Implementing Autonomic Elements

The devil lies in the details ...

Standards, data and control integration, interfaces, endpoints, services, SOA ...

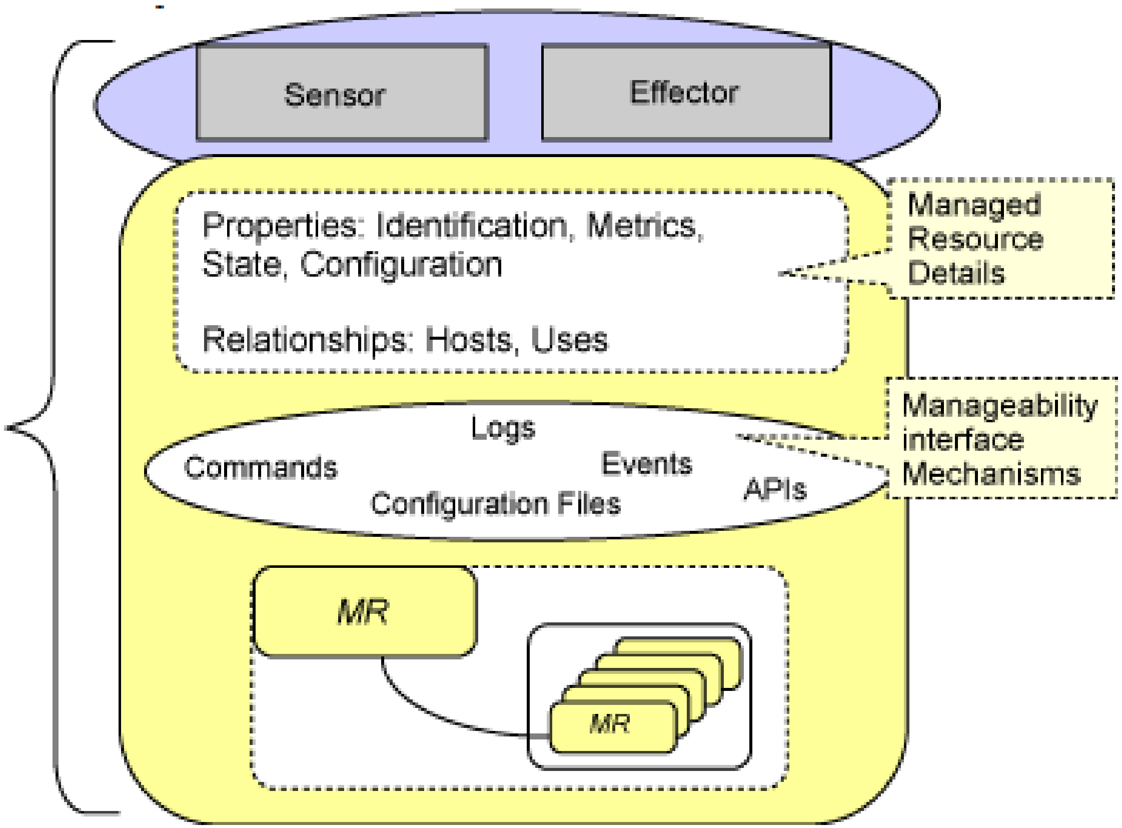
Manageability Endpoint and Interface

- Manageability Endpoint (ME)
  - A manageability endpoint—previously called a touchpoint—is the component in a system that exposes the state and management operations for a resource in the system. An autonomic manager communicates with a manageability endpoint through the manageability interface.
- Manageability Interface (MI)
  - A manageability endpoint is the implementation of the manageability interface for a specific manageable resource or a set of related manageable resources.
- Standards-based management interfaces and data formats

