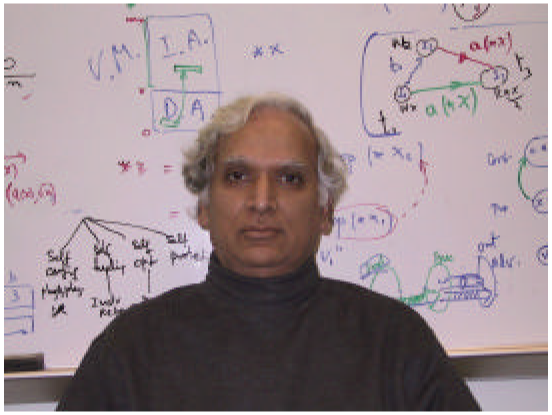
Autonomic Computing

Pratap Pattnaik
T. J. Watson Research Center
Yorktown Heights
IBM

Coworkers

K. Ekanadham (Eknath)



Joefon Jann

Goal of Autonomic Computing

reduction of complexity in the management of large computing systems

My Bias

- Where you stand depends on where you sit.
- What you see depends on where you stand.

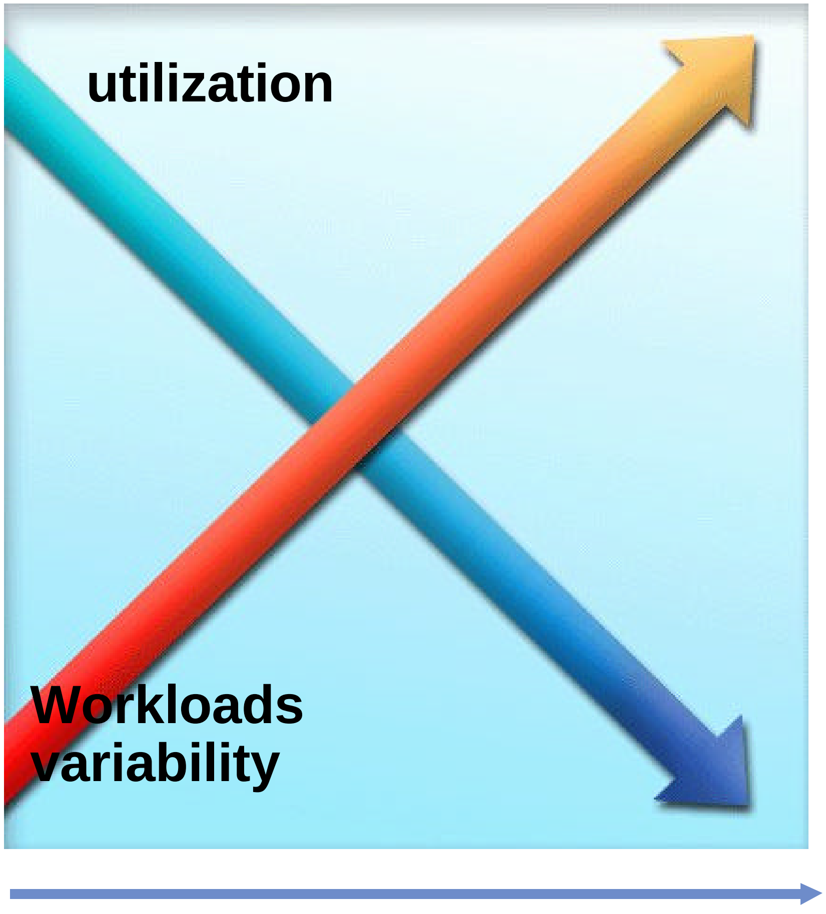
My experience is based on many years of working with our leading customers, and in developing systems (architecture, prototype and deployment) for their needs.

Today's business sees:

- Demand for Short Term Cost Savings.
- Explosion of Transactions through the IT Infrastructure.

Infrastructure Management

- Complexity
- Cost of software & services
- Skills shortage
- New kinds of workloads



Demand for Short Term Cost Savings.
Explosion of Transactions through the Infrastructure.

HOW DOES THIS COMPLEXITY ARISE?

More degrees of Freedom

$\Rightarrow$ More choices to be made

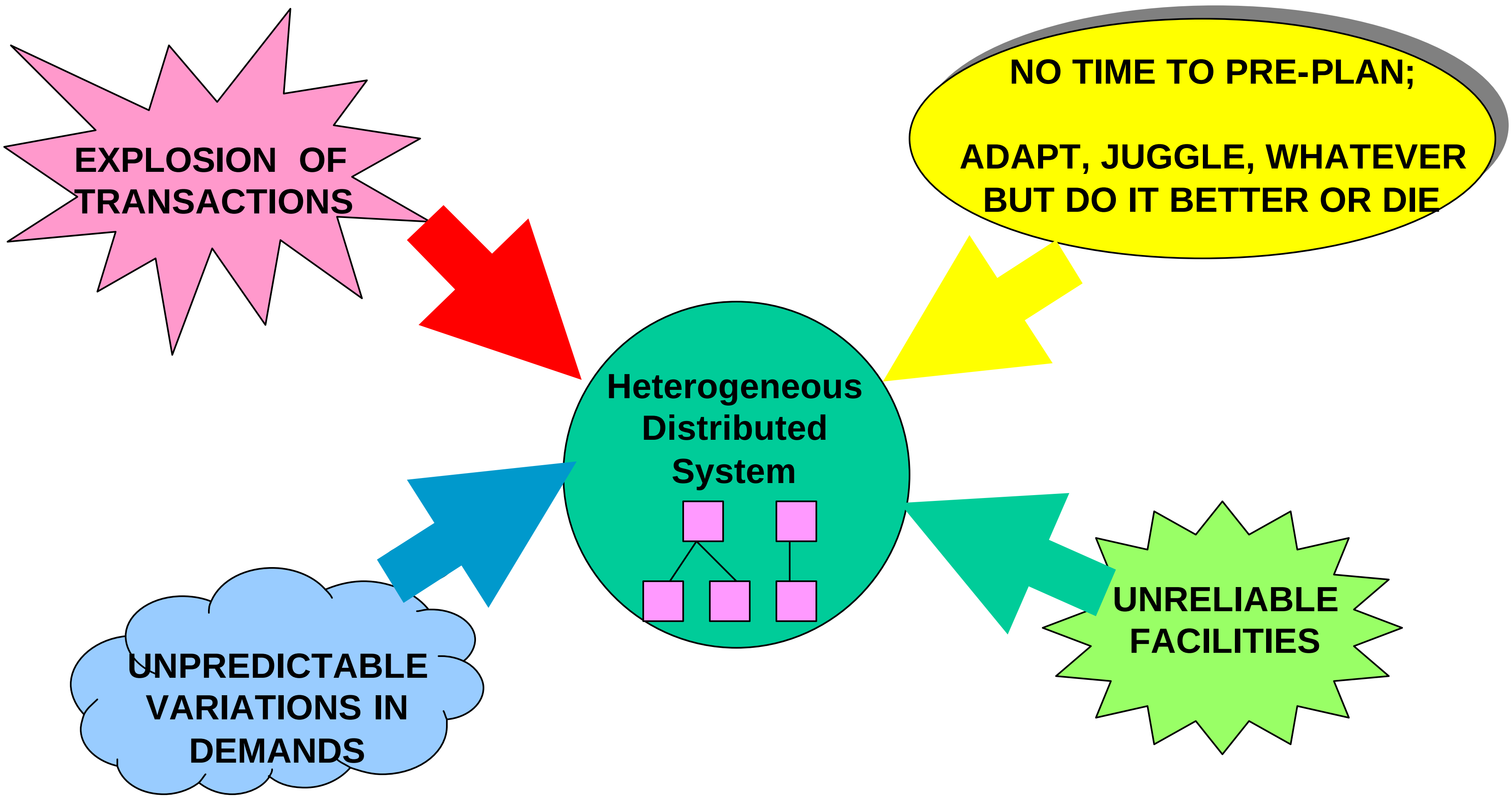
$\Rightarrow$ More information to be collected/sorted out

