

FORGE: Re-generating a forecast system for sustainability

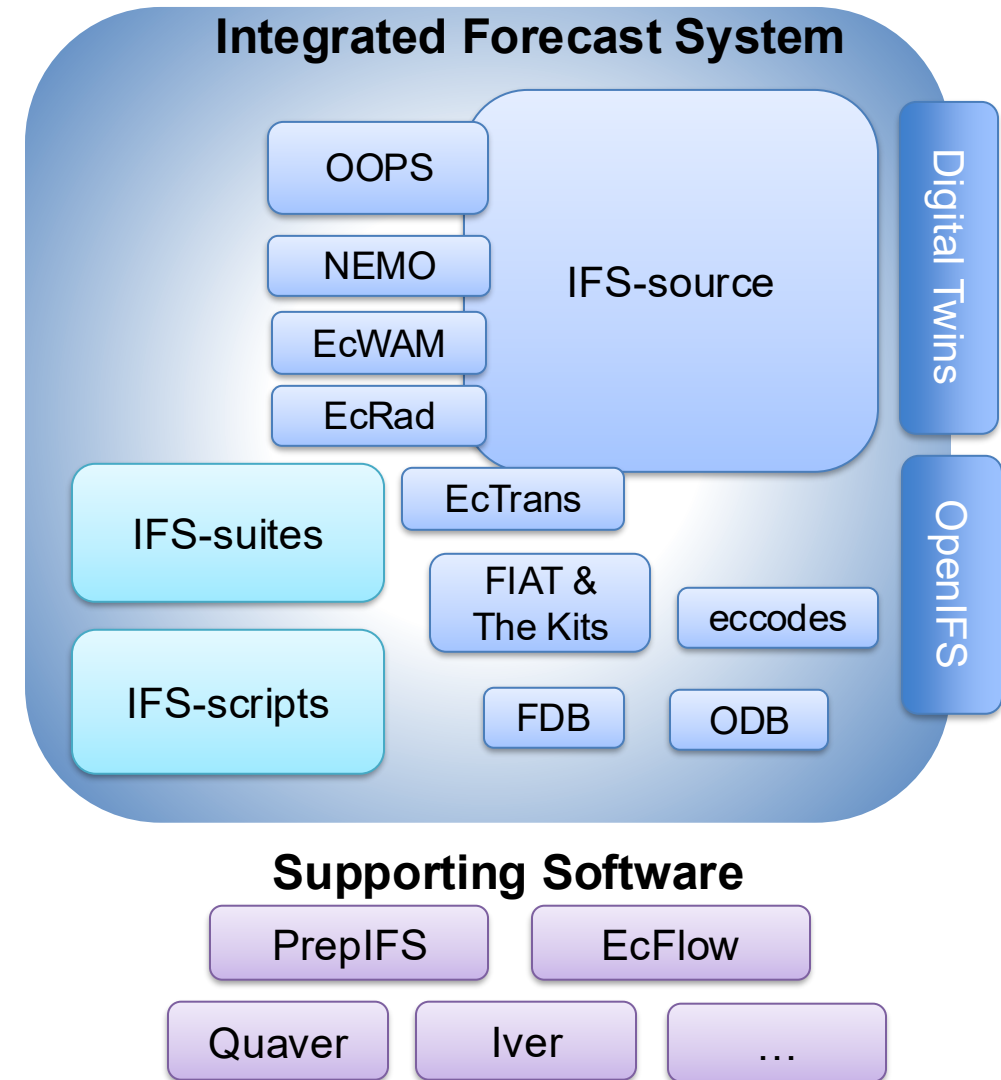
21st ECMWF workshop on high performance computing in meteorology

Michael Sleigh, **Michael Lange** (presenting), Andrew Bennett, Paul Burton, Zak Kipling, Ahmad Nawab



The IFS today

- The IFS is an extremely complex system
 - Development driven by meteorological quality
 - Large range of applications and expanding further
 - 4DVar, ensemble forecasting, ensemble DA
 - Extended and seasonal, atmosphere/ocean/land reanalysis
 - Atmospheric composition and GHG forecasting/reanalysis
 - Flood and fire forecasting, extreme weather and climate
- Need to adapt to a changing environment
 - External data centres and HPC machines (with GPUs)
 - AIFS, data-driven methods and hybrid ML



