e-Business and Open Source Update

Mandar Chitale

Office of OpenVMS Customer Programs

Europe 2009 Technical Update Days

Agenda

- eBusiness and Integration Strategy
- Application Modernization and Integration Technology - Overview
- Web Servers and Browsers
- Application Development Tools
- Open Source Tools
- Case Studies
- Q&A

Agenda

- eBusiness and Integration Strategy
- Application Modernization and Integration Technology - Overview
- Web Servers and Browsers
- Application Development Tools
- Open Source Tools
- Case Studies
- Q&A

eBusiness and Integration Strategy

- Vision:

  Integration of new and existing data and applications into web and internet environment

- Strategy:

  Enhance the OpenVMS operating system with an infrastructure that allows application, middleware, and data integration in a global, multi-platform environment.

Agenda

- eBusiness and Integration Strategy
- Application Modernization and Integration Technology - Overview
- Web Servers and Browsers
- Application Development Tools
- Open Source Tools
- Case Studies
- Q&A

Application Modernization and Integration Technology - Overview

- **Web Servers and Browsers**
  - Secure Web Server (Apache) available with components:
    - mod_PHP (CSWS_PHP)
    - mod_Perl (CSWS_Perl)
    - mod_JK and Tomcat (CSWS_JAVA)
  - Secure Web Browser (based on SeaMonkey)
  - Firefox Web Browser for OpenVMS Integrity
  - Secure Web Browser (based on Mozilla)
- **Application integration**
  - TPware .NET
  - BridgeWorks
  - Attunity Connect®
  - CONNX®
- **Middleware**
  - ACMS
  - COM (OPC Transport I64)
  - Reliable Transaction Router (RTR)
  - OPC Transport for OpenVMS
  - Oracle MessageQ®
  - IBM MQseries®
  - 2AB orb2 (CORBA)
- **Application Development Tools**
  - Java™ Standard Edition Development Kit (JDK)
  - Distributed NetBeans
  - gSOAP
  - GNV
  - ANT
  - Axis2
  - Perl
  - Web Services Integration Toolkit (WSIT)
  - Simple Object Access Protocol (SOAP) Toolkit (Apache Axis)
  - XML Technology (parsers and stylesheet processors) (Apache)
  - UDDI Client Toolkit (UDDI4J)
  - eCube NXTware® Remote
- **Application Servers**
  - Concerto®
  - JBOSS

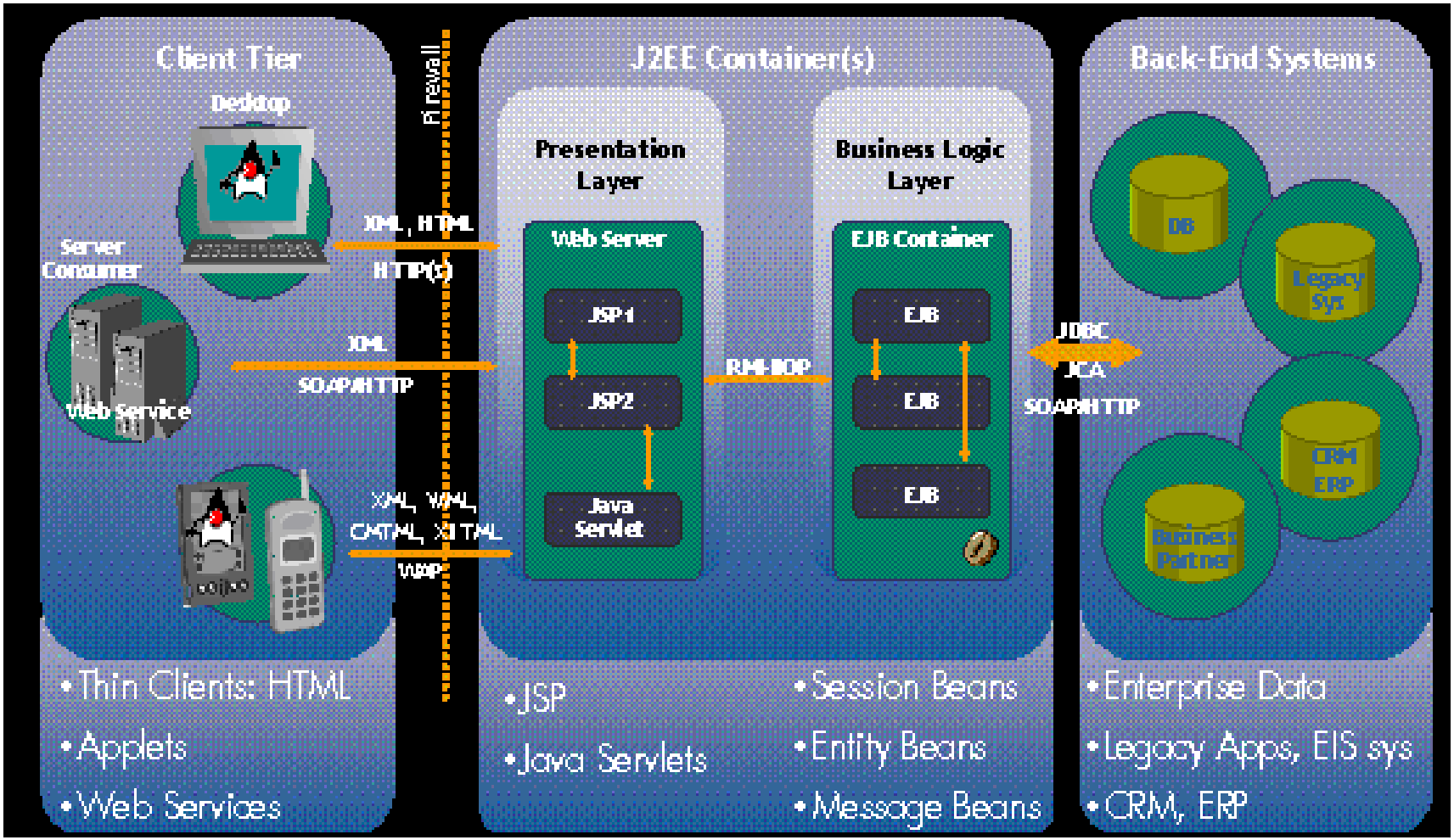
Application Modernization and Integration Technology