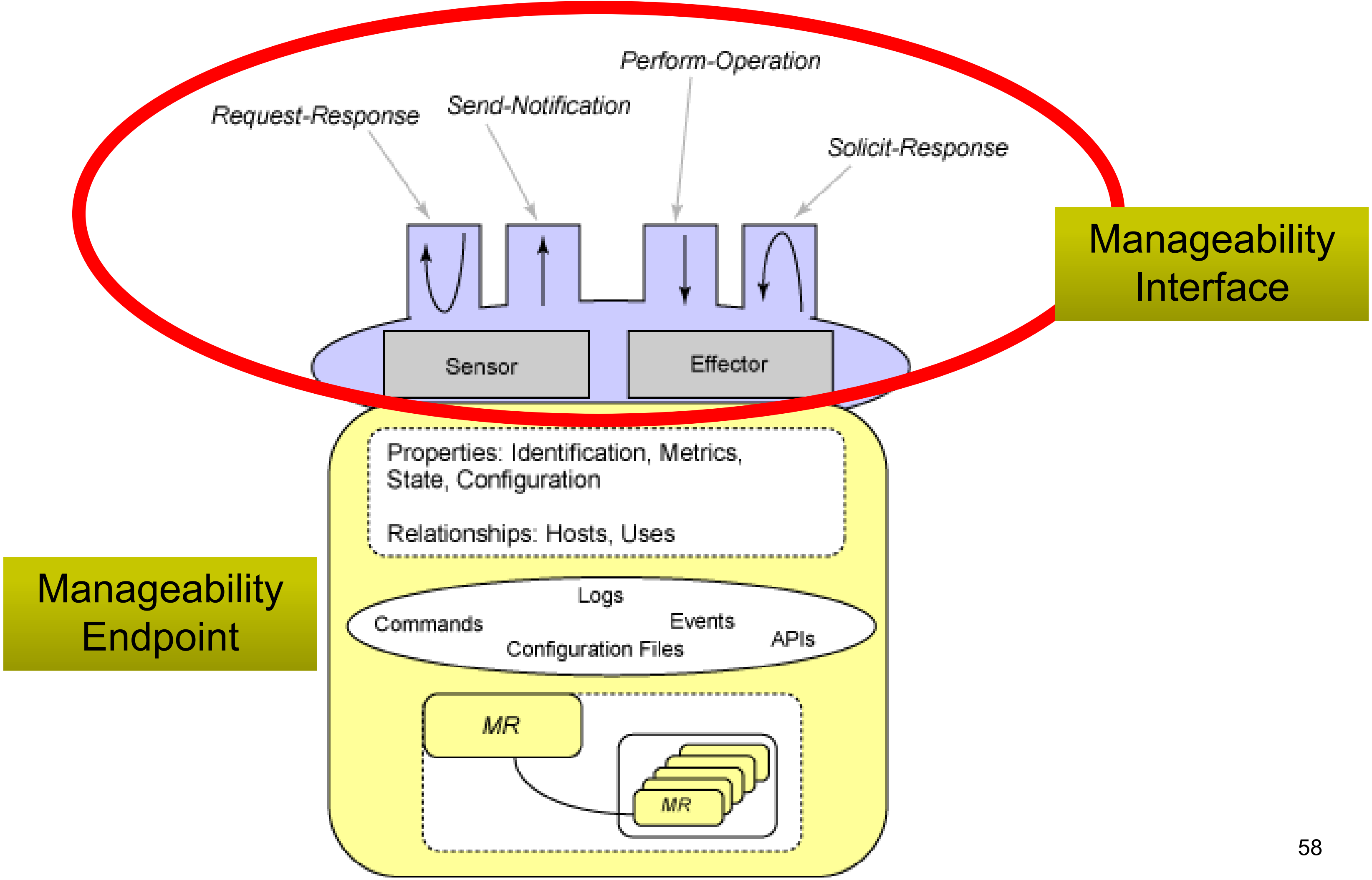
IBM: An Architecture Blueprint for Autonomic Computing, 4th Ed. 2006