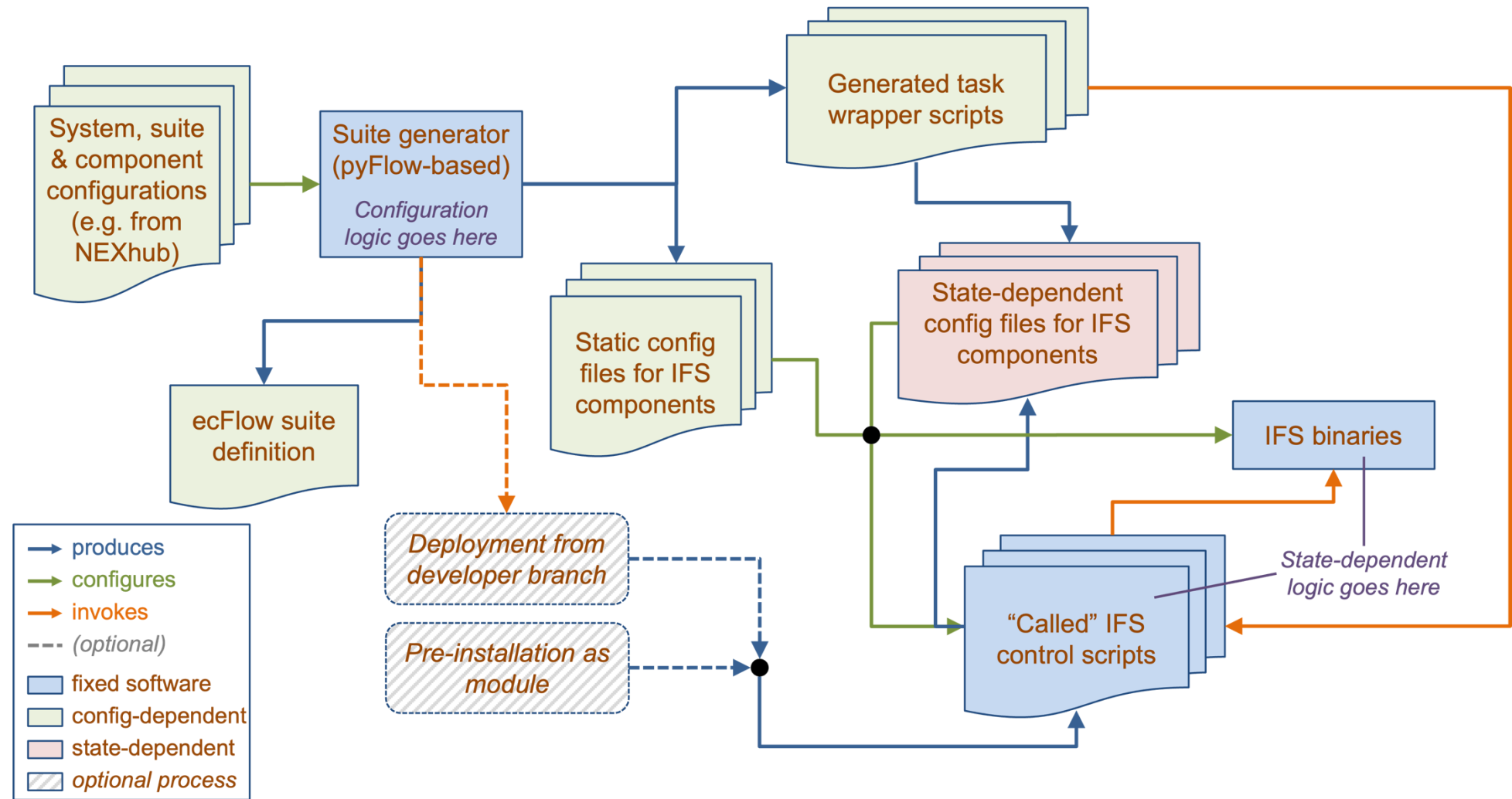
The IFS today

- The challenge
 - Motivation, scope – what are we doing, and why
 - Technical debt / entropy in software
 - Our forecasting systems
 - Computing landscape / environment, new methods
- The strategy
 - Software engineering principles
 - Separation of concerns / encapsulation
 - Testing
 - Added value to Member States

