IBM Software Group

IBM

e business software

Service Desk support for Open Grid Services Architecture (OGSA)

Yih-Shin Tan
WebSphere System House Design and Technology,
Application Integration Middleware, IBM Software Group
ystan@us.ibm.com

5/30/2002

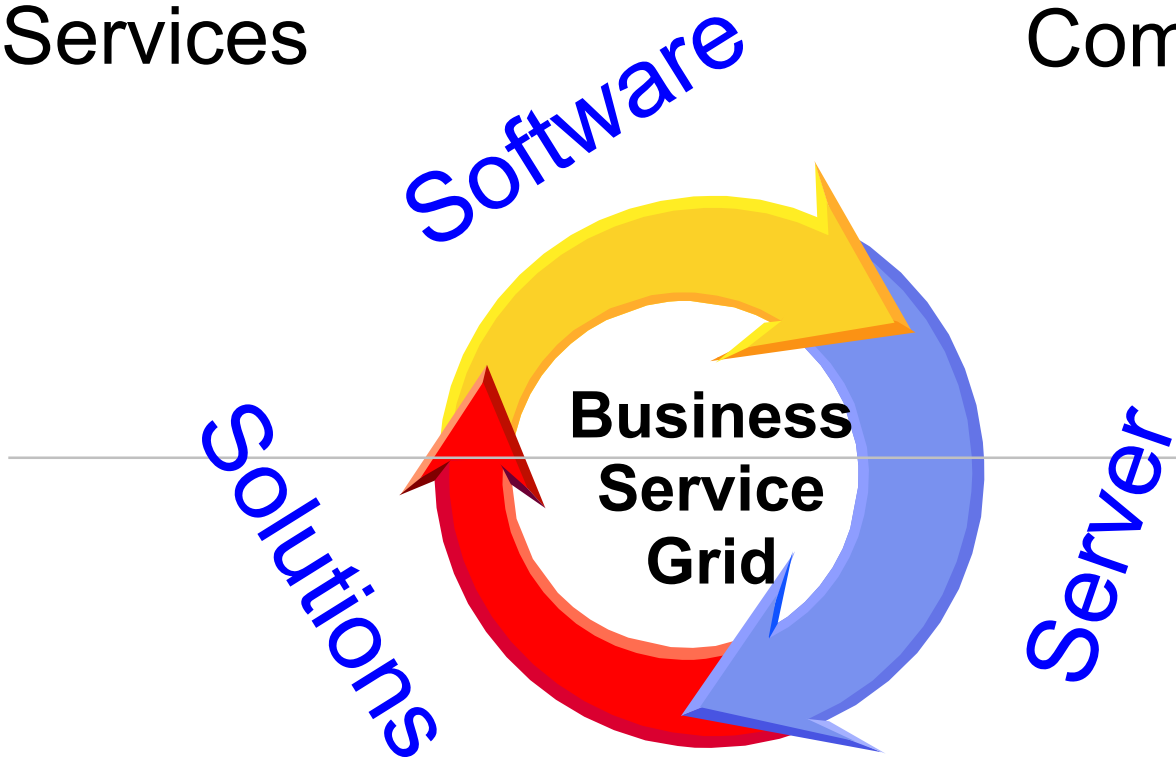
Web Services Integration and Automation

- The Environment
  - Multiple Web Services, many business integration opportunities, more automated electronic processing, new business methods - the characteristics of emerging Dynamic e-business
- Customer Problem
  - Web Services binding, interoperability, sharing, and aggregation are key integration problems
  - Many WS-* standards, APIs, architecture frameworks, and roadmaps
- The Solution
  - Integration of technologies for intelligent clustering built on Web Services
  - Derived from the concept of Autonomic computing
  - Synergized with the emergence of Grid computing