Manageability Endpoint

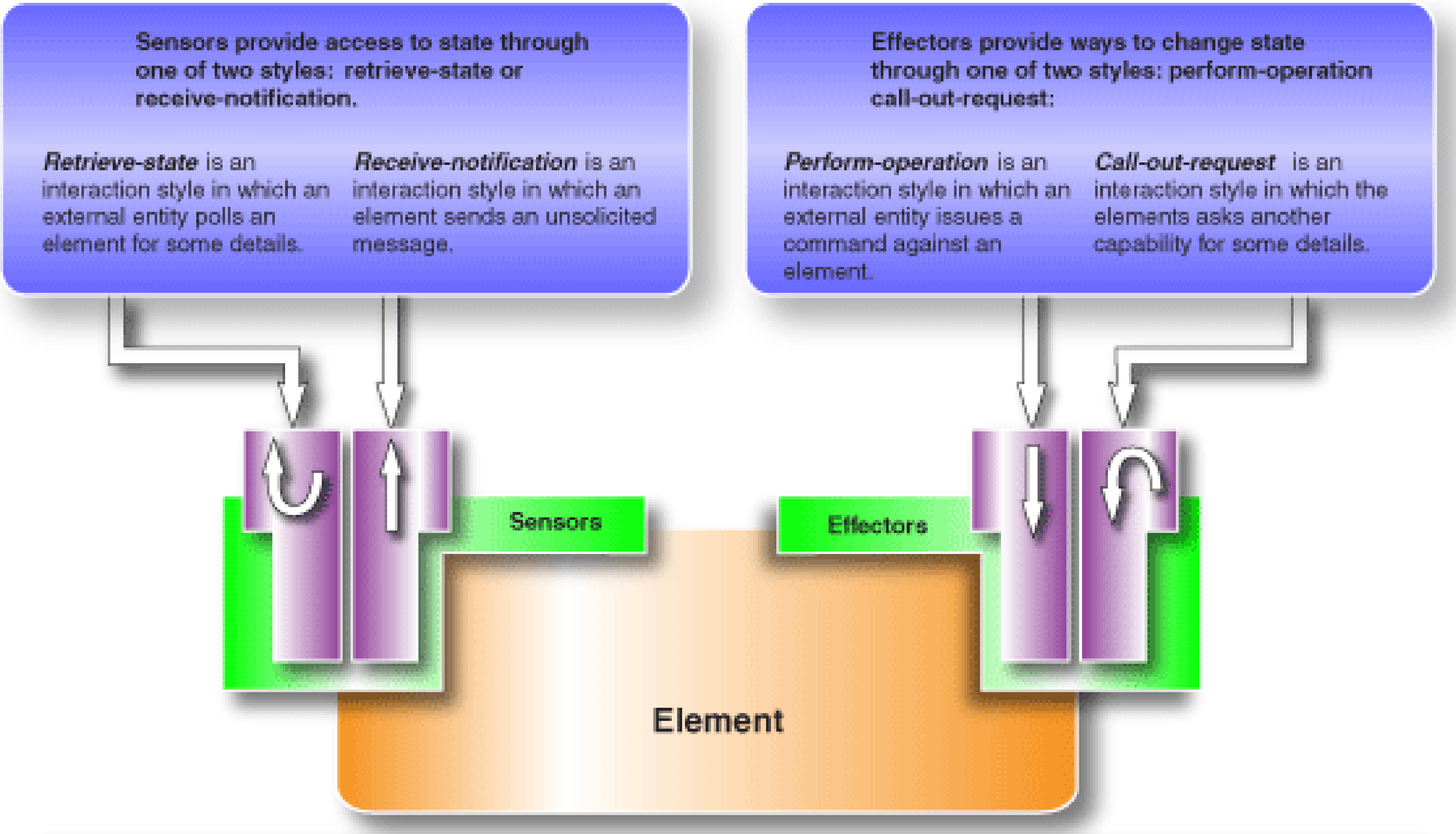
Manageability Endpoint


Sensor
Effector
Properties: Identification, Metrics, State, Configuration
Relationships: Hosts, Uses
Managed Resource Details
Logs
Commands
Events
Configuration Files
APIs
Manageability interface Mechanisms
MR
MR

Manageability Interface



Manageability Interface Interaction Styles



Sensors provide access to state through one of two styles: retrieve-state or receive-notification.

Retrieve-state is an interaction style in which an external entity polls an element for some details.

Receive-notification is an interaction style in which an element sends an unsolicited message.

Effectors provide ways to change state through one of two styles: perform-operation call-out-request:

Perform-operation is an interaction style in which an external entity issues a command against an element.

Call-out-request is an interaction style in which the elements asks another capability for some details.

IBM: Autonomic Computing Toolkit Developer's Guide (2004)

Manageability Interface Interaction Styles

Sensor retrieve-state

- Used by an AM to query state information from an ME
- The AM asks for information and the ME synchronously returns it

Sensor receive-notification

- A ME uses this style to asynchronously send event information to an AM

Effector perform-operation

- Used by an AM to issue a command to an ME
- Used to change states or properties in the endpoint

Effector call-out-request

- Used by a ME to consult with an external entity before taking certain actions—to check what changes are allowed prior to changing values
- Used to gather information from an AM before making a change

Manageability Endpoint Infrastructure

- The IBM Manageability Endpoint Builder
  - Includes tools and a run-time environment for building endpoints that allow products to expose manageability interfaces using the WSDM standard
  - Is accessible from a standard Eclipse environment and from the IBM Rational Application Developer product
  - With this interface, any WSDM-compliant tool or autonomic manager can view the status of the resource and make calls to modify the resource's state
- The IBM Manageability Endpoint Simulator
  - Assists in the development of autonomic managers by emulating a WSDM-compliant managed resource
  - A major hurdle in building autonomic managers is that developers need resources (and endpoints) with which to test
  - The Manageability Endpoint Builder addresses this problem.

