Combined document for the changes made in the manuscript

This document is color-coded as follows:

- Comments by reviewers are in **blue**.
- Our responses are in **black**.
- Blocks of text that is added to the are in **red**
- **The track-changes that show the changes between initial version (v2) of the manuscript and the last version (v5) can be also seen at the end of this document.**

<u>Changes made in the manuscript version 4</u>

Response to Topical Editor

1) The text about the mapped and unmapped grid points is much clearer now. I just suggest to change p.9 l. 35 "... to find mapped and unmapped grid points" for "... to identify unmapped grid points for the bilinear interpolation". And I think you can remove the "After finding mapped and unmapped grid points" at the beginning of the next sentence but add a "then" after "can" in that sentence ("... the field can then be interpolated from the source ..."). Also I think that Figure 4 and all the details it contains is not needed (but I'll let you decide on that).

The text is modified as suggested. I decided to keep Figure 4 for the clarification of the method used in the design of the coupled modeling system.

2) Figures 9 and 10, their captions and the related text are fine but I think that the speed-up is still not precisely defined.

The following text and also equation is added to the manuscript just after the first paragraph in Section 4.3 (Performance benchmark).

In the benchmark results, the slightly modified version of the speed-up is used because the best possible sequential implementation of the utilized numerical model (standalone and coupled) does not exist for the used demonstration application and model configurations. In this case, the speed-up is defined as the ratio of the parallel execution time for the minimum number of processors required to run the simulation ($T_p(N_{min})$; based on 140 cores in this study) to the parallel execution time ($T_p(N)$; see Eq.1).

$$S(N) = \frac{T_p(N_{min})}{T_p(N)}$$

3) P.15, l.11, please change "but it is not evident in HR case and scales very well ..." for "but it is not evident in HR case that scales very well ..."

The text is modified as suggested.

Changes made in the manuscript version 3

Response to Topical Editor

Title, Abstract

1) Your argument about keeping "Toward" in the title is fine. However, you don't answer the reviewer's remark about the use of the word "integrated". I agree with the reviewer that use of "integrated" is not appropriate, as it seems contrary to the modularity of the approach. So, what about "Toward modular in-situ visualization in Earth System Models: the regional modeling system RegESM 1.1".

The title is changed as suggested.

2) I still propose many language modifications in the attached ManuscriptAnnotated.pdf

All of them is fixed in the latest version of the manuscript.

43) To answer the referee's remark, you now mention the "implicit" coupling on l.30 but before defining it; the definition "The main difference between the implicit and semi-implicit coupling type is that the models interact on different time scales in implicit coupling scenarios. » comes only after but is not clear at all. Contrary to what was asked by the referee, I suggest to remove the text about the implicit coupling as it is not an option in RegESM (if I understand well).

The text related with implicit coupling is removed.

46) It is OK for me not to describe how ESMF handles this. But your description on p.9, l.19-32 is very hard to follow. In particular, I don't understand at all what "According to the algorithm, the mapped grid points have same land-sea mask type in both model components (i.e., both are sea or land). On the other hand, the land-sea mask type does not match completely in the case of unmapped grid points" means. Can you try to review and simplify it, maybe giving the essence of the method and not all the technical steps? Also on Fig. 4, a "on" is missing between "only" and "grid" in "All interpolations are performed only grid points over SEA".

The text is simplified (please see track changes) and Figure 4 is also fixed.

47) I am sorry to say that I think that the added paragraph does not help understanding the algorithm.

Please refer to the previous item. The unnecessary implementation details are removed from the paragraph. The algorithm is mainly very simple and perform two different interpolation (bilinear and nearest-neighbor) to the exchange field and compare the results to find mapped (land-sea mask is same in both source and destination grid) and unmapped (mismatch in land-sea mask) grid points. Then, using this information, it performs interpolation and extrapolation to transfer exchange field to the destination grid.

49) ... A visualization pipeline integrates a data flow network in which computation is described as a collection of executable modules that are connected in a directed graph representing how data moves between modules (Moreland, 2013). There are three types of modules: sources (file readers and synthetic data generators), filters (transforms data), and sinks (file writers and rendering module that provide images to a user interface) in the visualization pipeline. ...

There is some incoherency in the first paragraph of section 3. The paragraph starts by describing the "conventional co-processing" and without transition discusses the NUOPC cap (which, if I understand well, is part of the novel approach and not the conventional one); please clarify.

The paragraph is modified slightly and the wrongly placed "NUOPC cap" is replaced with the "simulation code". The first paragraph in Section 3 describes the conventional co-processing systems and second one is for ESMF integrated one.

Also, in the abstract and introduction, you use the word "conventional" associated to "post-processing", which may lead to some confusion. I would advice to use "traditional" instead of "conventional" when qualifying the post-processing and keep "conventional" when qualifying the co-processing.

The text is modified based on your suggestion. Please refer to the track changes.

52) Can you specify how you calculate the overhead? Do you compare the CPL wall clock-time to the sum of the standalone OCN and ATM wall clock time as they run sequentially? Please clarify.

Yes. It is correct. The text is modified as "… The overhead is calculated by comparing the CPL wall clock-time to the sum of the standalone OCN and ATM wall clock time as they run sequentially. …"

53) Please add the justification of the vertical interpolation in the manuscript.

The following sentence is added to the end of the Section 3. "… In this design, the vertical interpolation is introduced to have a consistency in the vertical scales and units of the data coming from the atmosphere and ocean components. …"

54) I agree with the reviewer that 40% is very significant. I suppose it is linked to the fact that the components are all run sequentially (if I understand well). You should say something about the overhead if the co-processing was run concurrently; more processes would be needed but the wall-time would probably not increase.

Yes. The components are run in an order using same compute resource (or cores) and components waits for co-processing components to render data. In a computing environment without GPU support, the rendering of the information in co-processing component takes more time and rest of the model components waits for their order and co-processing component become a bottleneck for whole modeling system. It is clear that having co-processing component that runs concurrently and process the data along with the simulation will boost the performance of the modeling system. The following text is added to the manuscript "In this case, the components are all run sequentially, and the performance of the co-processing component becomes a bottleneck for the rest of the modeling system especially for the computing environment without GPU support like the system used in the benchmark simulations. It is evident that if the co-processing were run concurrently in a dedicated computing resource, the overall performance of the modeling system would be improved because of the simultaneous execution of the physical models and co-processing components."

59) I cannot really comment the figures, as I don't understand the algorithm, see my remark above.

The paragraph is simplified and low-level design details are removed.

61) You should mention somewhere in the text that figures 9 and 10 show the wall clock time and the speed-up. Please define precisely how you calculate the speed-up either in the text or in the captions.

The following text in Section 4.3 "… The benchmark results of standalone model components (ATM and OCN) can be seen in Fig. 9 …" is changed as "… The measured wall clock time and the calculated speed-up of standalone model components (ATM and OCN) can be seen in Fig. 9 …". I also included "Similar to the benchmark results of the standalone model components, the measured wall clock time and the calculated speed-up of the coupled model simulations are also shown in Fig.~\ref{fig:10}." to the text. Additionally, calculation of the speed-up is included as caption.

Additional remarks

- For OASIS3-MCT, please cite: "Craig A., Valcke S., Coquart L., 2017: Development and performance of a new version of the OASIS coupler, OASIS3-MCT_3.0, Geoscientific Model Development, 10, pp. 3297-3308, doi:10.5194/gmd-10- 3297-2017"

It is added to the references and also the text.

- Many places in the text, you use « ParaView, Catalyst » to design the ParaView co- processing plugin. Please use « ParaView Catalyst » without the comma or « ParaView/Catalyst »

They are fixed and replaced by ParaView/Catalyst.

- COP is used sometimes to design the co-processing component and sometimes the three-component coupled system simulations. This is confusing. Please use "co-processing component" for the component and keep COP to design only the three-component coupled system simulations. Please define COP and CPL the first time it appears in the text.

The manuscript is modified based on your suggestion. For component "co-processing component" and for three-component simulations "COP" are used. In the Section 4.3, the following text "… in the coupled model simulations (CPL and COP)" is changed as "… in the coupled model simulations; CPL (two component case: atmosphere-ocean) and COP (three-component case: atmosphere, ocean and co-processing component) …".

- Fig. 7 is misleading. It looks like the ATM and OCN components are coupled through the co-processing component. If I understand well, this figure should just illustrate the interaction between ATM and the co-processing component on one side, and the interaction between OCN and the co-processing component on the other side. If I am right could it be possible to split the figure into two parts so to avoid the confusion?

The figure aims to demonstrate grid transfer feature and mapping domain decomposition configuration for each model components in co-processing side. In my opinion, splitting figure does not help because there is no direct interaction among model components and all information need to be passed through the driver component. This will still cause confusion. So, i prefer to keep only one figure but in this case, i modified the figure to include also driver component to be clear (please see new version of Fig. 7).

- In section 4.2, an HR (inner) atmosphere is nested in an LR (outer) atmosphere. In section 4.3, HR and LR are used, if I understand well, for two different atmospheres covering the whole atmospheric domain (no nesting); this is confusing. Can you keep HR and LR for section 4.3 only, and change the wording when referring to the HR-inner atmosphere nested in the LR-outer atmosphere in section 4.2?

In the use case and benchmark analysis, we used offline nesting approach (i changed word "one-way" to "offline" in Sec. 4.2) for atmosphere component. The HR domain is nested inside of LR domain and retrieve initial and boundary condition form LR model. This is already explained in Fig. 4.2. The Section 4.2 and 4.3 are checked again to prevent any confusion.

- P.14 , I.4: "their THREDDS server"; can you explain what THREDDS means or give a link?

Following sentence is added "… their THREDDS (Thematic Real-time Environmental Distributed Data Services) data server (TDS). THREDDS is a service that aims to provide access to an extensive collection of real-time and archived datasets, and TDS is a web server that provides metadata and data access for scientific datasets, using a variety of remote data access protocols. The ocean model …"

- P.14, I.30: You refer to section 2.5 for details on the limitation of the co-processing about its sequential type execution, but I don't see where this is detailed in section 2.5. Can you clarify or point me to the exact paragraph?

Section 2.5 includes detailed information about sequential and concurrent type execution of model components not the limitation of the co-processing component under sequential type execution. To prevent any confusion, the "… (see Section 2.5) for more information) …" is removed from the manuscript.

- Table 1: Pipeline details are not readable. Table 1 caption: please remove "in" in "… are shown in here …"

It is hard to use bigger figures for the visualization pipeline shown as a screenshot of ParaView pipeline browser due to the size of the table and used number of columns to show three pipeline in the same time. As a workaround, the visualization pipelines will be included as supplementary material in the final version of the manuscript (see Fig. 1-3 in the supplementary material). The caption is modified as suggested and "… The visualization pipelines are also given as supplementary material …"

- P.15, l.6-8: I don't understand why you write "It is also shown that around 588 processors, which is the highest available compute resource, the communication among the processors dominate the benchmark results and even HR case does not gain further performance": the HR curve (triangles) does not flatten as the LR curve. Please clarify.

The sentence "… results and even HR case does not gain further performance …" is changed as "… results of LR case, but it is not evident in HR case and scales very well without any performance problem …"

- Fig. 9: I don't understand what the envelope represents and what do you mean by "as a line". Why isn't the best configuration the lower limit of the envelope?

In Fig. 9b, the envelope represents the timing results that are done using same number of core but different two-dimensional decomposition configuration. For example, the possible two-dimensional decomposition configurations can be 1x28, 2x14, 4x7, 7x4, 14x2 and 28x1 for 28 cores. The figure is modified and now the solid line shows the lower limit of the envelope in wall-clock time measurements and upper limit in scaling results. The numbers represent the best tile configuration in timing results. The caption of the Fig. 9 is also modified as "Benchmark results of standalone (a) atmosphere (ATM; both LR and HR) and (b) ocean (OCN) models. Note that timing results of the atmosphere model are in log axes to show both LR and HR cases in the same figure. The black lines represent measured wall clock times in second and red lines show speed-up. The envelope represents the timing and speed-up results that are done using the same number of cores but different two-dimensional decomposition configuration. The best two-dimensional decomposition parameters are also shown in the timing results for the ocean model case."

- P. 15, l.23: I don't understand what "acceptable when increased number of MPI communication between the components are considered" means in this context. Is it a justification/explanation of the 5-10% overhead? Please rephrase.

Yes. It is just justification for slower model performance (5-10%) along with increased number of cores. The overhead also includes overhead of the driver component (data transfer, synchronization and remapping). The sentence is also modified as "… The extra overhead is mainly due to the interpolation (sparse matrix multiply performed by ESMF) and extrapolation along the coastlines to match land-sea masks of the atmosphere and ocean models and fill the unmapped grid points to exchange data (Fig. 4) and slightly increases along with increased number of cores as well as number of MPI communication between the model components (Fig. 9 and 10a). …"

- I think you don't need to put a capital letter to each word in titles in English.

They are fixed as suggested.

- Color scales in Fig 12, 13 and 14 are not readable.

Again, this is mainly related with the limited space. It is possible to zoom the figures to see the color scales for the online version. Also, they are included into the supplementary material as an individual figures to make them more readable and larger. The following text is also added to the captions of Fig. 13 and 14 "… The larger versions of figures are also given as supplementary material …"

- P.19, l.15: Please rephrase "have higher water content in a decreasing trend with height and spatial distribution" as it does not seem grammatically correct to me.

The sentence is modified as "Also, the analysis of cloud liquid water content shows that low and mid-levels of the hurricane have higher water content and spatial distribution of precipitation is better represented in high-resolution case (Fig14a-b and d-e), which is consistent with the previous modeling study of Trenberth et al., 2007."

- P.21, l.23: Please modify the sentence "that mounts the user mode components of the driver and the GPUs into the container at lunch" as it is not understandable for a non-expert reader.

The sentence is simplified as "…To enable portable GPU-based containers, NVIDIA developed a special container that loads GPU driver into the container at lunch. …"

Changes made in the manuscript version 2

Response to Reviewer I

Title, Abstract

1) title Prepositions/articles missing. Also, I don't agree with "integrated" but suggest to use the word "modular". As the visualization has been proven, it is not a perspective "towards", but already production ready, so don't use "towards". I would rephrase the title (this is a mere suggestion) as "Co-processing online visualization: the Regional Earth System RegESM version 1.1".

In the current implementation, the adaptor code between NUOPC cap and ParaView, Catalyst is just a prototype that supports only for regional modelling cases with limited spatial grid support (structured and curvilinear grids). It still needs additional work to be more generic and adaptor code needs to be extended to include support for different mesh types such as unstructured grid for both global and regional applications. These are not implemented yet and needs to design adaptor code carefully for more generic applications. This will be done in the future.

2) abstract Fix language issues, such as: articles, avoid "being" before participles, check use of due to/because of, avoid overly use of "used", avoid "have to", do processes literally "move"? Avoid use of "basically", and many more. It is a major revision task to carefully check each sentence

Abstract section is modified based on your suggestions.

3) abstract Avoid jargon especially in the abstract, it may not be clear to all interested readers what "multi-scale", "processing pipeline", "run process". Find alternative/easier wording or explain (as these terms are necessary and useful later in the manuscript). Define I/O (and consistently all abbreviations), define ESMF correctly.

Abstract section is modified and simplified by removing most of the domain specific jargons.

Introduction

4) p1 l23 Use consistent spelling for Earth System Model (capitalized), introduce abbreviation already here. Also use consistent spelling of Earth (name of our planet) versus earth.

The text is modified based on your suggestions.

5) p2 l4 Unsubstantiated claims "more demanding compared to conventional standalone models, and in many cases, to global climate models". Also standalone models can be computationally demanding and though less complex may employ a higher resolution in uncoupled contexts and lower resolution in coupled context, arriving at the same computational demand. And if global ESM were less demanding, also their resolution would increase aligned to available compute re- sources and user requirements.

The sentence is removed.

6) p2 l4 You have not yet defined "obvious needs". What are "truthful predictions" in a model context? Aren't they rather "accurate hindcasts" and "realistic projections".

The sentence is modified based on your suggestions and moved to the end of the paragraph.

7) p2 l10f avoid "in order", just use "to". Avoid "moreover" (throughout manuscript). Try to condense sentences and leave out all unnecessary words throughout the manuscript.

The sentence is modified based on your suggestions.

8) p2 l12 "are also simplified" is not necessarily true. This "can" be simplified or is sometimes even "enforced", such as the grid specification in ESMF, or the standard names by CF conventions.

The sentence is modified based on your suggestions.

9) p2 l16ff ESMF suggests to use Theurich 2016 et al. for referencing ESMF. "in addition" does not align with "can be given as examples". MCT only provides part of the functionality that is present in EMSF (also FMS), as does OASIS (in its different); OASIS may or may not contain MCT, depending on version. There are also more recent developments, such as Jcup3, OMUSE, MOSSCO, MESSy, FABM, CSDMS which you may mention. Please clean up the list, discriminate couplers from coupling frameworks, sort by relationship and highlight key differences or commonalities. A review of older technologies is provided by Bert Jagers 2010.

The more recent reference of ESMF is added to the list. The literature of couplers and coupling frameworks are also extended. Please refer to the supplementary document that includes track-changes.

10) p2 l19 "The ESMs make use of state-of-art modeling frameworks, model components and libraries to have better representation of the earth system processes". This sentence is irrelevant. "The key component of coupled modeling systems is often the driver". Please refer to Alexander/Easterbrook (2015) for typical coupling constellations in ESMs. There, it is quite obvious that many models do not employ a driver. You could also use the term "hub-and-spoke" architecture (consistent with your Figure 1) where the driver is the hub. The word "driver" itself is jargon and should be introduced beforehand.

The paragraph is also modified to refer Alexander/Easterbrook (2015) paper that briefly demonstrate the diversity of the current global earth system models.

11) p2 l19 "such as six hourly or daily averages". This is true only for coupled physical systems. For geology ESMs, typical timescales may be longer (years, centuries), see e.g., CSDMS.

It is removed from the sentence.

12) p2 l23 "Coupling between individual earth system components adds extra overhead... network bandwidth usage, computation, disk I/O and data storage". Again, not necessarily. Please substantiate or give an example or reference where this applies. In my opinion the real overhead is in the loss of flexibility/modularity: it is preferable to have 1 hub speak a common language to 4 components than have each of the 4 components speak to each of the other three. Though the latter may be less complex, it hinders extensibility of a coupled system.

The misleading sentence is removed from manuscript.

13) p3 l3f Fix articles. As everywhere.

I am aware that there are numerous missing articles in the manuscript and these are fixed in the revised version of manuscript.

14) p3 l7 Don't overly use "in short", also throughout manuscript.

It is modified as "as a result" in here.

15) p3 l7 Don't use judging language such as "brilliant". This entire sentence could be deleted (and the keyword "exascale" mentioned before.

It is changed to "state-of-art" and "exascale" is deleted.

16) p4 l4ff This paragraph is not needed.

It is removed.

Design

17) p4 l9 "couple four different model components". I don't understand. Version 1.0 coupled four components, version 1.1 adds a co-processing component, totalling five. This is also confusing in Figure 1. Modular frameworks such as NUOPC/ESMF do not discriminate scientific models, data components, out-put/visualization per se. All are just "components". I would suggest to use the term "component" for all five (four of them science models).

The paragraph is modified based on your suggestions.

18) p4 l11 delete "state-of-art"

It is removed.

19) p4 l11 You have not yet introduced "the coupling interfaces" (this may be an article problem). So far, a driver and four models have been introduced. The necessity to add also coupling interfaces, and possibly NUOPC caps (CMI in the framework of Peckham 2013), has not yet been mentioned.

The coupling interface is introduced before its use and hub-and-spoke architecture is also mentioned in here.

20) p4 l16ff delete this paragraph

It is removed.

21) p4 l section heads add ATM (later OCN, WAV, COP) to section heads

They are added

22) p4 l22 For in-text enumerations, please use (1) with double parentheses, and do not write the number in bold face.

It is fixed. Same convention is also used in everywhere in the manuscript.

23) p4 l26 "it is well tested with verified in". This does not make sense grammatically, please fix. Also, you have a verified model system for South and Eastern European/West Asian regions, but not for the Gulf of Mexico, which you provide as an example for the visualization.

It is fixed. The RegCM atmospheric model component is used because WRF is not ready to couple with co-processing component. I still need to add three-dimensional fields such as mixing ratio etc. along with support of vertical interpolation. We are currently working on it as a part of ongoing national science project and will be ready soon (in version 1.2).

24) p4 l27f Avoid "Additionally, ". What relevance has "does not support online nesting yet" for the application discussed here. Is online nesting needed for coupling to the co-processing model. Or is it just not "operational" in a sense that RegESM/WRF has not been scientifically proved?

It is fixed. No, there is no need to use on-line nesting to couple with co-processing. It is also not supported in the current version of the coupled modeling system. So, it is removed. We are planning to redesign whole modeling system from scratch to support online nesting among model components but this is our long-term goal for RegESM 2.0. We are also having fully coupled WRF-MITgcm model for Black Sea but the results are not published yet. Along with a help of newly introduced co-processing component, the Black Sea model will be used to investigate lake-snow effect as a part of same ongoing national project.

25) p5 sections 2.1.1/2.1.2 Align in model descriptions of RegCM and WRF the content. Use the same structure: dynamical core, spatial representation, explicitly modeled states and subgrid-scale parameterizations, sub-modules. This parallel construction helps to compare the ATM science models.

Both sections are restructured to align their content as much as possible. In this case, they have very brief information about models and reader must refer to their own documents to get more information.

