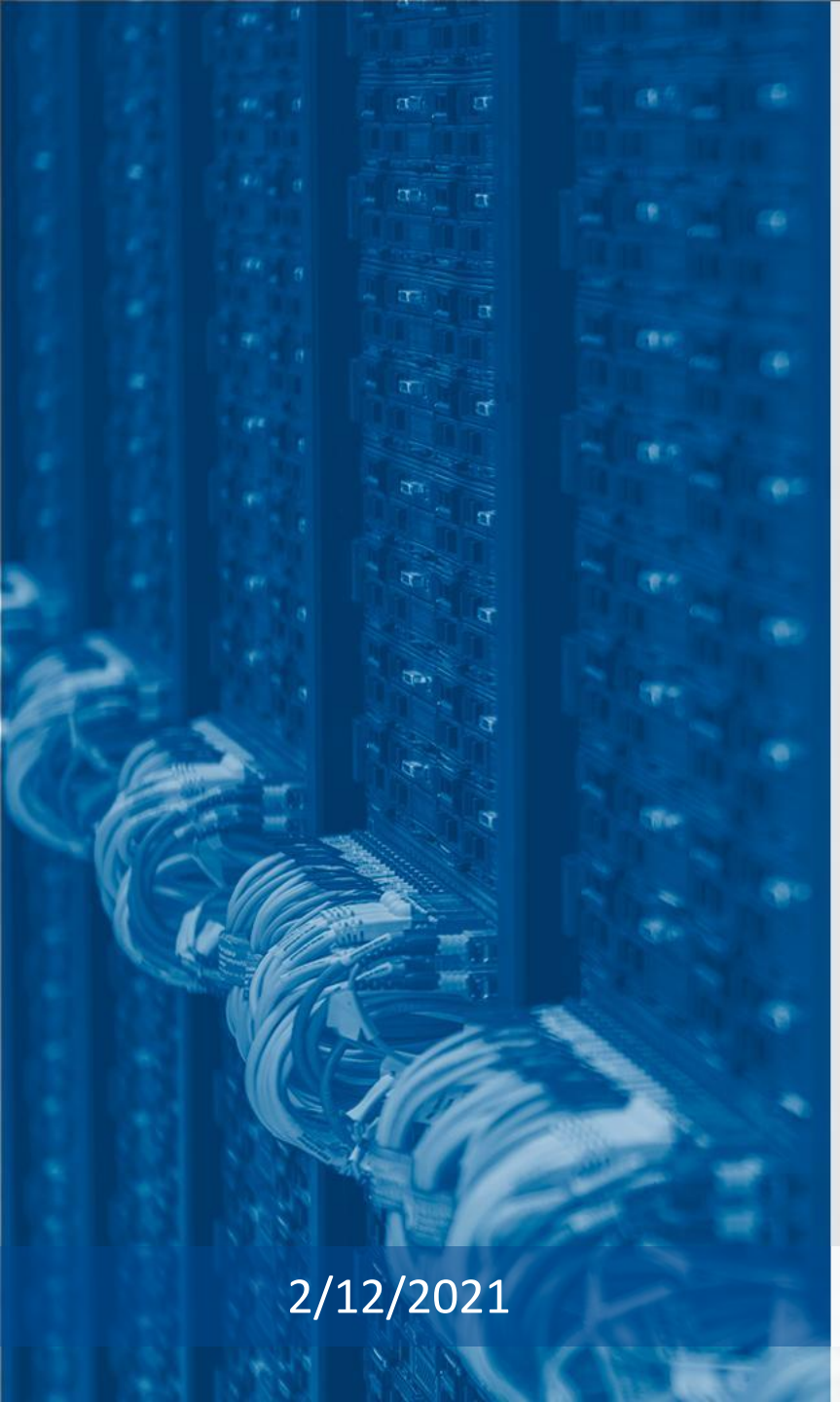


Barcelona Supercomputing Center
Centro Nacional de Supercomputación

AWICM3-XIOS workshop

Xavier Yepes-Arbós

esiwace
CENTRE OF EXCELLENCE IN SIMULATION OF WEATHER AND CLIMATE IN EUROPE

2/12/2021

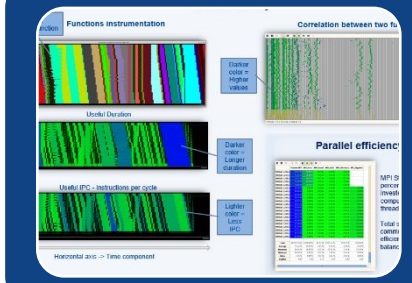
AWI, Bremerhaven, Germany

Index

1. Who we are

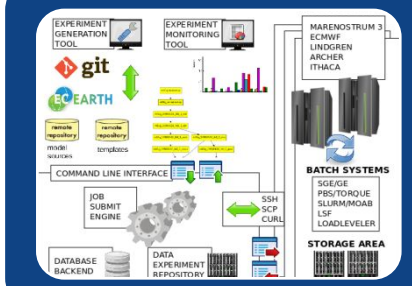
Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Computational Earth Sciences Group



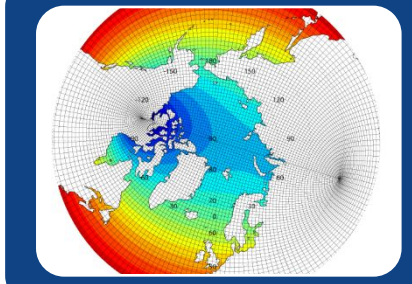
Performance Team

- Provide HPC Services (profiling, code audit, …)
- Apply new computational methods



Models and Workflows Team

- Development of HPC user-friendly software framework
- Support the development of atmospheric research software

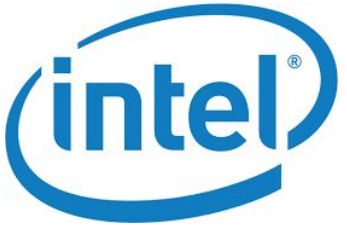


Data and Diagnostics Team

- Big Data in Earth Sciences
- Provision of data services
- Visualization

Performance Team

- The necessary refactoring of numerical codes is given a lot of attention and is stirring a number of discussions.
  - Computational performance analysis and new optimizations are needed for actual numerical models.
  - Studying new algorithms for the new generation of high performance platforms (path to exascale).
- We are collaborating with several institutions on different projects working together in the same direction.

