IBM Systems and Technology Group University 2005

IBM Systems and Technology Group University 2005

Designing and Architecting Grid Solutions

Course #: CB62

Brian Butte & Santosh Gaur
IT Architects

ON DEMAND BUSINESS™

Agenda

- A Short Grid Introduction
- Grid Architecture Overview
- Designing Solutions
- Lessons Learned
- Available Resources
- Appendix A: Basic Grid Component Definitions
- Appendix B: Service Oriented Architecture

A Short Grid Introduction

What is Grid Computing?

- Grid is:
  - Is a *set of technologies* that enables disparate infrastructure to be shared and used as a *single resource:*
    - Cycles: PCs, Servers, Clusters
    - Storage
    - Data: files, DBs, Images
    - Memory
  - Is heterogeneous
  - Is location independent
  - Is a Scale Out vs. Scale Up solution

Grid = Virtual Distributed & Heterogeneous Resource Sharing

When is Grid Applicable?

- **Application and/or Data acceleration**
  - Speed = Opportunity
- **Application resiliency**
  - Resiliency = Responsibility
- **Dispersed data**
  - Information = Power
- **Increase return on hardware assets**
  - Investment = Efficiency

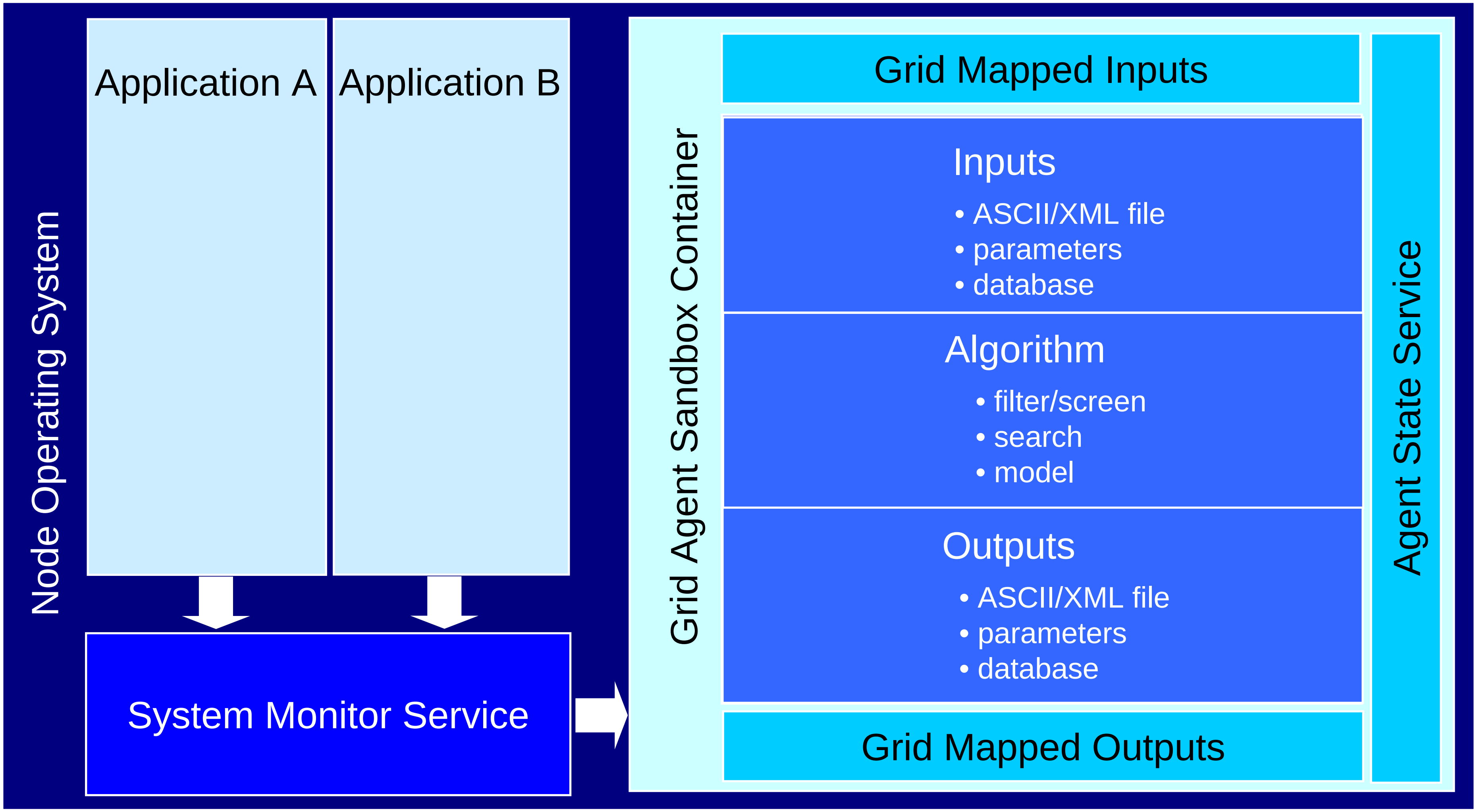
Grid Architectural Patterns

- **Application virtualization**
  - Execute more iterations
  - Execute more robust models
  - Faster execution time
- **Data virtualization**
  - Make data available regardless of location
  - Integrate heterogeneous data sources to provide single view based on need
- **Infrastructure virtualization**
  - Maximize application of resources to needs

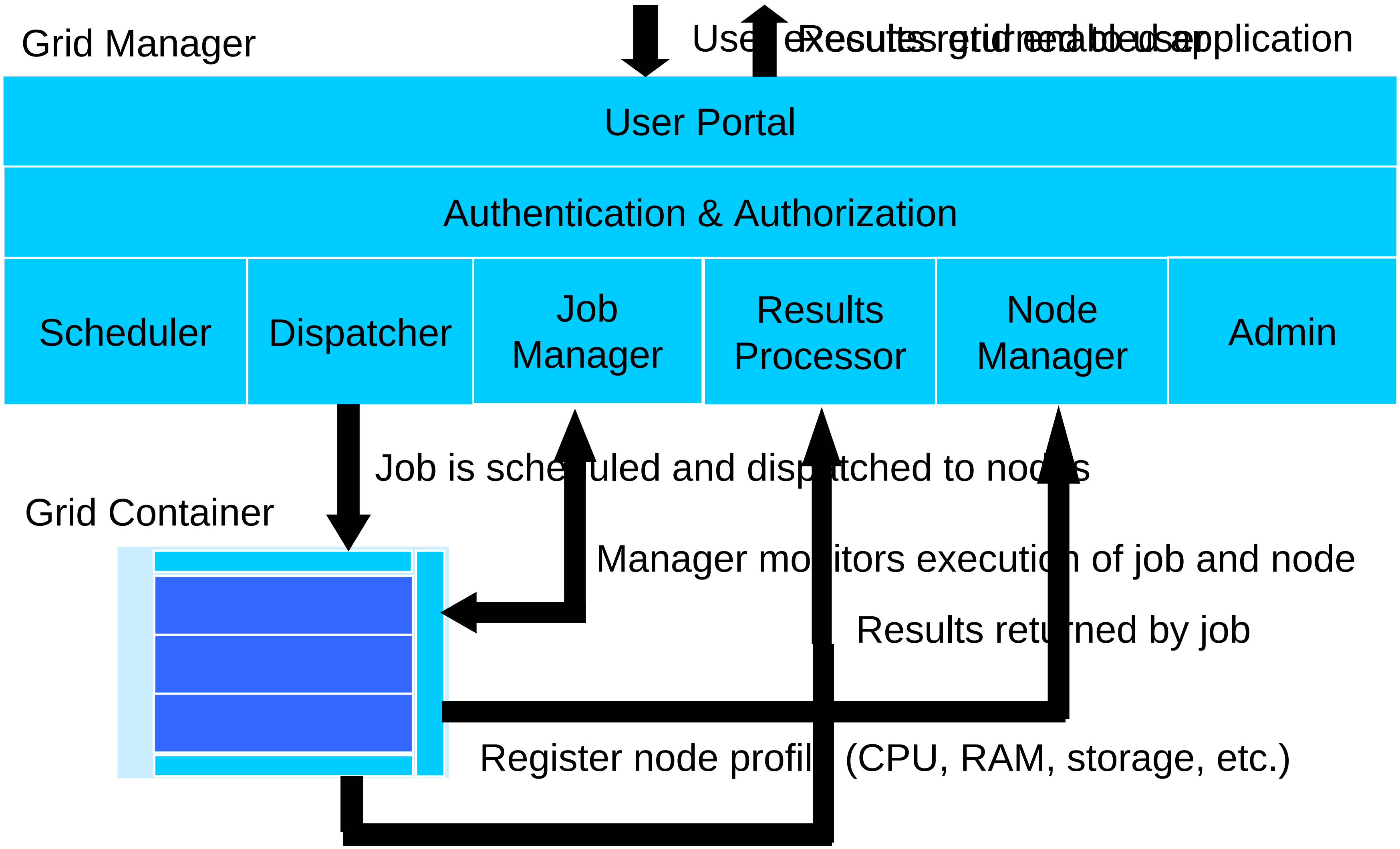
Example: Application Virtualization

- **Job**
  - Atomic unit of executable code (script, binary, jar, etc.)
  - Little or no inter-process communications
  - Includes data code manipulates during execution
  - Data can be a flat file, XML file, or table via database connection
- **Nodes**
  - Hardware (CPU, RAM, Storage) available to grid
  - Servers, clusters, PC’s, laptops
- **Grid Middleware**
  - Manager – schedules jobs, manages failures, returns results on dedicated server
  - Agent – middleware stub run on each Node

Application Job



Grid Middleware



Grid Manager

User enters grid enabled application

Results returned to user

User Portal

Authentication & Authorization

Scheduler

Dispatcher

Job Manager

Results Processor

Node Manager

Admin

Job is scheduled and dispatched to nodes

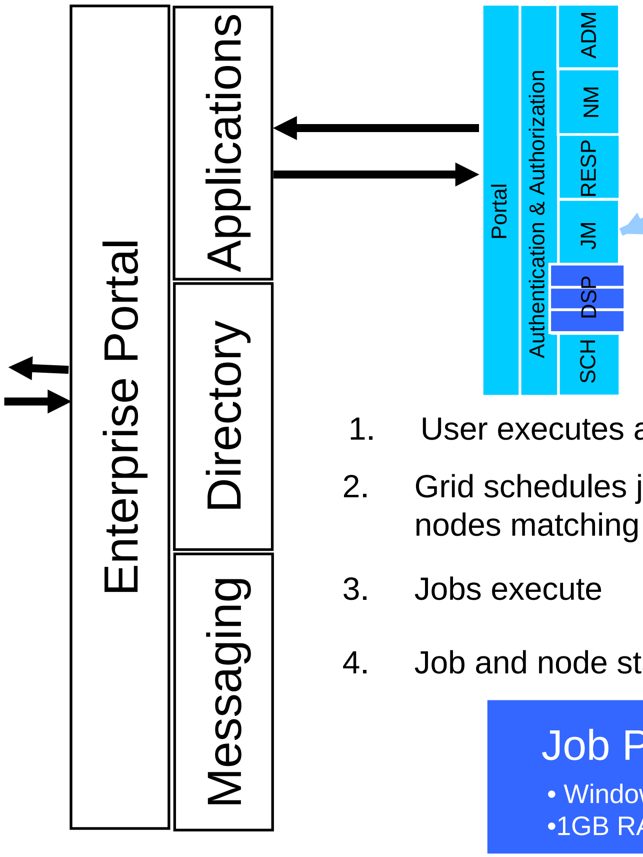
Grid Container

Manager monitors execution of job and node

Results returned by job

Register node profile (CPU, RAM, storage, etc.)

