



SCA Report for ICON

U. Schaettler & D. Rieger



Latest ICON Releases



ICON (open source)

- ICON-2026.04

ICONREMAP (not public)

- icontools-2.8.2: 15 April 2026
- icontools-2.8.3: 06 May 2026

← → ↺ 🏠 docs.icon-model.org

ICON Release Getting Started ICON Documentation Development Guidelines

Search Ctrl + K

ICON Model Documentation

ICON is a flexible, scalable, high-performance modelling framework for weather, climate and environmental prediction that provides actionable information for society and advances our understanding of the Earth's climate system.

Useful links: [NameList Overview](#), [ICON-Exchange](#), [Deprecated Features](#), [FAQ](#), [Community Newsletter](#)


Release

Get the latest release and information on changes to previous ICON versions.

[To the ICON release](#)


Getting Started

Don't know where to start? Find a short description for beginners here, including references to dig deeper.

[To the getting started guide](#)


Documentation

Collection of everything there is to know about ICON.

Chapters: [User Manual](#), [Interfaces & Tools](#), [Atmosphere](#), [Ocean](#), [Waves](#), [Land](#), [Infrastructure](#), [Literature](#)

[To the ICON documentation](#)


Contribute

Want to contribute to the ICON development or find out how the ICON partners guarantee the high standards of the ICON framework?

[To the development guidelines](#)

[Imprint, Copyright and Privacy Policy](#)

© Copyright 2026, The ICON Partners.
Created using [Sphinx](#) 9.1.0.

Built with the [PyData Sphinx Theme](#) 0.15.4.

ICON (open source)

- ICON-2026.04

ICONREMAP (not public)

- icontools-2.8.2: 15 April 2026
- icontools-2.8.3: 06 May 2026

The screenshot shows the homepage of the ICON Model Documentation website at docs.icon-model.org. The URL is highlighted with a blue box and a checkmark icon. Below the navigation bar, the 'Release' section is highlighted with a blue box. The 'Release' section contains the text 'Get the latest release and information on changes to previous ICON versions.' and a button labeled 'To the ICON release'. Other sections visible include 'Getting Started', 'Documentation', and 'Contribute'.

ICON Model Documentation

ICON is a flexible, scalable, high-performance modelling framework for weather, climate and environmental prediction that provides actionable information for society and advances our understanding of the Earth's climate system.

Useful links: [Learn about ICON](#), [ICON Exchange](#), [Deprecated Features](#), [FAQ](#), [Community Newsletter](#)

Release

Get the latest release and information on changes to previous ICON versions.

[To the ICON release](#)

Getting Started

Don't know where to start? Find a short description for beginners here, including references to dig deeper.

[To the getting started guide](#)

Documentation

Collection of everything there is to know about ICON.

Chapters: [User Manual](#), [Interfaces & Tools](#), [Atmosphere](#), [Ocean](#), [Waves](#), [Land](#), [Infrastructure](#), [Literature](#)

[To the ICON documentation](#)

Contribute

Want to contribute to the ICON development or find out how the ICON partners guarantee the high standards of the ICON framework?

[To the development guidelines](#)

[Imprint](#), [Copyright](#) and [Privacy Policy](#)

© Copyright 2026, The ICON Partners.
Created using [Sphinx](#) 9.1.0.

