e"speak.
The universal language of e-services

Alan H. Karp
Chief Scientist
Open Services Operation
Hewlett-Packard

http://www.hp.com/go/espeak
The Essential Difference

Hardware + Software

Tell the computer how to do the job

Services

Tell the computer what job you want done
What is E-speak?

E-speak is an open services platform for the

– creation,
– composition,
– mediation,
– virtualization,
– management, and
– accessing

of Internet-based services.
Technology Goal

Do for services what the Web has done for data

Make it as simple, in fact simpler and safer to create, compose, deploy, manage, personalize, and access services as it is to publish and access data on the Web.

Service: anything that can be transmitted digitally, including access to the communication channel itself.
E-Speak Technology Stack

- Infrastructure Services
- Financial Intermediary (E-Speak broker)
- Service
- Service

E-Speak Service Bus

E-Speak Service Framework

Other Frameworks (BizTalk, eCo, FIPA)

Web Access

NOM Access

E-Speak Core

Transport

The next E. E-services.
Why e-speak?
New services and service compositions,
New service providers

New customers

The next E. E-services.
Services Framework
Reducing the barrier to new, competitive services

Service Specification
- Workflow
  - Requirements
  - Access control
  - (security, billing, ...)

Service Discovery

Service Execution

Service Access
Internet Challenges

- Today’s e-business web sites are proprietary, massive and costly to develop.
- Companies are forced to build out their entire offerings from the ground up.
- Even though they are connected to the Net, getting e-businesses and e-commerce sites to talk to one another in a meaningful way is difficult, special-case work.

The volume of business is limited by the bandwidth of eyeballs.
The Big Shifts Coming

- Ubiquitous e-services
- Modular building blocks
- Easy access from a wide array of devices and platforms:
  - Info appliances
  - PCs
  - Servers
  - Supercomputers
- E-services talk to each other in order to:
  - advertise capabilities
  - discover and ally with services offering new capabilities
  - negotiate to broker, bill, manage and monitor each other
- E-services interact with each other in a way that ensures security
E-speak Origins

• 1982 - Joel Birnbaum, Information Utility
• 1985 - Bill Rozas, I just want to be me
• 1989 - Alan Karp, Global Computer
• 1990 - Rajiv Gupta, Use obsolete machines
• 1994 - Arindam Banerji, Extensible OSes
• 1996 - Rajiv Gupta, World of services
Monolithic, proprietary systems

Open systems
- 2-tier client-server systems

Open data (Web)
- 3-tier, 4-tier, … systems
- Proprietary, one-of services (Amazon.com, Expedia, eBay, …)

Open services (E-speak)
- Dynamic n-tier systems
- Brokered service composition (active personalization)

The next E. E-services.
Assumptions and Implications

- Large number of machines
  - No centralized anything, forget consistency
- Dynamic
  - Deal with failures, new services
- Heterogeneous
  - Different hardware, OS, capability
- Hostile environment
  - Security is critical
- Different fiefdoms
  - Never look inside another machine

The next E. E-services.
Architectural Principles

➤ Design for seamless, flexible, dynamic evolution
   ■ Current and future
   ■ Scalable, manageable, secure, extensible

➤ Simple and elegant abstractions and mechanisms
   ■ No "special-case" mechanisms, homogeneity requirements
   ■ Uniform abstractions for services and resources
   ■ Resource access virtualization and mediation

➤ Invent only where necessary
   ■ Leverage and complement industry standards
E-speak in Perspective

- Applications
- Kernel
- Operating System
- Physical Resources
e.g., CPU time slice, disk

The next E. E-services.
E-speak in Perspective

Kernel Operating System

Requests

Core

E-speak

Services

Applications

Operating System

Physical Resources e.g., CPU time slice, disk

The next E. E-services.
Client Utility Resource Model

Basic Services
- Scheduling
- Interrupt Handling
- Memory Management
- Timer Services

Legacy Resource Abstraction
(eg: NTFS)

Legacy OS
E-speak Resource Model

Virtual Resource Abstraction
- Naming
- Permissions
- Communication
- Attribute Descriptions
- Interfaces

Basic Services
- Authorization
- Attribute Vocabularies
- Transports
- Repositories

E-speak System
- LDAP
- DCOM/CORBA
- Java Platform
- HTTP

Legacy Resource Abstraction
(eg: NTFS)

Legacy OS
- Scheduling
- Interrupt Handling
- Memory Management
- Timer Services

The next E. E-services.
E-speak Environment

E-services Framework

Infrastructure e-services

E-speak Software

Int**net

HP + ...
Technology Innovations

- System Architecture
- Naming Model
- Security Model
- Meta-data Model
- Event Model
Key Abstractions

- Everything is a resource
- Naming
  - Only way to reference a resource
  - All names are private
- Security
  - Separate control of names and access rights
- Description
  - Customizable vocabularies
- Management
  - Every access mediated
Technology Innovations