- Transparent and seamless integration to the internet while keeping all of the advantages of your OpenVMS systems
  - Internet technologies available on other platforms are available on OpenVMS
  - Data, applications, business logic, and processes that are currently running very well on OpenVMS, are exposed to the larger Internet world
- Business-to-business (B2B) application that your business partners use simply works, and they don't even notice that they are running programs on OpenVMS.

How they all work together?

- Back-end systems
  - JDBC and Attunity Connect
  - XML
  - SOAP
- Interconnection Tier
  - EJB,etc
- Client side
  - HTML
  - XML, etc

Applications Integrated



http://h71000.www7.hp.com/openvms/journal/v2/articles/internet_for_openvms.htm

Agenda

- eBusiness and Integration Strategy
- Application Modernization and Integration Technology - Overview
- Web Servers and Browsers
- Application Development Tools
- Open Source Tools
- Case Studies
- Q&A

Secure Web Server

- Comprises
  - Secure Web Server (Based on Apache)
  - CSWS_PHP (Based on mod_PHP)
  - CSWS_Perl (Based on mod_Perl)
  - Perl (Based on Perl)
  - CSWS_Java (Tomcat)

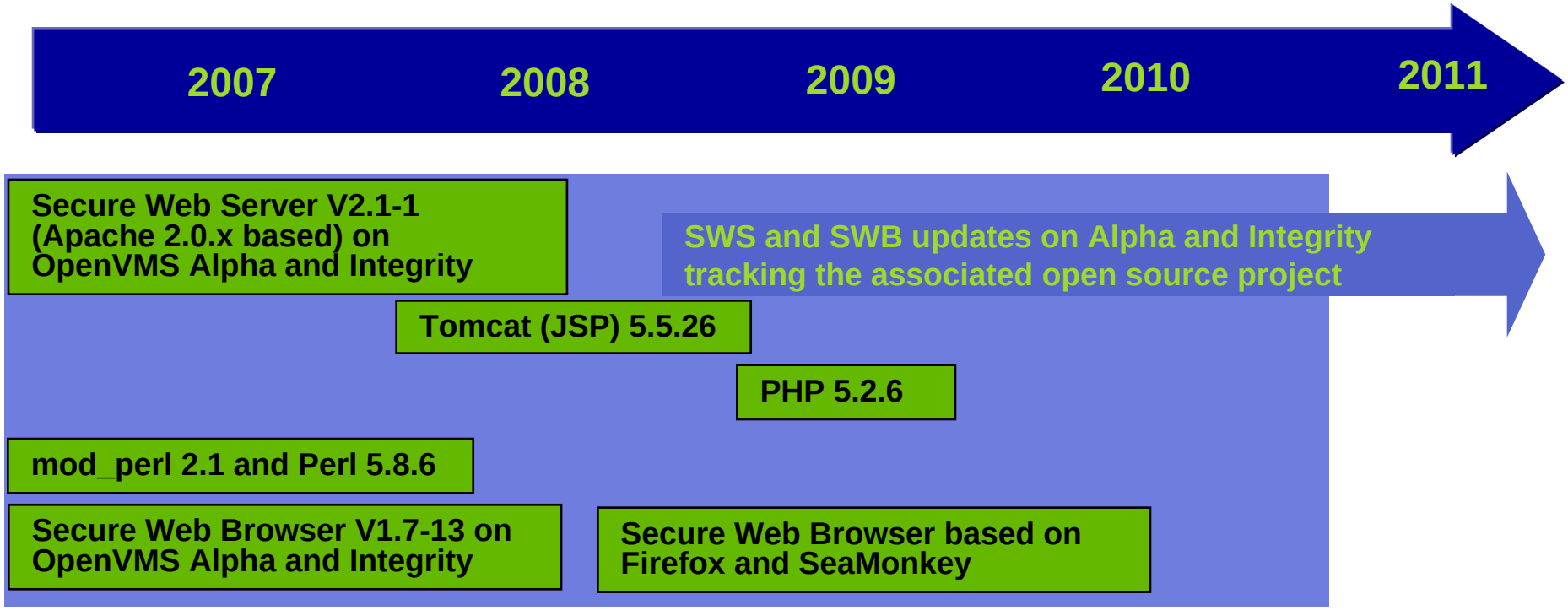
Secure Web Server

- All on Alpha and Integrity
  - SWS 2.1-1 (Apache 2.0.52)
    - Fixes for Security Vulnerabilities
    - Support for Mixed case password authentication
    - Performance Improvements
  - CSWS_PHP V2.1 (PHP 5.2.6)
    - Supports extensions/modules supported in PHP1.3.
    - No new extensions/modules have been enabled in PHP2.1 though it is based on open source version 5.2.6.
  - CSWS_Perl 2.1 ECO1 patch kit (mod_Perl 2.0.1)
    - Helps in writing apache modules in Perl
  - Perl 5.8.6 ECO1 patch kit
  - CSWS_Java (Tomcat) 3.1
    - Support for Tomcat 5.5.26
    - Works with Secure Web Server Versions 1.3-1 and 2.1 and higher. It does not work with SWS V2.0

Secure Web Browser

- **Firefox Web Browser**
  - HP Firefox Web Browser V2.0-18 based on Mozilla Firefox V2.0.0.18
