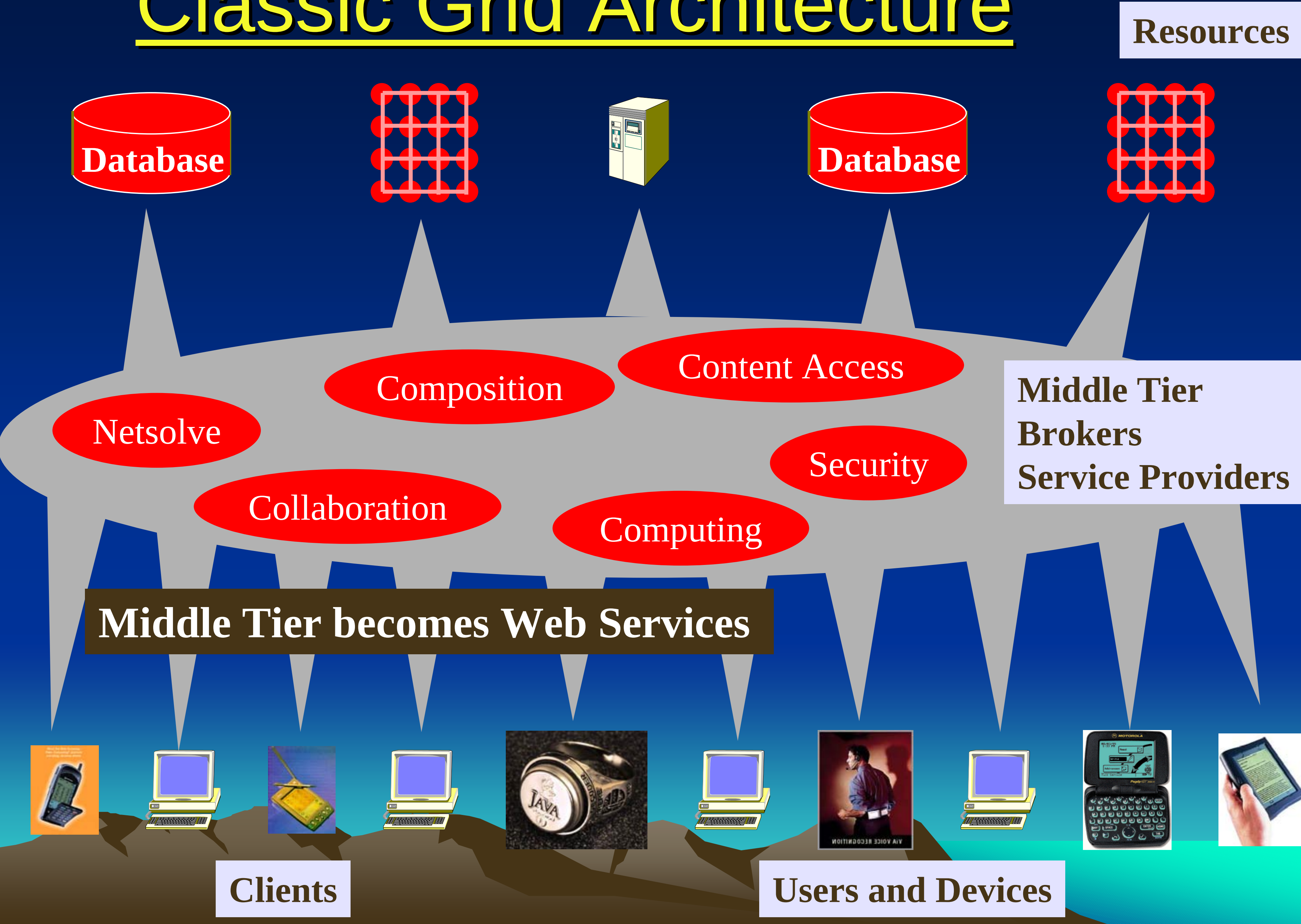
Peer-to-Peer Grids

Classic Grid Architecture



Resources

Database

Database

Content Access

Composition

Netsolve

Security

Collaboration

Computing

Middle Tier
Brokers
Service Providers

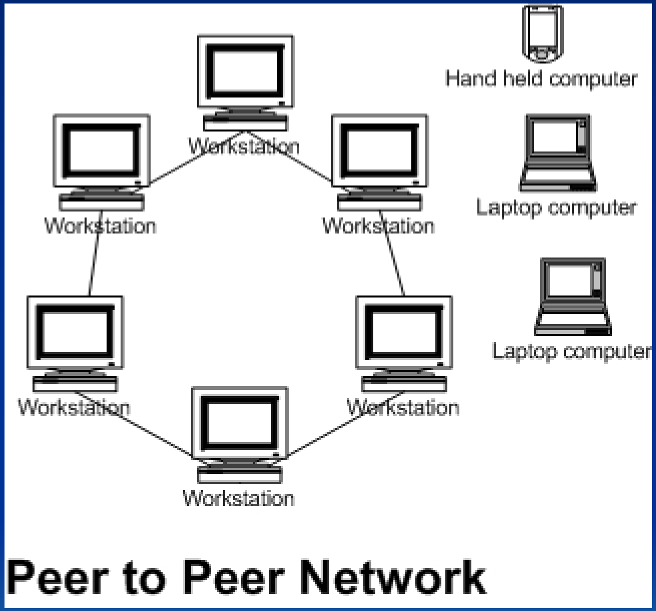
Middle Tier becomes Web Services

Clients

Users and Devices

Traditional P2P Network

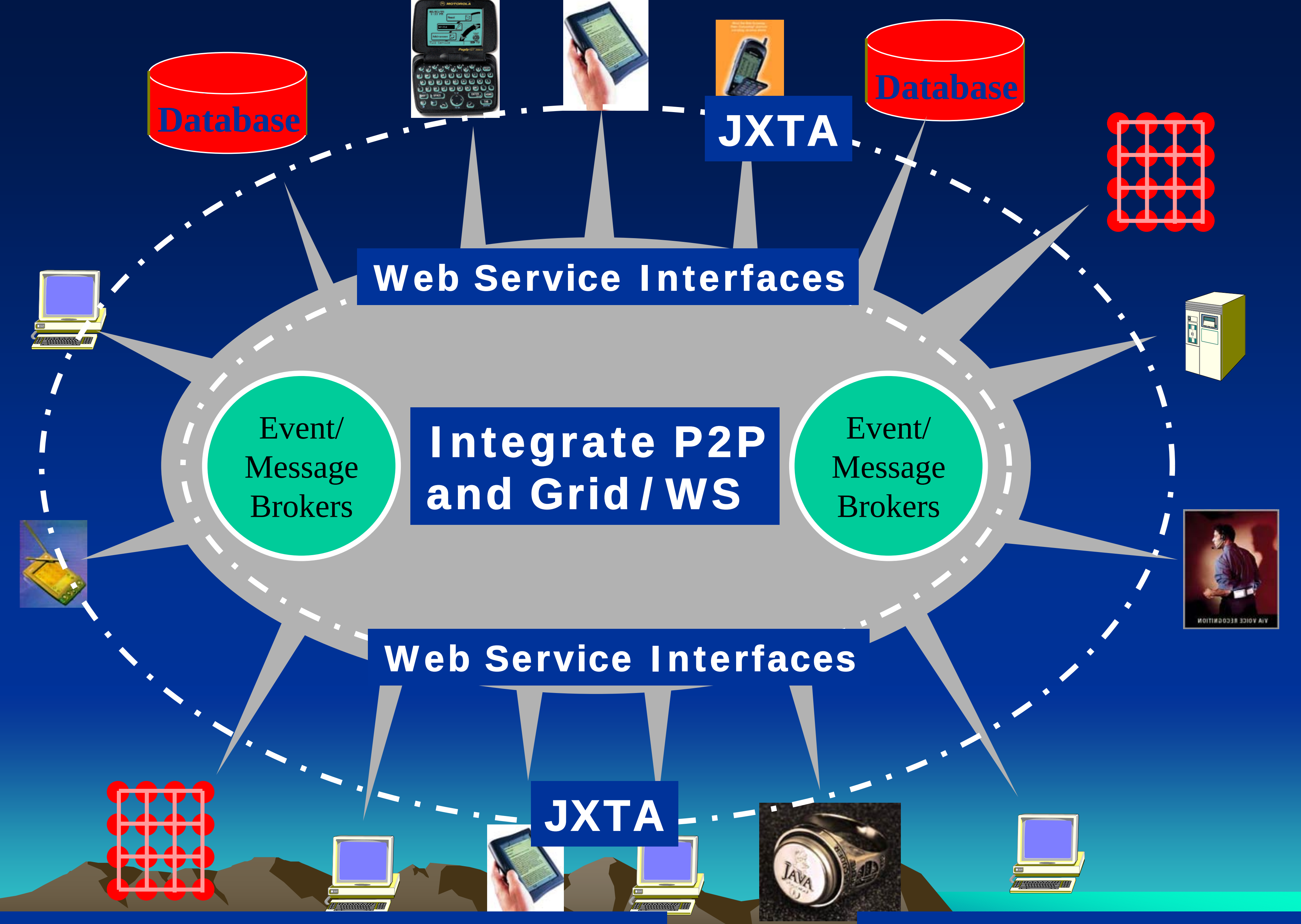
- The global sharing of resources for a specific task.
- Homogeneous resources within a distrustful environment
- Special uses such as SETI.



Peer to Peer Network

Peer-to-Peer Networks

  - Resource Sharing & Discovery
    - CPU cycles: SETI@home, Folding@HOME
    - File Sharing: Napster, Gnutella
- Deployments user driven
  - No dedicated management
- Management of resources
  - Expose resources & specify security strategy
  - Replicate resources based on demand
- Dynamic peer groups, fluid group memberships
- Sophisticated search mechanisms
  - Peers respond to queries based on their interpretations
  - Responses do not conform to traditional templates.



Database

Database

JXTA

Web Service Interfaces

Event/ Message Brokers

Integrate P2P and Grid / WS

Event/ Message Brokers

Web Service Interfaces

JXTA

A democratic organization

Peer to Peer Grid

Grid Web Services

- The grid web services uses the Open Grid Services Architecture (OGSA) to facilitate operations between clients and backend resources.
- The OGSA is the main link between resources such as databases and clients, users, and devices.
- This is accomplished through distributed object technology.

Web Services Definition Language

- The Web Services Definition Language (WSDL) is an XML-based IDL (Interface Definition Language) language.