System Architecture
- Mediated access to services
- Uniform resource model
- Manipulation of resource representations, not resource specifics

Creates Open Services Model for the Internet
- Anything can be created as a new service using same model
- Heterogeneous management tools, security policies can be applied without compromising simplicity
- New service types and semantics can be dynamically introduced
- Services can be seamlessly interposed and distributed even across firewalls
- Provide functionality to enable commerce in services, e.g., monitoring, auditing, billing

The next E. E-services.
System Structure

- Federation of Logical Machines
- Logical Machine
  - Active entity - Core
  - Passive component - Repository
- Mailbox metaphor for requests to Core
Fundamentals

- Every resource’s metadata registered with Core
- Tasks access resources by name
- Core associates name with resource metadata
Use Model

- Each task has an outbox connected to the Core
  - Outgoing message has envelope and payload
- Each task has zero or more inboxes
  - Incoming message has envelope and payload
- Core-related data in envelope
- Application data in payload
Single Machine View

Service User

Event Distributor

Monitor

Permission

Naming

Router

Repository

Name Space

Service Provider

Name Space

Host OS

The next E. E-services.
Connecting Two Machines
Distribution Model

- Seamless Distribution
  - Uniformity in service interactions
  - Support for both remote evaluation and remote operation
    - Dynamic loading is subsumed
  - Proxies mimic resource handlers
  - ESIP-ABI *defines* the inter-machine architecture
Architecture Overview

Dynamic Federation Model

Network
E-speak Thin Layer
Legacy Host OS
Native Hardware

The next E. E-services.
E-speak vs. Traditional Middleware

- Directory lookup
- Direct access to service
- No mediation
- No virtualization

- Virtual service intermediation
- Uniform services model
- Attribute-based lookup
- Individual name spaces
- Capability-based security

Traditional Middleware

Client → Directory → Server

Server

Client application → Advertising service → Core → Service

Proxies

e-speak

Core

Service

Service
Technology Innovations

Naming Model

- Local, context-sensitive naming
- Name associations allow resource lookup algorithms
- Partial associations provide hooks for external decision services

Client-Service provider bond scalably, can be reasoned about at run-time

- Services/agents do not require pre-negotiation
- Transformers can be seamlessly interposed
- Enables per-client, per-role, context-sensitive customization
- Enables hot-plug replacement, moving, mirroring of resources
Flexible Name Bindings

- Bind a name to
  - A resource
  - A set of resources
  - A look-up request
  - All of the above
- Name is bound to an algorithm for finding resource
- Can pass bindings between tasks
Security Model

- Access rights, capabilities are resources
- Separation of name visibility from access rights
- Remote access based on trust established between machines

Fine-grained, dynamic protected access to services

- Mapping between different security, authentication infrastructures
- Simple, selective yet dynamic delegation of privileges with revocation
Access Control

- **Name**
  - Client can only reference a resource by name
  - Name is local to client with mapping in name space

- **Rights**
  - Client presents keys that open locks
  - Core delivers unlocked permissions

- **Right to use name**
  - Keys are resources referenced by name
  - Keys express name visibility rules
Meta-data Model

- Vocabularies are resources
- Translations are integrated into design

Flexible, scalable services discovery and location

- Translation between XML-LDAP schema becomes a secure service
- Searches and service locations can be optimized through add-on services, e.g. find HP printer using Lexmark printer MIB grammar without requiring homogenization
- Advertising services can be used to scalably find remote resources
Lookup Usage

• Comparison shopping
  find ( "ServiceType == ‘AirLine’",
        "Path == ‘LAX-SFO’",
        "Cost < $90" );

• Locating services
  find ( "ServiceName == ‘Citibank’,
         "Location == ‘Sunnyvale’ OR ‘Santa Clara’" );

• XML specification support
Advertising Service

- Lookup in local Repository results in name binding
- Look in advertising service if not found
  - Get back a machine to contact
  - Ask for resource once connection established
- No permanent connection needed
  - Advertise in many places
  - Lookup in many advertising services
- Used to form communities

The next E. E-services.
Technology Innovations

Events

- Publish-Distribute-Subscribe
- Filters on subscription and publication
- Control of events with e-speak permissions

Flexible, controllable event infrastructure

- Used to build management infrastructure
- Appropriate for data consistency
- Discoverable events
- Unified model for management and application events
- Event state and filters based on Vocabularies
Event Handling

Message Loop → Event Callback

Normal Message → Message Queue
Programming Model

Three visible abstractions
- Service - invoked by client
- Contract - defines interface
- Vocabulary - describes service for discovery

Network Object Model
- Stubs provided by IDL
- Download Java stubs
- Reference stub by name

Direct messaging and Document Exchange also supported
main ( )
{
    ESConnection c = new ESConnection(argv[0]);