Example: App. Virtualization

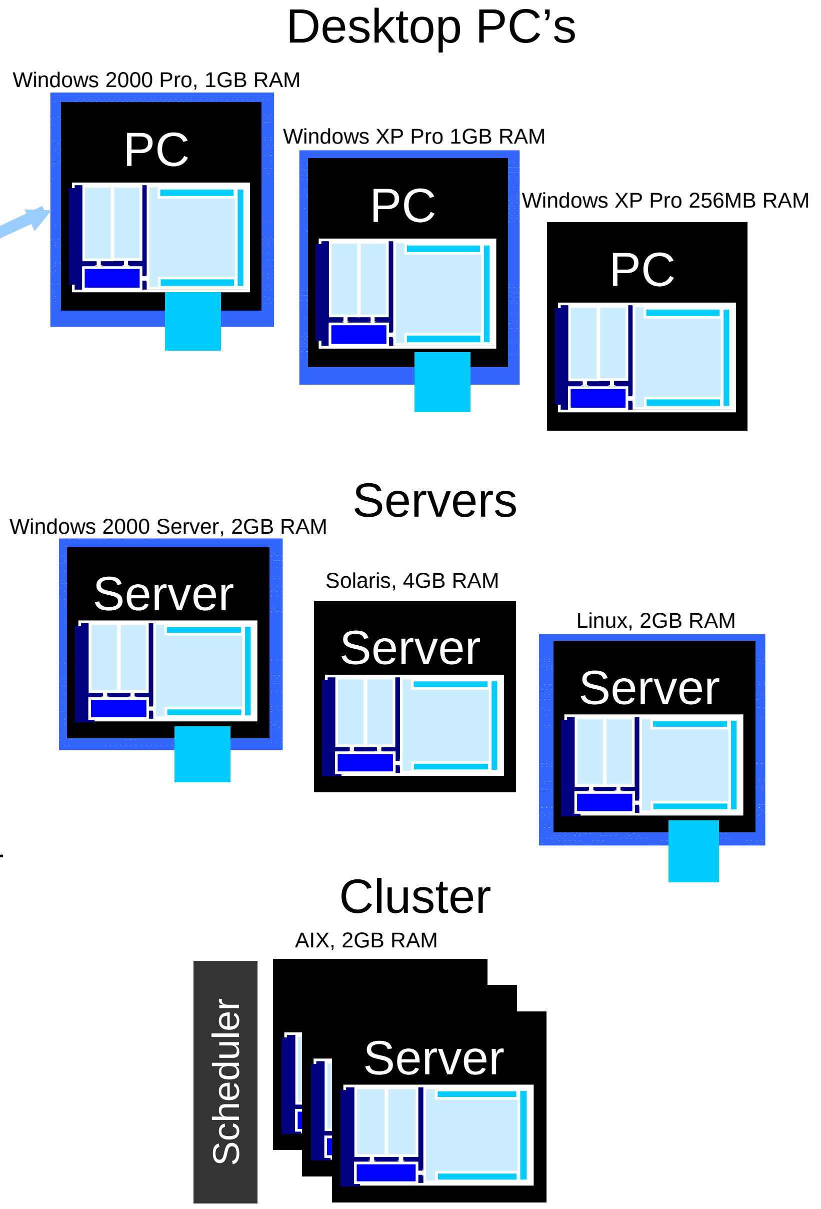


1. User executes application
2. Grid schedules jobs and dispatches to nodes matching job profile
3. Jobs execute
4. Job and node state provided to grid manager

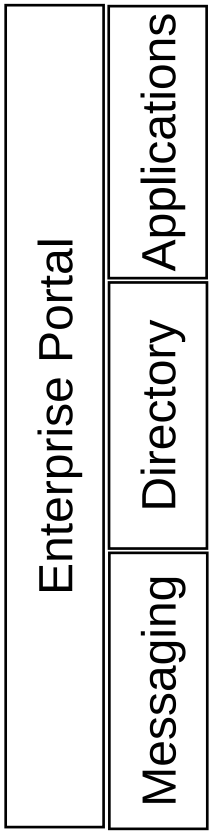
Job Profile

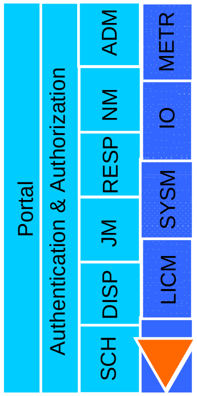
- Windows or Linux
- 1GB RAM

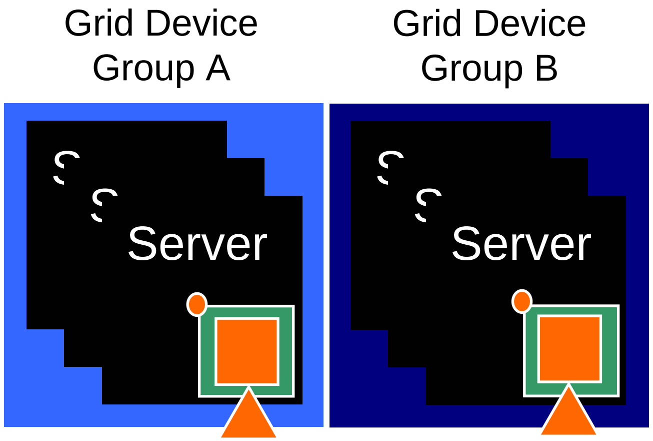
5. Results are returned to Grid Manager for post-processing
6. Results returned to user



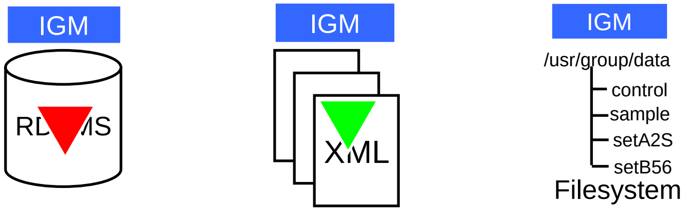
Example: Data Virtualization





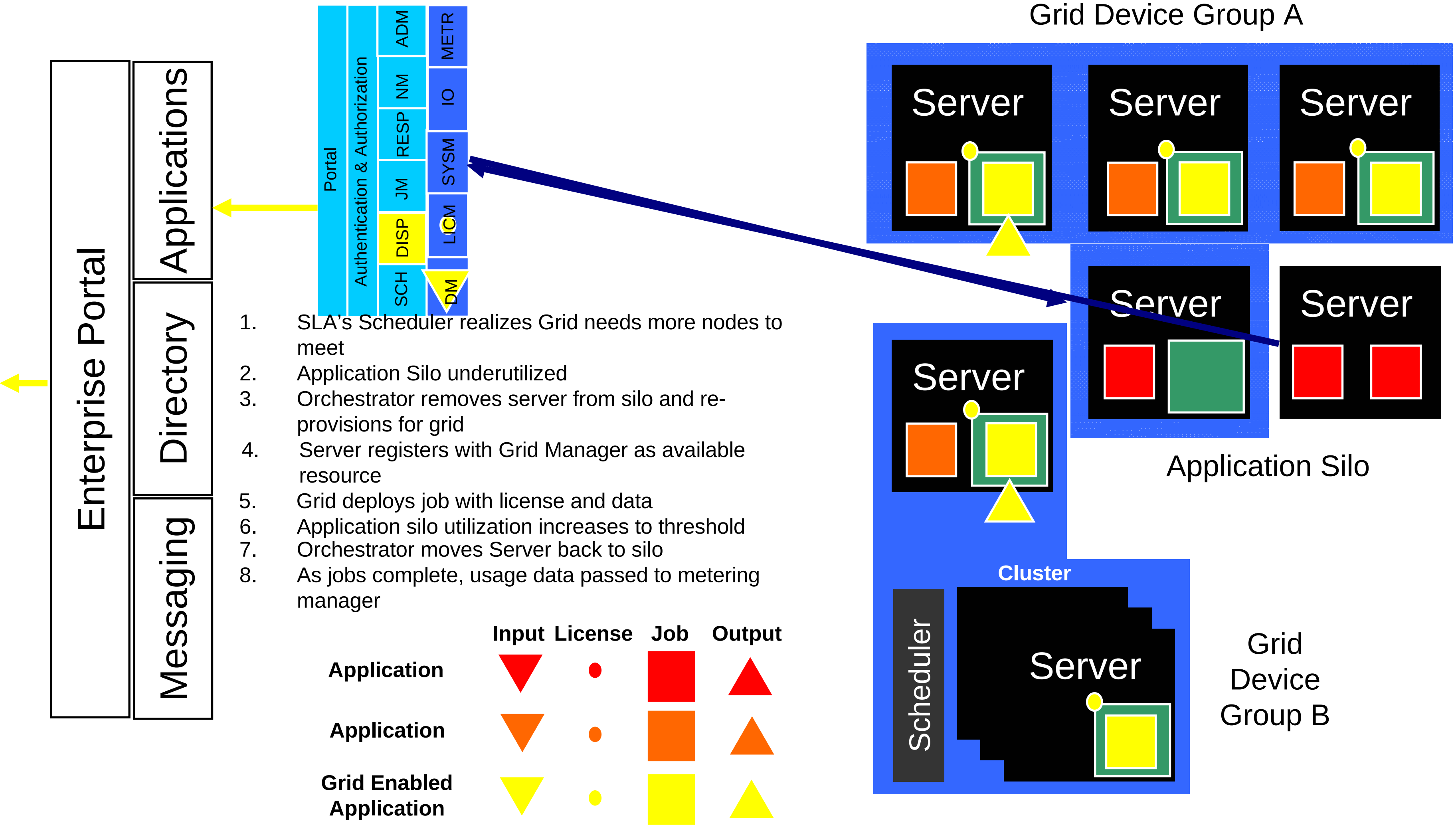


1. Data is collected from sources which can reside in different geographies
2. Data is transformed to meet input needs of jobs.
3. When jobs complete, inputs can be retrieved from cache if still “fresh”
4. Results are sent to their destination.