Business Service Grid

Web Services

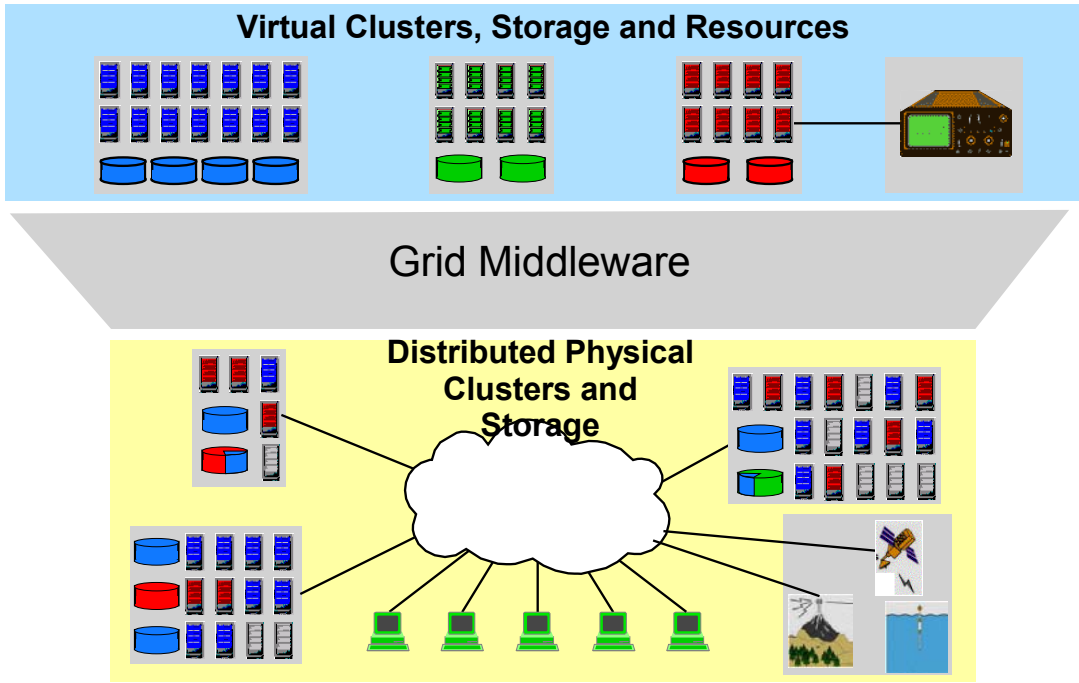
Autonomic Computing



Business Integration

GRID Computing

The Grid



- **Grid computing links servers, clients and storage from across the Internet to form virtual server and storage pools which may be dynamically allocated.**
- **Today's Grid software is focused on physical hardware allocation for high performance computing and collaboration, but will evolve to support an e-utility environment for most workload types.**