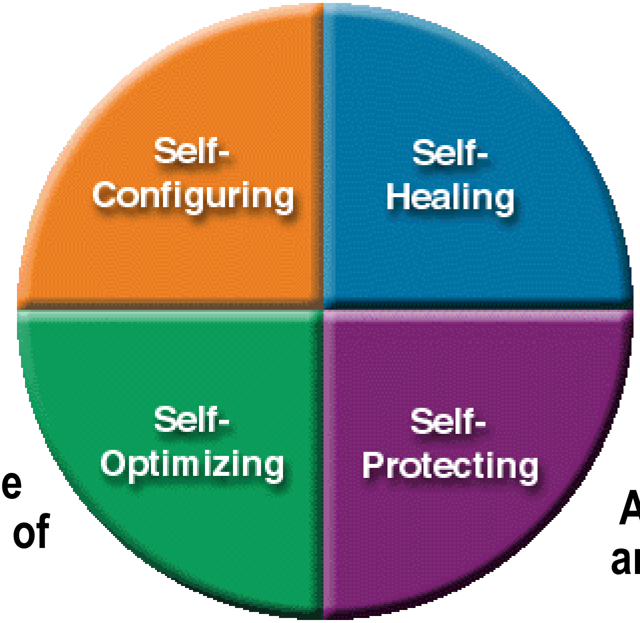
Autonomic Managers Implement Self-* MAPE-K Loops

Increased Responsiveness

Adapt to dynamically changing environments

Operational Efficiency

Tune resources and balance workloads to maximize use of IT resources



Business Resiliency

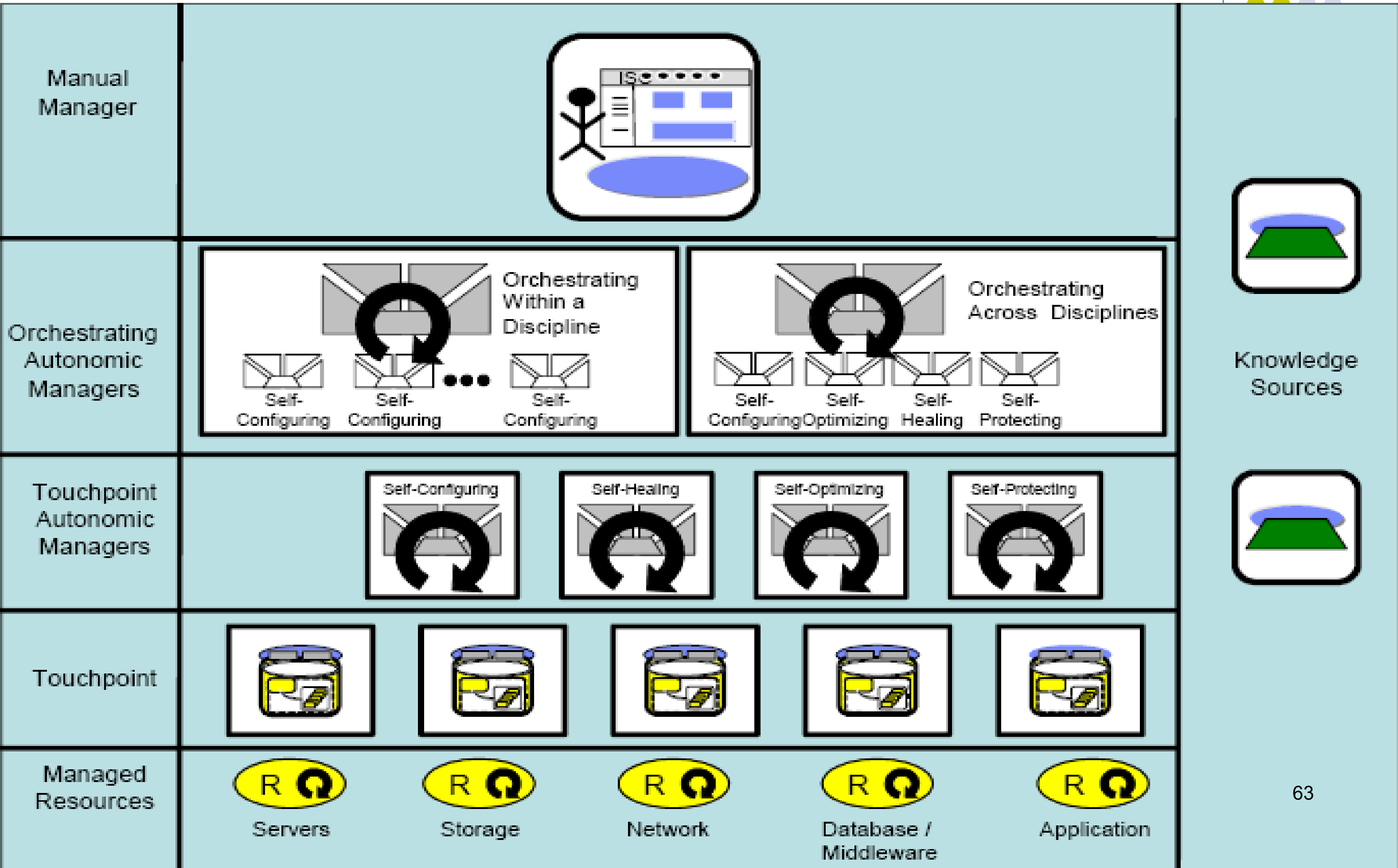
Discover, diagnose, and act to prevent disruptions

