

Project Report on

DATA VISUALIZATION AND ANALYSIS OF WRF-CHEM MODEL ON HIGH PERFORMANCE CLUSTER

At
Physical Research Laboratory (PRL)



External Guide :

Dr. Varun Sheel.
Associate Professor,
Space and Atmospheric Science Division,
Physical Research Laboratory,
Ahmedabad

Prepared By:

Pansuriya Dip Bharathbhai.
(10012011051)

**B.Tech Semester VIII
(Computer Engineering)
May 2014**

Submitted to,
Department of Computer Engineering
U.V. Patel College of Engineering
Ganpat University, Kherva - 384 012

U.V.PATEL COLLEGE OF ENGINEERING



10/05/2014

C E R T I F I C A T E

T O W H O M S O E V E R I T M A Y C O N C E R N

This is to certify that Mr. Pansuriya Dip Bharatbhai (10012011051) student of B.Tech. Semester VIII (Computer Engineering) has completed his full semester on site project work titled "***Data Visualization and Analysis of WRF-CHEM Model On High Performance Cluster***" satisfactorily in partial fulfillment of the requirement of Bachelor of Technology degree of Computer Engineering of Ganpat University, Kherva, Mehsana in the year 2013-2014.

College Project Guide

Sign
(Prof. Rajendra J. Patel)

Dr. Kiran R. Amin,
Head, Computer Engineering

COMPANY PROFILE



“PRL research encompasses The Earth, the Sun immersed in the field and radiations reaching from and to infinity, all that man’s curiosity and intellect can reveal.”

Known as the cradle of Space Sciences in India, the Physical Research Laboratory (PRL) was founded on 11 November 1947 by **Dr. Vikram Sarabhai**.

The institute was formally established at the M.G. Science Institute, Ahmedabad, with support from the Karmkshetra Educational Foundation and the Ahmedabad Education Society. Prof. Kalpathi Ramakrishna Ramanathan was the first Director of the institute. The initial focus was research on Cosmic Rays and the properties of the upper atmosphere. Research areas were expanded to include Theoretical Physics and Radio Physics later with grants from the Atomic Energy Commission.

Today PRL is actively involved in research, related to five major fields of science. PRL is also instrumental in the PLANEX planetary science and exploration programme.

Other Branches and their corresponding Research areas of PRL are:

- 1979:Infra-red Telescope at Mt.Abu
- 1980:Udaipur Solar Observatory becomes a part of PRL
- 1990: Thaltej Campus science at PRL.

Fields of Research at PRL:

- Astronomy & Astrophysics
- Solar Physics
- Space & Atmospheric Sciences
- Earth Sciences
- Planetary Sciences & exploration
- Gravitation & Cosmology
- Particle Physics
- Nuclear, Atomic & Molecular Physics
- Nonlinear Dynamics
- Quantum Optics & Quantum

Space and Atmospheric Sciences Division:

The activities of the Division aim to understand radiative, chemical, ionization and dynamical processes in the Earth's atmosphere by employing in situ rocket and balloon-borne experiments, ground based optical and radio probing, laboratory experiments, theoretical simulation and modelling of atmospheres of Earth and other Planets.

ACKNOWLEDGEMENTS

Though only my name appears on the cover of this Report, a great many people have contributed to its production. I owe my gratitude to all those people who have made this Dissertation possible and because of whom my Dissertation experience has been one that I will cherish forever.

I am extremely grateful to my guide honorable Prof. Rajendra J. Patel for being a source of inspiration and for his constant support in the Design and Evaluation of the Dissertation. He has been the constant constructive force throughout my pursuit of Bachelor Degree at UVPCE and I am sure that his active and passive teachings will always inspire me throughout my life.

I will always remain indebted to my external guide at PRL, Honorable Dr. Varun Sheel. I am thankful to him for his constant constructive criticism and invaluable suggestions, which benefited me a lot while doing research on Analysis & Data metrics in WRF model on HPC cluster. He has been always there to support me whenever I felt down or lost my way and always led me on right path and I am sure that he will continue to do so throughout the course of my stay at PRL.

I also express my gratitude to Mr. Samuel Johnson (PRL) for providing me the infrastructure to carry out the Dissertation and to all staff members who were directly and indirectly instrumental in enabling me to stay committed.

I would like to thank my parents & my friends for always giving me full support, inspiring advices and courage to always follow righteous path whenever my steps have faltered.

ABSTRACT

Weather Research and Forecast model is a flexible atmospheric simulation system that is portable and efficient on available parallel computing platforms. It is used in broad range of applications across scaling from meters to thousands of kilometers. Here WRF is used with chemistry option.

WRF system with its components like WRF pre processing System (WPS), WRF/chem. is installed on HPC cluster at Physical Research Laboratory. WPS pre Processing System is used here to prepare meteorological initial and boundary conditions. Main advantage of WRF model is meteorology plus chemistry coupled simulations and broad range of different types of data support.

A test simulation over 48 states of USA is performed for 60 km grid using National Emission Inventory of 2005. Inventory is Global that contains many kinds of emission like anthropogenic of type EDGAR and RETRO, biomass burning, fire, aerosols and emission for several green house gases. A WRF/Chem run is generated for area which is setting for 25 km grid using PREP_CHEM_SOURCES programming module. Forecast generated from Global inventory contains anthropogenic, fire and biomass burning data. Analysis of anthropogenic emissions like CO, NO, O₃ gives constant values for pressures at ground level that is the ideal case when no external boundary and initial conditions are specified.

TABLE OF CONTENTS

| | |
|---|-------------|
| Title Page..... | I |
| Certificate from Internal Guide..... | II |
| Certificate from External Guide..... | III |
| Company Profile..... | IV |
| Acknowledgements..... | VI |
| Abstract..... | VII |
| Index..... | VIII |
| List of Figures..... | X |
| List of Tables..... | X |
| List of Abbreviations..... | XI |
| | |
| 1. Introduction..... | 1. |
| 1.1 Introduction..... | 1. |
| 1.1.1 Parallel Memory Architecture..... | 1. |
| 1.1.2 Parallel Architecture..... | 4. |
| 1.1.3 Parallel Programming Model..... | 5. |
| 1.2 Problem Statement..... | 6. |
| 1.3 Objective..... | 6. |
| 1.4 Scope..... | 6. |
| | |
| 2. Related Study..... | 7. |
| 2.1 Overview of cluster Architecture..... | 8. |
| 2.2 Overview of WRFmodel Architecture..... | 9. |
| 2.3 WRF-CHEM Model Plan..... | 10. |
| | |
| 3. Literature Survey..... | 11. |
| 3.1 Cluster Components..... | 12. |
| 3.2 Benefits Of Cluster Computing | 13. |
| 3.3 HPCCluster..... | 13. |
| 3.4 PRLHPCCluster..... | 14. |
| 3.5 Study of WRF-CHEM model and its flow architecture..... | 15. |
| 3.6 Diagrams..... | 18. |
| 3.6.1 Use Case Diagram..... | 18. |
| 3.6.2 Class Diagram..... | 19. |
| 3.6.3 Sequence Diagram for Downloading Data..... | 20. |
| 3.6.4 Sequence Diagram For WPS..... | 21. |
| 3.6.5 Sequence Diagram for WRF/Chem..... | 22. |
| | |
| 4. Proposed Work..... | 23. |
| 4.1 Operational Running Flow of WRFModel..... | 24. |

| | |
|--|------------|
| 4.2 AlgorithmicSteps..... | 25. |
| 5. ImplementationWork..... | 26. |
| 5.1 WRF-CHEM Model Setup &Configuration..... | 27. |
| 5.2 Bash file Path settings..... | 29. |
| 5.3 Configuration &Compilation of WRF-CHEM Model..... | 30. |
| 5.4 WPSPre-processingConfiguration &Compilation..... | 34. |
| 5.5 Steps ofWRF-CHEM Simulation Process..... | 36. |
| 6. WRF-CHEM Model Specification..... | 40. |
| 6.1 Implementation Environment | 41. |
| 6.2 Module Specification..... | 42. |
| 6.3 Security Specification..... | 45. |
| 6.4 Contour Plotting..... | 45. |
| 7. ConclusionandFuture Work..... | 48. |
| 7.1 Conclusion..... | 49. |
| 7.2 FutureWork..... | 49. |
| References..... | 50. |

List of Figures

| Fig No. | Description. | Page No. |
|----------------|--|-----------------|
| 1.1 | Uniform MemoryAccess | 3. |
| 1.2 | Non-Uniform MemoryAccess | 3. |
| 1.3 | DistributeMemorySystem | 4. |
| 1.4 | Hybrid Distributed-Shared Architecture | 4. |
| 2.1 | Cluster Architecture | 8. |
| 2.2 | WRFSystem Components | 9. |
| 2.3.1 | Work Breakdown Structure of WRF Modeling System | 10. |
| 3.1 | Stages of ClusteringProcess | 12. |
| 3.2 | Cluster Components | 12. |
| 3.3 | PRLHPCCluster Architecture | 15. |
| 3.4 | PRL HPC Cluster View. | 16. |
| 3.5 | WRF-CHEM modeling system architecture | 17. |
| 3.6.1 | Usecase Diagram | 18. |
| 3.6.2 | Class Diagram | 19. |
| 3.6.3 | Sequence Diagram for Downloading Data | 20. |
| 3.6.4 | Sequence Diagram For WPS | 21. |
| 3.6.5 | Sequence Diagram for WRF/Chem. | 22. |
| 4.1 | Operational Running Flow of WRF Model. | 24. |
| 5.1 | WRF-CHEM ModelSetup& Configuration | 27. |
| 5.2 | bashfile Path settings | 29. |
| 5.3 | Configuration & Compilation ofWRF-CHEM Model | 31. |
| 5.4 | Errors in Compilation of WRF-CHEM | 33. |
| 5.5 | Recompilation ofWRF-CHEM modelafter resolvingthe Errors | 34. |
| 5.6 | WPS Pre-processingConfiguration& Compilation | 35. |
| 6.1 | Carbon Monoxide concentration contour. | 45. |
| 6.2 | CO Concentration contour mapping on India's map. | 46. |
| 6.3 | Ozone[O ₃] concentration contour. | 46. |
| 6.4 | O ₃ Concentration contour mapping on India's map. | 47. |

List of Tables

| Table No | Description | Page No. |
|-----------------|---------------------------------|-----------------|
| 3.1 | HPC ClusterMinimumRequirements. | 13. |
| 6.1 | OS Supported By WRF | 41. |

ABBREVIATIONS

| | |
|---------|---|
| WRF | Weather research and forecasting |
| ARW | Advanced Research WRF |
| HPC | High Performance Computer |
| S/W | Software |
| H/W | Hardware |
| NCAR | National Centre of Atmospheric Research |
| NCL | NCAR command language |
| netCDF | Network Common Data Form |
| OS | Operating System |
| WPS | WRF Pre-processing System. |
| WRF-DA | WRF Data Assimilation |
| OBSGRID | Observation grid |
| PBSpro | Portable Batch System professional |
| MPI | Message Passing Interface |
| CDL | Common Data form Language |
| NWP | Numerical Weather Prediction |
| MMM | Mesoscale and Microscale Meteorology |
| SCSI | Small Computer System Interface |
| SAS | Serial Attached SCSI |
| IPMI | Intelligent Platform Management Interface |

1.

INTRODUCTION

1.1 Introduction.

1.2 Problem Statement.

1.3 Objective.

1.4 Scope.

1.1 INTRODUCTION.

Numerical weather prediction models are critical tools for forecasters. WRF is designed to provide real-time, extremely accurate and sophisticated weather analysis. For efficient analysis, WRF requires high-performance computing system. Commodity clusters have become very important for high performance computing due to the price for performance, flexibility and scalability they can deliver. As Clustering is a group of loosely coupled commodity computers working together to achieve the same goal, maintain a Single System Image, good computational performance and Reliability. Clustering have been used in many fields including machine learning, pattern recognition, image analysis, information retrieval, and bioinformatics.

Cluster analysis as such is not an automatic task, but an iterative process of knowledge discovery or interactive multi-objective optimization. It will often be necessary to modify pre-processing parameters until the result achieves the desired properties.

1.1.1 Parallel Memory Architecture.

In the Traditional sense, software has been written for *serial* computation: To be run on a single computer having a single Central Processing Unit. A problem is broken into a discrete series of instructions. At single instance of time only one instruction may execute. In the case of *parallel computing*, it is the simultaneous use of multiple computing resources to resolve a computational problem: which are to be run using multiple processors. A problem is broken into discrete sections that can be solved concurrently. Each section is further broken down to a series of instructions. Instructions from each section execute simultaneously on different processors [3].

Classification Of Different Architectures.

1. Shared Memory Architecture:

In Shared memory architecture every processor can operate independently but share the same memory resources. In this all processors are sharing memory as global address space. In this architecture care should be taken care for changes in a memory location effected by one processor are visible to all other processors. Based upon memory access times Shared memory architecture is classified as UMA and NUMA [3].

i. Uniform Memory Access (UMA): UMA is the collection of identical processors also known as Symmetric Multiprocessor (SMP) machines. Each of having equal access and access times to memory as shown in figure 1.1.

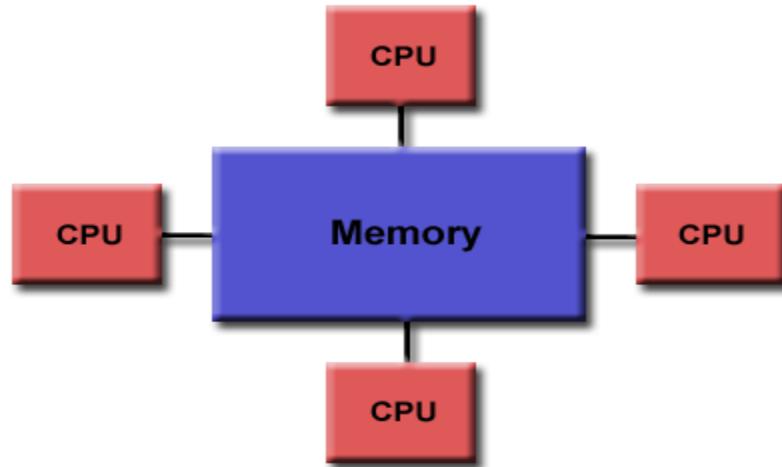


Figure 1.1 Uniform Memory Access

ii. Non-Uniform Memory Access (NUMA): NUMA is often made by physically linking two or more SMPs. One SMP can directly access memory of another SMP. In this not all processors have the same access time to all memories as shown in figure 1.2.

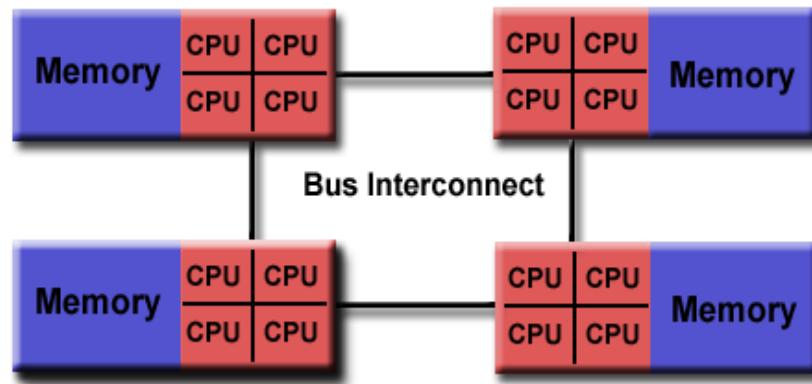


Figure 1.2 Non-Uniform Memory Access [3]

2. Distributed memory systems : It is vary widely but share a common characteristic. Distributed memory systems require a communication network to connect inter- processor memory as shown in figure 1.3.

In Distributed architecture processors have their own local memory. Memory addresses in one processor do not map to another processor, so there is no concept of global address space across all processors. Changes it makes to its local memory have no effect on the memory of other processors. When a processor needs access to data in another processor, it is usually the task of the programmer to explicitly define how and when data is communicated. Here it is the job of programmer to provide synchronization between tasks [3].

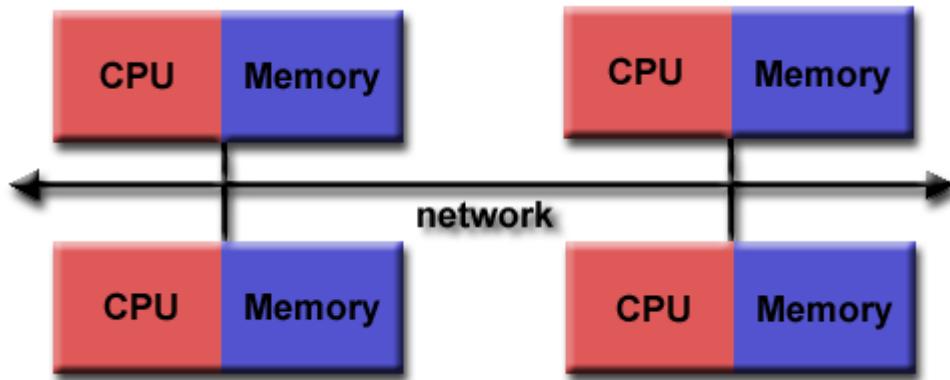


Figure 1.3 Distributed Memory System

3. Hybrid Distributed-Shared Architecture :

Hybrid distributed shared architecture as name itself suggest which employs both shared and distributed memory architectures. The shared memory component can be a shared memory machine and/or graphics processing units (GPU) as shown in figure 1.4.