  - Available only on OpenVMS Integrity
- **Secure Web Browser (based on SeaMonkey)**
  - Secure Web Browser V1.1-12 based on SeaMonkey V1.1.12
  - Available on OpenVMS Integrity and Alpha
- **Secure Web Browser (based on Mozilla)**
  - V1.7-13 based on Mozilla 1.7-13
  - Available on OpenVMS Integrity and Alpha
  - Mozilla on OpenVMS retirement by Feb 2010

Web Servers and Browsers



These timelines are only an indication of the future and may be subject to change

Agenda

- eBusiness and Integration Strategy
- Application Modernization and Integration Technology - Overview
- Web Servers and Browsers
- Application Development Tools
- Open Source Tools
- Case Studies
- Q&A

Application Development Tools

- Available today:
  - Java™ Standard Edition Development Kit (JDK)
  - Distributed NetBeans
  - gSOAP
  - Axis2
  - Web Services Integration Toolkit (WSIT)
  - Simple Object Access Protocol (SOAP) Toolkit (Apache Axis)
  - XML Technology (parsers and stylesheet processors) (Apache)
  - UDDI Client Toolkit (UDDI4J)
  - GNV
  - ANT

Application Development Tools

- Java™ Standard Edition Development Kit (JDK)
- Distributed NetBeans
- gSOAP
- Axis2
- Web Services Integration Toolkit (WSIT)
- Simple Object Access Protocol (SOAP) Toolkit (Apache Axis)
- XML Technology (parsers and stylesheet processors) (Apache)
- UDDI Client Toolkit (UDDI4J)
- GNV
- ANT

Java for OpenVMS Integrity and Alpha

- Current releases
  - JAVA 6.0 (only available on Integrity)
    - JDK and JRE 6.0 available on OpenVMS Integrity
  - JAVA 5.0
    - JDK and JRE 5.0-6 available for OpenVMS Alpha
    - JDK and JRE 5.0-5 available for OpenVMS Integrity
  - JAVA 1.4.2
    - Supported on Integrity and Alpha

Application Development Tools

- Java™ Standard Edition Development Kit (JDK)
- Distributed NetBeans
- gSOAP
- Axis2
- Web Services Integration Toolkit (WSIT)
- Simple Object Access Protocol (SOAP) Toolkit (Apache Axis)
- XML Technology (parsers and stylesheet processors) (Apache)
- UDDI Client Toolkit (UDDI4J)
- GNV
- ANT