Grid Enabled Application	Input	License	Job	Output
	▼	●	■	▲

Example: Infrastructure Virtualization



1. SLA's Scheduler realizes Grid needs more nodes to meet
2. Application Silo underutilized
3. Orchestrator removes server from silo and re-provisions for grid
4. Server registers with Grid Manager as available resource
5. Grid deploys job with license and data
6. Application silo utilization increases to threshold
7. Orchestrator moves Server back to silo
8. As jobs complete, usage data passed to metering manager

	Input	License	Job	Output
Application	▼	●	■	▲
Application	▼	●	■	▲
Grid Enabled Application	▼	●	■	▲

Grid Architecture Overview

Grid Architecture Overview

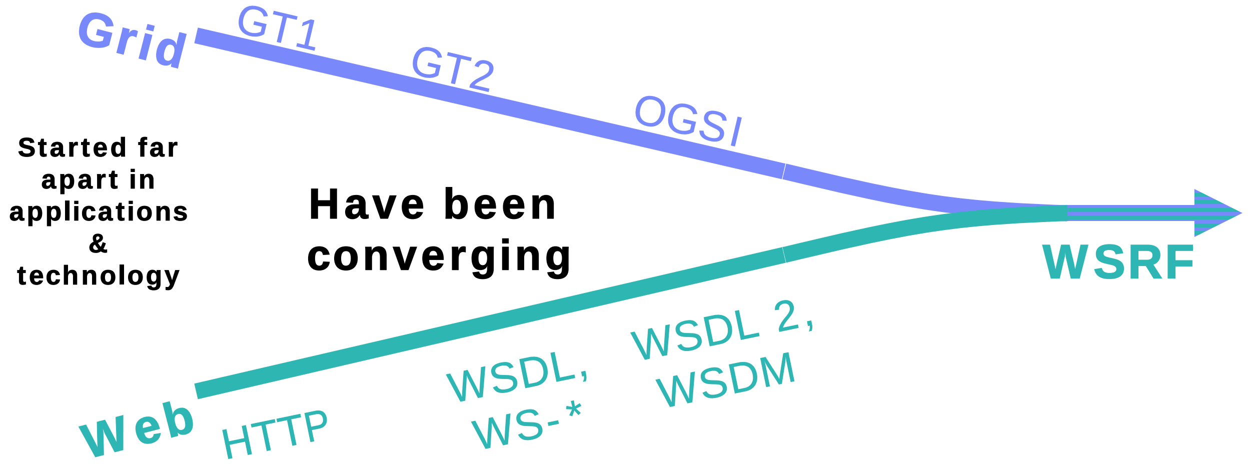
- **Open Grid Services Architecture (OGSA)**
  - Open standard governed by Global Grid Forum
  - Based on a Services Oriented Architecture
  - Defines the services required for a fully functional grid
  - Defines how these services should inter-relate
- **Next evolution of infrastructure**
  - Infrastructure components are resource providers
  - Focus is on dispatching requests to resources or moving resources to requests at the right time regardless of location (JIT)

Services Foundation – Step 1

- **Needed a link between grid services and web services**
  - Statefulness was the key issue
- **Open Grid Services Infrastructure (OGSI) defined**
  - Established by GGF
  - Provides for stateful web services
  - OGSI created as a glue mechanism between OGSA and Web Services
  - Implemented in Globus Toolkit

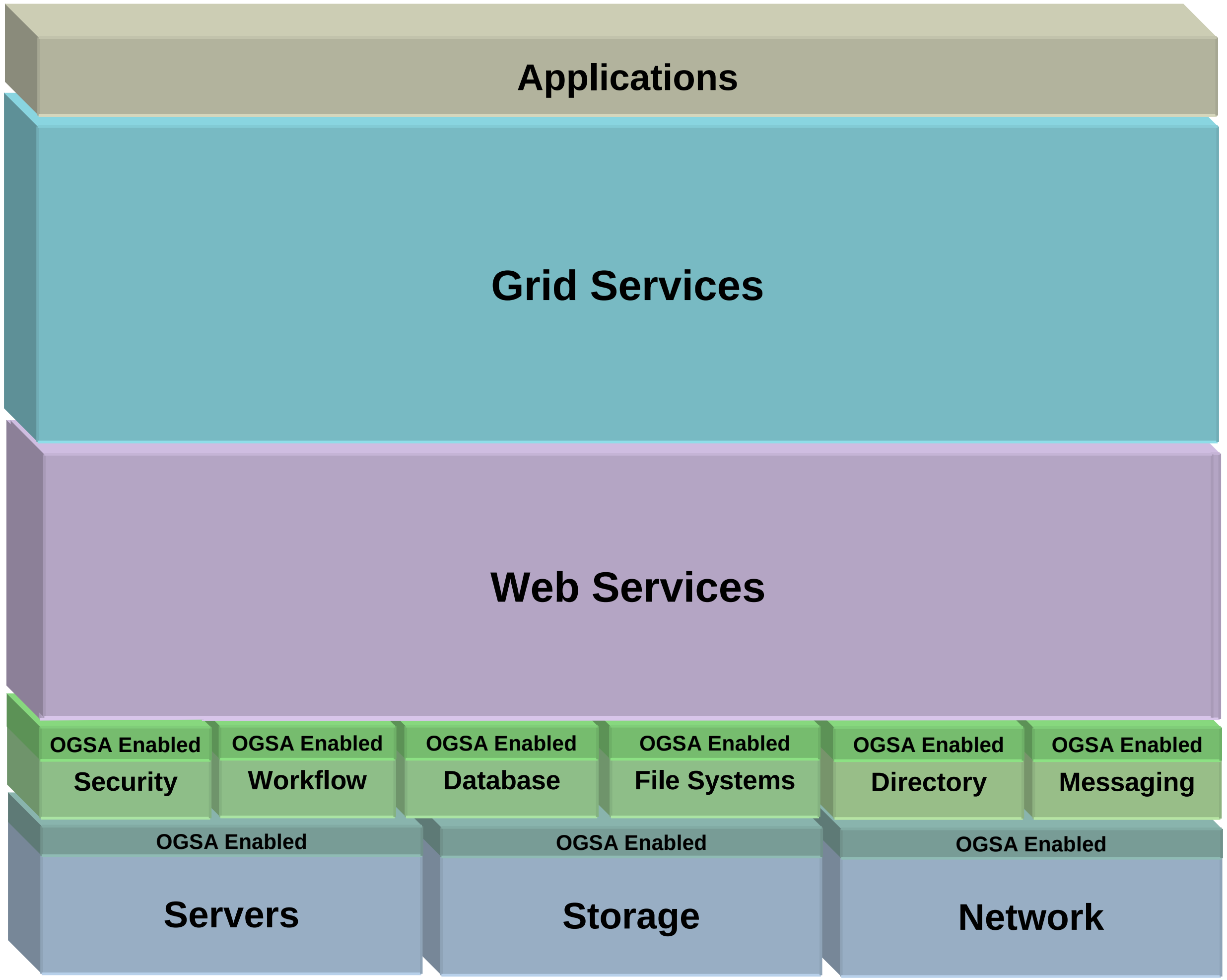
Services Foundation – Step 2

- **W3C recognizes need for stateful web services**
- **WSRF (Web Services Resource Framework) adopted**
  - Web services standard published by W3C
  - Provides stateful web services
  - Eliminates need for OGSI
  - Grid services now squarely reside on web services infrastructure

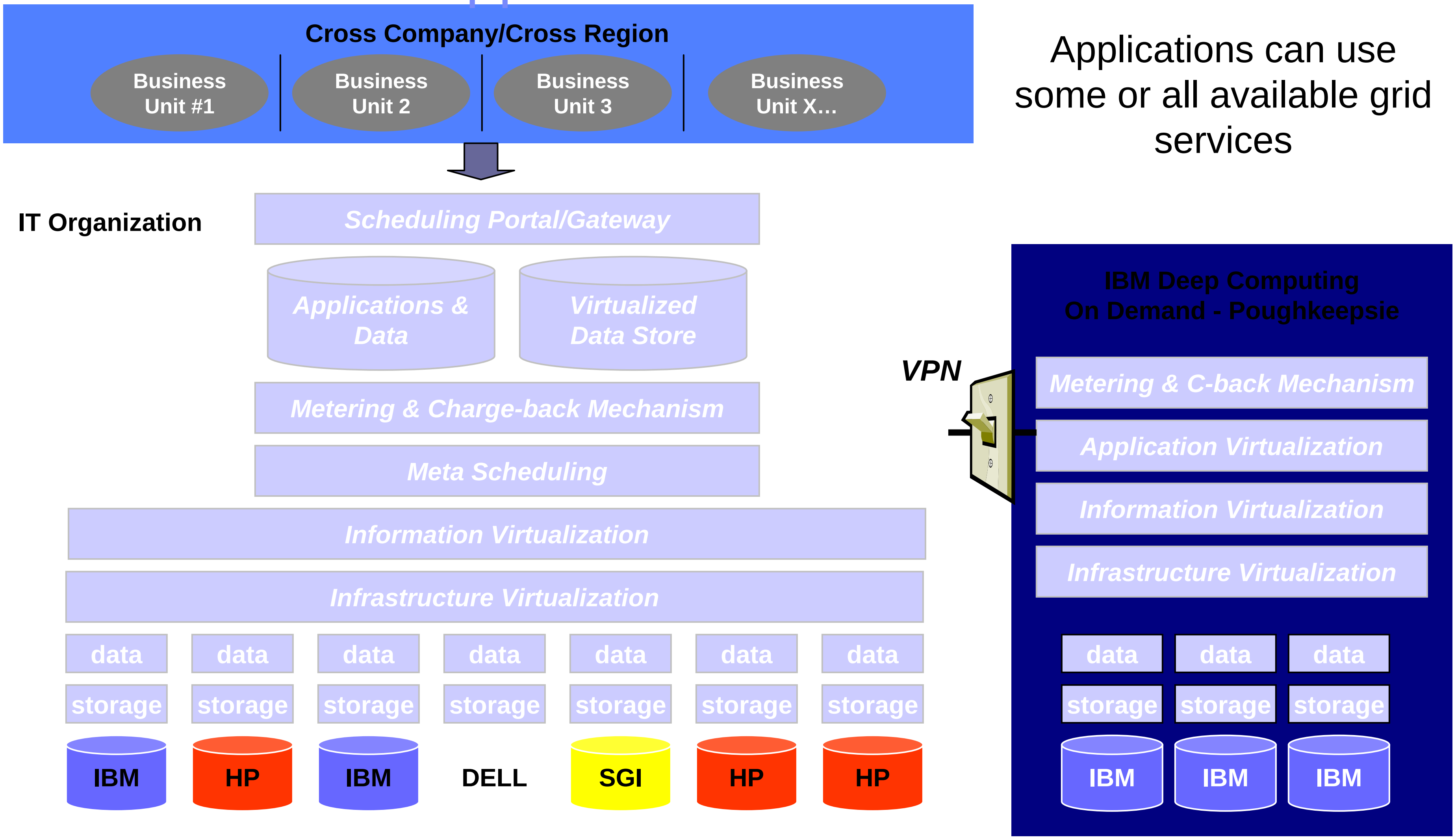


Simplified Grid Architecture

- OGSI concepts incorporated into Web Services
- Grid Services can be developed and implemented as Web services
- Grid Services can take advantage of other Web services standards
- **Grid applications will NOT require special Web services infrastructure**