$\Rightarrow$ More criteria to be applied for optimality

$\Rightarrow$ More complexity

Demand for Short Term Cost Savings.
Explosion of Transactions through the Infrastructure.

QUINTESSENTIAL FEATURES



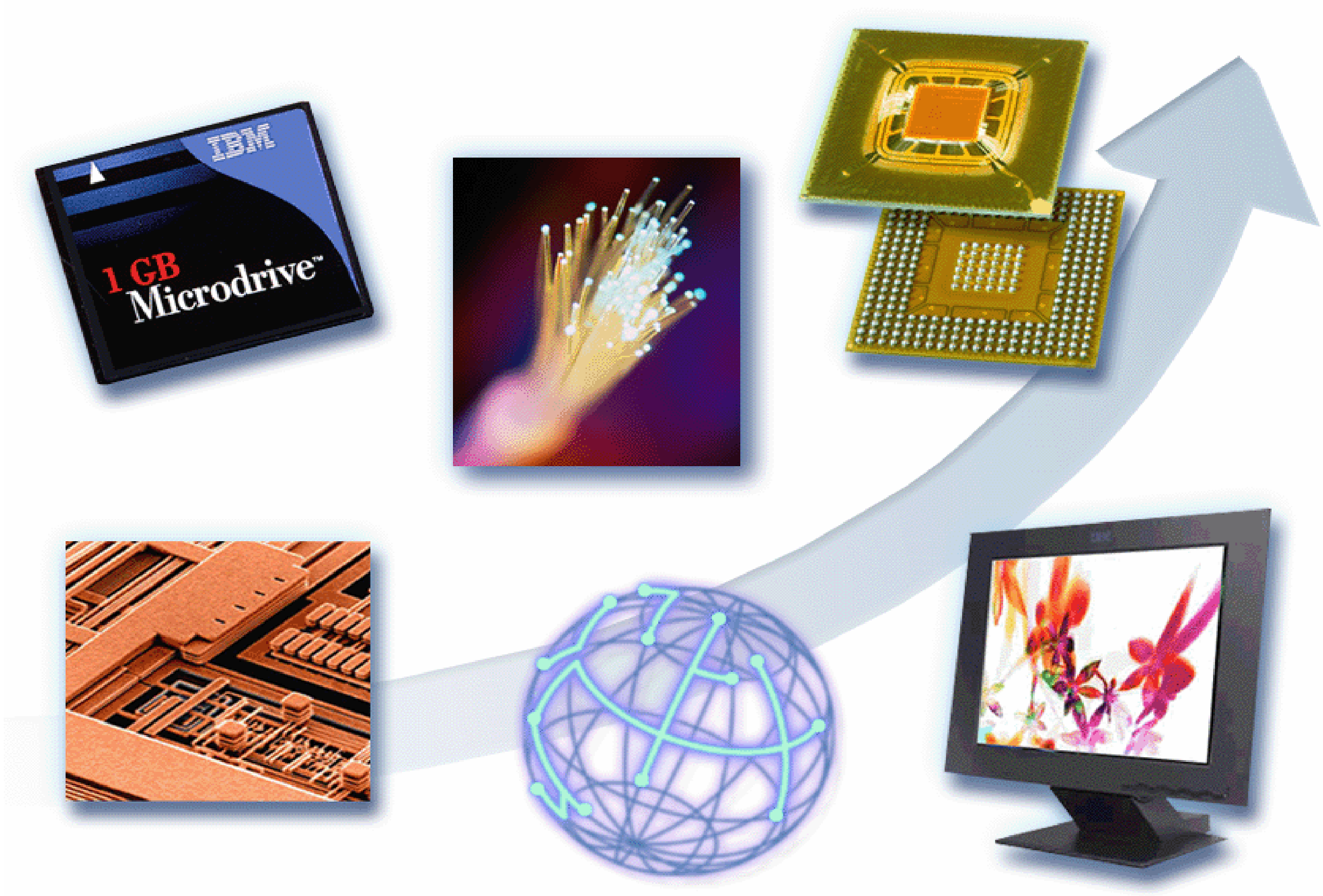
EVER-CHANGING GLOBAL KNOWLEDGE

1. **Non-uniform technological growth**
2. **Transactional explosion**
3. **New trends induced by growing society**

let us look at each of these aspects ……

1. Non-uniform technological growth

Technology Continues to Advance, But ....



Non uniform advancements of the components

Historically, computing elements, such as CPUs, memory, disks, network, etc., have non-uniform advancements.

For Example

- **HW (Moore's Law on speed and real estate)**
- **Firmware (Dynamic Reconfiguration, LPAR, Hypervisor etc.)**
- **OS (QoS, Scalable, etc.)**
- **Middleware (New runtime environments)**
- **Applications**

… Consequently

- This eliminates the possibility of developing a Stable knowledge base.
- The disparity between the capabilities/speeds of various elements opens up the opportunity for each element to introduce a number of different strategies depending upon the environment the element is encountering.

This lack of knowledge manifests as effective increase in the degrees of freedom.

Demand for Short Term Cost Savings.
Explosion of Transactions through the Infrastructure.