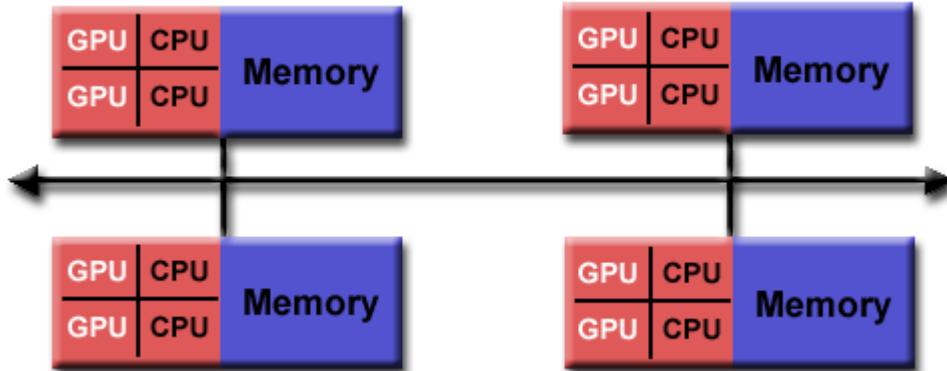


Figure 1.4 Hybrid Distributed-Shared Architecture

1.1.2 Parallel Architecture

1. Single Instruction Single Data (SISD) :

SISD is a term referring to a computer architecture in which a single processor, a uniprocessor, executes a single instruction stream, to operate on data stored in a single memory [12]. It is a serial (non-parallel) computer. **Single Instruction:** Only one instruction stream is being acted on by the CPU during any one clock cycle. **Single Data:** Only one data stream is being used as input during any one clock cycle. Deterministic execution. Examples: older generation mainframes, minicomputers and workstations; most modern day PCs [4].

2. Single Instruction, Multiple Data (SIMD) :

SIMD is a type of parallel computer. It describes computers with multiple processing elements that perform the same operation on multiple data points simultaneously. Thus, such machines exploit data level parallelism [12]. **Single Instruction:** All processing units execute the same instruction at any given clock cycle. **Multiple Data:** Each processing unit can operate on a different data element. Best suited for specialized problems characterized by a high degree of regularity, such as graphics/image processing. Examples: Processor Arrays, Vector Pipelines [4].

3. Multiple Instruction, Single Data (MISD) :

MISD is a type of parallel computing architecture where many functional units perform different operations on the same data [12]. It is a type of parallel computer. **Multiple Instruction:** Each processing unit operates on the data independently via separate instruction streams. **Single Data:** A single data stream is fed into multiple processing units. Examples of this class of parallel computer have ever existed [4].

4. Multiple Instructions Multiple Data (MIMD) :

MIMD is a technique employed to achieve parallelism. Machines using MIMD have a number of processors that function asynchronously and independently. At any time, different processors may be executing different instructions on different pieces of data [12]. It is a type of parallel computer. **Multiple Instruction:** Every processor may be executing a different instruction stream. **Multiple Data:** Every processor may be working with a different data stream. Execution can be synchronous or asynchronous, deterministic or non-deterministic. Examples: most current supercomputers, networked parallel computer clusters and "grids", multi-processor SMP computers, multi-core PCs [4].

1.1.3 Parallel Programming Model

1. Single Program Multiple Data (SPMD) :

SPMD is actually a "high level" programming model that can be built upon any combination of the previously mentioned parallel programming models [4].

2. Multiple Program Multiple Data (MPMD) :

MPMD is actually a "high level" programming model that can be built upon any combination of the previously mentioned parallel programming models [4].

1.2 Problem Statement

“Data Visualization & Analysis of WRF-CHEM model on HPC cluster”

1.3 Objective

- The WRF-CHEM is a Weather research forecasting model for chemistry. WRF-CHEM model is widely used for atmospheric research institutes for Prediction and simulation of weather, or regional and local climate.
- The purpose of the dissertation is to benchmark the WRF-CHEM model on PRL HPC cluster on the basis of different resolutions and for specific regional data of real case studies.
- Make WRF model as much as scalable and flexible by preparing the input data and visualizing it in well-mannered format.
- The main objective of this dissertation is all about to fix the standard for WRF-CHEM on HPC cluster and to make it run successfully over the MPI environment.

1.4 Scope

Aim of the research is to prepare emissions data and deployment it in WRF model which is configured at Physical Research Laboratory, Ahmedabad to help their space & Atmospheric Science Division for weather forecasting and prediction.

Hence the scope of the research is to make efficient and scalable visualization of data which are produced at the time of simulation. For that the jobs are need to be submit in configured HPC cluster. The research scope is to make WRF simulation whole process as much as reliable and effective so computation should be reduced.

2.

RELATED STUDY.

2.1 Overview of Cluster Architecture.

2.2 Overview of WRF model Architecture.

2.3 WRF-CHEM Model Plan.

2.1 Overview of Cluster Architecture.

A cluster computer is a type of parallel or distributed processing system, which consists of a collection of interconnected stand-alone computers working together as a single integrated resource. The typical architecture of a cluster is shown in Figure 2.1 below.

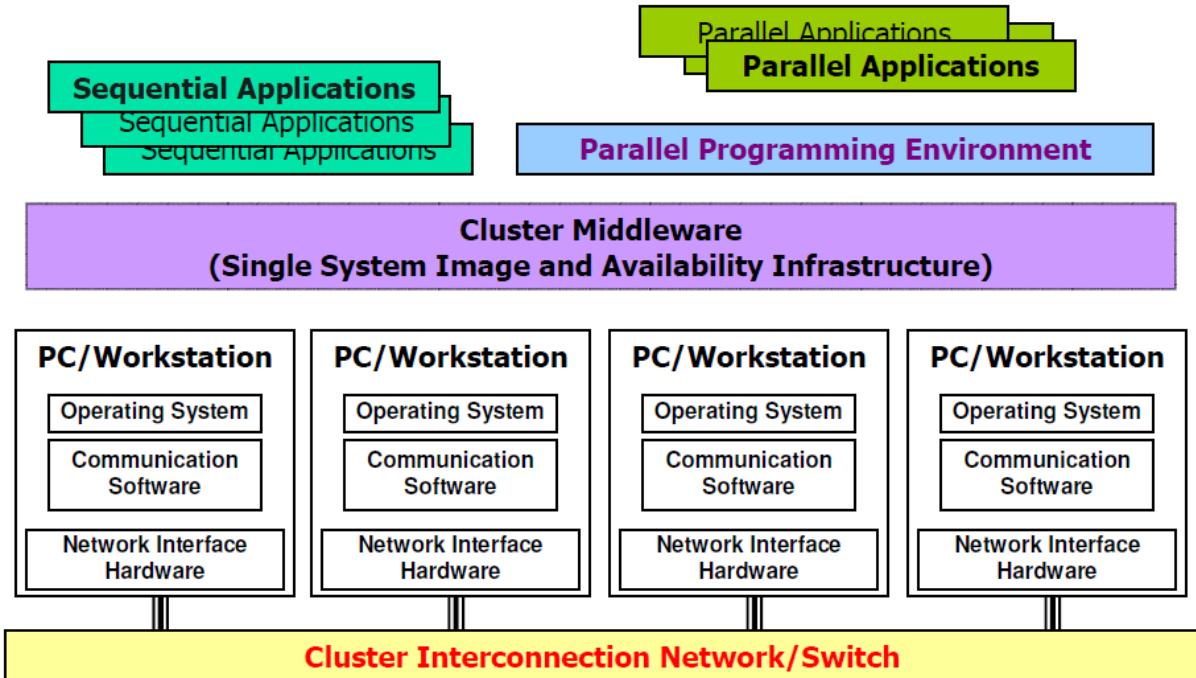


Figure 2.1 Cluster Architecture.

Today, clusters are widely used for research and development of science, engineering, commerce and industry applications that demand high performance computations.

In addition, clusters encompass strengths such as high availability and scalability that motivate wide usage in non-supercomputing applications as well, such as clusters working as web and database servers.

The key components of a cluster include multiple standalone computers (PCs, Workstations, or SMPs), operating systems, high-performance interconnects, middleware, parallel programming environments, and applications.

2.2 Overview of WRF model Architecture [15]

The development of the Weather Research and Forecasting (WRF) modeling system is a multi-agency effort intended to provide a next-generation mesoscale forecast model and data assimilation system to advance both the understanding and prediction of mesoscale weather and accelerate the transfer of research advances into operations. The model was developed as a collaborative effort among the NCAR Mesoscale and Microscale Meteorology (MMM) Division, the National Oceanic and Atmospheric Administration's (NOAA) National Centers for Environmental Prediction (NCEP) and Forecast System Laboratory (FSL), the Department of Defense's Air Force Weather Agency (AFWA) and Naval Research Laboratory (NRL), the Center for Analysis and Prediction of Storms (CAPS) at the University of Oklahoma, and the Federal Aviation Administration (FAA), along with the participation of a number of university scientists.

The WRF model is designed to be an efficient massively parallel computing code to be able to take advantage of advanced high-performance computing systems. The code can be configured for both research and operations and offers numerous physics options. WRF is maintained and supported as a community model to facilitate wide use, and is suitable for use in a broad spectrum of applications across scales ranging from meters to thousands of kilometers. Such applications include research and operational NWP, data assimilation and parameterized-physics research, downscaling climate simulations, driving air quality models, atmosphere-ocean coupling, and idealized simulations.

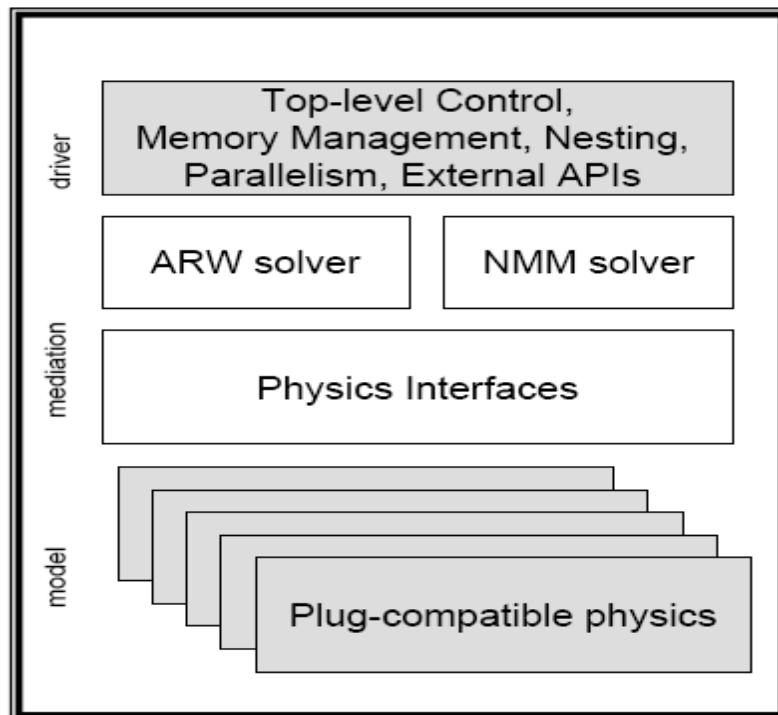


Figure 2.2 WRF System Components.

The WRF Software Framework (WSF) (Figure 2.2) provides the infrastructure that allows efficient use of an array of HPC systems, architectures which continue to evolve as we move into Petascale computing and beyond. The architecture accommodates multiple dynamics solvers, physics packages that plug into the solvers through a standard physics interface, programs for initialization, and the WRF variational data assimilation (WRFVar) system. There are two dynamics solvers in the WSF: the Advanced Research WRF (ARW) solver developed primarily at NCAR, and the NMM (Nonhydrostatic Mesoscale Model) solver developed at NCEP.

2.3 WRF-CHEM Model Plan.

2.3.1 Work Batch Down Structure.

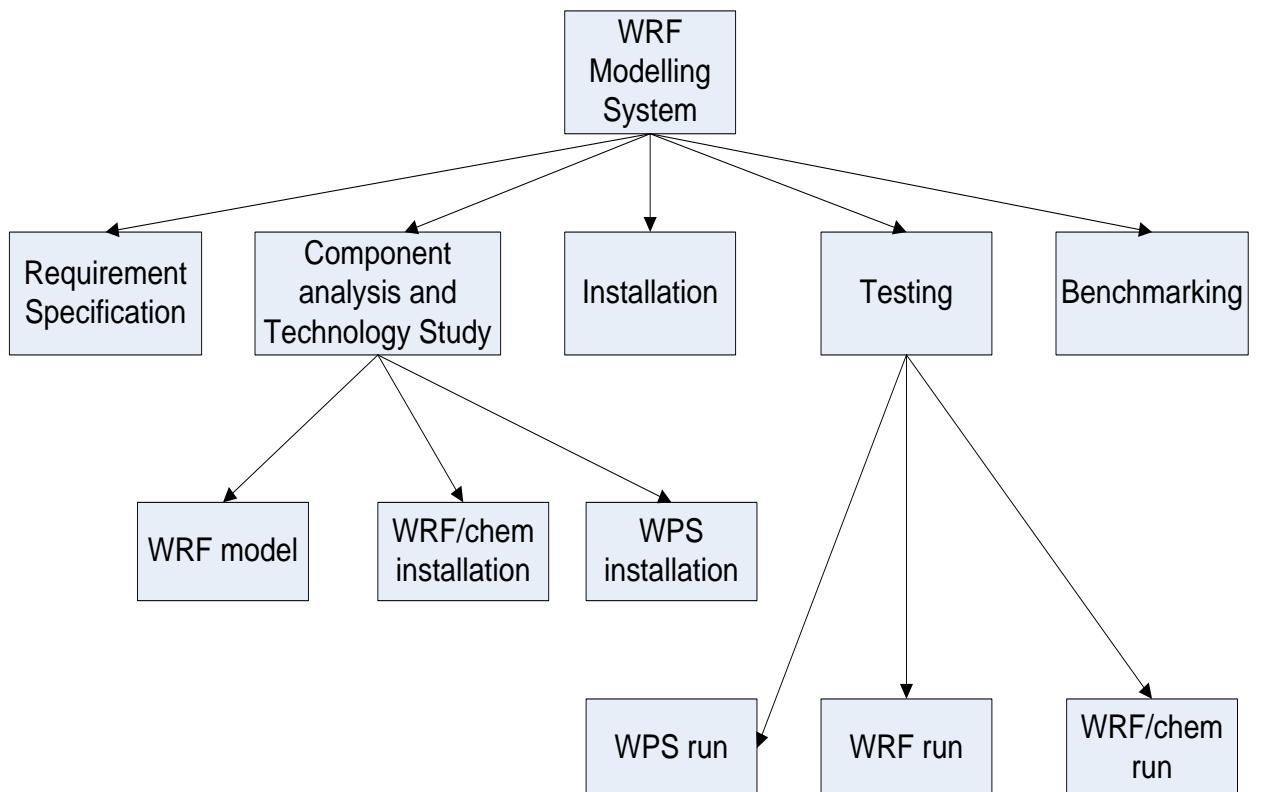


Figure 2.3.1: Work Breakdown Structure of WRF Modeling System

3.

LITERATURE SURVEY.

3.1 Cluster Components.

3.2 Benefits of Cluster Computing.

3.3 HPC Cluster.

3.4 PRL HPC Cluster.

3.5 Study of WRF-CHEM model and its flow architecture.

3.6 Diagrams.

3.1 Cluster Components.

Clustering is a process of grouping objects with similar properties. Any cluster should exhibit two main properties; low inter-class similarity and high intra-class similarity. Clustering is an unsupervised learning i.e. it learns by observation rather than examples. There are no predefined class label exists for the data points [14]. Clustering is a fundamental operation in data mining. A cluster is a group of data objects that are similar to one another within the same cluster and are dissimilar to the objects in other clusters. A good clustering algorithm is able to identify clusters irrespective of their shapes. Following figure 3.1 shows the stages of clustering process [2].

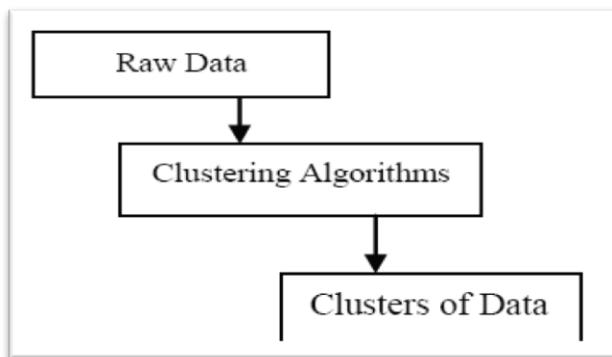


Figure 3.1 Stages of Clustering Process

Cluster components are listed as: Cluster Software, Cluster Hardware, Cluster Network, and cluster Storage.

