ExCALIBUR-NEPTUNE

Release v0.0.0

ExCALIBUR-NEPTUNE Authors

Jan 18, 2021

CONTENTS:

1 Acronyms 3

2 Symbols 9

3 ExCALIBUR NEPTUNE Charter 11

4 Writing Documentation 13

5 Indices and tables 15

This will be the main location of the documentation for the whole ExCALIBUR-NEPTUNE project.

CHAPTER

ONE

ACRONYMS

Acronym	Description
ACM	
ADC	
ADM	Alternating directions method
AIC	Akaike information criterion
ALM	Augmented Lagrange multiplier
AMR	
AMReX	
AND	
ANL	Argonne National Laboratory
ANN	Artificial neural network
ANN	Artificial neural network
ANOVA	Analysis of Variance
API	Application Programming Interface
ARMA	Autoregressive moving averaoe
ARMAX	Autoregressive moving average with exogenous input
ASQ	Adaptive sparse quadrature
ATS	Advanced Terrestrial Simulator, previously Arctic Terrestrial Simulator
AUXVAL	
BD	
BG/L	
BIC	Bayesian information criterion
BIM	Empirical interpolation method
BOX	
BP	
BPOD	Balanced proper orthogonal decomposition
CabanaMD	
CCA	Canonical correlation analysis
CD	
CEA	
CESM	
CFD	Computational fluid dynamics
CI	
CNN	Convolutional neural network
COGENT	
COMPAT	
CoSaMP	Compressive sampling matching pursuit
COSMO	

continues on next page

Table 1 – continued from previous page

Acronym	Description
COSSAN	
CPP	
CPU	Central Processing Unit
CRC	
CS	Compressed sensing
CSE	
CSMP	
CTO	
CUDA	Compute Unified Device Architecture
CWIPI	
CWT	Continuous wavelet transform
DAG	Direct Acyclic Graph
DAKOTA	
DCT	Discrete cosine transform
DDA	
DEIM	Discrete empirical interpolation method
DFT	Discrete Fourier fransform
DFT	Discrete Fourier Transform
DiMDc	Dynamic mode decomposition with control
DL	Deep learning
DMD	Dynamic mode decomposition
DMDc	Dynamic mode decomposition with control
DNS	Direct numerical simulation
DOE	Department of Energy
DOI	Digital Object Identifier
DSL	Domain-Specific Language
DWT	Discrete wavelet transform
ECOG	Electrocorticography
ECP	
ECP-copa	
eDMD	Extended DMD
EIRENE	
EM	Expectation maximization
EOF	Empirical orthogonal functions
ERA	Eigensystem realization algorithm
ESC	Extremum-seeking control
ESI	
ESMF	
ETS	
EU	European Union
FACETS	
FCI	Flux-Coordinate Independent (methods)
FEM	Finite Element Method
FEniCS	
FFT	Fast Fourier transform
FFT	Fast Fourier Transform
FFTW	Fastest Fourier Transform in the West (library)
FLASH	
GA	General Atomics

continues on next page

Table 1 – continued from previous page

Acronym	Description
GBD	
GBS	
GDB	
GMM	Gaussian mixture model
GMRES	Generalized Minimal Residual method
GNU	GNU's Not Unix!
gPC	Generalised polynomial chaos
GPU	Graphics Processing Unit
GSA	Global sensitivity analysis
GUI	Graphical User Interface
HAVOK	Hankel alternative view of Koopman
HDF5	Hierarchical Data Format (version 5)
HLA	
HPC	High Performance Computing
HVAR	
ICA	Independent component analysis
ICON	
IEEE	
IMAS	
IMEX	Implicit-Explicit Methods
IO	
ITG	Ion Temperature Gradient
ITM	Ion Tearing Mode
ITPA	
JET	Joint European Torus
JL	JohnsonLindensfrauss
JOREK	
KL	KullbackLeib1er
KLT	Karhunen-Loeve transform
LAD	Least absolute deviations
LAMMPS	Large-scale Atomic/Molecular Massively Parallel Simulator
LANL	Los Alamos National Laboratory
LASH	
LASSO	Least absolute shrinkage and selection operator
LASSO	Least Absolute Shrinkage and Selection Operator
LDA	Linear discriminant analysis
LGPL	GNU Lesser General Public License
LHSamp	
LLNL	Lawrence Livermore National Laboratory
LQE	Linear quadratic estimator
LQG	Linear quadratic Gaussian controller
LQR	Linear quadratic regulator
LTI	Linear time invariant system
MAP	Maximium A Posteriori
MC	Monte Carlo (methods)
MCMC	Markov Chain Monte Carlo
MCT	
MESAGE	
MFMA	