2. Transactional explosion through IT infrastructure

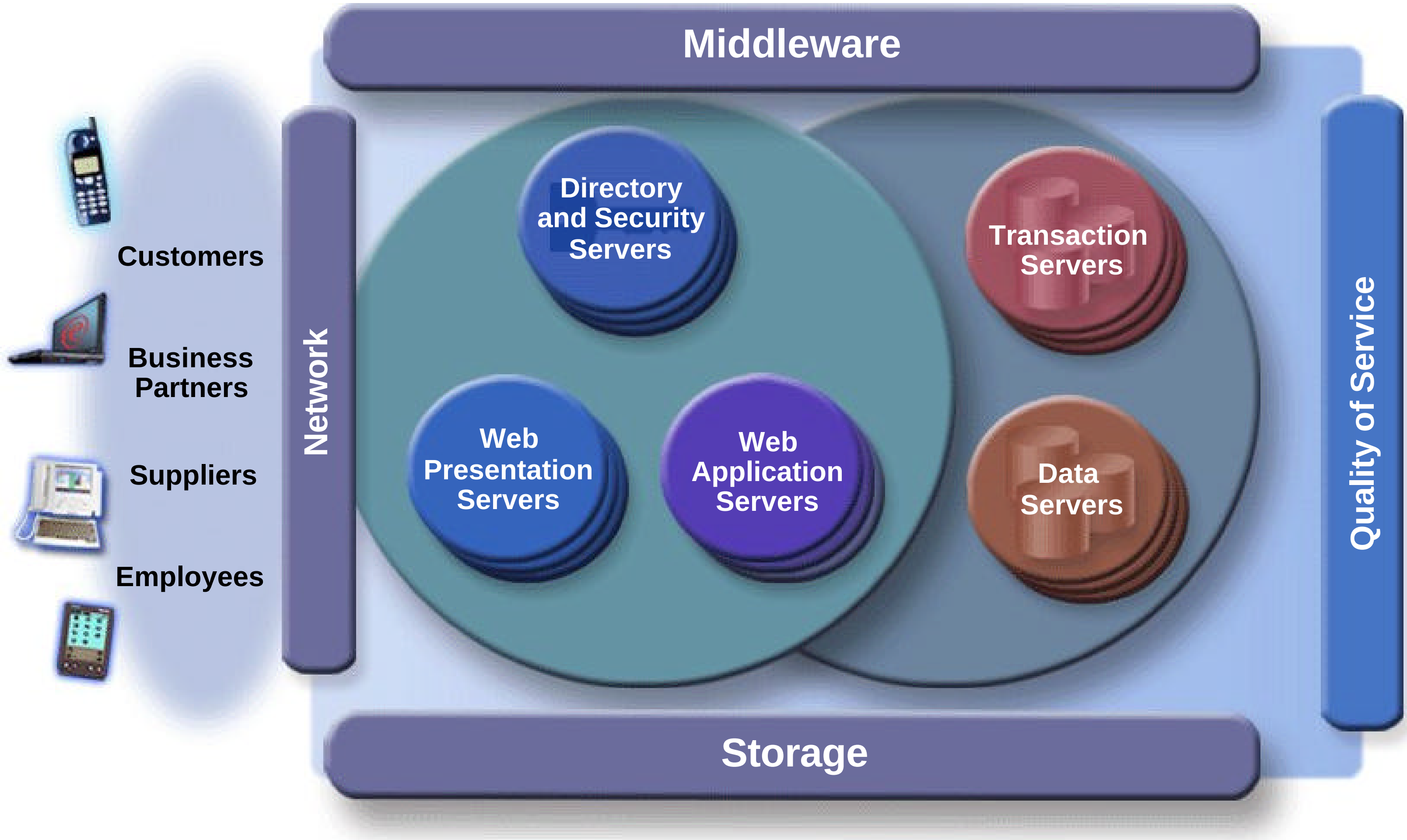
In today's IT infrastructure, processing a single transaction, explodes into numerous transactions

This will only continue to evolve this way

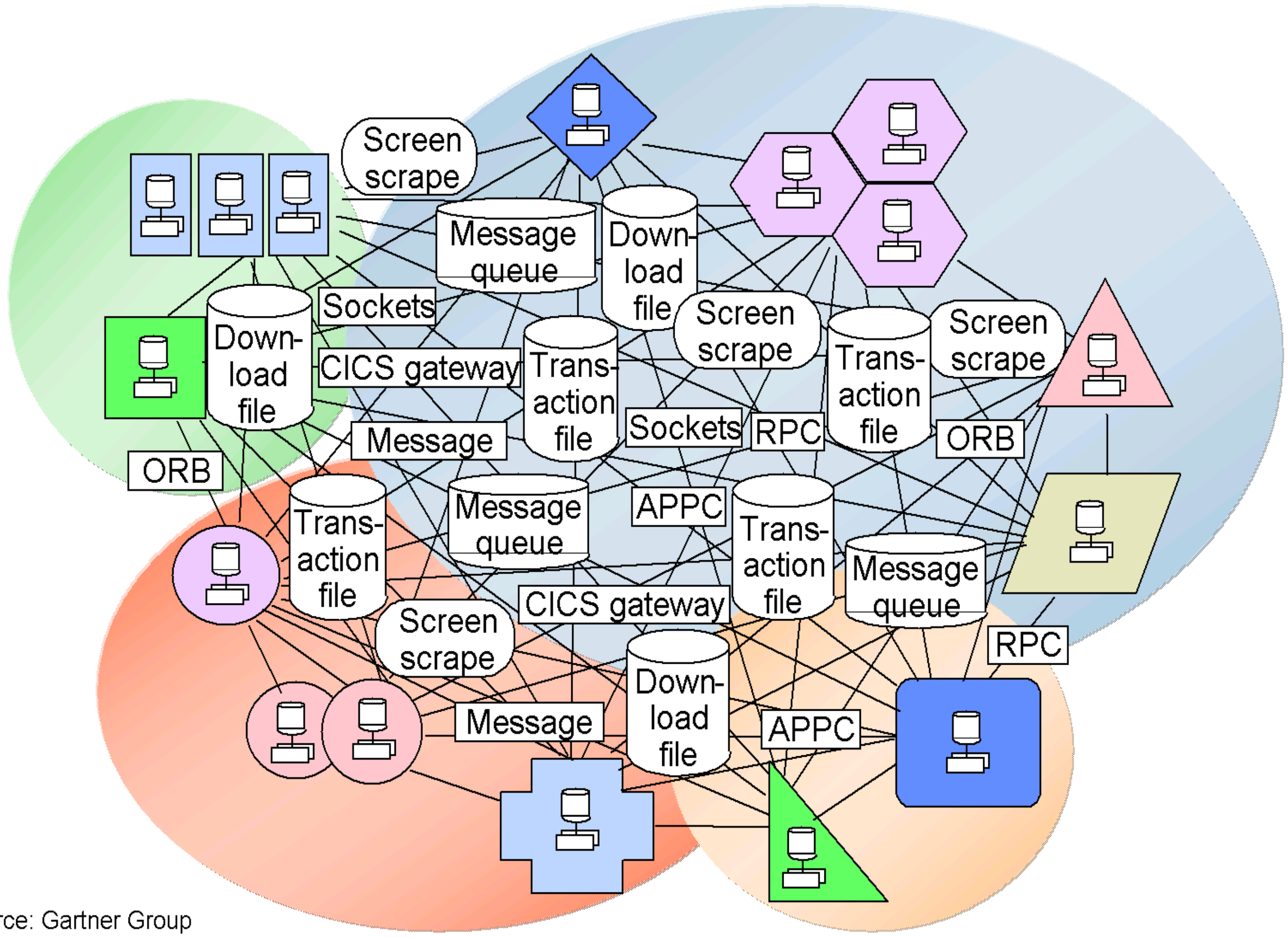
To see this, just visit a web site and see how many activities take place to show the contents