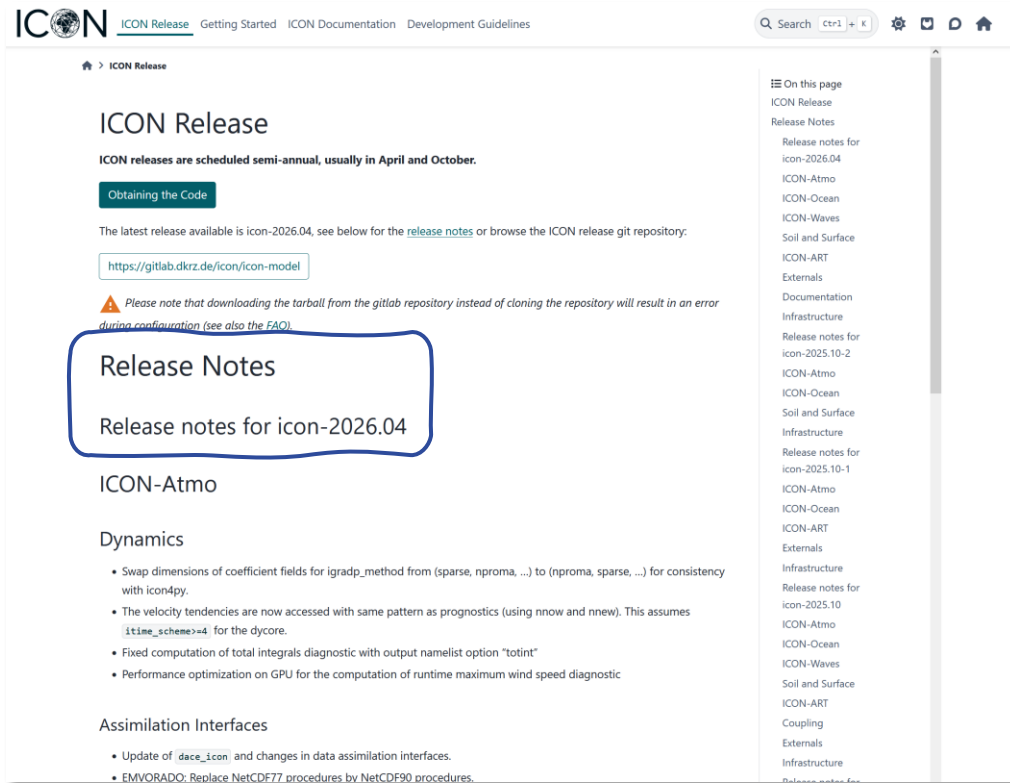
Built with the [PyData Sphinx Theme](#) 0.15.4.

ICON (open source)

- ICON-2026.04

ICONREMAP (not public)

- icontools-2.8.2: 15 April 2026
- icontools-2.8.3: 06 May 2026



The screenshot shows the 'ICON Release' page on the COSMO website. The page title is 'ICON Release'. Below the title, it states 'ICON releases are scheduled semi-annual, usually in April and October.' There is a button labeled 'Obtaining the Code'. Below this, it says 'The latest release available is icon-2026.04, see below for the [release notes](#) or browse the ICON release git repository.' A URL is provided: <https://gitlab.dkrz.de/icon/icon-model>. A warning icon and text state: 'Please note that downloading the tarball from the gitlab repository instead of cloning the repository will result in an error during configuration (see also the FAQ).' The 'Release Notes' section is highlighted with a blue box. Below it, the text 'Release notes for icon-2026.04' is visible. The 'ICON-Atmo' section is also visible. The 'Dynamics' section lists several bullet points: 'Swap dimensions of coefficient fields for igradp_method from (sparse, nproma, ...) to (nproma, sparse, ...) for consistency with icon4py.', 'The velocity tendencies are now accessed with same pattern as prognostics (using nnow and nnew). This assumes `itime_scheme>=4` for the dycore.', 'Fixed computation of total integrals diagnostic with output namelist option "totint"', and 'Performance optimization on GPU for the computation of runtime maximum wind speed diagnostic'. The 'Assimilation Interfaces' section lists two bullet points: 'Update of `dace_icon` and changes in data assimilation interfaces.' and 'EMVORADO: Replace NetCDF77 procedures by NetCDF90 procedures.' The right sidebar contains a table of contents with links to various sections like 'On this page', 'Release Notes', 'Release notes for icon-2026.04', 'ICON-Atmo', 'ICON-Ocean', 'ICON-Waves', 'Soil and Surface', 'ICON-ART', 'Externals', 'Documentation', 'Infrastructure', 'Release notes for icon-2025.10-2', 'ICON-Atmo', 'ICON-Ocean', 'Soil and Surface', 'Infrastructure', 'Release notes for icon-2025.10-1', 'ICON-Atmo', 'ICON-Ocean', 'ICON-ART', 'Externals', 'Infrastructure', 'Release notes for icon-2025.10', 'ICON-Atmo', 'ICON-Ocean', 'ICON-Waves', 'Soil and Surface', 'ICON-ART', 'Coupling', 'Externals', 'Infrastructure', and 'Release notes for icon-2025.10-1'.

