

Autonomic computing

Presented by

www.seminarlab.com

WHAT IS AUTONOMIC COMPUTING?

- ❑ **Autonomic computing is a new vision of computing initiated by IBM.**
- ❑ **This refers to the development of Computer systems that regulate themselves much in the same way our nervous systems regulates and protects our bodies.**
- ❑ **This new paradigm shifts the fundamental definition of the technology age from one of computing, to one defined by data. This new view of computing will necessitate changing the industry's focus on processing speed and storage to one of developing distributed networks that are largely self-managing, self-diagnostic, and transparent to the user.**

KEY FEATURES

- ✓ *Knows itself* : Its components must also possess a system identity. An autonomic system will need detailed knowledge of its components, current status, ultimate capacity, & all connections to other systems to govern itself.
- ✓ *Configure Itself* : Configure & reconfigure itself.
- ✓ *Optimizes Itself* :
- ✓ *Heal Itself* : It must be able to recover from routine and extraordinary events that might cause some of its parts to malfunction.
- ✓ *Protect Itself* : It must detect & protect itself against various types of attacks.
- ✓ *Adapt Itself* : It will find and generate rules for how best to interact with neighboring systems & tap available resources, even negotiate the use by other system of its underutilized elements, changing both itself and its environment in the process.
- ✓ *Open Itself* :
- ✓ *Hide Itself* : Hide complexity.

Autonomic Computing and Current Computing-A Comparison

- In an autonomic environment, system components from hardware such as desktop computers and mainframes to software such as operating systems and business applications are self-configuring, self-healing, self-optimizing and self-protecting. These self-managing attributes can be compared as given in the table.**

Table 1. Four aspects of self-management as they are now and would be with autonomic computing.

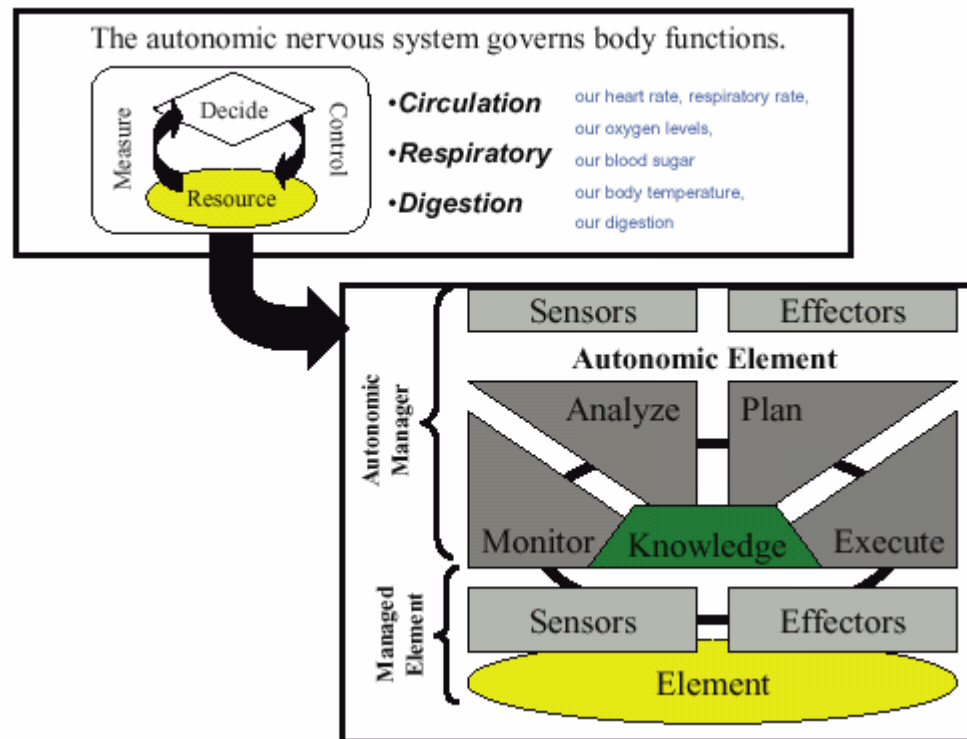
Concept	Current computing	Autonomic computing
Self-configuration	Corporate data centers have multiple vendors and platforms. Installing, configuring, and integrating systems is time consuming and error prone.	Automated configuration of components and systems follows high-level policies. Rest of system adjusts automatically and seamlessly.
Self-optimization	Systems have hundreds of manually set, nonlinear tuning parameters, and their number increases with each release.	Components and systems continually seek opportunities to improve their own performance and efficiency.
Self-healing	Problem determination in large, complex systems can take a team of programmers weeks.	System automatically detects, diagnoses, and repairs localized software and hardware problems.
Self-protection	Detection of and recovery from attacks and cascading failures is manual.	System automatically defends against malicious attacks or cascading failures. It uses early warning to anticipate and prevent systemwide failures.