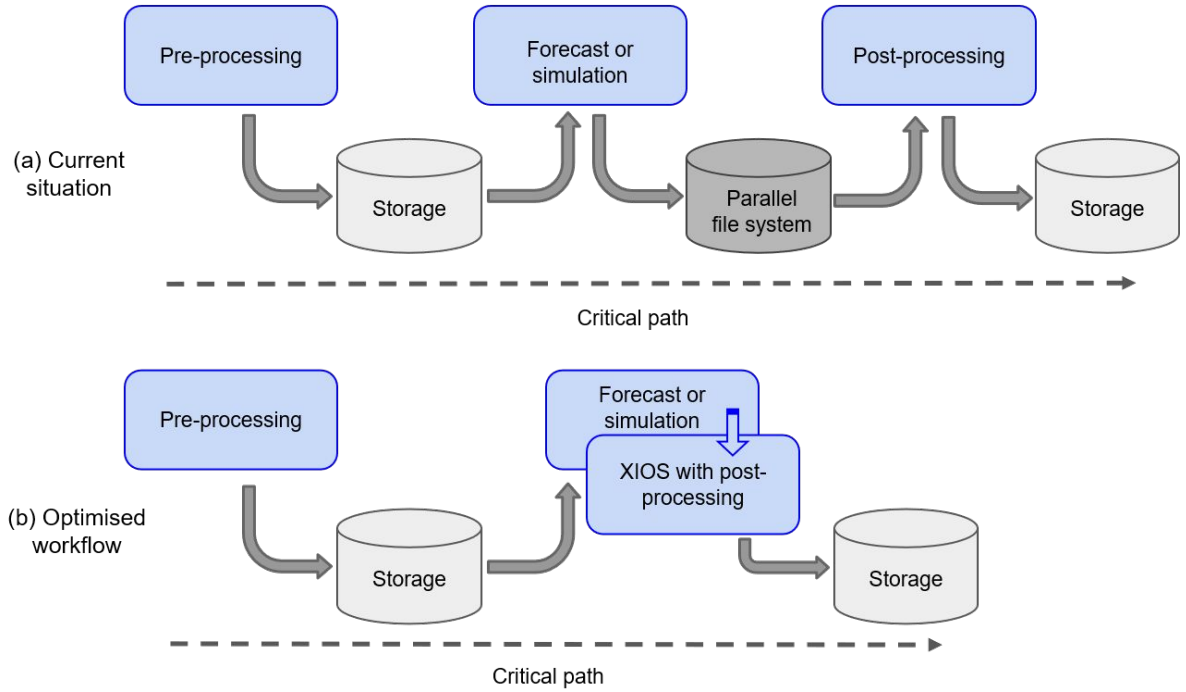


2. Context

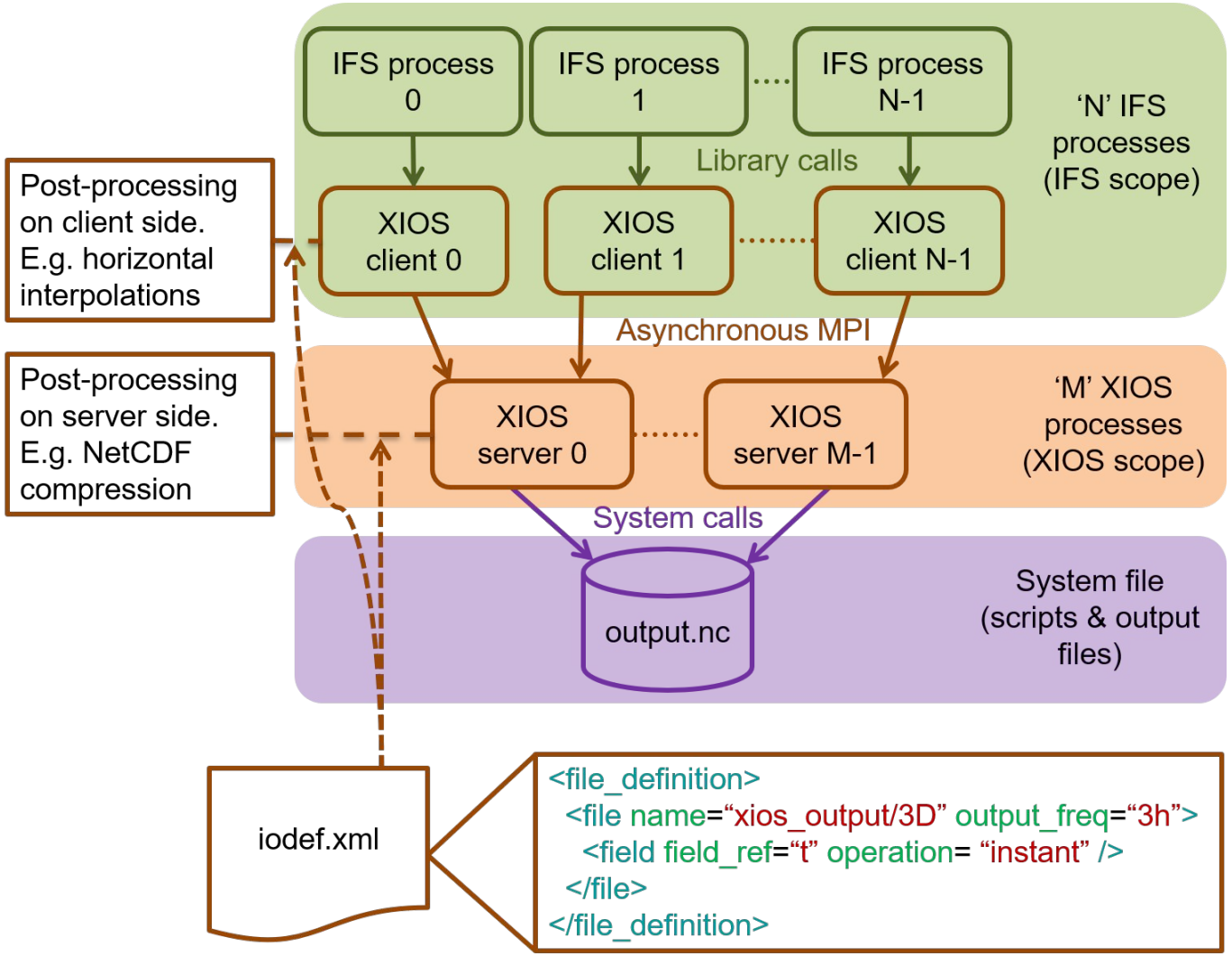

BSC
Barcelona
Supercomputing
Center
Centro Nacional de Supercomputación

Objective: Integrate XIOS

- The I/O issue is typically addressed by adopting scalable parallel I/O solutions such as XIOS.
- The XML Input/Output Server (XIOS) is an asynchronous MPI parallel I/O server developed by the Institute Pierre Simon Laplace (IPSL).
- XIOS is a widely I/O tool used in the climate community because of these **features**:
  - Output files are in **netCDF** format.
  - Written data is **CMIP-compliant** (CMORized).
  - It is able to post-process data inline to generate **diagnostics**.
- XIOS is thought to address:
  - The **inefficient** legacy read/write process.