Demand for Short Term Cost Savings.
Explosion of Transactions through the Infrastructure.

Example 1: e-business Infrastructure



Example 2: Transaction flow



Source: Gartner Group

3. New trends induced by society

- Society is becoming increasingly data-centric
- Greater role of information transfer in human interactions
- Increasing Variability in demand for services

EVER-CHANGING GLOBAL KNOWLEDGE

1. **Non-uniform advancements of the components**
2. **Transactional explosion through IT infrastructure**
3. **Variability due to increased interaction with the society**

Þ Cannot model based on extant knowledge (static)

Þ Must adapt to dynamically changing environment

How can we reduce the complexity of management of large computing systems ?

Observations

- Large Systems will force the lack of comprehensive global knowledge.
- Business needs and opportunities will continuously encourage uncoordinated growth. (In the past, a number of major inventions in CS have come from such dynamics).
- Systems will remain as conglomerations of several distributed components.
- It is desirable to develop systematic framework (theory, infrastructure, best practices, heuristics, etc.) to reduce the effective complexity of the IT infrastructure.
- Business constraints will expect the new frameworks to utilize significant portions of existing IT infrastructure.

ANALOGY WITH PHYSICS

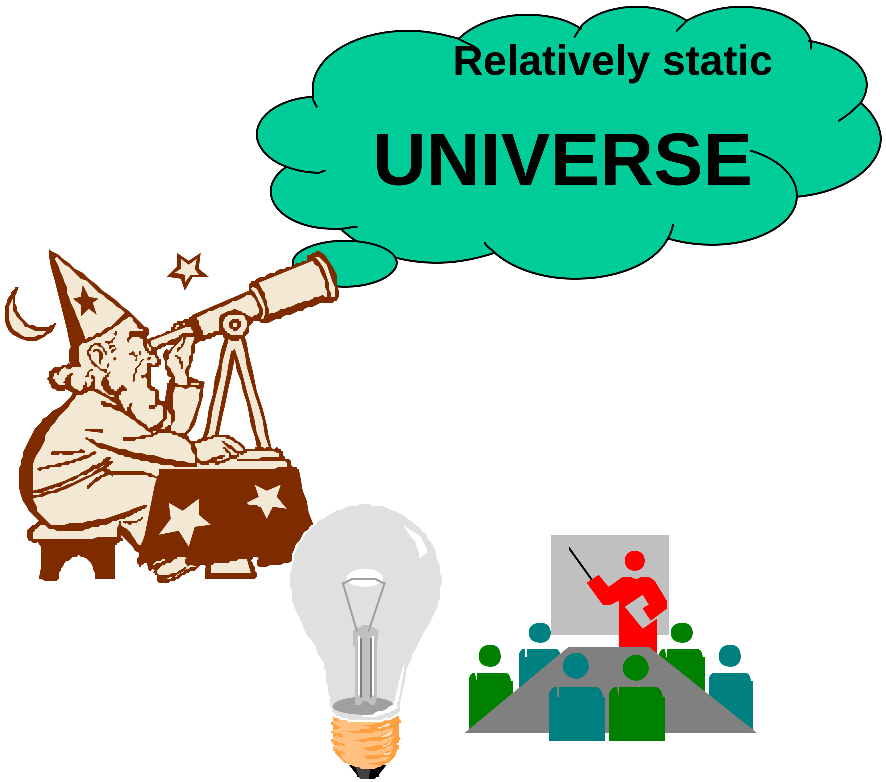
Complexity arising out of a vast number of degrees of freedom has been a major source of challenge in Physics.

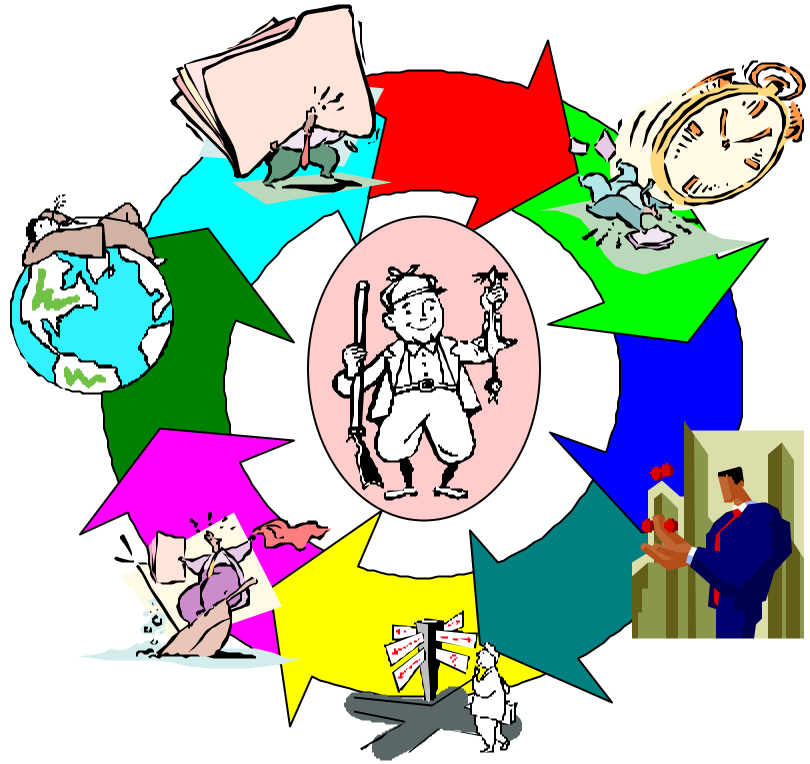
A successful approach in Physics has been to decompose a system into a set of appropriate components, and develop behavioral models for each component by reduction - that is,

> Design a small number of composite degrees of freedom that capture the effects of others in some relevant manner

This analogy may not completely carry over to Computer Science .., but may give us some guidance.

