Autonomic Clouds
Part I – Introduction

Omer Rana, Cardiff University
Manish Parashar, Rutgers University

Overview

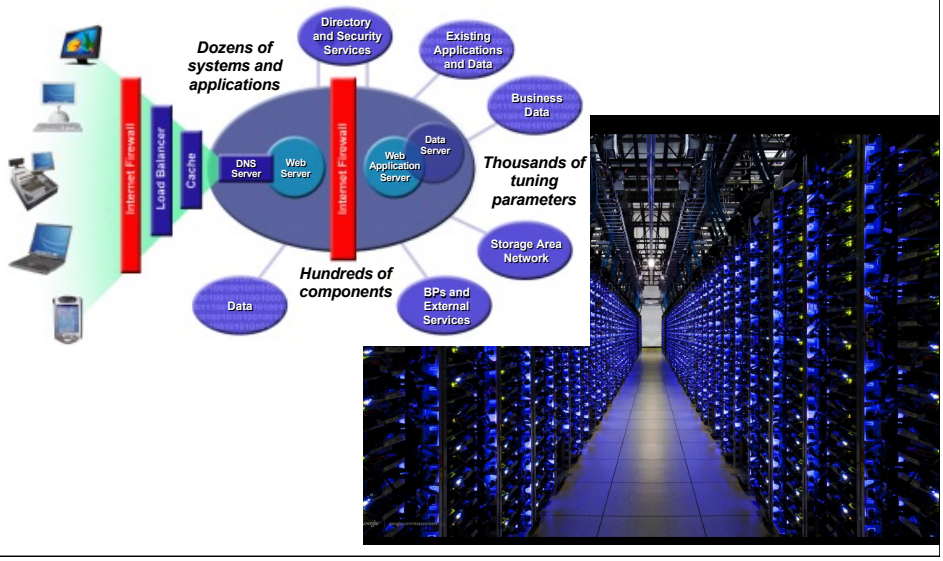
1. Autonomic Computing – Introduction, Motivations, Overview
2. Autonomics and Clouds
3. Exploring Cloud Autonomics using CometCloud

THE PROBLEM OF COMPLEXITY

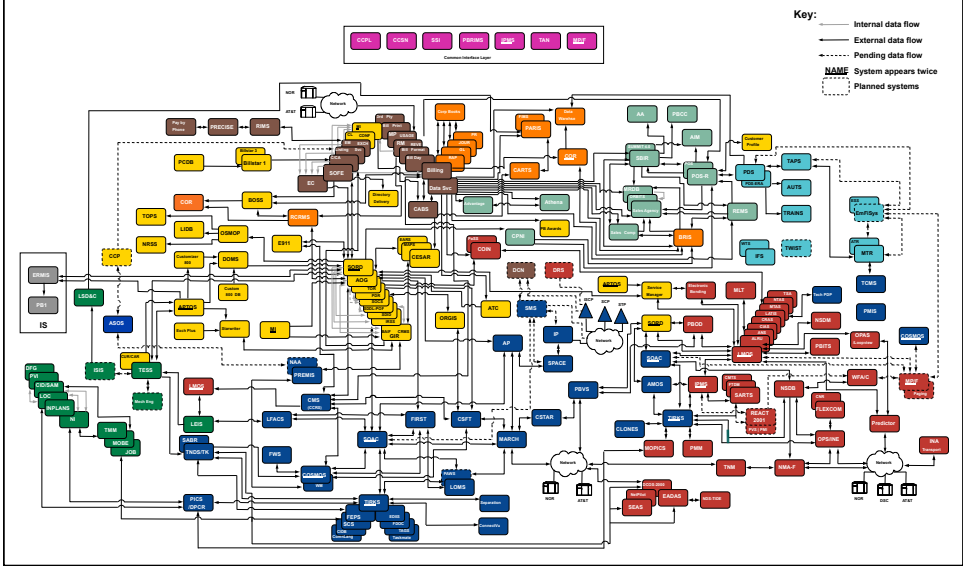
Emerging Information Ecosystems – Smaller/Cheaper/Faster/Powerful/ Connected ….