  - The unmanageable size of “raw” data by implementing **inline post-processing**.



OpenIFS-XIOS integration scheme

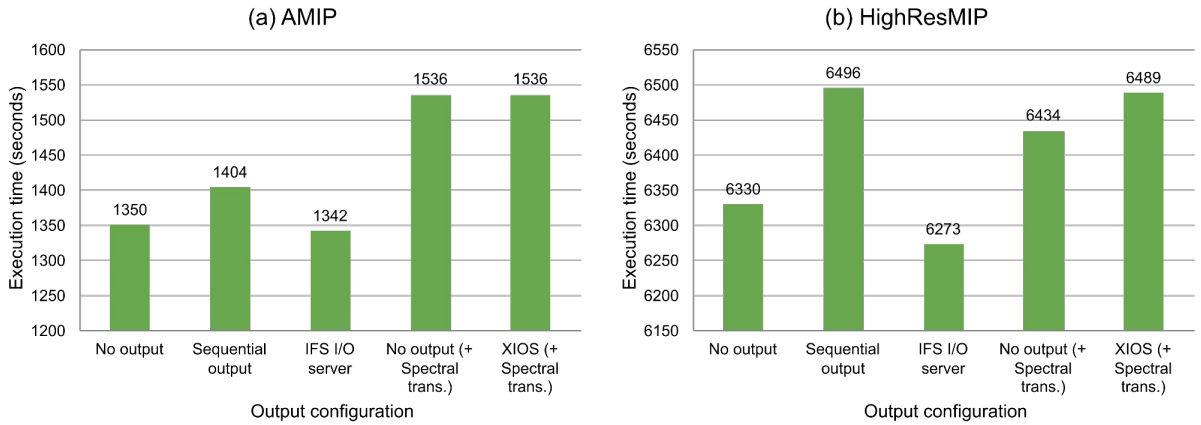


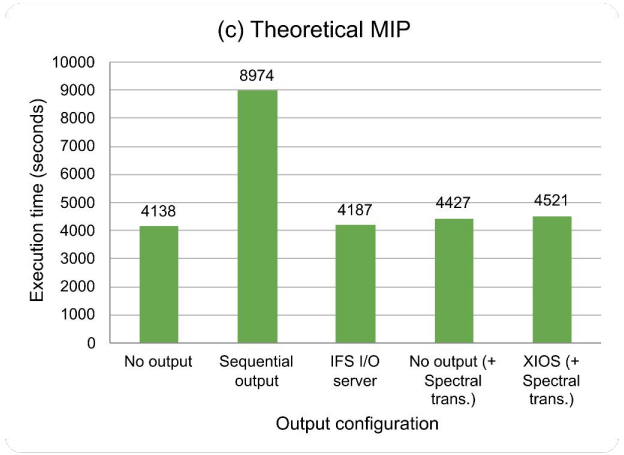
OpenIFS-XIOS integration summary

- Scientific highlights:
  - Both **grid-point** and **spectral fields** (transformed to grid-point space using TRANS) are supported.
  - All **surface** and **3D** fields can be output.
  - Different **vertical levels** are available: model, pressure, theta and PV levels.
  - **No** longer needed to set up the **FullPos namelist** (NAMFPC).
  - FullPos **spectral fitting** is available.
  - Physical tendencies and fluxes output (**PEXTRA fields**) are also supported.
- Computational performance highlights:
  - In-depth benchmarking: **small** data output **overhead** with enough computational resources.