AUTONOMIC COMPUTING ARCHITECTURE

This architecture concepts provide a mechanism discussing, comparing and contrasting the approaches different vendors use to deliver self-managing attributes in an autonomic computing system. It involve control loop.

- Self configure.
- Self-heal.
- Self-optimize.
- Self-protect.

architecture



-
- ❑ **Autonomic systems will be interactive collections of autonomic elements**
 - ❑ **An autonomic element will typically consist of one or more managed elements coupled with a single autonomic manager that controls and represents them.**
 - ❑ **A managed element is what the autonomic manager is controlling. An autonomic manager is a component that implements a control loop**

Control loops

A control loop can be provided by a resource provider, which embeds a loop in the runtime environment for a particular resource.

An intelligent control loop can enable the system to do such things as:

-
- ❑ **Self-configure, by installing software when it detects that software is missing**
 - ❑ **Self-heal, by restarting a failed element**
 - ❑ **Self-optimize, by adjusting the current workload when it observes an increase in capacity**
 - ❑ **Self-protect, by taking resources offline if it detects an intrusion attempt.**

AUTONOMIC ELEMENTS—individual system constituents that contain resources & deliver services to humans & other autonomic elements. An autonomic element will typically consist of one or more managed elements coupled with a single autonomic manager that controls & represents them.

In an autonomic environment, autonomic elements work together, communicating with each other & with high level management tools. They protect the system, while hiding the complexity.

Control loop divided into two major elements--- a managed element & an autonomic manager.

Managed elements

Controlled system component.

could be a hardware resource, such as storage, cpu, or a printer , or a software resource, such as a database, a directory service, or a large legacy system.

the managed element is controlled through its sensors & effectors:

Sensors: it provide mechanisms to collect information about the state & state transition of an element.