- Explosive growth in computation, communication, information and integration technologies
  - Data, computing & communication are ubiquitous
- Pervasive ad hoc "anytime-anywhere" data and access environments
  - Ubiquitous access to information
  - Peers capable of producing/consuming/processing information at different levels and granularities
  - Embedded devices in clothes, phones, cars, mile-markers, traffic lights, lamp posts, medical instruments …
- "On demand" computational/storage/software services
- **Internet of Things**

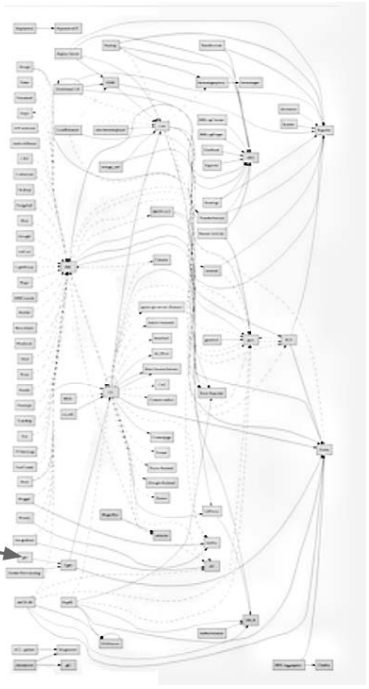
Architectural Complexity (Enterprise System)



Architectural Complexity (Traditional OSS/BSS)



GMail circa 2008



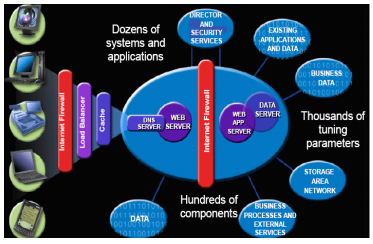
your browser

Image source: Hareesh Nagarajan

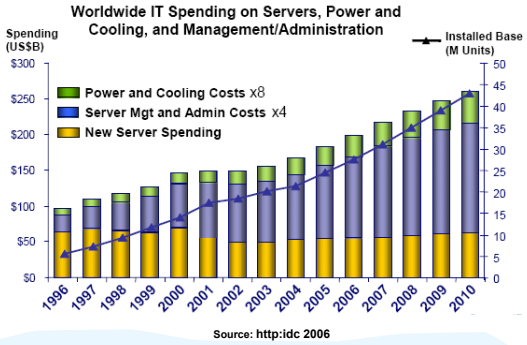
Operational Complexity

- Individual system elements increasingly difficult to maintain and operate
  - 100s of config, tuning parameters for commercial databases, servers, storage
- Heterogeneous systems are becoming increasingly connected
  - Integration becoming ever more difficult
- Behaviors, execution context, interactions not known a priori
  - Architects can't intricately plan component interactions
  - Increasingly dynamic, opportunistic and unanticipated
  - Very heterogeneous and complex workloads
- This places greater burden on system administrators, but
  - They are already overtaxed
  - They are already a major source of cost (6:1 for storage) and error
- We need self-managing computing systems
  - Behavior specified by sys admins via high-level policies
  - System and its components figure out how to carry out policies

Motivations for Autonomic Computing



Source: http://www.almaden.ibm.com/almaden/talks/Morris_AC_10-02.pdf



Source: http:idc 2006

2/27/07: Dow fell 546. Since worst plunge took place after 2:30 pm, trading limits were not activated

8/3/07: (EPA) datacenter energy use by 2011 will cost $7.4 B, 15 power plants, 15 Gwatts/hour peak

8/1/06: UK NHS hit with massive computer outage. 72 primary care + 8 acute hospital trusts affected.

8/12/07: 20K people + 60 planes held at LAX after computer failure prevented customs from screening arrivals

Key Challenge

Current levels of scale, complexity and dynamism make it infeasible for humans to effectively manage and control systems and applications

Some Effects of Complexity

- Expensive
  - Cost of management by administrators is increasing
- Fragile
  - Complex interdependencies make it hard to diagnose and fix problems
  - More prone to human error (additional cost)
  - Upgrades, performance tuning, re-purposing all suffer
- Inflexible
  - Reluctance to change infrastructure once it is working
  - Does not support agile business (new software, business processes)
- Worsening
  - Technology innovations typically exacerbate the problem, preventing product innovations from being deployed

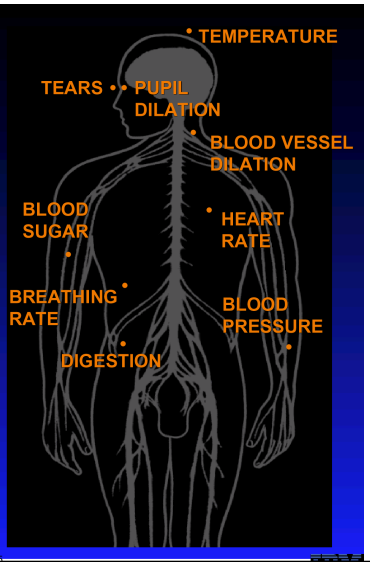
A Solution: Self-managing systems

WHAT IS AUTONOMIC COMPUTING

Can Biology Inspire IT Systems?

