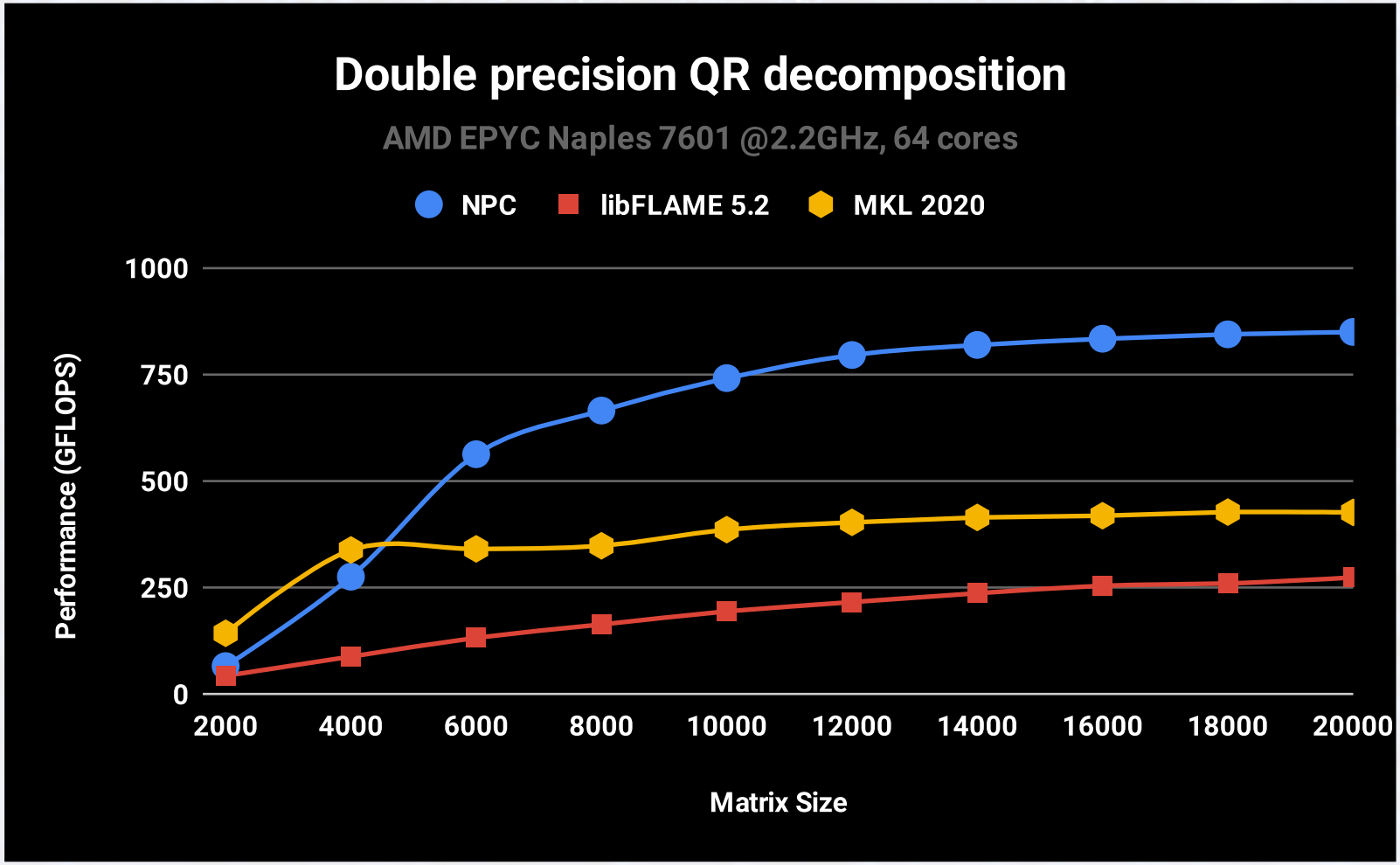
# Performance of NPC on AMD EPYC Naples



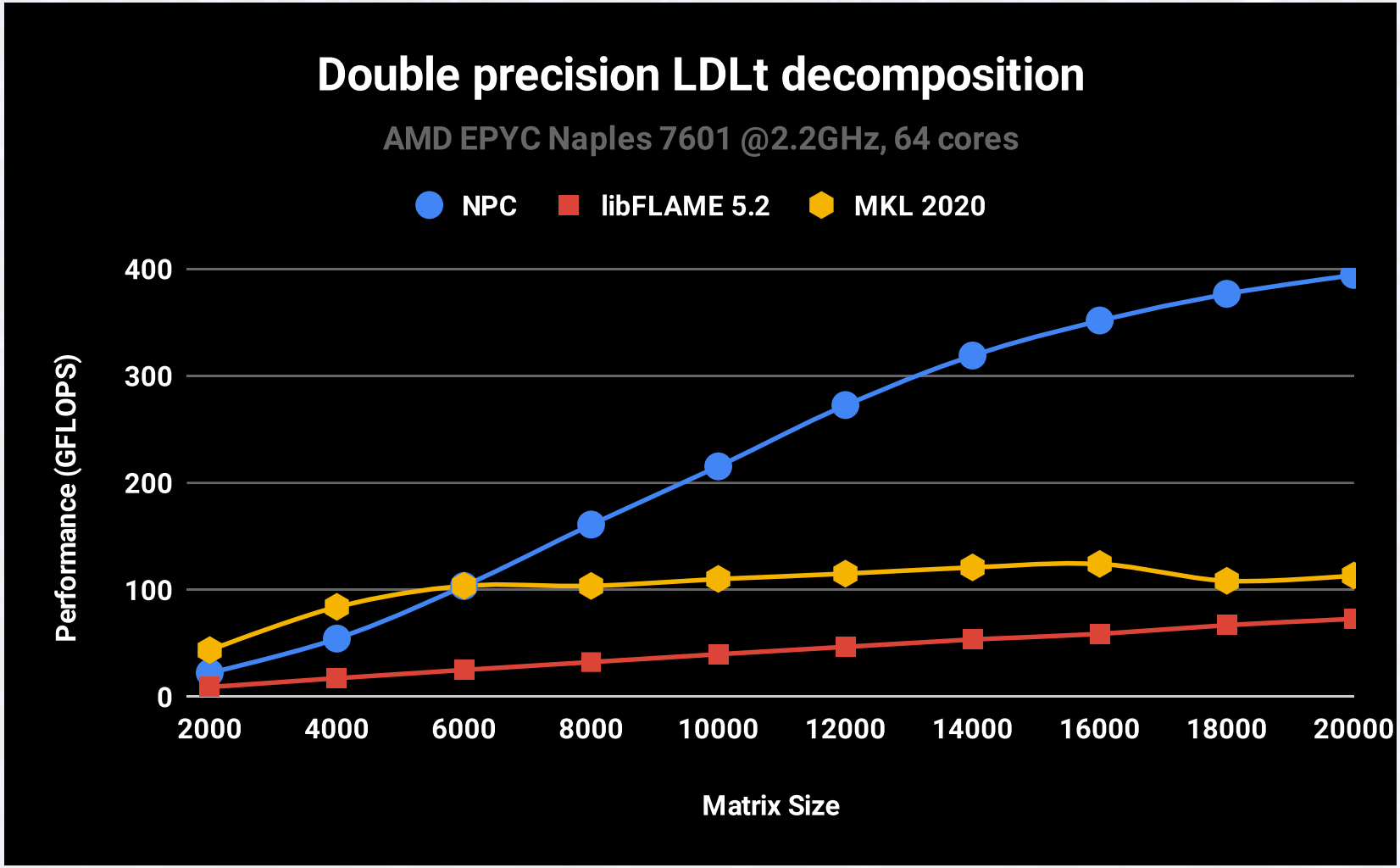
# Performance of NPC on AMD EPYC Naples



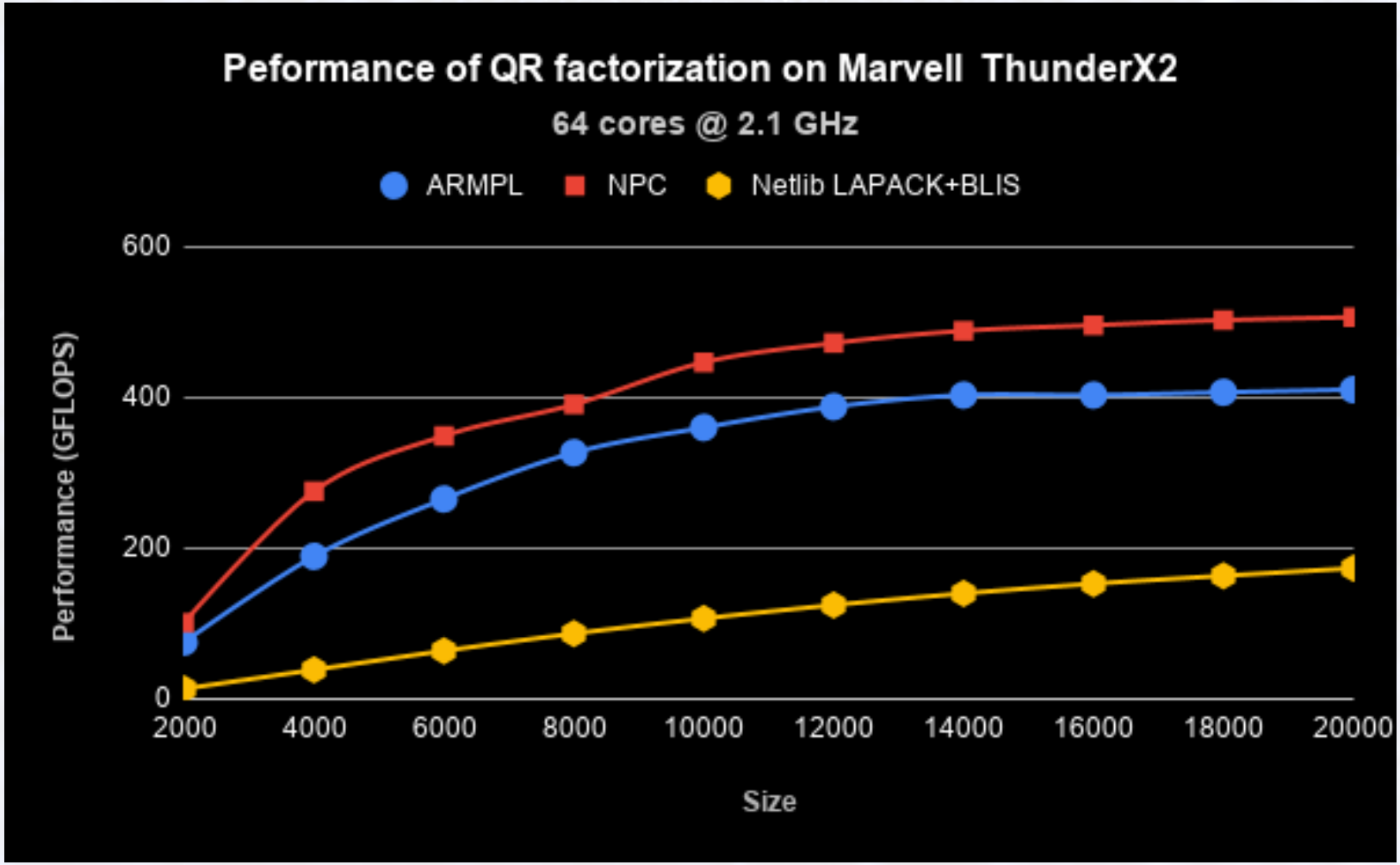
# Performance of NPC on AMD EPYC Naples

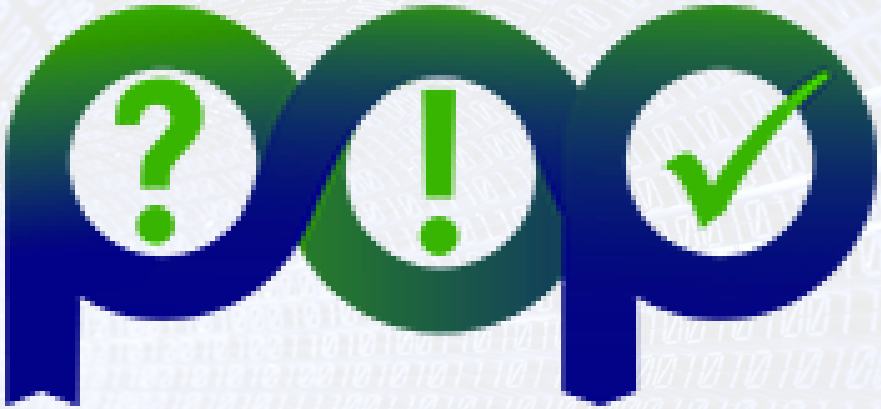


# Performance of NPC on AMD EPYC Naples



# Performance of NPC on Marvell ThunderX2





# A Centre of Excellence for Performance Optimisation and Productivity (POP)

# The POP CoE

- One of ten EU **Centres of Excellence in HPC Applications**
  - POP is a CoE in **Performance Optimisation and Productivity**