Secure Information and Resources

Anticipate, detect, identify, and protect against attacks

*Self – **

ACRA
AC Reference Architecture



ACRA: Autonomic Computing Reference Architecture

Level 5—highest

- Manual manger who operates a common system management interface

Level 4

- Autonomic Managers to integrate and orchestrate several self-* capabilities for a particular domain (e.g., DB, weather station)
- Implements system-wide capabilities

Level 3

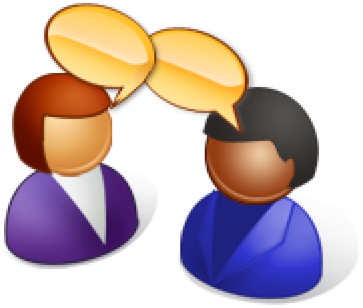
- Implements specific self-* using Autonomic Managers (AM)

Level 2

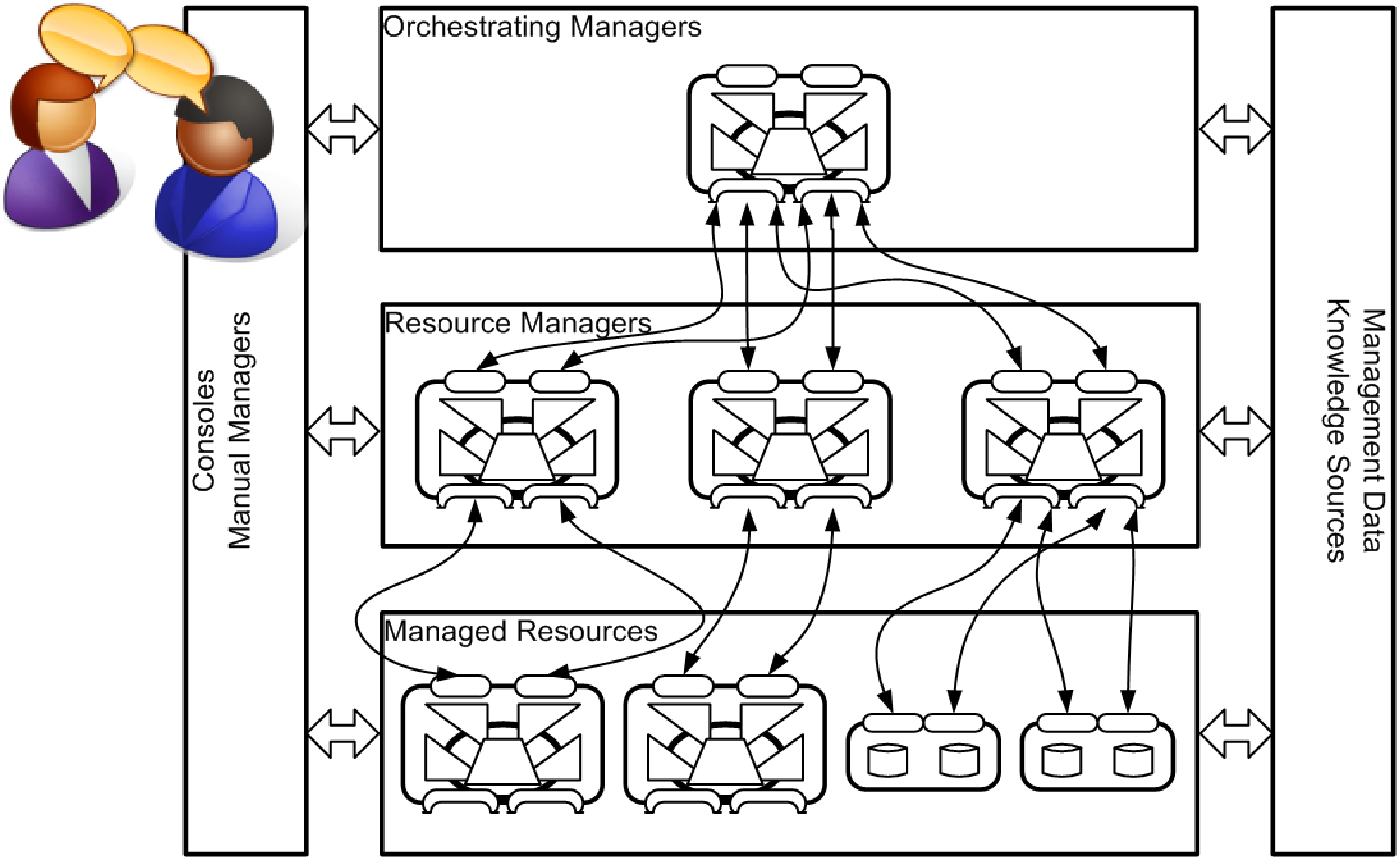
- Consistent, standard Manageability Interfaces (MI) for accessing and controlling the managed resources in a uniform manner
- The MIs are implemented with Manageability Endpoints (ME)