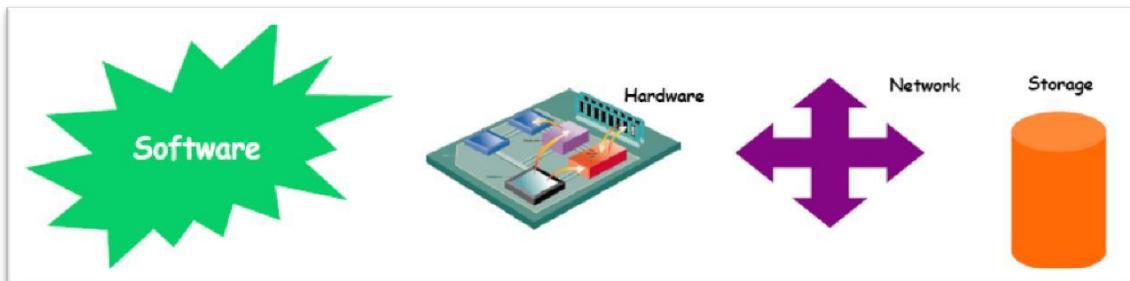


Figure 3.2 Cluster Components

- **Cluster Software:** Cluster compatible Operating system is required, Cluster software to generate and maintain cluster, Cluster aware application.
- **Hardware:** Uses commodity components, Advanced Processors with large cache, High speed memory, Advance chipsets, Faster I/O subsystem.

- **Network:** High speed, Low latency, Scalable, reliable, Accelerators.
- **Storage:** Linear scaling, Extreme bandwidth & I/O, Hierarchical storage management, single storage pool, Reliable.

3.2 Benefits of Cluster computing.

As cluster has many standard benefits its popularity increasing day by day. The clusters are also utilized to host many new internet service sites [10, 11]. In the commercial arena, servers can be consolidated to create an enterprise server that can be optimized, tuned and managed for increased efficiency and responsiveness depending on the workload through load balancing [13-24]. A large number of machines can be clustered along with storage and application for efficient performance. These are various measures of performance of clusters. However, the most important three parameters that need to be analyzed for high performance computing are **High Availability, Load Balancing and Fault Tolerance** [7, 8].

3.3 HPC Cluster.

Min Li et al. Presented HPC cluster monitoring and its different functionalities such as Job monitoring, System monitoring etc. They have discussed, the user allows to build HPC cluster computing monitoring environment on their demand and can customize based on needs.

High Performance Computing which includes computers, networks, algorithms and environments to make such system usable. High-performance computing (HPC) is a broad term that at its core represents compute intensive applications that need acceleration. Users of application acceleration systems range from medical imaging, financial trading, oil and gas expiration, to bioscience, data warehousing, data security, and many more. In the information age, the need for acceleration of data processing is growing exponentially and the markets deploying HPC for their applications are growing every day. The HPC expansion is being fueled by the coprocessor, which is fundamental to the future of HPC.

HPC allows scientists and engineers to solve complex science, engineering and business problems using applications that require high bandwidth, low latency networking, and very high compute capabilities. Typically, scientists and engineers

must wait in long queues to access shared clusters or acquire expensive hardware systems.

- HPC CLUSTER minimum Requirements.

| Minimum requirements | Master Node/ Slave Node | Network |
|--|---|----------------------------|
| Head /Master node, Compute/Slave node, Cluster software, Cluster interconnect and Storage | 1 Ethernet Port CDROM Drive 256MB RAM | Crossover network Cable |

Table 3.1 HPC Cluster Minimum Requirements.

3.4 PRL HPC Cluster.

Physical Research Laboratory (PRL) is a national Research institute for space and science, supported mainly by Department of Space, Government of India. PRL carries out fundamental research in select areas of Physics, Space & Atmospheric Sciences, Astronomy, Astrophysics & Solar Physics, and Planetary & Geosciences. PRL Uses HPC cluster for space and atmospheric science division.

PRL HPC Cluster Architecture: It is a 21 node cluster with 20 compute nodes and 1 master node with a peak performance of 3.2TF and a sustained performance of 2.2TF (approx.). It supports 64-bit Hardware and 32/64 bit software. It has different types of nodes like Backup Node, I/O Node, a Storage Node and a management node. The cluster is having a 10TB of usable storage based on FC disk drives (minimum 10k rpm). It has 20TB of raw storage with LTO Gen 4 Tape Library for Data Backup. The primary Network is Infiniband and the Secondary Network is Gigabit. There is an additional Management Switch for Node Management using intelligent platform Management Interface (IPMI). PRL HPC cluster architecture at given figure 3.3

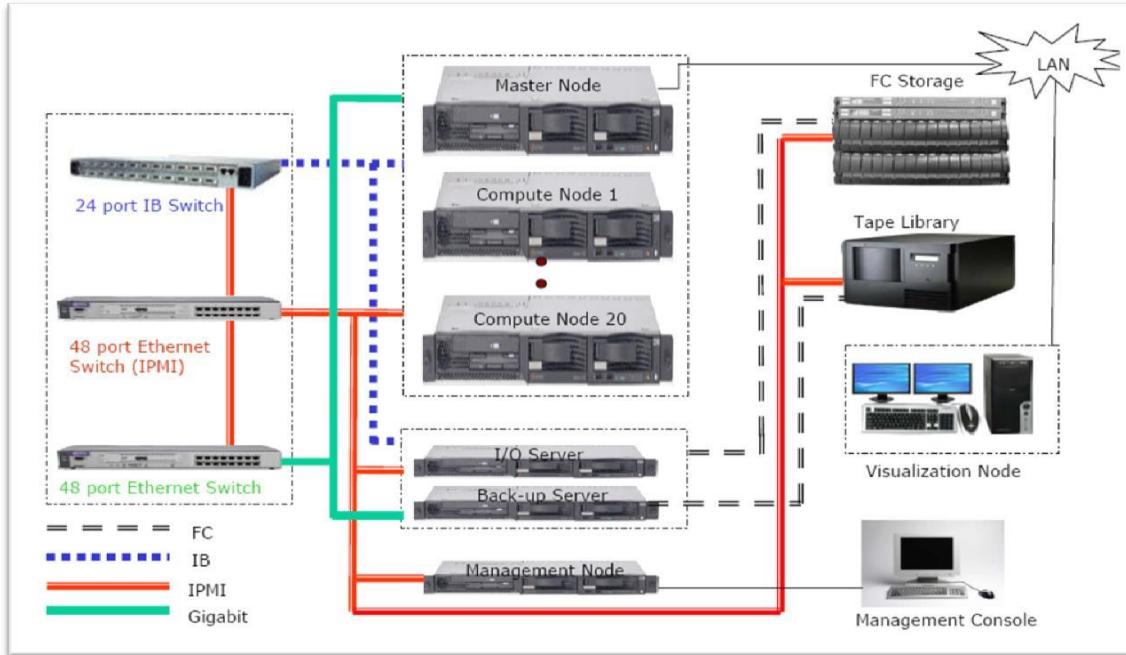


Figure 3.3 PRL HPC Cluster Architecture [6].

The Master Node and all compute Nodes (20 Nos.), are of the make HP DL 585G5.

- **Master Node:** It is having Quad core and Quad socket AMD Opteron 8360SE, 2.5GHz processor, 64 GB memory capacity and a capacity of 4*146GB SAS.
- **Computer Nodes:** All the 20 compute nodes are having Quad core and Quad socket AMD Opteron 8360SE, 2.5GHz processor, 64 GB memory capacity and Hard disc capacity of 2*73GB SAS.

The Master node and all Compute nodes are installed with Red Hat Linux Enterprise Linux 5.1(2.6.18-53.e15) as operating System with Rock 5.1 as cluster management Tool.

- **Storage Node:** It is having Quad core Dual Socket Xeon E5420, 2.5 GHz processor, 8GB RAM, 4*72GB10K SAS.
- **Backup Node:** It is having Quad core Dual Socket Xenon E5420, 2.5 GHz processor, 8GB RAM, 2*120GB SATA.
- **Disk Array:** The EVA 4400 Disk Array is having 4 Disk Enclosures, 42*422GB 10K RPM FC Disks, total capacity of 16 TB (approx.) and a usable storage of 10TB.
- **IPMI Management Node:** Quad Core Dual Socket AMD Opteron2360,

2.5GHz processor, 4GB RAM, 2*160GB SATA.

- Software used on PRL HPC Cluster: Intel C,C++,Fortran, GNU C,C++,Fortran, Parallel Compiler Suites-INTEL MPI and OpenMP Compilers, Intel Profiler and Analyser and Intel Debugger Suite, Torque Scheduler Suite.

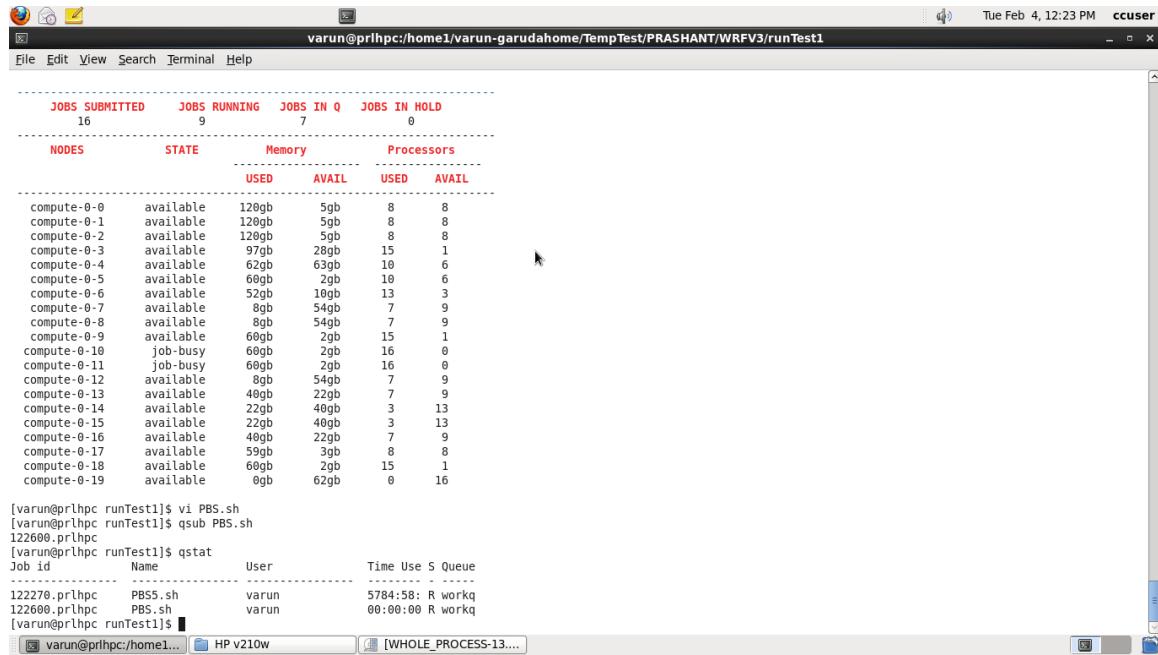


Fig 3.4 PRL HPC Cluster View.

3.5 Study of WRF-CHEM model and its flow architecture.

The WRF-CHEM model system consists of following three major programs with are shown in Fig. 1:

The WRF-CHEM Difference:

- The difference with regular WRF comes from **the chemistry part** of the model needing to be provided additional gridded input data related to emissions.
- This additional input data is **provided either by the WPS (dust emission fields), or read in during the real.exe initialization** (e.g., biomass burning, biogenic emissions, GOCART background fields, etc.), or read in during the execution of the WRF solver (e.g., anthropogenic emissions, boundary conditions, volcanic emissions, etc.). And while some programs are provided in an attempt to aid the user in generation

of these external input data files, as stated earlier, not all emissions choices are set-up to function for all possible namelist options related to the WRF-CHEM model [2].

- WRF-CHEM forecast model is very complex model. Users of this model will need to change and set the environmental variables and available code as well as its standard configuration and compilations, depending on the various specifications of Linux systems to get it to function properly for their project.

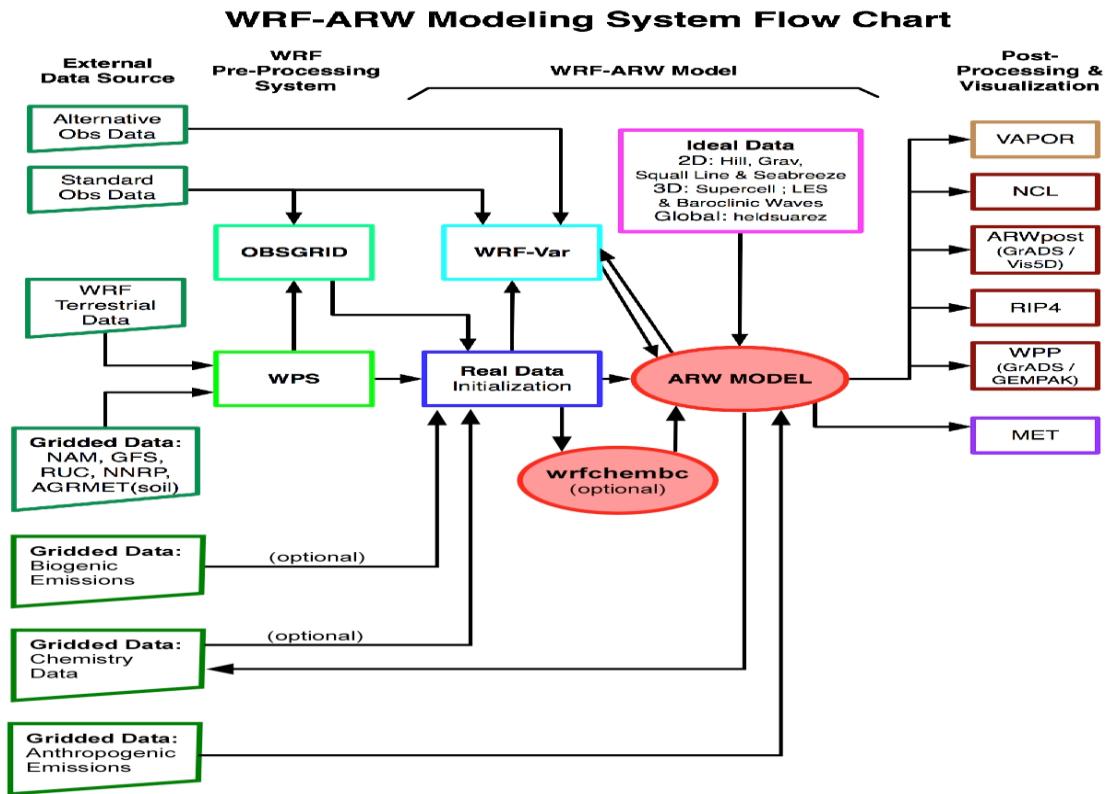


Figure 3.5 WRF-CHEM modeling system architecture

- WRF-CHEM model requires lots of memory computations and nodes for its additional variables settings.
- For that this model is deployed only in HPC computing environment. Also for its configuration and compilation process, distributed parallel memory option is selected to make it run.

3.6 Diagrams.

3.6.1 Use Case Diagrams.

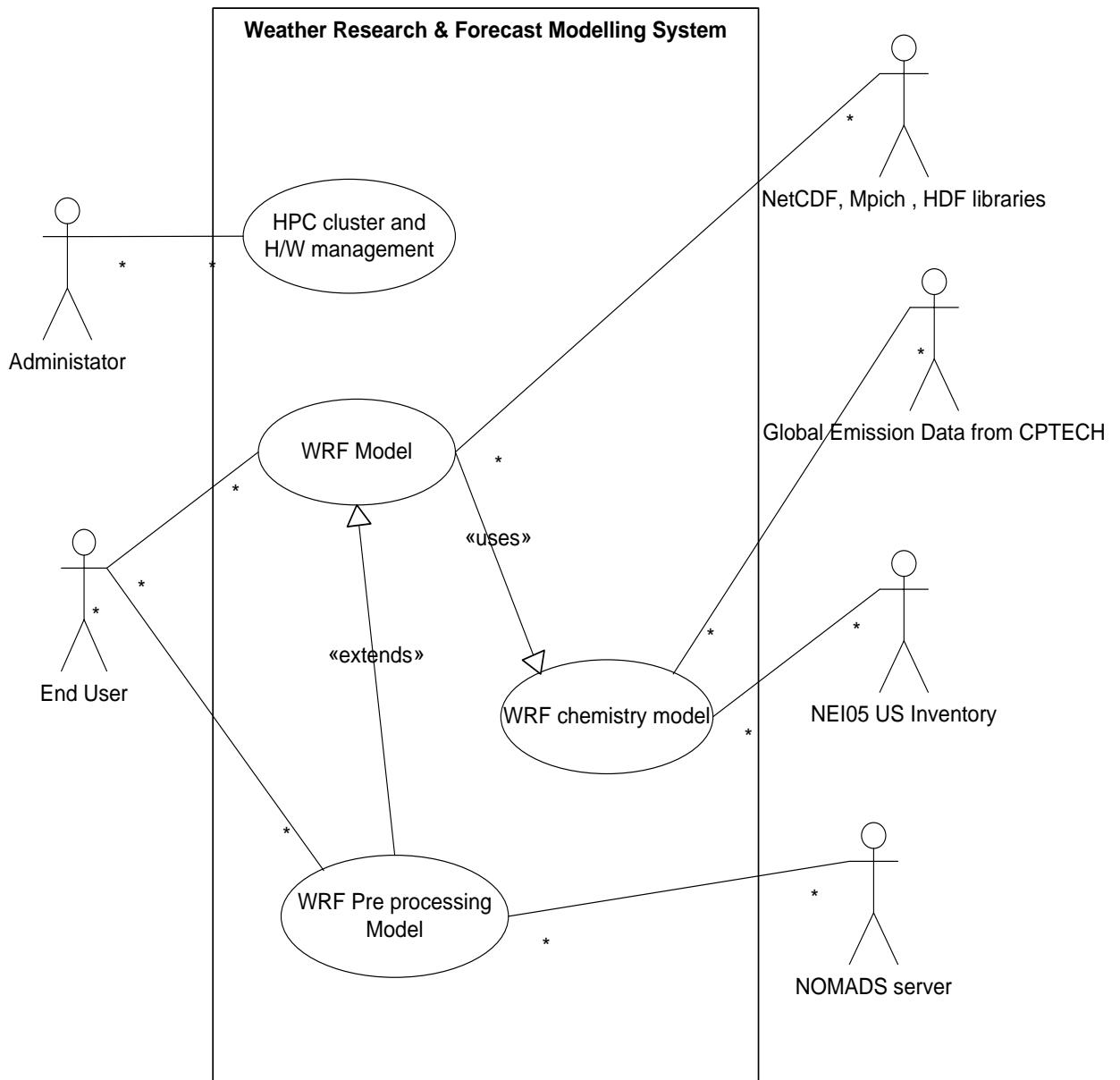


Figure 3.6.1: Usecase Diagram.