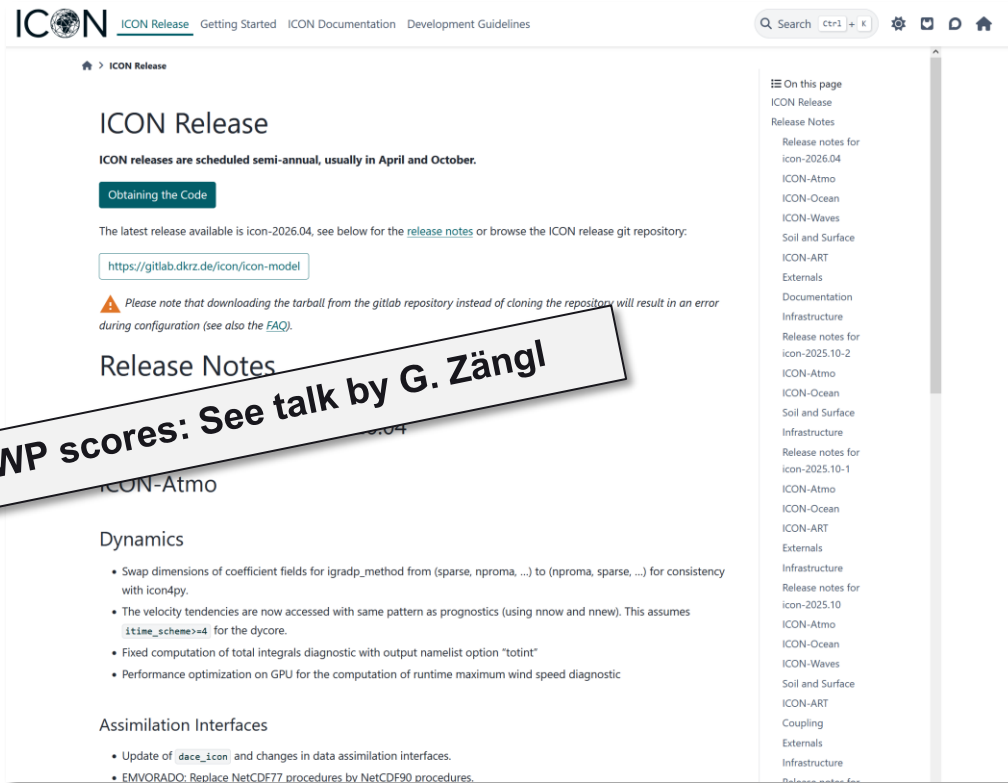
ICON (open source)

- ICON-2026.04

ICONREMAP (not public)

- icontools-2.8.2: 15 April 2026
- icontools-2.8.3: 06 May 2026

Changes with impact on NWP scores: See talk by G. Zängl





License & Support



Running ICON

- Brazil (Navy, INMET), Qatar, Oman, UAE

In preparation

- Botswana, Nigeria, Zimbabwe, Rwanda, Cyprus

Still need to migrate

- Turkmenistan (need new HPC)
- Malawi (in contact)
- Egypt (no contact right now)

Contacts to

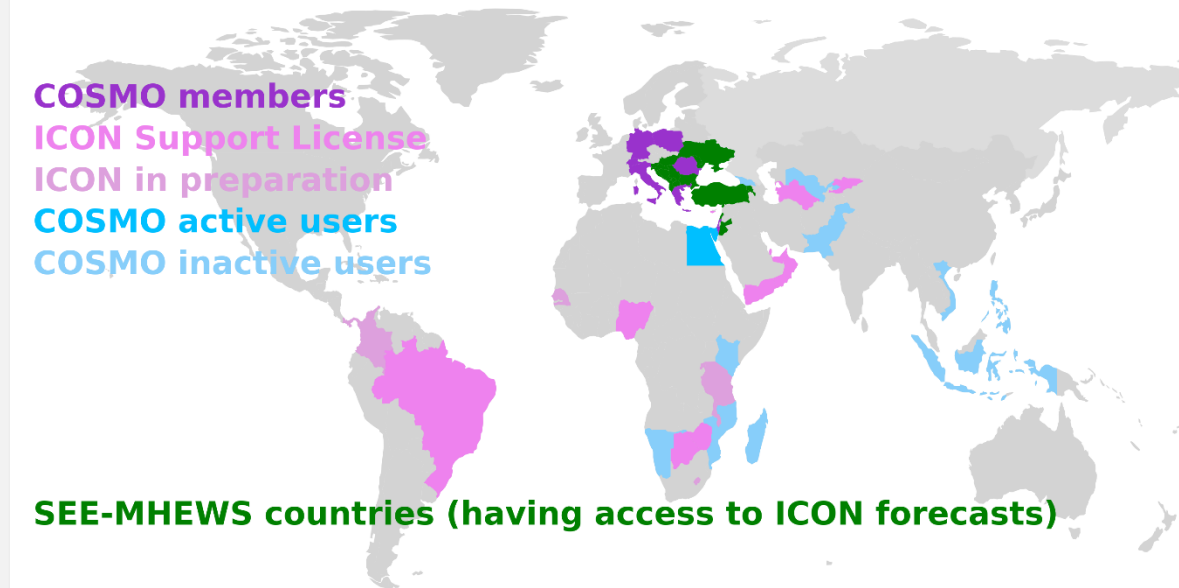
- Lesotho, Tanzania, Senegal

Possible paying countries

- In negotiation: Barbados
- Stalled: Bahrain, Columbia, Panama

COSMO members
ICON Support License
ICON in preparation
COSMO active users
COSMO inactive users