26) p5 l9 "the driver and coupling interface are developed by ITU". The driver should be independent of RegCM, why is this mentioned? What is the coupling interface? Is it the CMI/NUOPC cap? You may choose to introduce those two terms to make clear what you mean by coupling interface. Please describe the cap (e.g. what state variables does it export/import, which transformations does it encompass). You may also elaborate on the changes internal to RegCM (and WRF/MITgcm, WAM/HD)

The sentence that refers to ITU and ENEA is removed. The processes handled by NUOPC cap are listed in here because they are same for both atmospheric model component. The NUOPC cap and details about the implementation are added.

27) p5 l21 "model is modified to exchange data". Discriminate work done on cap (CMI) and within model (BMI, Peckham 2013).

The text is modified and now the distinction between modifications done in model side and cap is more apparent. In each model section, only work done in model side is described.

28) p5 l23ff All that was said in the ATM section also applies here (parallel structure, elab- oration ...)

Both sections are restructured to align their content as much as possible.

29) p6 l2ff Avoid judging language like "fortunately". Avoid "in general" if possible. "In some studies", please reference those studies or give a motivation for both cases.

 "fortunately" is removed from the manuscript. "In general" is also removed and i refered to our previous work for the bulk flux algorithm.

Turuncoglu, U. U., Giuliani, G., Elguindi, N., and Giorgi, F.: Modelling the Caspian Sea and its catchment area using a coupled regional atmosphere-ocean model (RegCM4-ROMS): model design and preliminary results, Geosci. Model Dev., 6, 283-299, https://doi.org/10.5194/gmd-6-283-2013, 2013.

In this study, we have to use bulk flux algorithm because ice model coupled with ROMS (not latest CICE version of ROMS, Budgell, 2005 version is used) was not working with the direct fluxes provided by atmospheric model.

30) p6 l7f "In the current design of the coupled modeling system, the driver allows to select the desired exchange fields from the predefined list". Please explain how this works. Is the list like a database with all known fields from all possible coupled models and there is an automated selection of suitable fields?

It is just a list of fields that component could import and export. The automatization could be implemented in future version of the modeling system by using ontologies, common vocabularies and conventions.

31) p6 l15 Leave out example technical detail "REGCM_COUPLING". I would appreciate a list of changes in the supplementary material, if that is practical to do.

It is removed. The documentation of the RegESM modeling system includes all those tiny details. The reader could refer to the following link https://github.com/uturuncoglu/RegESM

32) p6 l23 define PROTEUS abbreviation

Actually, I searched in web and their publications but i could not find the source of the abbreviation but ENEA is using it for their publications and web site. So, I prefer to keep it as it is.

33) p6 l26 PBL has already been defined

It is corrected.

34) p6 l26f third-generation wave model. What is "pure" physics? Use an en-dash to indicate a relationship in wave-wave / wave–wave (the second one is correct typography)

"pure physcis" is removed. wave-wave is also written with en-dash now.

35) p6/7 l32 WAM is not by ENEA but by GKSS (now HZG) in Germany, I believe. Please attribute correctly. Describe changes to original model (here in text or as table in SOM).

It is fixed now. The list of changes in WAV side is also included.

36) p7l5 Unit of z0(m).

It is added.

37) p7 l20f Explain your motivation for the two different ways. Do they pertain to ROMS and MITgcm coupling, respectively? How are they chosen?

The first option is used to define river plumes correctly while the second one is generally used to distribute river discharge to a large areal extent over the ocean. These are defined in the configuration file of the driver. The section is also modified to include this information.

38) p7 l22f The change to output format is irrelevant in this context.

It is removed.

39) p7 l30f "The ESMF is chosen because ... of the NUOPC". Please explain the relationship between ESMF and NUOPC. Why do you chose NUOPC in addition to ESMF, this needs elaboration for readers who are not familiar with NUOPC.

The section is extended to include more information about relationship between ESMF and NUOPC interface.

40) p8 l2f "It also provides the capability of transferring computational grids"; here, "it" should refer to ESMF, not NUOPC.

It is fixed.

41) p8 l2f "NUOPC layer to support various configuration of component interactions such as defining multiple coupling time steps (fast and slow time steps; Fig. 2)". There are other frameworks that provide add this capability to ESMF, namely MAPL and MOSSCO, please reference and contrast.

Actually, this feature is mainly inherited from ESMF by MAPL and MOSSCO. I think it is better to reference to ESMF/NUOPC for it.

42) p8 l9f "connector components provided by NUOPC layer. Connector components are mainly used to create link between individual model components and driver". As far as I know, RegESM includes its own instance of a NUOPC connector, which is a single one for all connections between the components. Please elaborate on the design and functionality of the connector.

There is no single connector in the modeling system. For example, when RegESM is configured to couple only atmosphere and ocean components. The modelling system creates two connectors to represent ATM-OCN and OCN-ATM interactions. This is little bit different when atmosphere model coupled with co-processing component. In this case, there is only one connector because only ATM-COP interaction is valid. COP component does not send any information to other components (like a sink). The section is extended to include this information.

43) p8 l18f You name stability as an advantage of semi-implicit coupling. Also add draw-backs of semi-implicit and advantages of implicit time stepping.

The paragraph is extended to include more information about different coupling schemes.

44) p8 l20f delete "mainly" and "In this case"

They are removed.

45) p8 l28 You may want to add the term "load balancing" for describing the challenges in concurrent coupling. There should be plenty of literature that describes load- balancing and possible ways to minimise this. In the geosciences, usually all OASIS-coupled models face this challenge.

Yes, the right term is load-balancing. A reference is added about LUCIA (Load-balancing Utility and Coupling Implementation Appraisal) tool for load-balancing analysis in OASIS-MCT based coupled systems.

46) p8 l31f The description of regridding could be a subsection, section 2.6 "Connector". How do other coupled model systems handle this (any references available?). Recently, I implemented into MOSSCO a very similar scheme using bilinear for interpolation and nearest neighbour for extrapolation, so I believe this is a very good way to handle this (but it is not published yet). Possibly, describe how ESMF 8 handles this rather than pointing vaguely to ongoing development.

I prefer to keep it as it is. The special interpolation implemented in RegESM modeling system is a part of driver.

47) p9 l14 Align your description with the terms used in Figure 4 (see comments below on this figure).

We used field names (Field_A, Field_B and Field_C) in also manuscript. The paragraph is also extended to describe the algorithm clearly.

Integration of co-processing

48) p9 l25 Use an em-dash "—" instead of a hyphen. One sentence is not a paragraph, so join with next paragraph.

It is fixed.

49) p9 l28 Define "conventional" or give a reference to such approaches. Define pipeline in this context.

The conventional co-processing enabled simulation systems interacts with single physical model component such as atmosphere along with co-processing support. This is defined also in text and references are added.

… A visualization pipeline integrates a data flow network in which computation is described as a collection of executable modules that are connected in a directed graph representing how data moves between modules (Moreland, 2013). There are three types of modules: sources (file readers and synthetic data generators), filters (transforms data), and sinks (file writers and rendering module that provide images to a user interface) in the visualization pipeline. …

50) p9 l30 How is "wrapper layer" different from Component Model Interface (CMI) or NUOPC cap or ESMF component interface?

A NUOPC cap is defined for co-processing component along with adaptor code. In this case, adaptor code is an interface for ParaView, Catalyst API and NUOPC cap passes information from ESMF to adaptor code. The text slightly modified to clear the design.

51) p10 l3 On what "other hand"? Describe the relationship between the adaptor and its NUOPC cap

It is removed. It is defined in previous paragraph.

Benchmark

52) p14 l26 The "driver component introduces additional 5-10%". In ESMF documentations, usually below 5% overhead is assumed. this makes sense as you to the SMM twice for each interpolation of mask extrapolation. It would be interesting to see whether the generic implementation of extrapolation is more efficient in ESMF 8 (out of scope of this MS).

I agree. As far as i know, the ESMF developers was trying to find a way to combine weight matrix of multiple interpolation (including extrapolation) into a single one. Similar methodology is using in transformation (scale, rotation, translation etc.) matrixes using homogeneous coordinates but i am not sure it could be used in regridding or not.

53) p14 l35 " includes vertical interpolation to map data from sigma coordinates to height coordinates". Why is this interpolation step necessary? In principle, the visualization should equally work on sigma layers. And is the interpolation performed within the ESMF connector, or within a Paraview pipeline?

I did not try yet but yes the visualization works with sigma coordinates without any problem. The vertical interpolation is introduced to have a consistency in the data coming from atmosphere and ocean. By this way, the vertical scales of the data can be compared without any problem. In general, both atmosphere and ocean components use sigma coordinates with different definitions. The interpolation is done in ESMF cap not in ParaView side. It could be done in ParaView side also but in this case all grid related parameters such as stretching functions etc. need to be passed to ParaView, Catalyst and special ProgrammableFilter need to be developed.

54) p15 l2 the computational demand for visualization "require 10-40% extra" is rather high and its evaluation would probably change depending on whether it is more 10% (acceptable) or 40% (not acceptable). The benchmark using software rendering with Mesa is not ideal to demonstrate this, especially as many new HPC systems come with dedicated GPU nodes that could much reduce the visualization overhead.

It is clear that Mesa is not right way to measure the performance of the co-processing component. The overall performance of the used computing system dominates the results in our case but we don't have accesses to a system with GPU accelerators. In the early stage of the performance benchmark, Dell provided us a grant in newly installed University Manchester system (10000 core/hour) but the system is mainly designed for GPU intensive and AI type researches and the nodes have small number of core (10 cores) but four NVIDIA P100 GPU. In fact, our test simulations are CPU intensive and requires high amount of compute resource (or cores) for the high-resolution simulations and rendering only done on a single GPU (limitation of ParaView, Catalyst and used filters). As a result, couple of benchmark run spend all allocated resources (unused GPU resources was also accounted) and the system was not used for the benchmark. There was also issue related with buggy NVIDIA driver and ParaView protobuf library. In the future, along with the development of m-VTK (GPU accelerated filters) it could be possible to use multi-GPU systems.

Results and Conclusions

55) p15 l31 This section is not well named. I suggest "Demonstration application" but make sure to choose something more elaborate than "results"

It is changed as suggested.

56) p15 l35 oblivious => obvious

It is fixed.

57) p16 l3 Explain how you choose between live mode and co-processing modes. Are these two different models? Or is this done by way of a configuration file. Or, is it possible to attach live viewing to any running system that has co-processing enabled?

No, it is not a different model or configuration. The selection of the mode is controlled by the Python script generated using ParaView, co-processing plugin. The user could also activate live visualization mode, just by changing a single line of code (coprocessor.EnableLiveVisualization needs to be set as True) in the co-processing Python script. Yes, if live visualization is enabled, the user can attach to the simulation anytime and pause it to analyze the results. This information is added to section.

Figures

58) Figure 1 Very clear and readable, consistent color and design with NUOPC web page. Please see my suggestions to treat everything as component, regardless of sci- ence "MODEL" or visualization "COMP". It is not clear from the text what "redist" is; also, this feature is not used when running sequentially. I also do not under- stand the many "regrid" arrows, as regridding occurs when the driver transports a field from one to another component, not from the component to the driver. This is confusing.

The figure is modified. All the boxes are named as "COMPONENT" and the texts are also removed from the arrows.

59) Figure 4 Rather unreadable. Please redesign. (1) Avoid bridges (this is possible by routing the arrow "Create ESMF FIELD_B" north to "Create ESMF FIELD_C". (2) don't use same line thickness/color for outlining the boxes around src/dst grid, rather make them visually less intrusive to highlight the flowchart. (3) Abstract away 1.0, LARGEST, "over sea" or other value-based masking, if possible.

The figure is modified. The hop is solved by changing the position of the arrow. The dashed line is used in the boxes and color is changed to grey to move focus to workflow. It is hard to abstract used values and mask values. The algorithm is tested with this configuration and it is better to keep them as it is to have a reproducible result (at least for finding mapped and unmapped grid points). I am also using same methodology using NCL ESMF interface to create forcing for ocean models and it works better than standard extrapolation method (based on solution of Possion Eq.) provided by NCL itself.

60) Figure 7 It is confusing that the visualization 2x2 tiles don't geographically overlap for data from OCN and ATM in CPL. This is not needed, but confusing at first. Maybe give a hint to the reader that this is not a problem.

The order of plots and labels of Fig 6 and Fig 7 was wrong. It is fixed now. Following section is also included to the manuscript

… In this case, ATM and OCN model components do not need to geographically overlap for co-processing. The only limitation is that the ATM component must cover the entire OCN model domain for an ATM-OCN coupled system because the ATM component provides the surface boundary condition for OCN component. …

61) Figures 9/10 It is very difficult to compare ATM versus OCN performance using different y scales. I recommend to use the same ylog-scale for all panels in figs 9 and 10. Also try loglog. Tile sizes are hard

The speed-up calculation of coupled model simulations starts from 140 core because this is the minimum number of cores that allows to run the modeling system. For standalone simulations, it is possible to run the model with 28 cores (single node). As a result, it was used 28 cores for standalone and 140 cores for coupled model simulations to calculate the speedup. The plots are modified to have consistent x and y axis (log scale) as well as number of cores used in the speed-up calculation (140 cores) except plot for standalone ocean model. In this case, using same log scale for it does not give good results as expected for example the effect of tile configuration cannot be seen clearly because the benchmark results fall into a very narrow interval. To that end, i prefer to keep results of standalone ocean simulations as it is. I also increased font sizes and move the text little bit far from the markers for standalone ocean model plot. The texts that indicate model version and name of the computing system that is used in the benchmark are removed because they are already mentioned in the manuscript.

62) Figure 10 8 lines in a graph is too messy. Even if on ylog, this may be difficult. Possibly choose different or more subpanels (e.g. CPL_LR + CPL_HR in one panel, then P1, P2, P3 each in one panel). Or use more colors for lines or different line styles ...

The plot is modified. Now it has four panel. The first one (a) compares coupled model simulations (w/o co-processing component), and the rest compares coupled model simulations with co-processing component enabled (LR vs. HR). (b) for P1 visualization pipelines, (c) for P2 visualization pipelines and (d) for P3 visualization pipelines. The both y scales (the left one uses log scale) are also consistent now.

63) Figure 11 Zoom in to Paraview Window only (delete lower and upper OS information bands)

It is fixed.

64) Figure 13 Use kg kg−1 and the like, not the slash for division in units.

The figure label is fixed.

Code

65) I have been able to download and compile all components, except for getting the catalyst to run. It would be helpful to provide a fully functional reference system as a docker container, for example.

The Docker container was used to do online demo of the modeling system in the last NVIDIA GTC (GPU Technology Conference). More information about demo case and created Docker container (using software rendering) can be found in following public repository (DOI is also created by Zenodo and referenced in code availability section).

https://github.com/uturuncoglu/GTC2018_demo

There is also Docker container (NVIDIA docker need to be installed - https://github.com/NVIDIA/nvidia-docker) that supports NVIDIA GPUs and can be found in following link

https://drive.google.com/file/d/1hz0Frbawm2UxNtSBnjIrGSHlqDk1kZFY/view?usp=sharing

In this case, the file size is greater than upper limit of GitHub LFS (Large File Support; 2 GB) and it is shared with Google Drive. After getting file following commands can be used to run the Docker container and low-resolution version of coupled model with co-processing support.

1) import container

```
          docker import nvidia_opengl_gtc2018-egl.tar.gz
```
2)   list images to retrieve IMAGE ID
```
      docker images
```
3)   tag IMAGE (52ebebd92125 is just an example for Image ID, use your own from previous command)
```
      docker tag 52ebebd92125 nvidia/opengl:gtc2018-egl
```
4)   run container and get interactive shell
```
      nvidia-docker run -it nvidia/opengl:gtc2018-egl /bin/bash
```
5)   go to simulation directory (all required libraries installed under /opt/progs)
```
      cd /opt/progs/COP_LR
```
6)   run simulation (it will produce png files for each coupling time step)
```
      ./run.sh
```

A new section (Sec. 6) is also included to the manuscript to discuss interoperability, portability and reproducibility.

<u>Supplementary videos</u>

66) Useful and nice! Please make sure the video dois are referenced in the text, and add to the videos a link to your publication.

The video DOIs are included into the manuscript and the reference sections.

Response to Reviewer II

1) "… Since a key focus on the paper is interoperability afforded by using a standard coupling framework, some additional discussion on details of the software engineering and approach to interoperability could be discussed including more details on the actual interfaces used between components as well as issues related to portability. Follow on work could look at applying the same co-processing component to a completely different model to understand how generic the approach is. A related question is how hard it would be to change out the visualization package itself, since there are a number of packages that offer custom analyses …"

The modeling system is tested in different variety of computing environment ranges from single server with dedicated NVIDIA GPU (M60) to cluster designed to GPU intensive applications (each node has four NVIDIA P100). Using experience to teste the modeling system with a variety different configuration and computing environment, a completely new section (Sec. 6) named as "Discussion of the concepts associated with interoperability, portability, and reproducibility" is added to the manuscript just before result and conclusion section.

Currently, as a part of ongoing national project, we are working on applying model to different use cases that air-sea interaction plays important role such as wind introduced upwelling in Aegean Sea and the relationship between SST anomaly and precipitation in Black Sea. In both cases, different coupled model configurations are used. In the first case, the coupled modeling system configured to use WRF and ROMS but in the second use case WRF and MITgcm configuration is used. The initial results show that the co-processing component is able to render information coming from different model setups. The only requirement is that the NUOPC cap of each different ocean and atmosphere model need to be modified slightly to provide also three-dimensional fields along with two dimensional ones. This also requires to define 3d grid representation in the NUOPC cap of the model component as well as states to allow transferring 3d fields to co-processing component. In theory, using different visualization tool such as VisIt (developed by LLNL, https://visit.llnl.gov) only requires modification in the adaptor code resides between NUOPC cap and visualization tool.

2) "… In addition to timing profiles and since the initial motivation was around the problem of data volumes, the paper could benefit from plots describing the amount of data exchanged and used in the in-situ case versus the amount of data that would be required for offline visualization at the same temporal frequency. This would allow an "apples-to-apples" comparison of the online vs. offline approaches.…"

The amount of data exchanged with co-processing component is given in Table 1 for three different visualization pipelines (P1, P2 and P3). In in-situ visualization mode, the data is read from the memory and passed to the ParaView, Catalyst for rendering. In this case, live visualization allows to connect the running simulations and make real-time data analysis of simulation results. If the modeling system configured to run in co-processing mode then rendered image files or data can be stored in the disk. Besides, processing data in co-processing component concurrently with the simulation, the offline visualization (post-processing) uses stored data to make analysis. In in-situ visualization mode, user needs to store variables only for single time snapshot in the memory. Unlike in-situ visualization, offline visualization performed after simulation ends and requires to store data produced by the whole simulation. For example, in case of using offline visualization, 3-days long simulation with 6-minutes coupling interval produces around 160 GB data (720 time step) just for single variable from high-resolution atmosphere component (P1 visualization pipeline). The same analysis can be done with in-situ visualization just by storing single time step, which is the size of 224 MB. Moreover, the netCDF formatted output of RegCM atmosphere model (used in this study) contains 7 x 3d fields and 28 x 2d fields in default configuration and only three-day high-resolution simulation produce around 1.5 TB data in case of using 6-minutes interval. In this case, the data size that needs to be stored in the disk is also depend on used number of active model components, and selected time interval. It is oblivious that the detailed examination of the simulation results with multiple visualization pipeline requires more variables to be stored in the disk (in VTK binary format) when post-processing method is used. The size of the data also depends on the horizontal and vertical resolution of the model components as well as specified time interval to interact with co-processing component. The desired fps (frame-per-second) in the animation that is produced using output of the co-processing component also defined by the interaction

interval (or coupling time step) and usually 20-24 fps is required to have a real world like animation. The following section is added to Section 4.3.

In addition to the analysis of timing profiles of modeling system under different rendering load, the amount of data exchanged and used in the in-situ visualization case can be compared with the amount of data that would be required for offline visualization at the same temporal frequency to reveal the added value of the newly introduced co-processing component. For this purpose, the amount of data exchanged with co-processing component is given in Table~\ref{tab:01} for three different visualization pipelines (P1, P2, and P3). In co-processing mode, the data retrieved from model components (single time step) through the use of the driver read from memory and passed to the ParaView, Catalyst for rendering. Besides, processing data concurrently with the simulation on co-processing component, the offline visualization (post-processing) consists of the computations that are done after the model is run and requires to store numerical results in a disk environment. For example, 3-days long simulation with 6-minutes coupling interval produces around 160 GB data (720 time-step) just for a single variable from high-resolution atmosphere component (P1 visualization pipeline) in case of using offline visualization. In the case of using co-processing, the same analysis can be done by applying same visualization pipeline (P1), which requires to process only 224 MB data stored in the memory, in each coupling interval. Moreover, storing results of three-day long, high-resolution simulation of RegCM atmosphere model (in netCDF format) for offline visualization requires around 1.5 TB data in case of using 6-minutes interval in the default configuration (7 x 3d fields and 28 x 2d fields). It is evident that the usage of co-processing component reduces the amount of data stored in the disk and allows more efficient data analysis pipeline.

