

NAG Library --> NAG Adjoint Library
Lawrence Mulholland
Numerical Algorithms Group (NAG) Ltd

Advances In Numerical Linear
Algebra: Celebrating the centenary
of the birth of James H. Wilkinson

School of Mathematics,
University of Manchester
29th May 2019

Lawrence Mulholland
NAG Ltd



Experts in numerical software and
High Performance Computing

Major thanks to:

stce.rwth-aachen: Uwe Neumann

Aachen/Nag: Klaus Leppkis

Johannes Lotz

Nag: Viktor Mosenkis,
Martyn Byng,
David Carlisle



Experts in numerical software and
High Performance Computing

NAG AD Library: Why, what and how

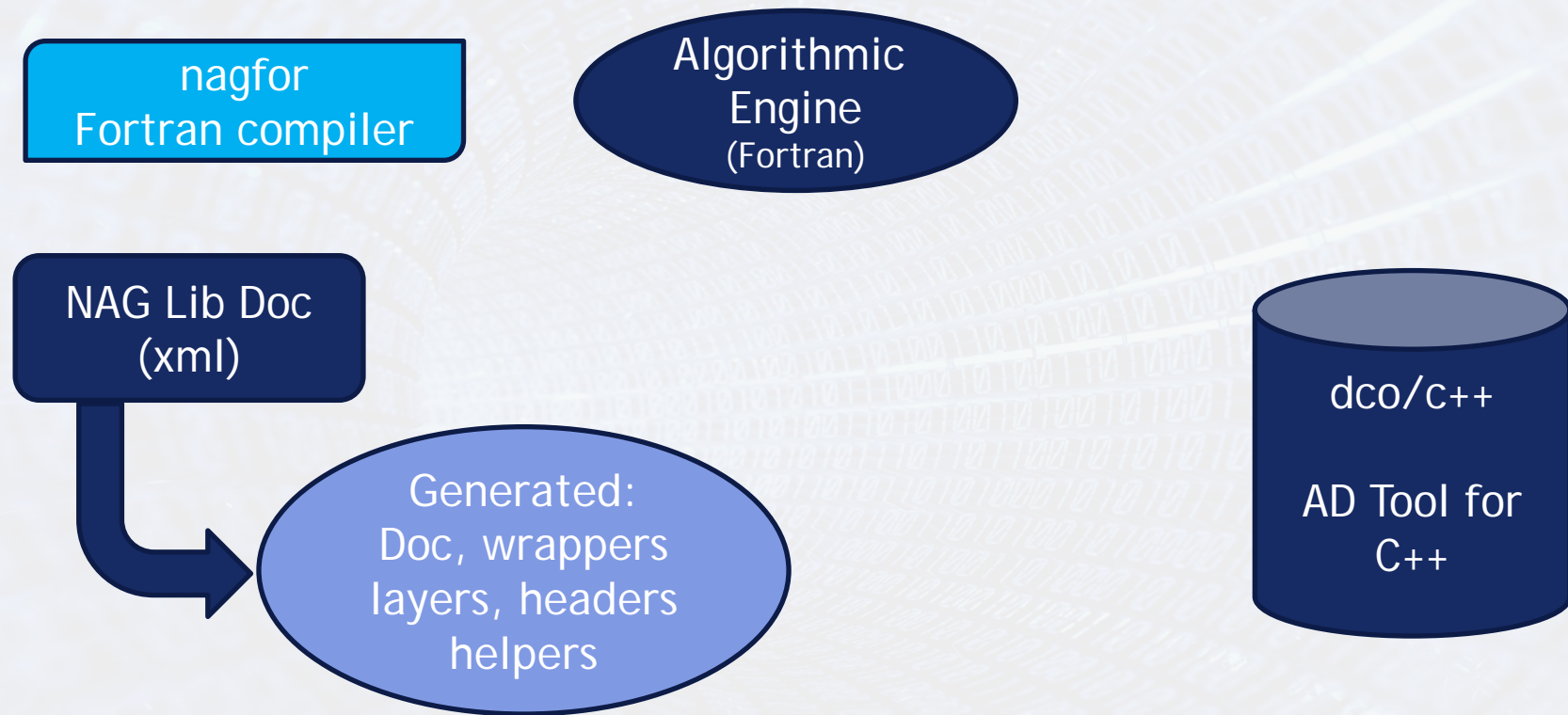
Why

- ▶ Because important NAG customers want it
 - Measure sensitivity of results to changes in inputs
 - Customers have large codes calling out to NAG Routines
 - New customers?