SEE-MHEWS countries (having access to ICON forecasts)



Country	2026		2027	
	GDPpc ¹	Fee	GDPpc ²	Fee
Botswana	7695.2	11,400	7.778.1	11,000
Brazil (INMET)	10280.3	13,700	10,713.3	16,900
Brazil (Navy)	10280.3	13,700	10,713.3	16,900
Cyprus	---	---	41,783.1	20,000
Oman	20248.4	20,000	19,947.4	20,000
Qatar	76275.9	20,000	72,524.9	20,000
Turkmenistan	8571.6	13,400	6,540.2	8,600
UAE	49377.6	20,000	50,273.5	20,000
Sum		112,200		133,400

2026

- Additional 13,700€ from Brazil (INMET) for 2025

2027

- We will adapt the fee to the GDPpc
- Cyprus contract recently signed

1) GDPpc taken from <https://data.worldbank.org/indicator/NY.GDP.PCAP.CD> (August 2025); BUT: fees have not been adapted.

2) GDPpc taken from <https://data.worldbank.org/indicator/NY.GDP.PCAP.CD> (July 2026);

Support Activities

- Special Training for 2 colleagues from Meteo Rwanda in May 2026
- Planned for ANACIM Senegal in Dec 2026.
 - ANACIM is interested in „ICON in the Cloud“: what about COSMO Money to develop and test?
- Replace Request Tracker by <https://exchange.icon-model.org>
- Changes in data transmission by DWD:
 - will introduce ccscs GRIB compression on 24 September.
 - have to check that all eccodes installations are installed with libaec.

The screenshot shows the 'exchange.icon-model.org' website. The header includes navigation icons and the URL. The main content area is titled 'Welcome back, drierger!' and features a search bar. Below the header, there are tabs for 'categories', 'tags', 'Latest', 'Hot', and 'Categories'. A 'New Topic' button is visible on the right. The main list displays various topics with their respective icons, titles, and metadata (Replies, Views, Activity). The left sidebar contains a 'Topics' menu with options like 'My posts', 'My messages', 'Review', 'Admin', 'More', 'CATEGORIES', 'Site Feedback', 'Staff', 'Open Discussion', 'All categories', 'TAGS', 'Search chat', 'CHANNELS', 'General', 'Staff', and 'DMS'.

Topic	Replies	Views	Activity
Confusing variable names in icon forcing data (iefff)	5	25	1d
Confusing "ECCODES ERROR" messages in ICON log files	1	26	13d
NWP atmosphere-ocean coupling on gpu	3	43	21d
Mkexp restart workflow stops after first restart interval (ICON 2025.10 public)	2	48	Jul 29
Full precision output of height levels	1	26	Jul 27
Minimize loss of information on vertical interpolation from ICON model levels	0	26	Jul 21
Adding ComIn registered variables to the METEOGRAM output?	7	108	Jul 20
MSL (Mean Sea Level) and height transformation	0	23	Jul 20
What kind of 'geographic' coordinates are used for ICON output?	1	41	Jul 20



COSMO NWP Testsuite



Actions in 2026

- ICON OSR icon-2026.04 released on 30 April 2026.
- Mail to the „NWP Testsuite Group“ on 28 May (no reaction at all) and reminder mail on 09 June.
- Scripts for the suite ported from ARPAE github account to DKRZ gitlab
- Discussions about namelist settings and what to test
- Encountered some technical problems
(e.g. running iconremap for IFS data with DWD GRIB2 shortnames: see next slide)
- Suite was finally started mid/end of July, finished on 08 August.
- Not clear, whether evaluation can be done during summer time; might not be ready until GM