SCIENCE VS. COMMERCE

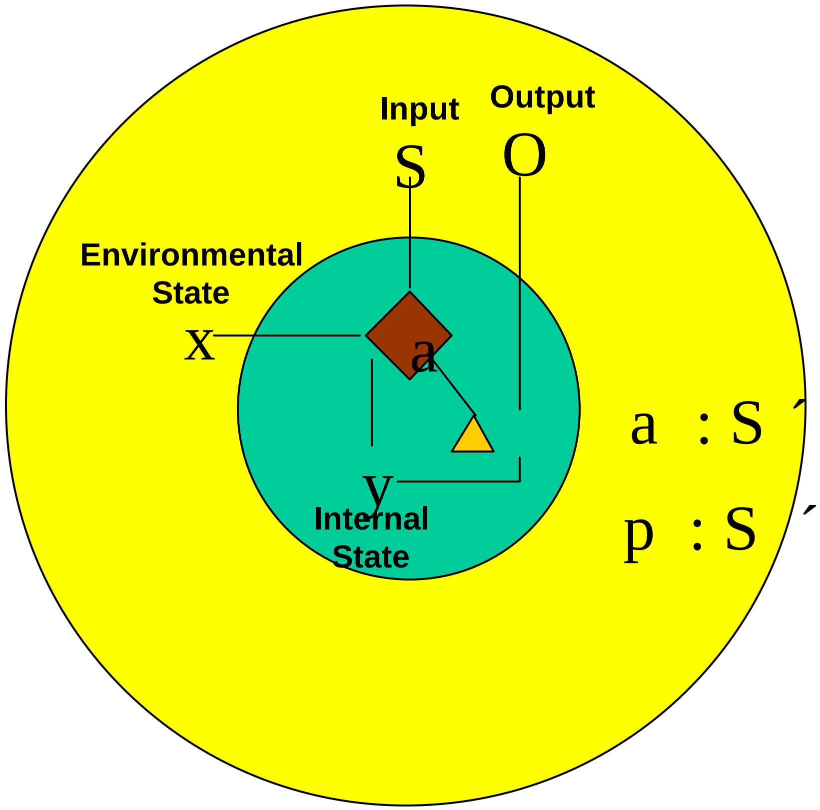




DISCOVER, MODEL, VERIFY, EXPLAIN A UNIVERSE WHICH DOES NOT CHANGE RULES ON YOU

TRY EXPLAINING A BYZANTINE PHENOMENON WHERE THE VALUE OF YOUR GOAL IS CHANGED ON YOU AS YOU IMPROVE

FORMAL SYSTEM PARAMETERS



a : S ´ x ´ y ® P

p : S ´ y ® O ´ y

[S, O, b, x, y, P, a, h, **Q**]