Grid Reference Application Architecture



Cross Company/Cross Region

Business Unit #1

Business Unit 2

Business Unit 3

Business Unit X…

IT Organization

Scheduling Portal/Gateway

Applications & Data

Virtualized Data Store

Metering & Charge-back Mechanism

Meta Scheduling

Information Virtualization

Infrastructure Virtualization

data data data data data data data

storage storage storage storage storage storage storage

IBM HP IBM DELL SGI HP HP

VPN

IBM Deep Computing On Demand - Poughkeepsie

Metering & C-back Mechanism

Application Virtualization

Information Virtualization

Infrastructure Virtualization

data data data

storage storage storage

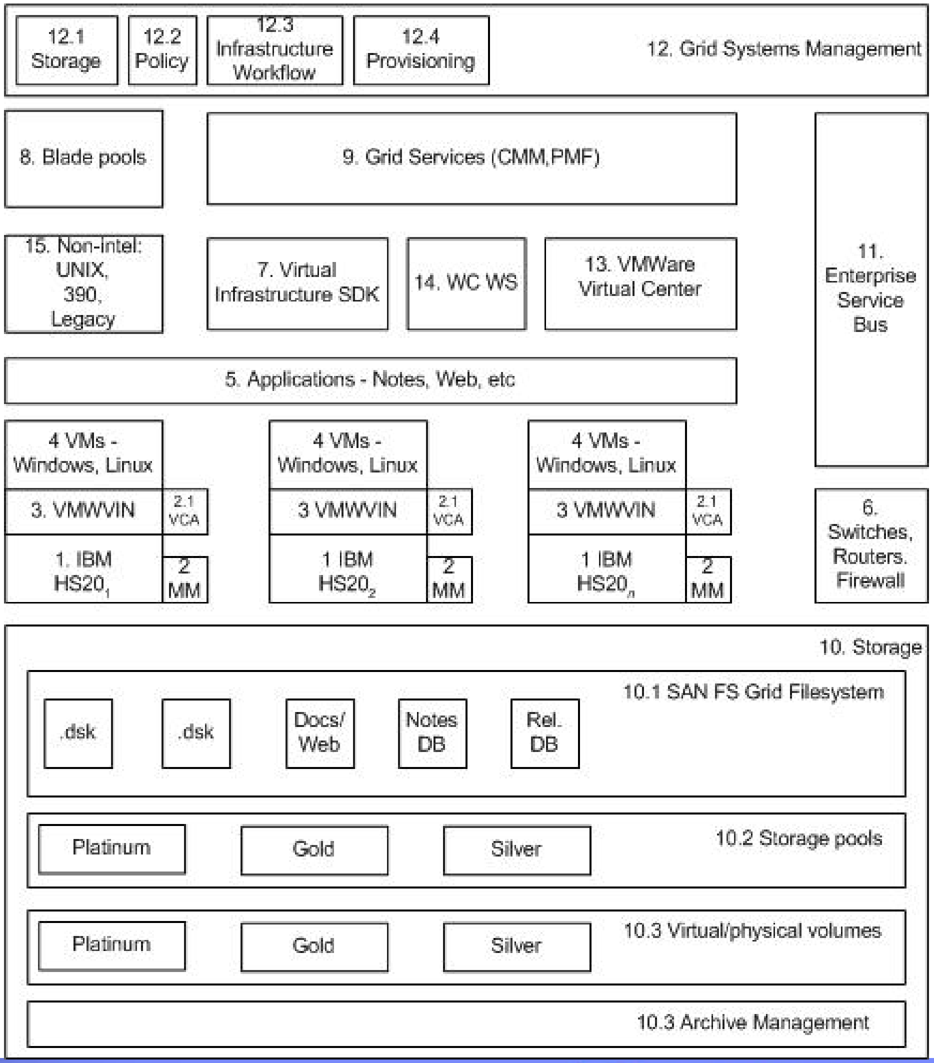
IBM IBM IBM

Applications can use some or all available grid services

Grid Reference Infrastructure Architecture

Conceptual Architecture

- Virtualized distributed infrastructure
- Single site



Designing Solutions

The Solution Process

Problem has been identified, quantified, and defined:

✓**Select the applicable grid business pattern(s)**

- Application virtualization
- Data virtualization
- Infrastructure virtualization

✓**Define architecture**

- **Select a grid enablement technology**
  - Build (Globus)
  - Buy (Grid Middleware ISV's)
- **Design the infrastructure**
- **Design and implement grid enablement**

Grid Enablement Technologies

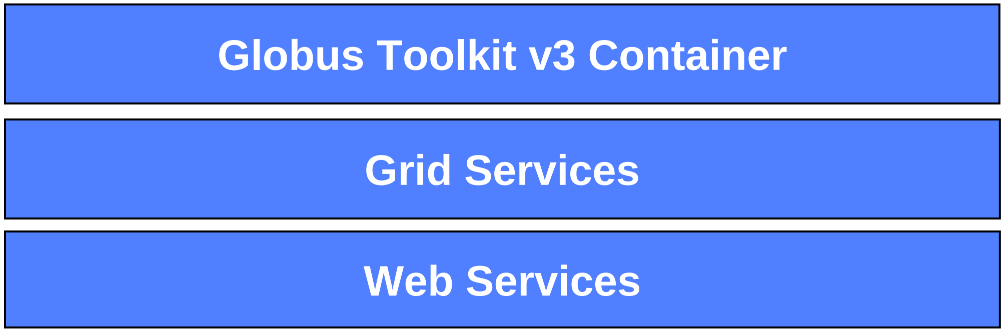
- **Open Source - Globus**
  - IBM Grid Toolbox v3 (www.ibm.com/grid) – application and data virtualization
- **Leading Software Solutions**
  - United Devices – application virtualization
  - Data Synapse – application virtualization
  - Platform – application virtualization
  - Avaki – data virtualization
  - Tacit Networks – data virtualization
  - IBM Virtualization Engine – infrastructure virtualization

Grid Enablement - Build (Globus) vs. Buy (ISV)

- **Not an apple to apple comparison - Globus is a framework**
  - Value of Globus = total control, inter-connectivity
  - Value of ISV = lowest risk point, less time to implement, less administration overhead, support
- **Common trends**
  - Globus – federal government, academia, large research institutions
  - ISV's – private sector

Grid Enablement - The Globus Toolkit

- **Implementation of OGSI v1.0 specification**
- **Focuses on**
  - Resource management
  - Information services
  - Information management
- **Not a functioning grid out of the box**
  - Requires architecture, design, and development time
  - Requires addition of scheduler (Condor, OpenPBS, etc.)
- **Layered model:**



Globus Toolkit v3 Container

Grid Services

Web Services

Grid Enablement - Grid Middleware ISV's

- Leaders have adopted OGSA
- Provide "out of the box" grid solutions
- Have both API and web services interface
- Benefit from customer experience in particular verticals being built into solution
- Lines blurring between vendors, management software is the new frontier for differentiation
- IBM Partners:
  - United Devices
  - Platform Computing
  - Avaki
  - Data Synapse
  - GridXpert
  - Tacit Networks

The Solution Process

Problem has been identified, quantified, and defined:

✓**Select the applicable grid business patterns**

–Application virtualization

–Data virtualization

–Infrastructure virtualization

✓**Customize the reference architecture**

✓**Select a grid enablement technology**

–Build (Globus)

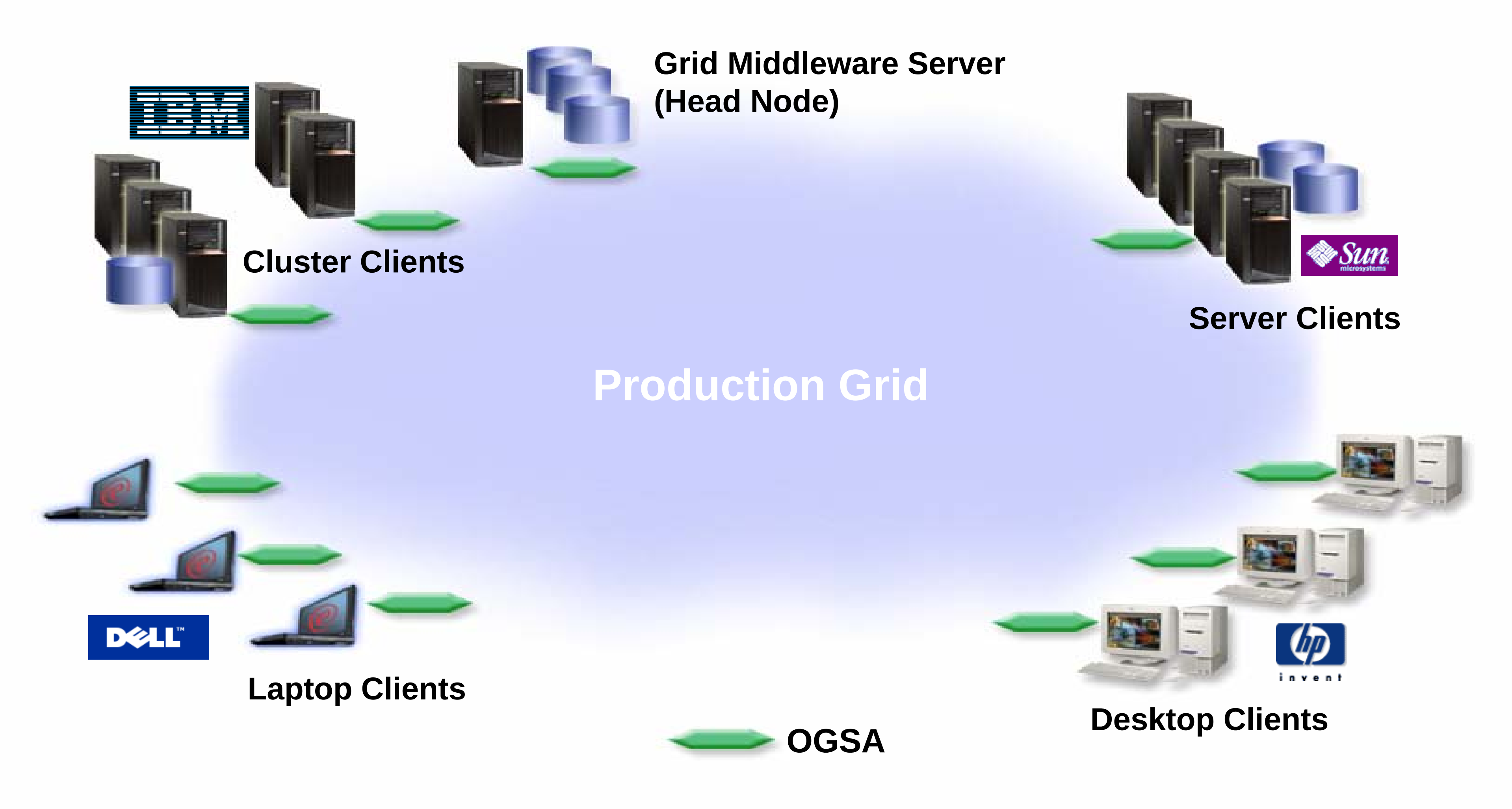
–Buy (Grid Middleware ISV’s)

- **Design the infrastructure**
- **Design and implement grid enablement**

Infrastructure Virtualization Design

- **Common limiting factors**
  - Data file size due to network impact
  - Number of concurrent database connections
  - Geographic location of data
- **Applications need one of the following:**
  - O/S Independence (Java, Python, PERL, etc.)
  - O/S Binary compatibility (RedHat, SuSE, etc.)
  - O/S Specific executables (C, C++, etc.)
  - O/S Homogeneity
- **Can be deployed on heterogeneous hardware**
  - Blade servers
  - Linux / UNIX servers
  - Intel / Windows servers and desktops
  - Mainframe