    - Promoting **best practices in parallel programming**
- POP provides **FREE** services
  - for (EU) academic and industrial (parallel) codes and users
  - across all application areas, platforms, scales, languages and parallel methods
  - Currently approaching 200 services delivered
- giving users
  - a precise understanding of application and system behaviour
  - suggestions/support on how to refactor code in the most productive way

# The POP Partners

## Who?

- BSC, ES (coordinator)
- HLRS, DE
- IT4I, CZ
- JSC, DE
- NAG, UK
- RWTH Aachen, IT Center, DE
- TERATEC, FR
- UVSQ, FR

## A team with

- Excellence in performance tools and tuning
- Excellence in programming models and practices
- A research and development background and a proven commitment to real academic and industrial applications

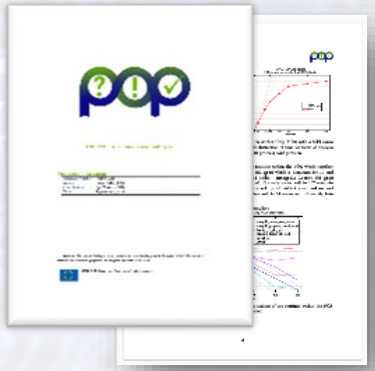
# POP Tools

- A number of profiling tools are developed by POP partners
  - BSC Tools – Extrae, Paraver and Dimemas
  - Score-P and Scalasca
  - MAQAO
  - PyPOP
- Further development of these tools will take place as part of POP, with a view to improving usability.
- The POP website provides online training and documentation for these and other useful performance tools.

# FREE Services provided by the CoE

## Parallel Application Performance Assessment

- Primary service
- Identifies performance issues of customer code
- If needed, identifies the root causes of the issues found and qualifies and quantifies approaches to address them (recommendations)
- 1-3 months effort

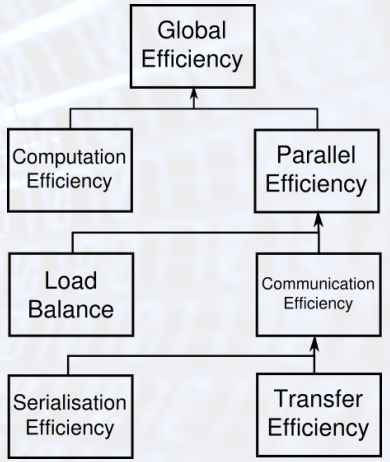


## Proof-of-Concept

- Follow-up service
- Experiments and mock-up tests for customer codes
- Kernel extraction, parallelisation, mini-app experiments to show effect of proposed optimisations
- 3-6 months effort