BehaviorSpec, StateSpec, MethodSpec, StrategySpec

Example Malloc

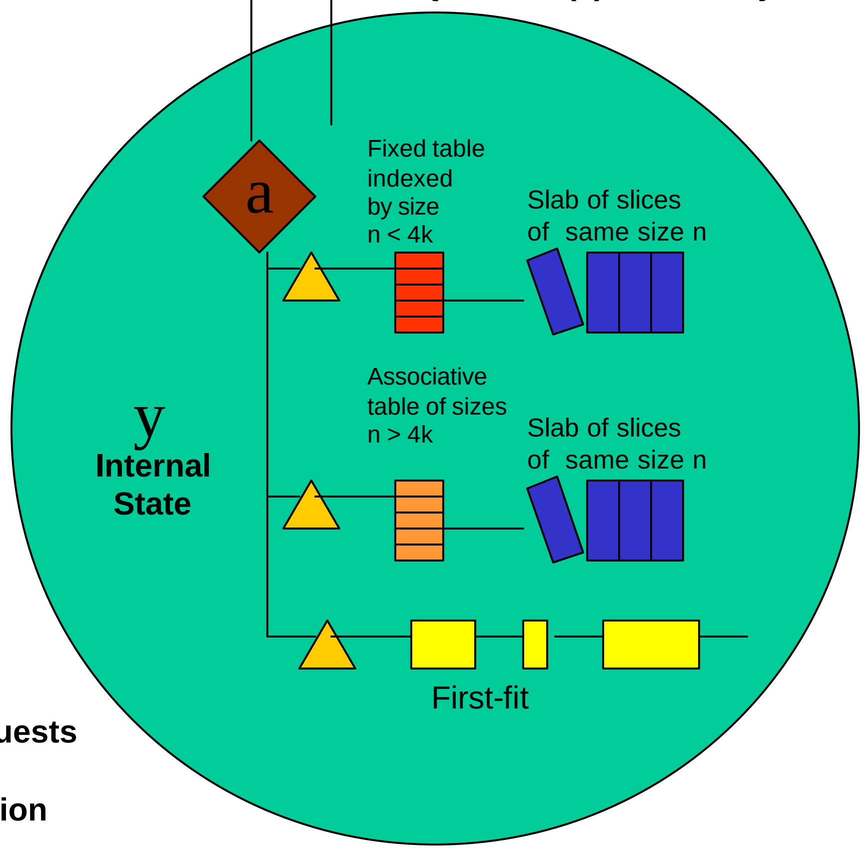
S = {Allocate(n), free(a)}

O = {Address(a), ok/error}

b (n,a) Þ {Addresses a thru a+n are currently free}

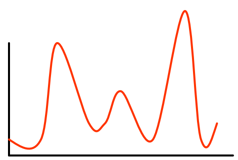
h Þ quick response time

searching linear list does not scale well



On average for many applications n < 4k, 60% of the time

For n > 4k, exploit temporal locality for frequent requests of same size



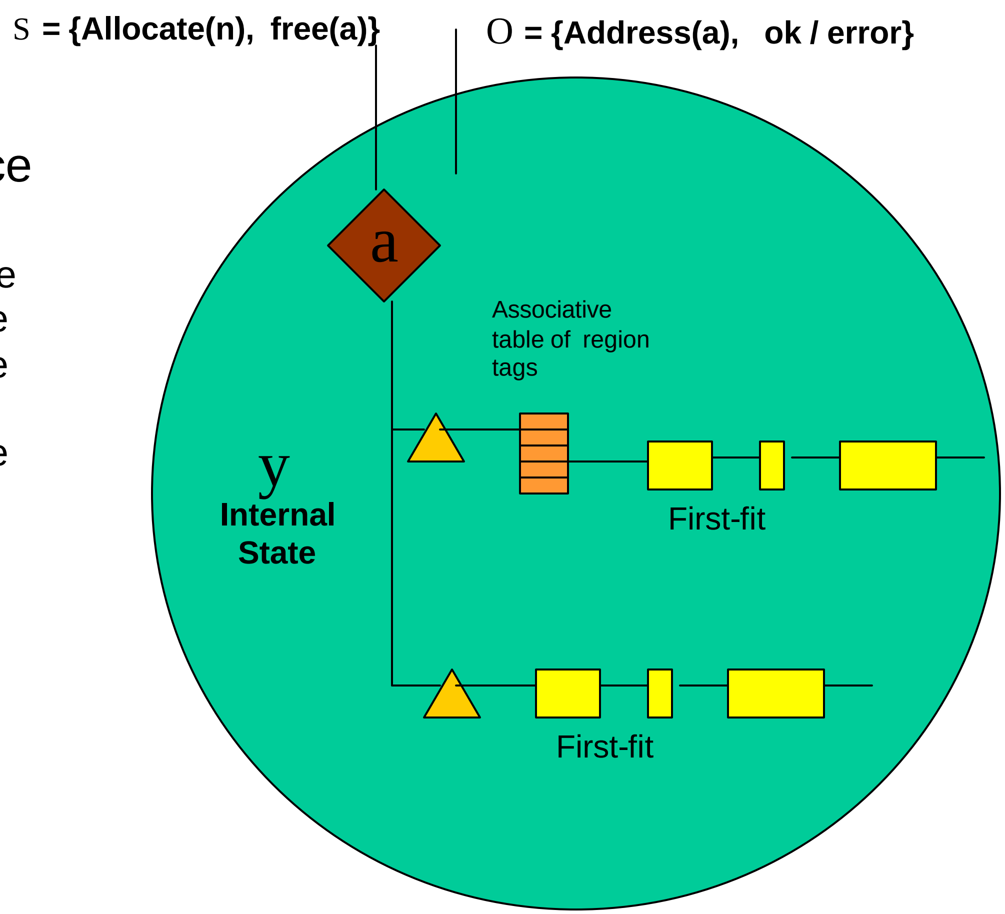
Track history

x is the pattern of arriving requests
It is extrapolated from history
and corrected by self-observation

Example Malloc (contd.)

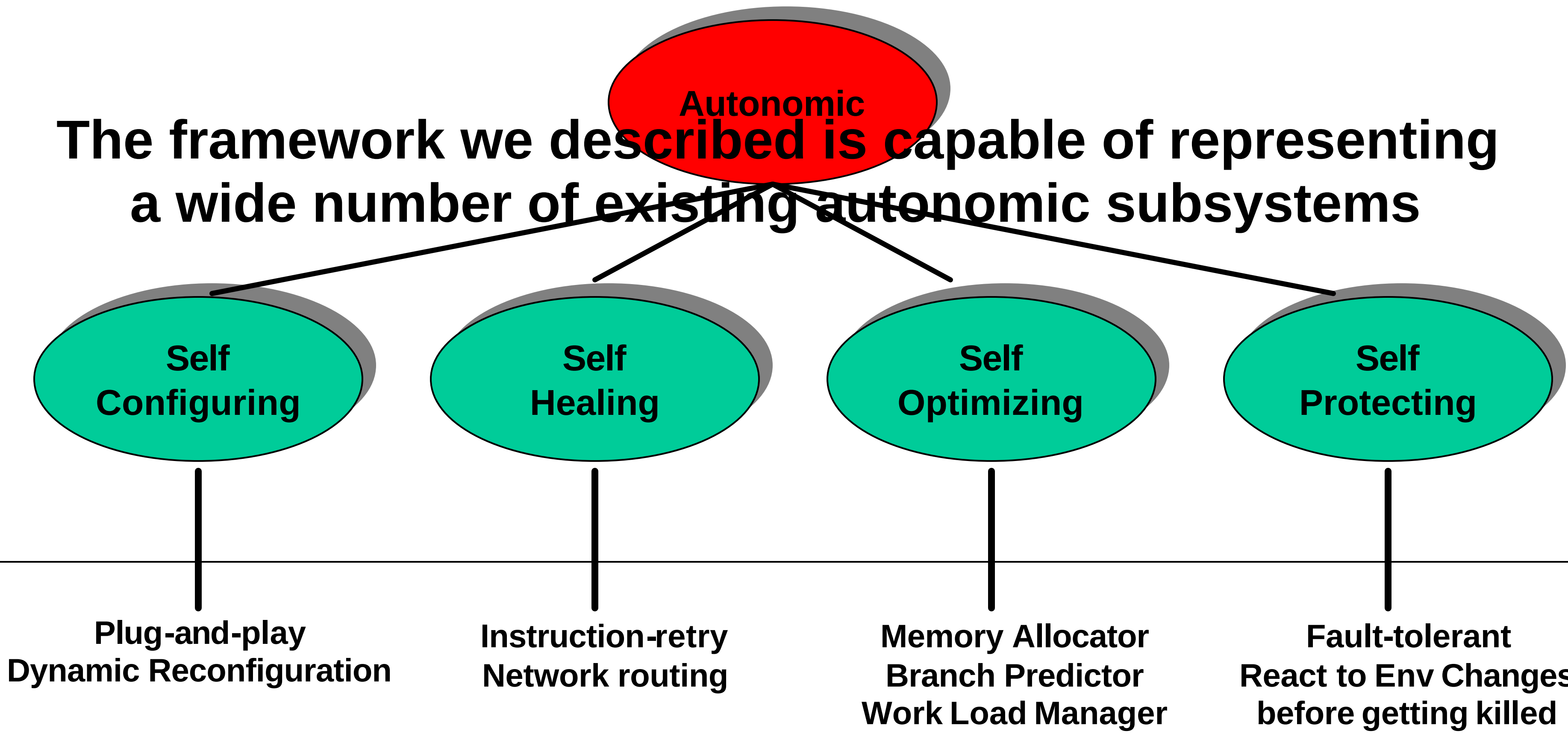
TLB Performance

Malloc Outer Structure
- Malloc substructure
- Malloc substructure
- …..
- Malloc substructure



OTHER EXAMPLES

The framework we described is capable of representing a wide number of existing autonomic subsystems



What needs to be done?