Toward modular in-situ visualization in Earth System Models: the regional modeling system RegESM 1.1

Ufuk Utku Turuncoglu [1]

[1]Informatics Institute, Istanbul Technical University, 34469, Istanbul, Turkey

Correspondence: Ufuk Utku Turuncoglu (ufuk.turuncoglu@itu.edu.tr)

Abstract. The data volume produced by regional and global multi-component Earth System Models is rapidly increasing because of the improved spatial and temporal resolution of the model components, and the sophistication of the numerical models regarding represented physical processes and their complex non-linear interactions. In particular, very small time steps need to be defined in non-hydrostatic high-resolution modeling applications to represent the evolution of the fast-moving processes such as turbulence, extra-tropical cyclones, convective lines, jet streams, internal waves, vertical turbulent mixing and surface gravity waves. Consequently, the employed small time steps cause extra computation and disk input-output overhead in the modeling system even if today's most powerful high-performance computing and data storage systems are considered. Analysis of the high volume of data from multiple Earth System Model components at different temporal and spatial resolution also poses a challenging problem to efficiently perform integrated data analysis of the massive amounts of data when relying on the traditional post-processing methods today. This study mainly aims to explore the feasibility and added value of integrating existing in-situ visualization and data analysis methods within the model coupling framework. The objective is to increase interoperability between Earth System multi-component code and data processing systems by providing an easy-to-use, efficient, generic and standardized modeling environment. The new data analysis approach enables simultaneous analysis of the vast amount of data produced by multi-component regional Earth System Models during the runtime. The presented methodology also aims to create an integrated modeling environment for analyzing fast-moving processes and their evolution both in time and space to support a better understanding of the underplaying physical mechanisms. The state-of-art approach can also be employed to solve common problems in the model development cycle e.g. designing a new sub-grid scale parameterization that requires inspecting the integrated model behavior at a higher temporal and spatial scale simultaneously and supporting visual debugging of the multi-component modeling systems, which usually are not facilitated by existing model coupling libraries and modeling systems.

1

1 Introduction

The multi-scale and inherently coupled ~~(atmosphere, ocean, land etc.) earth system models~~ Earth System Models (ESMs) make them challenging to study and understand. Rapid developments in ~~earth~~ Earth system science, as well as in ~~high performance~~ high-performance computing and data storage systems, have enabled fully coupled regional or global ~~Earth System Models~~

30 ~~(ESMs)~~ ESMs to better represent relevant processes~~and~~, complex climate feedbacks, and interactions among the coupled components. ~~Regional ESMs are generally used~~ In this context, regional ESMs are employed when the spatial and temporal resolution of the global climate models are not sufficient to resolve local features such as complex topography, land-sea gradients and the influence of human activities in a smaller spatial scale. ~~The complexity of the fully coupled state-of-art regional ESMs have made them computationally more demanding compared to conventional standalone models, and in many cases, to~~

35 ~~global climate models. To that end, there are obvious needs to develop novel modelling tools, model coupling frameworks (at the required level of sophistication in terms of both physics and computational methods) as well as innovative and efficient modelling systems to enable truthful predictions based on accurate and realistic representation of earth systems~~ Along with the development of the modeling systems, specialized software libraries for the model coupling become more and more critical to reduce the complexity of the coupled model development and increase the interoperability, reusability, and efficiency of the

40 existing modeling systems. Currently, the existing model coupling software libraries have two main categories: couplers and coupling frameworks.

Couplers are mainly specialized in performing specific operations more efficiently and quickly such as coordination of components and interpolation among model components. For example, OASIS3 (**?**) uses multiple executable approaches for coupling model components but sequentially performing internal algorithms such as sparse matrix multiplication (SMM)

45 operation for interpolation among model grids become a bottleneck along with increased spatial resolution of the model components. To overcome the problem, OASIS4 uses parallelism in its internal algorithms (**?**), and OASIS3-MCT (**?**) interfaced with the Model Coupling Toolkit (MCT; **??**) provides a parallel implementation of interpolation and data exchange. Besides generic couplers like OASIS, domain-specific couplers such as Oceanographic Multi-purpose Software Environment (OMUSE; **?**) that aims to provide a homogeneous environment for ocean modeling to make verification of simulation models with different

50 codes and numerical methods and Community Surface Dynamics Modeling System (CSDMS; **?**) to develop integrated software modules for modeling of Earth surface processes are introduced.

A ~~modeling framework is defined as~~ coupling framework is an environment for coupling model components through a ~~common calling interface .~~ ~~The modeling framework~~ standardized calling interface and aims to reduce the complexity of regular tasks such as performing spatial interpolation across different computational grids and ~~to transfer~~ transferring data among

55 model components ~~, in order~~ to increase the efficiency and interoperability of multi-component ~~coupled model systemsby providing standardized calling interfaces. Moreover,~~ modeling systems. Besides the synchronization of the execution of individual model components~~and~~, a coupling framework can simplify the exchange of metadata ~~among them are also simplified with model coupling frameworks~~ related to model components and exchanged fields through the use of existing conventions such as CF (Climate and Forecast) convention. The Earth System Modeling Framework (ESMF) is one of the most ~~popular~~

60 ~~example for this approach (???)~~famous examples of this approach (?). The ESMF consists of a standardized superstructure for coupling components of Earth system applications through a robust infrastructure of high-performance utilities and data structures that ensure consistent component behavior ~~(Hill et al. , 2004). In addition to the ESMF model coupling framework , the Model Coupling Toolkit (MCT; ??), Model Coupling Environmental Library (MCEL; ?), OASIS (??) and C-Coupler (?) can be given as other examples of this approach.~~

65 ~~The ESMs make use of state-of-art modeling frameworks, model components and libraries to have better representation of the earth system processes.~~ (?). The ESMF framework is also extended to include the National Unified Operational Prediction Capability (NUOPC) layer. The NUOPC layer simplifies component synchronization and run sequence by providing additional programming interface between coupled model and ESMF framework through the use of a NUOPC "cap". In this case, a NUOPC "cap" is a Fortran module that serves as the interface to a model when it is used in a NUOPC-based coupled system.

70 The ~~key component of coupled modeling systems is often the driver, which is mainly responsible for~~ term "cap" is used because it is a small software layer that sits on top of a model code, making calls into it and exposing model data structures in a standard way. In addition to generic modeling framework like ESMF, Modular System for Shelves and Coasts (MOSSCO; ?) creates a state-of-art domain and process coupling system by taking advantage of both ESMF and Framework for Aquatic Biogeochemical Models (FABM; ?) for marine coastal Earth system community.

75 The recent study of ? to investigate the degree of modularity and design of the existing global climate models reveals that the majority of the models use central couplers to support data exchange, spatial interpolation~~and synchronization between model components, subject to their individual computational grids~~, and synchronization among model components. In this approach, direct interaction ~~do~~ does not have to occur between individual model components or modules, since the ~~data transfer is managed by the specialized driver component . Coupling between individual earth system components adds extra overhead~~

80 ~~and complexity to the overall modeling system in terms of network bandwidth usage, computation, disk I/O and data storage. Due to the high volume of data produced by multiple model components, the results are generally stored in a limited temporal resolution such as six hourly or daily averages, which are processed after numerical simulations are finished. This conventional approach is commonly known as post-processing. A typical configuration, e.g. a thirty years simulation of a fully coupled regional atmosphere-ocean model might produce some Terabyte of output (in four dimensions; time, height or depth, latitude~~

85 ~~and longitude), which strongly depend on horizontal and vertical resolution of model components, the number of variables and the time interval defined to store the simulation results. In this case, the post-processing step commonly involves creating temporal (i.e. seasonal climatology, daily and /or monthly averages) and spatial averages, time series, cross-sections in a defined axis and/or custom pathways, added value information produced by set of statistical methods and the visualization of the model results~~specific coupler component manages the data transfer. This approach is also known as the hub-and-spoke

90 method of building a multi-component coupled model. A key benefit of using a hub-and-spoke approach is that it creates a more flexible and efficient environment for designing sophisticated multi-component modeling system regarding represented physical processes and their interactions. The development of the more complex and high-resolution modeling systems leads to an increased demand for both computational and data storage resources. In general, the high volume of data produced by the numerical modeling systems may not ~~enable to store~~ allow storing all the critical and valuable information to use later, despite

recent advances in storage systems. As a result, the simulation results are stored in a limited temporal resolution (i.e., monthly averages), which are processed after numerical simulations finished (post-processing). The poor representation of the results of numerical model simulations prevents to analyze the fast-moving processes such as extreme precipitation events, convection, turbulence and non-linear interactions among the model components in a high temporal and spatial scale with the traditional post-processing approach.

The analysis of leading high-performance computing systems reveals that the rate of disk input-output (I/O) performance is not growing at the same speed as the peak computational power of the systems (**??**). The recent report of U.S. Department of Energy (DOE) also indicates that the expected rate of increase in I/O bandwidth (100 times) will be slower than the peak system performance (500 times) of the new generations of exascale computers (**?**). Besides, the movement of large volumes of data across relatively slow network bandwidth servers fails to match the ultimate demands of data processing and to archive tasks of the present high-resolution multi-component ESMs. As a result, the traditional post-processing approach has become a bottleneck in monitoring and analysis of fast-moving processes that require very high spatial resolution, due to the present technological limitations in high-performance computing and storage systems (**?**). In the upcoming computing era, state-of-art new data analysis and visualization methods are needed to overcome the above limitations evocatively.

Besides the traditional data analysis approach, the so-called in-situ visualization and co-processing approaches allow researchers to analyze the output while running the numerical simulations simultaneously. The coupling of computation and data analysis helps to facilitate efficient and optimized data analysis and visualization pipelines and boosts the data analysis workflow. Recently, a number of in-situ visualization systems for analyzing numerical simulations of Earth system processes have been implemented. For instance, the ocean component of Model for Prediction Across Scales (MPAS) has been integrated with an image-based in-situ visualization tool to examine the critical elements of the simulations and reduce the data needed to preserve those elements by creating a flexible work environment for data analysis and visualization (**??**). Additionally, the same modeling system (MPAS-Ocean) has been used to study eddies in large-scale, high-resolution simulations. In this case, the in-situ visualization workflow is designed to perform eddy analysis at higher spatial and temporal resolutions than available with traditional post-processing facing storage size and I/O bandwidth constraints (**?**). Moreover, a regional weather forecast model (Weather Research and Forecasting Model; WRF) has been integrated with in-situ visualization tool to track cyclones based on an adaptive algorithm (**?**). Despite the lack of generic and standardized implementation for integrating model components with in-situ visualization tools, the previous studies have shown that in-situ visualization can produce analyses of simulation results, revealing many details in an efficient and optimized way. It is evident that more generic implementations could facilitate smooth integration of the existing standalone and coupled ESMs with available in-situ visualization tools (**???**) and improve interoperability between such tools and non-standardized numerical simulation codes.

The main aim of this paper is to explore the added value of integrating in-situ analysis and visualization methods with a model coupling framework (ESMF) to provide in-situ visualization for easy to use, generic, standardized and robust scientific

130　applications of ~~earth~~ Earth system modeling. The implementation allows existing ~~earth system models~~ ESMs coupled with the ESMF library to take advantage of in-situ visualization capabilities without extensive code restructuring and development. Moreover, the integrated model coupling environment allows sophisticated analysis and visualization pipelines by combining information coming from multiple ~~earth system model~~ ESM components (i.e., atmosphere, ocean, wave, land-surface) in various spatial and temporal resolutions. Detailed studies of ~~key~~ fundamental physical processes and interactions among model

135　components are vital to the understanding of complex physical processes and could potentially open up new possibilities for the development of ~~earth system models~~ ESMs.

~~The next section (Sect. ??) describes the design of a fully coupled regional earth system model used in this study. Section ?? provides information about the implementation and design of the in-situ visualization integrated modelling framework employed. Then, the initial results from newly designed modeling framework will be demonstrated in Section ??, along with~~

140　~~some benchmark results (Sect. ??). Finally, a summary and discussion are given in the last section (Sect. ??).~~

2　The ~~Design~~ design of the ~~Modeling System~~ modeling system

The RegESM (Regional Earth System Model; 1.1) modeling system ~~is able to couple four~~ can use five different model components ~~(atmosphere, ocean, wave and river routing)~~ to support many different modeling applications that might require detailed representation of the interactions among different ~~earth~~ Earth system processes (Fig. ??a-b). The ~~state-of-art~~ implementation

145　of the modeling system follows the hub-and-spoke architecture. The driver that is responsible for the orchestration of the overall modeling system resides in the middle and acts as a translator among model components ~~. In the design of the coupled modeling system, the coupling interfaces and driver are mainly developed by Istanbul Technical University (ITU) while MITgcm ocean model component is modified by the Italian National Agency for New Technologies, Energy and Sustainable Economic Development (ENEA) to work as a part of RegESM modeling system (?).~~

150　~~The following sections basically aim to give brief information about the individual model components, which are used to construct the modeling system, as well as the design details of the driver to combine independently developed and maintained model components in an efficient and standardized way~~ (atmosphere, ocean, wave, river routing, and co-processing). In this case, each model component introduces its NUOPC cap to plug into the modeling system. The modeling system is validated in different model domains such as Caspian Sea (?), Mediterranean Basin (??), and Black Sea Basin.

2.1　Atmosphere ~~Models~~ models (ATM)

The flexible design of RegESM modeling system allows ~~to choose~~ choosing a different atmospheric model component (ATM) in the configuration of the coupled model for a various type of application. Currently, two different atmospheric model ~~are~~ is compatible with RegESM modeling system: ~~)~~ (1) RegCM4 (?), which is developed by the Abdus Salam International Centre for Theoretical Physics (ICTP) and ~~)~~ (2) the Advanced Research Weather Research and Forecasting (WRF) Model (ARW;

160　?), which is developed and sourced from National Center for Atmospheric Research (NCAR). ~~The details of the individual atmospheric model can be found in following section but RegCM4~~ In this study, RegCM 4.6 is selected as an atmospheric

5

model component ~~in this study because it is well tested with verified in different model domains such as Caspian Sea (**?**), Mediterranean Basin (**??**) and Black Sea. Additionally,~~ because the current implementation of ~~RegESM WRF~~ WRF coupling interface is still experimental and does not support ~~online nesting yetbut this will be included in~~ coupling with co-processing component yet, but the next version of the modeling system (RegESM ~~2.0) .~~ 1.2) will be able to couple WRF atmospheric model with co-processing component. The NUOPC cap of atmospheric model components defines state variables (i.e., sea surface temperature, surface wind components), rotates the winds relative to Earth, apply unit conversions and perform vertical interpolation to interact with the newly introduced co-processing component.

2.1.1 RegCM

The dynamical core of the RegCM4 is based on the primitive equation, hydrostatic version of the National Centre for Atmospheric Research (NCAR) and Pennsylvania State University mesoscale model MM5 (**?**). The ~~model~~ latest version of the model (RegCM 4.6) also supports non-hydrostatic dynamical core to support applications with high spatial resolutions (< 10 km). The model includes two different land surface models: ~~)~~ (1) Biosphere-Atmosphere Transfer Scheme (BATS; **?**) and ~~)~~ (2) Community Land Model (CLM), version 4.5 (**?**). The model also includes ~~specialized~~ specific physical parameterizations to define air-sea interaction over the sea and lake (one-dimensional lake model; **?**). The Zeng Ocean Air-Sea Parameterization (**?**) is ~~basically~~ extended to introduce the atmosphere model as a component of the coupled modeling system. In this way, the atmospheric model ~~is able to~~ can exchange both two and three-dimensional fields ~~(i.e. sea surface temperature, surface roughness, runoff, surface wind speed, cloud liquid water content)~~ with other model components such as an ocean, wave and river routing components that are active in an area inside of the atmospheric model domain as well as in-situ visualization component. ~~In this design, the driver and coupling interface are developed by ITU to adapt easily to the newer version of RegCM model without major code change in the model component and the driver.In this study, specifically RegCM 4.6 is used to perform in-situ visualization integrated simulations.~~

2.1.2 WRF

~~The WRF~~

The WRF model consists of fully compressible non-hydrostatic equations, and the prognostic variables include the three-dimensional wind, perturbation quantities of pressure, potential temperature, geo-potential, surface pressure, turbulent kinetic energy and scalars (i.e., water vapor mixing ratio, cloud water). The model is suitable for a broad range of applications and has a variety of options to choose parameterization schemes for ~~convection,~~ the planetary boundary layer (PBL), convection, explicit moisture, radiation, and soil processes to ~~make available investigation of different earth~~ support analysis of different Earth system processes. The ~~model consists of fully compressible non-hydrostatic equations and the prognostic variables include the three-dimensional wind, perturbation quantities of pressure, potential temperature, geo-potential, surface pressure, turbulent kinetic energy and scalars (water vapor mixing ratio, cloud water etc)~~ PBL scheme of the model has a significant impact on exchanging moisture, momentum, and energy between air and sea (and land) due to the used alternative surface layer options (i.e., drag coefficients) in the model configuration. A few modifications are done in WRF (version 3.8.1) model itself to couple

195 it with RegESM modeling system. These modifications include rearranging of WRF ~~time related~~ time-related subroutines, which are inherited from the older version of ESMF Time Manager API (Application Programming Interface) that was available in 2009, to compile model with the newer version of ESMF library (version 7.1.0) together with the older version that requires mapping of time manager data types between old and new versions. ~~In addition, the model is modified to exchange data with driver component to support different coupled model applications such as atmosphere-ocean interaction.~~

200 ## 2.2 Ocean ~~Models~~models (OCN)

The current version of the coupled modeling system supports two different ocean model components (OCN): ~~)~~(1) Regional Ocean Modeling System (ROMS revision 809; **??**), which is developed and distributed by Rutgers University and ~~)~~(2) MIT General Circulation Model (MITgcm version c63s; **??**). In this case, ROMS and MITgcm models are selected due to their large user communities and different vertical grid representations. Although the selection of ocean model components ~~depend~~
205 depends on user experience and application, often the choice of vertical grid system has a determining role in some specific applications. For example, the ROMS ocean model uses terrain following (namely s-coordinates) vertical grid system that allows a better representation of the coastal processes but MITgcm uses z levels ~~and~~ generally used for ~~the applications that involves~~ applications that involve open oceans and seas. Similar to the atmospheric model component, both ocean models ~~(ROMS and MITgcm)~~ are slightly modified to allow data exchange with the other model components. In the current version
210 of the coupled modeling system, there is no interaction between wave and ocean model components, which could be crucial for some applications (i.e., surface ocean circulation and wave interaction~~etc.~~) that need to consider the two-way interaction between waves and ocean currents. ~~Fortunately, there is a plan to implement ocean-wave interface in the future release of the~~ The exchange fields defined in the coupled modeling system ~~(RegESM 2.0). In general, the exchange fields~~ between ocean and atmosphere ~~closely~~ strictly depend on the application and the studied problem. In some studies, the ocean model
215 requires heat, freshwater and momentum fluxes to be provided by the atmospheric component, while in others, the ocean component retrieves ~~atmospheric data~~ surface atmospheric conditions (i.e., surface temperature, humidity, surface pressure, wind components, precipitation) to calculate fluxes internally, by using bulk formulas (**?**). In the current design of the coupled modeling system, the driver allows ~~to select~~ selecting the desired exchange fields from the predefined list of the available fields. The exchange field list is a simple database with all fields that can be exported or imported by the component. In this way,
220 the coupled modeling system can be adapted to different applications without any code customizations in both the driver and individual model components.

2.2.1 ROMS

The ROMS is a three-dimensional, free-surface, terrain-following numerical ocean model that solves the Reynolds-averaged Navier-Stokes equations using the hydrostatic and Boussinesq assumptions. The governing equations are in flux form, and
225 the model uses Cartesian horizontal coordinates and sigma vertical coordinates with three different stretching functions. The model also supports second, third and fourth order horizontal and vertical advection schemes for momentum and tracers via its

preprocessor flags. ~~To allow coupling with RegESM modeling system, a set of preprocessor flags such as *REGCM_COUPLING* are included to the model.~~

2.2.2 MITgcm

The MIT general circulation model (MITgcm) is a generic and widely used ocean model that solves the Boussinesq form of Navier-Stokes equations for an incompressible fluid. It supports both hydrostatic and non-hydrostatic applications with a spatial finite-volume discretization on a curvilinear computational grid. The model has an implicit free surface in the surface and partial step topography formulation to define vertical depth layers. The ~~model~~ MITgcm model supports different advection schemes for momentum and tracers such as centered second order, third-order upwind and second-order flux limiters to support a variety of applications. The model used in the coupled modeling system ~~is~~ was slightly modified by ENEA to allow data exchange with other model components. The detailed information about the regional applications of the MITgcm ocean model ~~was initially~~ is described in the study of **?** using ~~PROTEUS modeling system , which is also developed by ENEA~~PROTHEUS modeling system specifically developed for the Mediterranean Sea.

2.3 Wave ~~Model~~model (WAV)

Surface waves play a crucial role in the dynamics of ~~the planetary boundary layer (PBL)~~ PBL in the atmosphere and the currents in the ocean. ~~To have~~Therefore, the wave component is included in the coupled modeling system to have a better representation of atmospheric PBL and ~~the~~surface conditions (i.e., surface roughness, friction velocity, wind speed)~~, the wave component is included to the coupled modeling system~~. In this case, the wave component is based on WAM Cycle-4 (4.5.3-MPI). The WAM is a third-generation model ~~, based on pure physics~~without any assumption on the spectral shape (**?**). It considers all the main processes that control the evolution of a wave field in deep water, namely the generation by wind, the nonlinear ~~wave-wave~~wave–wave interactions, and also white-capping. ~~To include WAM model (provided by ENEA) as a component of coupled modeling system, the model again is~~The model was initially developed by Helmholtz-Zentrum Geesthacht (GKSS, now HZG) in Germany. The original version of the WAM model was slightly modified to ~~exchange fields with other model components. In this case, RegCM4 atmospheric model component provides~~ retrieve surface atmospheric conditions (i.e., wind speed components or friction velocity and wind direction) from the RegCM4 atmospheric model and to send back calculated surface roughness. In the current version of the modeling system, wave component cannot be coupled with the WRF model due to the missing modifications in the WRF side. In the RegCM4, the received surface roughness is used to ~~wave model and receives surface roughness to~~ calculate air-sea transfer coefficients and fluxes over sea using Zeng ocean air-sea parameterization (**?**). In this ~~implementation~~design, it is also possible to define a threshold for maximum roughness length (the default value is 0.02 m) and friction velocity (the default value is 0.02 m) in the configuration file of ~~atmospheric model component to ensure~~ RegCM4 to ensure the stability of the overall modeling system. ~~The initial results to investigate~~ Initial investigation of the added value of atmosphere-wave coupling in the Mediterranean Sea can be found in **?**.