Grid Infrastructure



The Solution Process

Problem has been identified, quantified, and defined:

✓**Select the applicable grid business patterns**

- Application virtualization
- Data virtualization
- Infrastructure virtualization

✓**Customize the reference architecture**

✓**Select a grid enablement technology**

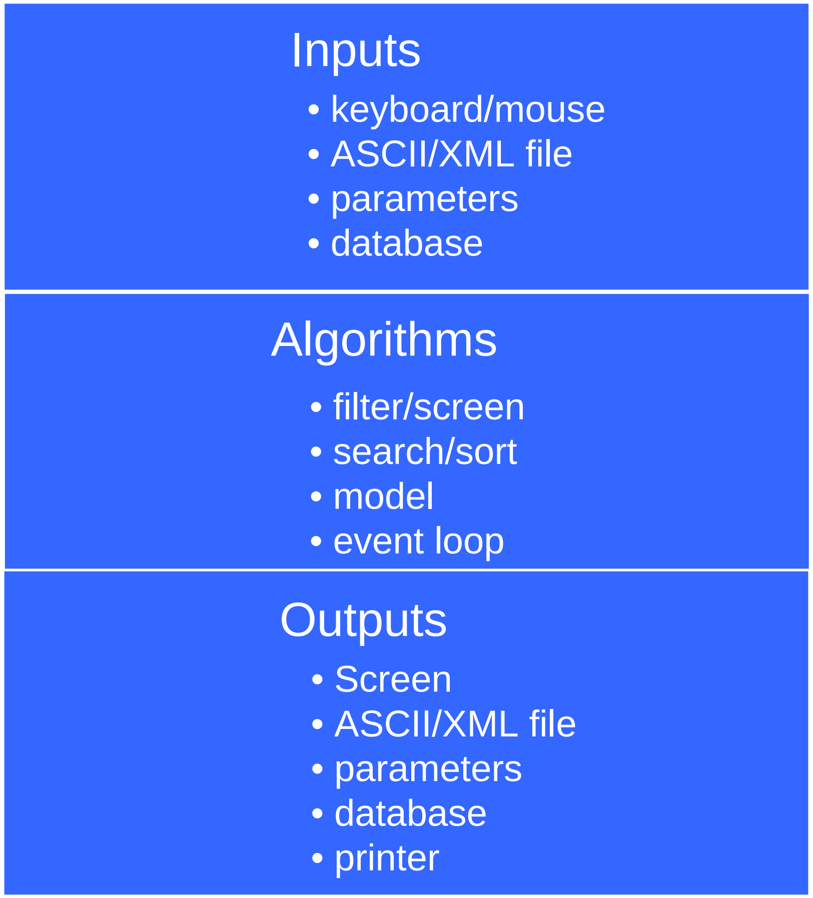
- Build (Globus)
- Buy (Grid Middleware ISV’s)

✓**Design the infrastructure**

- **Design and implement grid enablement**

Grid Enablement Considerations

Generic Application Model



Inputs

- keyboard/mouse
- ASCII/XML file
- parameters
- database

Algorithms

- filter/screen
- search/sort
- model
- event loop

Outputs

- Screen
- ASCII/XML file
- parameters
- database
- printer

Qualifications:

- Does not use keyboard, mouse, or video for inputs/outputs
- Algorithm(s) do not use inter-process communications (IPC)
- Algorithm(s) can be parallelized
  - For loops where the next iteration is independent of the previous iteration
- Data input/output size is reasonable
  - Network bandwidth dependent
  - Large files can often be broken down into smaller files and recompiled later

Strong Candidates

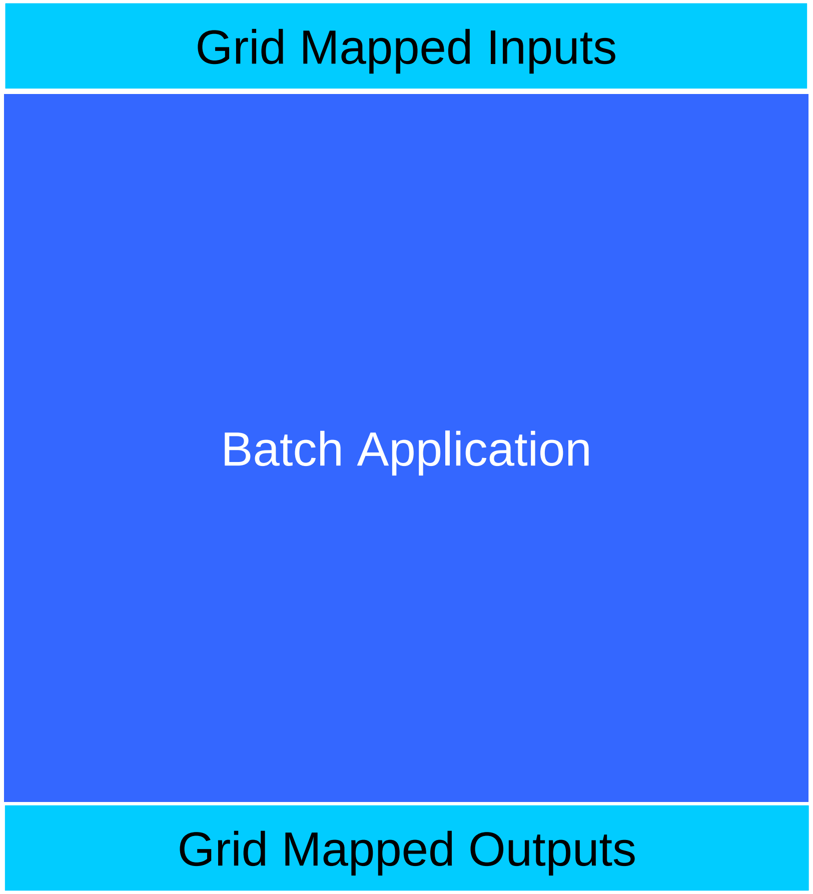
- Classic batch jobs
- Non IPC applications

Weak Candidates

- User interactive applications
- IPC heavy applications

Grid Enablement - Batch Applications

Grid Job Model



Grid Mapped Inputs

Batch Application

Grid Mapped Outputs

Grid Enablement:

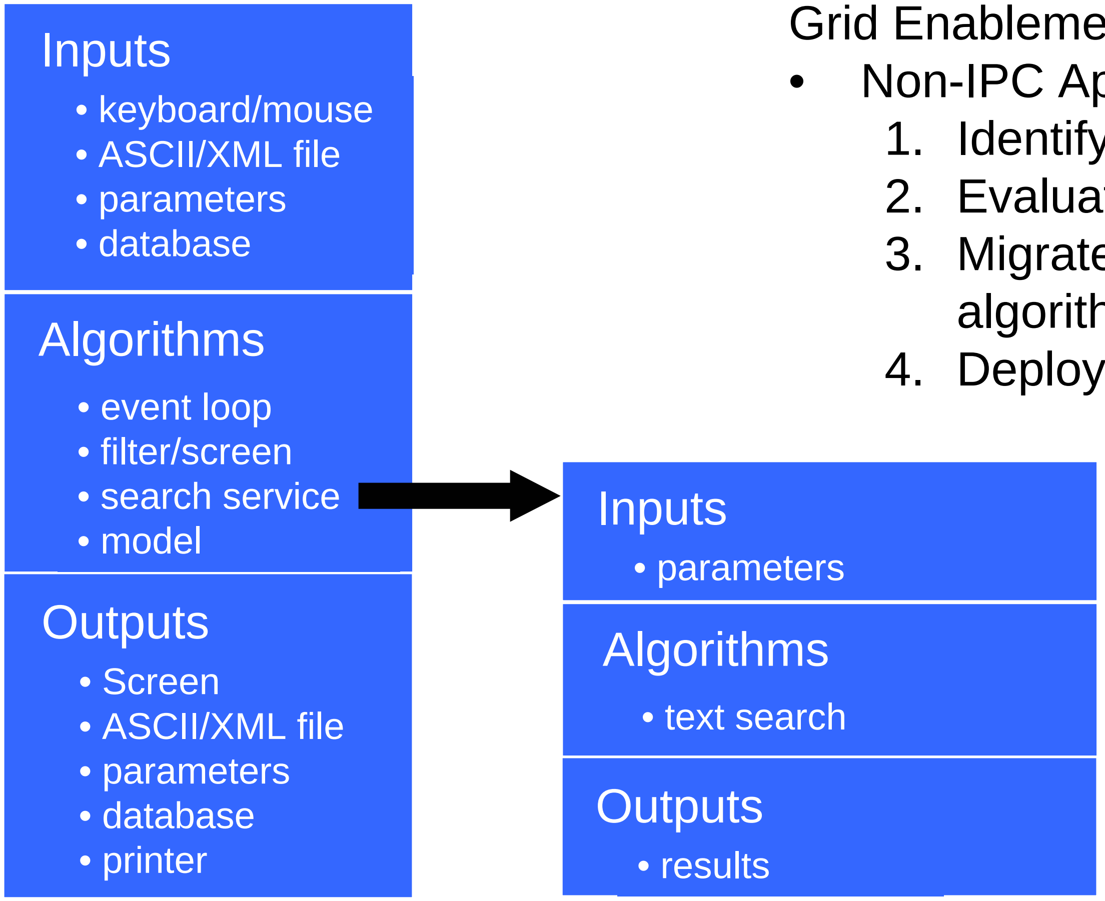
- Batch Application
  1. Map inputs and outputs to grid container (via XML wrappers)
  2. Deploy batch via grid

Grid Enablement - Non-IPC Application

Grid Service Model

Grid Enablement:

- Non-IPC Application
  1. Identify execution time of algorithms
  2. Evaluate algorithms for parallelization
  3. Migrate time intensive parallel algorithms to services
  4. Deploy services on grid



Non-IPC Application

Grid Search Service

Lessons Learned

Keys to Success

- **Have a well defined problem**
  - Cause and effect understood
  - Business impact and value of resolution
- **The problem is application, data, or infrastructure driven**
  - Long process times
  - Inability to share data
  - Low return on assets
- **The problem is addressable with grid technology**
  - Access to source code or grid enabled version
  - Data is sharable
  - Mainstream hardware components
- **Start small and grow (crawl, walk, run)**
- **Build toward a Service Oriented Architecture**

Aventis Pharmaceuticals

Challenge

- Distributed, diverse data sources across continents
- Data was Heterogeneous, Cross Platform, consisting of Files or Databases
- Limited ability to consolidate, construct and analyze data sets

Solution

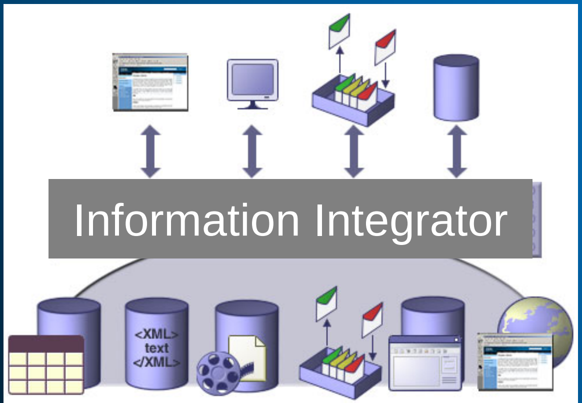
- Linux
- IBM eserver
- IBM Information Integrator