3.6.2 Class Diagram.

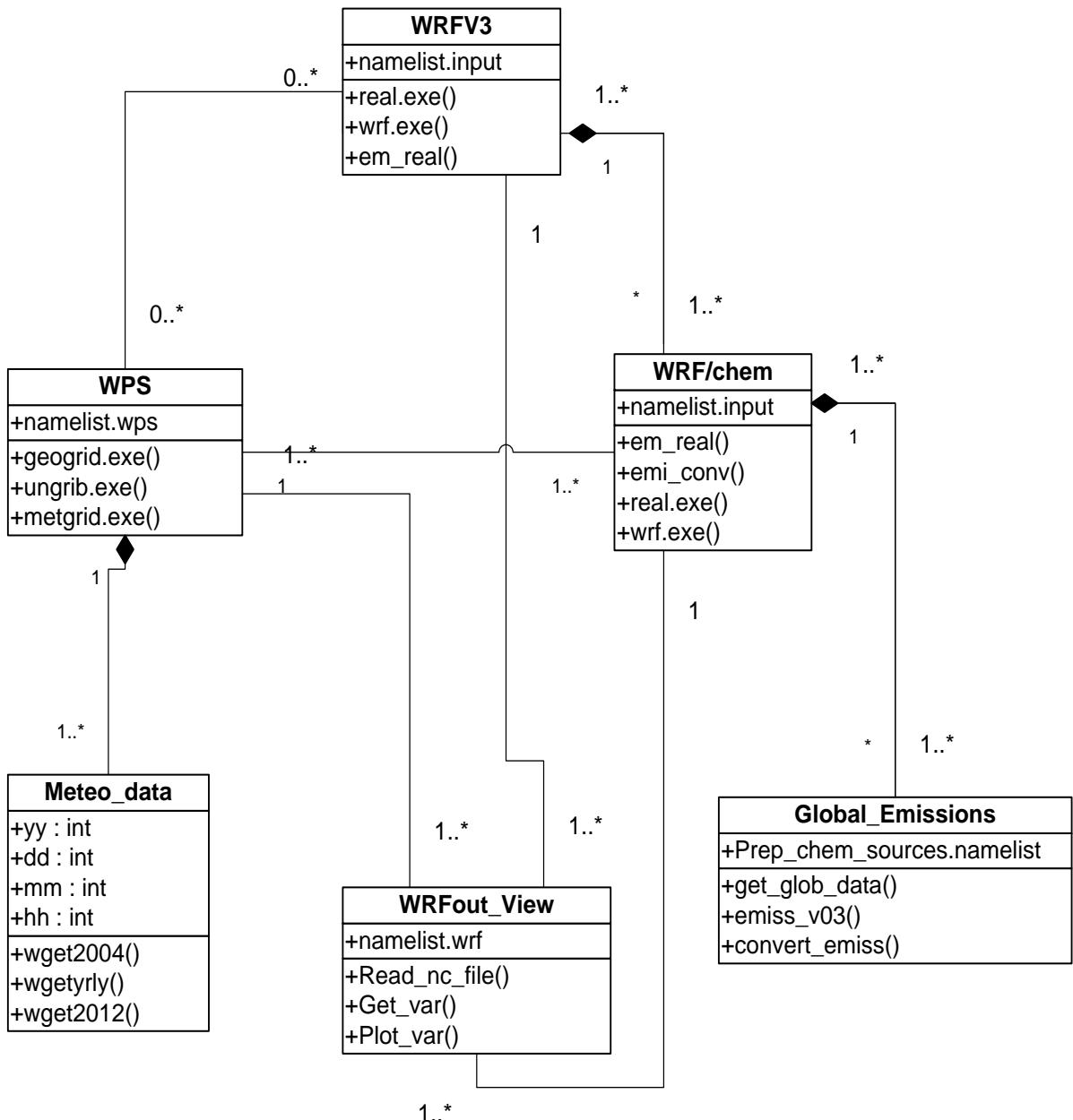


Figure 3.6.2 : Class Diagram.

3.6.3 sequence Diagram:

➤ Sequence Diagram For Downloading Data From Server:

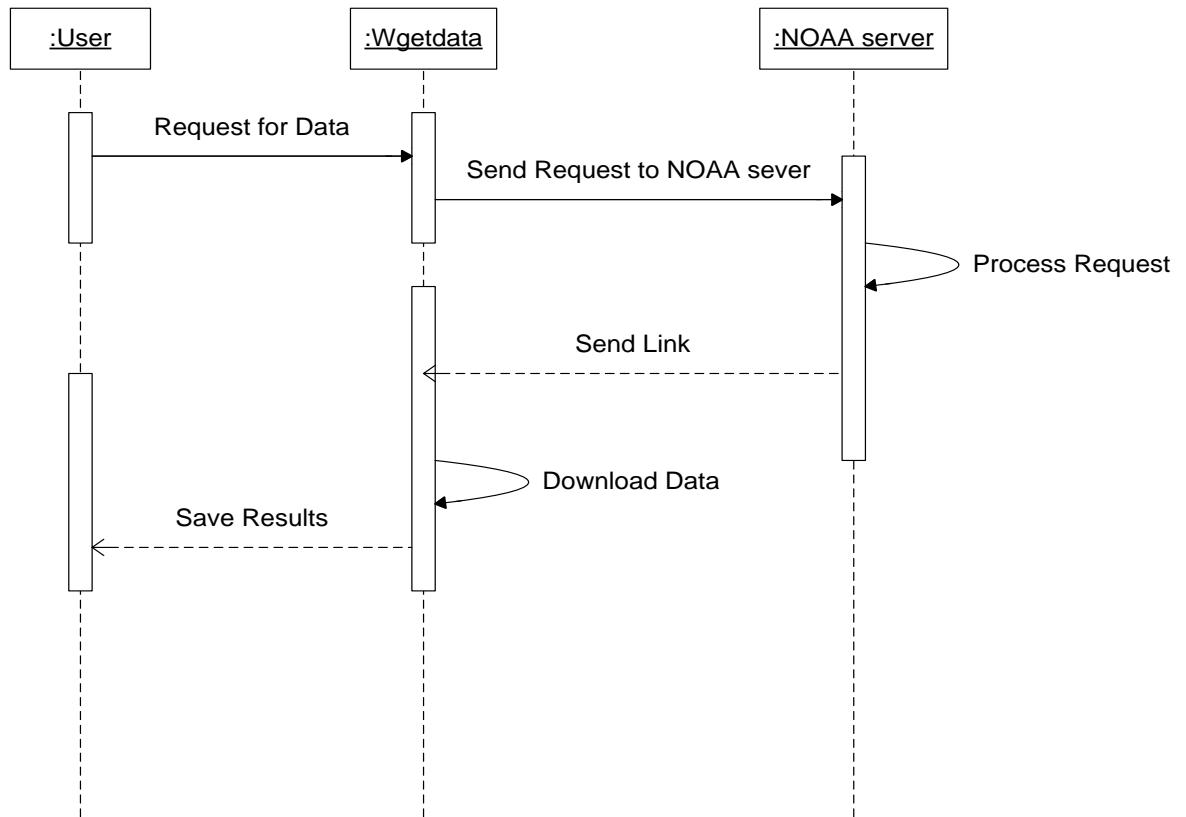


Figure 3.6.3: Sequence Diagram for Downloading Data.

➤ Sequence Diagram For WRF Pre Processing system:

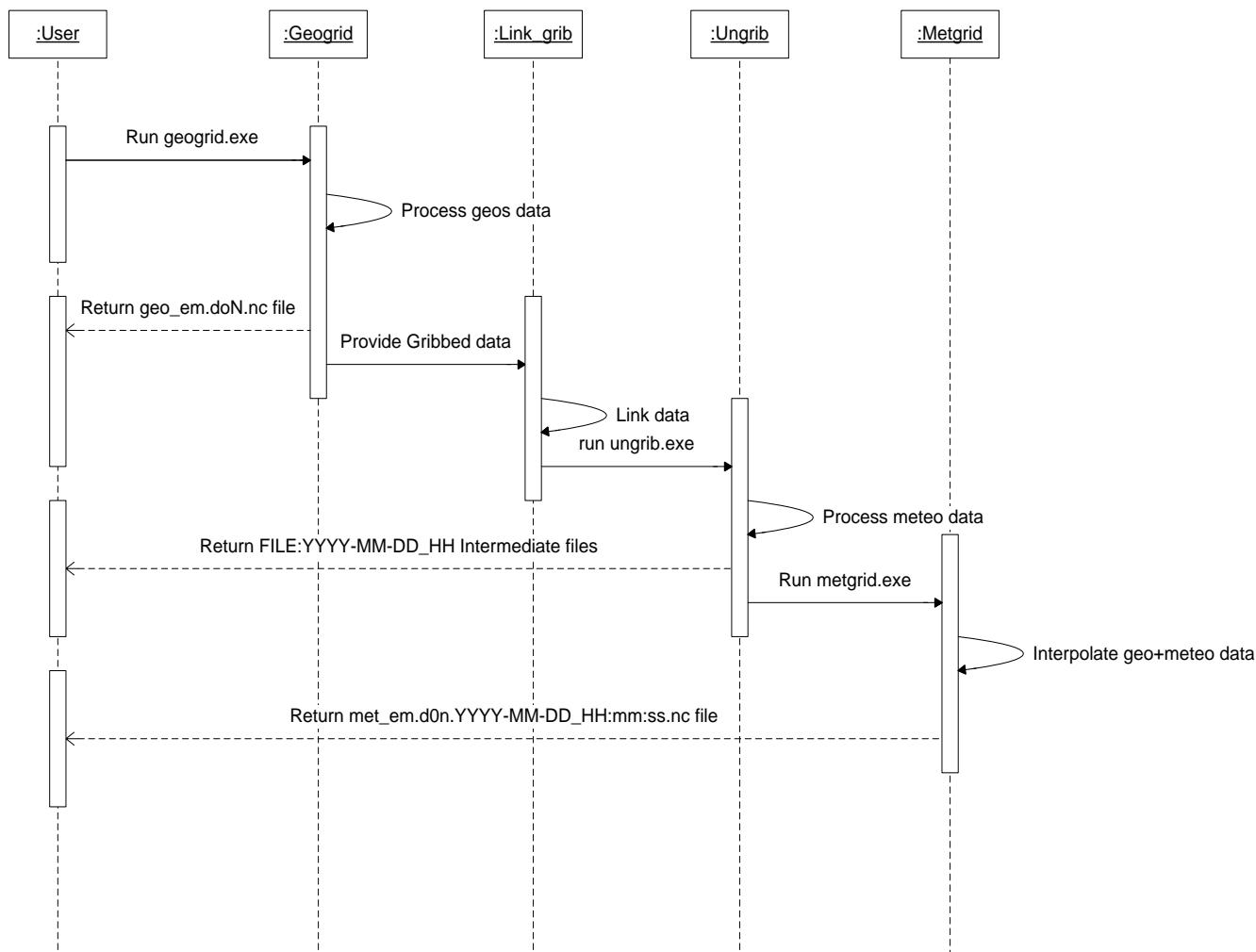


Figure 3.6.4 : Sequence Diagram For WPS

➤ Sequence Diagram for Chemistry coupled WRF model:

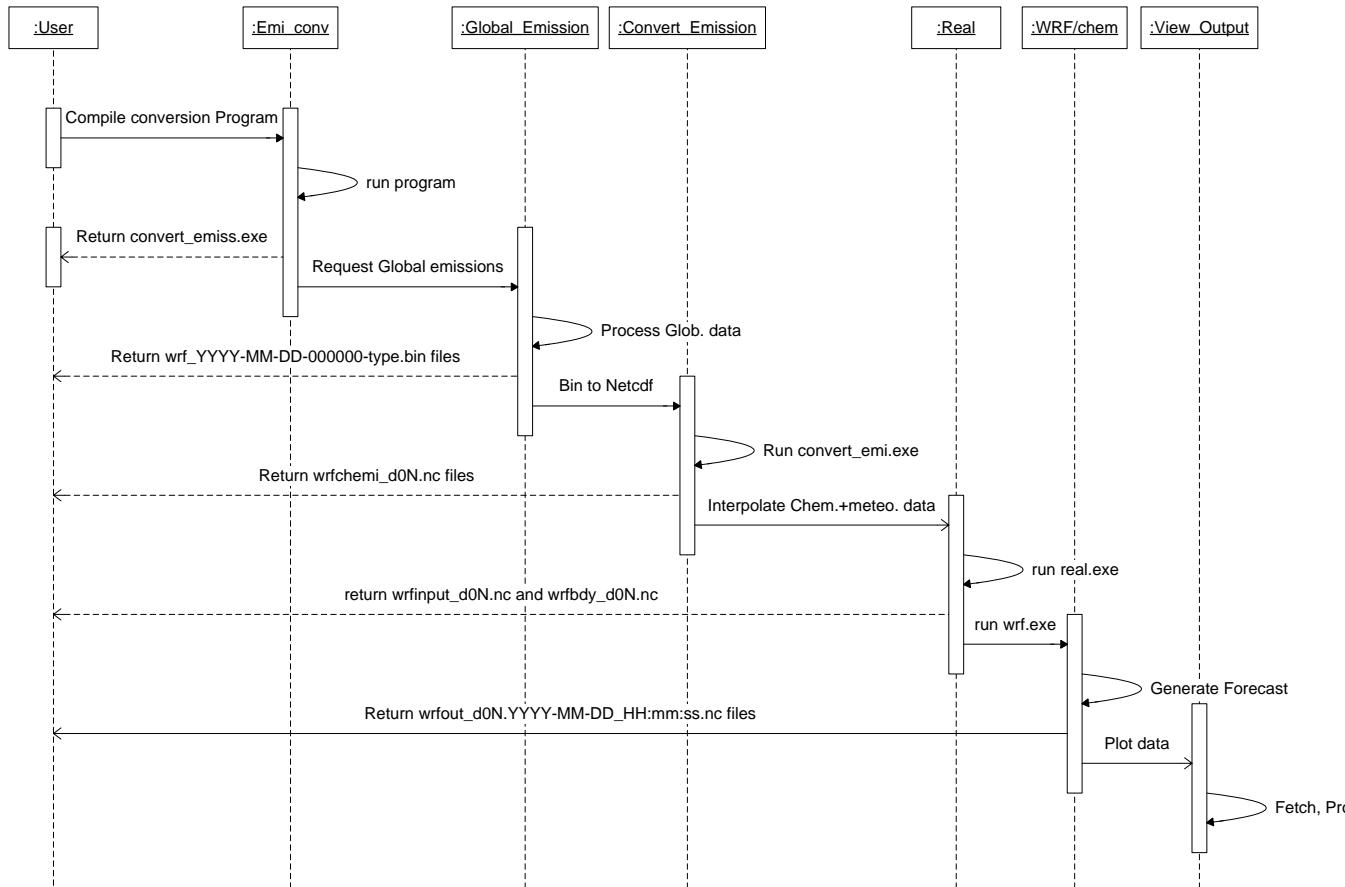


Figure 3.6.5 : Sequence Diagram for WRF/Chem.

4.

PROPOSED WORK.