2.4 River ~~Routing Model~~routing model (RTM)

To simulate the lateral freshwater fluxes (river discharges) at the land surface and to provide river discharge to ocean model
component, the RegESM modeling system uses Hydrological Discharge (HD, version 1.0.2) model developed by Max Planck
Institute (**??**). The model is ~~basically~~ designed to run in a fixed global regular grid with 0.5° horizontal resolution ~~. In this case, model uses~~ using daily time series of surface runoff and drainage as input fields. In that case, the model uses the pre-computed
river channel network to simulate the horizontal transport of the runoff within model watersheds using different flow processes
such as overland flow, baseflow and riverflow. ~~In this case, the model integrated with driver uses a daily time step and requires daily time series of surface runoff and drainage as input fields.~~ The river routing model (RTM) plays ~~important~~ an essential role
in the freshwater budget of the ocean model by closing the water cycle between the atmosphere and ocean model components.
The original version of the model ~~is~~ was slightly modified to support interaction with the coupled model components. To
close water cycle between land and ocean, model retrieves surface and sub-surface runoff from ~~atmospheric component~~ the
atmospheric component (RegCM or WRF) and provides estimated river discharge to the selected ocean model component
(ROMS or MITgcm). In the current design of the driver, rivers can be represented in two different ways: ~~)~~(1) individual
point sources that are vertically distributed to model layers, and ~~)~~(2) imposed as freshwater surface boundary condition like
precipitation (P) or evaporation minus precipitation (E-P). ~~The model is also modified to write restart and output files directly in NetCDF (Network Common Data Form) format rather than using Max Planck Institute's SERVICE data type~~In this case,
the driver configuration file is used to select the river representation type (1 or 2) for each river individually. The first option is
preferred if river plumes need to be defined correctly by distributing river discharge vertically among the ocean model vertical
layers. The second option is used to distribute river discharge to the ocean surface when there is a need to apply river discharge
to a large areal extent close to the river mouth. In this case, a special algorithm implemented in NUOPC cap of ocean model
components (ROMS and MITgcm) is used to find affected ocean model grids based on the effective radius (in km) defined in
the configuration file of the driver.

2.5 The ~~Driver~~driver: RegESM

The RegESM (version 1.1) is completely redesigned and improved version of the previously used and validated coupled
atmosphere-ocean model (RegCM-ROMS) to study the regional climate of Caspian Sea and its catchment area (**?**). To simplify
the design and to create more generic, extensible and flexible modeling system that aims to support easy integration of multiple
model components and applications, the RegESM uses ~~driver based model coupling~~ a driver to implement the hub-and-spoke
approach. In this case, all the model components are combined using ESMF (version 7.1.0) framework to structure coupled
~~modelling~~ modeling system. The ESMF framework is selected because of its unique online re-gridding capability, which
allows the driver to ~~readily~~ perform different interpolation types (i.e., bilinear, conservative~~etc.~~) over the exchange fields (i.e.,
sea surface temperature, heat and momentum fluxes) and the ~~National Unified Operational Prediction Capability (NUOPC)~~
NUOPC layer. The NUOPC layer ~~basically simplifies component synchronization and run sequence by providing additional~~
~~programming interface between coupled model and ESMF framework. It~~is a software layer built on top of the ESMF. It refines

the capabilities of ESMF by providing a more precise definition of a component model and how components should interact and share data in a coupled system. The ESMF also provides the capability of transferring computational grids ~~among the model components~~ in the model component memory, which has critical importance in the integration of the modeling system with a co-processing environment (see also Sect. **??**). The RegESM ~~modelling~~ modeling system also uses ESMF and NUOPC

295   layer to support various configuration of component interactions such as defining multiple coupling time steps ~~(fast and slow time steps; Fig. **??**)~~ among the model components ~~and explicit and semi-implicit type of model interaction (Fig. **??**).~~ An example configuration of the four-component (ATM, OCN, RTM, and WAV) coupled modeling system can be seen in Fig. **??**. In this case, the RTM component runs in a daily time step (slow) and interacts with ATM and OCN components, but ATM and OCN components can interact ~~each other~~ more frequently ~~(i.e. 3-hours, fast)~~ fast) such as every three hours.

300   The interaction (also called as run sequences) among the model components and driver are facilitated by the connector components provided by NUOPC layer. Connector components are mainly used to create a link between individual model components and driver. ~~The NUOPC layer also specifies an interface to define the model and connector~~ In this case, the number of active components and their ~~interactions (also called as run sequences). To support different applications,~~ interaction determines the number of connector component created in the modeling system. The interaction between model components

305   can be in two way: (1) bi-directional such as atmosphere and ocean coupled modeling system or (2) unidirectional such as atmosphere and co-processing modeling system. In the uni-directional case, the co-processing component does not interact with the atmosphere model and only process retrieved information; thus there is one connector component.

     The RegESM modeling system can ~~be configured with two different type~~ use two different types of time-integration ~~scheme (coupling type) between~~ coupling scheme between the atmosphere and ocean components: ~~) explicit and)~~ (1) explicit and (2)

310   semi-implicit (or leap-frog) (Fig. **??**). In ~~explicit type of~~ the explicit type coupling, two connector components (ATM-OCN and OCN-ATM direction) are executed concurrently at every coupling time step and model components start and stop at the same model time (Fig. **????**a). ~~Unlike, explicit coupling , a leap-frog style interaction is supported between atmosphere and ocean components using semi-implicit type of model coupling.In this case, at every coupling time step,~~ In the semi-implict coupling type (Fig. **??**b), the ocean model receives surface boundary conditions from the atmospheric model at one coupling

315   time step ahead of the current ocean model time ~~(Fig. **??**b). The main advantage of using of the~~. The semi-implicit coupling ~~is that it ensures the stability of the overall modeling system~~ aimed at lowering the overall computational cost of a simulation by increasing stability for longer coupling time steps.

     As described earlier, the execution of the model components is ~~mainly~~ controlled by the driver. ~~In this case, both~~ Both sequential and concurrent execution of the model components ~~are~~ is allowed in the current version of the modeling system.

320   If the model components and the driver are configured to run in sequence on the same set of PETs (Persistent Execution Threads), then the ~~modelling~~ modeling system executes in a sequential mode. This ~~is~~ mode is a much more efficient way to run the modeling system in case of limited computing resources. In the concurrent type of execution, the model components run in mutually exclusive sets of PETs ~~but the driver uses~~, but the NUOPC connector component uses a union of available computational resources (or PETs) of interacted model components. ~~In~~ By this way, the ~~modelling~~ modeling system can

325   support a variety of computing systems ranging from local servers to large computing systems that could include high-speed

performance networks, accelerators (i.e., Graphics Processing Unit or GPU) and parallel I/O capabilities. The main drawback of concurrent execution approach is to assign correct amount of computing resource to individual model components, which is not ~~a~~ an easy task and might require ~~a~~ an extensive performance benchmark of a specific configuration of the model components, to ~~achive~~ achieve best available computational performance. In this case, a load-balancing analysis of individual components

330 and driver play a critical role in the performance of the overall modeling system. For example, the LUCIA (Load-balancing Utility and Coupling Implementation Appraisal) tool can be used to collect all required information such as waiting time and calculation time of each system components for a load-balancing analysis in the OASIS3-MCT based coupled system.

In general, the design and development of the coupled modeling systems involve a set of technical difficulties that arise due to the usage of the different computational grids in the model components. One of the most common examples is the mismatch

335 between the land-sea masks of the model components (i.e., atmosphere and ocean models). In this case, the unaligned land-sea masks might produce artificial ~~and/~~or unrealistic surface heat and momentum fluxes around the coastlines, narrow bays, straits and seas. The simplest solution ~~of this issue~~ is to modify the land-sea masks of the individual model components manually to align them ~~. However, the main disadvantage of this solution is the required time and the difficulty to fix the land-sea masks of the different model components~~ however, this requires time and is complex (especially when the horizontal grid

340 resolution is high). ~~In addition, same procedure must be repeated every time when~~ Besides, the procedure needs to be repeated each time the model domain (i.e., shift or change in the model domain) or horizontal grid resolution is changed. ~~As a result, this approach is considered as application specific and very time-consuming. Unlike manual editing of the land-sea masks, customized interpolation techniques that also include extrapolation support might help to create more generic and automatized solutions.~~

345 The RegESM modeling system uses ~~extrapolation approach~~ customized interpolation technique that also includes extrapolation to overcome the mismatched land-sea mask problem for the interaction between atmosphere, ocean and wave components. ~~To perform extrapolation, the driver uses a specialized algorithm (Fig. ??) to find the mapped and unmapped ocean grid points in the interpolation stage for every coupling direction (i.e. atmosphere-ocean, ocean-atmosphere, atmosphere-wave). According to the algorithm, the mapped grid points have same~~ This approach helps to create more generic and automatized solutions for

350 the remapping of the exchange fields among the model components and enhance the flexibility of the modeling system to adapt to different regional modeling applications. There are three main stages in the customized interpolation technique: (1) finding destination grid points that the land-sea ~~mask type in both model components (i. e. both are sea or land) . On the other hand, the land-sea~~ mask type does not match completely ~~in the case of~~ with the source grid (unmapped grid points~~and they cannot be filled with the standard interpolation method such as bilinear interpolation. In this case, the two-step interpolation is performed~~

355 ~~to~~; Fig. ??), (2) perform bilinear interpolation to transfer the exchange field from source to destination grid, and (3) perform extrapolation in destination grid to fill unmapped grid points that are found in first step.

~~In the first step, exchange field is interpolated~~ To find the unmapped grid points, the algorithm first interpolates the field from source to destination grid (~~i.e. from atmosphere to ocean~~)~~using grid points just over the sea~~ just over the sea) using a nearest-neighbor type interpolation (from Field_A to Field_B). Similarly, the same operation is repeated by using a bilinear

360 type interpolation (from Field_A to Field_C). Then, the results of both interpolation (Field_B and Field_C) is compared to

11

identify unmapped grid points for the bilinear interpolation (Fig. **??**). ~~Then, result field is used to fill unmapped grid points using~~

The field can then be interpolated from the source to the destination grid using a two-step interpolation approach. In the first step, the field is interpolated from source to destination grid using a bilinear interpolation. Then, nearest-neighbor type interpolation ~~that is performed in~~ is used on the destination grid ~~(from mapped grid points to~~ to fill unmapped grid points~~)~~. One of the main ~~drawback~~ drawbacks of this method is that the result field might include unrealistic values and ~~/or~~ sharp gradients in the areas of complex land-sea mask structure (i.e.~~channels and/or straits)but this might~~, channels, straits). The artifacts around the coastlines can be fixed by applying a light smoothing after interpolation or using more sophisticated extrapolation techniques such as the sea-over-land approach (**??**), which are not included in the current version of the modeling system. ~~In addition~~Also, the usage of the mosaic grid along with second-order conservative interpolation method, which gives smoother results when the ratio between horizontal grid resolutions of the source and destination grids are high, ~~might also help to solve the~~ can overcome unaligned land-sea mask problem. ~~Fortunately, the next public~~The next major release of ESMF ~~framework will supports higher-order conservative interpolation as well as extrapolation~~ library (8.0) will include the creep fill strategy (**?**) to fill unmapped grid points.

3   Integration of a co-processing component in RegESM modeling system~~with co-processing component~~

The newly designed modeling framework is ~~defined as~~ a combination of the ParaView co-processing plugin ~~—~~ which is called Catalyst (**?**) ~~—~~ and ESMF library that is specially designed ~~to couple different earth system models~~ for coupling different ESMs to create more complex regional and global ~~modelling systems.~~ modeling systems. In conventional co-processing enabled simulation systems (single physical model component such as atmosphere along with co-processing support), the Catalyst is used to integrate ParaView visualization pipeline ~~(defined as a Python script)~~ with the simulation code to support in-situ visualization through the use of ~~application specific~~ application-specific custom adaptor code ~~.~~ (**????**). A visualization pipeline is defined as a data flow network in which computation is described as a collection of executable modules that are connected in a directed graph representing how data moves between modules (**?**). There are three types of modules in a visualization pipeline: sources (file readers and synthetic data generators), filters (transforms data), and sinks (file writers and rendering module that provide images to a user interface). The adaptor code ~~basically~~ acts as a wrapper layer and ~~transform~~ transforms information coming from simulation code to the ~~ParaView~~ co-processing component in a compatible format that is defined using ParaView/Catalyst and VTK (Visualization Toolkit) ~~API~~APIs. Moreover, the adaptor code is responsible for defining the underlying computational grid and associating them with the multi-dimensional fields. After defining computational grids and fields, ~~the~~ ParaView processes the received data to perform co-processing to create desired products such as rendered visualizations, added value information (i.e., spatial and temporal averages, derived fields) ~~and/or writing raw model~~ as well as writing raw data to the ~~storage system~~ disk storage (Fig. **??**a).

~~On the other hand, the~~ The implemented novel approach aims to create a more generic and standardized co-processing environment ~~specifically designed for earth~~ designed explicitly for Earth system science (Fig. **??**b). ~~By~~ With this approach,

existing ESMs, which are coupled with ESMF library using NUOPC interface, may benefit from the use of an integrated modeling framework to analyze the data flowing from the multi-component and multi-scale modeling system without extensive code development and restructuring. In this design, the adaptor code interacts with the driver through the use of NUOPC cap and provides an abstraction layer for the co-processing component. As discussed previously, the ESMF framework uses a standardized interface (initialization, run and finalize routines) to plug new model components into existing modeling system such as RegESM in an efficient and optimized way. To that end, the new approach will benefit from the standardization of common tasks in the model components to integrate co-processing component with the existing modeling system. In this case, all information (grids, fields, and metadata) required by ParaView/Catalyst is received from the driver, and direct interaction between other model components and the co-processing component is not allowed (Fig. **??**b). The implementation logic of the adaptor code is very similar to the conventional co-processing approach (Fig. **??**a). However, in this case, it uses the standardized interface of the ESMF framework and NUOPC layer to define the computational grid and associated two and three-dimensional fields of model components. The adaptor layer maps the field (i.e., *ESMF_Field*) and grid (i.e., *ESMF_Grid*) objects to their VTK equivalents through the use of VTK and co-processing APIs, which are provided by ParaView and co-processing plugin (Catalyst). Along with the usage of the new approach, the interoperability between simulation code and in-situ visualization system are enhanced and standardized. The new design provides an easy-to-develop, extensible and flexible modeling environment for Earth system science.

The development of the adaptor component plays an essential role in the overall design and performance of the integrated modeling environment. The adaptor code mainly includes a set of functions for the initialization (defining computational grids and associated input ports), run and finalize the co-processing environment. Similarly, the ESMF framework also uses the same approach to plug new model components into the modeling system as ESMF components. In ESMF framework, the simulation code is separated into three essential components (initialization, run and finalize) and calling interfaces are triggered by the driver to control the simulation codes (i.e., atmosphere and ocean models). In this case, the initialization phase includes definition and initialization of the exchange variables, reading input (initial and boundary conditions) and configuration files and defining the underlying computational grid (step 1 in Fig. **??**). The run phase includes a time stepping loop to run the model component in a defined period and continues until simulation ends (step 4 in Fig. **??**). The time interval to exchange data between model and co-processing component can be defined using coupling time step just like the interaction among other model components. According to the ESMF convention, the model and co-processing components are defined as a gridded component while the driver is a coupler component. In each coupling loop, the coupler component prepares exchange fields according to the interaction among components by applying re-gridding (except coupling with co-processing component), performing a unit conversion and common operations over the fields (i.e., rotation of wind field). In the new version of the RegESM modeling system (1.1), the driver is extended to redistribute two and three-dimensional fields from physical model components to allow interaction with the co-processing component. In the initialization phase, the numerical grid of ESMF components is transformed into their

13

VTK equivalents using adaptor code (step 3 in Fig. **??**). In this case, *ESMF_Grid* object is used to create *vtkStructuredGrid* along with their modified parallel two-dimensional decomposition configuration, which is supported by ESMF/NUOPC grid transfer capability (Fig. **??**). According to the design, each model component transfers their numerical grid representation to co-processing component at the beginning of the simulation (step 1 in Fig. **??**) while assigning independent two-dimensional decomposition ratio to the retrieved grid definitions. The example configuration in Figure **??** demonstrates mapping of 2x3 decomposition ratio (in x and y-direction) of ATM component to 2x2 in the co-processing component. Similarly, the ocean model transfers its numerical grid with 4x4 decomposition ratio to co-processing component with 2x2 (Fig. **??**). In this case, ATM and OCN model components do not need to have the same geographical domain. The only limitation is that the domain of ATM model component must cover the entire OCN model domain for an ATM-OCN coupled system to provide the surface boundary condition for OCN component. The main advantage of the generic implementation of the driver component is to assign different computational resources to the components. The computational resource with accelerator support (GPU) can be independently used by co-processing component to do rendering (i.e., iso-surface extraction, volume rendering, and texture mapping) and processing the high volume of data in an efficient and optimized way. The initialization phase is also responsible for defining exchange fields that will be transferred among the model components and maps *ESMF_Field* representations as *vtkMultiPieceDataSet* objects in co-processing component (step 2-3 in Fig. **??**). Due to the modified two-dimensional domain decomposition structure of the numerical grids of the simulation codes, the adaptor code also modifies the definition of ghost regions a small subset of the global domain that is used to perform numerical operations around edges of the decomposition elements. In this case, the ghost regions (or halo regions in ESMF convention) are updated by using specialized calls, and after that, the simulation data are passed (as *vtkMultiPieceDataSet*) to the co-processing component. During the simulation, the co-processing component of the modeling system also synchronizes with the simulation code and retrieves updated data (step 5 in Fig. **??**) to process and analyze the results (step 6 in Fig. **??**). The interaction between driver and the adaptor continues until the simulation ends (step 4, 5 and 6 in Fig. **??**) and the driver continues to redistribute the exchange fields using *ESMF_FieldRedist* calls. The NUOPC cap of model components also supports vertical interpolation of the three-dimensional exchange fields to height (from terrain-following coordinates of RegCM atmosphere model) or depth coordinate (from s-coordinates of ROMS ocean model) before passing information to the co-processing component. In this design, the vertical interpolation is introduced to have a consistency in the vertical scales and units of the data coming from the atmosphere and ocean components. Then, finalizing routines of the model and co-processing components are called to stop the model simulations and the data analysis pipeline that destroy the defined data structure/s and free the memory (step 7-8 in Fig. **??**).

14

4 Use case and performance benchmark

To test the capability of the newly designed integrated modeling system described briefly in the previous section, the three component (atmosphere, ocean, and co-processing) configuration of RegESM 1.1 modeling system is implemented to analyze category 5 Hurricane Katrina. Hurricane Katrina was the costliest natural disaster and has been named one of the five deadliest hurricanes in the history of the United States, and the storm is currently ranked as the third most intense United States land-falling tropical cyclone. After establishing itself in the southern Florida coast as a weak category 1 storm near 22:30 UTC 25 August 2005, it strengthened to a category 5 storm by 12:00 UTC 28 August 2005 as the storm entered the central Gulf of Mexico (GoM). The model simulations are performed over a 3-day period i.e. 27-30 Aug. 2005, which is the most intense period of the cyclone, to observe the evolution of the Hurricane Katrina and understand the importance of air-sea interaction regarding its development and predictability. The next section mainly details the three-components configuration of the modeling system as well as the computing environment, preliminary benchmark results performed with limited computing resource (without GPU support), and analysis of the evolution of Hurricane Katrina.

4.1 Working environment

The model simulations and performance benchmarks are done on a cluster (SARIYER) provided by the National Center for High-Performance Computing (UHeM) in Istanbul, Turkey. The CentOS 7.2 operating system installed in compute nodes are configured with a two Intel Xeon CPU E5-2680 v4 (2.40GHz) processor (total 28 cores) and 128 GB RAM. In addition to the compute nodes, the cluster is connected to a high-performance parallel disk system (Lustre) with 349 TB storage capacity. The performance network, which is based on Infiniband FDR (56 Gbps) is designed to give the highest performance for the communication among the servers and the disk system. Due to the lack of GPU accelerators in the entire system, the in-situ visualization integrated performance benchmarks are done with the support of software rendering provided by Mesa library. Mesa is an open-source OpenGL implementation that supports a wide range of graphics hardware each with its back-end called a renderer. Mesa also provides several software-based renderers for use on systems without graphics hardware. In this case, ParaView is installed with Mesa support to render information without using hardware-based accelerators.

4.2 Domain and model configurations

The Regional Earth System Model (RegESM 1.1) is configured to couple atmosphere (ATM; RegCM) and ocean (OCN; ROMS) models with newly introduced the co-processing component (ParaView/Catalyst version 5.4.1) to analyze the evolution of Hurricane Katrina and to assess the overall performance of the modeling system. In this case, two atmospheric model domains were designed for RegCM simulations using offline nesting approach, as shown in Fig. **??**. The outer atmospheric model domain (low-resolution;

, LR) with a resolution of 27-km is centered at 77.5°W, 25.0°N and covers almost entire the United States, the western part of Atlantic Ocean and north-eastern part of Pacific Ocean for better representation of the ~~large scale~~ large-scale atmospheric circulation systems~~and~~. The outer domain is enlarged as much as possible to minimize the effect of the lateral boundaries of the atmospheric model in the simulation results of the inner model domain~~(high-resolution; HR)~~. The horizontal grid spacing of ~~second domain~~ inner domain (high-resolution, HR) is 3-km and covers the entire GoM and the western Atlantic Ocean to provide ~~high resolution~~ high-resolution atmospheric forcing for coupled atmosphere-ocean model simulations and perform ~~cloud resolving~~ cloud-resolving simulations. Unlike the outer domain, the model for the inner domain is configured to use the non-hydrostatic dynamical core (available in RegCM 4.6) to allow better representation local scale vertical acceleration and ~~important~~ essential pressure features.