NetBeans

- NetBeans is a modular, integrated development environment (IDE) for Java and JavaBeans development.
  - Written in 100 percent pure Java™
  - Released to the open-source community by Sun Microsystems
  - Versatility, extensible architecture, and relatively ease to use.
- Distributed NetBeans for OpenVMS
  - Distributed NetBeans was developed by OpenVMS engineering
  - IDE Server for OpenVMS and Distributed NetBeans client
  - Support for desktop operating systems such as Windows, HP-UX, Linux, and Mac-OS
- Native NetBeans for OpenVMS
  - Runs on your desktop OpenVMS system.
  - Distributed NetBeans as replacement Product NetBeans for OpenVMS

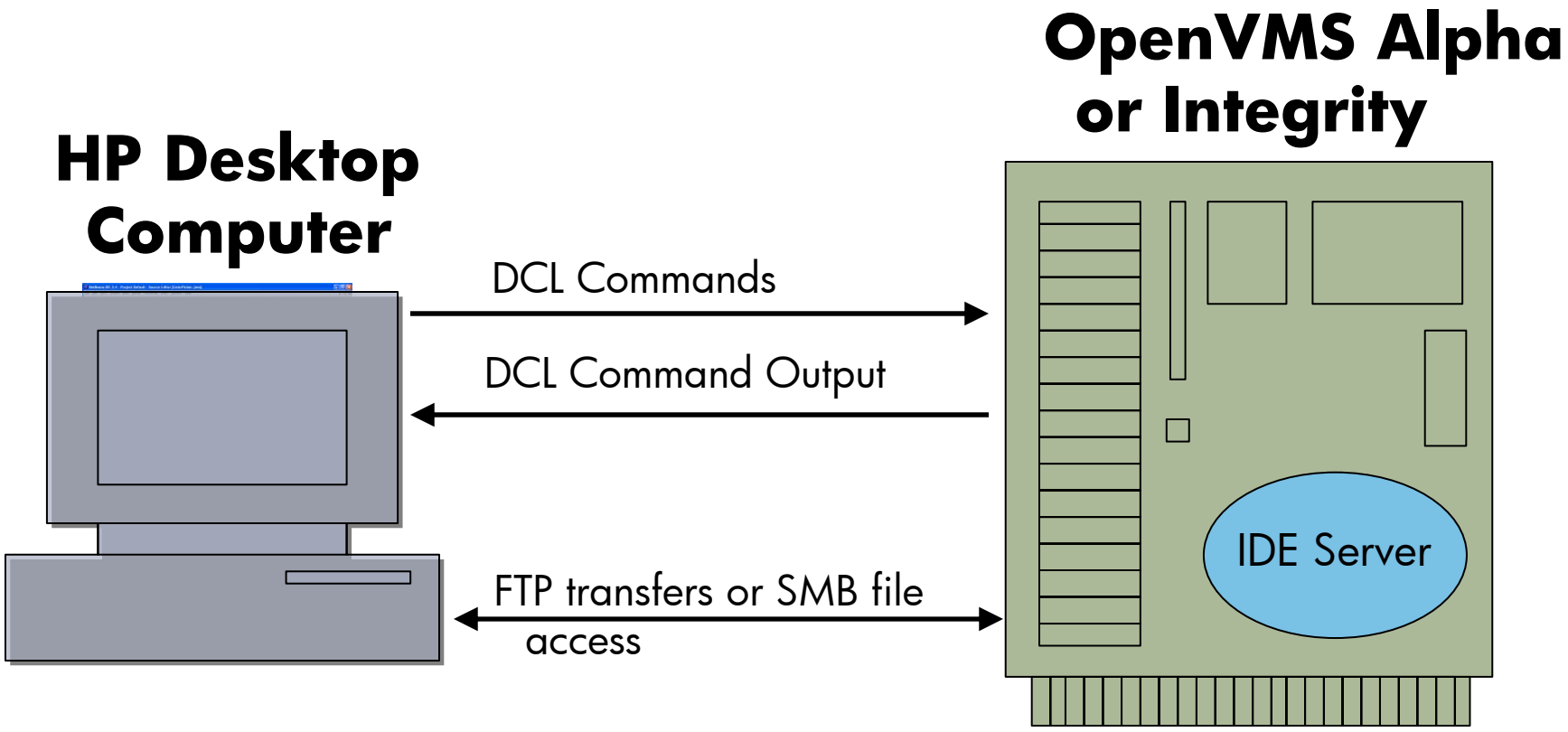
Distributed NetBeans

- Desktop (Windows, Linux,HP-UX, etc.) used for remote OpenVMS development
- Distributed NetBeans runs on the non-OpenVMS desktop
- Provides remote file access (using FTP or SMB) and operations
- Provides
  - Remote compilation and editing
  - Error navigation
  - Remote execution
  - 3GL debugging
- Also provides remote Ant operations