Forecast System Regeneration

- A major initiative to regenerate the IFS code
- Five-year timeframe, 2025-2029
- Covering:
 - **WP1: Workflow Code**
 - **WP2: Model Code Refactoring**
 - **WP3: Object-oriented data structures**
 - **WP4: Infrastructure**
 - **WP5: Processes and Workflows**

IFS Workflow Code: overview



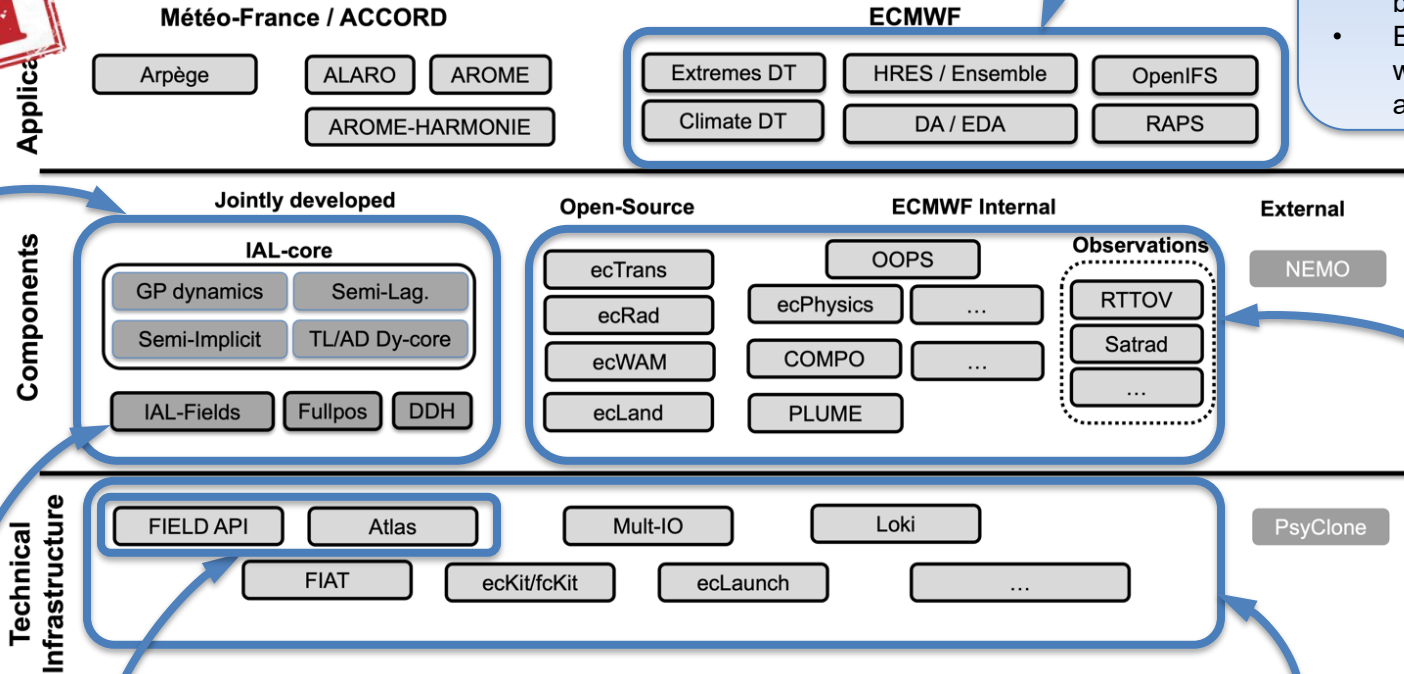
Proposed structure of the compiled IFS code base

DRAFT

WARNING: This is a draft!
We will improve contours with input from scientists and Member States

Release IFS cycles

- No change to nature of scientific cycle releases!
- Component version tested before integration
- External contributions welcome (PRs), but vetted and tested by ECMWF



Jointly developed code

- Coordinated developments
- Individual organizational forks, but frequently sync'd
- Increased test coverage (not just IFS/Arpège)
- PR review and early assessment made easier

ECMWF-owned code

- ECMWF manages code
- Open-source or closed source (but open to MS)
- Clean interfaces for external use; tested(!)
- External contributions welcome (PRs), but vetted and tested by ECMWF

Data structures

- Underlying foundation for all components
- Interface between science components and technical layer
- Requires special attention (next slide)

Technical infrastructure

- Separate scientific from technical code
- Encapsulation and specialisation
- Dedicated performance optimization
- More detailed testing and test coverage