The lateral boundary condition for the outer domain is obtained from European Centre for Medium-Range Weather Forecasts (ECMWF) latest global atmospheric reanalysis (ERA-Interim project; **?**), which is available at 6-h intervals at a resolution of 0.75°x0.75° in the horizontal and 37 pressure levels in the vertical. On the other hand, the lateral boundary condition of the ~~HR domain , which is used in the fully coupled model simulations, is~~ inner domain is specified by the results of the ~~LR domain. Concerning cumulus convection,~~ outer model domain. Massachusetts Institute of Technology-Emanuel convective parameterization scheme (MIT-EMAN; **??**) ~~is used in outer model simulations. Along with selected cumulus convection parameterization,~~ for the cumulus representation along with sub-grid explicit moisture (SUBEX; **?**) scheme ~~is used to represent~~ for large-scale ~~precipitation for LR~~ precipitations are used for low-resolution outer domain.

As it ~~also shown~~ can be seen in Fig. **??**, the ROMS ocean model is configured to cover entire the GoM to allow better tracking of the Hurricane Katrina. In this case, the used ocean model configuration is very similar to the configuration used by Physical Oceanography Numerical Group (PONG), Texas A&M University (TAMU), in which the original model configuration can be ~~accessible~~ accessed from their THREDDS ~~server.~~ (Thematic Real-time Environmental Distributed Data Services) data server (TDS). THREDDS is a service that aims to provide access to an extensive collection of real-time and archived datasets, and TDS is a web server that provides metadata and data access for scientific datasets, using a variety of remote data access protocols. The ocean model has a spatial resolution of 1/36°, which corresponds to a non-uniform resolution of around 3 km (655 x 489 grid points) with highest grid resolution in the northern part of the domain. The model has 60 vertical sigma ~~layer~~ layers ($\theta_s = 10.0, \theta_b = 2.0$) to provide detailed representation of the main circulation ~~pattern~~ patterns of the region and vertical tracer gradients. The bottom topography data of the GoM is constructed using the ETOPO1 dataset (**?**), and minimum depth (h_c) is set to 400 m. The bathymetry ~~data are~~ is also modified so that the ratio of ~~depths~~ depth of any two adjacent ~~grids~~ columns does not exceed 0.25 to enhance the stability of the model and ensure hydrostatic consistency ~~creation~~ that prevents pressure gradient error. The Mellor-Yamada level 2.5 turbulent closure (MY; **?**) is used for vertical mixing, while rotated tensors of the harmonic formulation are used for horizontal mixing. The lateral boundary conditions for ROMS ocean model are provided by Naval Oceanographic Office Global Navy Coastal Ocean Model (NCOM) during 27-30 August 2005.

The model coupling time step between atmosphere and ocean model component is set to 1 hour but 6 minutes coupling time step is used to provide one-way interaction with co-processing component to study Hurricane Katrina in a very high temporal resolution. In the coupled model simulations~~that use same configuration of~~, the ocean model ~~but different resolution of the~~

16

atmospheric model component, the SST data provided provides SST data to the atmospheric model in the region where their numerical grids overlap. In the rest of the domain, the atmospheric model uses SST data provided by ERA-Interim dataset (prescribed SST). The results of the performance benchmark also include additional tests with smaller coupling time step such as 3 minutes for the interaction with the co-processing component. In this case, the model simulations for the analysis of the Hurricane Katrina runs over three days, but only one day of simulation length is chosen in the performance benchmarks to reduce amount of used computing resourcesthe compute time.

4.3 Performance Benchmarkbenchmark

To A set of simulations are performed with different model configurations to assess the overall performance of the coupled modeling system by focusing on the overhead of the newly introduced co-processing component , a set of simulations are performed with different model configurations (Table **??**). The performance benchmarks include analysis of the extra overhead provided by the co-processing component, coupling interval between physical models and co-processing component under different rendering load such as various visualization pipelines (Table **??**). To scale up to large number of processors, two different atmospheric model configurations are defined (a low-resolution, LR and high-resolution HR) In this case, same model domains that are described in the previous section (Sect. **??**) are also used in the benchmark simulations. The LR atmospheric model domain includes around 900.000 grid points in atmospheric model while while the HR domain contains 25 million grid points to test scaling up to a large number of processors. In both casecases, the ocean model configuration is same the same, and it has around 19 million grid points. Beside the change of the dynamical core of Besides the use of a non-hydrostatic dynamical core in the atmospheric model component in HR case (non-hydrostatic)the HR case, the rest of the model configurations are configuration is preserved. To isolate the overhead of the driver from the overhead of the co-processing component, first individual model components (ATM and OCN) are run in standalone mode and then, the best scaled model configurations in terms of best-scaled model configurations regarding two-dimensional decomposition configuration are used in the coupled model simulations(CPL and COP; CPL (two component case: atmosphere-ocean) and COP (three-component case: atmosphere, ocean and co-processing component). Due to the current limitation in the integration of the co-processing component, the coupled model only supports sequential type execution (see section 2.5 for more information) when when the co-processing component is activated, but this limitation will be removed in the future version of the modeling system (RegESM 2.0). As mentioned in the previous section, the length of the simulations are is kept relatively short (1 day) in the benchmark analysis to perform many simulations with different model configurations (coupling interval etc.)i.e., coupling interval, visualization pipelines and domain decomposition parameters).

The benchmark resultsof standalone model In the benchmark results, the slightly modified version of the speed-up is used because the best possible sequential implementation of the utilized numerical model (standalone and coupled) does not exist for the used demonstration application and model configurations. In this case, the speed-up is defined as the ratio of the parallel

execution time for the minimum number of processors required to run the simulation ($T_p(N_{min})$; based on 140 cores in this study) to the parallel execution time ($T_p(N)$; see Eq. **??**).

$$S(N) = \frac{T_p(N_{min})}{T_p(N)} \tag{1}$$

The measured wall clock time and the calculated speed-up of standalone model components (ATM and OCN) can be seen in Fig. **??**. In this case, two different atmospheric model configurations are considered to see the effect of the domain size and non-hydrostatic dynamical core in the benchmark results (LR and HR; Fig. **??**). The results show that the model scales pretty well and it is clear that the HR case shows better scaling results than LR configuration of the atmospheric component (ATM) as expected. It is also shown that around 588 processors, which is the highest available compute resource, the communication among the processors dominate the benchmark results ~~and even HR case does not gain further performance~~ of LR case, but it is not evident in HR case that scales very well without any performance problem (Fig. **??**a). Similar to the atmospheric model component, the ocean model (OCN) is also tested to find the best two-dimensional domain decomposition configuration (tiles in x and ~~y direction~~ y-direction). As it can be seen from the Fig. **??**b, the selection of the tile configuration affects the overall performance of the ocean model. In general, model scales better if ~~tile in x direction~~ the tile in the x-direction is bigger than ~~tile in y direction~~ the tile in the y-direction, but this is more evident in the small number of processors. ~~This~~ The tile effect is mainly due to the memory management of Fortran programming language (column-major order) as well as the total number of active grid points (not masked as land) placed in each tile. ~~On the other hand, the~~ The tile options must be selected carefully while considering the dimension of the model domain in each direction. In some tile configuration, it is not possible to run the model due to the ~~used~~ underlying numerical solver and the required minimum ghost points. To summarize, the ocean model scales well until 588 cores with the best tile configurations ~~shown~~ indicated in Fig. **??**b.

~~Using the~~ The performance of the two-component modeling system (CPL) can be investigated using the benchmark results of the standalone atmosphere and ocean models~~, the performance of the two-component modeling system (CPL) can be investigated.~~ Similar to the benchmark results of the standalone model components, the measured wall clock time and the calculated speed-up of the coupled model simulations are also shown in Fig. **??**. In this case, the best two-dimensional decomposition parameters of the standalone ocean model simulations are used in the coupled model simulations (Fig. **??**b). The overhead is calculated by comparing the CPL wall clock-time to the sum of the standalone OCN and ATM wall clock time as they run sequentially. The comparison of the standalone and coupled model simulations show that the driver component introduces additional 5-10% (average is 5% for LR and 6% for HR cases) overhead in the total execution time, which ~~slight~~ slightly increases along with the used total number of processors~~, which is acceptable when increased number of MPI communication between the components are considered (Fig. ?? and ??a-b).~~ The extra overhead is mainly due to the interpolation (sparse matrix multiply performed by ESMF) and extrapolation along the coastlines to match land-sea masks of the atmosphere and ocean models and fill the unmapped grid points to exchange data (Fig. **??**) ~~.~~ and slightly increases along with increased number of cores as well as number of MPI communication between the model components (Fig. **??** and **??**a).

To further investigate the overhead introduced by the newly designed co-processing component, the three-component modeling system (COP) is tested with three different visualization pipelines (P1, P2, and P3; Table **??**) using two different atmospheric model configurations (LR and HR) and coupling interval (3 and 6 minutes with co-processing). In this case, the measured total execution time during the COP benchmark results also includes vertical interpolation (performed in ESMF cap of the model components) to map data from sigma coordinates to height (or depth) coordinates for both physical model components (ATM and OCN).

As shown in Fig. **??**a-bb-d, the co-processing components require 10-40% extra execution time for both LR and HR cases depending on used visualization pipeline when it is compared with CPL simulations. The results also reveal that the fastest visualization pipeline is P3 and the slowest one is P1 for the HR case (Fig. **??**b)- and d). In this case, the components are all run sequentially, and the performance of the co-processing component becomes a bottleneck for the rest of the modeling system especially for the computing environment without GPU support like the system used in the benchmark simulations. It is evident that if the co-processing were run concurrently in a dedicated computing resource, the overall performance of the modeling system would be improved because of the simultaneous execution of the physical models and co-processing components. Table **??** also includes the execution time of the single visualization pipeline (measured by using *MPI_Wtime* call) isolated from the rest of the tasks. In this case, each rendering task gets 2-4 seconds for P1 and P2 cases and 7-15 seconds for the P3 case in LR atmospheric model configuration. For HR case, P1 and P2 takes take around 17-80 secondsand , and the P3 case is rendered in around 8-10 seconds. These results show that the time spent in the co-processing component (basically sending data to ParaView, Catalyst/Catalyst, and rendering to create the output) fluctuates too much and do not show predictable that this component does not present a predictable and stable behavior. This It might be due to the special particular configuration of the ParaView, which is configured to use software-based rendering to process data in CPUs and load in the used high-performance computing system (UHeM) even if the benchmark tests are repeated multiple times.

In addition to the testing modelling modeling system with various data processing load, a benchmark with increased coupling time step is also performed (see P23M in Fig. **??**bc). In this case, the coupling time step between physical model components and the co-processing component is increased decreased (from 6 minutes to 3 minutes) to produce output in doubled frame rate, but coupling interval between physical model components (ATM and OCN) are kept same (1 hourshour). The benchmark results show that increased coupling time step also rises overhead due to the co-processing from 45% to 60% for HR case and pipeline P2 when it is compared with the results of two-component simulations (CPL; Fig. **??**ba). It is also shown that the execution time of co-processing enabled coupled simulations increase but the difference between P2 and P23M cases are reduced from 66% to 37% when number of processor the number of processors increased from 140 to 588.

In addition to the analysis of timing profiles of modeling system under different rendering load, the amount of data exchanged and used in the in-situ visualization case can be compared with the amount of data that would be required for offline visualization at the same temporal frequency to reveal the added value of the newly introduced co-processing component. For this purpose, the amount of data exchanged with co-processing component is given in Table **??** for three different visualization pipelines (P1, P2, and P3). In co-processing mode, the data retrieved from model component memory (single time step) by the driver is passed to the ParaView/Catalyst for rendering. In addition to processing data concurrently with the simulation on co-processing

component, the offline visualization (post-processing) consists of computations done after the model is run and requires to store numerical results in a disk environment. For example, 3-days long simulation with 6-minutes coupling interval produces around 160 GB data (720 time-step) just for a single variable from high-resolution atmosphere component (P1 visualization pipeline) in case of using offline visualization. With co-processing, the same analysis can be done by applying the same visualization pipeline (P1), which requires to process only 224 MB data stored in the memory, in each coupling interval. Moreover, storing results of three-day long, high-resolution simulation of RegCM atmosphere model (in netCDF format) for offline visualization requires around 1.5 TB data in case of using 6-minutes interval in the default configuration (7 x 3d fields and 28 x 2d fields). It is evident that the usage of co-processing component reduces the amount of data stored in the disk and allows more efficient data analysis pipeline.

Besides the minor fluctuations in the benchmark results, the ~~modelling~~ modeling system with co-processing component scales pretty well to the higher number of processors (or cores) without any ~~major~~ significant performance pitfalls in the current configuration. On the other hand, the usage of accelerator enabled ParaView configuration (i.e., using NVIDIA EGL library) and ParaView plugins with accelerator support such as NVIDIA IndeX volume rendering plugin and new VTK-m filters to process data on GPU will ~~definitely~~ improve the benchmark result. The NVIDIA IndeX for ParaView Plugin ~~basically~~ enables large-scale and high-quality volume data visualization capabilities of the NVIDIA IndeX library inside the ParaView and might help to reduce time to process high-resolution spatial data (HR case). ~~This will be investigated future when~~ In addition to NVIDIA IndeX plugin~~supports in-situ visualization under ParaView. In addition, the~~, VTK-m is a toolkit of scientific visualization algorithms for emerging processor architectures such as GPUs (**?**). The model configurations used in the ~~benchmark~~ simulations also write simulation results to the disk in netCDF format. In case of disabling of writing data to disk or configure the models to write data with large time intervals (i.e., monthly), the simulations with active co-processing component will run much faster and make the analysis of the model results in real time efficiently especially in live mode (see Section **??**).

5   ~~Results~~Demonstration application

~~As it indicated in the previous sections, the newly designed modelling system is able to~~ The newly designed modeling system can analyze numerical simulation results in both in-situ (or live) and co-processing ~~(or post-processing) mode and this~~ modes. In this case, a Python script, that defines the visualization pipeline, mainly controls the selection of the operating mode and is generated using ParaView, co-processing plugin. The user could also activate live visualization mode, just by changing a single line of code (need to set coprocessor.EnableLiveVisualization as True) in Python script. This section aims to give more detailed information about two different approaches by evaluating numerical simulation of Hurricane Katrina in both ~~mode~~ models to reveal the designed ~~modelling~~ modeling system capability and its limitations.

5.1 Live ~~Visualization Mode~~visualization mode

While the live visualization designed to examine the simulation state at a specific point in time, the temporal filters such as ParticlePath, ParticleTracer, TemporalStatistics that are designed to process data using multiple time steps cannot be used in this mode. However, live visualization mode allows ~~to connect to~~ connecting to the running simulation anytime through the ParaView GUI in order to make detailed analysis by modifying existing visualization pipelines defined by a Python script. In this case, the numerical simulation can be paused while the visualization pipeline is modified and will continue to run with the revised one. It is ~~oblivious~~ evident that the live visualization capability gives ~~a~~ full control to the user to make further investigation about the simulation results and facilitate better insight into the underlying physical process and its evolution in time.

The current version of the co-processing enabled modeling system ~~is able to~~ can process data of multiple model components by using multi-channel input port feature of ParaView/Catalyst. In this case, each model has two input channels based on the rank of exchange fields. For example, atmospheric model component has *atm_input2d* and *atm_input3d* input channels to make ~~available processing~~ processing available both two and three-dimensional exchange fields. The ~~underlaying ESMF adaptor resides in the driver side and~~ underlying adaptor code resides between the NUOPC cap of co-processing component and ParaView/Catalyst and provides two grid definitions (2d and 3d) for each model components for further analysis. In this design, the ParaView Co-processing Plugin is used to generate Python co-processing scripts~~and user need~~, and user needs to map data sources to input channels by using predefined names such as *atm_input2d* and *ocn_input3d*. Then, adaptor provides the required data to co-processing component through each channel to perform rendering and data analysis in real time. The fields that are used in the co-processing component are defined by generic ASCII formatted driver configuration file (*exfield.tbl*), which is also used to ~~exchange data among physical~~ define exchange fields among other model components such as atmosphere and ocean models. Fig. **??** shows a screenshot of live visualization of three-dimensional relative humidity field provided by the low-resolution atmospheric model component, underlying topography information, and vorticity of ocean surface that is provided by ocean model component.

5.2 Co-processing ~~Mode~~mode

In addition to live visualization ~~model~~ mode that is described briefly in the previous section, ParaView/Catalyst also allows to ~~process~~ render and store data using predefined co-processing pipeline (in Python) for further analysis. Co-processing mode can be used for ~~two purposes: 1)~~ three purposes: (1) the simulation output can be directed to the co-processing component to ~~calculate~~ render data in batch mode and write image files to the disk, (2) added value information ~~such as~~ (i.e., vorticity from wind components~~or~~, eddy kinetic energy from ocean current) can be calculated and stored in a disk for further analysis and ~~2)~~ (3) storing simulation output in a higher temporal resolution to process it later (post-processing) or create a representative dataset that can be used to ~~create~~ design visualization pipeline for co-processing or live visualization modes. In this case, the newly designed ~~modelling system is able to~~ modeling system can apply multiple visualization and data processing ~~pipeline~~ pipelines to the simulation results at each coupling time step to make a different set of analysis ~~in the~~ over the results of

same numerical simulation for more efficient data analysis. The modeling system also facilitates multiple input ports to process data flowing from multiple ESM components. In this design, input ports are defined automatically by the co-processing component based on activated model components (ATM, OCN, etc.) and each model component has two ports to handle two and three-dimensional grids (and fields) separately such as *atm_input2d*, *atm_input3d*, *ocn_input2d* and *ocn_input3d*.

To test the capability of the co-processing component, the evolution of Hurricane Katrina is investigated by using two different configurations of the coupled model (COP_LR and COP_HR) that are also used to analyze the overall computational performance of the modeling system (see Section **??**). In this case, both model configurations use the same configuration of OCN model component, but the different horizontal resolution of the ATM model is considered (27 km for LR and 3 km for HR cases).

Figure **??** shows 3-hourly snapshots of the model simulated clouds that are generated by processing three-dimensional relative humidity field calculated by the low-resolution version of the coupled model (COP_LR) using NVIDIA IndeX volume rendering plugin as well as streamlines of Hurricane Katrina, which is calculated using three-dimensional wind field. The visualization pipeline also includes sea surface height and surface current from the ocean model component to make an integrated analysis of the model results. Figure **??**a-b shows the streamlines that are produced by extracting the hurricane using ParaView *Threshold* filter. In this case, the extracted region is used as a seed to calculate backward and forward streamlines. In Figure **??**c-e, sea surface height, sea surface current and surface wind vectors (10-meters) are shown together to give insight about the interaction of ocean-related variables with the atmospheric wind. Lastly, the hurricane reaches to the land and starts to disappear due to increased surface roughness and lack of energy source (Fig. **??**f). While the low-resolution of atmosphere model configuration is used, the information produced by the new modeling system enabled to investigate the evolution of the hurricane in a very high temporal resolution, which was impossible before. A day-long animation that is also used to create Figure **??** can be found as a supplemental video (**?**).

In addition low-resolution model results revealing the evolution of the hurricane in a very high temporal resolution, low and high-resolution model results are also compared to see the added value of the increased horizontal resolution of the atmospheric model component regarding representation of the hurricane and its structure. To that end, a set of visualization pipelines are designed to investigate the vertical updraft in the hurricane, simulated track, precipitation pattern, and ocean state. In this case, two time snapshots are considered: (1) 28 August 2005 0000 UTC, at the early stage of the hurricane in Category 5 and (2) 29 August 2005 0000 UTC just before Katrina makes its third and final landfall near Louisiana–Mississippi border, where the surface wind is powerful, and surface currents had a strong onshore component (**??**). In the analysis of vertical structure, the hurricane is isolated based on the criteria of surface wind speed that exceeds 20 ms^{-1} and the seed (basically set of points defined as *vtkPoints*) for ParaView *StreamTracerWithCustomSource* filter are defined dynamically using *ProgrammableFilter* as a circular plane with a radius of 1.2° and points distributed with 0.2° interval in both direction (x and y) around the center of mass of the isolated region. Then, forward and backward streamlines of vorticity are computed separately to see inflow at low and mid levels and outflow at upper levels for both low (COP_LR; Fig. **??**a, b, d and e) and

22

high-resolution (COP_HR; Fig. **??**a, b, d and e) cases. The analysis of simulations reveal that the vertical air movement shows higher spatial variability in high-resolution simulation (COP_HR) case even if the overall structure of the hurricane is similar in both cases. As ~~it is~~ expected, the strongest winds occur in a region ~~formed as~~ forming a ring around the eyewall of the hurricane, which is where the lowest surface pressure occurs. ~~In addition~~

Also, the analysis of cloud liquid water content ~~also~~ shows that low and mid-levels of the hurricane have higher water content ~~in a decreasing trend with height~~ and spatial distribution of precipitation is better represented in ~~high resolution~~ high-resolution case (Fig. **??**a-b and d-e), which is consistent with the previous ~~modelling~~ modeling study of **?**.