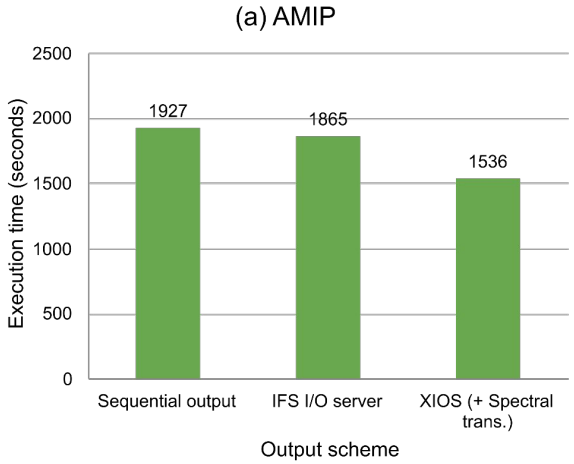
Computational performance of the integration

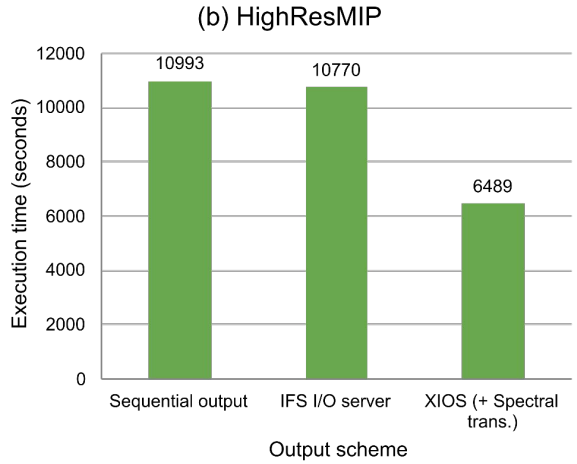
Output schemes comparison

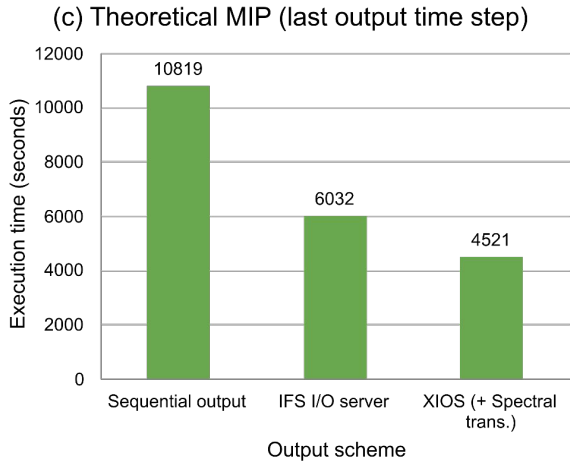




Output schemes comparison including post-processing

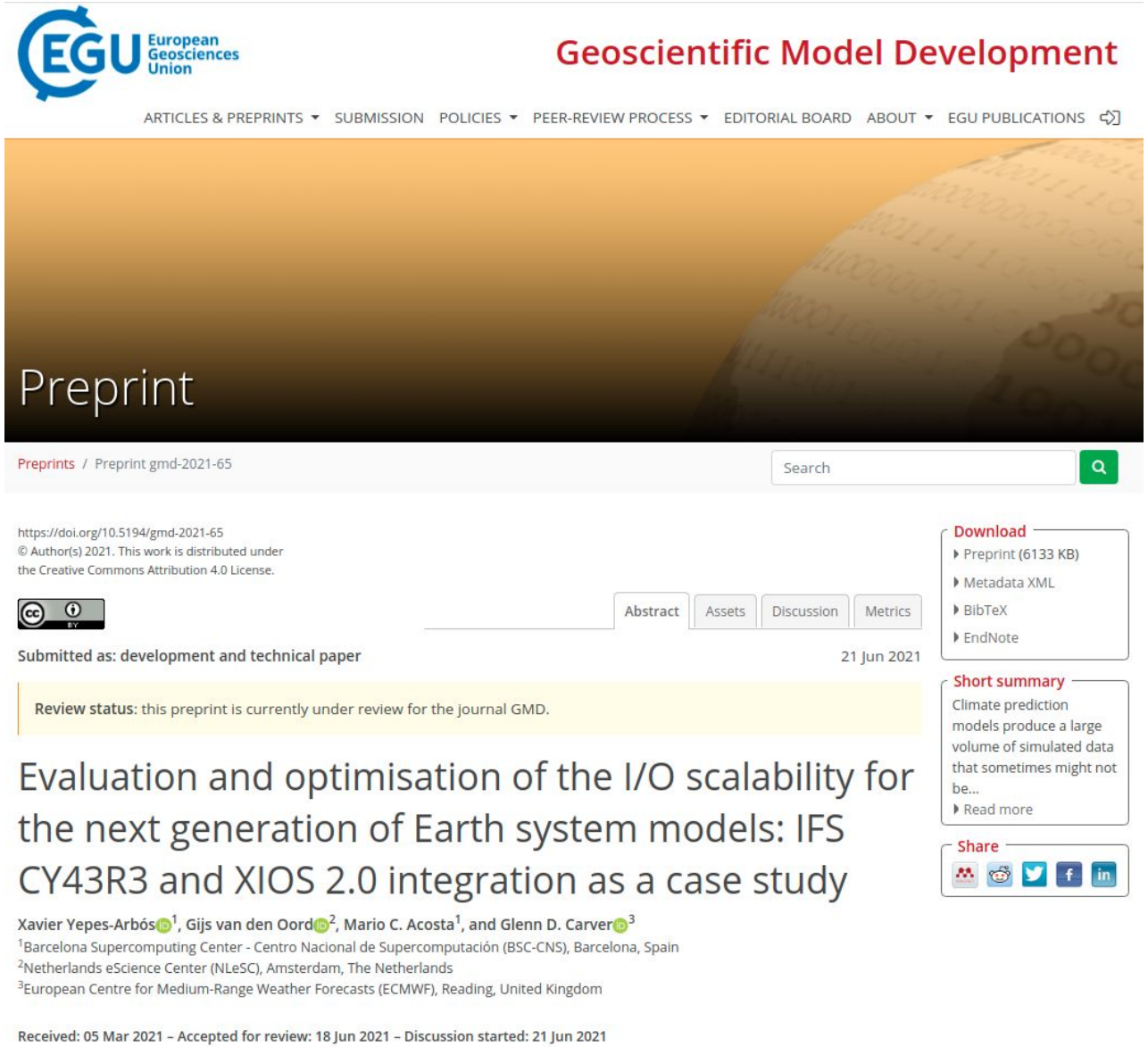






Barcelona Supercomputing Center
Centro Nacional de Supercomputación

esiwace
CENTRE OF EXCELLENCE IN SIMULATION OF WEATHER AND CLIMATE IN EUROPE



Geoscientific Model Development

ARTICLES & PREPRINTS SUBMISSION POLICIES PEER-REVIEW PROCESS EDITORIAL BOARD ABOUT EGU PUBLICATIONS

Preprint

Preprints / Preprint gmd-2021-65

Search

https://doi.org/10.5194/gmd-2021-65

Abstract Assets Discussion Metrics

Submitted as: development and technical paper 21 Jun 2021

Review status: this preprint is currently under review for the journal GMD.

Evaluation and optimisation of the I/O scalability for the next generation of Earth system models: IFS CY43R3 and XIOS 2.0 integration as a case study

Xavier Yepes-Arbós[1], Gijs van den Oord[2], Mario C. Acosta[1], and Glenn D. Carver[3]
[1]Barcelona Supercomputing Center - Centro Nacional de Supercomputación (BSC-CNS), Barcelona, Spain
[2]Netherlands eScience Center (NLeSC), Amsterdam, The Netherlands
[3]European Centre for Medium-Range Weather Forecasts (ECMWF), Reading, United Kingdom

Received: 05 Mar 2021 – Accepted for review: 18 Jun 2021 – Discussion started: 21 Jun 2021

Download
- Preprint (6133 KB)
- Metadata XML
- BibTeX
- EndNote

Short summary
Climate prediction models produce a large volume of simulated data that sometimes might not be...
Read more

Share

https://doi.org/10.5194/gmd-2021-65

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

esiwace
CENTRE OF EXCELLENCE IN SIMULATION OF WEATHER AND CLIMATE IN EUROPE

3. OpenIFS-XIOS usage

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Introduction

- The XIOS basic usage explanation will be done interactively using the terminal.
- For more advanced XIOS features and functionalities check this tutorial.
- You can also check the official XIOS webpage.
- This section also contains slides that review XIOS aspects that are particular to OpenIFS.
- A complete OpenIFS-XIOS user guide is available in this ECMWF webpage.

XIOS XML files for OpenIFS

- **iodef.xml**: it contains basic XIOS parameters.
- **context_oifs.xml**: it contains FullPos parameters to tune the vertical interpolations.
- **axis_def_oifs.xml**: it defines the different types of vertical levels available.
- **domain_def_oifs.xml**: it defines the different domains (or types of grids) available.
- **grid_def_oifs.xml**: it defines the grids (XIOS terminology) available used to map fields.
- **field_def_oifs.xml**: it defines all the available fields to be output.
- **file_def_oifs.xml**: it defines netCDF files to be written.

FullPos parameters

- It is possible to control some **FullPos** variables.