- export `ECCODES_DEFINITION_PATH=<...>/definitions.edzw:<...>/definitions`
- To start ICON from IFS data, we need:
 - lowest model level geopotential available in GRIB2: recognized as **FI** (DWD GRIB2 shortName) even for **centre=ecmf**!
- surface level geopotential available in GRIB1: recognized as **z** (ECMWF GRIB1 shortname)
- In this way, iconremap can distinguish these two variables by the shortName.
- Later versions of eccodes (at least since 2.42.0) definitely use the ECMWF definitions for **centre=ecmf**
- But now model level and surface level have the same shortName **z** and cannot be distinguished any more!
- **Preliminary solution:** still use older eccodes version or reset centre to edzw!

```

.gitlab-ci.yml 5.19 KiB
56 include:
57   - local: .gitlab/ci/basic.yml
58   rules:
59     - if: $CI_PIPELINE_SOURCE == "web" && "${[ inputs.web-add-basic-stage ]}" == "true"
60     - if: $CI_PIPELINE_SOURCE != "web"
61   - local: .gitlab/ci/unit/icon.yml
62   rules:
63     - if: $CI_PROJECT_PATH == "icon/icon-model"
64     when: never
65     - if: $CI_PIPELINE_SOURCE == "web" && "${[ inputs.web-add-unit-stage ]}" == "true"
66     - if: $CI_PIPELINE_SOURCE == "merge_request_event"
67     changes:
68       - "*.in"
69       - .gitlab/ci/unit/icon.yml
70       - config/dkrz/*
71       - configure
72       - externals/**/*
73       - src/**/*
74       - support/**/*
75       - test/unit-tests/**/*
76       - utils/**/*
77   - local: .gitlab/ci/unit/ragnarok.yml
78   rules:

```



Gitlab-CI Overview



Structure

- Central configuration in `.gitlab-ci.yml`
- **Stages:** logical phases, e.g. *Build* → *Test* → *Deploy*
- **Jobs:** individual tasks within a stage
- **Runners:** execute jobs in defined environments
- **Artifacts & cache:** store results and accelerate workflows

```
66 - if: $CI_PIPELINE_SOURCE == "merge_request_event"
67   changes:
68     - "*.in"
69     - .gitlab-ci/unit/icon.yml
70     - cosmo/dune/*
```

How it works

- A commit or merge request triggers a pipeline
- GitLab runs jobs according to dependencies and rules
- Independent jobs can run in parallel
- Status, logs, and results are available in GitLab

```
81 - if: $CI_PIPELINE_SOURCE == "web" && "$[[ inputs.web-add-unit-stage ]]" == "true"
```

```
"$[[ inputs.web-add-basic-stage ]]" == "true"
```

Objectives

- Automate builds, tests, and deployments
- Detect errors early through continuous integration
- Ensure reproducible and transparent software delivery
- Enable faster development with consistent quality

Gitlab-CI - ICON Overview

icon-nwp / Pipelines / #17846

warnings

- 'dkrz:dolpung' is inactive
- 'dkrz:levante:cpu' is inactive
- 'dkrz:levante:gpu' is inactive

basic

- autoconf
- dochtml:publish:pages
- iconbot
- licenses
- style:coding
- style:general

unit

- icon:cpu
- icon:gcc
- icon:gpu
- icon:nag
- icon:nag:serial
- ragnarok:cpu:openmp
- ragnarok:cpu:serial
- ragnarok:gpu
- ragnarok:icon:bridge

functional

- nwp:muphys:gcc
- nwp:muphys:intel

system

- Accept new checksums
- aes:bubble:cpu:analyze
- aes:bubble:cpu:build
- aes:bubble:cpu:run
- aes:bubble:cpu:sp:analyze
- aes:bubble:cpu:sp:build
- aes:bubble:cpu:sp:run
- aes:bubble:gcc:analyze
- aes:bubble:gcc:build
- aes:bubble:gcc:run
- aes:bubble:gpu:analyze
- aes:bubble:gpu:build
- aes:bubble:gpu:run
- aes:bubble:gpu:sp:analyze
- aes:bubble:gpu:sp:build
- aes:bubble:gpu:sp:run
- aes:bubble:nag:analyze
- aes:bubble:nag:build
- aes:bubble:nag:run
- aes:bubble:nag:sp:analyze
- aes:bubble:nag:sp:build
- aes:bubble:nag:sp:run
- aes:bubble:ragnarok:cpu:analyze
- aes:bubble:ragnarok:cpu:build
- aes:bubble:ragnarok:cpu:run
- aes:bubble:ragnarok:gpu:analyze
- aes:bubble:ragnarok:gpu:build
- aes:bubble:ragnarok:gpu:run