    String intf = "echo.echoService";
    ESServiceFinder sf = new ESServiceFinder(c,intf);
    EchoServiceIntf echoSvc = (EchoServiceIntf) sf.find("Name='echoServer'");

    String echo = echoSvc.echo("Hellooooooo");

    System.out.println(echo);
}
main ()
{
    ESConnection c = new ESConnection(argv[0]);

    ServiceComponent sc = new ServiceComponent(c, "echoServer");
    sc.setImplementation(new EchoServiceImpl("EchoServer"));
    sc.registerService();

    sc.start ();
}

public interface EchoServiceIntf // E-speak IDL
{  public String echo(String in); }

Class EchoServiceImpl implements EchoServiceIntf
{  String echo(String in) { return in; } }
Business to Business Procurement/ SCM

Current status:
1. Static/ preferred supplier
2. Hard-coded, not easily extensible
3. Integrated with ERP
4. EDI, cross-device not automated, security(?)
5. OK with VPN, and with real-time integration (complexity unknown)
6. Not fully automated
7. OK for centralized, else some user intervention
8. Not automated

Supplier discovery
Policy-based search browse
Obtain approval/ERP
Transmit purchase order
Sale order & confirmation receipt
Check status
Desktop delivery update
Invoice payment

Dynamic lookup, scalability
Transparency, seamless distribution
Integration with ERP systems
Virtualization, cross-device, security
Data heterogeneity
Cross-device, security, Data heterogeneity
Granting authorization, delegation cross-enterprise
Scan-device(?) security, cross-enterprise, delegation
Automated remittance, security

The next E. E-services.
E-speak BPC Framework

Collaborator A

E-speak Connector

Collaborator B

E-speak Components

The next E. E-services.
Build Service

IA-64
32/64-bit
Build

PA-RISC
11.0 (32/64-bit)
Build

E-speak
Core2

E-speak
Core3

PA-RISC
10.20 (PA1.1)
Build

HP E-make
Web Server

E-speak
Core1

Intranet

Ad
Agent

Client
PC / Unix

Browser
E-make
Inputs

User

The next E. E-services.
Truly Universal Environment

Ports
- Heart monitoring
- Print broker service
- Disk storage
- Document translation
- Real Estate relocation
- Streaming media
- ERP via web
- Dynamic VPN
- Supply chain brokering
- Virtual training
- Language translation
- etc.

Component Bus
- Billing
- Authentication
- Authorization
- Other Components

E-speak Adapter
- UPnP
- Jini
- XML
- Windows
- Java
- Devices
- Intel
- Devices
- Internet

Platforms
- Java
- C
- Python
- UPnP
- Jini
- XML
- Web
- Intel
- Devices

Other Components
- Heart monitoring
- Print broker service
- Disk storage
- Document translation
- Real Estate relocation
- Streaming media
- ERP via web
- Dynamic VPN
- Supply chain brokering
- Virtual training
- Language translation
- etc.

The next E. E-services.
E-speak’s Benefits

- Simplifies composition of services
- Dynamic creation of new value chains
- Spontaneous, ad-hoc, secure interactions across firewalls (without pre-negotiated names and standards)
- Slower rate of obsolescence
- Choice of standards
E-speak’s Unique Features

- No global name space
- Novel capability model for access control
- Vocabularies as resources
- Separation of control data and resource semantics
- System structure
- Consistent service interface
- Dynamic service virtualization
- Brokering, delegation, revocation
- WAN-based fine grain security
- WAN-based scalability
Summary

- E-speak will do for services what the Web has done for data
- New business opportunities will be driven by the spontaneous composition of services
- E-speak will fuel the Internet’s shift from the do-it-yourself model to the do-it-for-me model
E-speak Roll Out Plan

Brokers and Communities
  Relocation
  Insurance
  Instant Extranet
  Small business community
  Calendar and communication

1Q99
  Alpha 1.0

2Q99
  Beta 1.0

3Q99
  Beta 2.0

4Q99
  Open Source Offering
  Building Blocks
  Developer Program

1Q00
  Service Composition & Deployment Service

The next E. E-services.
Success Factors

- Technology
- Business Model
- Partnerships
- Technology (largest section)
Success Factors

- Partnerships
- Business Model
- Technology
- You

The next E. E-services.
What is E-speak?

E-speak is an operating environment for the Internet that reduces the barriers to creating e-services.
What You Can Do

• Try it out.
  • The price is right - free.
  • It’s easy to get started.
• Build e-services for new environments.
  • The whole world is not commercial services.
  • Find novel uses - device OS, educational software, etc.
• Build new programming models.
  • The whole world is not commercial services.
  • Define new abstractions for real time, scientific, etc.
• Work on the open source code.
  • Make your mark by fixing what we’ve done wrong.
• Join the open source board.
  • Seats are still available.

The next E. E-services.
http://www.hp.com/go/espeak