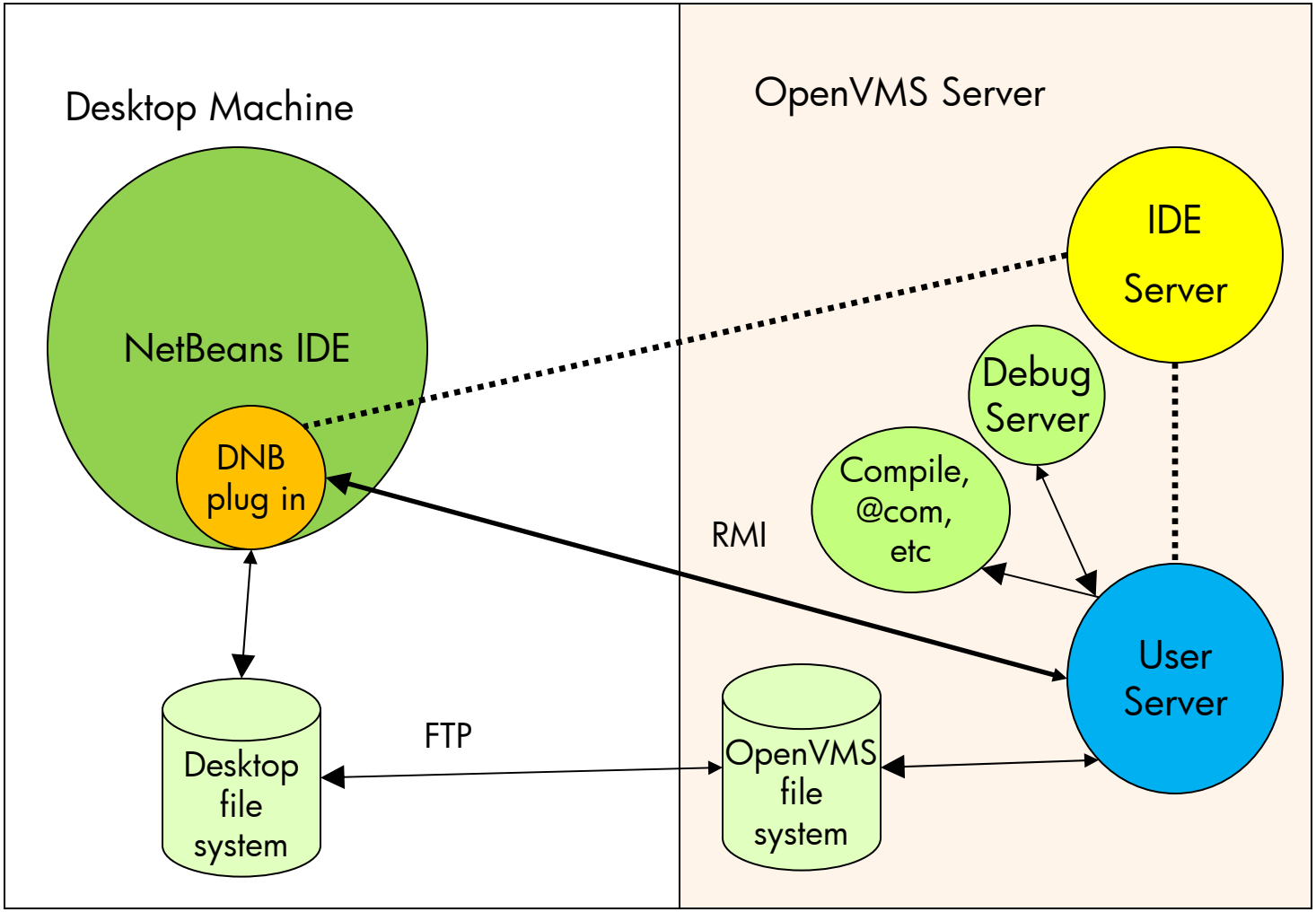
Distributed NetBeans

- Distributed NetBeans comprises two parts:
  - **Distributed NetBeans Client** :
    - plug-in for NetBeans running on your desktop.
    - Install the NetBeans IDE (from netbeans.org)
  - **IDE Server for OpenVMS** :
    - Runs on OpenVMS
    - Provides remote services for the client plug-in.