continues on next page

Table 1 – continued from previous page

Acronym	Description
MFMC	Multi-fidelity Monte Carlo
MHD	Magnetohydrodynamics
MIMC	
MIMO	Multiple input, multiple output
MIT	Massachusetts Institute of Technology
ML	Machine Learning
MLC	Machine learning control
MLMC	Multi-level Monte Carlo
MLMF	
MMF	
MOOSE	
MOR	Model Order Reduction
Most	Common Acronyms
MPE	Missing point estimation
MPI	Message Passing Interface
mrDMD	Multi-resolution dynamic mode decomposition
MSSC	
MUMPS	
MUSCLE 3	Multiscale Coupling Library and Environment 3
NARMAX	Nonlinear autoregressive model with exogenous inputs
NEMO	
NEPTUNE	
NLS	Nonlinear Schrdinger equation
NUCODE	
OASIS	
OASIS4	
ODE	Ordinary differential equation
OKID	Observer Kalman filter identification
OLYMPUS	
OMFIT	
OU	Oxford University
OUU	Optimisation under uncertainty
OUUWA	
PASTIX	
PBH	PopovBelevitchHautus test
PC	Polynomial chaos
PCA	Principal components analysis
PCE	Polynomial chaos expansion
PCP	Principal component pursuit
PDE	Partial differential equation
PDE-FIND	Partial differential equation functional identification of nonlinear dynamics
PDF	Probability distribution function
PFC	
PGD	
PICPIF	
PID	Proportional-integral-derivative control
PIV	Particle image velocimetry
POD	Proper orthogonal decomposition
POOMA	

continues on next page

Table 1 – continued from previous page

Acronym	Description
PP20	
PRESET	
QA	
QCG	
QMC	Quasi-Monte-Carlo
QoI	Quantity of interest
RIP	Restricted isometry property
RKF23	
RKHS	Reproducing kernel Hilbert space
RNG	
RNN	Recurrent neural network
ROM	Reduced order model
ROM	Reduced-order model
RPCA	Robust principal components analysis
rSVD	Randomized SVD
SAMRAI	
SGD	Stochastic gradient descent
SIAM	Society for Industrial and Applied Mathematics
SINDy	Sparse identification of nonlinear dynamics
SISO	Single input, single output
SLSQT	Sequential Least-Squares' Thresholding
SMARDDA	
SMART	
SMITER	
SMwiki	
SNOWPAC	Stochastic Nonlinear Optimisation with Path-Augmented Constraints (software package)
SOL	Scrape-Off Layer
SOLEDGE	
SOLPS	
SRC	Sparse representation for classification
SRS	
SSA	Singular spectrum analysis
STARWALL	
STFT	Short time Fourier transform
STIXGeneral	
STLS	Sequential thresholded least-squares
STRUMPACK	
SVD	Singular value decomposition
SVD	Singular value decomposition
SVM	Support vector machine
SVM	
TAE	
TICA	Time-lagged independent component analysis
TM	
TOKAM	
TOKAM3X	
TOMS	
TRIMEG	
TUM	

continues on next page

Table 1 – continued from previous page

Acronym	Description
UK	United Kingdom
UKAEA	United Kingdom Atomic Energy Authority
UKRI	United Kingdom Research and Innovation
UQ	Uncertainty quantification
US	
USA	
UTF-8	
VAC	Variational approach of conformation dynamics
VDE	
VECMAtk	
VORPAL	
XGC1	
XML	
XMSF	

CHAPTER

TWO

SYMBOLS

Symbol	Description
$[a, b]$	arbitrary finite interval
d	number of dimensions over which the integral is performed
f_0	constant in the expansion of $f\left(x_1, \ldots, x_d\right)$
$f\left(x_1, \ldots, x_d\right)$	joint probability distribution
$f_i(x_i)$	coefficient in the expansion of $f\left(x_1, \ldots, x_d\right)$
$f_{ij}(x_i, x_j)$	coefficient in the expansion of $f\left(x_1, \ldots, x_d\right)$
$p(x)$	probability distributions
r	order of higher order term
x_i	generic parameter or variable
$\mathbf{x} = (x_1, x_2, \ldots, x_d)$	is a d-dimensional vector
$P(x)$	Cumulant probability distribution
$\parallel Q \parallel_E$	the ‘energy’ norm
S_i	Sobol sensitivity index, gives a normalised measure of the sensitivity of the distribution of f to the parameter x_i
S_{ij}	Sobol sensitivity index, gives a normalised measure of the sensitivity of the distribution of f to the parameters x_i and x_j
$\mathrm{Var}(f)$	variance of the distribution of f computed by integrating over all variables x_i
V_i	variance of the distribution of f as the parameter x_i varies
V_{ij}	variance of the distribution of f as the parameters x_i and x_j vary
$\mathbb{E}$	expectation
$\mathbb{E}_{x_{k \neq i}}$	expectation computed by integrating over all the x_k except for x_i
$\mathbb{E}_{k \neq i, l \neq j}$	expectation computed by integrating over all the x_k except for x_i and x_j
ξ_i	randon number within the unit interval $[0, 1]$