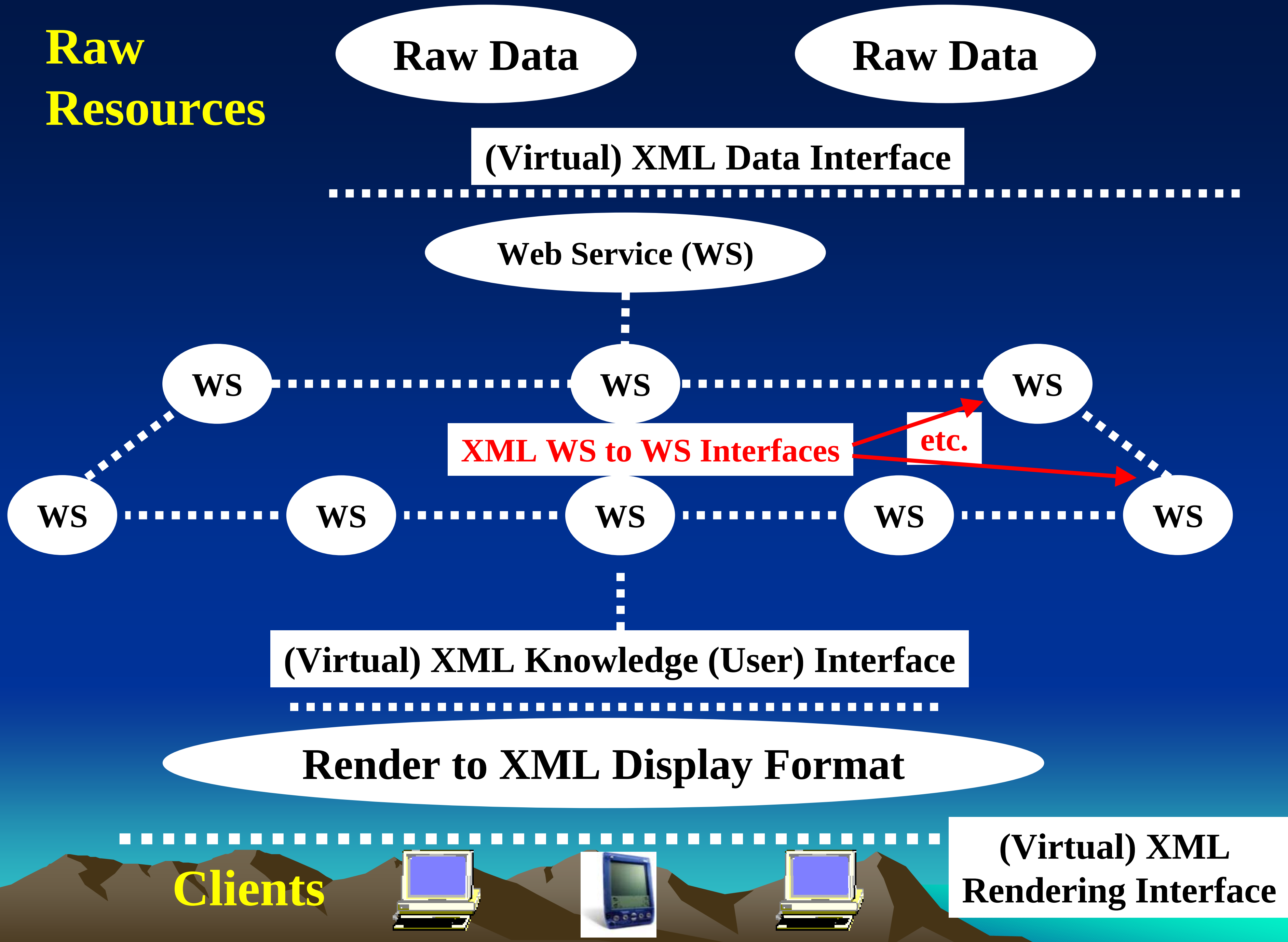
- WSDL is used as a service description layer within the web service protocol stack. It is an XML based language for specifying a public interface for a web service.

- Not tied to any specific messaging system, has some extensions for SOAP services.

- Can allow use of CORBA or Java RMI to be used with an XML wrapper to provide a uniform interface.

XML and IDL

- XML stands for E**X**tensible **M**arkup **L**anguage
- XML is a **markup language** much like HTML
- XML was designed to **describe data**
- XML uses a **Document Type Definition** (DTD) or an **XML Schema** to describe the data



Raw Resources

Raw Data

Raw Data

(Virtual) XML Data Interface

Web Service (WS)

WS

WS

WS

XML WS to WS Interfaces

etc.

WS

WS

WS

WS

WS

(Virtual) XML Knowledge (User) Interface

Render to XML Display Format

Clients

(Virtual) XML Rendering Interface

Service Model

- A service accepts one or more inputs and gives some type of output.
- In WSDL, these inputs and outputs are known as “ports”.
- WDSL defines a structure for these messages.
- Allows the interoperability of services from different sources.

Areas on which the Peer-to-Peer Grid is Built

- Basic capabilities for web services. Such as security, access to computers and databases. The databases include LDAP (Lightweight Directory access protocol and XML).

- Messaging subsystems: Between web services and resources addressing such issues as fault tolerance and performance.

- Creation of libraries and toolkits containing web services. Libraries for bioinformatics, gene searching etc…

- Provide services for network monitoring and collaboration.

- Use of the services of the “Semantic” grid to discover grid resources.

- User interfaces using portals on web services.

Implementation of Web Services

- Access to resources can be done by “servers” or by direct P2P interactions.
- The distributed object paradigm allows either of these two implementations. This is due to the fact that there is a separate service and message layer.
- The performance is the deciding factor on the implementation choice.
- P2P is the best for local dynamic use and server best for global but not short-lived use.

Peer-to-Peer Grid

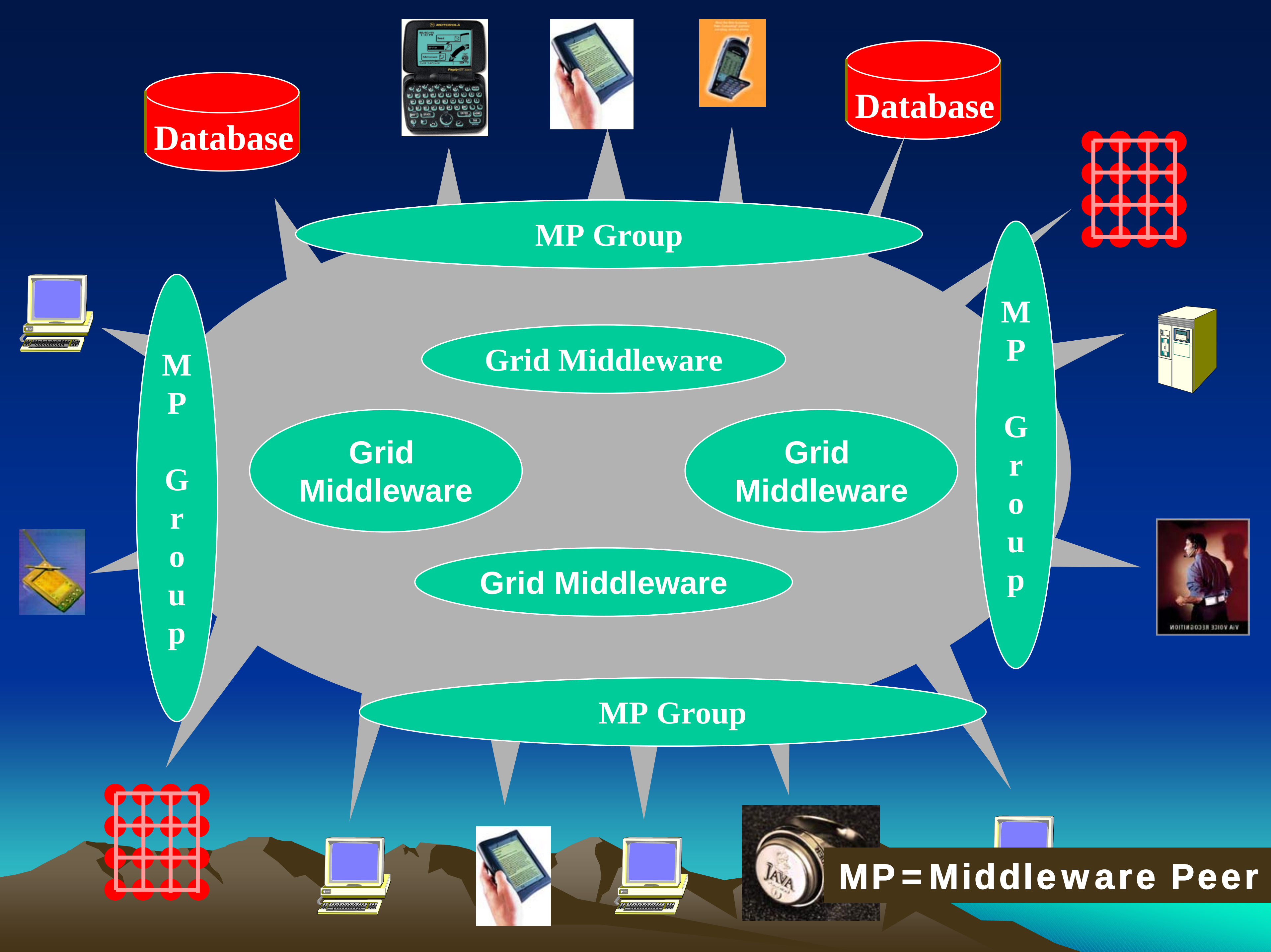
- A P2P grid with peer groups managed locally arranged into a global system supported by servers.