- FullPos spectral fitting:
  - NFITP
  - NFITT
  - NFITV
- Other types of variables:
  - NFPCLI
  - LFPQ
  - LTRACEFP
  - RFPCORR

```
<variable_group id="spectral_fitting">
  <variable id="nfitp" name="NFITP" type="int"> 2 </variable>
  <variable id="nfitt" name="NFITT" type="int"> 2 </variable>
  <variable id="nfitv" name="NFITV" type="int"> 2 </variable>
</variable_group>

<variable_group id="fullpos_other">
  <variable id="nfpcli"   name="NFPCLI"   type="int"   > 0       </variable>
  <variable id="lfpq"     name="LFPQ"     type="bool"  > false   </variable>
  <variable id="ltracefp" name="LTRACEFP" type="bool"  > false   </variable>
  <variable id="rfpcorr"  name="RFPCORR"  type="double"> 60000.0 </variable>
</variable_group>
```

Define different regular Gaussian domains

- OpenIFS has **two default grids**:
  - Native reduced Gaussian grid.
  - Regular lat-lon grid with 256 latitudes and 512 longitudes.
- It is possible to change the sizes of the regular grid or even to declare more than one regular grid. There are two key **attributes**:
  - 'ni_glo': the number of longitudes.
  - 'nj_glo': the number of latitudes.
- It is necessary to specify ‘generate_rectilinear_domain’ and ‘interpolate domain’ to indicate that a horizontal interpolation (remapping) is needed.

```
<domain_group id="regular_domains" type="rectilinear" >
  <domain id="regular" long_name="regular grid" ni_glo="512" nj_glo="256" >
    <generate_rectilinear_domain />
    <interpolate_domain write_weight="true" />
  </domain>
</domain_group>
```

Enable regular lat-lon grid output

- To output a field in a regular lat-lon grid it is necessary to use the attribute 'grid_ref'.
- For example, to output the 3D temperature field in a 256x512 lat-lon grid and model levels, you can use this XML code:

```
<field field_ref="t"    name="t"    grid_ref="regular_ml" freq_op="6h" operation="instant" />
```

where 'regular_ml' is defined in grid_def_ifs.xml as follows:

```
<grid id="regular_ml" description="3D interpolated regular grid with hybrid model levels" >
  <domain domain_ref="regular" />
  <axis axis_ref="model_levels" />
</grid>
```

Understanding output frequency, sampling frequency, NFRHIS and NFRPOS

- It is necessary to **distinguish** between the scope of attributes
  - 'output_freq' and 'freq_op' -> XIOS
  - 'NFRHIS' and 'NFRPOS' -> FullPos.
- **'output_freq':** it controls the **writing frequency** of a netCDF file.
- **'freq_op':** it controls the **sampling frequency**, this is, the frequency of sending data from OpenIFS to XIOS.
- **'NFRHIS'** & **'NFRPOS':**
  - They control the **post-processing frequency** of FullPos and the TRANS package to transform spectral fields to grid-point fields.
  - Must be set up taking the **greatest common divisor** (gcd) of all 'freq_op' values defined.
  - Note that **always** 'NFRHIS' = 'NFRPOS'.