IFS data structures – FIELD/VARIABLE abstraction layer

Shared data layout across IFS-Arpège

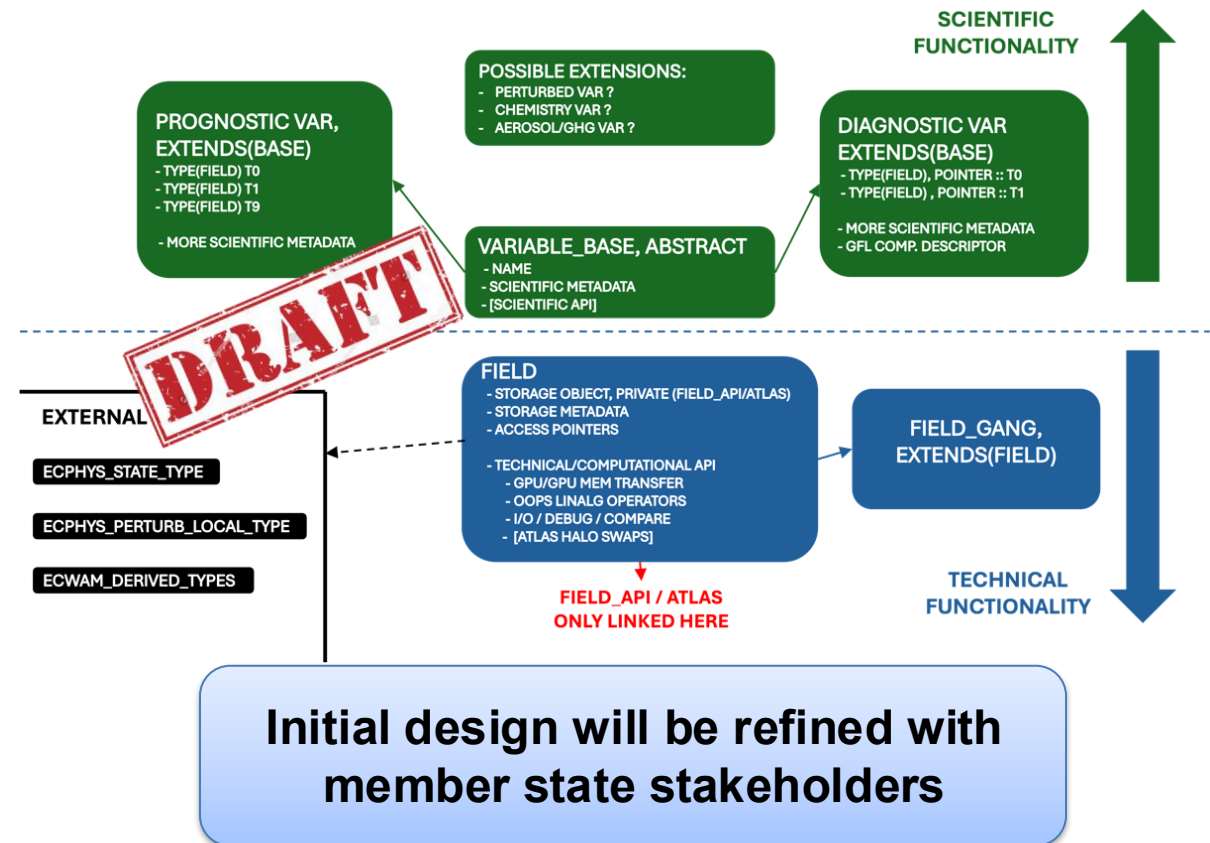
- Object-oriented field abstractions
 - Technical extensions for GPU porting
- Used in shared and non-shared modules

Object-oriented data structures for science

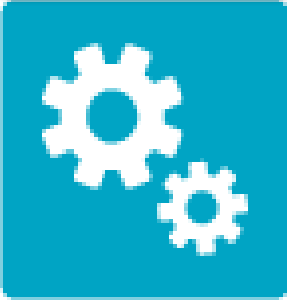
- Recent addition of Field-API was technical
- How can we improve the science coding?
 - Shared utilities across modules
 - Robustness and testing

Atlas as an abstraction for grid data

- Field interpolation with GPU-support
- Support for many grid types



ExpERT



Experiment Edit & Run Tool (prelFS replacement)