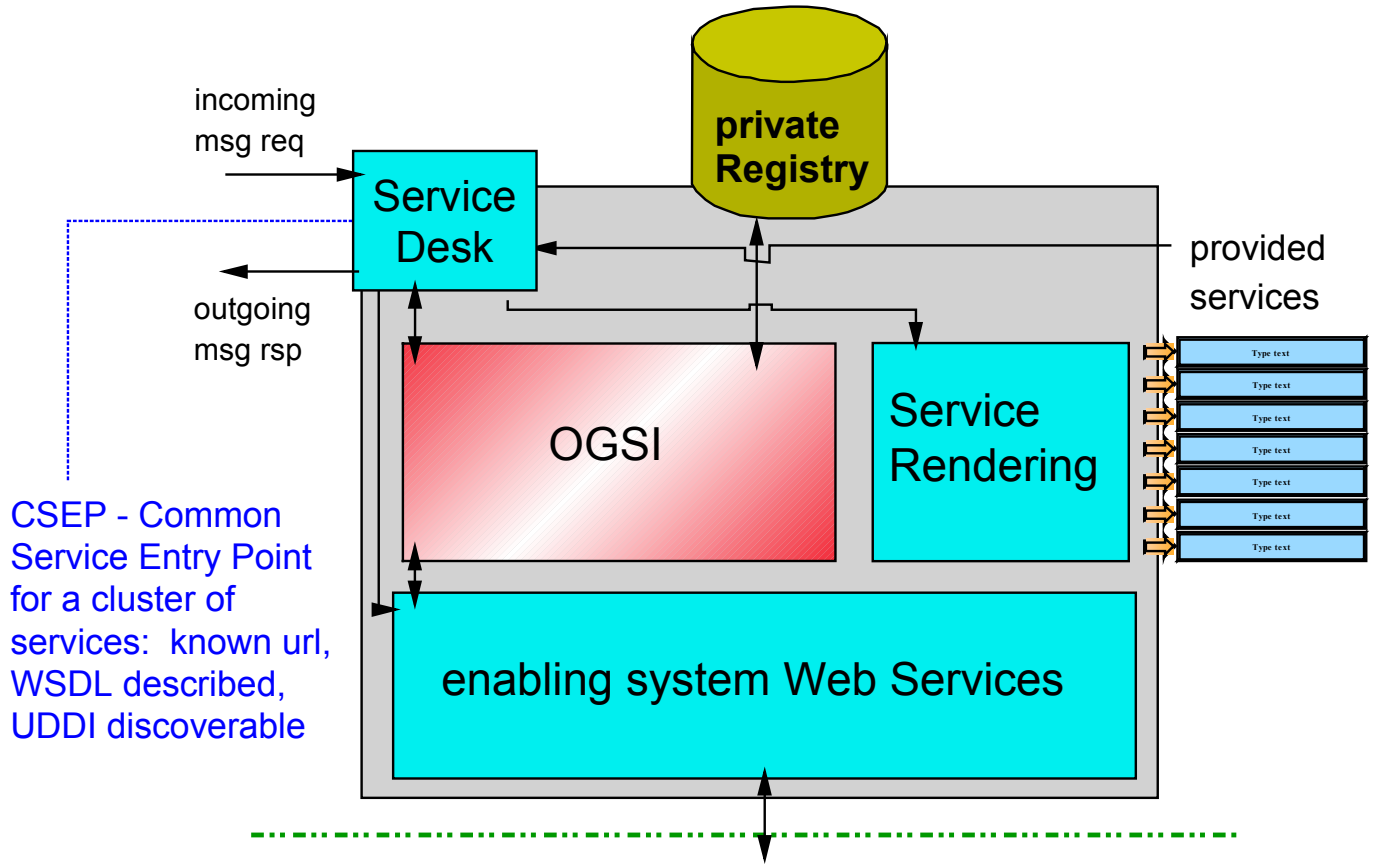
A Vision for the Commercial Grid

Business Services Grid:

Model a distributed application as a collection of composed business components, render them as web services, publish and discover them dynamically, make them accessible through customized, capability based