Understanding output frequency, sampling frequency, NFRHIS and NFRPOS

Example: you want to output these three variables:

- 't': output every 6 hours the average of hourly data. You should set up 'output_freq=6h' and 'freq_op=1h'.
- 'u': output every 12 hours the maximum of 3 hourly data. You should set up 'output_freq=12h' and 'freq_op=3h'.
- 'cc': output every 6 hours instant data. You should set up 'output_freq=6h' and 'freq_op=6h'.
- 'NFRHIS' and 'NFRPOS' must be set up taking the gcd of 'freq_op' values across all fields ('t', 'u' and 'cc'), which is 1h. If the time step duration is 900 seconds for instance, then 'NFRHIS' and 'NFRPOS' should be set up to 4 (1h).

- It is also possible to specify these two FullPos variables in hours by using negative values: 'NFRHIS' = 'NFRPOS' = '-1'

Not correctly setting freq_op, NFRHIS and NFRPOS

It is very important to correctly set up 'NFRHIS' and 'NFRPOS' to produce correct data and do not waste computational resources:

- If 'NFRHIS' and 'NFRPOS' use a **smaller value** than the gcd, FullPos would be called **unnecessarily** (post-processed data will not be sent to XIOS as it is not required). This increases the computational cost with no gains in the accuracy of the results.
- If 'NFRHIS' and 'NFRPOS' use a **bigger value** than the gcd, FullPos would **not be called enough times** (post-processed data will not be correct).

Optimizations for sending data from OpenIFS processes to XIOS servers

- There are two available optimizations that might be useful to improve the execution time under some circumstances. It is not possible to predict in which conditions (many factors), so it is necessary to test them.
- They are disabled by default, but can be enabled in context_ifs.xml:
  - **'LOPT_SEND':** it enables a mechanism to change where data is sent from XIOS clients (OpenIFS processes) to XIOS servers to truly **overlap** OpenIFS computations with XIOS communications.
  - **'LSINGLE_PREC_SEND':** it sends data from XIOS clients to XIOS servers in **single precision** (32 bits) instead of double precision (64 bits). This allows you to send half of the data to decongest the network.

```
<variable_group id="client_server_comm">
  <variable id="lopt_send"         name="LOPT_SEND"         type="bool"> false </variable>
  <variable id="lsingle_prec_send" name="LSINGLE_PREC_SEND" type="bool"> false </variable>
</variable_group>
```

4. Optimal XIOS setup

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

What factors affect XIOS performance?

There are several **factors** that can be tuned to directly **improve** the XIOS performance:

- Number of servers
- Number of dedicated nodes for servers
- ‘one_file’ vs. ‘multiple_file’ mode
- ‘optimal_buffer_size’
- ‘buffer_size_factor’
- 2-level server mode in combination with “time series”.
- Lustre striping

What factors affect XIOS performance?

There are other **factors** that can implicitly **impact** on the XIOS performance:

- Optimization compilation flags
- MPI placing
- Output size
- Output frequency
- Arithmetic and temporal operations such as averages
- Spatial operations such as remapping

XIOS resources

- Theses two factors are critical to make XIOS scalable:
  - Number of XIOS servers.
  - Number of dedicated nodes for XIOS servers.
- Having more XIOS nodes increases the bandwidth between model processes and servers, which is necessary to perform an asynchronous and fast transfer.
- Having more XIOS servers increases the computational power on server side (beneficial depending on the post-processing operation such as netCDF compression), but:
  - Makes the ‘one_file’ mode slower.
  - Data is spread across more netCDF files if ‘multiple_file’ mode is used (see 2-level server mode).

```
xios:
    xml_dir: "${general.esm_namelist_dir}/oifs/43r3/xios/"
    with_model: oifs
    nproc: 1
    omp_num_threads: 48
```

‘one_file’ vs. ‘multiple_file’ mode

- **'one_file'** mode has a limited computational efficiency as it **does not scale well** when outputting a large volume of data for **high resolution** configurations.
- **'multiple_file'** mode achieves a good computational efficiency as it **scales** with many resources. However, each XIOS server writes its own netCDF file, so **output data is splitted** between all these partial files.
  - It is necessary to study if there is any existing tool capable of efficiently combining these partial netCDF files into a single one.
  - If it does not exist, it would be necessary to develop such a tool, like in NEMO.
  - Alternatively, we will see how to deal with this issue using the 2-level server mode and “time series”.

```
<file_group
    type="multiple_file"
    format="netcdf4"
    par_access="collective"
    name="awi3_atm"
    split_freq="1y">
```

Buffer size settings

There are two parameters to control the buffer size to send data between clients and servers

- ‘optimal_buffer_size’: it controls whether using asynchronous or synchronous communications:
  - ‘performance’: it uses as much memory as it is needed to bufferize all data between two output periods, so it is the fastest option.
  - ‘memory’: it uses the minimum amount of memory needed, so no performance at all.
- ‘buffer_size_factor’: XIOS automatically computes the size of the buffers. However, users can adjust it using a multiplying factor.

```
<variable_group id="buffer" > <!-- Tune both "buffer" variables for performance purposes -->
  <variable id="optimal_buffer_size" type="string"> performance </variable>
  <variable id="buffer_size_factor"  type="double"> 1.0         </variable>
</variable_group>
```

2-level server mode

XIOS 2.5 offers a 2-level server mode where the pools of XIOS servers is distributed between 2 levels:

- Level 1: They are in charge of receiving the data from OpenIFS processes and redistributing it to subsets of level-two servers (called pools).
- Level 2: They are in charge of writing netCDF files that contain the entire domain into the storage system.