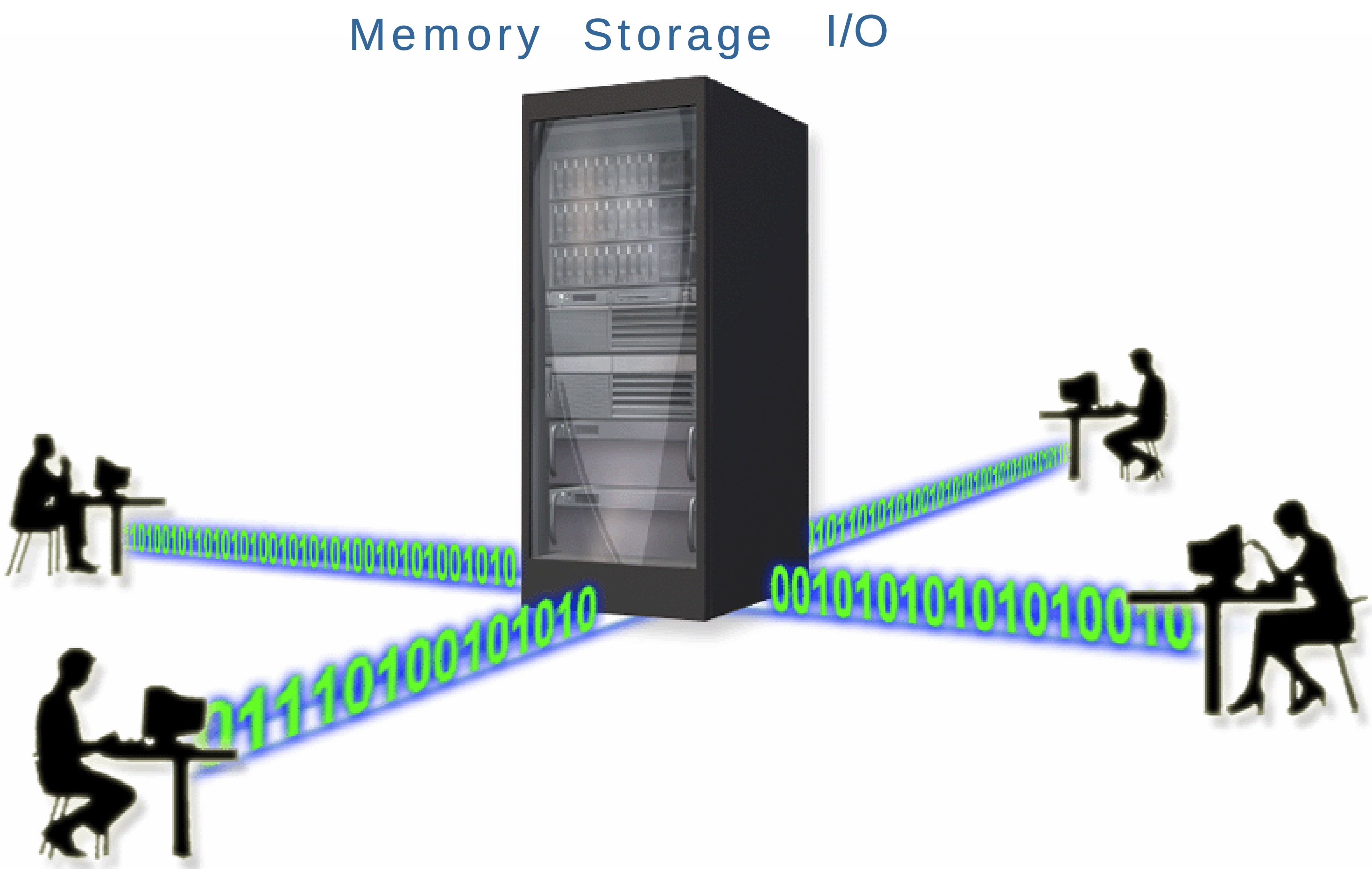
- **A theory must be developed to study common properties of any system expressed in this parameterized framework**
- **For example**
  - **Canonical representations for** x **and** y
  - **Monitoring and Sampling criteria**
  - **Concise representations for observed Characteristics**
  - **Stability of switching among different strategies**
  - **Effects of latencies between observations and resource changes**

Migration from an entitlement-approach to goal-oriented approach

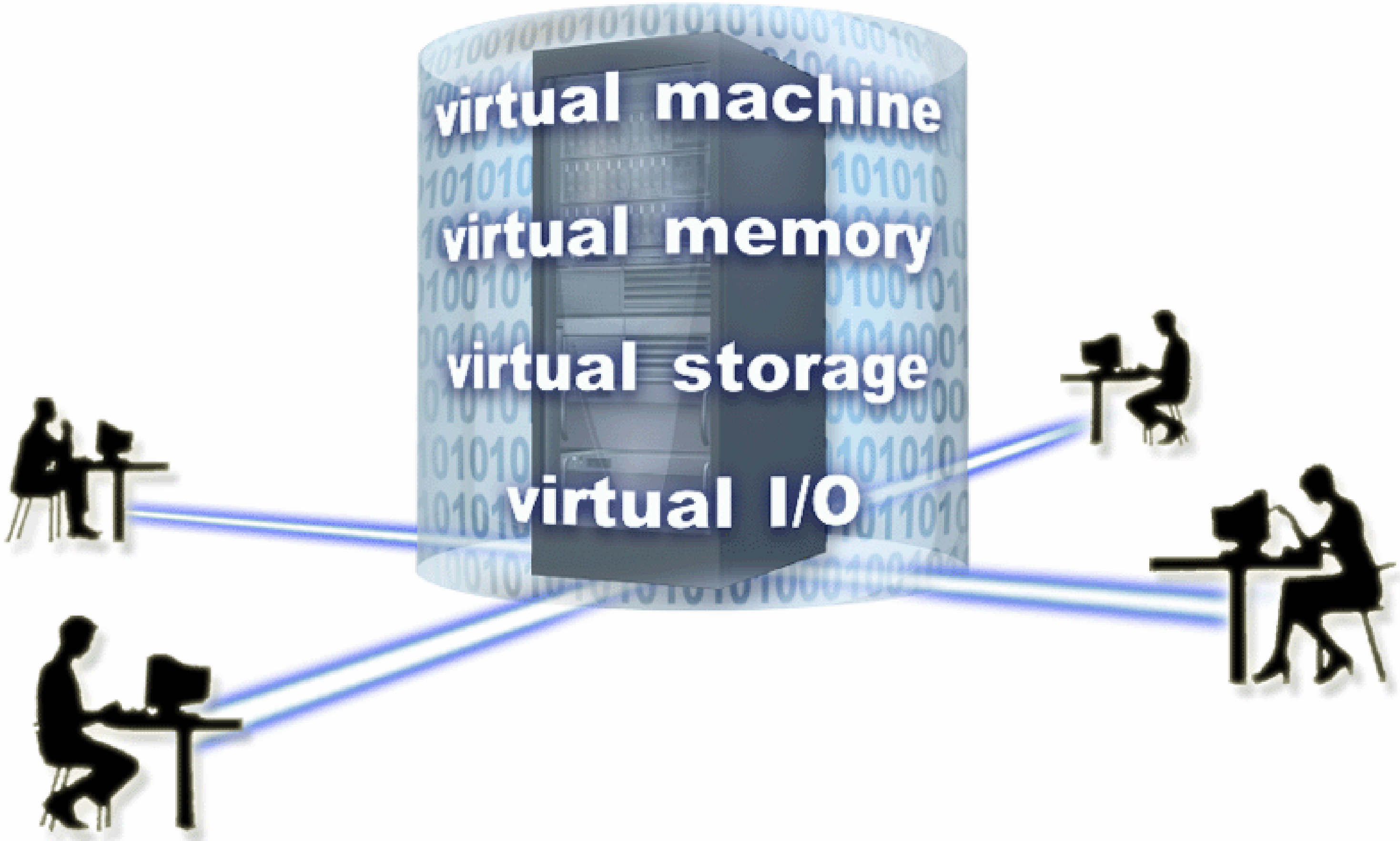
- Represent systems as services
- Attach QOS to the services
- Multi-tasking ➔ Virtual Machine ➔ OGSA