Level 1—lowest

- System components or managed resources (hardware, software) possibly with embedded self-management



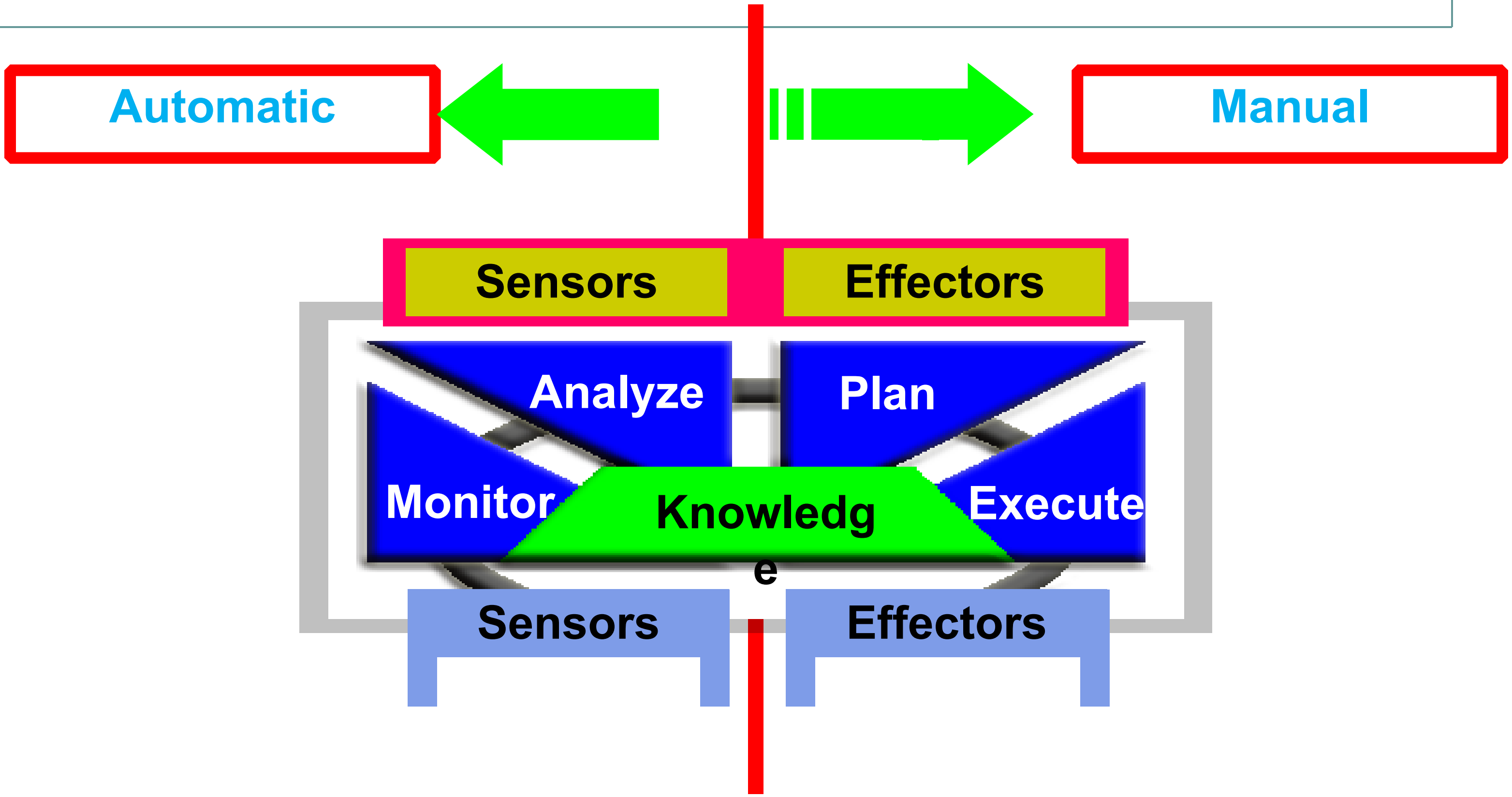
ACRA
AC Reference Architecture



Manual Manager

- Management or integrated solutions console
- Enables a human to perform and delegate management functions
- Collaborates with or orchestrates autonomic managers
- Set-up, configuration, run-time monitoring, control
- Manage trust—different levels of feedback
- Connecting knowledge source
- Specifying policies