external

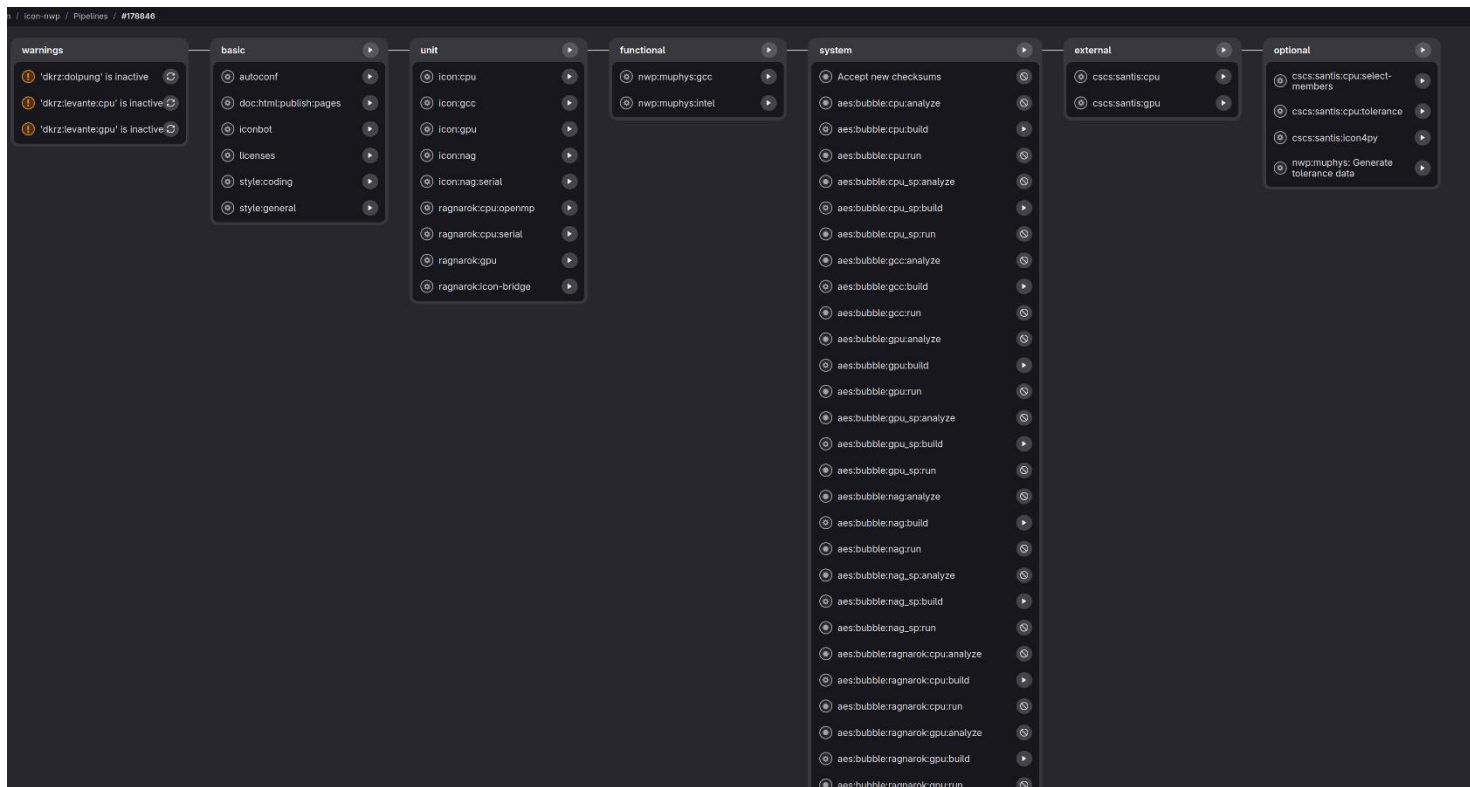
- cscs:santis:cpu
- cscs:santis:gpu

optional

- cscs:santis:cpu:select-members
- cscs:santis:cpu:tolerance
- cscs:santis:icon4py
- nwp:muphys:Generate tolerance data