Hiding Complexity

Physical Computing Resources



Hiding Complexity
Virtual Computing Resources

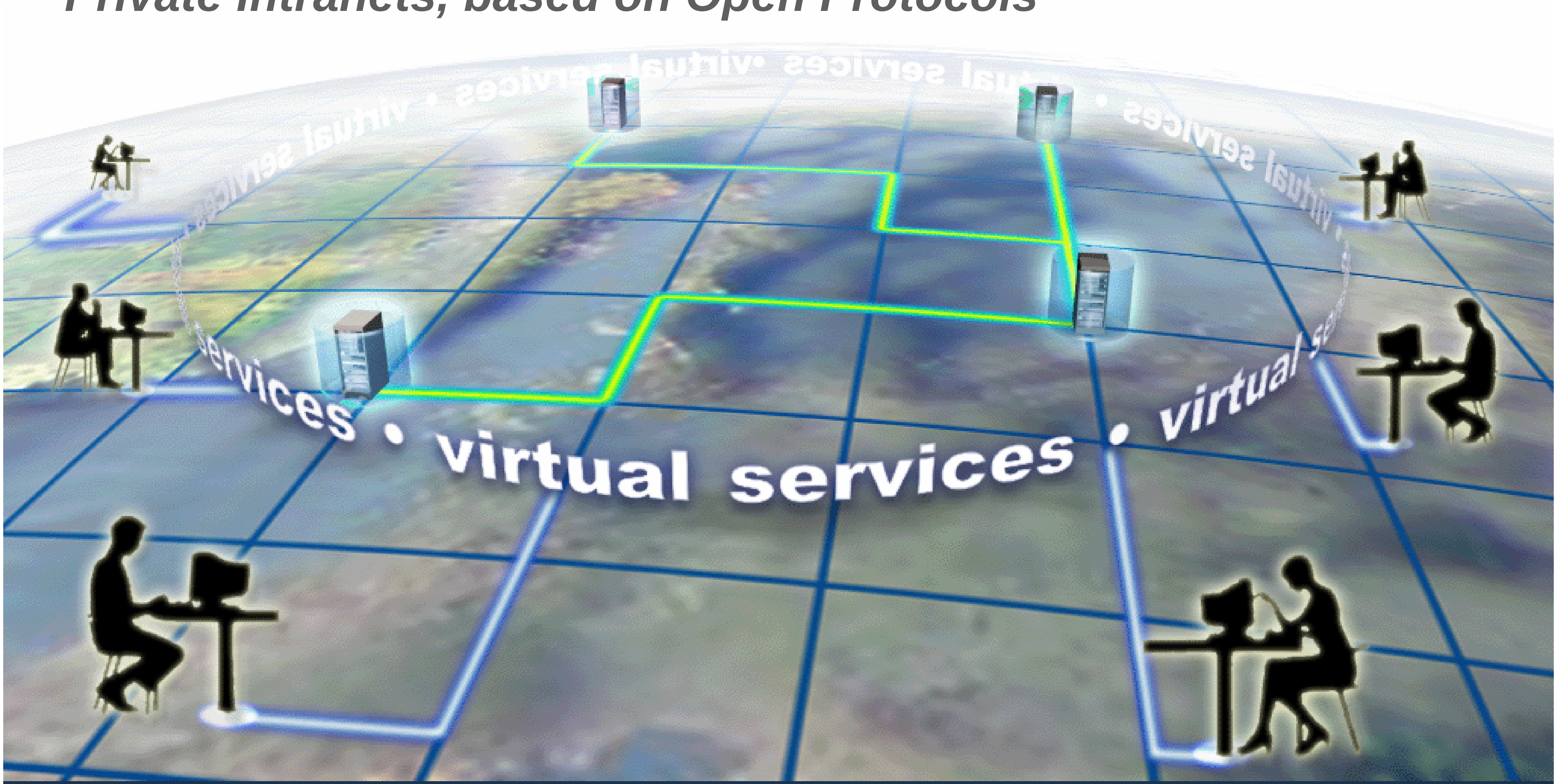


Hiding Complexity

Physical Internet

Hiding Complexity: Grid Computing

Accessing and Sharing Resources over the Internet, or Private Intranets, based on Open Protocols



Summary

- Need to develop a Theory
- Entitlement-approach➔Goal-oriented approach
- The following practical considerations are essential
  - Open Standards
  - Must carry existing infrastructures as much as possible
  - Components and best practices must be reusable

DRIVING FORCES - NEEDED CAPABILITIES

"on demand computing"
discover, detect, monitor, reconfigure
cannot afford to plan and build
react quickly, even if ad hoc