Autonomic Manager



Automatic

Manual

Sensors

Effectors

Analyze

Plan

Monitor

Knowledge

Execute

Sensors

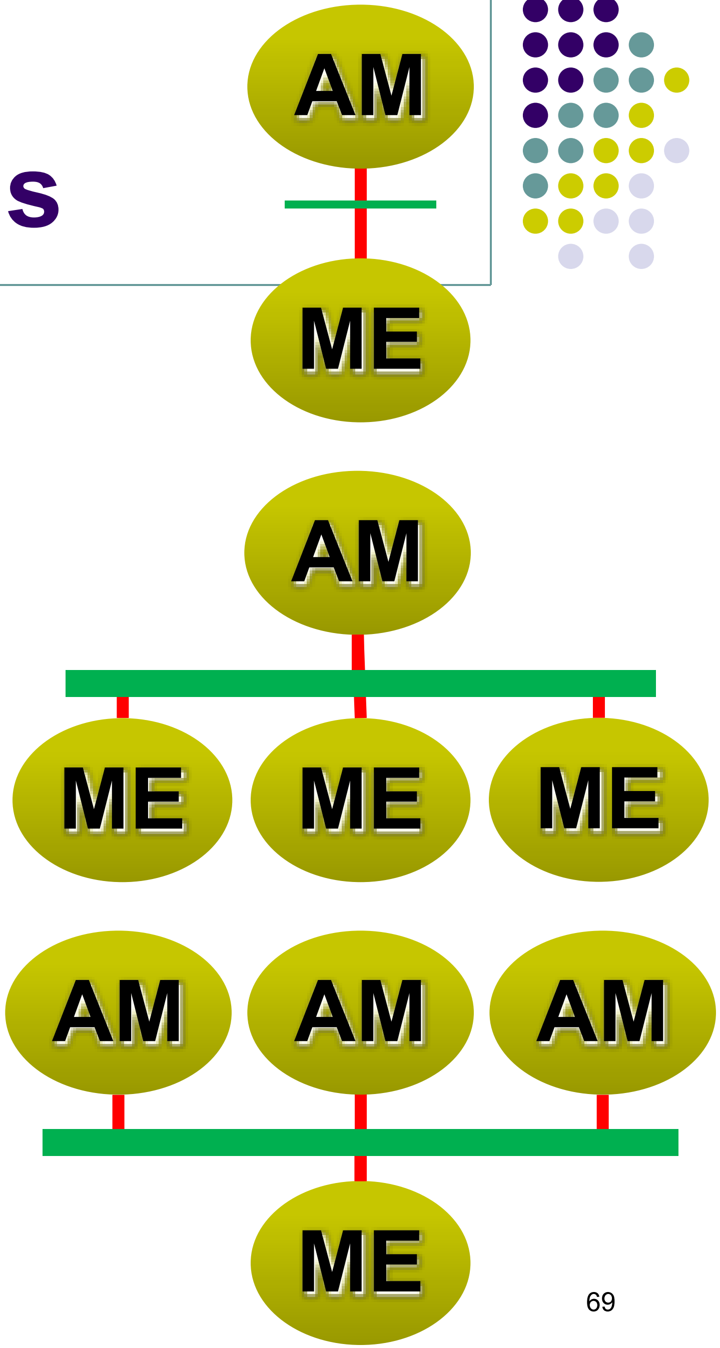
Effectors

Knowledge Base

- Repository in the form of a registry, dictionary, database
- An AM can load knowledge from multiple repositories or knowledge sources
- Data produced by an AM can be shared among AMs
- Knowledge stored in the repository has syntax and semantics for describing autonomic computing artifacts such as symptoms, policies, change requests, plans, execution scripts
- Standardization of syntax and semantics is critically important for the success of autonomic computing
  - Configuration Management Data Base (CMDB)
  - Logging Data Base

Enterprise Service Bus

- Connects and integrates various AC building blocks
  - Autonomic Managers (AMs)
  - Manageability Endpoints (MEs)
  - Knowledge repositories
  - Aggregating multiple manageability mechanisms for a single manageable resource
  - Facilitating one or more AMs to manage one or more MEs
- WSDM standard
  - Web Service Distributed Management



Useful Papers under Resources Course Web Site

- Ganek, A.G., Corbi, T.A.: The Dawning of the Autonomic Computing Era. IBM Systems Journal 42(1):5-18 (2003)
- Kephart, J.O., Chess, D.M.: The Vision of Autonomic Computing. IEEE Computer 36(1):41-50 (2003)
- Kluth, A.: Information Technology: Make It Simple. The Economist (2004)
- Huebscher, M.C., McCann, J.A.: A Survey of Autonomic Computing—Degrees, Models, and Applications. ACM Computing Surveys, 40 (3):7:1-28 (2008)