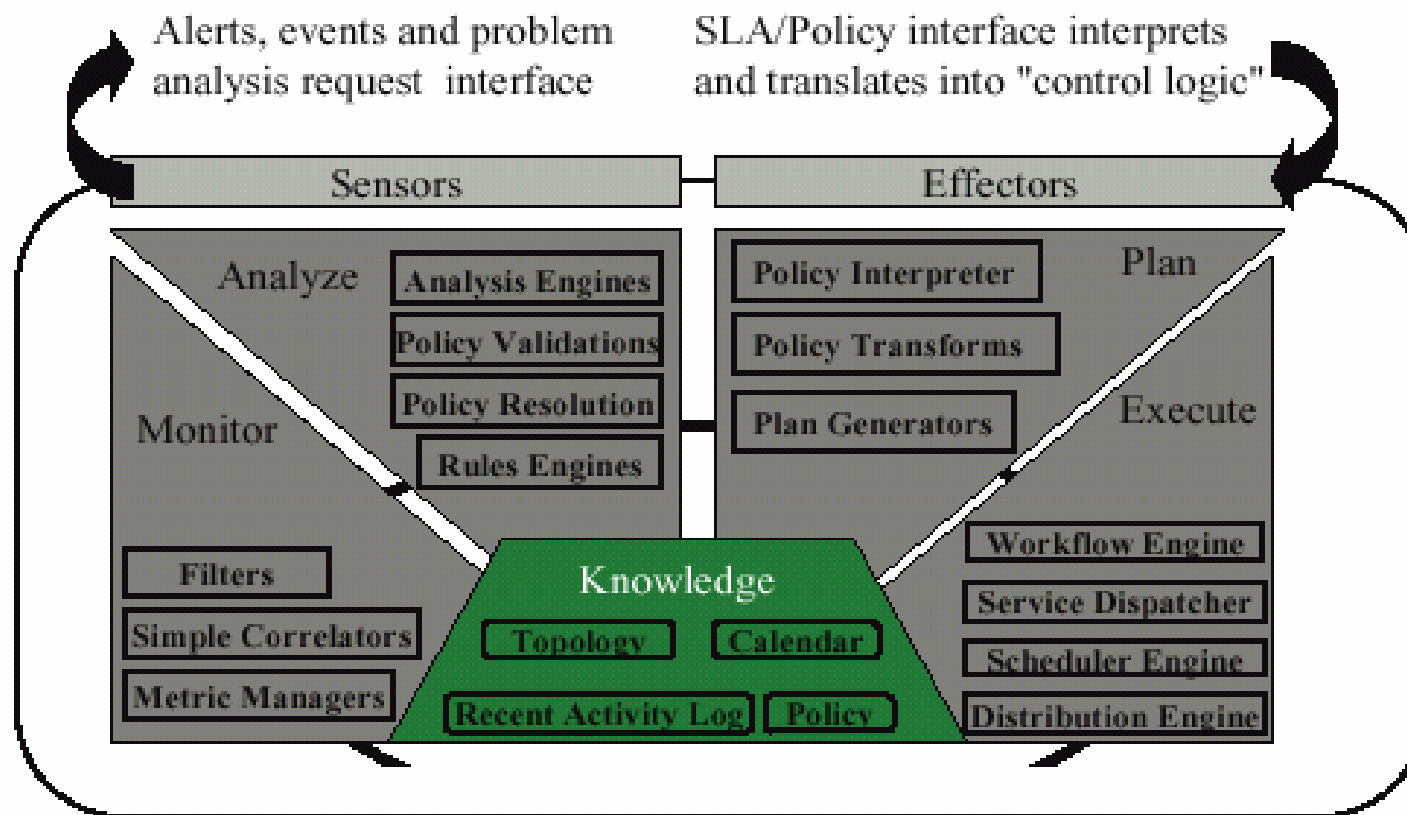
to implement the sensors, you can either use a set of " get " operations to retrieve information about the current state, or a set of management events.

Effectors are mechanism that change the state of an element.

Autonomic manager

it is a component that implements the control loop. The architecture of autonomic manager include 4 parts.