- Grids would control the central servers while services at the edge are grouped into “middleware peer groups”.

- In this case the P2P technologies are part of the services of the middleware.

Peer-to-Peer grid continued

- Can use the JXTA search technology to use middle-tier database systems.
- Can use grid or P2P technologies for organization and management of services.



Database

Database

MP Group

MP Group

Grid Middleware

Grid Middleware

Grid Middleware

Grid Middleware

MP Group

MP Group

MP=Middleware Peer

Peer-to-Peer Grid Event Services

- The event service provides the messaging between web services and resources.

- The idea is to use a single messaging subsystem to provide service to all forms of different technologies such as TCP/IP, RMI, UDP, XML, etc…

Event Service Implementation

- The event service is implemented through messaging by three different ways:
- 1) Simple Object Access Protocol (SOAP)
- 2) The JXTA peer-to-peer Protocol
- 3) The Java Message Service (JMS)

Simple Object Access Protocol (SOAP)

- SOAP is an XML based protocol used for exchanging information by remote procedure calls transported via HTTP.
- Easily invokes remote services and methods.
- Can be used in both grid and P2P Networks.

JXTA Protocol

- JXTA short for Juxtapose is a Sun technology used on peer-to-peer networks.

- It creates a virtual network where any peer can interact with other peers and resources directly.

- Advantages include:
- Interoperability - across different peer-to-peer systems and communities

- Platform independence - multiple/diverse languages, systems, and networks

- Ubiquity - every device with a digital heartbeat

Java Message Service

- The Java Message Service (JMS) is a messaging standard that allows application components based on the Java 2 Platform to create, send, receive, and read messages.
- It enables distributed communication that is loosely coupled, reliable, and asynchronous.

Event service continued

- The servers provide services at the event service level.

- An event and a message are defined by an XML schema and can be a resource also.

- Event service uses some form of the “publish-subscribe” system.

- Messages are queued from “publishers” and then clients “subscribe” to them.

Routers and brokers

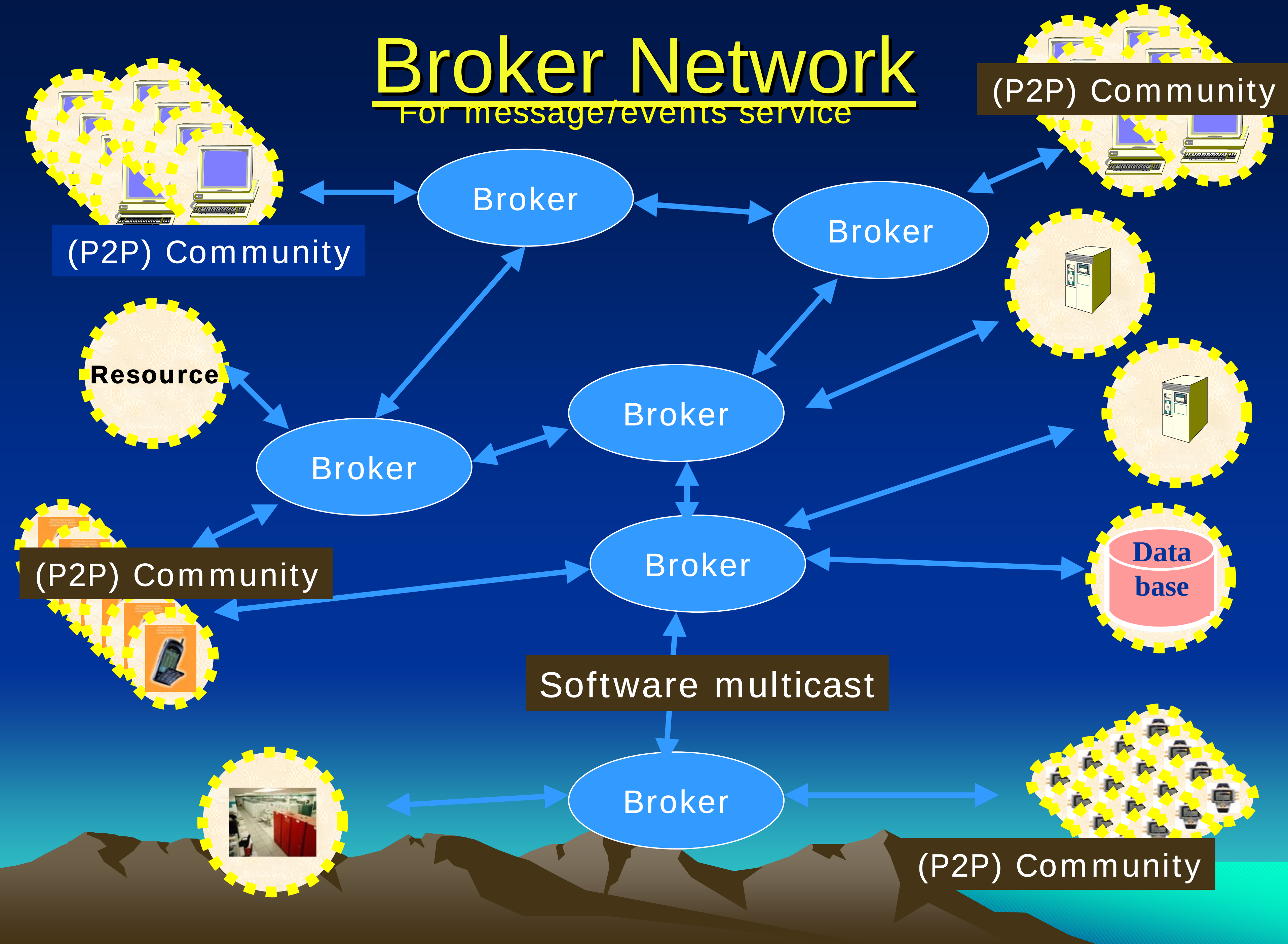
- Provide message or event services whereas servers provide traditional distributed object services as Web services.