The Autonomic Nervous System Monitors and Regulates:

Without requiring our conscious involvement - e.g., when we run, it increases our heart and breathing rate



A View of Biological Adaptation and Evolution

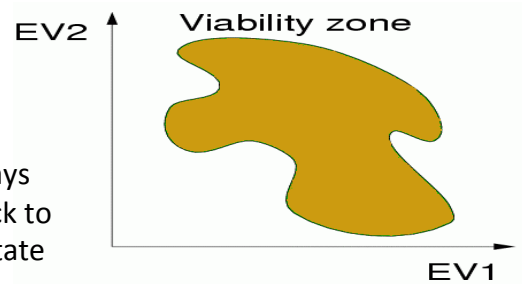
- Living systems can be described in terms of interdependent variables:
  - each capable of varying over a range with upper and lower bounds, e.g. bodily temperature, blood pressure, heart rate etc.
  - environmental change may cause fluctuations but bodily control mechanisms autonomically act to maintain variables at a stable level, i.e. homeostatic equilibrium with the environment

A View of Biological Adaptation and Evolution

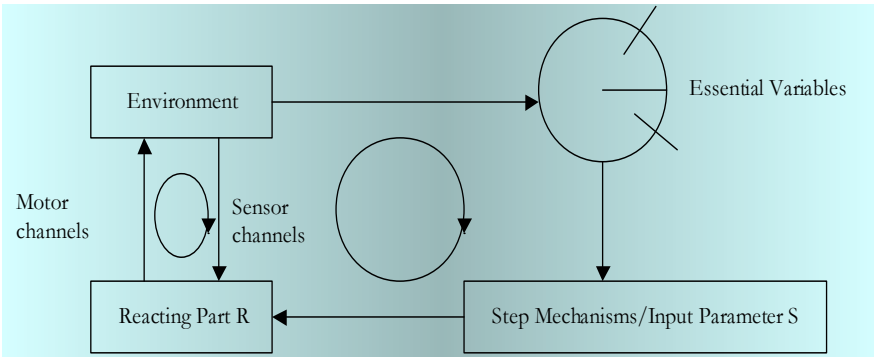
- Three types of adaptation to environmental disturbance are available to higher organisms:
  - Short-term change - e.g. Environmental temperature change moves the bodily temperature variable to an unacceptable value. This rapidly induces an autonomic response in the (human) organism i.e. either perspiring to dissipate heat or shivering to generate heat.
    - **Such adaptation is quickly achieved and reversed.**
  - Somatic change - prolonged exposure to environmental temperature change results in the impact of the change being absorbed by the organism i.e. acclimatization.
    - **Such change is slower to achieve and reverse.**
  - Genotypic change - adaptation through mutation and hence evolution. A species adapts to change by shifting the range of some variables. e.g. in a cold climate grow thicker fur.
    - **Such genotypic change is recorded at a cellular level and becomes hereditary and is irreversible in the lifetime of the individual.**

Adaptive Biological Systems

- The body's internal mechanisms continuously work together to maintain essential variables within physiological limits that define the viability zone
- Two important observations:
  - The goal of the adaptive behavior is directly linked with the survivability
  - If the external or internal environment pushes the system outside its physiological equilibrium state the system will always work towards coming back to the original equilibrium state



Ashby's Ultrastable System



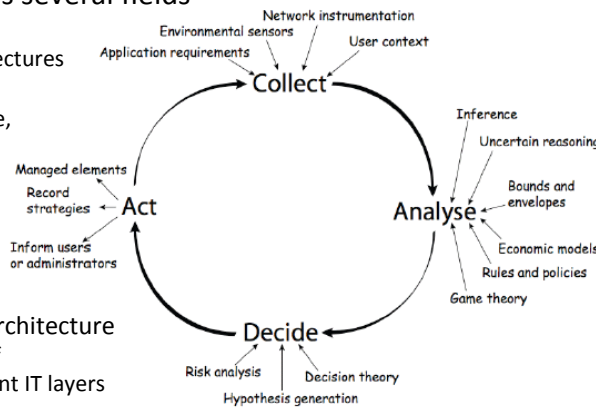
Autonomic Computing – A Pragmatic Approach