CHAPTER

THREE

EXCALIBUR NEPTUNE CHARTER

All members of the ExCALIBUR NEPTUNE team should be aware that to meet the challenges of the NEPTUNE project, and the ExCALIBUR overarching pillars, a distributed team of scientists, software engineers and architecture specialists from different UK institutions will be required to form a community around the NEPTUNE project (and will connect across the overarching ExCALIBUR programme). A high-level objective is to ensure that developed software is of the highest quality, implying a rigid requirement around the production of high-quality documentation and reproducible verification and validation tests for the codebase as it evolves. Since development work may transfer between institutions, it is important that common standards for documentation and testing be available and easy to deploy. The initial NEPTUNE exploratory Proxyapps may be written in a range of languages including for example Python, C++/DPC++, Object Fortran or Julia, however it is envisaged that there will be an emerging steer towards a reduced set of languages and technologies to ensure interoperability across the NEPTUNE software stack, ultimately leading to coupled simulations covering all the physics necessary to deliver an "actionable" simulation for the plasma edge. It is not yet clear for example whether SYCL, Kokkos or OpenMP 5 will offer the most performance portable and sustainable solution for NEPTUNE. The team is therefore expected to be agile and amenable to change once it is clear which are the most promising long–term solutions. For example, a selection of SYCL for the long-term framework/code(s) would force refactoring of any code that is not consistent with a NEPTUNE library and code base instantiated in DPC++, and where feasible, team members should support this process.

Source code for all development should be accessible by the entire NEPTUNE team and all tests should be repeatable by different workers without the need for re-training and/or any possible confusion as to the procedures and metrics needed to declare a test successful.

NEPTUNE will be developed as a sequence of 'core' Proxyapps (to be distinguished from other Proxyapps designed to test some novel technique). Core Proxyapps will all need a documentation and testing framework, which must be agreed between all partners for the entire project. This will require developers to work closely with UKAEA and other team members.

A commitment is also expected by all parties to help UKAEA and the Met Office (as SRO for ExCALIBUR) to publicise the project and build a fully connected community across the ExCALIBUR programme, UKRI and Academia, focused upon a team of approximately twenty UK Fusion use case experts. This will be essential for meeting the grand challenge goal of developing a state-of-the-art, Exascale targeted, UK-based plasma physics simulation capability for the tokamak edge plasma (see Science Plan[1]).

All Core Proxyapps and related infrastructure/documentation across the NEPTUNE project should meet the demands of the Code Structure and Coordination work package FM-WP4 in so far as the developing project standards:

- adopt a consistent choice of definitions (ontology) of objects or equivalently classes,
- adhere to clearly defined common file formats and interfaces to components for data input and output.
- provide suitably flexible data structures for common use by all developers,
- are established through good scientific software engineering best practice,

[1] W. Arter, L. Anton, D. Samaddar, and R. Akers. ExCALIBUR Fusion Modelling System Science Plan. Technical Report CD/EXCALIBUR-FMS/0001, UKAEA, 2019.

- demonstrate performance portability and exploit agreed DSL-like interfaces where possible targeting Exascale-relevant architectures,
- can be integrated into a VVUQ framework and
- are embedded within a coordination and benchmarking framework for correctness testing and performance evaluation.

In order to meet Strategic Priorities Fund terms around eligibility, and to steer the project towards a modular platform where developments across all partners can be integrated into an eventual code or platform available for open use by the European fusion community, a requirement is that all ExCALIBUR NEPTUNE Grant beneficiaries make technology / source code developed through the programme (foreground Intellectual Property) available as open source under a programme–wide permissive license (currently selected as 3-clause BSD for core/foundational infrastructure). Government Digital Service guidance (to which the project subscribes), discussing the benefits of open versus closed technology/software/data can be found in: https://www.gov.uk/government/publications/open-source-guidance/when-code-should-be-open-or-closed and https://www.gov.uk/guidance/be-open-and-use-open-source.

GLOSSARY:

Table 1: Glossary of terms

Term or Acronym	**Definition**
C++	Programming language, see https://isocpp.org/
DPC++	Data Parallel C++, Intel compiler for C++ with SYCL extension
DSL	Domain Specific Language, programming language developed for a specific area of resrearch and development
Object Fortran	Programming language, more precisely Fortran 95 or later exploiting object-oriented features of the language, see https://wg5-fortran.org/f2008.html
Python	Programming language, see https://python.org/
SYCL	C++ library for portable high performance computing, see https://www.khronos.org/sycl/
Kokkos	C++ library for portable high performance computing https://cfwebprod.sandia.gov/cfdocs/CompResearch/docs/Kokkos-Multi-CoE.pdf
OpenMP	Software for parallel programming
SRO	Senior Responsible Owner role in UK government project delivery
Julia	Programming language, see https://julialang.org/
UKRI	United Kingdom Research and Innovation, a non-departmental public body encompassing the research councils and Innovate UK
VVUQ	Verification, Validation and Uncertainty Quantification

CHAPTER

FOUR

WRITING DOCUMENTATION

This documentation is written using reStructuredText (RST) and built using Sphinx on ReadTheDocs. For a good introduction on writing RST, see the reStructuredText primer in the Sphinx docs.

CHAPTER

FIVE

INDICES AND TABLES

- genindex
- modindex
- search