- Müller, H.A., Kienle, H.M., Stege, U.: Autonomic Computing: Now You See It, Now You Don’t—Design and Evolution of Autonomic Software Systems. In: De Lucia, A.; Ferrucci, F. (eds.): Software Engineering International Summer School Lectures: University of Salerno. LNCS, Springer-Verlag, Heidelberg, pp. 32–54 (2009)
- Dobson, S., Denazis, S., Fernandez, A., Gaiti, D., Gelenbe, E., Massacci, F., Nixon, P., Saffre, F., Schmidt, N., Zambonelli, F.: A Survey of Autonomic Communications. ACM Transactions on Autonomous and Adaptive Systems (TAAS) 1(2):223-259 (2006)

Useful Papers under Resources Course Web Site

- Diao, Y., Hellerstein, J.L., Parekh, S., Griffith, R., Kaiser, G.E., Phung, D.: A Control Theory Foundation for Self-Managing Computing Systems. IEEE Journal on Selected Areas in Communications 23(12):2213-2222 (2005)
- Müller, H.A., Pezzè, M., Shaw, M.: Visibility of Control in Adaptive System. In: 2nd ACM/IEEE International ICSE Workshop on Ultra-Large-Scale Software-Intensive Systems (ULSSIS 2008), pp. 23-26, ACM, New York, NY, USA (2008)
- Dawson, R., Desmarais, R., Kienle, H.M., Müller, H.A.: Monitoring in Adaptive Systems Using Reflection. In: 3rd ACM/IEEE International ICSE Workshop on Software Engineering for Adaptive and Self-Managing Systems (SEAMS 2008), pp. 81-88, ACM, New York, NY, USA (2008)
- OASIS: Web Services Distributed Management: Management of Web Services (WSDM-MOWS) 1.1 OASIS Standard (2006)
- OASIS: Web Services Distributed Management: Management Using Web Services (WSDM-MUWS) 1.1 OASIS Standard (2006)
- Kreger, H., Studwell, T.: Autonomic Computing and Web Services Distributed Management (2005)
- IBM: Symptoms Reference specification Version 2.0 (2006)

Useful Papers under Resources Course Web Site

- Study these papers
- Immerse yourself in the autonomic computing literature and technology