- Separation + Integration + Automation !
- Separation of knowledge, policies and mechanisms for adaptation
- The integration of self–configuration, – healing, – protection,–optimization, …
- Self-* behaviors build on automation concepts and mechanisms
  - Increased productivity, reduced operational costs, timely and effective response
- System/Applications self-management is more than the sum of the self-management of its individual components

M. Parashar and S. Hariri, ***Autonomic Computing: Concepts, Infrastructure, and Applications,*** CRC Press, Taylor & Francis Group, ISBN 0-8493-9367-1, 2007.

Autonomic Computing Theory

- Integrates and advances several fields
  - Distributed computing
    - Algorithms and architectures
  - Artificial intelligence
    - Models to characterize, predict and mine data and behaviors
  - Security and reliability
    - Designs and models of robust systems
  - Systems and software architecture
    - Designs and models of components at different IT layers
  - Control theory
    - Feedback-based control and estimation
  - Systems and signal processing theory
    - System and data models and optimization methods
- Requires experimental validation

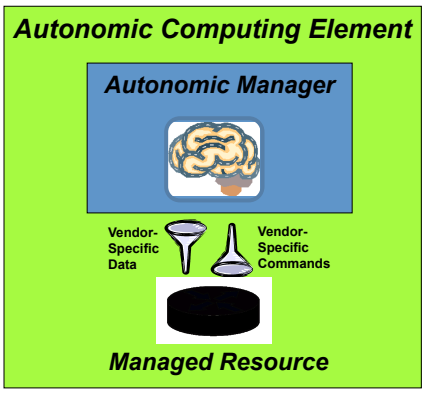


(From S. Dobson et al., ACM Tr. on Autonomous & Adaptive Systems, Vol. 1, No. 2, Dec. 2006.)

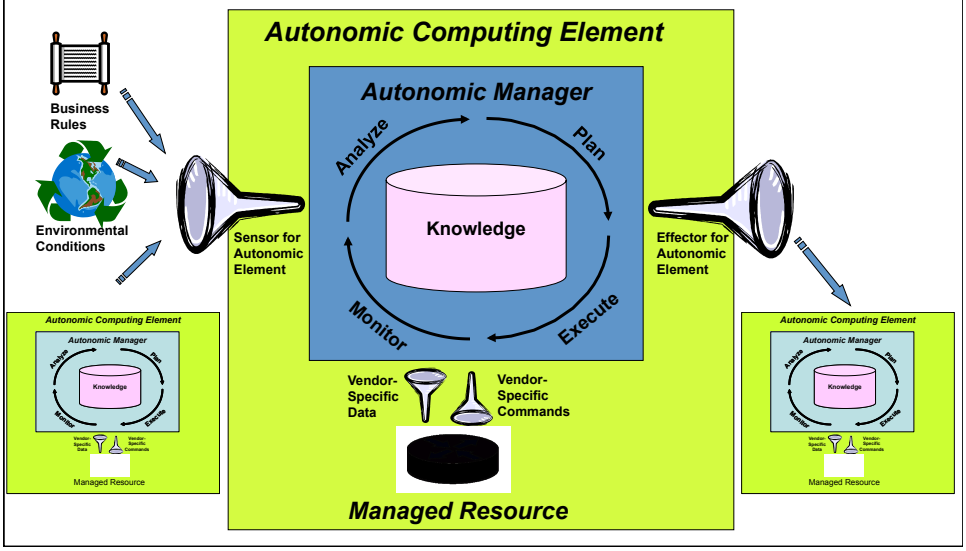
BASIC ARCHITECTURE PRINCIPLES

Autonomic Computing Element (ACE)