- Only depend on event itself and perhaps the data format; they do not depend on details of application and can be shared among several applications

- These are called “event brokers” by the authors or are called “rendezvous peers” in JXTA.

Broker Network

For message/events service



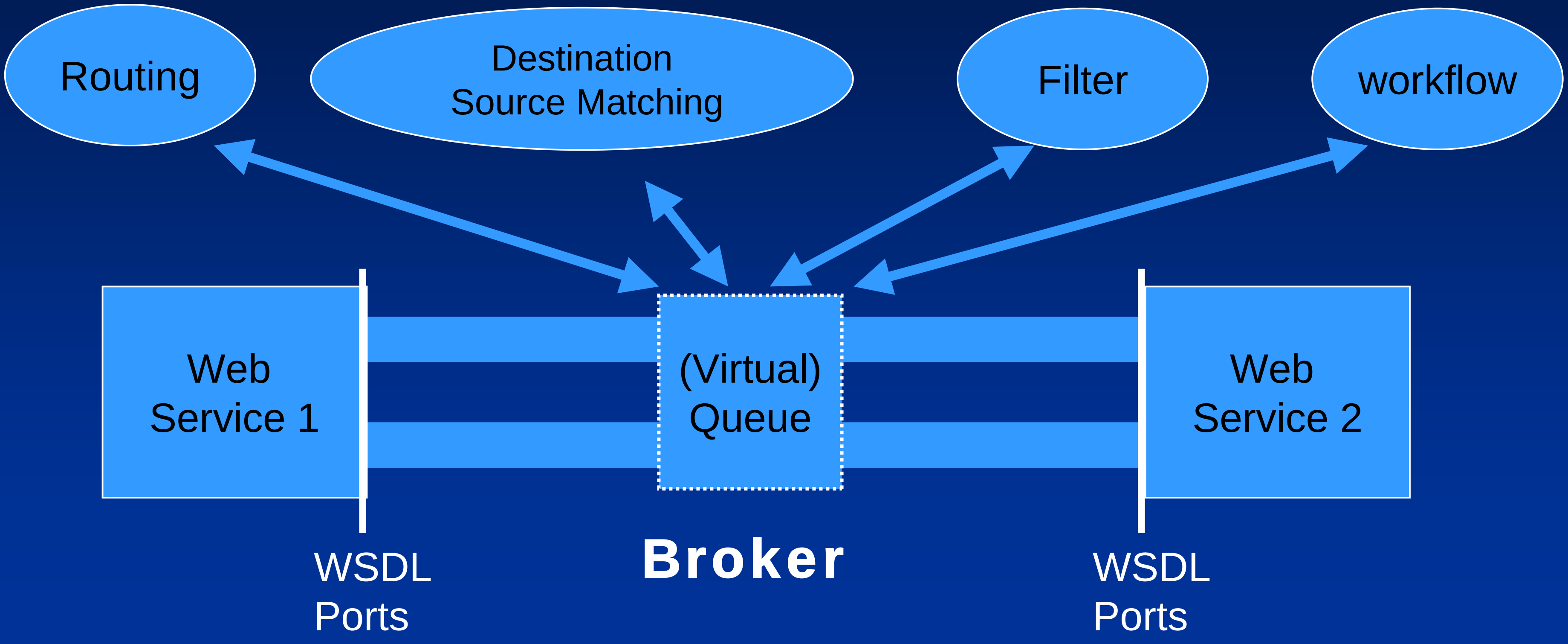
Event service continued-2

- All services that are independent of the web service that produced the message are gathered together.

- Such services that depend on message headers (destination) , message formats (multimedia), or message processes (such as publish-subscribe).

- Figure 18.9 event service architecture

Event Web Service Architecture



Event service Architecture

- Filter: is mapping to PDA or slow communication channel (universal access) – see our PDA adaptor
- Workflow: implements message process
- Routing: illustrated by JXTA
- Destination-Source: matching illustrated by JMS using Publish-Subscribe mechanism

Event service architecture

- The event service architecture supports communication channels between web services which can be direct or pass through.

- Can be low-level events such as routing or higher-level such as publish-subscribe.

- Messages must support multiple interfaces such as: SOAP, filtering, pub-sub, collaboration, and workflow.

- XML is an example which defines all the above services.

Collaboration in P2P Grids

- Both P2P networks and grids are used in collaborative environments.
- P2P are used in “ad hoc” environments and grids are parts of virtual organizations.
- The collaboration deals with sharing of web services, objects, and resources.
- Two main approaches to collaboration asynchronous and synchronous.

Asynchronous collaboration

- In asynchronous collaboration different members have access to the same resource.
- Such as a web page being updating by one and accessed by many.
- Asynchronous collaboration has no time constraint, is fault-tolerant, and uses caching techniques to improve performance.

Synchronous collaboration

- Synchronous collaboration is more difficult than asynchronous.

- Fault-sensitive , has modest real-time constraints and requires fine grain object states.

- Example: Recovering from an error. Does not have a mechanism to go back to original place.