Gitlab-CI - ICON Overview



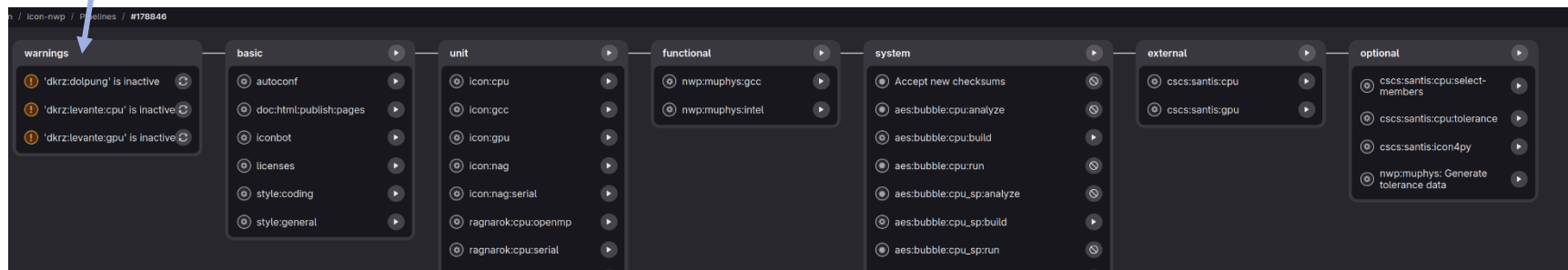
AI GENERATED

Structure: Stages

- **warnings:** Information on downtime of systems (*reduced CI functionality*)
- **basic:** Tests that ensure a proper workflow and code quality (*no execution of ICON code*)
- **unit:** Test small code units (functions, ...) independently
- **functional:** Test larger code units in standalone (*e.g. microphysics, cloud cover, ...*)
- **system:** Complete ICON tests *managed by the gitlab-ci*
- **external:** Complete ICON tests, *externally managed (mostly buildbot)*
- **optional:** Generate tolerance updates, ...

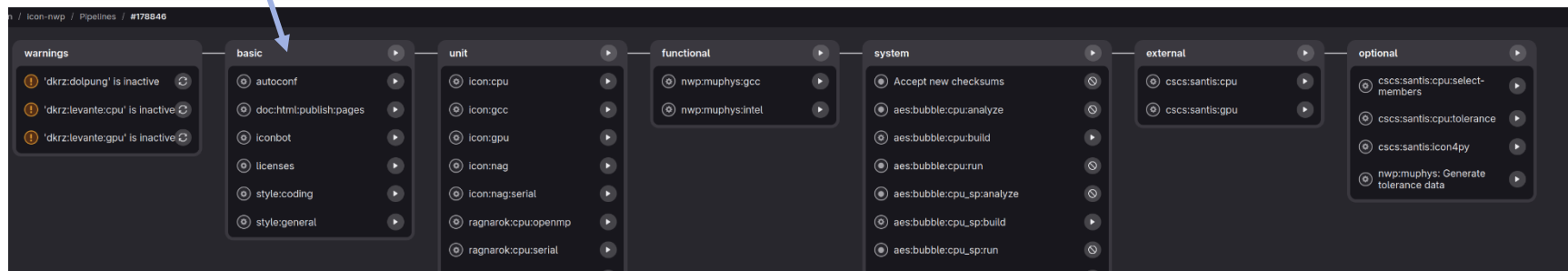
Structure: Stages

- **warnings:** Information on downtime of systems (*reduced CI functionality*)
- **basic:** Tests that ensure a proper workflow and code quality (*no execution of ICON code*)
- **unit:** Test small code units (functions, ...) independently
- **functional:** Test larger code units in standalone (*e.g. microphysics, cloud cover, ...*)
- **system:** Complete ICON tests *managed by the gitlab-ci*
- **external:** Complete ICON tests, *externally managed (mostly buildbot)*
- **optional:** Generate tolerance updates, ...