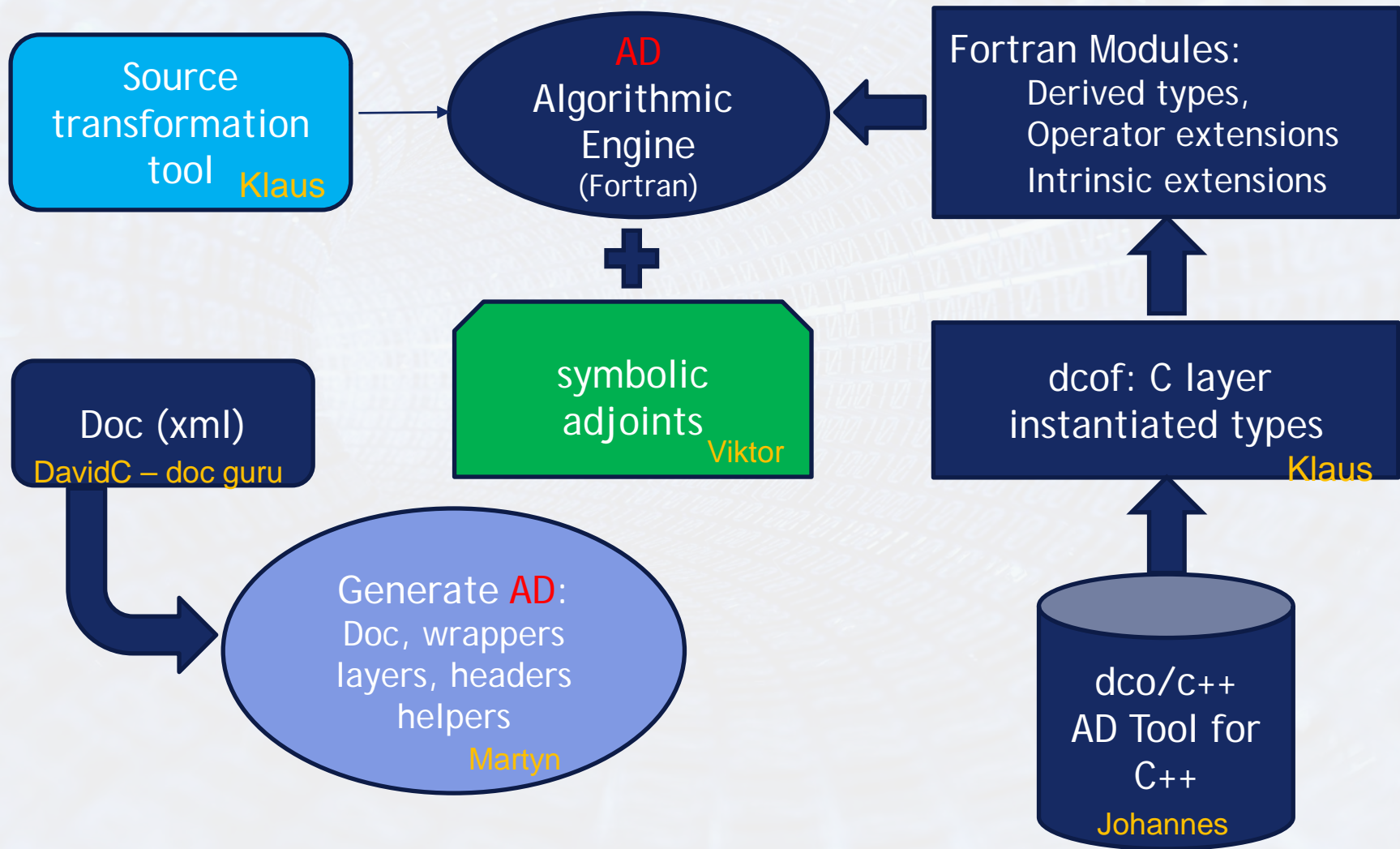
What

- ▶ AD versions of NAG Routines (Fortran C/C++) in:
 - Nonlinear systems, quadrature, Interpolation
 - Linear Algebra (Lapack, Blas, Sparse systems, nearest correlation)
 - Optimization
 - Statistics, special functions
 - Tangent Linear Derivatives and Adjoint (some symbolic)

NAG AD Library: How? -- From here



NAG AD Library: How? -- via



NAG AD Library: Peek at Doc

[MK27 Doc](#)

NAG AD Library: Linear Algebra

- Adjoint from Bottom Up
 - AD Versions of BLAS (BLAS_AD Lib ?)
 - Relatively Easy, some symbolic adjoints trivial
 - AD versions of Lapack
 - can use algorithmic or symbolic BLAS_AD
- Performance
 - AD versions of common sub-tasks written as calls to BLAS
 - e.f. gemm, symm, trmm, trsm
 - PLASMA (OpenMP tasks, Tiled)
 - --> DAG of tasks + AD of tasks --> fast adjoint
- Primal code problems
 - `BIGNUM = ONE / SMLNUM`
 - 58 D & 53 Z Lapack routines
 - e.g. DGGEV3

NAG AD Library: Roadmap

- 2019 Dec
 - Another 64 routines adjoint and tangent linear
- 2020
 - Symbolic (Bottom up)
 - Most used BLAS, Linear solvers
 - Algorithmic
 - Lots
 - Second-order ? Mixed adjoint-tangent ?
- 2021
 - Symbolic
 - Further Linear Algebra
 - A few higher level routines
 - Algorithmic
 - Lots more (Good functionality coverage)

NAG AD Library - Summary

- Challenging task:
 - Square peg, round hole, big hammer, precision tools
 - Easy user migration from NAG Library to AD
 - Algorithmic and symbolic AD
 - C++ and Fortran
- It works 😊