4.1 Operational Running Flow of WRF Model .

4.2 Algorithmic Steps.

4.1 Operational Running Flow of WRF Model .

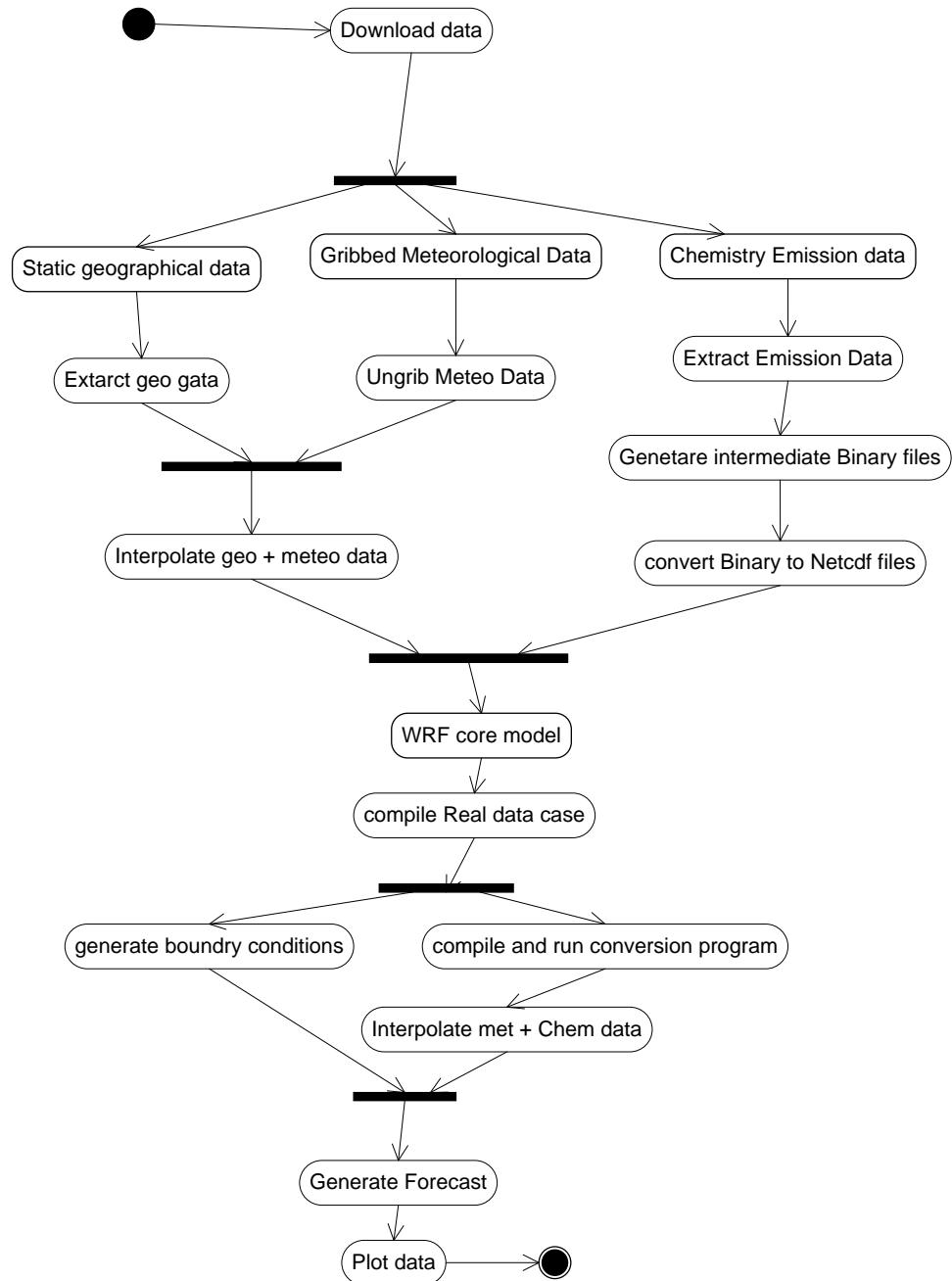


Figure 4.1 Operational Running Flow of WRF Model.

4.2 Algorithmic Steps.

Part A

Step 1: Set up the initial environmental variables and add correct libraries paths settings in bash file.

Step 2: Prepare the input data and make ready namelist file, which are given as input to proposed Operational WRF model.

Step 3: Pre-process the input data files which are in our case mostly in netCDF/GRIB files format, which are not in human-readable format they are in binary forms.

Step 4: Operational running of Proposed WRF model on PRL HPC cluster. If any bugs occur at the simulation time then

Find & correct it Go to step 4 repeat until final output file will not generated.

Step 5: If the WRF-CHEM model will run successfully the Benchmark the WRF-CHEM model result.

Part B

Step 6: Read the output files which are also in netCDF/GRIB format.

Step 7: Plot the output in well-mannered format on the bases of needs for scientific analysis purpose using any available post-processing tools like ncView, NCL, Matlab etc..

Step 8: If (output is in correct format) then

Show it in any image file format

ELSE

Make changes in program which is written for plotting the output

Step 9: Analyze and test the produced output from Operational model.

Step 10: calculate minimum time, accuracy & scalability of operational process. if the operational process runs comparatively in decided time then

Perform strong testing and analysis again and again for steady output.

5.

IMPLEMENTATION WORK.

5.1 WRF-CHEM Model Setup & Configuration.

5.2 bash file Path settings.

5.3 Configuration & Compilation of WRF-CHEM Model.

5.4 WPS Pre-processing Configuration & Compilation.

5.5 Steps of WRF-CHEM Simulation Process.

5.1 WRF-CHEM Model Setup & Configuration.

WRF USERS PAGE

Welcome to the WRF Modeling System Download page. This page provides links to the latest releases of the WRF modeling system, including source code, documentation, and graphics software. The WRF system is a public domain modeling system that is freely available for use. It is designed to be used for a wide range of applications, from simple idealized simulations to complex global applications. The WRF system is currently being used for a variety of research projects, including climate research, data assimilation research, forecast research, and more. The WRF system is also being used for a variety of applications, including mesoscale and microscale meteorology, and coupled-model applications.

WRF SOURCE CODE DOWNLOAD: RETURNING USERS

Returning users: please fill in the E-mail Address Field and click on the "Submit" button below.

E-Mail Address: shuktatoramp@gmail.com

WRF SOURCE CODES AND GRAPHICS SOFTWARE DOWNLOAD PAGE

For quick navigation, click any link in the below boxes:

- Version 3.5.1
- WRF-ARW
- WPS
- WRF-DA
- WRF-Chem
- Post Processing
- Utilities
- Testing

This page provides downloads for the latest release of Versions 3.5, 3.4, 3.3, and 3.2 (for previous releases click [here](#)).

If you are a first-time WRF user, learn how to run the programs via the [online tutorial](#).

Please read this [special note](#) if you are using NCAR's Yellowstone.

WRF Downloads for the most recent version: 3.5.1

| WRF-ARW | tar file | Known Problems | Updates |
|----------|--------------------------|----------------|---------|
| WPS | tar file | Known Problems | Updates |
| WRF-DA | tar file | Known Problems | Updates |
| WRF-Chem | tar file | Known Problems | Updates |

WRF-ARW Code Downloads

| Version | Date | tar file | Known Problems | Updates |
|---------------|--------------------|--------------------------|----------------|---------|
| Version 3.5.1 | September 23, 2013 | tar file | Known Problems | Updates |
| Version 3.5 | April 18, 2013 | tar file | Known Problems | Updates |
| Version 3.4.1 | August 16, 2012 | tar file | Known Problems | Updates |
| Version 3.4 | April 6, 2012 | tar file | Known Problems | Updates |

WRF-ARW Code Downloads

| WRF-ARW | tar file | Known Problems | Updates |
|----------|--------------------------|----------------|---------|
| WPS | tar file | Known Problems | Updates |
| WRF-DA | tar file | Known Problems | Updates |
| WRF-Chem | tar file | Known Problems | Updates |

WRF Modeling System Download - Mozilla Firefox

```
[varun@rlhpc PRASHANT]$ ls
WRFV3_WRFV3.4.TAR WRFV3-Chem-3.4.TAR
[varun@rlhpc PRASHANT]$ rm -rf *
[1]+ Done ./.compile_em real >print.out (wd: /home1/varun-garudahome/TempTest/PRASHANT)
[wd now: /home1/varun-garudahome/TempTest/PRASHANT]
[varun@rlhpc PRASHANT]$ rm -rf *
[varun@rlhpc PRASHANT]$ ls
[varun@rlhpc PRASHANT]$ ls -al
total 8
drwxrwxr-x 2 varun varun 3064 Jan 15 2014 .
drwxrwxr-x 8 varun varun 3064 Jan 15 2014 ..
[varun@rlhpc PRASHANT]$ wget http://www.mmm.ucar.edu/wrf/src/WRFV3.5.1.TAR.gz
--15:43:45-- http://www.mmm.ucar.edu/wrf/src/WRFV3.5.1.TAR.gz
Connecting to 172.16.0.1:3128... connected.
Proxy request sent, awaiting response... 200 OK
Length: 38997831 (37M) [application/x-gzip]
Saving to: 'WRFV3.5.1.TAR.gz'

0% [=====] 2,1597 13.3K/s
```

WRF Modeling System Download - Mozilla Firefox

```
[varun@rlhpc PRASHANT]$ ls
WRFV3_WRFV3.4.TAR WRFV3-Chem-3.4.TAR
[varun@rlhpc PRASHANT]$ rm -rf *
[1]+ Done ./.compile_em real >print.out (wd: /home1/varun-garudahome/TempTest/PRASHANT)
[wd now: /home1/varun-garudahome/TempTest/PRASHANT]
[varun@rlhpc PRASHANT]$ rm -rf *
[varun@rlhpc PRASHANT]$ ls
[varun@rlhpc PRASHANT]$ ls -al
total 8
drwxrwxr-x 2 varun varun 3064 Jan 15 2014 .
drwxrwxr-x 8 varun varun 3064 Jan 15 2014 ..
[varun@rlhpc PRASHANT]$ wget http://www.mmm.ucar.edu/wrf/src/WRFV3.5.1.TAR.gz
--15:43:45-- http://www.mmm.ucar.edu/wrf/src/WRFV3.5.1.TAR.gz
Connecting to 172.16.0.1:3128... connected.
Proxy request sent, awaiting response... 200 OK
Length: 38997831 (37M) [application/x-gzip]
Saving to: 'WRFV3.5.1.TAR.gz'

0% [=====] 2,639,377 92.6K/s eta 1m 22s
```

WRF Modeling System Download - Mozilla Firefox

```
[varun@rlhpc PRASHANT]$ ls
WRFV3_WRFV3.4.TAR WRFV3-Chem-3.4.TAR
[varun@rlhpc PRASHANT]$ rm -rf *
[1]+ Done ./.compile_em real >print.out (wd: /home1/varun-garudahome/TempTest/PRASHANT)
[wd now: /home1/varun-garudahome/TempTest/PRASHANT]
[varun@rlhpc PRASHANT]$ rm -rf *
[varun@rlhpc PRASHANT]$ ls
[varun@rlhpc PRASHANT]$ ls -al
total 8
drwxrwxr-x 2 varun varun 3064 Jan 15 2014 .
drwxrwxr-x 8 varun varun 3064 Jan 15 2014 ..
[varun@rlhpc PRASHANT]$ wget http://www.mmm.ucar.edu/wrf/src/WRFV3.5.1.TAR.gz
--15:44:26-- http://www.mmm.ucar.edu/wrf/src/WRFV3.5.1.TAR.gz
Connecting to 172.16.0.1:3128... connected.
Proxy request sent, awaiting response... 200 OK
Length: 2708572 (2.6M) [application/x-gzip]
Saving to: 'WRFV3.5.1.TAR.gz'

100% [=====] 2,708,572 43.6K/s in 45s
15:45:11 (59.2 KB/s) - 'WRFV3.5.1.TAR.gz' saved [2708572/2708572]
```

WRF Modeling System Download - Mozilla Firefox

```
[varun@rlhpc PRASHANT]$ ls
WRFV3.5.1.TAR.gz WRFV3-Chem-3.5.1.TAR.gz
```

```

varun@prlhpc:/home1/varun-garudahome/TmpTest/PRASHANT
File Edit View Search Terminal Help
Saving to: 'WRFV3.5.1.TAR.gz'

50% [=====] 19,830,889 --.-K/s eta 8m 55s
[varun@prlhpc PRASHANT]$ wget -c http://www.mmm.ucar.edu/wrf/src/WRFV3.5.1.TAR.gz
-:15:33:09-- http://www.mmm.ucar.edu/wrf/src/WRFV3.5.1.TAR.gz
Connecting to 172.16.0.1:3128... connected.
Proxy request sent, awaiting response... 200 OK
Length: 38997831 (37M) [application/x-gzip]
-:15:33:10-- (try: 2) http://www.mmm.ucar.edu/wrf/src/WRFV3.5.1.TAR.gz
Connecting to 172.16.0.1:3128... connected.
Proxy request sent, awaiting response... 206 Partial Content
Length: 38997831 (37M), 19166941 (18M) remaining [application/x-gzip]
Saving to: 'WRFV3.5.1.TAR.gz'

100%[=====] 38,997,831 90.5K/s in 5m 54s
15:59:04 (52.9 kB/s) - 'WRFV3.5.1.TAR.gz' saved [38997831/38997831]

[varun@prlhpc PRASHANT]$ tar -zxvf WRFV3.5.1.TAR.gz
varun@prlhpc:/home1/varun-garudahome/TmpTest/PRASHANT$ ls
WRFV3/Registry/Registry.NMM_NEST
WRFV3/Registry/Registry.wrfvar
WRFV3/Registry/registry.avgflx
WRFV3/Registry/registry.bdy_perturb
WRFV3/Registry/registry.cam
WRFV3/Registry/registry.chem
WRFV3/Registry/registry.clm
WRFV3/Registry/registry.diags
WRFV3/Registry/registry.dimspec
WRFV3/Registry/registry.fire
WRFV3/Registry/registry.io_boilerplate
WRFV3/Registry/registry.les
WRFV3/Registry/registry.ssib
WRFV3/Registry/registry.stoch
WRFV3/Registry/registry.var
WRFV3/Registry/registry.var_chem
[varun@prlhpc PRASHANT]$ ls
WRFV3 WRFV3.5.1.TAR.gz WRFV3-Chem-3.5.1.TAR.gz
varun@prlhpc PRASHANT$ mv WRFV3-Chem-3.5.1.TAR.gz WRFV3
[varun@prlhpc PRASHANT]$ cd WRFV3
[varun@prlhpc WRFV3]$ tar -zxvf WRFV3-Chem-3.5.1.TAR.gz

varun@prlhpc:/home1/varun-garudahome/TmpTest/PRASHANT/WRFV3
File Edit View Search Terminal Help
WRFV3/Registry/registry.avgflx
WRFV3/Registry/registry.bdy_perturb
WRFV3/Registry/registry.cam
WRFV3/Registry/registry.chem
WRFV3/Registry/registry.clm
WRFV3/Registry/registry.diags
WRFV3/Registry/registry.dimspec
WRFV3/Registry/registry.fire
WRFV3/Registry/registry.io_boilerplate
WRFV3/Registry/registry.les
WRFV3/Registry/registry.ssib
WRFV3/Registry/registry.stoch
WRFV3/Registry/registry.var
WRFV3/Registry/registry.var_chem
[varun@prlhpc PRASHANT]$ ls
WRFV3 WRFV3.5.1.TAR.gz WRFV3-Chem-3.5.1.TAR.gz
[varun@prlhpc PRASHANT]$ mv WRFV3-Chem-3.5.1.TAR.gz WRFV3
[varun@prlhpc PRASHANT]$ cd WRFV3
[varun@prlhpc WRFV3]$ tar -zxvf WRFV3-Chem-3.5.1.TAR.gz

varun@prlhpc:/home1/varun-garudahome/TmpTest/PRASHANT/WRFV3
File Edit View Search Terminal Help
chem/KPP/inc/cbm4/kpp_mechd_b_cbm4.inc
chem/KPP/inc/cbm4/kpp_mechd_e_cbm4.inc
chem/KPP/inc/cbm4/kpp_mechd_ia_cbm4.inc
chem/KPP/inc/cbm4/kpp_mechd_ib_cbm4.inc
chem/KPP/inc/cbm4/kpp_mechd_ibu_cbm4.inc
chem/KPP/inc/cbm4/kpp_mechd_l_cbm4.inc
chem/KPP/inc/cbm4/kpp_mechd_u_cbm4.inc
chem/KPP/documentation/gpl/
chem/KPP/documentation/latex/
chem/KPP/documentation/wkc_kpp.txt
chem/KPP/documentation/latex/figs/
chem/KPP/documentation/gpl/gpl_kpp.txt
chem/KPP/documentation/gpl/gpl_wkc.txt
[varun@prlhpc WRFV3]$ ls
arch configure external main README.DA README.rsl_output Registry tools
chem dyn_em frame Makefile README.hydro README.SSIB run WRFV3-Chem-3.5.1.TAR.gz
clean dyn_exp hydro phys README.io_config README_test_cases share
compile dyn_nmm inc README README.NMM README.windturbine test
[varun@prlhpc WRFV3]$ 
```

Figure 5.1 WRF-CHEM Model Setup & Configuration.

5.2 Bash file Path settings.

```
varun@prlhpc:~ ls
varun@prlhpc:~ which ncdump
varun@prlhpc:~ vi .cshrc
```

```
# .bashrc

# Source global definitions
if [ -f /etc/bashrc ]; then
    . /etc/bashrc
fi

# User specific aliases and functions

export PATH=/rhome/intel/impi/3.2.1.009/bin64:/rhome/intel/impi/3.2.1.009/lib64:/rhome/intel/impi/3.2.1.009/include64:/home1/varun-garudahome/WRF/LIB/flex-2.5.3/lib:/home1/varun-garudahome/WRF/LIB/flex-2.5.3/bin:/home1/varun-garudahome/WRF/LIB/flex-2.5.3/include:$PATH
export MPI_HOME=/rhome/intel/impi/3.2.1.009
export NETCDF=/share/apps/INTEL_NETCDF.4
export FLEX_LIB_DIR=/home1/varun-garudahome/WRF/LIB/flex-2.5.3/lib
export FLEX=/home1/varun-garudahome/WRF/LIB/flex-2.5.3
export YACC=/home1/varun-garudahome/WRF/LIB/YACC/byacc-20120115/bin/yacc -d
export WRF_EM_CORE=1
".bashrc" 26L, 1019C 15,1 Top
```

```
NETCDF=/share/apps/INTEL_NETCDF.4
PFDIR=/export/apps/lf9562/bin
LOGNAME=varun
CVS_RSH=ssh
CLASSPATH=/usr/java/latest/lib/tools.jar:
SSH_CONNECTION=172.16.15.72 41806 172.16.24.91 22
MPI_HOME=/rhome/intel/impi/3.2.1.009
OMP_NUM_THREADS=24
LESSOPEN=| /usr/bin/lesspipe.sh %
BLASTMAT=/opt/Bio/ncbi/data
WRF_EM_CORE=1
HISTTIMEFORMAT=%F %T
G_BROKEN_FILERAMES=1
I_MPI_ROOT=/rhome/intel/impi/4.0.3.008
_=bin/env
OLDPWD=/home1/varun-garudahome/TmpTest/PRASHANT
[varun@prlhpc WRFV3]$ cd
[varun@prlhpc ~]$ vi .bashrc
[varun@prlhpc ~]$ source .bashrc
```

Figure 5.2 bash file Path settings.