- Find experiments (all users)
- Copy & Create experiments
- Edit experiments
- Run macros
- Rule based updates (checks)
- Compare experiments
- Submit experiments

- Largely invisible to users
- Provides services to plugin apps
- Databases for apps
- https API



DLM



Quaver



IVER



Plots



Performance



ecFlow View



ecFlow Viewer

- Limited functionality
- Shows state of experiment (read only)
- Tree-based view, no customisation

OUTDATED

Repo: <https://git.ecmwf.int/scm/ifs/ifs-defaults.git>
Branch: CY49R1

CHANGE

CY49R1 -
fc
Demo session

undo redo saved macros run checks publish

Search

Model uncertainty representations

NEMO physical settings for ocean/ice...

NEMOICE physical settings

Ocean general, resolution and sea-ice

Resolution, general experiment setup ...

Submit options

Accounting, system properties

Add your own Variables, Namelists, etc.

Archiving

Atmospheric composition

Basic geometry switches

Basic IFS control switches

Basic IFS control switches/namrip

Compiler options, debugging

Copy history

Coupled wave model setup

Dimensioning...

Dr Hook

External Package Versions

Full Pos postprocessing, main switches

Gridpoint variables and tracer mass fixer setti...

IO server control

Limited area diagnostics, global diagnostics

Main IFS control switches

Resolution, general experiment setup ...
Specify Levels

RESOL

Horizontal resolution
(spectral truncation) Change
this item in the popup menu
on the tree

1279

LEVELS

Vertical resolution (number
of levels)

137

GTYPE

Grid type

cubic octahedral reduced Gaussian grid

FLENGTH

Forecast length in hours

240

COUPLED

Coupled run with data-
assimilation

☐

LESUITE

Switch for esuite bit
reproducibility check

☐

CLIMVERSION

Which version of the climate
files to use

climate.v021

CLIM_PATH

PATH for climate fields - must
follow pattern

default

ECMWF | NEXhub

Help Paul Burton

Dashboard

ExpERT

ecFlow viewer

Ecflow Web Viewer

io1o

io1y

io1s

io1r

io1q

io1p

io1o

an

make

obs

main

lw00

lw12

an

black

odb

preCleanFDB

vardata

fetcherr

lowres

ocean_initial

4dvar

uptraj_0

oopsvar

uptraj_1

uptraj_2

uptraj_3

restart_999

surf_anal

forceinv

emiskf

aeolus_2c

INFO

JOB

SCRIPT

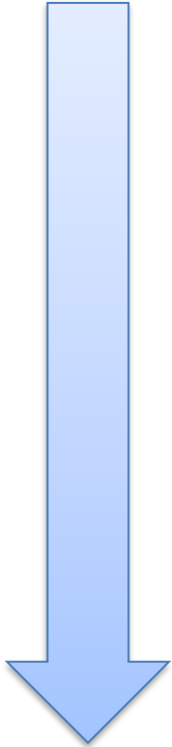
OUTPUT

VARIABLES

Name	Value	Inherited from
LSCREEN_IN_OOPS	true	-
mode	trajmin	-
PRIORITY	71	-
TASK	oopsvar	-
ECF_JOB	/var/cache/ecflow/rd/dabio1o/an/main/lw12/an/4dvar/uptraj_0/oopsvar.job1	-
ECF_SCRIPT	/var/cache/ecflow/rd/dabio1o/an/main/lw12/an/4dvar/uptraj_0/oopsvar.ecf	-
ECF_JOBOUT	/ecflow1/sb/log/rd/dabio1o/an/main/lw12/an/4dvar/uptraj_0/oopsvar.1	-
ECF_TRYNO	1	-
ECF_RID	20585328	-
ECF_NAME	/dabio1o/an/main/lw12/an/4dvar/uptraj_0/oopsvar	-
ECF_PASS	p2YsaHF	-
FAMILY1	uptraj_0	uptraj_0
FAMILY	io1o/an/main/lw12/an/4dvar/uptraj_0	uptraj_0
uptraj	0	uptraj_0
I_EXTRACT		uptraj_0
USE_HUGEPAGE	False	4dvar
THREADS	16	4dvar
OCPERT	0	4dvar
NPES_CCMA	%NPPOOLS%	4dvar
NPES	%NPES_AN%	4dvar
NEMOVAR_NPROCY	6	4dvar
NEMOVAR_NPROCX	8	4dvar
NEMOVAR_NPROC	48	4dvar
HYPERTHREADING	False	4dvar
YMD	20220601	lw12
YMD_JULIAN	2459732	lw12
YMD_DOW	3	lw12
YMD_DD	1	lw12
YMD_MM	6	lw12
YMD_YYYY	2022	lw12