service portals, choreograph them together using BPM and workflow tools, apply a liberal dose of rule-based decision points,

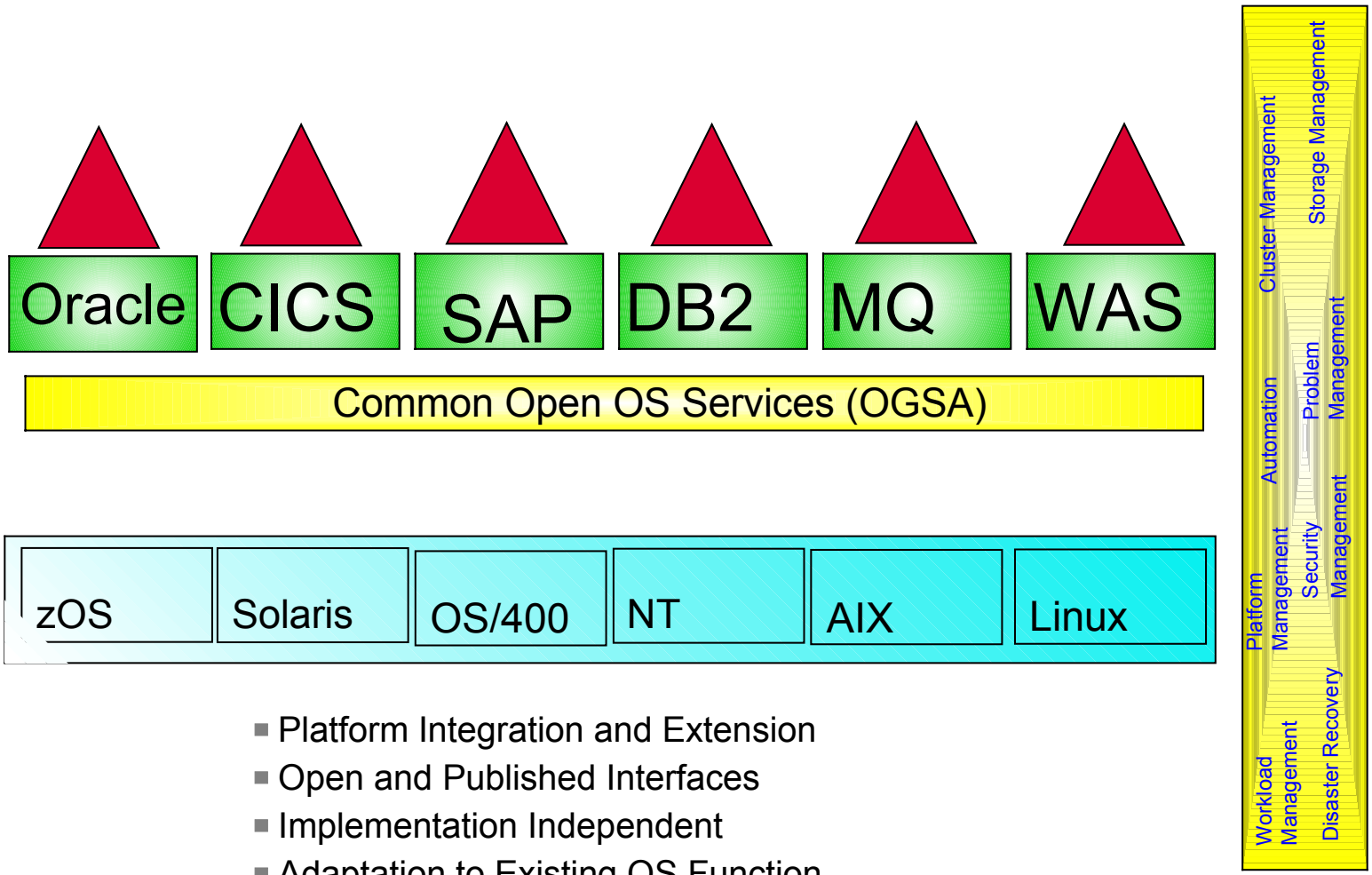
and deploy it all on a robust, well managed, coherent, resilient, heterogeneous distributed system infrastructure that you pay to use.