Key Business Benefits

1. Using IBM Information Integrator to bring together disparate LS data sources in one coherent view
2. Significant increase in researcher productivity due to improved collaboration & data access
3. Better data quality and currency

Information/Collaboration Grid

Aventis



Integrating diverse Life Sciences information across and beyond the enterprise

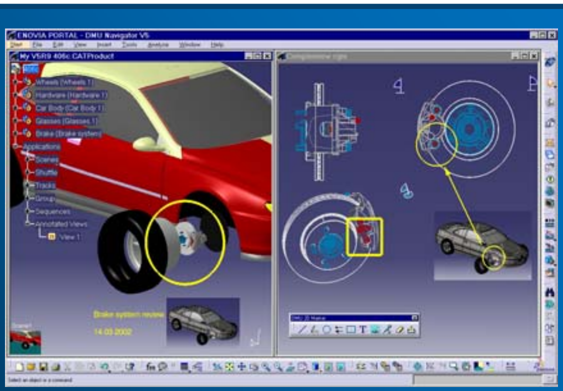
Magna Steyr

Challenge: Accelerate the execution of automotive simulation tests

- Business Issues
  - Too much time required to effectively run clash tests between complex sub-assemblies
  - Too much administrative time required from design engineers
- Business Benefits
  - Improved quality of the end product
  - Reduced time to market
  - Faster evaluation of design alternatives
- Chosen Grid Infrastructure
  - Platform Computing as Grid Middleware provider w/ Dassault Systemes CATIA & ENOVIA DMU applications

-- "Grid technology from IBM and Platform Computing reduced the time required for our clash testing from 72 – 4 hours and contributed significantly to enhancing our design quality," said Dr. Heinz Mayer, MAGNA STEYR.

Engineering and Design



Technology Benefits:
• Significant performance improvement (72 – 4 hrs)

Grid Computing Business Benefits:
• Reduces risk and errors in development driving cost savings and improved time to market

IBM

Challenge: Increase the speed of innovation in systems development

- Business Issues
  - Maximize system resource utilization in Microprocessor Design, Benchmarking & Test and Server Design
- Business Benefits
  - Microprocessor Design Grid
    - Reduced development cycle
    - improved ROI and design engineer productivity
  - Benchmarking/Testing Grid
    - Larger scaling tests at lower costs
  - Z Series Design Grid
    - Increased computing power for HW simulations
    - 40% increase in productivity of hardware engineers
- Chosen Grid Infrastructure
  - Globus Toolkit developed solution hosted on IBM eServer platform build by IBM Global Services

Engineering & Design Grid

Technology Benefits:

• Reduced cycle time by increasing resource utilization

Grid Computing Business Benefits:

• Improved time to market
• Increased productivity
• Ability to execute more tests in same time

10 Reasons Why Clients Should Choose IBM Grid Computing

1) IBM has implemented Grid computing for 100+ customer organizations worldwide

2) Clients can start small and grow with IBM's 21 Industry-focused Grid offerings

3) Analysts and Media continually cite IBM's Grid computing leadership

4) Relationships with many leading application ISVs, like SAS, Dassault, Cadence, Accelrys, etc - Grid-enabling their applications via IBM technologies and workshops

5) Expertise in scaling key grid technologies like: WebSphere, DB2,Tivoli, Platform, DataSynapse, Avaki, Globus, etc. -

6) IBM's deep commitment to open standards (e.g. OGSA, LINUX, Web Services)

7) IBM Business Consultants can help clients leverage grid technology to create business value

8) IBM has trained grid computing technology experts in every major global market

9) IBM world-class and worldwide support - Grid Briefing Centers, Grid Design Centers, Grid Integration Center, ISV Innovation Centers, etc.

10) IBM is enabling many Business Partners to help take Grid to local markets

Available Resources

Where can I learn more about Grid?

1. Grid Web site   www.ibm.com/grid

- Articles, press releases, analyst reports
- Web casts, brochures, solutions briefs
- Customer success stories

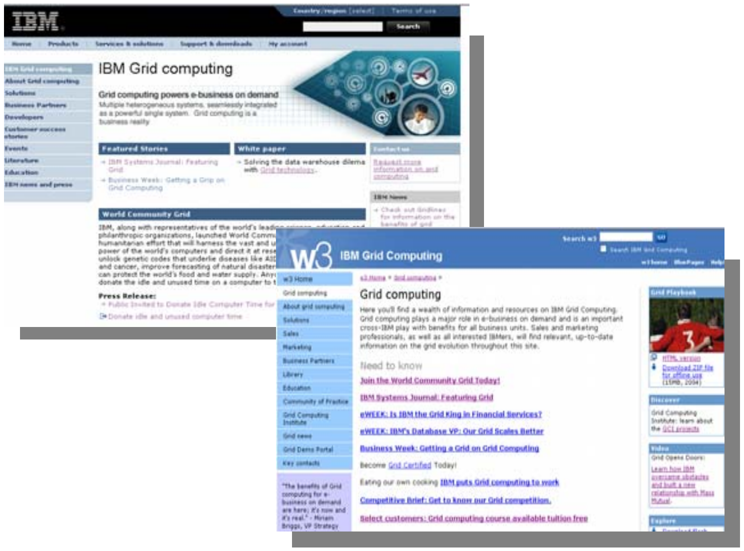
2. Grid w3 site   w3.grid.ibm.com

- SGB Seller Playbook
- Education / Classes, FAQ, Pocket Guide
- Demos, videos, key grid contacts

3. *NEW* SGB Seller Playbook

w3.grid.ibm.com

- Includes Grid, VS, Deep and Linux
- New and improved industry messages and customer materials
- Seller resource guides, mapped to SSM steps
- Qualification questions and next steps to take with your client



IBM Grid computing

Grid computing powers e-business on demand

Multiple heterogeneous systems, seamlessly integrated as a powerful single system. Grid computing is a business reality

Featured Stories

- IBM Systems Journal: Featuring Grid
- Business Week: Getting a Grip on Grid Computing

White paper

- Solving the data warehouse dilema with Grid technology.

World Community Grid

IBM, along with representatives of the world's leading science, education and philanthropic organizations, launched World Comm… humanitarian effort that will harness the vast and u… power of the world's computers and direct it at rese… unlock genetic codes that underlie diseases like AI… and cancer, improve forecasting of natural disaster… can protect the world's food and water supply. Any… donate the idle and unused time on a computer to t…

Press Release:

- Public Invited to Donate Idle Computer Time for…
- Donate idle and unused computer time

W3 IBM Grid Computing

Grid computing

Here you'll find a wealth of information and resources on IBM Grid Computing. Grid computing plays a major role in e-business on demand and is an important cross-IBM play with benefits for all business units. Sales and marketing professionals, as well as all interested IBMers, will find relevant, up-to-date information on the grid evolution throughout this site.

Need to know

Join the World Community Grid Today!

IBM Systems Journal: Featuring Grid

eWEEK: Is IBM the Grid King in Financial Services?

eWEEK: IBM's Database VP: Our Grid Scales Better

Business Week: Getting a Grid on Grid Computing

Become Grid Certified Today!

Eating our own cooking IBM puts Grid computing to work

Competitive Brief: Get to know our Grid competition.

Select customers: Grid computing course available tuition free

"The benefits of Grid computing for e-business on demand are here; it's now and it's real." - Miriam Briggs, VP Strategy

Grid Playbook

Discover: Grid Computing Institute: learn about the GCI projects

Video: Grid Opens Doors: Learn how IBM overcame obstacles and built a new relationship with Mass Mutual.

SGB Seller Playbook

VERSION 1

IBM Grid Computing: Government

Selling Scenarios

IBM Grid Computing: Government

Seller Resource Guide

Where can I learn more about Grid?

4. **“Grid Computing” – IBM Press**
   - Author: Joshy Joseph & Craig Fellenstein
   - ISBN: 0-13-145660-1

Supporting Resources

1. IBM Strategic Growth Initiatives – Grid Architects

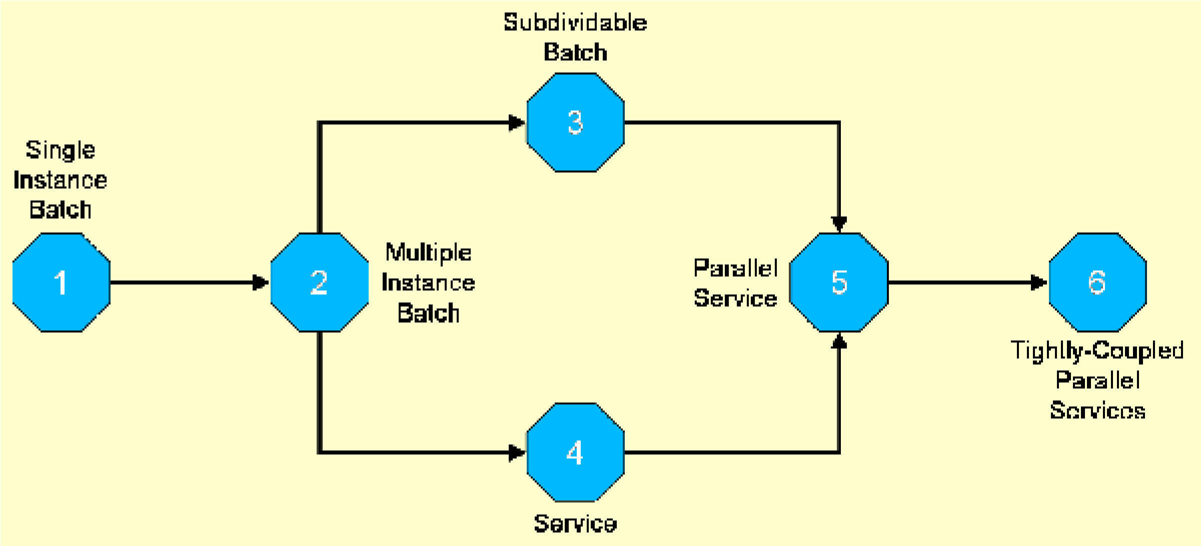
- Financial Services – Bob Gross
- Communication Sector – Santosh Gaur
- Industrial – Greg Kettman, Fadel Fiani
- Higher Education – Martin Maldonado
- Government – Michael Osias
- Life Sciences & Healthcare, Distribution – Brian Butte

Thank-you

Appendix A: Basic Grid Component Definitions

Application Types

1. **Batch Anywhere - Single instance of a job running on any of several systems (nodes)**
2. **Independent Batch - Multiple independent instances of a job**
3. **Parallel Batch - Spreading a job's work among nodes**
4. **Service – A callable service**
5. **Parallel Service – Multiple concurrently callable services**
6. **Tightly Coupled Parallel Program**