Sharing mechanisms for collaboration

- For asynchronous or synchronous the sharing mechanism is the same.
- Need to establish peer groups by either direct or indirect methods.
- In direct methods the members join a specific session. Audio-Video conferencing.
- In indirect the members express an interest and is used in P2P systems with JXTA using XML.

Asynchronous collaboration methods

- In an asynchronous collaboration the pub-sub method is used .
- In this method the authors assume that every Web service has one or more ports in each of three classes.
- 1) Resource –facing input ports: which supply information used to define the state of the web service.
- 2) User-facing input ports: allow control information to be passed by the user.
- 3) User-facing output ports: Supply information used to create the user interface.

Synchronous collaboration methods

- Three cases exist in synchronous collaboration shared display, shared input port, and shared user-facing output port.

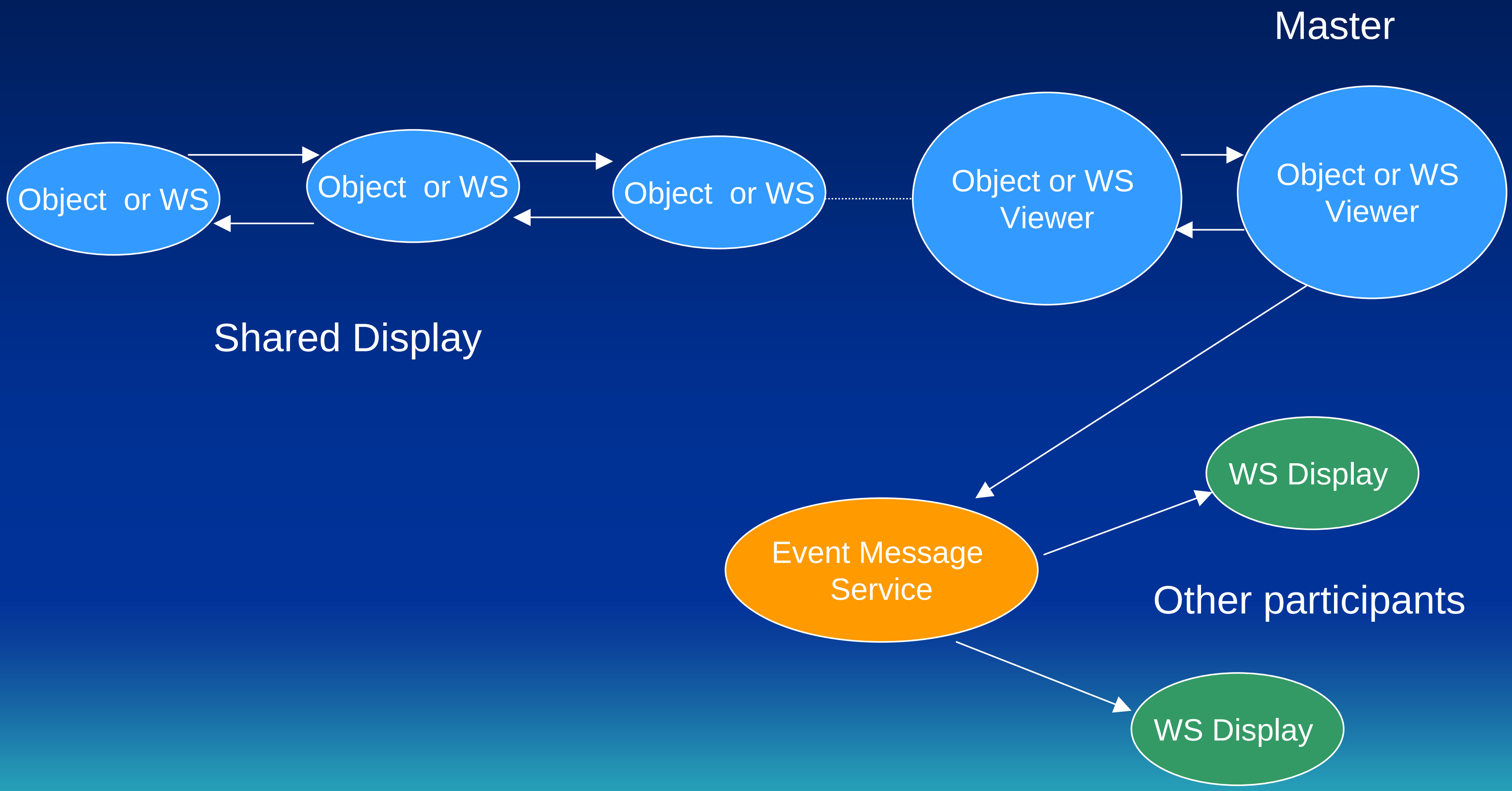
- In shared display model: a bitmap display is shared and the state is maintained between the clients by transmitting the changes in the display.

- Uses multiple event types with full and update displays.

- Key advantage – can immediately be applied to all shared objects.

- Disadvantages: Difficult to customize and require a lot of bandwidth.

Shared Display Collaboration



Shared input port

- Shared input port collaboration: a replication is made of the Web service with one copy for each client.