The Registration and Discovery Architecture from the Business Service Grid/the Service Desk



- **Autonomic Service Routing**
- **Concealed Complexity**
- **Integration of Web Services and Grid**

A Virtualization Layer – Common Services



- Platform Integration and Extension
- Open and Published Interfaces
- Implementation Independent
- Adaptation to Existing OS Function

The OGSA Model

- Interfaces defined with WSDL and Extensions
- Multiple, pluggable Web Services Bindings
- Implementation technology independent
  - But maps very well to J2EE
- Base Function
  - Factories/Instances
  - Naming and Service References
  - Lifecycle Management
  - Registration and Discovery
  - Event Notification
  - Manageability
  - Service Data (XML Based)

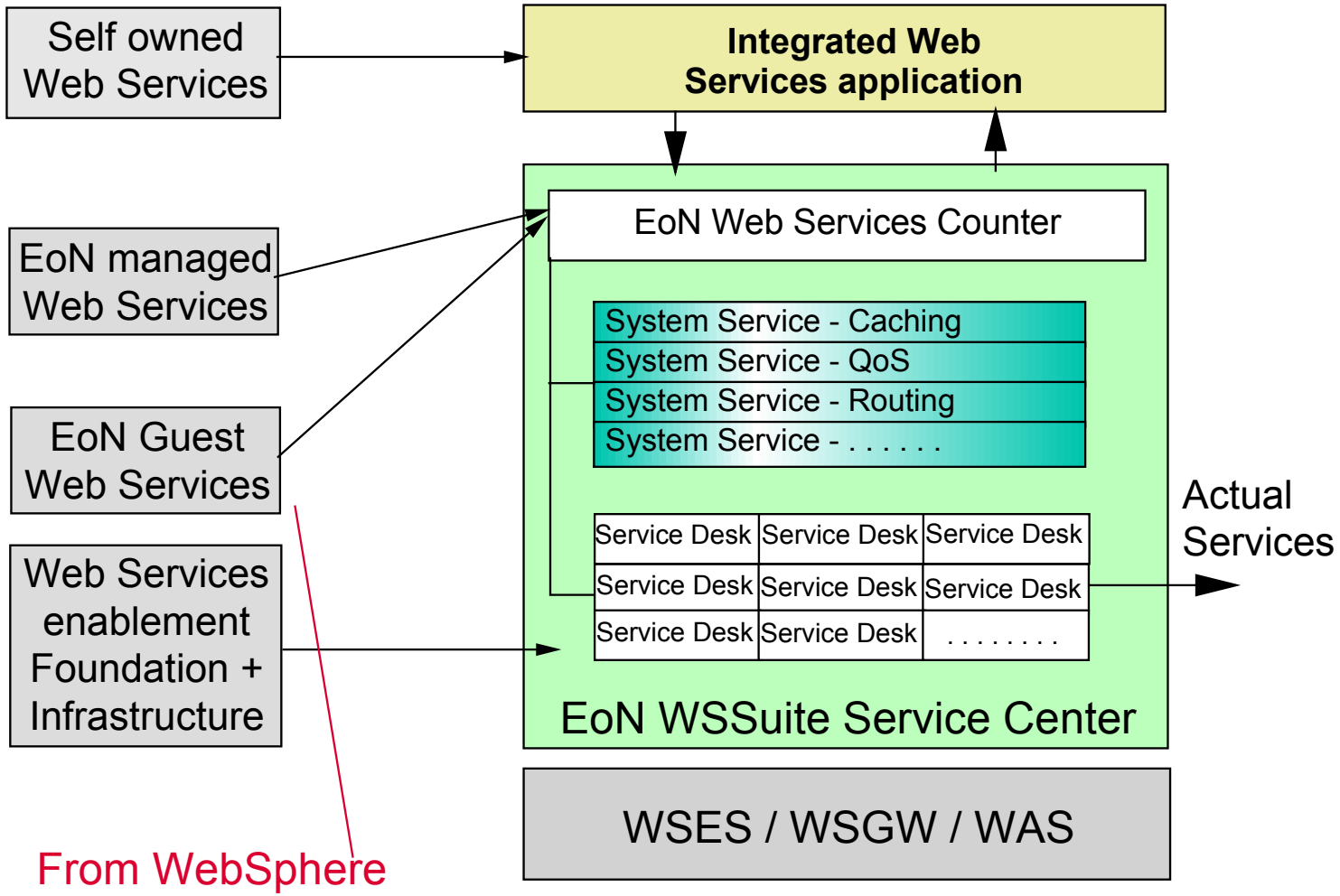
Service Desk as a core component of OGSA

- Core processing engines for entering and operating the Grid
  - Registration and discovery for intelligent service clustering
  - Grid Service Type to enable both Grid and non-Grid Web Services
  - QoS/SLA subsystem abstracted at the Web Services level
  - QoS/SLA mapping to the platform dependent resource managers
- Value add for easy Grid exploitation by applications
  - common enabling system services
  - reliable service aggregation
  - service Interoperation
- Coordination for multiple levels of Grid
  - Business Service Grid (Service Desk, Common Service Entry Point)
  - Service Grid bridge to Computing Grid
    - dynamic registry and discovery of service instances from service grid
    - dynamic pool management of service instances
    - dynamic computing resource allocation/deallocation to service instances from computing grid

Ways to Exploit and Refine OGSI

- **Factory**
  - CreateService
- **Registry**
  - RegisterService
  - UnregisterService
- **PrimaryKey**
  - FindByPrimaryKey
  - DestroyByPrimaryKey)
- **Handle to Reference**
  - HandleMap
- **Notify**
  - NotifySource
  - NotifySink

Conceptual Architecture for a Virtual Business Complex at the Edge of the Network (EoN)



Summary

- Exploit and evolve the Web services opportunity by provide stateful integration, composition and selection capabilities
  - ►WebServices Architecture provides a method to register, discover and inspect services and service characteristics to address application heterogeneity
  - ►Opportunity to exploit availability of multiplicity of related Web services to facilitate state-aware aggregation/composition of Web services
    - –Different service providers, QoS, composite Web services and aggregate QoS
  - ►Service Desk concept
    - –Define, create and manage stateful web services
      - Service attributes include availability, QoS, cost
      - Implement ability to create, notify, register, discover, lifetime management etc.
- Service Desk concept parallel OGSA registration and discovery requirements
  - ►Provides the entry portal service (registry and discovery) for a Grid
  - ►Built and extend on the Web Services Architecture
  - ►Part of the core OGSA components