Job

- **A job is a software executable (binary, script, etc.)**
- **Job is an application or application component**
- **Job is executed in a grid container installed on each node**
- **Execution of a job is triggered by a user request**
- **The software and data are downloaded to nodes as needed (may already exist)**
- **Job executes on node until completion, failure or infinite loop occurs:**
  - Completion: resulting data set returned to grid manager which may aggregate date for return to user
  - Failure: job can be rescheduled to an alternate node.
  - Infinite Loop: job can exceed limits resulting in treatment as a failure
- **Results are returned to grid manager upon completion**
- **Multiple jobs can run simultaneously on a node**

Nodes

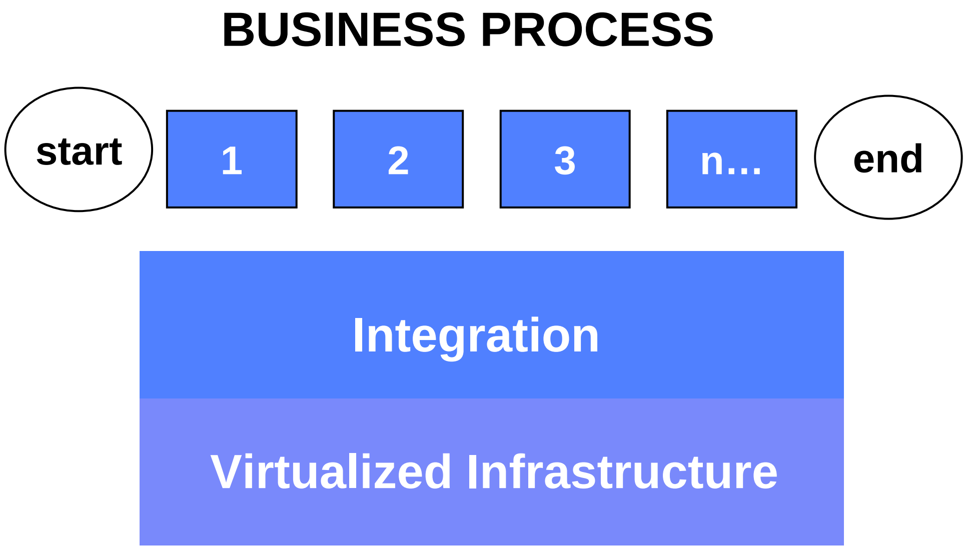
- **A node is a piece of computing hardware such as:**
  - Server
  - Cluster
  - Desktop PC's & laptops
- **A profile is created for each node indicating:**
  - CPU speed and type
  - Operating system
  - RAM and storage
  - Hours of availability
  - Location
  - etc.
- **A node self-registers with grid manager to become available**

Grid Middleware

- **Grid Middleware Client : Grid Container**
  - A grid software agent (container) is installed on each node
  - Communicates with grid middleware server (manager)
  - The grid container segments the job from other applications:
    - Protects other applications from grid crashes
    - Protects grid jobs from tampering
    - Prevents grid job / application interaction
  - Applications run co-resident with the grid container
- **Grid Middleware Server : Grid Manager**
  - Assembles and manages nodes individually or by groups
  - Accepts execution requests from user
  - Matches application profile to node profiles to select target nodes
    - Can schedule jobs at locations closest to source data
  - Dispatches job with executable and data as necessary (scheduler)
  - Monitors execution of jobs and can reschedule failed jobs
    - IP based heartbeat between node and grid manager
    - Node removed from group if heartbeat stops
  - Receives results and may aggregate before returning to user
  - Returns results to user

Appendix B: Service Oriented Architecture

Software Architecture DNA - SOA



What is SOA?

SOA enables connectivity of applications or resources by:

- Representing every application or resource as a service with a standard interface
- Enabling them to exchange structured information.

Why do our customers care?

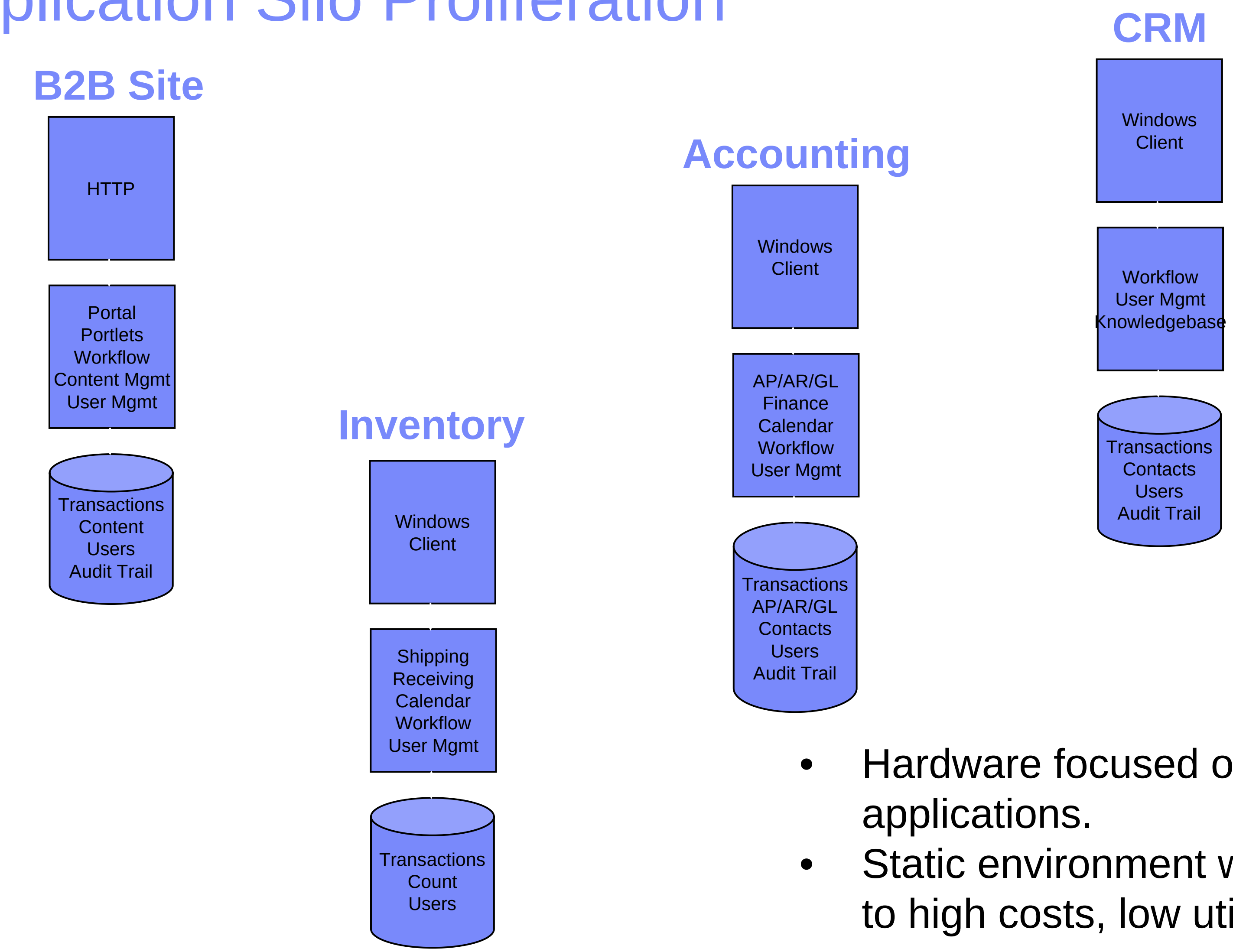
Customers need the flexibility to treat business processes and the underlying infrastructure as components that can be mixed and matched at will.

Service Oriented Architecture Key Principles

1. **An architectural approach allowing an application to be composed by independent, distributed and co-operating components called Services**
2. **Functionality implemented by a service is exposed via a standard-based interface declaration**
3. **Implementation details are hidden from the users of the service**
4. **Users invoke services based on the operations exposed by interfaces**
5. **Most often implemented as web services**

Standardized Abstraction Drives Flexibility

Application Silo Proliferation



B2B Site

HTTP

Portal
Portlets
Workflow
Content Mgmt
User Mgmt

Transactions
Content
Users
Audit Trail

Inventory

Windows Client

Shipping
Receiving
Calendar
Workflow
User Mgmt

Transactions
Count
Users

Accounting

Windows Client

AP/AR/GL
Finance
Calendar
Workflow
User Mgmt

Transactions
AP/AR/GL
Contacts
Users
Audit Trail

CRM

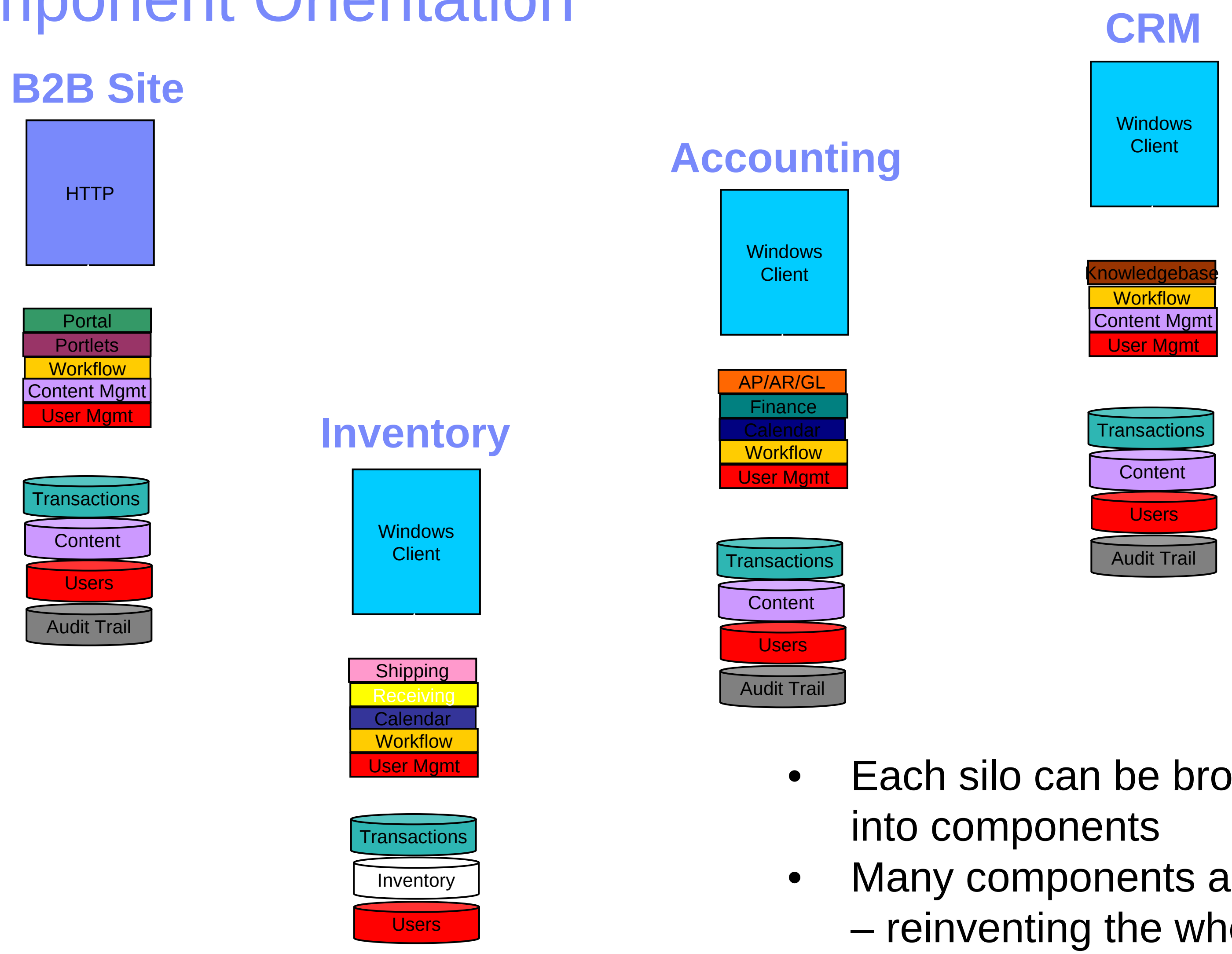
Windows Client

Workflow
User Mgmt
Knowledgebase

Transactions
Contacts
Users
Audit Trail

- Hardware focused on individual applications.
- Static environment which leads to high costs, low utilization and