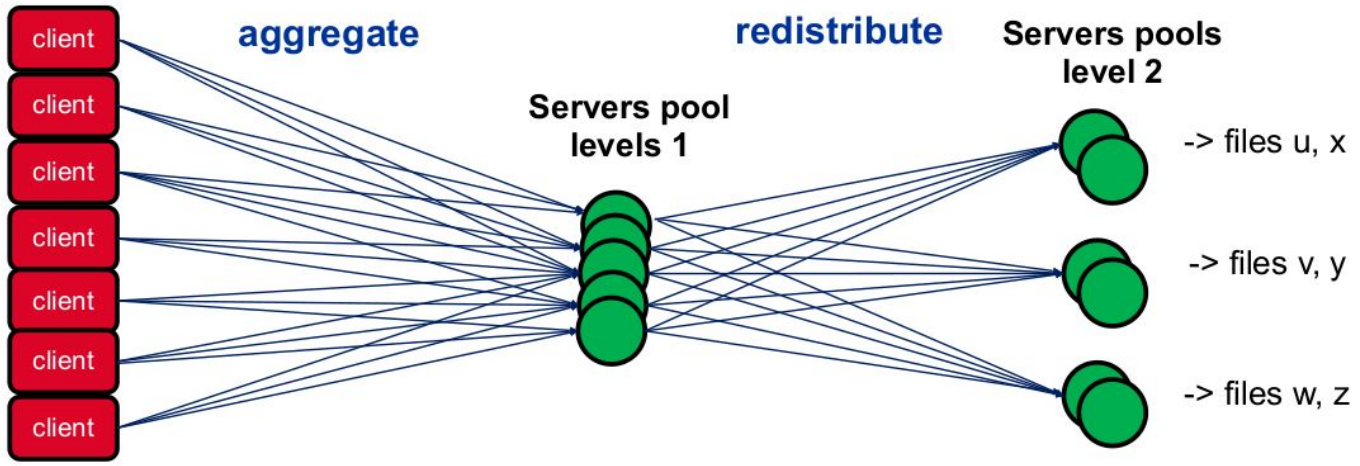


Figure source: XIOS team

2-level server mode

- ‘using_server2’ -> it enables the 2-level server mode.
- ‘ratio_server2’ -> it specifies the percentage of XIOS server to be used in the second level.
- ‘number_pools_server2’ -> the number of subsets of servers within the second level. By default the number of pools is equal to the number of servers.

```
<variable_group id="server" > <!-- Tune 2-level servers for performance purposes -->
  <variable id="using_server2"         type="bool"> true </variable>
  <variable id="ratio_server2"         type="int" > 75   </variable>
  <variable id="number_pools_server2" type="int" > 3    </variable>
  <variable id="server2_dist_file_memory"       type="bool"  > false </variable>
  <variable id="server2_dist_file_memory_ratio" type="double"> 0.5   </variable>
```

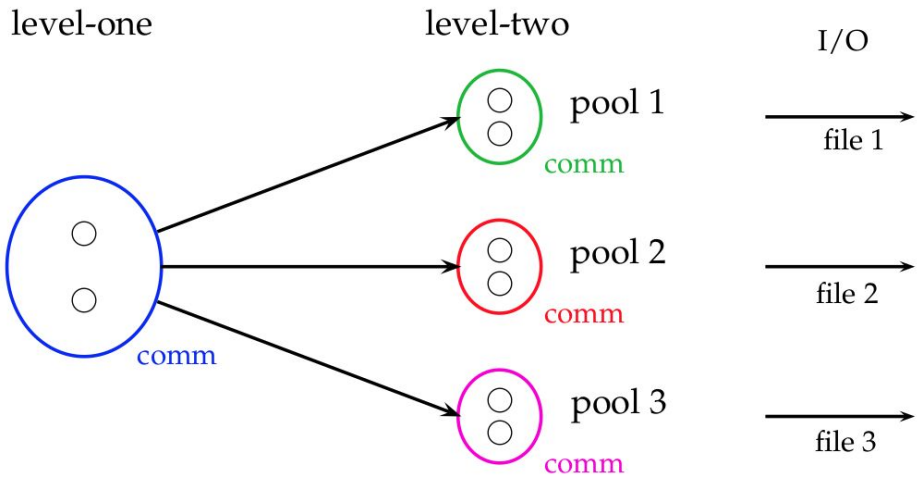


Figure source: XIOS team

2-level server mode

When enabling ‘timeseries’ with 2-level server mode and one second level server per pool:

- Each field is written into a different netCDF file.
- Each file contains the entire domain of a field.
- Files are well-balanced across all second level servers.

```
<file
    enabled="false"
    timeseries="only"
    output_freq="6h"
    name_suffix="_6h"
    description="ECE4/OIFS 6hourly surface fields">
```

Lustre filesystem

- The **Lustre** filesystem stores a file in one or more Object Storage Target (OST) devices.
- If OpenIFS is run on a cluster that uses Lustre it is important to pay attention to the **striping**, which allows to divide a file into chunks that are stored in different OSTs.
  - When using the 'one_file' mode, it is important to set up a striping for each netCDF at least as equal as to the number of XIOS servers.
  - This allows each XIOS server to write into a different OST, which prevents to affect the performance of the whole system.

Spatial operations

- This kind of operations are **very expensive** depending on different parameters (number of fields, size of the fields, frequency, etc), so it can have a large impact on the total execution time.
- Why? Because remapping is performed on the **client side** of XIOS, so OpenIFS processes have to first interpolate before resuming the time stepping.
- Recommendation: use remapping when it is strictly necessary and not systematically.

Other performance considerations

- **Optimization compilation flags**: It is very important to pay attention in the optimization flags as they might considerably affect the computational efficiency.
- **Output size and frequency**: These two factors directly affect the performance of XIOS. When larger volume of output and at a higher frequency, more overhead.
- **Arithmetic and temporal operations**: These operations might have a cost increase that is part of the critical path of OpenIFS.
- **MPI placing**: changing the default placing of XIOS servers might help to improve the computational efficiency.