It is also seen that the realistic principal and secondary precipitation bands around the eye of the hurricane are more apparent and well structured in the high-resolution simulation while the low-resolution case does not show those ~~small scale~~ small-scale features (Fig. **??**a-b and d-e). ~~In addition to the analysis of the atmospheric model component~~On the ocean side, the loop current, which is a warm ocean current that flows northward between Cuba and the Yucatan Peninsula and moves north into the Gulf of Mexico, loops east and south before exiting to the east through the Florida Straits and joining the Gulf Stream ~~,~~ and is well defined by the ocean model component in both cases (Fig. **??**c and f; Fig. **??**c and f). The track of the hurricane is also compared with the HURDAT2 second-generation North Atlantic (NATL) hurricane database, which is the longest and most complete record of tropical cyclone (TC) activity in any of the world's oceans (**?**). In this case, the eye of the hurricane is extracted as a region ~~that~~ with surface pressure anomaly is greater than 15 ~~mb~~ millibar (shown as a circular region near the best track). As it can be seen from the figures, Katrina ~~move~~ moves over in the central Gulf, which is mainly associated with the loop current and persistent warm and cold eddies, and ~~intensified as it passed~~ intensifies as it passes over the region due to the high ocean heat content in both simulation (Fig. **??**c and f and Fig. **??**c and f). The comparison of the low and high-resolution simulations also indicate that the diameter of hurricane-force winds at peak intensity is bigger in high-resolution simulation case at 29 August 2005 0000 UTC (Fig. **??**f and Fig. **??**f). An animation that shows the comparison of low and high-resolution model results can be found as a supplemental video (**?**).

While the main aim of this paper is to give design details of the new in-situ visualization integrated modeling system and show its capability, the performance of the coupled modeling system to represent one of the most destructive ~~hurricane~~ hurricanes is very satisfactory especially for high-resolution case (COP_HR). Nonetheless, the individual components (atmosphere and ocean) of the modeling system can be tuned to have better agreement with the available observations and previous studies. Specifically for the analysis of the hurricane, a better storm tracking algorithm ~~need~~ needs to be implemented using ParaView *Programmable Filter* by porting existing legacy Fortran codes for more accurate storm tracking in both live and co-processing mode.

6 Discussion of the concepts associated with interoperability, portability, and reproducibility

In the current design of the RegESM modeling system, the NUOPC cap of the co-processing component is designed to work with regional modeling applications that have specific horizontal grid (or mesh) types such as rectilinear and curvilinear grids. The newly introduced co-processing interface (NUOPC cap and adaptor code) now needs to be generalized to be compatible

with other regional and global modeling systems coupled through ESMF and NUOPC layer. Specifically, the following issues need to be addressed to achieve better interoperability with existing modeling systems and model components: (1) redesigning the NUOPC cap of the co-processing component to support various global and regional mesh types such as Cubed-Sphere and unstructured Voronoi meshes, (2) extending the adaptor code to represent mesh and exchange fields provided by NUOPC cap using VTK and ParaView/Catalyst APIs, (3) adding support to co-processing interface for models with online nesting capability, and (4) adding support to have common horizontal grid definitions in the co-processing component and in the other components to make integrated analysis of data (i.e., calculating air-sea temperature difference and correlation) produced by the various model components. Moreover, the co-processing interface can be tightly integrated with the NUOPC layer to provide a simplified API for designing new in-situ visualization integrated modeling systems in an efficient and standardized way. Besides the configuration used in this study, the ~~newly developed state-of-art~~ RegESM modeling system is also tested with different model configurations such as coupling RegCM, MITgcm, and co-processing component to investigate air-sea interaction in the Black Sea basin. Initial results show that the co-processing component can also successfully process data flowing from different model configurations supported by RegESM.

When the diverse nature of high-performance computing systems, their hardware infrastructure (i.e., performance networks and storage systems) and software stacks (i.e., operating systems, compilers, libraries for inter-node communication and their different versions) are considered, realizing fully portable modeling system becoming increasingly crucial for the scientific community. In this case, the detailed examination of possible configurations of the modeling system and exiting computing environments can help to improve the flexibility and portability of the developed modeling system. Specifically for RegESM modeling system, the use case application and benchmark simulations revealed that the single executable approach (combining all model components into one program) used in the design of the modeling system can cause a portability problem when visualization and simulation run on concurrent resources. In the case of a homogeneous computing environment (all nodes with or without GPU support), the in-situ enabled modeling system runs without any particular problem because each MPI (Message Passing Interface) process has access to the same software and hardware resources. In contrast, some computing systems may have homogeneous underlying hardware and software stack (e.g., mixed servers with and without GPU support). As a result, the simulation with in-situ visualization would fail due to missing shared software libraries in underlying GPU. In this case, two approaches can be used to overcome the problem: (1) installation of required libraries on the entire system even on servers that do not have GPU support, and (2) restructuring modeling system to support two executables, one for the co-processing component and one for the physical model component. The second approach is considered a more generic and flexible solution and enhances the portability of the modeling system. It also allows implementing a loosely coupled in-situ visualization system and enables the use of specialized hardware (GPU and more memory) for rendering (**?**). The main drawback of the loosely coupled in-situ visualization approach is that it requires transferring data over the network. As a result, the network performance can be a bottleneck for the modeling system, especially for high-resolution multi-component modeling applications.

When the complexity of regional and global ESMs are considered, developing fully reproducible, and portable modeling system is a challenging task and requires significant human interaction to keep track of detailed metadata and provenance

24

information about the model, simulation and computing environment (in both software and hardware levels). The use of scientific workflows in Earth System science has demonstrated advantages in terms of metadata, provenance, error handling, and reproducibility in an automatized and standardized way (**???**). Additionally, the rapid development in the software container technology can help to design flexible and portable computing environments. Hence, the Docker container was implemented to examine the feasibility of using the container approach for our newly developed in-situ visualization integrated modeling system. A container is a standard unit of software that helps to create a software package including all its dependencies, which can then be ported from one computing environment to another without worrying about the underlying hardware infrastructure and software stack. It also enhances the numerical reproducibility of simulations by creating a standardized computing environment isolated from any dependencies. In this study, the Docker is selected as a container environment because it is widely adopted across the software industry and has a very active user community. Despite the flexibility and easy to use nature of Docker containers, using specialized hardware such as NVIDIA GPUs, which require kernel modules and user-level libraries to operate, is not supported natively. Therefore, Docker container cannot access the underlying GPU resource to perform hardware level rendering for visualization and data analysis. To enable portable GPU-based containers, NVIDIA developed a special container that loads GPU driver into the container at lunch. As a part of this study, the newly developed RegESM modeling system was tested with both Docker (software rendering through the use of Mesa library) and NVIDIA Docker (hardware based rendering). The initial results show that RegESM can take advantage of the container approach to create portable and reproducible modeling system in both in-situ and co-processing modes without considerable performance loss (5-10%). The added value of using NVIDIA Docker is that it enables to utilize the underlying GPU resource to perform rendering (i.e., representation of clouds using direct volume rendering method). More information about a Docker container for in-situ visualization enabled modeling system can be found in the dedicated GitHub repository (see code availability section).

7 Summary and conclusions

In this study, the newly developed in-situ visualization integrated modeling system (RegESM 1.1) is used to demonstrate the feasibility and added value of the integrated modeling environment to analyze the high volume of data coming from a multi-component ESM in an integrated way, which was not possible before. In this case, ParaView/Catalyst is used as a co-processing component to process and render data. The results of the selected use case (evolution of Hurricane Katrina) show that the co-processing component provides an easy-to-use and generic modeling and data analysis environment, which is independent of the underlying physical model components used. Moreover, it promotes the usage of co-processing capability with the existing ESMs coupled using ESMF framework and NUOPC layer, without significant code restructuring and development and helps to increase the interoperability between ESMs and ParaView, co-processing plugin (Catalyst). In the current implementation, the prototype version of the adaptor code acts as an abstraction layer to simplify and standardize the regular tasks to integrate the simulation code with in-situ visual-

825 ization and analysis environment. The driver is also responsible ~~to redistribute~~ for redistributing the data to the co-processing

component while preserving its numerical grid along with the support of vertical interpolation. ~~The coupling of~~ Coupling of

the co-processing component with the generic driver ~~facilitate to define~~ facilitates the definition of custom data processing

pipelines (defined by ~~special~~ Python scripts) ~~easily and allows integrated~~ and allows analysis of data originated from differ-

ent components (i.e., atmosphere and ocean models) of the RegESM modeling system in a very high temporal resolution.

830 In this way, RegESM modeling system can be used to study various physical processes (i.e., extreme precipitation events,

air-sea interaction, convection, and turbulence) that could not be analyzed with the ~~conventional~~ traditional post-processing

approaches.

While the results of the in-situ visualization integrated modeling system are encouraging, the co-processing component will

be extended to support different regional and global computational grid representations supported by ESMF library such as

835 unstructured meshes for having a generic adaptor for various model applications. Additionally, we are currently exploring:

~~1)~~ (1) the way to optimize the ~~grid transfer feature and mapping~~ transfer of grid features and mapping of exchange fields

to enhance the overall performance of the ~~modelling~~ modeling environment in terms of memory usage and computational

efficiency especially for very ~~high resolution~~ high-resolution applications (< 3 km), ~~2)~~ (2) possibility of automatic detection

of accelerators (GPUs) through the use of driver component and assigning available GPU resources automatically to the co-

840 processing component for rendering, ~~3) improving modelling~~ (3) improving modeling system and co-processing component

to allow nested applications (both atmosphere and ocean), ~~4)~~ (4) developing more application of the integrated modeling

environment (possibly with other ocean and atmosphere components such as WRF and MITgcm) to analyze different physical

processes such as air-sea interactions in upwelling regions under extreme atmospheric forcing conditions.

Code availability. The RegESM modeling system is open source and available under the MIT License, making it suitable for community

845 usage. The license allows modification, distribution, private and commercial uses. The source code for all versions of RegESM driver

including 1.1 is distributed through the public code repository hosted by GitHub (https://github.com/uturuncoglu/RegESM). The user guide

and detailed information about the modeling system are also distributed along with the source code in the same code repository. The RegESM

source code includes the required code patches for the individual model components to be used as a component in the modeling system. On

the other hand, the source code of individual model components such as the ocean, wave, and river routing components and co-processing tool

850 (ParaView/Catalyst) used in the modeling system are distributed mainly by their home institutes that might apply different licensing types.

The reader who wants to get more information about the individual model components and their license type can refer to their websites. The

release version 1.1 is permanently archived on Zenodo and accessible under the digital object identifier doi:10.5281/zenodo.1307212. The

demo configuration of the modeling system that is used in NVIDIA GPU Technology Conference (GTC) 2018 is also permanently archived

on Zenodo and accessible under the digital object identifier doi:10.5281/zenodo.1474753. The repository also includes detailed information

855 about the installation of the individual components of the modeling system, third-party libraries, and commands to create Docker container.

Competing interests. The author declares that he has no conflict of interest.

26

860

References

865   Ahrens, J.: Increasing scientific data insights about exascale class simulations under power and storage constraints. IEEE Computer Graphics and Applications, 35, 2, pp. 8–11, 2015.

Ahrens, J., Jourdain, S., O'Leary, P., Patchett, J., Rogers, D.H., Fasel, P., Bauer, A., Petersen, M., Samsel, F.: In Situ MPAS-Ocean Image-Based Visualization. Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis SC14, 16–21 Nov. 2014, New Orleans, LA, USA, 2014.

870   Ahrens, J., Geveci, B., Law, C.: ParaView: An End-User Tool for Large Data Visualization, Visualization Handbook, Elsevier, ISBN-13: 978-0123875822, 2005.

Ahern, S.: High Performance Visualization: Enabling Extreme- Scale Scientific Insight. CRC Press/Francis–Taylor Group, 2012, ch. The Path to Exascale, pp. 331–353. ISBN 978-1439875728

Alexander, K. and Easterbrook, S. M.: The software architecture of climate models: a graphical comparison of CMIP5 and EMICAR5
875   configurations, Geosci. Model Dev., 8, 1221–1232, 2015.

Amante, C. and Eakins, B. W.: ETOPO1 1 Arc-Minute Global Relief Model: Procedures, Data Sources and Analysis, NOAA Technical Memorandum NESDIS NGDC-24, 19 pp, 2009.

Artale, V., Calmanti, S., Carillo, A., Dell'Aquila, A., Hermann, M., Pisacane, G., Ruti, P.M., Sannino, G., Struglia, M.V., Giorgi, F., Bi, X., Pal, J.S., Rauscher, S.: An atmosphere ocean regional climate model for the mediterranean area: assessment of a present climate
880   simulation. Climate Dynamics, 35, 721–740, 2010.

Ashby, S., Beckman, P., Chen, J., Colella, P., Collins, B., Crawford, D., Dongarra, J., Kothe, D., Lusk, R., Messina, P., Mezzacappa, T., Moin, P., Norman, M., Rosner, R., Sarkar, V., Siegel, A., Streitz, F., White, A., Wright, M.: The opportunities and challenges of exascale computing: summary report of the advanced scientific computing advisory committee (ASCAC) subcommittee at the US Department of Energy Office of Science. pp. 71, 2010.

885   Ayachit, U.: The ParaView Guide: A Parallel Visualization Application, Kitware, ISBN 978-1930934306, 2010.

~~Bettencourt, M.T.: Distributed model coupling framework , in: Proceedings of the 11th IEEE Symposium on High Performance Distributed Computing, Edinburgh, UK, 23–26 July 2002, 284–290, 2002.~~

Bruggeman, J., Bolding, K.: A general framework for aquatic biogeochemical models. Environmental Modelling & Software, 61, 249–265, 2014.

890   Childs, H., Brugger, E., Whitlock, B., Meredith, J., Ahern, S., Pugmire, D., Biagas, K., Miller, M., Harrison, C., Weber, G.H., Krishnan, H., Fogal, T., Sanderson, A., Garth, C., Bethel, E.W., -Camp, D., Rubel, O., Durant, M., Favre, J.M., Navr'atil, P.: VisIt: An End-User Tool For Visualizing and Analyzing Very Large Data. High Performance Visualization-Enabling Extreme-Scale Scientific Insight, Chapman and Hall/CRC, pp. 520, 2012.

~~Collins, N., Theurich, G., Deluca, C., Suarez, M., Trayanov,~~

895   Craig A., ~~Balaji, V., Li, P., Yang, W., Hill, C., and Da Silva, A.,: Design and implementation of components in the Earth System Modeling Framework, International Journal of High Performance Computing Applications, 19, 341–350, 2005.~~ Valcke S., Coquart L.: Development and performance of a new version of the OASIS coupler, OASIS3-MCT_3.0, Geoscientific Model Development, 10, pp. 3297–3308, 2017.

Dee, D. P., Uppala, S. M., Simmons, A. J., Berrisford, P., Poli, P., Kobayashi, S., Andrae, U., Balmaseda, M. A., Balsamo, G., Bauer, P.,
900   Bechtold, P., Beljaars, A. C. M., van de Berg, L., Bid- lot, J., Bormann, N., Delsol, C., Dragani, R., Fuentes, M., Geer, A. J., Haimberger,

L., Healy, S. B., Hersbach, H., Holm, E. V., Isaksen, L., Kallberg, P., Kohler, M., Matricardi, M., McNally, A. P., Monge-Sanz, B. M., Morcrette, J.-J., Park, B.-K., Peubey, C., de Rosnay, P., Tavolato, C., Thepaut, J.-N., and Vitart, F.: The ERA-Interim reanalysis: configuration and performance of the data assimilation system, Q. J. Roy. Meteor. Soc., 137, 553–597, 2011.

Dickinson, R. E., Errico, R. M., Giorgi, F., and Bates, G. T.: A regional climate model for the western United States, Climatic Change, 15, 383–422, 1989.

Dominicis, M.D., Falchetti, S., Trotta, F., Pinardi, N., Giacomelli, L., Napolitano, E., Fazioli, L., Sorgente, R., Haley, P., Lermusiaux, P., Martins, F., Cocco, M.: A relocatable ocean model in support of environmental emergencies. The Costa Concordia emergency case. Ocean Dyn., 64, pp. 667–688, 2014.

Emanuel, K. A.: A scheme for representing cumulus convection in large-scale models, J. Atmos. Sci., 48(21), 2313–2335, 2011.

Emanuel, K. A., and Zivkovic-Rothman, M.: Development and evaluation of a convection scheme for use in climate models, J. Atmos. Sci., 56, 1766–1782, 1999.

Fabian, N., Moreland, K., Thompson, D., Bauer, A., Marion, P., Geveci, B., Rasquin, M., Jansen, K.: The paraview coprocessing library: A scalable, general purpose in situ visualization library. In Large Data Analysis and Visualization (LDAV), 2011 IEEE Symposium on (Oct 2011), pp. 89–96, 2011.

Giorgi, F., Coppola, E., Solmon, F., Mariotti, L., Sylla, M. B., Bi, X., Elguindi, N., Diro, G. T., Nair, V., Giuliani, G., Turuncoglu, U. U., Cozzini, S., Guttler, I., O'Brien, T. A., Tawfik, A. B., Shalaby, A., Zakey, A. S., Steiner, A. L., Stordal, F., Sloan, L. C., and Brankovic, C.: RegCM4: model description and preliminary tests over multiple CORDEX domains, Climate Research, 52, 7–29, 2012.

Grell, G., Dudhia, J., and Stauffer, D. R.: A description of the fifth-generation Penn State/NCAR mesoscale model (MM5), Technical Note NCAR/TN-398+STR, NCAR, 117 pp.,1995.

Haidvogel, D. B., Arango, H. G., Budgell, W. P., Cornuelle, B. D., Curchitser, E., DiLorenzo, E., Fennel, K., Geyer, W. R., Hermann, A. J., Lanerolle, L., Levin, J., McWilliams, J. C., Miller, A. J., Moore, A. M., Powell, T. M., Shchepetkin, A. F., Sherwood, C. R., Signell, R. P., Warner, J. C., and Wilkin, J.: Ocean forecasting in terrain-following coordinates: formulation and skill assessment of the Regional Ocean Modeling System, J. Comput. Phys., 227, 3595–3624, 2008.

Hagemann, S., Dumenil, L.: A parameterization of the lateral waterflow for the global scale. Clim Dyn, 14, pp. 17–31, 1998.

Hagemann, S., Lydia, D.G.: Validation of the hydrological cycle of ECMWF and NCEP reanalyses using the MPI hydrological discharge model. J Geophys Res, 106, pp. 1503–1510, 2001.

Hill, C., DeLuca, C., Balaji, V., Suarez, M., and Da Silva, A.: The architecture of the Earth System Modeling Framework, Comput. Sci. Eng., 6, 18–28, 2004.

Hill, C., DeLuca, C., Balaji, V., Suarez, M., Da Silva, A., Sawyer, W., Cruz, C., Trayanov, A., Zaslavsky, L., Hallberg, R., Boville, B. A., Craig, A., Collins, N., Kluzek, E., Michalakes, J., Neckels, D., Schwab, E., Smithline, S., Wolfe, J., Iredell, M., Yang, W., Jacob, L. R., and Larson, J. W.: Implementing Applications with the Earth System Modeling Framework, Lecture Notes in Computer Science, 3732, 563–572, 2004.

Hostetler, S. W., Bates, G. T., and Giorgi, F.: Interactive Coupling of a Lake Thermal Model with a Regional Climate Model, J. Geophys. Res, 98, 5045–5057, 1993.

Jacob, R., Larson, J., and Ong, E.: M X N Communication and Parallel Interpolation in Community Climate System Model Version, International Journal of High Performance Computing Applications, 19, 293–307, 2005.

Kara, B.A., Wallcraft, A.J., and Hurlburt, H.E.: A Correction for Land Contamination of Atmospheric Variables near Land?Sea Boundaries. J. Phys. Oceanogr., 37, pp. 803–818, 2007.

Landsea, C.W. and Franklin, J.L.: Atlantic Hurricane Database Uncertainty and Presentation of a New Database Format. Mon. Wea. Rev.,

940   141, 3576–3592, 2013.

Larson, J., Jacob, R., and Ong, E.: The Model Coupling Toolkit: a new Fortran90 toolkit for building multiphysics parallel coupled models, International Journal of High Performance Computing Applications, 19, 277–292, 2005.

~~Liu, L., Yang, G., Wang, B., Zhang, C. , Li,~~

Lemmen, C., Hofmeister, R., ~~Zhang, Z., Ji, Y~~Klingbeil, K., Nasermoaddeli, M. H., Kerimoglu, O., Burchard, H., Kösters, F., and ~~Wang, L.

945   : C-Coupler1: a Chinese community coupler for Earth system modeling~~Wirtz, K. W.: Modular System for Shelves and Coasts (MOSSCO v1.0) – a flexible and multi-component framework for coupled coastal ocean ecosystem modelling, Geosci. Model Dev., ~~7, 2281–2302, 2014.~~11, 915–935, 2018.

McTaggart-Cowan, R., Bosart, L.F., Gyakum, J.R., and Atallah, E.H.: Hurricane Katrina (2005). Part I: Complex Life Cycle of an Intense Tropical Cyclone. Mon. Wea. Rev., 135, 3905–3926, 2007.

950   McTaggart-Cowan, R., Bosart, L.F., Gyakum, J.R., and Atallah, E.H.: Hurricane Katrina (2005). Part II: Evolution and Hemispheric Impacts of a Diabatically Generated Warm Pool. Mon. Wea. Rev., 135, 3927–3949, 2007.

Malakar, P., Natarajan, V., Vadhiyar, S.: Integrated parallelization of computations and visualization for large-scale applications. IEEE 26th International Parallel and Distributed Processing Symposium Workshops PhD Forum (IPDPSW), 21-25 May 2012, Shanghai, China, 2012.