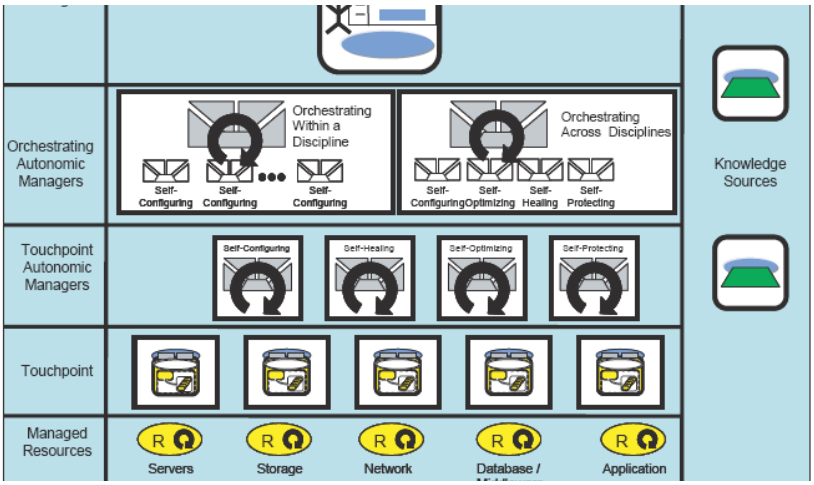
- An abstraction that enables an autonomic system to manage the functionality of a managed resource
- ACEs manage their own behavior
- ACEs are building blocks that are arranged to provide higher-level system behavior



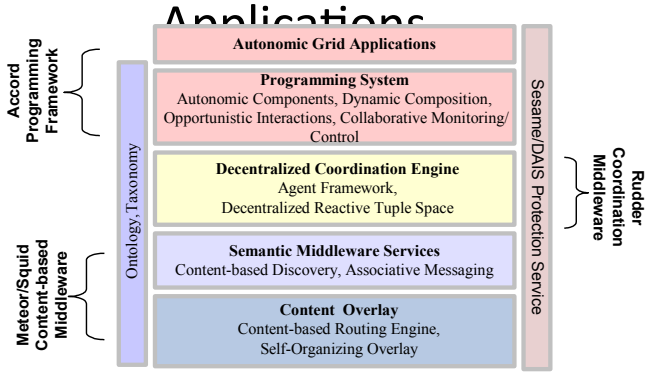
IBM's Autonomic Control Loop



Autonomic Computing – Conceptual Architecture from IBM



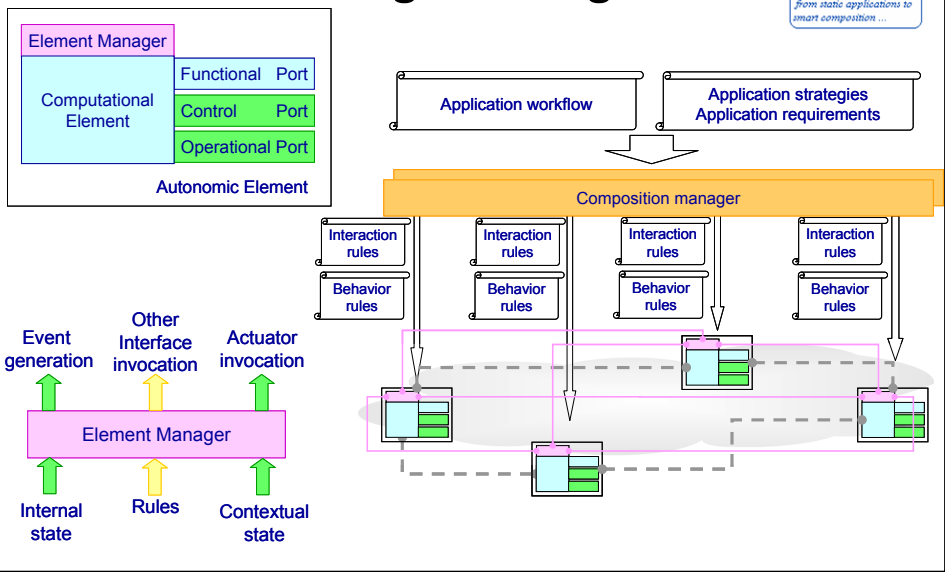
Project AutoMate: Autonomics for Applications



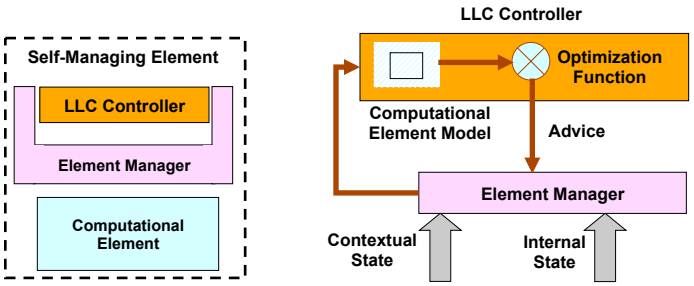
- Conceptual models and implementation architectures
  - programming systems
    - extend popular programming models to address uncertainty
    - Integrated physical and computational worlds
  - asynchronous computational engines
  - data quality estimation
  - coordination and messaging middleware

"AutoMate: Enabling Autonomic Grid Applications," M. Parashar et al, Cluster Computing: The Journal of Networks, Software Tools, and Applications, Vol. 9, No. 2, pp. 161 – 174, 2006.