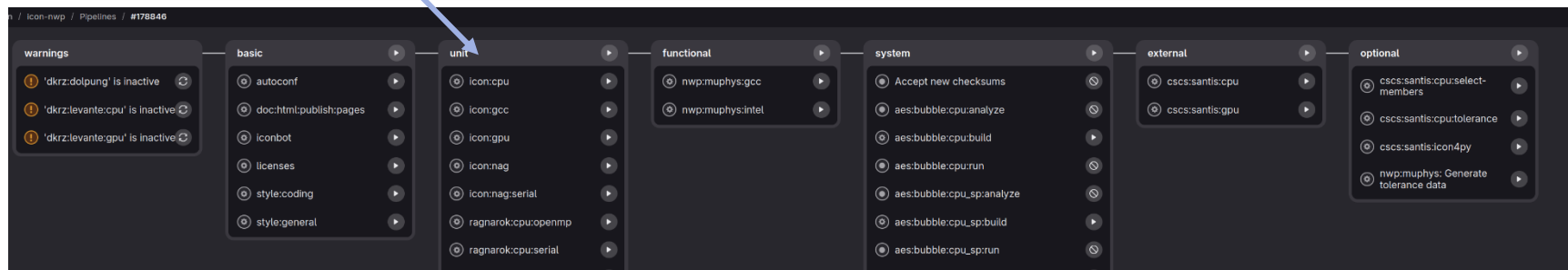
Structure: Stages

- **warnings:** Information on downtime of systems (*reduced CI functionality*)
- **basic:** Tests that ensure a proper workflow and code quality (*no execution of ICON code*)
- **unit:** Test small code units (functions, ...) independently
- **functional:** Test larger code units in standalone (*e.g. microphysics, cloud cover, ...*)
- **system:** Complete ICON tests *managed by the gitlab-ci*
- **external:** Complete ICON tests, *externally managed (mostly buildbot)*
- **optional:** Generate tolerance updates, ...



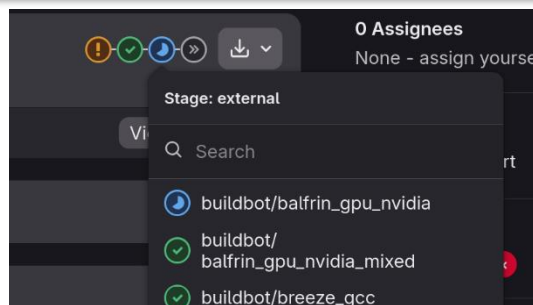
Structure: Stages

- **warnings:** Information on downtime of systems (*reduced CI functionality*)
- **basic:** Tests that ensure a proper workflow and code quality (*no execution of ICON code*)
- **unit:** Test small code units (functions, ...) independently
- **functional:** Test larger code units in standalone (*e.g. microphysics, cloud cover, ...*)
- **system:** Complete ICON tests *managed by the gitlab-ci*
- **external:** Complete ICON tests, *externally managed (mostly buildbot)*
- **optional:** Generate tolerance updates, ...

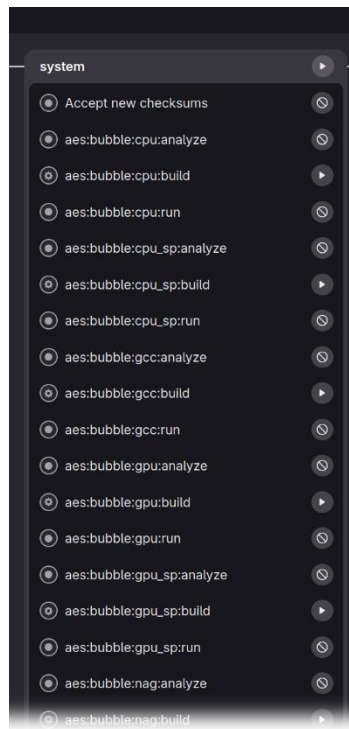


Structure: Stages

- **warnings:** Information on downtime of systems (*reduced CI functionality*)
- **basic:** Tests that ensure a proper workflow and code quality (*no execution of ICON code*)
- **unit:** Test small code units (functions, ...) independently
- **functional:** Test larger code units in standalone (*e.g. microphysics, cloud cover, ...*)
- **system:** Complete ICON tests *managed by the gitlab-ci*
- **external:** Complete ICON tests, *externally managed (mostly buildbot)*
- **optional:** Generate tolerance updates, ...



Job	Objectives
licenses	Check license header of all files. Exceptions: <code>REUSE.toml</code>
release-notes	Check if release notes were added. Exception: <code>[no_release_notes]</code>
cla-check	Check if authors signed an agreement to ICON's license conditions
style:coding	<code>pre-commit</code> for coding style. Generates a patch (<code>git apply</code>)
style:general	Multiple scripts to check for encapsulation, access, encoding, ...
autoconf	Generate <code>configure</code> from human-readable <code>configure.ac</code> (<i>fix software stack</i>) Generates a patch (<code>git apply</code>)
iconbot	Notifier in Merge Requests (<i>CLA links, ping subscribed users</i>)
doc:html:check	Generates HTML documentation from Markdown (<i>View app button</i>)



Systems tests (*currently MPI-M/AES only, but planned for NWP!*)

- Direct integration in gitlab-ci using jacamar-runner
runs as user on HPC and offers slurm & PBS support
- Very long list due to job chains
build → run → analyze → Accept new checksums
- For ICON, this is rather new and will likely replace buildbot step-by-step
Only a technical difference! Tests can be identical

Questions?



Daniel Rieger
Numerical Models Division
daniel.rieger@dwd.de