The functional details of an autonomic manager

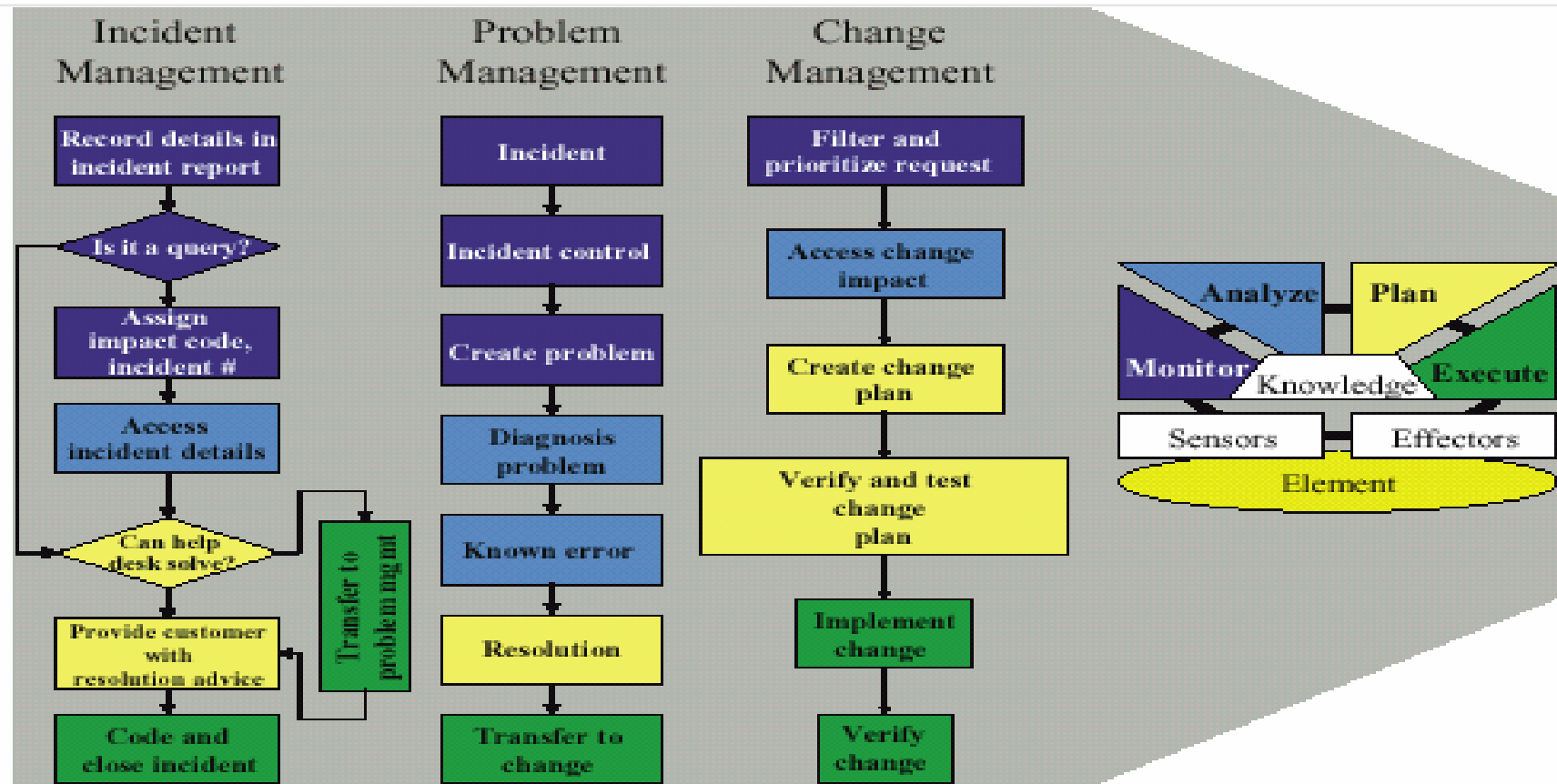


Interface to real and virtualized resources and components that regulate control.

FUNCTIONAL PARTS

- ❑ **MONITOR** : It provides the mechanisms that collect, aggregate, filter, manage & report details collected from an element.
- ❑ **ANALYZE** : It provide mechanisms to correlate & model complex situations.
- ❑ **PLAN** :This part provides the mechanisms to structure the action needed to achieve goals & objectives. It uses policy information to guide its work.
- ❑ **EXECUTIVE** :This part provides the mechanisms that controls the execution of a plan with considerations for on-the-fly updates.

Self-managing systems change the IT business



A GRAND CHALLENGE

The difficulty in developing and implementing autonomic computing is daunting enough to constitute a Grand Challenge.

ENGINEERING CHALLENGES

Life cycle of an autonomic element

- Design, test, and verification.
- Installation and configuration
- Monitoring
- Upgrading
- Managing the life cycle

Relationships among autonomic elements

- ✓ **Specification**
- ✓ **Location**
- ✓ **Negotiation**
- ✓ **Provision**
- ✓ **Operation**
- ✓ **Termination**

System wide issues

- ✓ **Goal specification**

SCIENTIFIC CHALLENGES

- ❖ Behavioral abstractions and models
- ❖ Robustness theory
- ❖ Learning and optimization theory
- ❖ Negotiation theory
- ❖ Automated statistical modeling

TECHNOLOGIES

- The following table gives a summary of some current components and their proposed development under autonomic computing.

Levels of sophistication	Known examples	Current directions	Future goal
Serving the world (i.e., people, business processes)	SMS	Policy management, Storage tank	Policy language and protocols
Heterogeneous components interacting	SNMP	Mounties, Workload management	Autonomic computing stack, Social policy
Homogeneous components interacting	Adaptive network routing, network congestion control, high availability clustering	Collective intelligence, Storage Bricks, Oceano	New packaging concepts for storage, Subscription computing
Components	ESS, RAID, DB optimizer, virus management	eLiza, SMART/LEO, Software rejuvenation	More of the same and better

TECNOLOGIES DEVELOPED BY IBM

- **Log and Trace Tool**
- **Agent Building and Learning Environment (ABLE)**
- **Monitoring Engine**
- **Business Workload Manager**

OTHER RELATED TECHNOLOGIES

- **OGSA (Open Grid Systems Architecture) standard**
- **Generic Log Adapter**

thank you...

www.seminarlab.com

www.seminarlab.com