Component Orientation



B2B Site

HTTP

Portal
Portlets
Workflow
Content Mgmt
User Mgmt

Transactions
Content
Users
Audit Trail

Inventory

Windows Client

Shipping
Receiving
Calendar
Workflow
User Mgmt

Transactions
Inventory
Users

Accounting

Windows Client

AP/AR/GL
Finance
Calendar
Workflow
User Mgmt

Transactions
Content
Users
Audit Trail

CRM

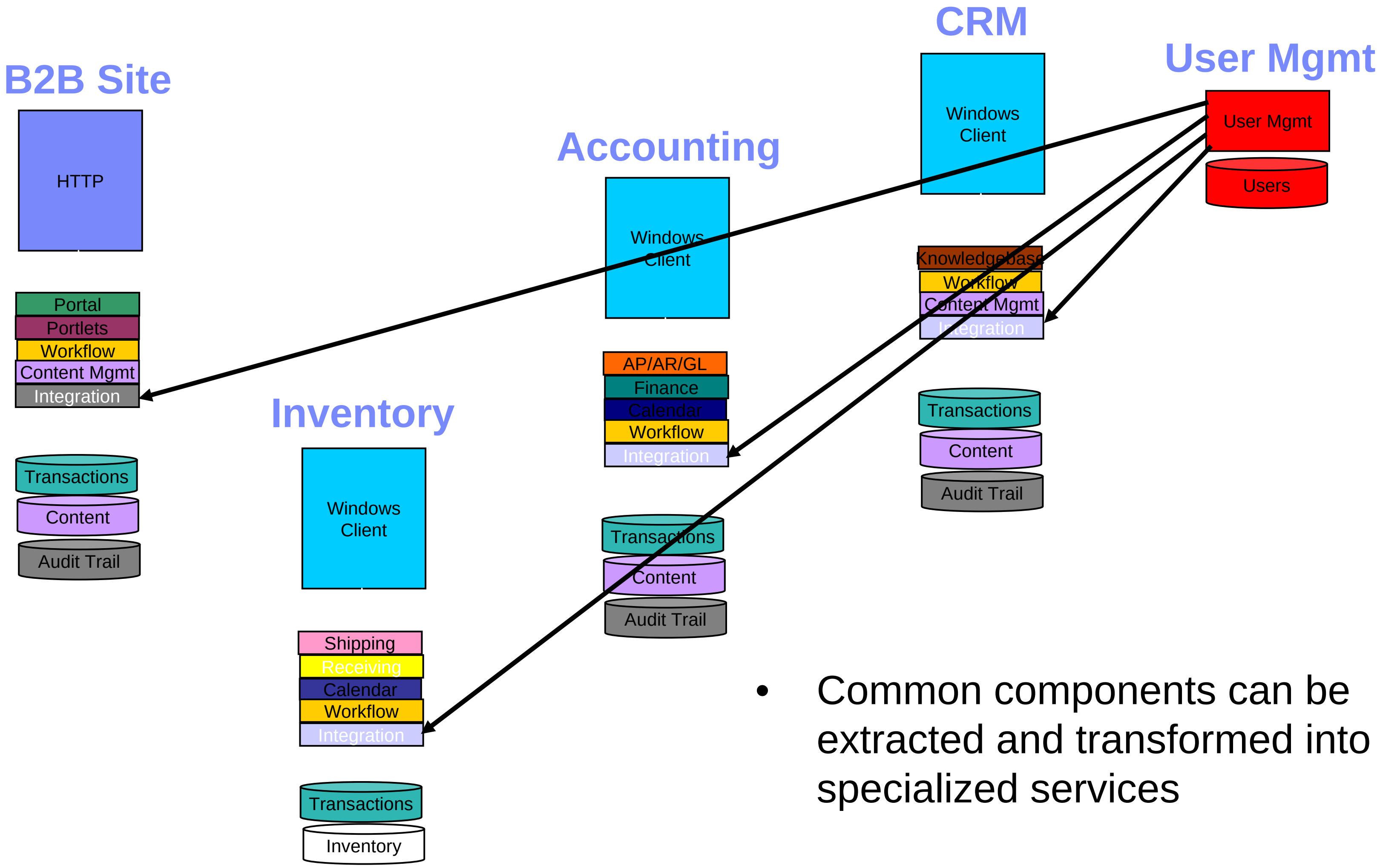
Windows Client

Knowledgebase
Workflow
Content Mgmt
User Mgmt

Transactions
Content
Users
Audit Trail

- Each silo can be broken down into components
- Many components are common – reinventing the wheel

Specialization through Services



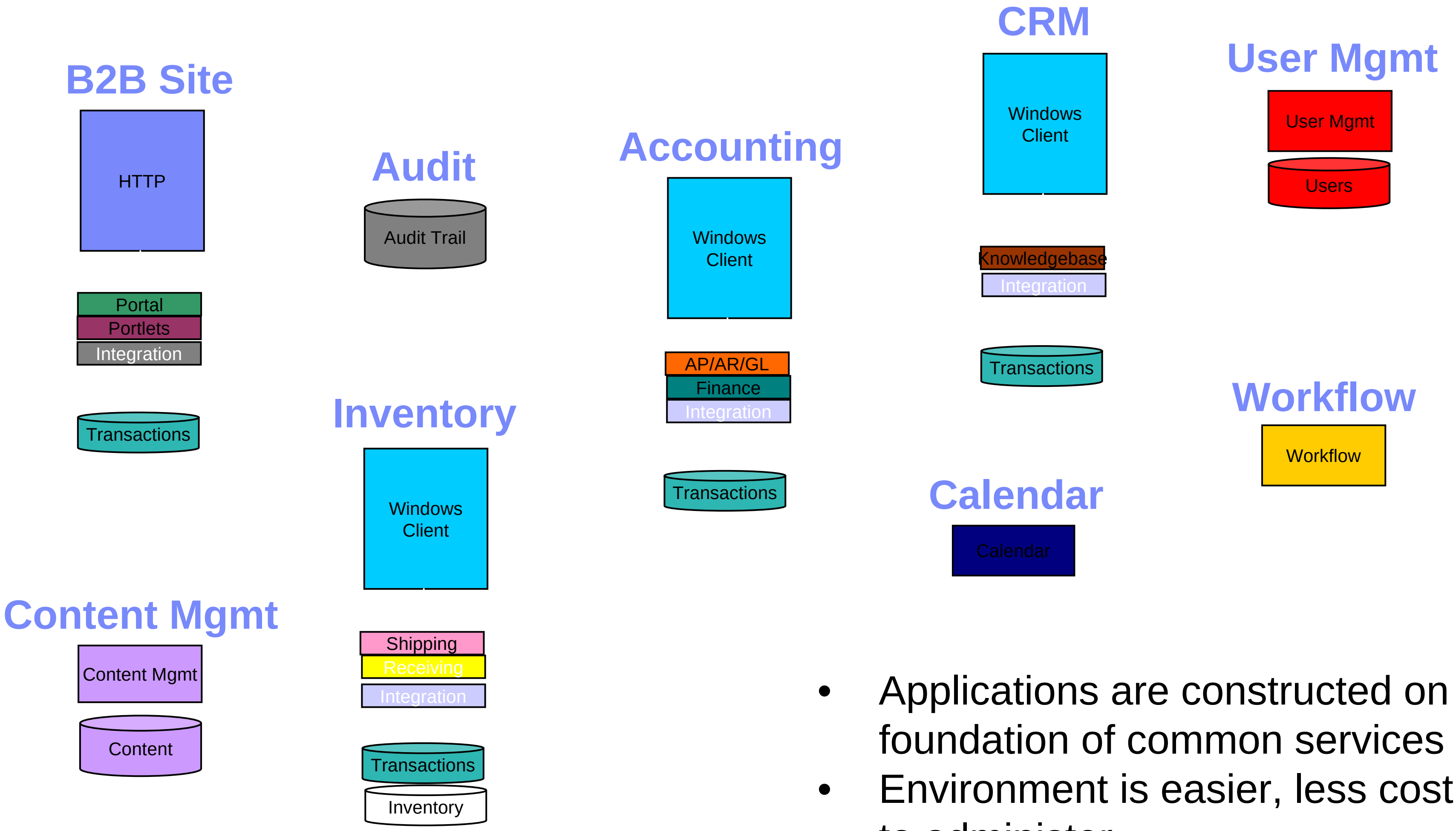
- Common components can be extracted and transformed into specialized services

User Management Service

- **Single interface for developers**
- **Available to all applications**
- **Common user definition**
- **Customizable profiles**
- **Single security mechanism**
- **Scalable independent of applications**
- **Integrated management tools**
- **Availability defined by dependent systems**

The above benefits hold true for all services!

Service Orientation



- Applications are constructed on foundation of common services
- Environment is easier, less costly to administer

Trademarks

The following are trademarks of the International Business Machines Corporation in the United States and/or other countries. For a complete list of IBM Trademarks, see www.ibm.com/legal/copytrade.shtml: AS/400, DBE, e-business logo, ESCO, eServer, FICON, IBM, IBM Logo, iSeries, MVS, OS/390, pSeries, RS/6000, S/30, VM/ESA, VSE/ESA, Websphere, xSeries, z/OS, zSeries, z/VM

The following are trademarks or registered trademarks of other companies

Lotus, Notes, and Domino are trademarks or registered trademarks of Lotus Development Corporation
Java and all Java-related trademarks and logos are trademarks of Sun Microsystems, Inc., in the United States and other countries
LINUX is a registered trademark of Linux Torvalds
UNIX is a registered trademark of The Open Group in the United States and other countries.
Microsoft, Windows and Windows NT are registered trademarks of Microsoft Corporation.
SET and Secure Electronic Transaction are trademarks owned by SET Secure Electronic Transaction LLC.
Intel is a registered trademark of Intel Corporation
* All other products may be trademarks or registered trademarks of their respective companies.

NOTES:

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.

This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

References in this document to IBM products or services do not imply that IBM intends to make them available in every country.

Any proposed use of claims in this presentation outside of the United States must be reviewed by local IBM country counsel prior to such use.

The information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.