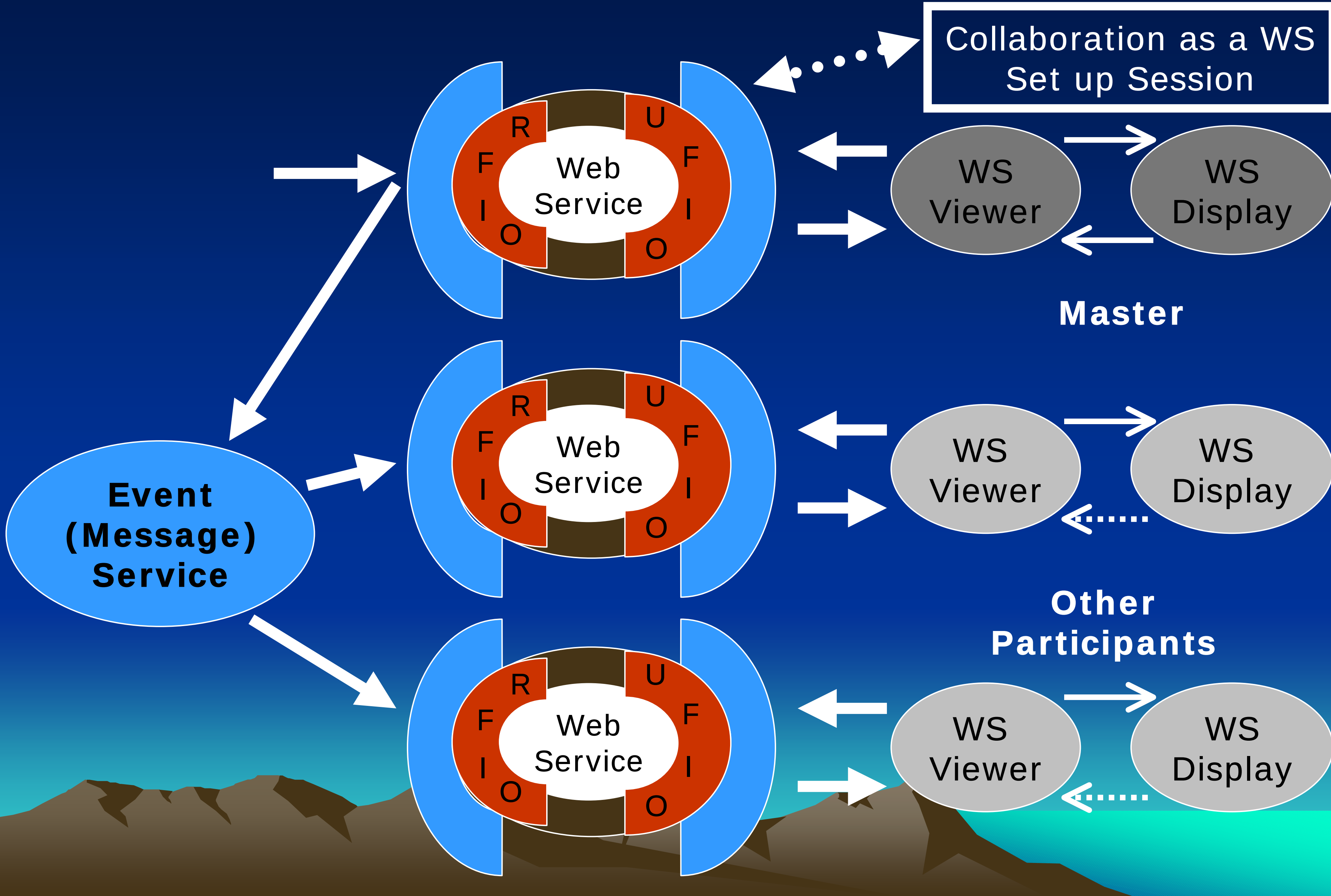
- The sharing is achieved by intercepting the Web service and directing copies of the messages on each input port to the replicated copies.

- Example: All clients have a copy of PowerPoint. On the master client uses a form of COM wrapper to detect PowerPoint events.

- The changes are then sent to all participating clients.

- Still being developed for replicated Web services with shared input ports.

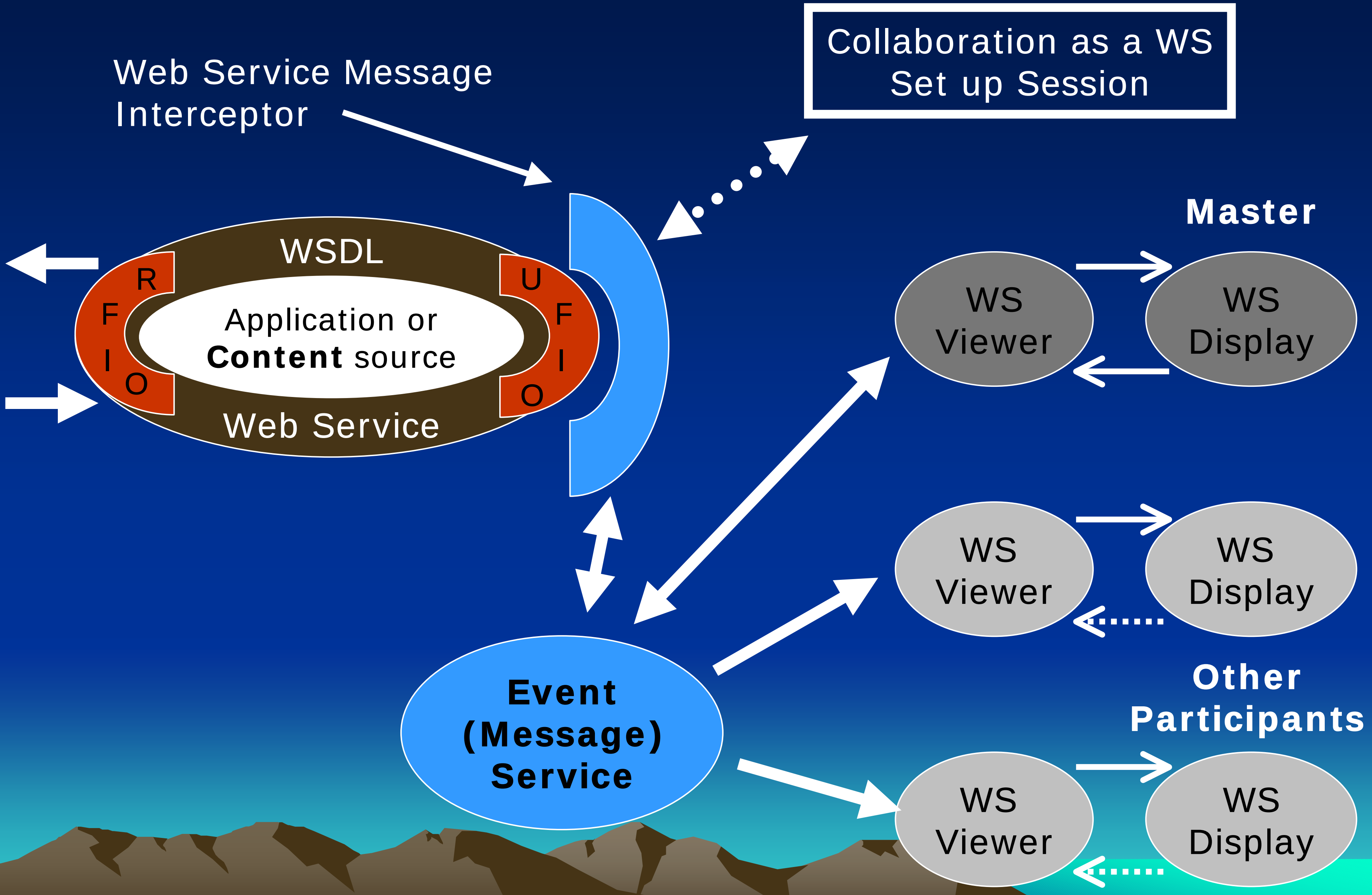
Shared Input Port (Replicated WS) Collaboration



Shared output ports

- Shared output port collaboration: uses a single Web service with ports providing interfaces to the client.
- The user-facing ports defines a user interface.
- Example: A multimedia server with multiple output streams. Gives identical view for each user but with less bandwidth than shared display model.