# POP Metrics

- The following metrics are used in a POP Performance Audit of a pure MPI code:
- Global Efficiency (GE): GE = PE * CompE
  - Parallel Efficiency (PE): PE = LB * CommE
    - **Load Balance** Efficiency (LB): LB = avg(CT)/max(CT)
    - **Communication** Efficiency (CommE): CommE = SerE * TE
      - Serialization Efficiency (SerE): SerE = max (CT / TT on ideal network)
      - Transfer Efficiency (TE): TE = TT on ideal network / TT
  - (Serial) **Computation** Efficiency (CompE)
    - Computed out of IPC Scaling, Instruction Scaling and Frequency Scaling
    - For strong scaling: ideal scaling -> efficiency of 1.0



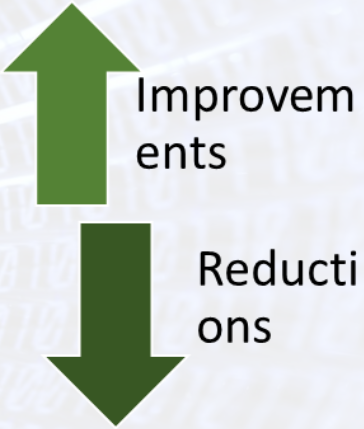

CT = Computational time
TT = Total time

# POP Metrics Example

| | *MPI Processes* | | | |
|---|---|---|---|---|
| | 6 | 12 | 24 | 48 |
| **Global Efficiency** | **92%** | **95%** | **62%** | **49%** |
| Parallel Efficiency | 92% | 89% | 50% | 40% |
| Load Balance | 95% | 94% | 88% | 88% |
| Communication Efficiency | 96% | 95% | 57% | 45% |
| Serialisation Efficiency | 97% | 96% | 76% | 62% |
| Transfer Efficiency | 99% | 99% | 75% | 73% |
| Computational Scalability | 100% | 106% | 124% | 123% |
| IPC Scalability | 100% | 106% | 127% | 130% |
| Instruction Scalability | 100% | 99% | 98% | 96% |
| Frequency Scalability | 100% | 101% | 97% | 95% |