955   Marshall, J., Adcroft, A., Hill, C., Perelman, L., Heisey, C.: A finite-volume, incompressible Navier Stokes model for, studies of the ocean on parallel computers. J Geophys Res C Oceans, 102, C3, 5753–5766, 1997.

Marshall, J., Hill, C., Perelman, L., Adcroft, A.: Hydrostatic, quasi-hydrostatic, and non-hydrostatic ocean modeling. J Geophys Res C Oceans,102, C3, 5733–5752, 1997.

Mellor, G. L., and Yamada, T.: Development of a turbulence closure model for geophysical fluid problems, Rev. Geophys., 20, 851–875,

960   1982.

Monbaliu, J., Hargreaves, R., Albiach, J., Luo, W., Sclavo, M., and Gunther, H.: The spectral 5 wave model, WAM, adapted for applications with high spatial resolution, Coast. Eng., 41, 41–62, 2000.

Moreland, K.: A Survey of Visualization Pipelines. IEEE Transactions on Visualization and Computer Graphics,19(3), 2013.

Moreland, K., Sewell, C., Usher, W., Lo, L-t., Meredith, J., Pugmire, D., Kress, J., Schroots, H., Ma, K-L., Childs, H., Larsen, M., Chen,

965   C-M., Maynard, R., Geveci, B.: VTK-m: Accelerating the Visualization Toolkit for Massively Threaded Architectures. IEEE Computer Graphics and Applications, 36(3), May/June 2016.

O'Leary, P., Ahrens, J., Jourdain, S., Wittenburg, S., Rogers, D.H., Petersen, M.: Cinema image-based in situ analysis and visualization of MPAS-ocean simulations. Parallel Computing, 55, pp. 43–48, 2016.

Pal, J. S., Small, E.E, and Eltahir, E.A.B.: Simulation of regional-scale water and energy budgets: Representation of subgrid cloud and

970   precipitation processes within RegCM, J. Geophys. Res.-Atmospheres, 105(D24), 29579–29594, 2000.

Pelupessy, I., van Werkhoven, B., van Elteren, A., Viebahn, J., Candy, A., Portegies Zwart, S., and Dijkstra, H.: The Oceanographic Multipurpose Software Environment (OMUSE v1.0), Geosci. Model Dev., 10, 3167–3187, 2017.

Redler, R., Valcke, S., and Ritzdorf, H.: OASIS4 ~~?~~ – a coupling software for next generation earth system modelling, Geosci. Model Dev., 3, 87–104, 2010.

975   Rivi, M., Calori, L., Muscianisi, G., and Salvnic, V.: In-situ Visualization: State-of-the-art and Some Use Cases. Tech. rep., Whitepaper PRACE, 2012.

Overeem, I., Berlin, M.M., Syvitski, J.P.M.: Strategies for integrated modeling: The community surface dynamics modeling system example, Environmental Modelling & Software, 39, 2013.

Shchepetkin, A. F. and McWilliams, J. C.: The Regional Ocean Modeling following coordinates ocean model, Ocean Modell., 9, 347–404, 2005.

980

Skamarock, W. C., Klemp, J.B., Dudhia, J., Gill, D.O., Barker, D.M., Wang, W., and Powers, J.G.: A description of the Advanced Research WRF version 2. NCAR Tech. Note TN-468+STR, 88 pp., 2005.

Surenkok, G. and Turuncoglu, U.U.: Investigating the Role of Atmosphere–Wave Interaction in the Mediterranean Sea using coupled climate model (RegESM). EGU General Assembly 2015, 12–17 April 2015, 17, EGU2015-3644, Vienna, Austria, 2015.

985

Tawfik, A.B. and Steiner, A.L.: The role of soil ice in land–atmosphere coupling over the United States: a soil moisture precipitation winter feedback mechanism. J Geophys Res. 116, D02113, 2011.

Theurich, G., DeLuca, C., Campbell, T., Liu, F., Saint, K., Vertenstein, M., Chen, J., Oehmke, R., Doyle, J., Whitcomb, T., Wallcraft, A., Iredell, M., Black, T., Da Silva, A.M., Clune, T., Ferraro, R., Li, P., Kelley, M., Aleinov, I., Balaji, V., Zadeh, N., Jacob, R., Kirtman, B., Giraldo, F., McCarren, D., Sandgathe, S., Peckham, S., and Dunlap, R.: The Earth System Prediction Suite: Toward a Coordinated U.S.

990

Modeling Capability. Bull. Amer. Meteor. Soc., 97, 1229–1247, 2016.

Trenberth, K.E., Davis, C.A., and Fasullo, J.: Water and energy budgets of hurricanes: Case studies of Ivan and Katrina, J. Geophys. Res., 112, D23106, 2007.

Turuncoglu, U. U.: Tools for Configuring, Building and Running Models (Vol. 5) - Applying Scientific Workflow to ESM (Sec. 2). Earth System Modeling, SpringerBriefs in Earth System Sciences. Series Editor: Rupert Ford. 70p, ISBN: 978-3-642-23931-1, 2012.

995

Turuncoglu U.U., Murphy S., DeLuca C., Dalfes N.: A scientific workflow environment for earth system related studies, Computers & Geosciences, Vol. 37(7), pp. 943–952, 2011.

Turuncoglu U.U., Dalfes N., Murphy S., DeLuca C.: Towards self-describing and workflow integrated Earth system models: a coupled atmosphere-ocean modeling system application, Environmental Modeling and Software, 39, pp. 247–262, 2012.

Turuncoglu, U.U., Giuliani, G., Elguindi, N., Giorgi, F.: Modelling the Caspian Sea and its catchment area using a coupled regional

1000

atmosphere-ocean model (RegCM4-ROMS): model design and preliminary results, Geosci. Model Dev., 6, pp. 283–299, 2013.

Turuncoglu, U.U.: Simulation of Hurricane Katrina using in situ visualization integrated regional earth system model (RegESM). German National Library of Science and Technology (TIB), AV Portal of TIB Hannover, doi:10.5446/37227, 2018.

Turuncoglu, U.U.: In situ visualisation of Hurricane Katrina. German National Library of Science and Technology (TIB), AV Portal of TIB Hannover, doi:10.5446/37219, 2018.

1005

Turuncoglu, U.U., and Sannino, G.: Validation of newly designed regional earth system model (RegESM) for Mediterranean Basin. Climate Dynamics, 48, 9–10, pp. 2919–2947, 2017.

Valcke, S.: The OASIS3 coupler: a European climate modelling community software, Geosci. Model Dev., 6, 373–388, 2013.

Woodring, J., Petersen, M., Schmeisser, A., Patch-Ett, J., Ahrens, J., Hagen H.: In situ eddy analysis in a high-resolution ocean climate model. IEEE Transactions on Visualization and Computer Graphics, 22, 1, pp. 857–866, 2016.

1010

Zeng, X., Zhao, M., and Dickinson, R. E.: Intercomparison of bulk aerodynamic algorithms for computation of sea surface fluxes using TOGA COARE and TAO data, J. Climate, 11, 2628–2644, 1998.
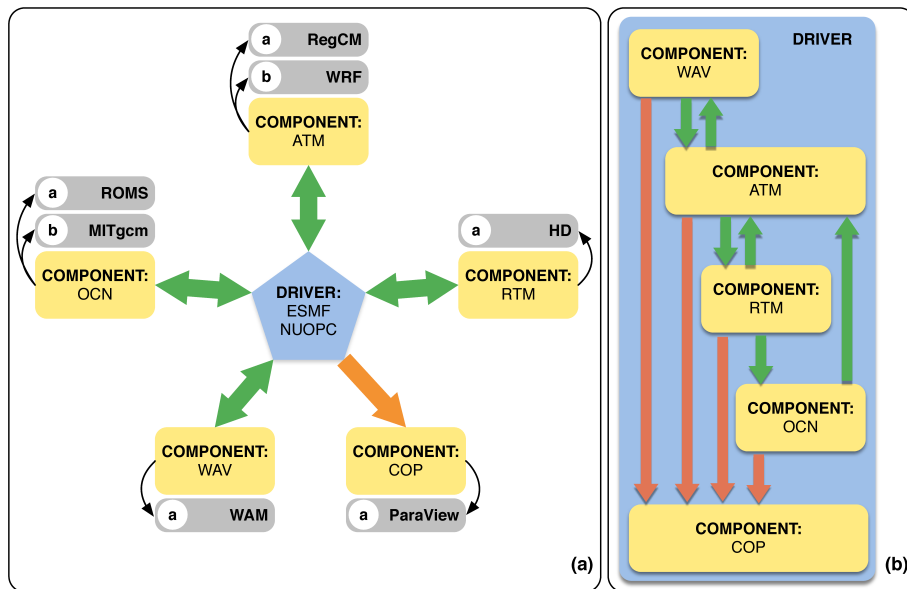
Figure 1. Design of the RegESM coupled modeling system: ~~(a)~~ (a) model components including co-processing component, ~~(b)~~ (b) their interactions (orange arrows represent the redistribution and green arrows shows ~~regridding~~interpolation).
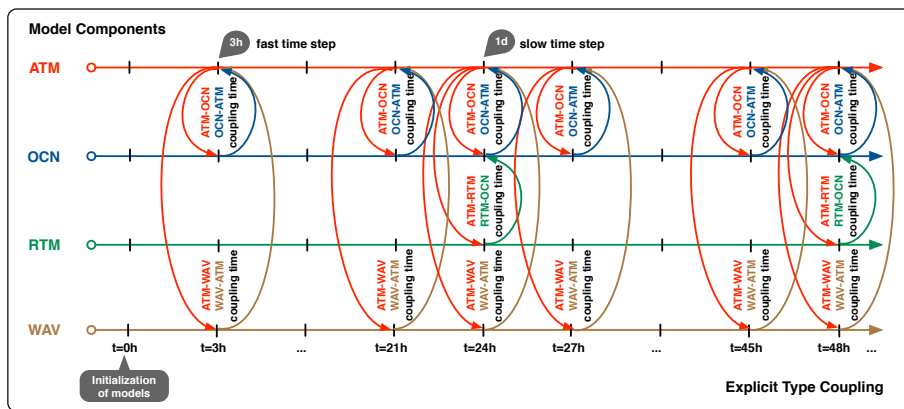
Figure 2. The run sequence of model components in case of explicit type coupling. In this case, the fast coupling time step is used for the interaction between the atmosphere, ocean and wave components. The slow coupling time step is only used to interact with the river routing component.
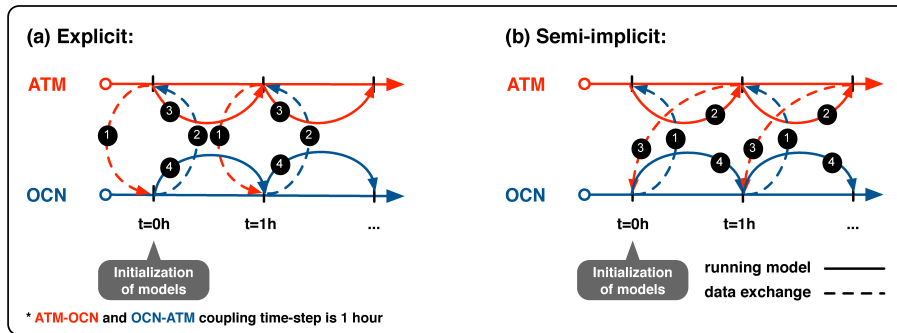
Figure 3. Schematic representation of ~~(a)~~ (a) explicit and ~~(b)~~ (b) semi-implicit model coupling between two model components (atmosphere and ocean). The numbers indicate the execution orders, which is initialized in each coupling interval.

Source Grid | Destination Grid

Start

Create ESMF Field_A

Create ESMF Field_B

Create ESMF Field_C

Interpolation *
Field_A to Field_B

Interpolation *
Field_A to Field_C

Nearest-neighbor

Bilinear

Do not change mask of Field_C

Initial Values

Field_A
LARGE over land, ONE over sea

Field_B
LARGE in both land and sea

Field_C
ZERO in both land and sea

Field_B < LARGE/2 ?

AND

Field_C is over SEA ?

False

True

Set Field_C mask as **MAPPED**

False

Field_C < ONE/2 ?

True

Set Field_C mask as **UNMAPPED**

* All interpolations are performed only on grid points over SEA

LARGE = 1.0e20

Figure 4. Processing flow chart of the algorithm to find mapped and unmapped grid points for two-step interpolation.
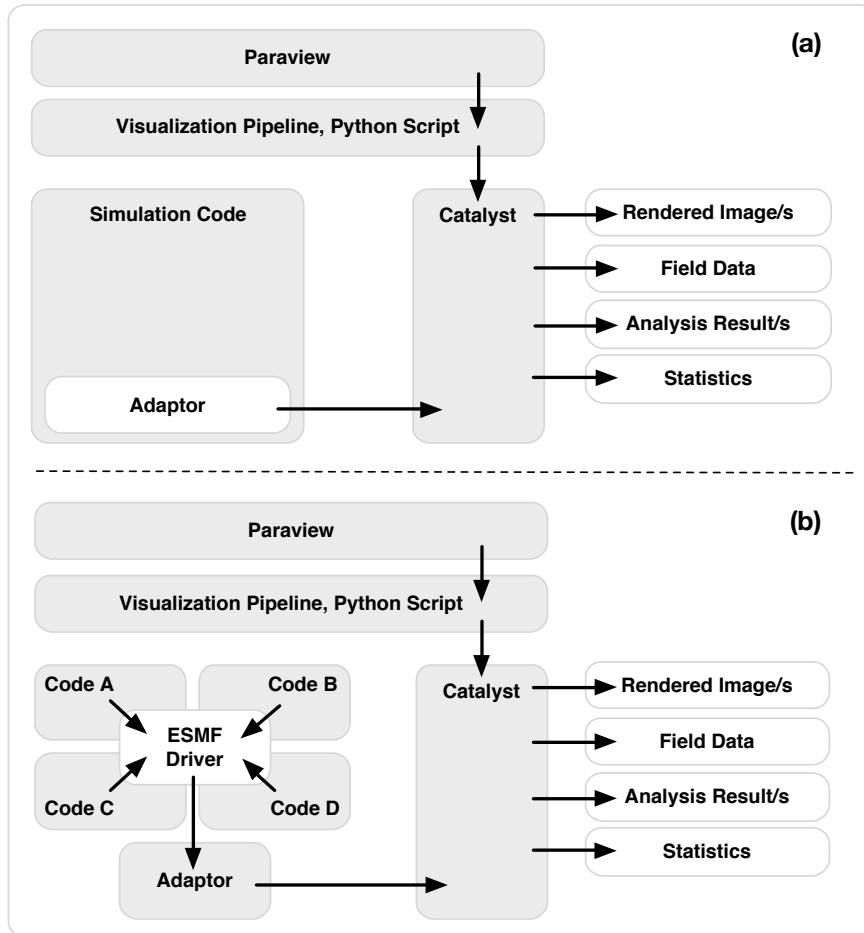
Figure 5. Comparison of the ~~(a)~~ (a) conventional and ~~(b)~~ (b) ESMF integrated in-situ visualization system.
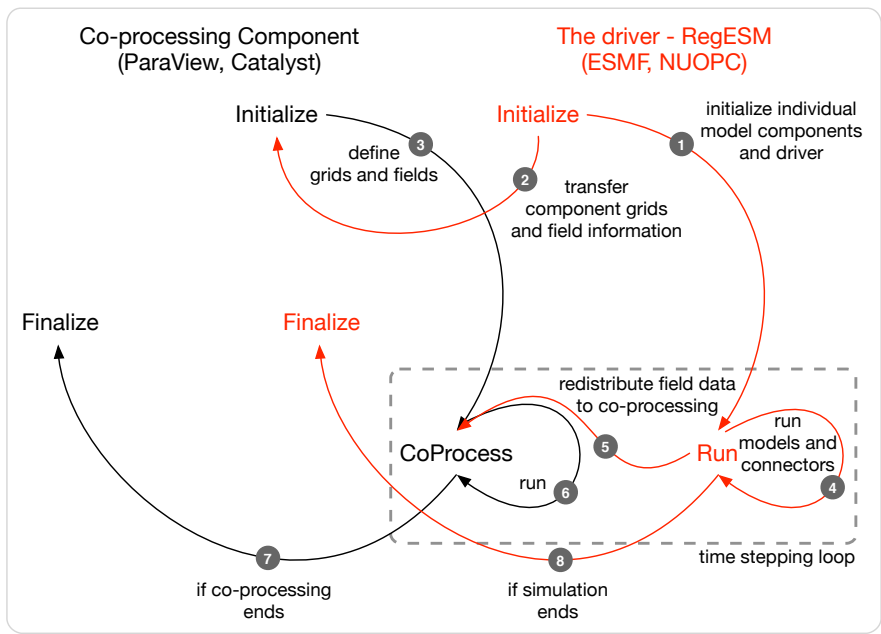
Figure 6. The interaction between driver defined by ESMF, NUOPC and co-processing component (Paraview/Catalyst).
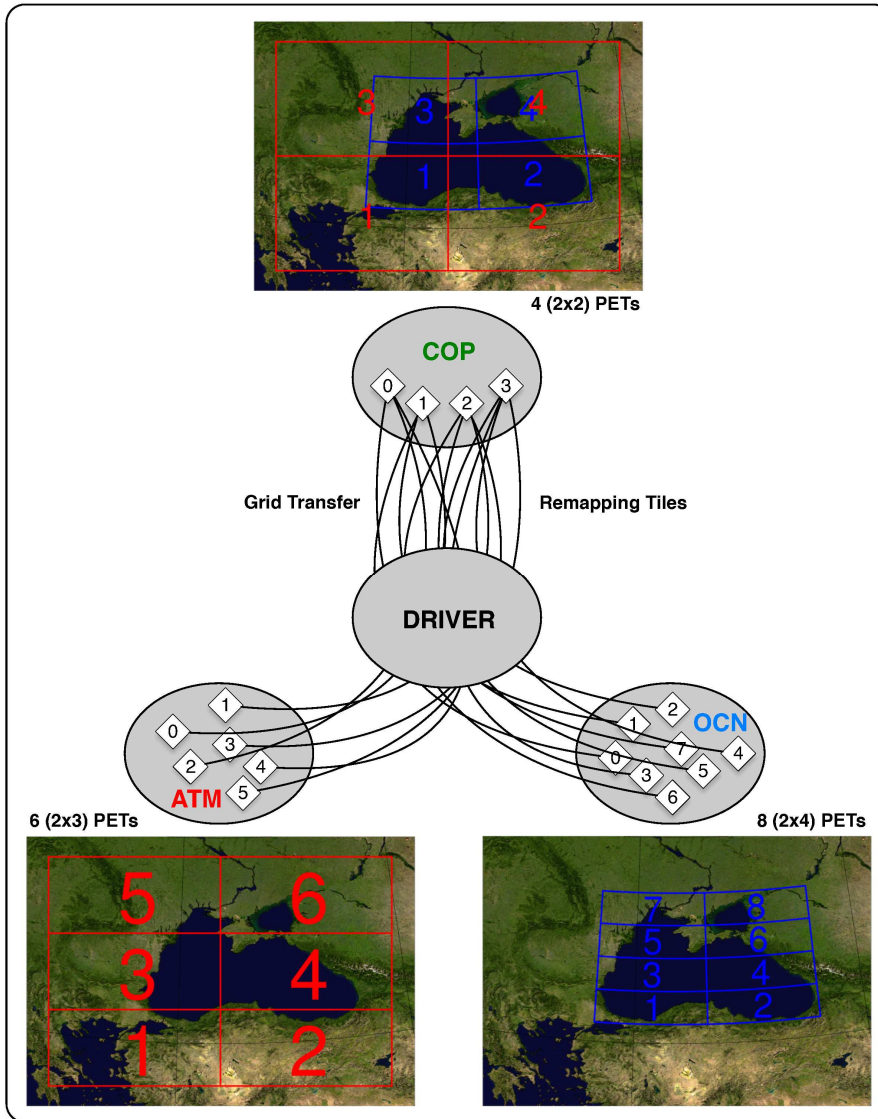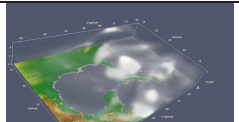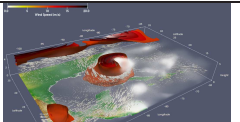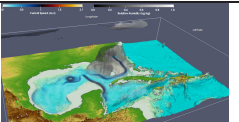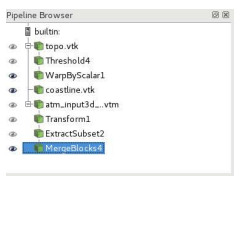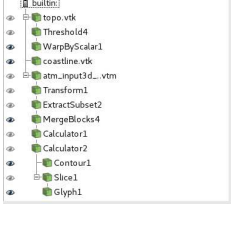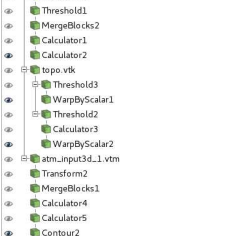
Figure 7. ~~Two component~~ Two-component (atmosphere and ocean) representation of grid transfer and remapping feature of ESMF/NUOPC interface.
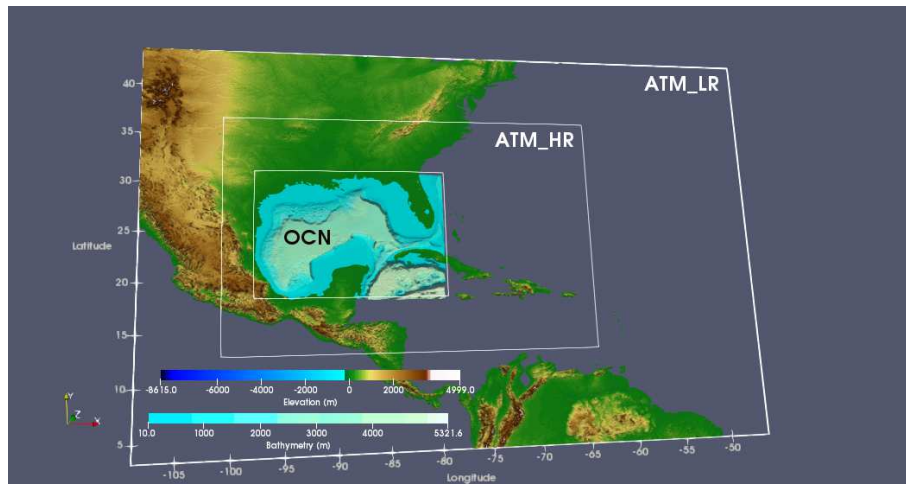
Figure 8. Domain for the RegESM simulations with topography and bathymetry of the region. The solid white boxes represent boundaries of the atmosphere (both outer and inner) and ocean model domains.