Shared Output Port Collaboration



User interfaces and Universal Access

- Universal access means a user interface which has the capability to define the interaction between the user profile and the web service.

- Three User-facing ports are used for the interaction:

- 1) Main user-facing output port: delivers a menu with many possible views.

User interfaces and Universal Access

- 2) Customized user-facing output port: delivers the view from 1) to the Web service
- 3) Input/output ports: Is the main control channel in figure 18.14.
- Example: Apache portal – “event service” for all web services.
- Shares workflow, filters, and collaboration.

NaradaBrokering

- Based on a network of cooperating broker nodes
  - Cluster based architecture allows system to scale to arbitrary size

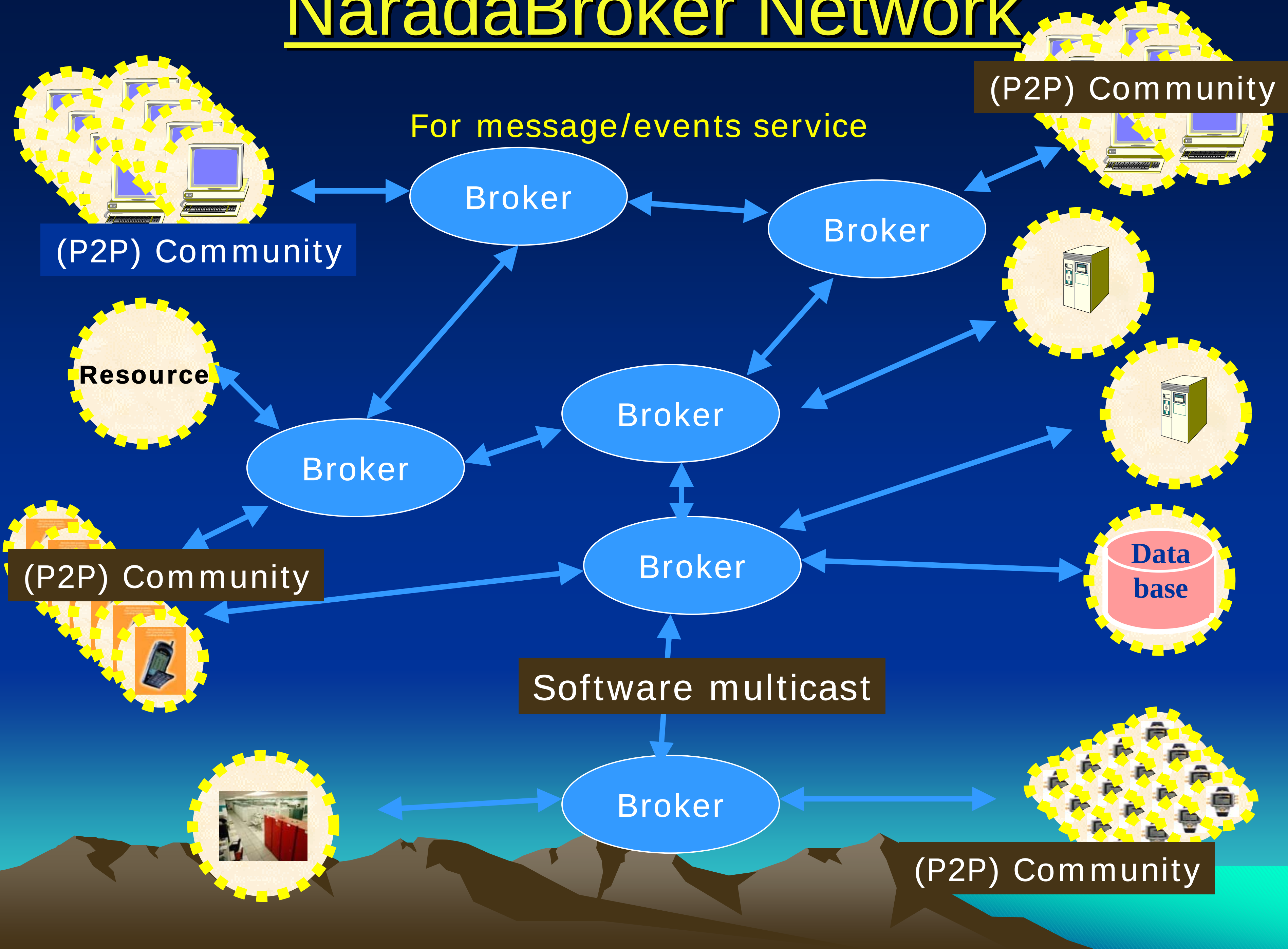
- Originally to provide uniform software multicast to support real-time collaboration linked to publish-subscribe for asynchronous systems.

- Incorporates algorithms for
  - Topic matching and calculation of destinations
  - Efficient routing to computed destinations

NaradaBrokering continued

- Now has four major core functions
  - Message transport: (based on performance) in multi-link fashion
  - Publish-subscribe including JMS & JXTA
  - Support for RTP-based audio/video conferencing.
  - Federation of multiple instances (just starting) of Grid services

NaradaBroker Network



(P2P) Community

For message/events service

Broker

Broker

(P2P) Community

Resource

Broker

Broker

(P2P) Community

Broker

Data base

Software multicast

Broker

(P2P) Community

Bibliography

- “Grid Computing : Making the Global Structure a Reality” by Berman, Fox, and Hey. 2003 Wiley publishers.

- http://www.naradabrokering.org

- “The GRID2: The Blueprint for a New Computing Infrastructure” by Foster and Kesselman. 2004 Morgan Kaufamnn publishers