# Some Success Stories

- See ⇨ https://pop-coe.eu/blog/tags/success-stories





Performance Improvements for SCM's ADF Modeling Suite


**3x Speed Improvement** for zCFD Computational Fluid Dynamics Solver


**25% Faster time-to-solution** for Urban Microclimate Simulations


**2x performance improvement** for SCM ADF code


Proof of Concept for BPMF leads to around **40% runtime reduction**


POP audit helps developers **double their code performance**


**10-fold scalability improvement** from POP services


POP performance study improves performance **up to a factor 6**


POP Proof-of-Concept study leads to **nearly 50% higher performance**


POP Proof-of-Concept study leads to **10X performance improvement** for customer

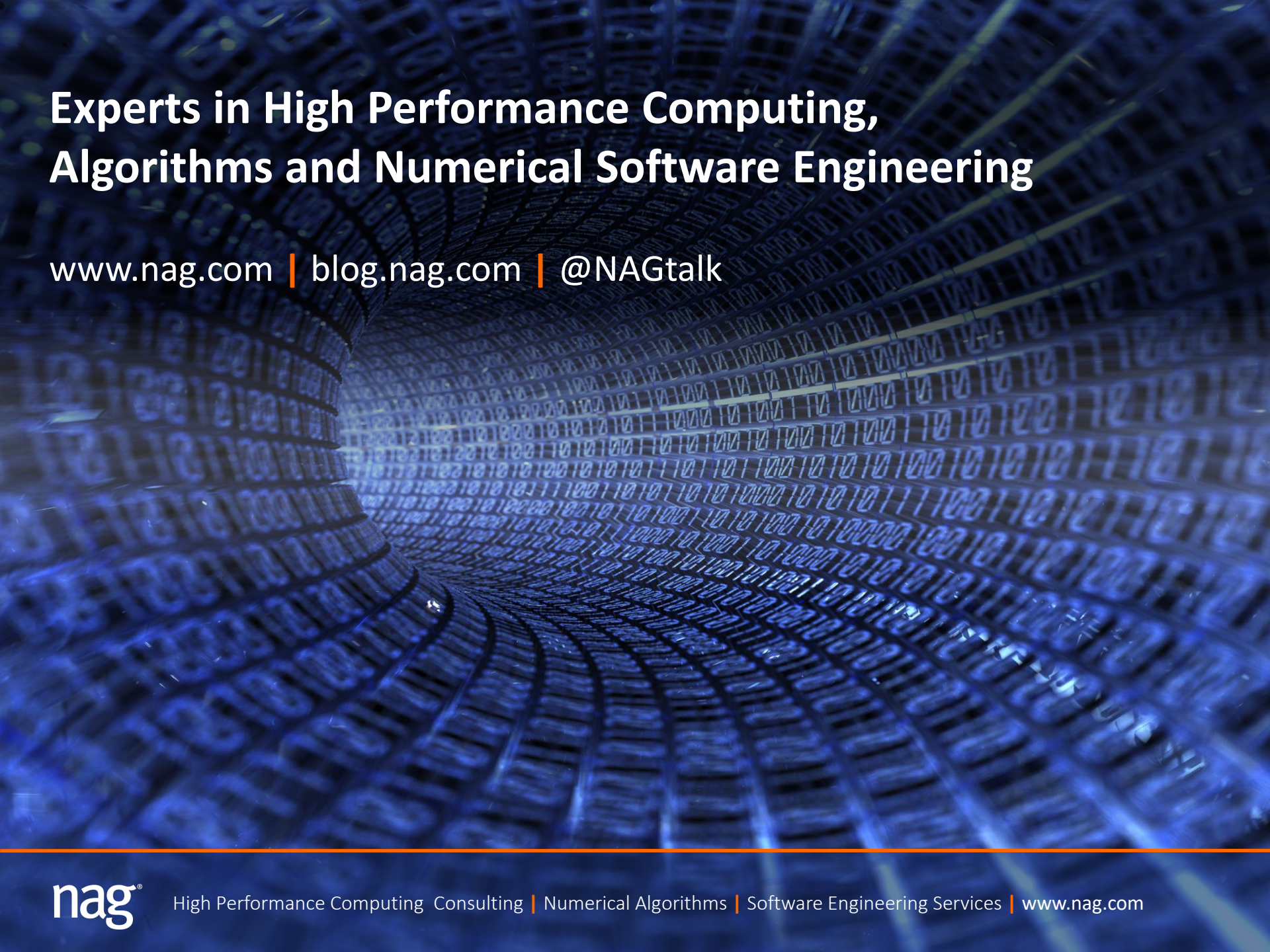
# Accessing POP Services

- If you're a code developer or user interested in a free performance assessment of a code, you can sign up to the service directly via the POP website.
  - Feel free to contact us first to discuss the service and what might be possible.
- Alternatively, if you're part of a service with a number of candidate codes on your systems, we'd be happy to discuss how we might work together.
- If you're hosting or know of any events which we could attend to inform people about our services, then let us know.

# Join the POP Community

- Browse the POP website at https://pop-coe.eu

  and subscribe to the newsletter.
- Follow us on twitter @POP_HPC
- Subscribe to https://www.youtube.com/POPHPC
  - Webinars, training videos, POPCasts.
- Join the LinkedIn group



# Experts in High Performance Computing, Algorithms and Numerical Software Engineering

www.nag.com | blog.nag.com | @NAGtalk