Table 1. Tested model configurations for benchmark simulations. Note that the dimension of vertical coordinates of ATM and OCN components are shown ~~in~~ here after vertical interpolation from sigma to height and s-coordinates to depth. The visualization pipelines are also given as supplementary material.

|	P1: Case I	P2: Case II	P3: Case III
Visualization			
Pipeline			
Primitives	**ATM:** Contour for topography polyline for coastline and direct volume rendering for clouds	**ATM:** same with previous case but it includes iso-surface for wind speed and glyph for wind at specified level	**ATM:** Contour for topography, iso-surface for wind speed colored by relative humidity **OCN:** Contour for bathymetry, direct volume rendering for current
Domain Size	**ATM** **LR:** 170 x 235 x 27 **HR:** 880 x 1240 x 27	**ATM** Same with **Case I**	**ATM** Same with **Case I** **OCN** 653 x 487 x 21
Number of Fields	**1 x 3D ATM** Relative Humidity	**4 x 3D ATM** Relative Humidity Wind (u, v, w)	**4 x 3D ATM** Relative Humidity Wind (u, v, w) **4 x 3D OCN** Ocean Current (u, v, w) Land-Sea Mask
Data Size ATM+OCN (MB)	**LR:** 8.3 **HR:** 224.0	**LR:** 33.2 **HR:** 896.0	**LR:** 33.2+25.4 = 58.6 **HR:** 896.0+25.4 = 921.4
Time (s)	**LR:** 2.3 – 3.7 **HR:** 17.7 – 65.0	**LR:** 2.3 – 3.8 **HR:** 18.4 – 79.3	**LR:** 6.8 – 14.6 **HR:** 7.8 – 10.1
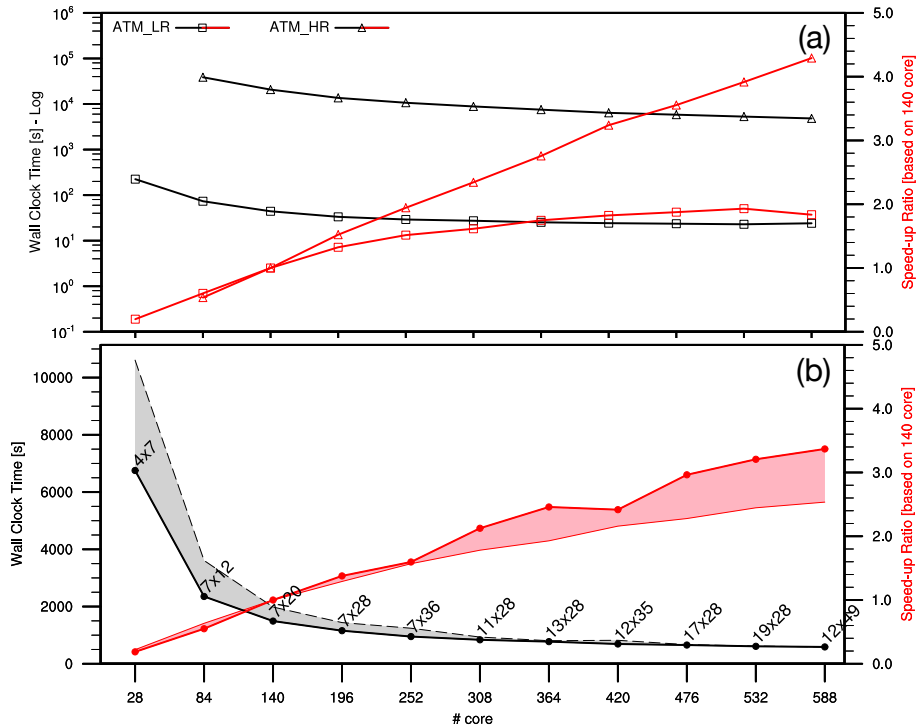
Figure 9. Benchmark results of standalone ~~(a)~~ (a) atmosphere (ATM; both LR and HR) and ~~(b)~~ (b) ocean (OCN) models. Note that timing results of the atmosphere model ~~is~~ are in log axes to show both LR and HR cases in the same figure. The black lines represent measured wall clock times in second and red lines ~~(and shaded envelope)~~ show speed-up. The envelope represents the timing and speed-up results that are done using the same number of cores but different two-dimensional decomposition configuration. The best two-dimensional decomposition parameters ~~, timing results and speed-up~~ are also shown ~~as line~~ in the timing results for the ocean model case.
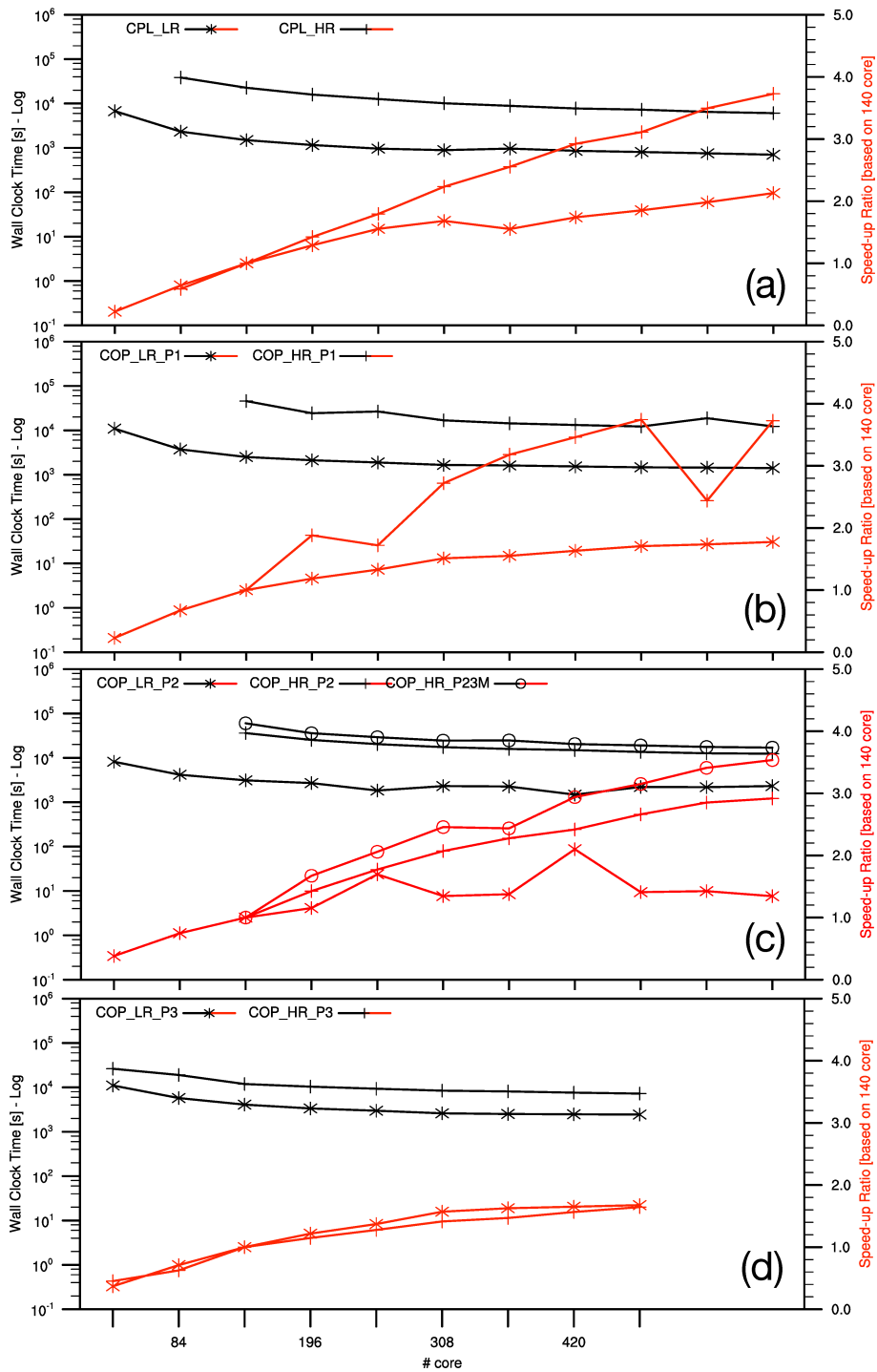
Figure 10. Benchmark results of ~~coupled model (a) LR~~ (a) CPL simulations (b) COP simulations with P1 visualization pipeline, (c) COP simulations with P2 visualization pipeline and ~~(b) HR cases~~(d) COP simulations with P3 visualization pipeline. CPL represents the two-component ~~case~~ modeling system (ATM and OCN), and COP ~~shows~~ indicates three-component ~~case including~~ modeling system (ATM, OCN and co-processing~~component~~). Note that the HR case requires at least 140 cores to run and the speed-up results are given based on 140 cores.
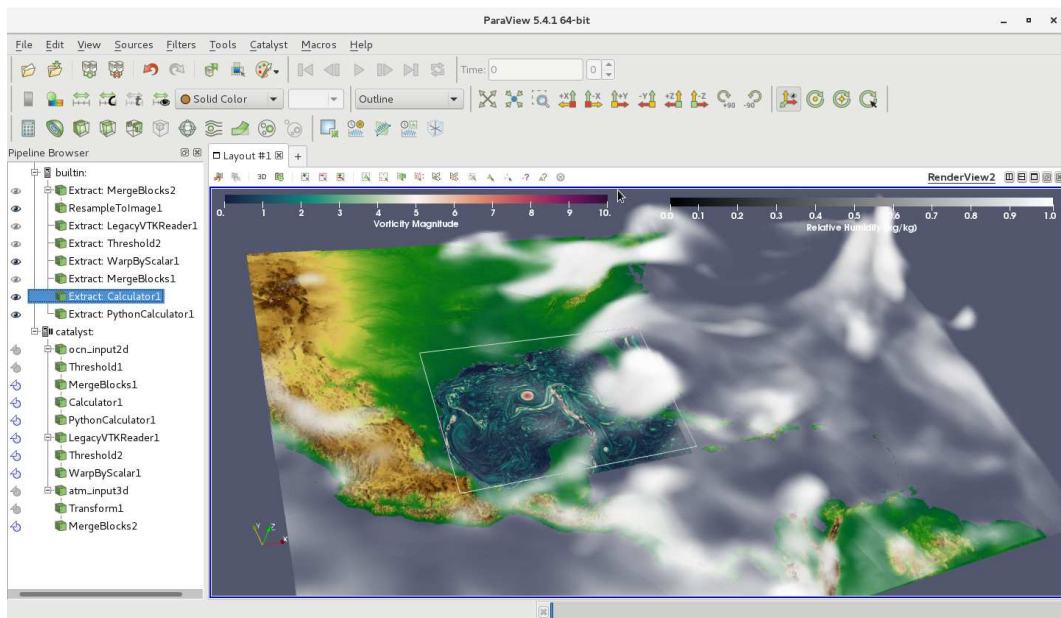
Figure 11. Volume rendering of atmospheric relative humidity field (*atm_input3d*) as well as vorticity field in the ocean surface (*ocn_input2d*) from COP_LR simulation using ParaView/Catalyst in live mode.
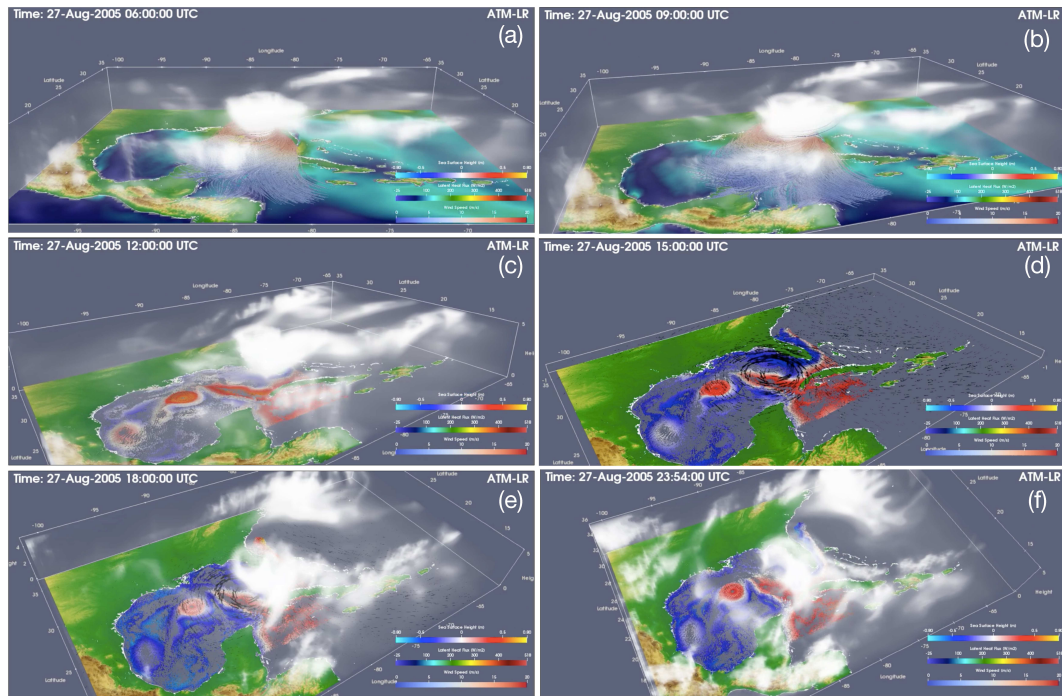
Figure 12. Rendering of multi-component (ATM-OCN-COP) fully coupled simulation using ParaView. The temporal interval for the processed data is defined as 6-minutes. ~~The evolution of the hurricane is shown in supplemental video.~~

44
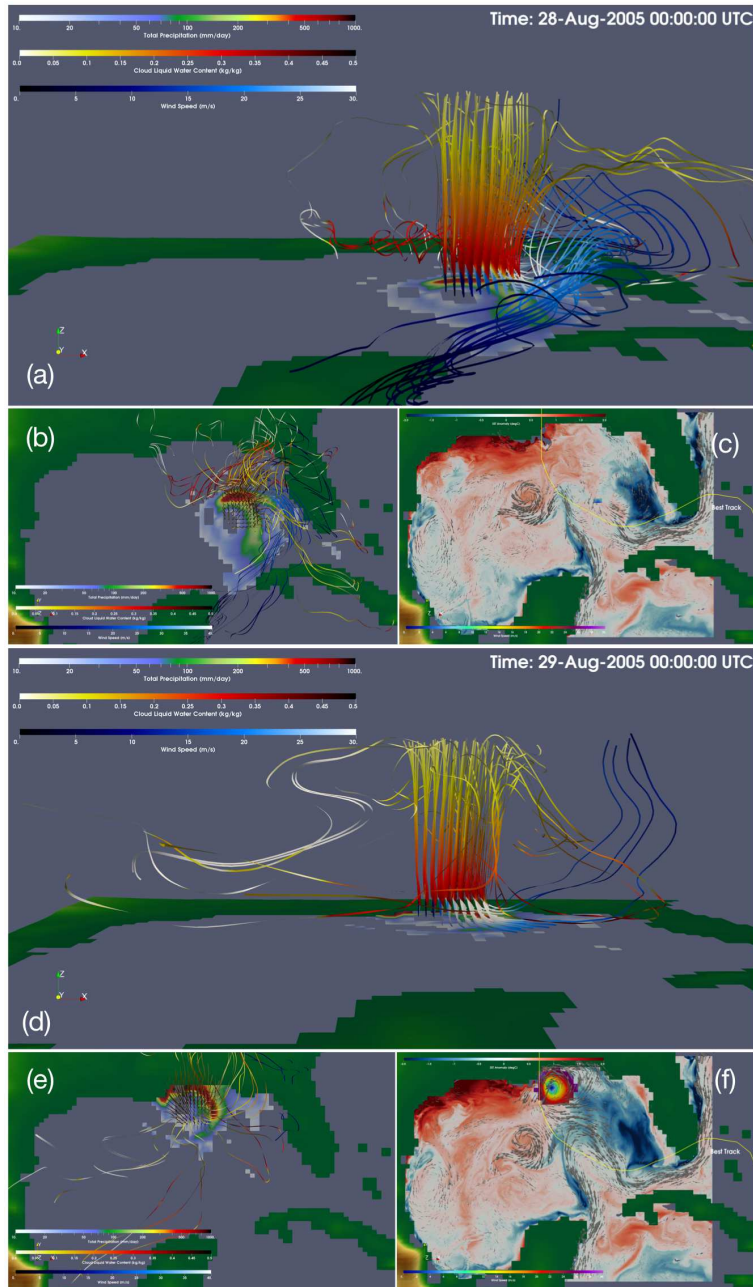
Figure 13. Rendering of three-dimensional vorticity streamlines (~~1/s~~ s^{-1}), total precipitation (~~mm/day~~ $mmday^{-1}$) and sea surface temperature anomaly (degC) of COP_LR simulation for 28-Aug-2005 00:00 UTC **(a-c)** and 29-Aug-2005 00:00 UTC **(d-f)**. Streamlines are calculated only from the eye of the hurricane. In this case, red and yellow colored forward streamlines ~~represents~~ represent cloud liquid water content (~~kg/kg~~ $kgkg^{-1}$), and blue colored backward streamlines ~~indicates~~ indicate wind speed (~~m/s~~ ms^{-1}). The ~~yellow~~ solid yellow line represents the best track of Hurricane Katrina, which is extracted from HURDAT2 database. The larger versions of figures are also given as supplementary material.
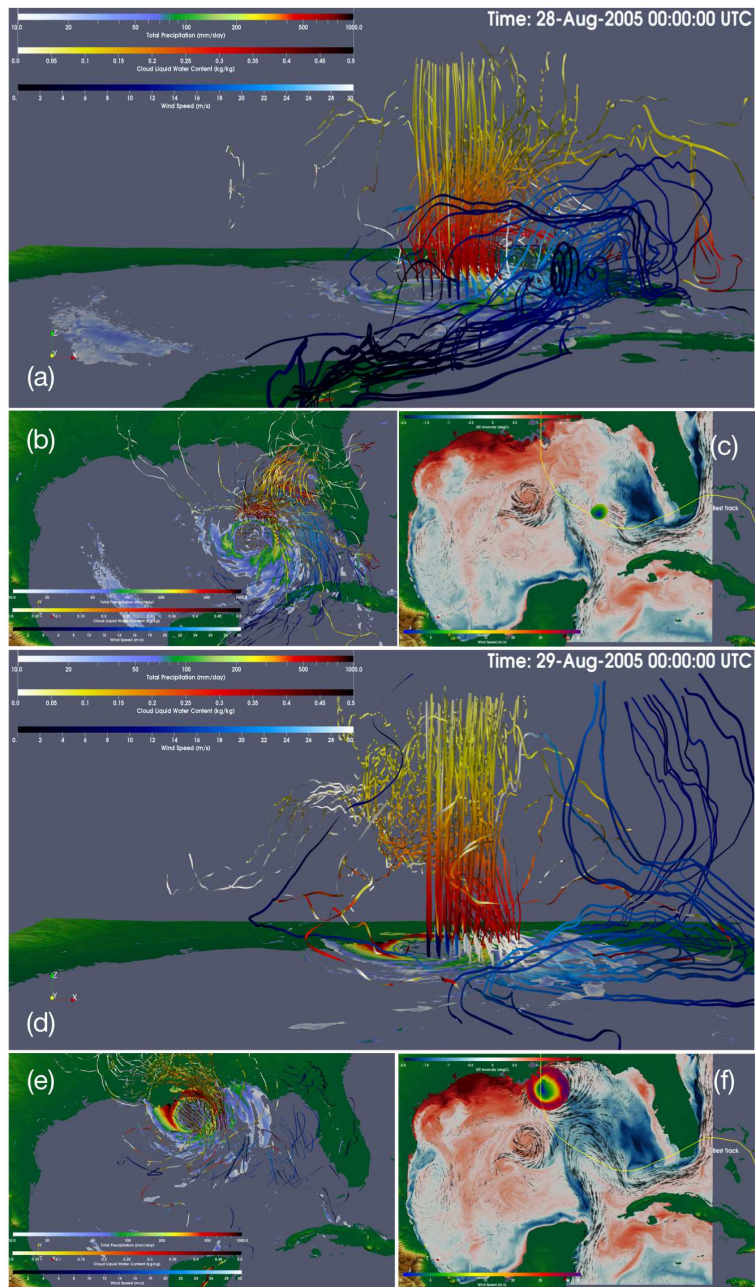
Figure 14. Same with Fig. **??** but for COP_HR simulation. The larger versions of figures are also given as supplementary material. The comparison of low and ~~high resolution~~ high-resolution model results is shown in the supplemental video.