3. AREAS FOR ARCHITECTURAL IMPROVEMENTS

Meeting Provider-centric Goals
- Predictive and proactive resource auto-scaling
- Agile and predictable reliability and security mechanisms
- Dynamic orchestration and effective configuration of services

Meeting User-centric Goals
- Real-time resource monitoring and elasticity
- Highly effective resource management mechanisms
- Energy conservation
- Load balancing

“Need algorithms and middleware that address both these objectives simultaneously, which is the focus of the CLARINET project”

4. CLOUD SERVICE PROVIDER-CENTRIC IMPROVEMENTS

I. A Cloud Middleware for Assuring Performance and High Availability of Soft Real-time Applications
- Enables hosting of real-time streaming applications in the cloud that demand both high availability and low latency
- An architecture for a fault-tolerant framework that can be used to automatically deploy replicas of virtual machines in data centers in a way that optimizes resources while assuring availability and responsiveness
- A pluggable framework within the fault-tolerant architecture that enables plugging in different placement algorithms for VM replica deployment
- A framework for real-time dissemination of resource utilization information using the DDS real-time publish/subscribe framework, which is required by the replica selection and placement selection

II. iOverbook: Managing Cloud-based Soft Real-time Applications in a Resource-Overbooked Data Center
- A machine learning approach to make systematic and online determination of overbooking ratios such that the quality of service needs of soft real-time systems can be met while still benefiting from overbooking
- Utilizes historic data of tasks and host machines in the cloud to extract their resource usage patterns to predict future resource usage and expected mean performance of host machines
- Used a large usage trace made available by Google of one of its production data centers
- iOverbook can help CSPs improve their resource utilization by an average of 33% and save 65% power in the data center

III. hALT: A Performance Interference-aware Virtual Machine Placement Strategy for Supporting Soft Real-time Applications in the Cloud
- Performance of soft-real-time applications jitter caused due to resource overbooking
- A machine learning-based, online virtual machine placement solution
- Classifies the VMs based on their historic mean CPU and memory usage, and performance features
- Learns the best patterns of collocating the classified VMs by employing machine learning techniques

5. CLOUD USER’S APPLICATION-CENTRIC IMPROVEMENTS

I. Model-driven Performance Estimation, Deployment, and Resource Management for Cloud-hosted Services
- Different cost models and heterogeneous APIs offered by cloud service providers (CSPs)
- Domain-specific modeling languages in our MDE framework and the associated generative mechanisms to estimating performance and cost to host the services in the cloud and automated deployment

II. A Self-Tuning System based on Application Profiling and Performance Analysis for Optimizing Hadoop MapReduce Cluster Configuration
- Manages the resources of a MapReduce cluster, and configures the framework that will optimize the performance and reliability of MapReduce applications
- Profiling and Performance Analysis-based System (PPABS) framework automates the tuning of Hadoop configuration settings based on deduced application performance requirements
- Analyzer, trains PPABS to form a set of equivalence classes of MapReduce applications
- Recognizer, classifies an incoming unknown job to one of these equivalence classes

III. A Model-driven Approach for Price/Performance Tradeoffs in Cloud-based MapReduce Application Deployment
- How to select the best CSP for MapReduce applications?
- A web service which helps MapReduce users estimate the cost of deployment and execution a priori allowing them a choice of CSPs by making appropriate price/performance tradeoffs
- Automates the deployment of MapReduce application on a chosen CSP platform

6. EXPERIMENTAL VALIDATIONS

iOverbook: Actual and Predicted Hourly Mean CPU and Memory Value Comparison

Without Remus: Consistent standard deviation and average latency (VM is not replicated) compared to Remus with Efficient Placement

Higher fluctuation of latency is observed when Remus is used whereas average latency does not increase significantly

7. FUTURE WORK

- Address the challenges at the 3-layers of industrial internet: Machine, Plant and Enterprise
- Design the Peer-to-Peer Cloud architecture to support IoT and Industrial internet applications
- iPlace: Intelligent Tunable Power- and Performance-aware Virtual Machine Placement Middleware
- Perform experiments and sensitivity analysis to figure out the independent factors affecting performance for different CSPs
- Analyze Hadoop log data to find the performance patterns and come up with optimal configurations for running MapReduce jobs
- While DDS can be used as a Data aggregation and distribution at plant/machine level, for long-term analysis, machine learning and deriving business logic; cloud resources and distributed computing paradigms like Hadoop (batch processing) and Storm (stream processing) must be leveraged.