Distributed NetBeans



The Anatomy of Distributed NetBeans



Distributed NetBeans for OpenVMS

- Distributed Netbeans V5.5 – Current version
- Native NetBeans
  - OpenVMS Version 8.3 Alpha and Integrity last releases on which NetBeans 3.6 for OpenVMS is supported.
  - NetBeans 3.6 will be supported over the support life of OpenVMS 8.3.
  - Only supported on Java Platform, Standard Edition, Development Kit (JDK) v 1.4.2-x. Media Distribution

Application Development Tools

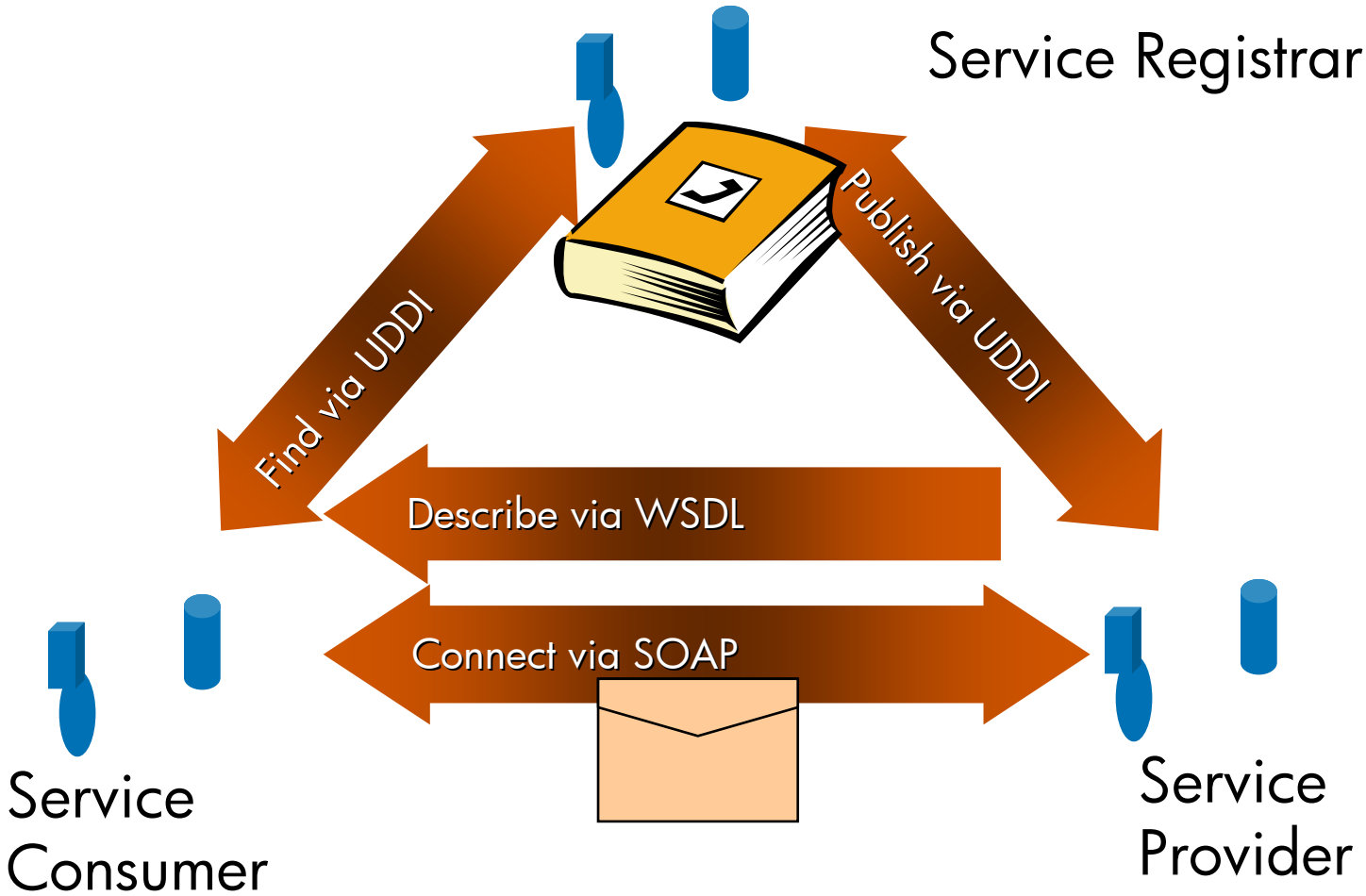
- Java™ Standard Edition Development Kit (JDK)
- Distributed NetBeans
- gSOAP
- Axis2
- Web Services Integration Toolkit (WSIT)
- Simple Object Access Protocol (SOAP) Toolkit (Apache Axis)
- XML Technology (parsers and stylesheet processors) (Apache)
- UDDI Client Toolkit (UDDI4J)
- GNV
- ANT