5.3 Configuration & Compilation of WRF-CHEM Model.

```
varun@prlhpc:/home1/varun-garudahome/TmpTest/PRASHANT/WRFV3
File Edit View Search Terminal Help
Contact No.: 4035 / 4036 / 4042
(3) Kindly change your passwd after the first login

NEW:Before submitting the job, Kindly specify memory limit (Maximum 60gb) in your PBSPro job submission script.
Kindly visit http://prlhpc.lan.prl.res.in to read PBSPro job submission template

Kindly Use command -- prlhpcstat to get status of cluster resources

[varun@prlhpc ~]$ source .bashrc
[varun@prlhpc ~]$ ulimit
unlimited
[varun@prlhpc ~]$ cd /home1/varun-garudahome/TmpTest/PRASHANT/WRFV3
[varun@prlhpc WRFV3]$ ./configure

varun@prlhpc:/home1/varun-garudahome/TmpTest/PRASHANT/WRFV3
File Edit View Search Terminal Help
1. Linux x86_64 i486 1586 1686 PGI compiler with pgcc YELLOWSTONE (dm+sm)
2. Linux x86_64 i486 1586 1686 PGI compiler with pgcc YELLOWSTONE (dm+sm)
3. Linux x86_64 i486 1586 1686 PGI compiler with pgcc, SGI MPT (serial)
4. Linux x86_64 i486 1586 1686 PGI compiler with pgcc, SGI MPT (dm+sm)
5. Linux x86_64 i486 1586 1686 PGI Compiler with pgcc, SGI MPT (dm+sm)
6. Linux x86_64 i486 1586 1686 PGI Compiler with pgcc, SGI MPT (dm+sm)
7. Linux x86_64 i486 1586 1686 PGI accelerator compiler with gcc (serial)
8. Linux x86_64 i486 1586 1686 PGI accelerator compiler with gcc (dm+sm)
9. Linux x86_64 i486 1586 1686 PGI accelerator compiler with gcc (dm+sm)
10. Linux x86_64 i486 1586 1686 PGI accelerator compiler with gcc, SGI MPT (dm+sm)
11. Linux x86_64 i486 1586 1686 PGI compiler with pgcc, SGI MPT (dm+sm)
12. Linux x86_64 i486 1586 1686 PGI compiler with pgcc, SGI MPT (dm+sm)
13. Linux x86_64 i486 1586 1686 PGI accelerator compiler with gcc (serial)
14. Linux x86_64 i486 1586 1686 PGI accelerator compiler with gcc (dm+sm)
15. Linux x86_64 i486 1586 1686 PGI accelerator compiler with gcc (dm+sm)
16. Linux x86_64 i486 1586 1686 PGI accelerator compiler with gcc (dm+sm)
17. Linux x86_64 i486 1586 1686 ifort compiler with icc (serial)
18. Linux x86_64 i486 1586 1686 ifort compiler with icc (dm+sm)
19. Linux x86_64 i486 1586 1686 ifort compiler with icc (dm+sm)
20. Linux x86_64 i486 1586 1686 ifort compiler with icc (dm+sm)
21. Linux x86_64 i486 1586 1686, Xeon Phi (MIC architecture) ifort compiler with icc (dm+sm)
22. Linux x86_64 i486 1586 1686, Xeon (SNB with AVX mods) ifort compiler with icc (serial)
23. Linux x86_64 i486 1586 1686, Xeon (SNB with AVX mods) ifort compiler with icc (dm+sm)
24. Linux x86_64 i486 1586 1686, Xeon (SNB with AVX mods) ifort compiler with icc (dm+sm)
25. Linux x86_64 i486 1586 1686, Xeon (SNB with AVX mods) ifort compiler with icc (dm+sm)

varun@prlhpc:/home1/varun-garudahome/TmpTest/PRASHANT/WRFV3
File Edit View Search Terminal Help
39. Linux i486 i586 i686 x86_64, PathScale compiler with pathcc (dm+sm)
40. x86_64 Linux, gfortran compiler with gcc (serial)
41. x86_64 Linux, gfortran compiler with gcc (dm+sm)
42. x86_64 Linux, gfortran compiler with gcc (dm+sm)
43. x86_64 Linux, gfortran compiler with gcc (dm+sm)
44. Cray XT CLE/Linux x86_64, PGI compiler with gcc (serial)
45. Cray XT CLE/Linux x86_64, PGI compiler with gcc (dm+sm)
46. Cray XT CLE/Linux x86_64, PGI compiler with gcc (dm+sm)
47. Cray XT CLE/Linux x86_64, PGI compiler with gcc (dm+sm)
48. Cray XE and XC30 CLE/Linux x86_64, Cray CCE compiler (serial)
49. Cray XE and XC30 CLE/Linux x86_64, Cray CCE compiler (dm+sm)
50. Cray XE and XC30 CLE/Linux x86_64, Cray CCE compiler (dm+sm)
51. Cray XE and XC30 CLE/Linux x86_64, Cray CCE compiler (dm+sm)
52. Cray XC30 CLE/Linux x86_64, Xeon ifort compiler (serial)
53. Cray XC30 CLE/Linux x86_64, Xeon ifort compiler (dm+sm)
54. Cray XC30 CLE/Linux x86_64, Xeon ifort compiler (dm+sm)
55. Cray XC30 CLE/Linux x86_64, Xeon ifort compiler (dm+sm)

Enter selection [1-55] : 19

varun@prlhpc:/home1/varun-garudahome/TmpTest/PRASHANT/WRFV3
File Edit View Search Terminal Help
io_only: esmf_time wrfio_nf wrf_ioapi_includes wrfio_grib_share wrfio_grib1 wrfio_int

#####
Settings listed above are written to configure.wrf.
If you wish to change settings, please edit that file.
If you wish to change the default options, edit the file:
    arch/configure_new.defaults

Testing for NetCDF, C and Fortran compiler
This installation of NetCDF is 64-bit
    C compiler is 64-bit
    Fortran compiler is 64-bit
    It will build in 64-bit

[varun@prlhpc WRFV3]$
```

```

varun@prlhpc:/home1/varun-garudahome/TempTest/PRASHANT/WRFV3
File Edit View Search Terminal Help
io_only: esmf_time wrfio_nf wrf_ioapi_includes wrfio_grib_share wrfio_grib1 wrfio_int

#####
-----
Settings listed above are written to configure.wrf.
If you wish to change settings, please edit that file.
If you wish to change the default options, edit the file:
arch/configure_new.defaults

Testing for NetCDF, C and Fortran compiler

This installation of NetCDF is 64-bit
    C compiler is 64-bit
    Fortran compiler is 64-bit
    It will build in 64-bit

[varun@prlhpc WRFV3]$ ./compile em_real >&print.out

```

Figure 5.3 Configuration & Compilation of WRF-CHEM Model

Error in Compilation of WRF-CHEM.

```

varun@prlhpc:/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/chem/KPP/kpp/kpp-2.1
File Edit View Search Terminal Help
14592 pts/30  00:00:00 make
14594 pts/30  00:00:00 sh
14595 pts/30  00:00:00 sh
14596 pts/30  00:00:00 make
14597 pts/30  00:00:00 sh
14599 pts/30  00:00:00 sh
14601 pts/30  00:00:06 registry
14898 pts/30  00:00:00 ps
[1]+ Killed          ./compile em_real >&print.out
[varun@prlhpc WRFV3]$ cd chem/
Display all 295 possibilities? (y or n)
[varun@prlhpc WRFV3]$ cd chem/KPP/
clean_kpp           configure_kpp           documentation/      kpp/
compile_wkc         configure.kpp          inc/               mechanisms/
[varun@prlhpc WRFV3]$ cd chem/KPP/kpp/kpp-2.1/
[varun@prlhpc kpp-2.1]$ ls
bin   cflags.guess  drv     gpl   kpp_compile  Makefile.defs  readme   src     test
cflags doc          examples int  Makefile     models       site-lisp  src.org  util
[varun@prlhpc kpp-2.1]$ vi Makefile
[varun@prlhpc kpp-2.1]$ vi Makefile
varun@prlhpc:/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/chem/KPP/kpp/kpp-2.1
File Edit View Search Terminal Help
14597 pts/30  00:00:00 sh
14599 pts/30  00:00:00 sh
14601 pts/30  00:00:06 registry
14898 pts/30  00:00:00 ps
[1]+ Killed          ./compile em_real >&print.out
[varun@prlhpc WRFV3]$ cd chem/
Display all 295 possibilities? (y or n)
[varun@prlhpc WRFV3]$ cd chem/KPP/
clean_kpp           configure_kpp           documentation/      kpp/
compile_wkc         configure.kpp          inc/               mechanisms/
[varun@prlhpc WRFV3]$ cd chem/KPP/kpp/kpp-2.1/
[varun@prlhpc kpp-2.1]$ ls
bin   cflags.guess  drv     gpl   kpp_compile  Makefile.defs  readme   src     test
cflags doc          examples int  Makefile     models       site-lisp  src.org  util
[varun@prlhpc kpp-2.1]$ vi Makefile.defs
[varun@prlhpc kpp-2.1]$ vi readme
[varun@prlhpc kpp-2.1]$ pwd
/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/chem/KPP/kpp/kpp-2.1
[varun@prlhpc kpp-2.1]$ export KPP_HOME=/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/chem/KPP/kpp/kpp-2.1
[varun@prlhpc kpp-2.1]$ vi Makefile.defs
[varun@prlhpc kpp-2.1]$ make
make[1]: Entering directory '/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/chem/KPP/kpp/kpp-2.1/src'
make[1]: Warning: File `../Makefile.defs' has modification time 2.8e+03 s in the future
  icc -O -l y.tab.o lex.yy.o scanner.o scanutil.o kpp.o gen.o code.o code_c.o code_f77.o code_f90.o code_matlab.o debug.o -L/home1
  /varun-garudahome/WRF/LIB/flex-2.5.3/lib -lfl -ll -o kpp
make[1]: warning: Clock skew detected. Your build may be incomplete.
make[1]: Leaving directory '/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/chem/KPP/kpp/kpp-2.1/src'
[varun@prlhpc kpp-2.1]$ 
```

```

varun@prlhpc:/home1/varun-garudahome/TempTest/PRASHANT/WRFV3
File Edit View Search Terminal Help
Makefile:58: warning: overriding commands for target `c.o'
../../../../configure.wrf:384: warning: ignoring old commands for target `c.o'
make[1]: Warning: File `../../../../configure.wrf' has modification time 2.7e+03 s in the future
icc -o integr_edit.exe integr_edit.o
make[1]: warning: Clock skew detected. Your build may be incomplete.
make[1]: Leaving directory `/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/chem/KPP/util/write_decomp'
integr_edit.exe cbm4
make: execvp: integr edit.exe: Permission denied
make: *** [all] Error 127
=====
mechanisms/cbmz_bb
model cbmz_bb
make: Warning: File `cbmz_bb Integrator.f90' has modification time 2.7e+03 s in the future
/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/chem/KPP/kpp/kpp-2.1/bin/kpp cbmz_bb.kpp

This is KPP-2.1.

KPP is parsing the equation file.

174.17 4%

```

```

varun@prlhpc:/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/ch
File Edit View Search Terminal Help
Makefile:50: warning: overriding commands for target `F.o'
../../../../configure.wrf:357: warning: ignoring old commands for target `F.o'
Makefile:58: warning: overriding commands for target `c.o'
../../../../configure.wrf:384: warning: ignoring old commands for target `c.o'
make[1]: Warning: File `../../../../configure.wrf' has modification time 2.7e+03 s in the future
icc -o integr_edit.exe integr_edit.o
make[1]: warning: Clock Skew detected. Your build may be incomplete.
make[1]: Leaving directory `/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/chem/KPP/util/write_decomp'
integr_edit.exe cbm4
make: execvp: integr edit.exe: Permission denied
make: *** [all] Error 127
=====
mechanisms/cbmz_bb
model cbmz_bb
make: Warning: File `cbmz_bb Integrator.f90' has modification time 2.7e+03 s in the future
/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/chem/KPP/kpp/kpp-2.1/bin/kpp cbmz_bb.kpp

This is KPP-2.1.

KPP is parsing the equation file.
KPP was told to generate WRF conform code
KPP is computing Jacobian sparsity structure.
KPP is starting the code generation.
KPP is initializing the code generation.

KPP is using the WRF conform integrator routine
write_decomp.csh
write_decomp.F
write_decomp.F90
write_decomp.o
[varun@prlhpc write_decomp]$ ls *exe
integr_edit.exe
[varun@prlhpc write_decomp]$ vi Makefile

```

```

varun@prlhpc:/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/ch
File Edit View Search Terminal Help
all:
$(RM) decomp_uses.inc
echo USE $(MECH) Parameters > decomp_uses.inc
echo USE $(MECH) JacobianSP >> decomp_uses.inc
$(LN) ${CPATH}/module_kpp_${(MECH)}_Precision.F
$(LN) ${CPATH}/module_kpp_${(MECH)}_Parameters.F
$(LN) ${CPATH}/module_kpp_${(MECH)}_JacobianSP.F
$(LN) ${CPATH}/module_kpp_${(MECH)}_Integr.F
$(MAKE) comp
./write_decomp.exe
$(MAKE) integr edit
integr_edit.exe $(MECH)
$(MAKE) clean

comp: $(MODULES) $(OBJS)
$(SFC) -o write_decomp.exe $(MODULES) $(OBJS)
echo $(MECH) > mech.tmp

[

integr_edit: integr_edit.o

```

```

OBJS = \
    write_decomp.o

MODULES = \
    module_kpp_${(MECH)}_Precision.o \
    module_kpp_${(MECH)}_Parameters.o \
    module_kpp_${(MECH)}_JacobianSP.o

all:
$(RM) decomp_uses.inc
echo USE $(MECH) Parameters > decomp_uses.inc
echo USE $(MECH) JacobianSP >> decomp_uses.inc
$(LN) ${CPATH}/module_kpp_${(MECH)}_Precision.F
$(LN) ${CPATH}/module_kpp_${(MECH)}_Parameters.F
$(LN) ${CPATH}/module_kpp_${(MECH)}_JacobianSP.F
$(LN) ${CPATH}/module_kpp_${(MECH)}_Integr.F
$(MAKE) comp
./write_decomp.exe
$(MAKE) integr edit
./integr_edit.exe $(MECH)
$(MAKE) clean
-- INSERT --

```

WRF User's Forum > View topic - Compiling WRF/CHEM with KPP

write_decomp.exe Error :

In the chemical mechanism built with kpp, there are some error related with write_decomp.exe not found. You need to edit chem/KPP/util/write_decomp/Makefile like this:

```

integr_edit.exe $(MECH)
# replace it with
integr_edit.exe $(MECH)

```

Jerome

Re: Compiling WRF/CHEM with KPP

On Saturday March 17, 2012 11:06:19 am Jerome wrote:

I have similar problem and the solution was to use version 2.5.3 of flex.

Display posts from previous: All posts Sort by Post time Ascending Go

8 posts • Page 1 of 1

```

varun@prlhpc:/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/chem/KPP
File Edit View Search Terminal Help
[varun@prlhpc write decomps] vi Makefile
[varun@prlhpc write decomps] ls module_kpp__Precision.F
module_kpp__Precision.F
[varun@prlhpc write decomps] vi Makefile
[varun@prlhpc write decomps] cd ../
[varun@prlhpc util] cd ../
[varun@prlhpc KPP] ls
clean_kpp compile_wkc configure_kpp configure.kpp documentation inc kpp mechanisms module_wkppc_constants.F util
[varun@prlhpc KPP] ./compile_wkc
=====
starting compile wkc
/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/chem/KPP/chem/KPP/configure_kpp: Command not found.
ln: creating symbolic link `/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/chem/KPP/kpp/kpp-2.1/Makefile.defs' to `/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/chem/KPP/chem/KPP/kpp-2.1/Makefile.defs': No such file or directory
cp: cannot stat `chem/Makefile': No such file or directory
compile kpp
/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/chem/KPP/chem/KPP/kpp-2.1: No such file or directory.
[varun@prlhpc KPP] ls /home1/varun-garudahome/TempTest/PRASHANT/WRFV3/chem/KPP/kpp/kpp-2.1: No such file or directory
ls: /home1/varun-garudahome/TempTest/PRASHANT/WRFV3/chem/KPP/chem/KPP/kpp/kpp-2.1: No such file or directory
[varun@prlhpc KPP] ls /home1/varun-garudahome/TempTest/PRASHANT/WRFV3/chem/KPP/chem/KPP/kpp/
ls: /home1/varun-garudahome/TempTest/PRASHANT/WRFV3/chem/KPP/chem/KPP/kpp/: No such file or directory
[varun@prlhpc KPP] ls /home1/varun-garudahome/TempTest/PRASHANT/WRFV3/chem/KPP/
chem clean_kpp compile_wkc configure_kpp configure.kpp documentation inc kpp mechanisms module_wkppc_constants.F util
[varun@prlhpc KPP] vi compile_wkc
[varun@prlhpc KPP] vi c 3110618 1:34:27

```



```

varun@prlhpc:/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/chem/KPP
File Edit View Search Terminal Help
#!/bin/csh -f

echo =====
echo starting compile_wkc

setenv WKC_DIRNAME KPP

setenv WRFC_ROOT `pwd`
#setenv WKC_HOME ${WRFC_ROOT}/chem/${WKC_DIRNAME}
setenv WKC_HOME ${WRFC_ROOT}/

# KPP_HOME: environment variable needed by KPP
# note: this is not plain KPP
setenv KPP_HOME ${WKC_HOME}/kpp/kpp-2.1
setenv WKC_KPP ${KPP_HOME}/bin/kpp

#write Makefile.kpp.defs
rm -f ${WKC_HOME}/configure.kpp
${WKC_HOME}/configure_kpp

:wq

```

Screenshot of a terminal window showing compilation errors for WRF-CHEM. The terminal shows multiple command-line sessions with syntax errors and missing files.

```

Model Users Sit... varun@prlhpc:/home1... varun@prlhpc: WRF-CHEM Model Setup and Configuration
[varun@prlhpc KPP] $ pwd
/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/chem/KPP
[varun@prlhpc KPP] $ vi compile_wkc
[varun@prlhpc KPP] $ ./compile_wkc

```

Screenshot of a terminal window showing compilation errors for WRF-CHEM. The terminal shows multiple command-line sessions with syntax errors and missing files.

```

Model Users Sit... varun@prlhpc:/home1... varun@prlhpc: WRF-CHEM Model Setup and Configuration

```

Figure 5.4 Errors in Compilation of WRF-CHEM.