A multi-layered approach to testing

Modular,
targeted



System,
end-to-end

- Multiple levels of testing:

NEW

- **Unit/modular testing**

- Testing modularised components of Fortran/C/C++ source, shell scripts and ecFlow suites

- Targeted configuration testing - manually triggered

- Git ifstest, IFS experiments

- Automated

- Automatically triggered on pull-requests or events

- Continuous *Integration* (CI)

- IFS citest (running standard set of experiments, if all succeed automatically merging to master)

NEW

- **Continuous *Deployment* (CD)**

- Completely automated process from testing through to creating a release and deployment to D-suite/production (with appropriate human approval points)

- **D-suite**

- An end-to-end clone of the operational suite, running in real-time

Open collaborative development

- Since 2022 have been making components of IFS open-source under Apache-2 licence
- ECMWF strategy states: “... *in the spirit of ‘open science’ and ... to further collaboration (e.g. with academia) and better serve the scientific community, ECMWF will build on the successful OpenIFS efforts and move to an open-source approach for the whole of the forecast model*”



European Centre for Medium-Range Weather Forecasts (IFS)

Open-source components of and auxiliary software closely related to ECMWF's Integrated Forecasting System

108 followers · Shinfield Park, Reading, United Kin... · <https://www.ecmwf.int>

Overview · Repositories 21 · Projects · Packages · People 2

README.md
























Welcome to ECMWF IFS GitHub!

[ECMWF](#) is the European Centre for Medium-Range Weather Forecasts.

[IFS](#) is the Integrated Forecasting System.

Have a look at some of our popular topics:

- [loki](#)
- [ectrans](#)
- [ecrad](#)
- [fiat](#)

21 repositories		Last pushed	
 ectrans	Efficient and scalable spectral transforms	Fortran	42 23
 field_api		Fortran	12 4
 ecrad	ECMWF atmospheric radiation scheme	Fortran	43 49
 dwarf-p-cloudsc	Standalone mini-app of the ECMWF cloud microphysics parameterization	Fortran	26 11
 loki	Freely programmable source-to-source translation for Fortran	Python	17 31
 ifsbench	A pythonic benchmarking package for IFS development	Python	3 2
 ecwam	The ECMWF wave model ecWAM	Fortran	23 16
 ecland	ecLand	Fortran	3 2
 fiat	The Fortran IFS and Arpege Toolkit	Fortran	32 8
 easybuild-easyconfigs	A collection of easyconfig files that describe which software to build using which build options with EasyBuild.	Python	752 0
 ifs-arpege-coding-standards	Official coding standard for IFS / Arpège	Python	5 3
 CaMa-Flood_v4	CaMa-Flood_v4	Fortran	47 0
 dwarf-p-cloudsc2-tl-ad	A dwarf to test tangent-linear and adjoint versions of the CLOUDSC2 cloud microphysics scheme.	Fortran	5 0
 ecckd	ECMWF gas optics tool	C++	1 5
 easybuild-framework	EasyBuild is a software installation framework in Python that allows you to install software in a structured and robust way.	Python	211 0
 easybuild-easyblocks	Collection of easyblocks that implement support for building and installing software with EasyBuild.	Python	297 0
 ecaeropt	ECMWF aerosols optics package	Python	2 5
 .github	Default issue and PR templates for ECMWF-IFS repos		1 0
 ectrans-bench	Global spherical harmonics transforms library underpinning the IFS	Fortran	42 0
 docinv	IFS/Arpège documentation inventory		0 0
 ifs-scripts-os			1 1

FORGE - Forecast System Regeneration

- **A major initiative to regenerate the IFS code**
 - Tackling technical debt over the coming 5 years
 - Restructure model code and data structures
 - Modernize the workflows supporting software stack
- **Increased resilience in a changing environment**
 - Ease deployment and use on external systems
 - Ease the addition of new capabilities
- **Pave the way for further modernization efforts**
 - Workshop on code modernization for physical NWP models planned for 2026

Thank you! Any questions?

✉ michael.lange@ecmwf.int
🐦 [MLange805](https://twitter.com/MLange805)