Web Services

- Web Services is fundamental, enabling technology for integration solutions
  - Vendor, platform, and language independent (Industry Std)
  - The way to integrate with Microsoft .NET
  - An easy way to integrate with J2EE
- Think of Web Services as "middleware for seamless integration"
- Dynamic computing environment for applications

Base Web Service Standards *

(aka First Generation Web Services)

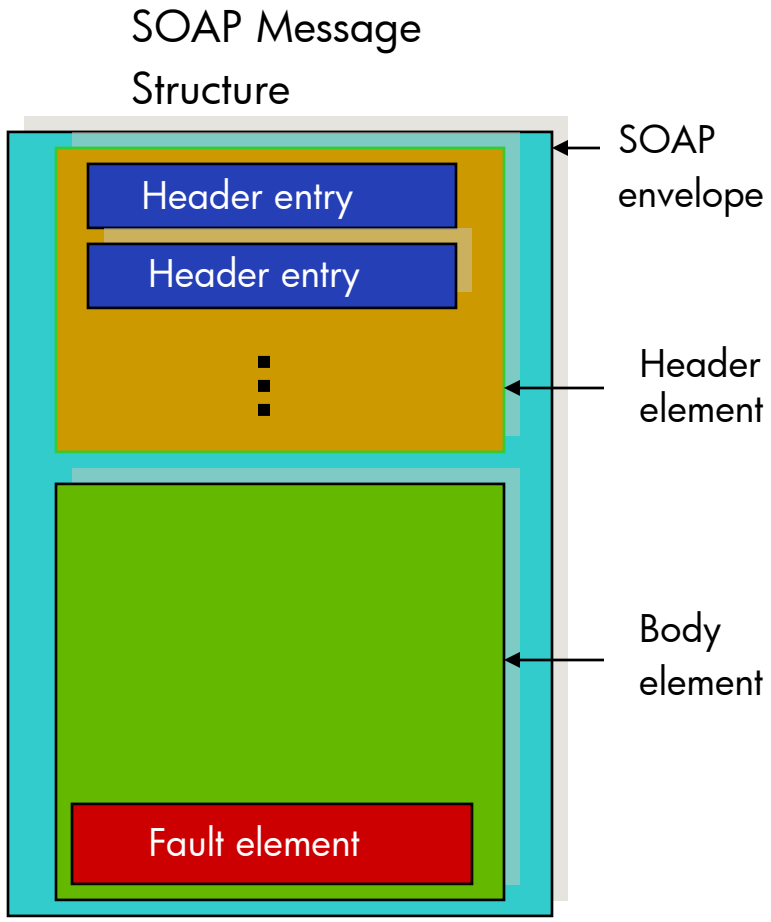


- **SOAP** **S**imple **O**bject **A**ccess **P**rotocol
- **WSDL** **W**eb **S**ervices **D**escription **L**anguage
- **UDDI** **U**niversal **D**iscovery, **D**escription, **I**ntegration

From CBDI Forum: http://roadmap.cbdiforum.com/reports/

SOAP in one slide

- Light-weight protocol based on XML as the marshalling format for data in request and response messages
  - Encoding rules for data type instances
  - Vendor and platform-neutral
  - Language-neutral
  - Object model neutral
  - Transport neutral
- Designed for loosely-coupled distributed computing
- XML allows data transformation (XSLT)
- XML enables long-term data persistence



What is gSOAP?

- Full-featured Open Source SOAP technology
  - See http://www.cs.fsu.edu/~engelen/soap.html
- More than 2500 registered users
- Uses a source-to-source stub and skeleton compiler to automate the integration of SOAP RPC in applications
- Suitable for high-performance computing (very fast)
- Major components ported to OpenVMS (Alpha and IA64)
  - Extensions to simplify use from languages other than C/C++
  - ACMS support
  - RTR support

gSOAP goals

- Application-centric
  - Minimize legacy application code adaptation
  - Support (de)marshalling of application's native data structures in SOAP/XML
  - Preserve the logical structure of data
- Minimize data migration overhead and formatting errors
  - Avoid (hand-written) wrappers
  - Generate fast (de)marshalling routines and streaming XML parsers
  - Efficient run-time remote object allocation