Autonomic Formulations/ Programming

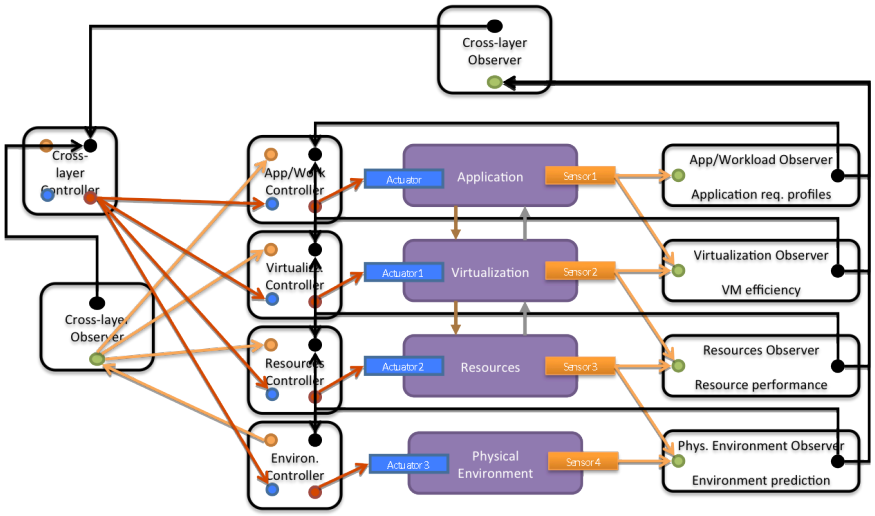


LLC-based Self Management in Accord



- Element/Service Managers are augmented with LLC Controllers
  - monitors state/execution context of elements
  - enforces adaptation actions determined by the controller
  - augment human defined rules

Crosslayer Autonomics



Other Approaches

- Emergence
  - A system exhibits emergence when there are coherent **emergents** at the **macro-level** that dynamically arise from the interactions between the parts at the **micro-level**. Such emergents are **novel** w.r.t. the individual parts of the system.
- Self-Organisation
  - Self-organisation is a dynamical and adaptive process where systems **acquire and maintain structure** themselves, **without external control**

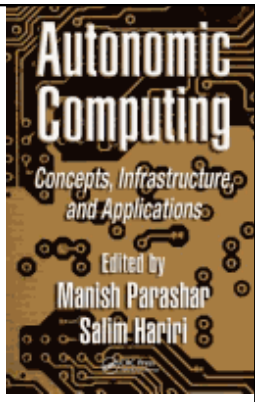
Research Issues and Challenges

- Defining autonomic elements
  - Programming paradigms and development models/frameworks
    - Autonomic element definition and construction
    - Rule definition, representation, and enforcement
- Constructing autonomic systems/applications
  - Composition, coordination, interactions models and infrastructures
    - Dynamic (rule-based) configuration, execution and optimization
    - Dynamic (opportunistic) interactions, coordination, negotiation
- Execution and management
  - Runtime and middleware services
    - Discovery, Coordination, Messaging, Security, Management, …
  - Security, protection
  - Fault tolerance, reliability, availability, …
- Learning, AI
- Non-determinism, correctness, robustness, performance/scalabilty, …

Summary and Concluding Remarks

- Increased complexity, heterogeneity, uncertainty, and scale require new paradigms to design, control and manage systems and applications
- Systems and Applications need to operate reliably, securely, efficiently and cost-effectively
  - Need a holistic approach that can dynamically integrate and address all these issues simultaneously at the layers of the system and application hierarchy
- Autonomic Computing provides an interesting, pragmatic approach to address these issues
- Many challenges are ahead including composing and analyzing in real-time the operations and states of systems and applications
- Need new bio-inspired metrics that accurately characterize and quantify the system and application normal and abnormal states

Questions ?



o.f.rana@cs.cardiff.ac.uk
parashar@rutgers.edu