XIOS performance reports

- XIOS can generate **performance reports** for each client and server at the end of the execution.
- The client ones are really important to know if OpenIFS processes are **blocked waiting** for the send buffer to be freed.
- The waiting ratio should be **close to zero**.

```
-> report :  Performance report : Whole time from XIOS init and finalize: 150.221 s
-> report :  Performance report : total time spent for XIOS : 25.3604 s
-> report :  Performance report : time spent for waiting free buffer : 0.344329 s
-> report :  Performance report : Ratio : 0.229215 %
-> report :  Performance report : This ratio must be close to zero. Otherwise it may be usefull to increase buffer size or numbers of server
-> report :  Memory report : Minimum buffer size required : 80476 bytes
-> report :  Memory report : increasing it by a factor will increase performance, depending of the volume of data wrote in file at each time step of the file
```

- The server ones are also important. The ratio shouldn't be more than 60%.

```
-> report : Performance report : Time spent for XIOS : 143.277
-> report : Performance report : Time spent in processing events : 10.2116
-> report : Performance report : Ratio : 7.12718%
```

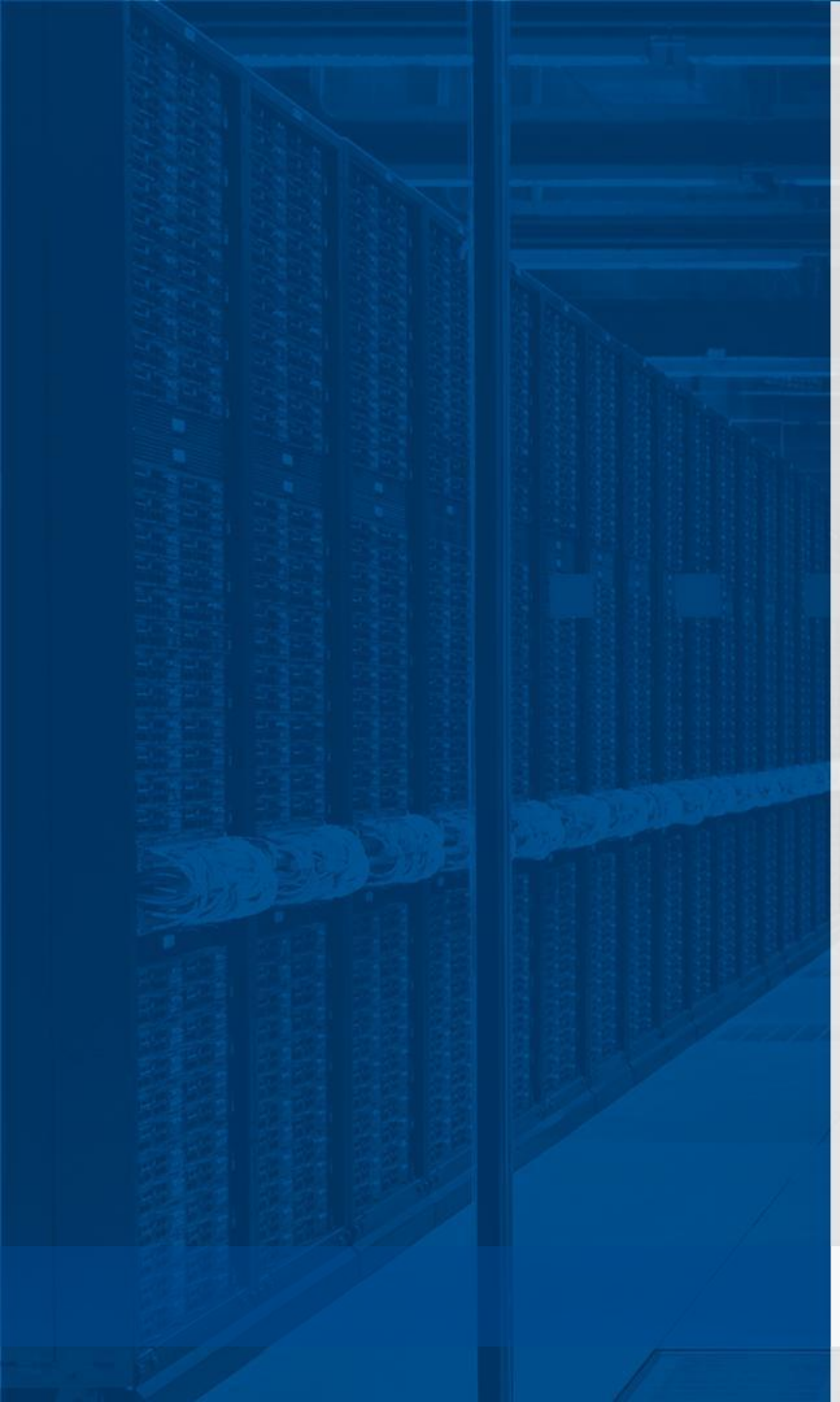
5. AWICM3-XIOS results

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Sequential I/O vs. XIOS on Juwels

- Tco95L91 (100 km) - CORE2
  - Sequential I/O -> **124 SYPD**
  - XIOS -> **134 SYPD**
- Tco159L91 (61 km) - CORE2
  - Sequential I/O -> **60 SYPD**
  - XIOS -> **64 SYPD**
- Tco319L137 (31 km) - DART
  - Both output schemes achieve the same performance as FESOM is the limiting component.

NOTE: XIOS outputs data already averaged, remapped and in netCDF format.



Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Thank you

esiwace
CENTRE OF EXCELLENCE IN SIMULATION OF WEATHER AND CLIMATE IN EUROPE

xavier.yepes@bsc.es