Recompilation of WRF- CHEM Model after Resolving Errors



```
varun@prlhpc:/home1/varun-garudahome/TempTest/PRASHANT/WRFV3
File Edit View Search Terminal Help
=====
starting compile_wkc
/home1/varun-garudahome/TempTest/PRASHANT/WRFV3//configure_kpp: Command not found.
ln: creating symbolic link `/home1/varun-garudahome/TempTest/PRASHANT/WRFV3//kpp/kpp-2.1/Makefile.defs' to `/home1/varun-garudahome/TempTest/PRASHANT/WRFV3//configure.kpp': No such file or directory
compile kpp
/home1/varun-garudahome/TempTest/PRASHANT/WRFV3//kpp/kpp-2.1: No such file or directory.
copying Registry/Registry.EM_CHEM to Registry/Registry

Compiling: WRF_EM_CORE .

setting parallel make -j 2
make: Warning: File `configure.wrf' has modification time 3e+03 s in the future
make -i -r MODULE_DIRS="I../dyn em I./dyn nmm -I/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/external/esmf_time_f90 -I/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/main -I/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/external/io_netcdf -I/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/external/io_int -I/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/frame -I/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/share -I/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/phys -I/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/chem -I/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/inc -I/share/apps/INTEL_NETCDF.4/include " ext
make[1]: Entering directory `/home1/varun-garudahome/TempTest/PRASHANT/WRFV3'
make[1]: Warning: File `configure.wrf' has modification time 3e+03 s in the future
-----
( cd frame ; make -i -r externals )
make[2]: Entering directory `/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/frame'
make[2]: Warning: File `..../configure.wrf' has modification time 3e+03 s in the future
( cd /home1/varun-garudahome/TempTest/PRASHANT/WRFV3/external/esmf_time_f90 ; \
    make -j 2 FC="ifort -i4 -ip -fp-model precise -w -fno-alias -FR -convert big_endian" "RANLIB="ranlib" \
    CPP="/lib/cpp -C -P -I/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/inc -I. -DEM_CORE=1 -DNMM_CORE=0 -DNMM_MAX_DIM=2600 \
    -DCOAMPS_CORE=0 -DDA_CORE=0 -DEXP_CORE=0 -DIWORDSIZE=4 -DDWORDSIZE=8 -DRWORDSIZE=4 -DLWORDSIZE=4 -DNONSTANDARD_SYSTEM_FUNC -DWRF_USE_CLM -DDM_PARALLEL -DNETCDF -DUSE_ALLOCATABLES -DGRIB1 -DINTIO -DLIMIT_ARGS -DCONFIG_BUFSIZE=65536 -DMAX_DOMAINS_F=21 -DMAX_HISTORY=25 -DNMM_NEST=0 -traditional" AR="ar" ARFLAGS="ru" )
make[3]: Entering directory `/home1/varun-garudahome/TempTest/PRASHANT/WRFV3/external/esmf_time_f90'
make[3]: Pattern not found: Error
E486: Pattern not found: Error
1,1 Top
```

Figure 5.5 Recompilation of WRF-CHEM model after resolving the Errors.

5.4 WPS Pre-processing Configuration & Compilation.



```
[varun@prlhpc WPS]$ ./configure
** WARNING: No path to NETCDF and environment variable NETCDF not set.
** would you like me to try to fix? [y]
y
Enter full path to NetCDF include directory on your system
/share/apps/INTEL_NETCDF.4
Enter full path to NetCDF library directory on your system
/share/apps/INTEL_NETCDF.4
```

```

Enter full path to NetCDF library directory on your system
/share/apps/INTEL_NETCDF.4
created new ./netcdf_links directory
total 8
lrwxrwxrwx 1 varun varun 26 Jan  3  2014 include -> /share/apps/INTEL_NETCDF.4/
lrwxrwxrwx 1 varun varun 26 Jan  3  2014 lib -> /share/apps/INTEL_NETCDF.4/
Will use NETCDF in dir: /home1/varun-garudahome/TmpTest/WRF NEW/WPS/netcdf_links
$JASPERLIB or $JASPERINC not found in environment. Using default values for library paths...
-----
Please select from among the following supported platforms.

1. Linux x86_64, PGI compiler (serial)
2. Linux x86_64, PGI compiler (serial_NO_GRIB2)
3. Linux x86_64, PGI compiler (dmpar)
4. Linux x86_64, PGI compiler (dmpar_NO_GRIB2)
5. Linux x86_64, PGI compiler, SGI MPT (serial)
6. Linux x86_64, PGI compiler, SGI MPT (serial_NO_GRIB2)
7. Linux x86_64, PGI compiler, SGI MPT (dmpar)
8. Linux x86_64, PGI compiler, SGI MPT (dmpar_NO_GRIB2)
9. Linux x86_64, IA64 and Opteron (serial)
10. Linux x86_64, IA64 and Opteron (serial_NO_GRIB2)
11. Linux x86_64, IA64 and Opteron (dmpar)
12. Linux x86_64, IA64 and Opteron (dmpar_NO_GRIB2)
13. Linux x86_64, Intel compiler (serial)
14. Linux x86_64, Intel compiler (serial_NO_GRIB2) -----
15. Linux x86_64, Intel compiler (dmpar) -----
16. Linux x86_64, Intel compiler (dmpar_NO_GRIB2)
17. Linux x86_64 g95 compiler (serial)
18. Linux x86_64 g95 compiler (serial_NO_GRIB2)
19. Linux x86_64 g95 compiler (dmpar)
20. Linux x86_64 g95 compiler (dmpar_NO_GRIB2)
21. Cray XT/XE Linux x86_64 (Opteron), PGI compiler 5.2 or higher (serial)
22. Cray XT/XE Linux x86_64 (Opteron), PGI compiler 5.2 or higher (serial_NO_GRIB2)
23. Cray XT/XE Linux x86_64 (Opteron), PGI compiler 5.2 or higher (dmpar)
24. Cray XT/XE Linux x86_64 (Opteron), PGI compiler 5.2 or higher (dmpar_NO_GRIB2)
25. Cray XT/XE Linux x86_64 (Opteron), Cray CCE compiler 7.0 or higher (dmpar)
26. Cray XT/XE Linux x86_64 (Opteron), Cray CCE compiler 7.0 or higher (dmpar_NO_GRIB2)

Enter selection [1-26] : 15
-----
[varun@prlhpc WPS]$ ./compile >& compile_wps.output
-----
[varun@prlhpc WPS]$ cp configure.wps_hpcrtl configure.wps
[varun@prlhpc WPS]$ ./compile >& compile_wps.output
[varun@prlhpc WPS]$ vi compile_wps.output
[varun@prlhpc WPS]$ ls
arch      compile_wps.output  configure.wps_hpcrtl  link_grb.csh  namelist.wps.all_options  PBS.sh    PBS.sh.e115356  print.out  ungrb.log
clean     configure          configure.wps       metgrid      namelist.wps.fire        PBS.sh.e115356  PBS.sh.e115362  README   ungrb.output
compile   configure.old      geogrid           metgrid.exe   namelist.wps.global      PBS.sh.e115362  PBS.sh.e115363  test.wps  util
compile.log  configure.wps  geogrid.exe       metgrid.log  namelist.wps.hmm       PBS.sh.e115363  PBS.sh.e118840  ungrb    varun
compile.output  configure.wps.backup  geogrid.log  namelist.wps   netcdf_links          PBS.sh.e118840  pp       ungrb.exe Vtable
[varun@prlhpc WPS]$ -----
xox - shuklaforamp... -----
[varun@prlhpc WPS]$ -----
[varun@prlhpc WPS]$ -----
[varun@prlhpc WPS]$ -----

```

Figure 5.6 WPS Pre-processing Configuration & Compilation.

5.5 Steps of WRF-CHEM Simulation Process.

Step 1: Goto WPS directory , cd /home1/varun-garudahome/TempTest/PRASHANT/WPS/

Step 2: see & edit namelist.wps for all namelist settings. Set the variables of the file namelist , they are mainly case study regional date and timing , resolutions and latitude longitude parameters etc.

Step 3: Now next step is to run geogrid.exe by invoking following command :

./geogrid.exe

Geogrid will read data from /home1/varun-garudahome/WRF/LIB/DATA/geog specified in namlist in geog_data_path variable.

After Successful completion of geogrid.exe it will give a message like :

!!!!!!!!!!!!!!

! Successful Completion of geogrid. !

!!!!!!!!!!!!!!

It will generate geo_em.d01 netcdf file.

NOTE: we can run ./geogrid.exe in MPI environment for lengthy and time consuming simulation.

Step 4: Now link correct Vtable from WPS/ungrib/Variable_tables/Vtable.GFS to Vtable in WPS directory

Ln -sf /home1/varun-garudahome/TempTest/PRASHANT/WPS/ungrib/Variable_Tables/Vtable.GFS Vtable

Step 5: Now link GRIB files to WPS directory for ungrib program using :

./link-grib.csh /home1/varun-garudahome/TempTest/PRASHANT/data/KEDARNATH/* It will
create soft links in WPS directory like GRIBFILE.*

Step 6: Now run ungrib.exe to generate intermediate files using :

./ungrib.exe

If execution will successfully completed then following message will display :

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

! Successful Completion of ungrib. !

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

It will generate FILE:/* files with the timestamps according to start and end date given in namelist.wps.

Step 7: To implement geographical data and meteorological data metgrid program is used as follow :

./metgrid.exe

If execution will successfully completed then following message will display :

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

! Successful Completion of metgrid. !

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

NOTE: If metgrid will find any missing values for any variable defined in Vtable then it will abort the compilation. For this kinds of errors check your Vtable and generate your own if necessary.

Step 8: Generated met_em* files using above command need to move in /home1/varun-garudahome(TempTest/PRASHANT/WRFV3/run/ folder.

Step 9: Now up to Previous step pre-processing steps of model is done, Now prepare global emissions data and its path for real data computations.

Goto :

```
cd /home1/varungarudahome(TempTest/PRASHANT/DATA/Global_emissions/PREP-CHEM-SRC- 1.2_21jun2012/bin
```

open prep_sources_chem.inp file and do settings for

domains and global data paths.

Vi prep_sources_chem.inp

In our case all global emissions are placed in main directory :
home1/varungarudahome/TempTest/PRASHANT/DATA/Global_emissions/

Global_emissions_V3/ After properly settings of domain and its related parameters run the following file :

```
./prep_chem_sources_RADM_WRF_FIM.exe
```

It will generated intermediate FIRE* files , which are binary files.

NOTE: see step-9 is an independent step , we can run prep_chem_sources_RADM_WRF_FIM.exe parallel with Pre- processing stage.

Step 10: Now go to run folder for final processing with real domain data :

```
cd /home1/varun-garudahome/TempTest/PRASHANT/WRFV3/run/
```

Step 11: Link boundry FIRE* files which are generated from PREP_SOURCEs to run directory using following command :

```
ln -sf /home1/varun-garudahome/TempTest/PRASHANT/DATA/Global_emissions/PREP-CHEM-SRC-
```

```
1.2_21jun2012/bin/FIRE-T-2013-06-14-20000-g1-gocartBG.bin wrf_gocart_backg
```

Likewise do for all generated files.

NOTE: It is good practice to move all files rather to provide soft links in run folders.

Step 12: Next step is to run real.exe but before to give command for execution copy namelist_input_ex1 to namelist.input file. Make changes according to namelist.wps file which we have prepared it in WPS directory.

```
./real.exe
```

It will generate wrfout_d01 and wrfbdy_d01 files. If they are not generated then look into rsl.error.0000 file for errors.

Step 13: Now convert generated binary files into netcdf files using covert_emiss.exe ,but before that copy namelist_input_ex2 to namelist.input which contains chemistry options . Edit values according to namelist.wps file which we have prepared in WPS directory.

```
./convert_emiss.exe
```

It will create wrfchemi_d01, wrffirechemi_d01,wrfchemi_gocart_bg_01 files.

Step 14: Next again run real.exe to include initial and boundary conditions for emissions in wrfinput_d01 and wrfbdy_d01.

Step 15: Now run wrf.exe to generate final forecast :

```
./wrf.exe
```

It will create wrfout_d01_2013_06_14_00:00:00(according to time and date given in namelists files) that contains forecast for chemistry.

6.

WRF-CHEM MODEL SPECIFICATION.

6.1 Implementation Environment.

6.2 Module Specification.

6.3 Security Specification.

6.4 Contour Plotting.

6.1 IMPLEMENTATION ENVIRONMENT :

The ARW is designed to be a flexible, state-of-the-art atmospheric simulation system that is portable and efficient on available parallel computing platforms. It is implemented on LINUX platform on HPC Custer at PRL.

The model-component portion of the package is mostly self-contained. The WRF model has been successfully ported to a number of Unix-based machines. Below is a list of the supported combinations of hardware and software for WRF.

| Vendor | Hardware | OS | Compiler |
|---------------------------------------|----------------|--------|--|
| Cray | X1 | UniCOS | Vendor |
| Cray | AMD | Linux | PGI |
| IBM | Power Series | AIX | Vendor |
| IBM | Power Series | Linux | Vendor |
| SGI | IA64 / Opteron | Linux | Intel |
| COTS(Commercial Off-The-Shelf system) | IA32 | Linux | Intel / PGI /gfortran / g95 /PathScale |
| COTS | IA64 / Opteron | Linux | Intel / PGI /gfortran /PathScale |
| Mac | Power Series | Darwin | xlf / g95 / PGI / Intel |
| Mac | Intel | Darwin | g95 / PGI / Intel |
| NEC | NEC | Linux | Vendor |

Table 6.1 : OS Supported By WRF

The WRF model may be built to run on a single-processor machine, a shared-memory machine (that uses the OpenMP API), a distributed memory machine (with the appropriate MPI libraries), or on a distributed cluster (utilizing both OpenMP and MPI) so it supports a multiuser system.

WRF runs from command prompt i.e. it does not contain any GUI based application interface. The utility programs are invoked through commands for compilation and build. Necessary inputs and variables are set in namelist.input file that is accessed during WRF run.

GUI version of WPS program is available called WRF Wizard. It is used to set up domain for analysis and WPS utilities can also be accessed through it like geogrid ,ungrib ,metgrid programs, namelist and Vtables. This Wizard is really helpful to create correct and desired simulation domain for Forecast. It can display domain in 2D and 3D.

Post processing utilities like Ncview , Xconv and Grace contains GUI interface. It is very easy and user friendly applications to plot 1D and 2D plots and graphs. It provides easier manipulations of plot and graphs. Ncview directly reads WRF generated netcdf files and create contour plots and 1D plot.

6.2 MODULES SPECIFICATION:

WRF modeling system is divided in further sub modules. Each of them has significant functionalities. They are as following:

- WRF Pre Processor
- WRF solver
- WRF/Chem solver
- Mozbc
- WRF Post Processor

6.2.1 Functionalities of WPS:

The WRF Preprocessing System (WPS) is a set of three programs whose collective role is to prepare input to the *real* program for real-data simulations. Each of the programs performs one stage of the preparation:

- It Defines simulation domain and fields to be extracted from raw data through Vtable.
- geogrid defines model domains and interpolates static geographical data to the grids.
- link_grib.sh program links GRIB data to WPS directory
- ungrib extracts meteorological fields from GRIB formatted files.
- metgrid horizontally interpolates the meteorological fields extracted by ungrib to the model grids defined by geogrid.

6.2.2 Functionalities of WRF Solver:

The WRF model code contains an initialization program (either for real-data, *real.exe*, or idealized data, *ideal.exe*), a numerical integration program (*wrf.exe*), a program to do one-way nesting (*ndown.exe*), and a program to do tropical storm bogussing (*tc.exe*). The WRF model functionalities include:

- Real-data and idealized simulations
- Various lateral boundary condition options for real-data and idealized simulations
- Full physics options, and various filter options

- Positive-definite advection scheme
- Non-hydrostatic and hydrostatic (runtime option)
- One-way and two-way nesting, and a moving nest
- Three-dimensional analysis nudging
- Observation nudging
- Regional and global applications
- Digital filter initialization

6.2.3 *Functionalities of WRF/Chem Solver:*

The WRF/Chem model is a part of the Weather Research and Forecasting (WRF) modeling package. WRF/Chem model is related to generating a forecast that includes chemical constituents beyond what is typically used by today's meteorological forecast models.

Functionalities of The WRF/Chem model are as following:

- Online calculation of biogenic emissions like MEGAN, BEIS (Biogenic Emissions Inventory System), Simpson and Guenther scheme that includes emissions of isoprene, monoterpenes, and nitrogen.
- Calculation of anthropogenic emissions like:
 - Global emissions data from the one-half degree RETRO and ten-degree EDGAR data sets.
 - User-specified anthropogenic emissions such as those available from the U.S. EPA NEI-05 data inventory. The user must provide the emissions data for their own domain in the proper WRF data file format.
 - Several choices for gas-phase chemical mechanisms including RADM2, RACM, CB-4 and CBM-Z chemical mechanisms.
 - The use of the Kinetic Pre-Processor, (KPP) to generate the chemical mechanisms. The equation files (using Rosenbrock type solvers) are currently available for RADM2, RACM, RACM-MIM, SAPRC-99, MOZART chemical mechanisms as well as others.
- Calculation of aerosol like:
 - The Modal Aerosol Dynamics Model for Europe - MADE/SORGAM
 - The Modal Aerosol Dynamics Model for Europe with the Volitity Basis Set aerosols – MADE/VBS
 - The Model for Simulating Aerosol Interactions and Chemistry (MOSAIC - 4 or 8 bins) sectional model aerosol parameterization
 - A total mass aerosol module from GOCART.
- An option for the passive tracer transport of greenhouse gases.
- A plume rise model to treat the emissions of wildfires.

6.2.4 Functionalities of Mozbc:

Mozbc is the utility that is designed to create time-varying chemical boundary conditions from the MOZART global model to a WRF/Chem simulation.

Functionalities of Mozbc are as following:

- It is a single cpu code that maps species concentrations from MOZART datasets to WRF/Chem concentrations for initial condition and boundary condition datasets.
- Conversion of MOZART species concentrations that is in volume mixing ratio to species units of WRF/Chem depending on whether it is a gas or aerosol.
- The mozbc utility allows for a versatile mapping to each WRF chemical species from MOZART species including multiple with choices for An individual weight factor per MOZART species or An overall conversion factor per WRF/Chem species.

6.2.4 Functionalities of WRF Post Processor:

WRF post processing tools are used for visualizing data. WRF output files are in Netcdf format. Various post processing tools are available, I have used Ncview and Grace for now.

Ncview:

Ncview reads the wrfout files (also all input files, including the emissions) directly using the simple command “>ncview wrfout....”. This makes it a very useful tool for a quick look analysis or diagnosis of a problem with the model run, including problems with any of the input files.

We can change the color maps, invert the data, generate line plots, etc. In addition, if the file contains a time series, Ncview permits simple animation of the data.

Grace:

Grace is tool to make two-dimensional plots of numerical data. It is a GUI-based program plus script-based tool. Its strength lies in the fact that it combines the convenience of a graphical user interface with the power of a scripting language which enables it to do sophisticated calculations or perform automated tasks. Functionalities of Grace are as following:

- Exporting vector graphics in text formats to (E)PS, PDF, MIF, and SVG formats
- Reading data form ASCII file that contains data in form of columns.
- Reading 1D netCDF files.
- Generating 1D and 2D graphs. It allows user to select column to plot on axis and other plot manipulation options.
- Formatting of graph as per user requirement.

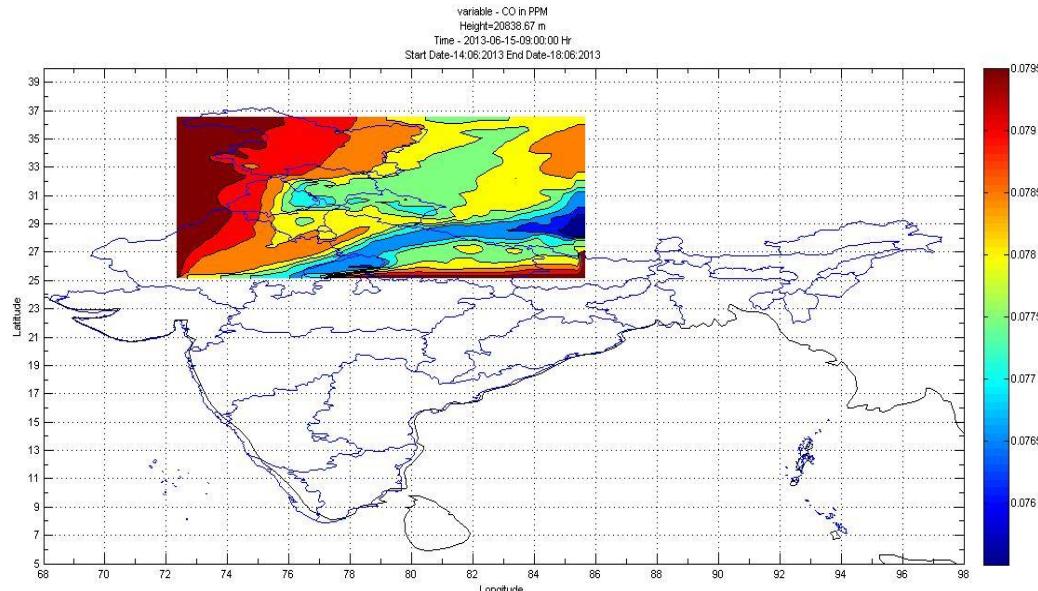
6.3 SECURITY FEATURES.

- WRF is implemented in Linux platform with all the security features included in it. There is no threat of malfunctioning programs because all the data goes to and from PRL Firewall and it is protected with many security layers.
- There is no direct access to WRF system i.e. User who is authorized to access the system can only access the system and perform operations on it. WRF is implemented in cluster and user who wants to have access to the cluster need a cluster account on master node. WRF is implemented under Dr. Varun's account so only he has access to the system. Apart from this it does not require internet connection during build so it is safe from outside network.

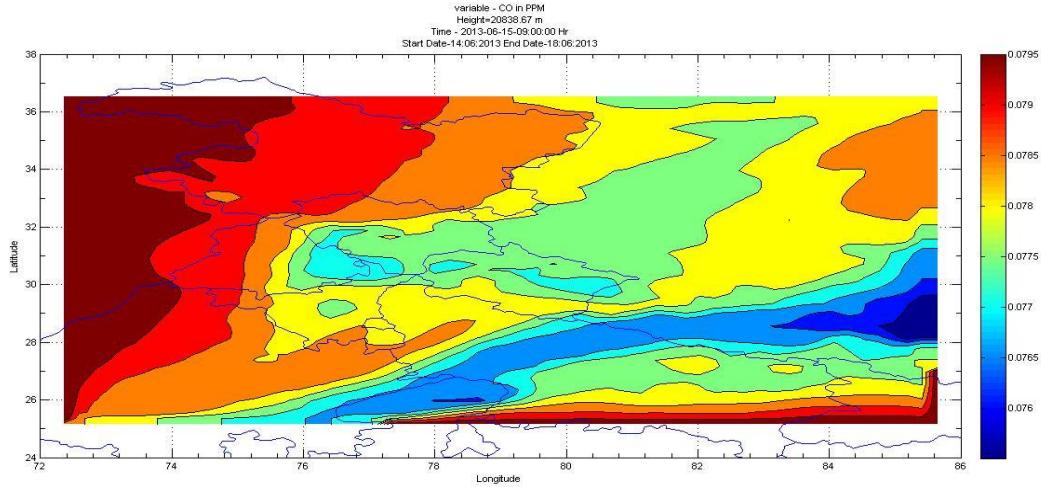
6.4 CONTOUR PLOTTING.

6.4.1 Contour Plotting for simulation of June, 14 to 18 ,2013.

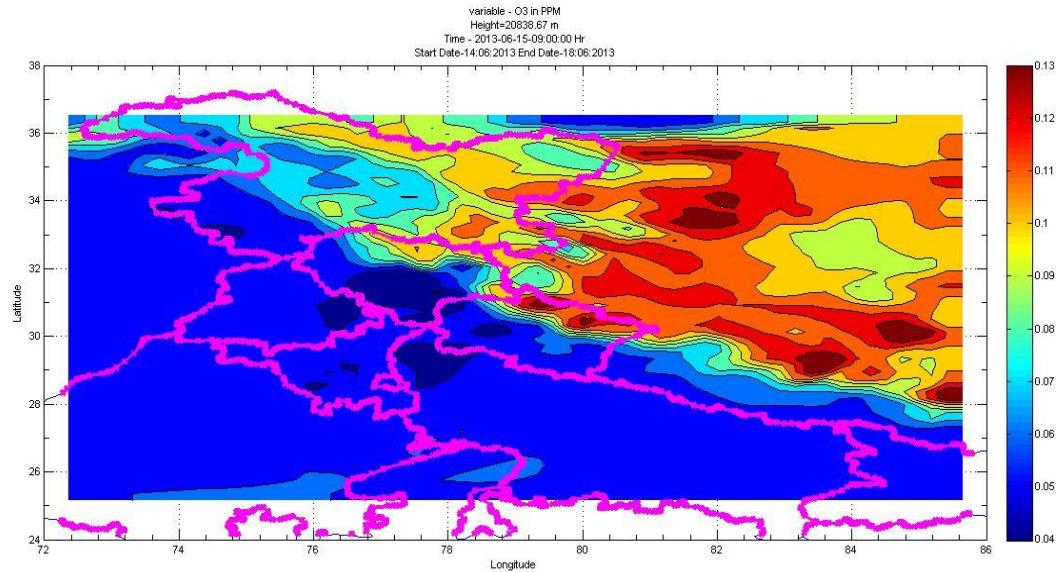
WRF-CHEM Model delivered the output file as simulation date for the run such as the wrfout_d01_2013-06-04_00:00:00 this in the netcdf format and we can plot the contour the for the specific variable in Matlab. Here diffenert contour for the simulation of June 14 to 18 , 2013 is shown for the Carbon Monoxide [co] and the Ozone [O3].



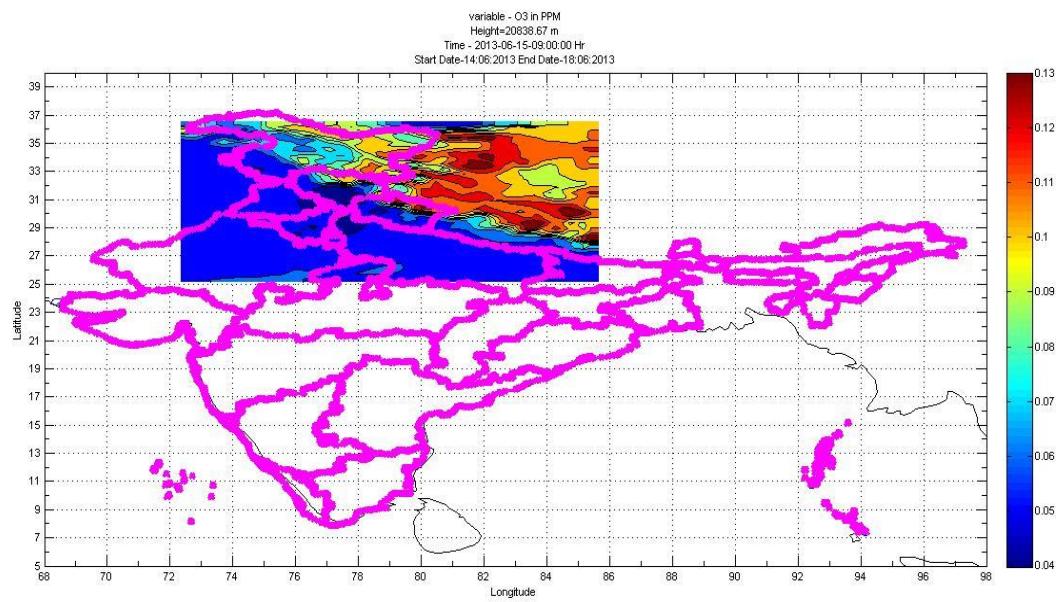
*Figure 6.1 Carbon Monoxide concentration [In PPM] contour.
(June 15, 2013 At 9:00 PM, 2.13Km)*



*Figure 6.2 CO[PPM] Concentration contour mapping on India's map.
(June 15, 2013 At 9:00 PM, 2.13Km)*



*Figure 6.3 Ozone Concentration [In PPM] Contour.
(June 15, 2013 At 9:00 PM, 2.13Km)*



*Figure 6.3 Ozone Concentration [In PPM] Contour.
(June 15, 2013 At 9:00 PM, 2.13Km)*

7.

CONCLUSION AND FUTURE WORK.

7.1 Conclusion.

7.2 Future Work.

7.1 CONCLUSION.

In this report we have seen the need of HPC clustering in WRF-CHEM model for weather forecasting. WRF-CHEM simulation process takes huge amount of processing time and requires complex computing. For that, making the whole simulation process more flexible and efficient it should require correct analysis and data metrics.

We have given the details of parallel processing which is used in HPC cluster computing. As clustering is one of the data mining technique by which we can utilize the memory and processing power of systems.

WRF-CHEM simulation process takes huge amount of processing time and requires complex computing. For that, making the whole simulation process more flexible and efficient it should require correct deployment of WRF-CHEM model on HPC clustering.

The information and architectures given in this survey will be helpful to the researchers for Physics, Space & Atmospheric Sciences, Astronomy, Astrophysics & Solar Physics, and Planetary & Geosciences.

7.2 Future Work.

My this semester work would be of simulation, testing and make operational runs of WRF-CHEM model on HPC cluster to forecast weather and atmospheric chemistry over Grid of India.

Analysis part of generated output is also completed by me, But the simulation for the very long time is taken too much time so it will be done after the strong testing of WRF-CHEM model on real case study data over India.

REFERENCES

➤ WEB REFERENCES.

- <http://www.mcs.anl.gov/research/projects/mpich2>
- www.nomad3.ncep.noaa.gov
- <http://www.nesdis.noaa.gov>
- <http://www.noaa.gov>
- <http://grads.iges.org/grads/head.html>
- <http://www.centos.org/docs/2/rhl-cg-en-7.2/cron-task.html>
- <http://icdc.zmaw.de/ncep.html?&L=1>
- http://jisao.washington.edu/data_sets/reanalysis/
- http://ruc.noaa.gov/wrf/WG11/wrf_tutorial_exercises_v33/exercise_1.html
- <http://publib.boulder.ibm.com>
- <http://forum.abinit.org>
- <http://www.linuxquestions.org>
- <http://stackoverflow.com/>
- <http://www.rhinocerus.net>
- <http://wrfportal.org/DomainWizard.html>
- <http://nomads.ncdc.noaa.gov>
- <http://www.wrf-model.org/index.php>
- http://www.mmm.ucar.edu/wrf/users/download/get_source.html
- <http://www.unidata.ucar.edu/packages/netcdf/examples/files.html>
- http://www.fs.fed.us/rm/pubs_int/int_gtr122.pdf
- <https://forge.gridforum.org/sf/projects/ogsa-hpcp-wg>

➤ BOOKS.

- Brown, S. A, M. Folk, G. Goucher, and R. Rew, "Software for Portable Scientific Data Management," Computers in Physics, American Institute of Physics, Vol. 7", No. 3, May/June 1993.
- "PBS Professional User's Guide", Altair PBS Professional 10.2, Updated:11/23/09. Edited by: Anne Urban
- Russ Rew, Glenn Davis, Steve Emmerson, Harvey Davies, Ed Hartnett, and Dennis Heimbigner ,“ netCDF programming guide”, Unidata Program Center.
- Wei Wang, Cindy Bruyère, Michael Duda, Jimy Dudhia, Dave Gill, “ ARW Ver. 3, Modeling System”.

- Michael Thorne, “Introduction to using the netCDF data format with Fortran 90”, Last Updated: July 20, 2010
- Russ Rew, Unidata Program Center, and Robert Pincus, “NetCDF User’s Guide for Fortran 90”, NOAA/CIRES Climate Diagnostics Center.
- Andy Heaps , “An Introduction to Using IDL in Meteorology” Version 3, NCAS-Climate, Department of Meteorology.
- Steven Peckham, Georg Grell, “WRF/Chem Set-up and Running in Real Time v3.3”
- Steven Peckham , “A Quick Review Of How ToSet-Up & Run WRF/Chem”.
- Roger S. Pressman, Ph.D., “Software Engineering”.
- Rajib-Mall “Software-Engineering”.
- Dr. Steven E. Peckham, Dr. Georg A. Grell, Dr. Jerome D. Fast, “WRF/Chem version 3.4 User’s Guide”.
- Rajesh Kumar,”Weather Research and Forecasting Model with Chemistry (WRF-CHEM) over South Asia”.
- Rafael Santos Lima, Marcelo Alonso, Megan Bela, Valter Oliveira, Rafael Fonseca, deleine Gácita, Gabriel Pereira, Karla M. Longo, Saulo R. Freitas, Georg Grell, “Generating Emissions Fields for WRF-Chem with PREP-CHEM-SRC”.