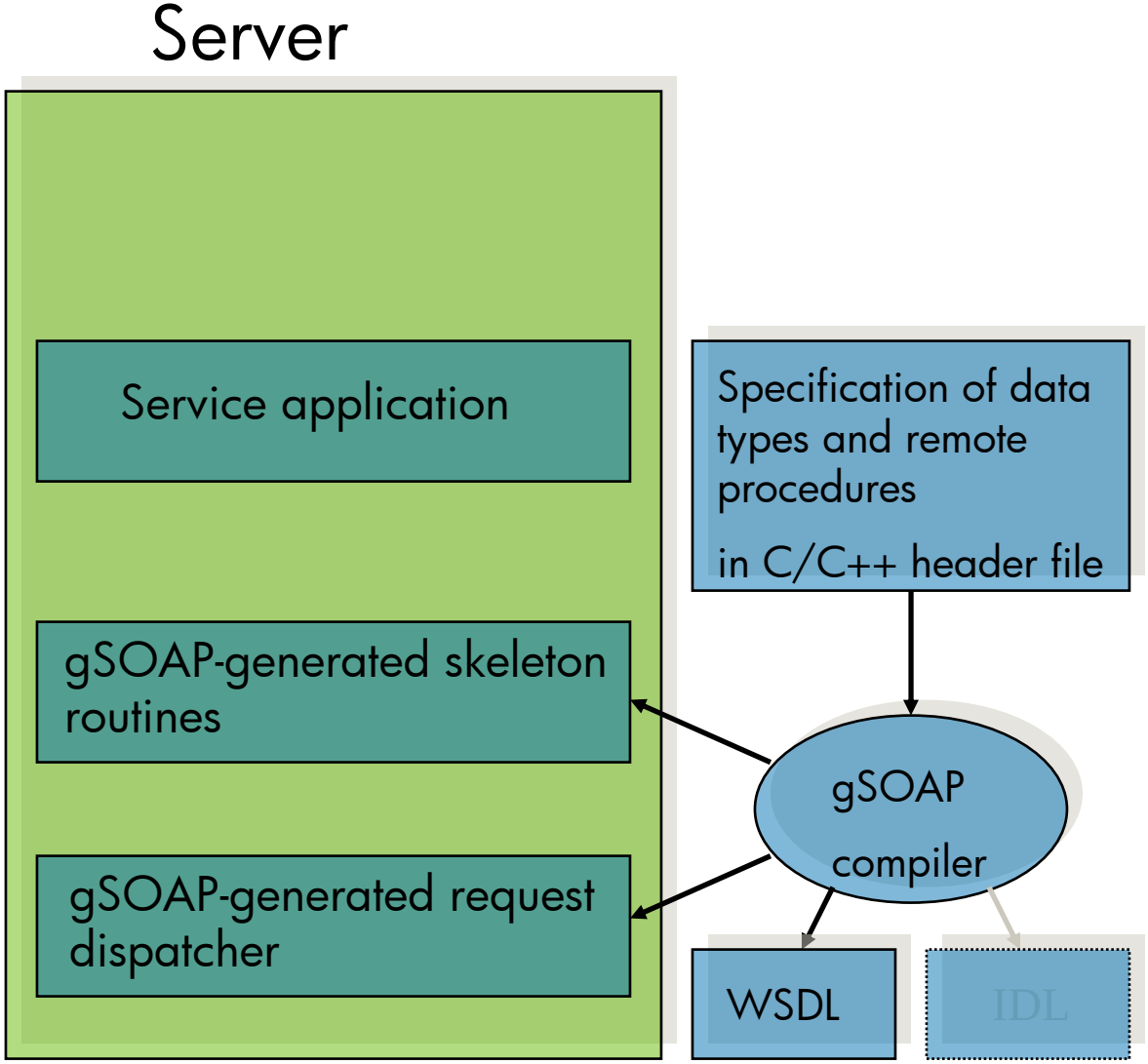
gSOAP tools

- Stub/skeleton compiler (`soapcpp2`)
  - Generates source code stubs and skeletons for SOAP RPC
  - Generates XML (de)marshalling routines for native and user-defined C/C++ data types
- WSDL/schema parser (`wsdl2h`)
  - Imports one or more WSDL files and XSDs to generate a C/C++ header file that defines the service prototypes and data types
    - The C/C++ header file would then be used as input to `soapcpp2`
- gSOAP runtime
  - Provides low-level HTTP, TCP, SOAP/XML handling, and memory management capabilities

gSOAP development

- Two basic approaches to development…
  - Start with a WSDL (*top down*)
    - Approach typically used when wanting to call an existing Web service
    - Use `wsdl2h` to convert WSDL to C/C++ header file
    - Use `soapcpp2` to generate stubs and skeletons
    - Develop client application
    - Link client application with generated code and gSOAP runtime
  - Start *without* a WSDL (*bottom up*)
    - Approach might typically be used to expose existing (legacy) functionality as Web services
    - Create a C/C++ header file containing the necessary data type and service (function prototype) definitions
    - Use `soapcpp2` to generate stubs and skeletons
    - Link generated code and gSOAP runtime with existing (or new) application code

gSOAP development

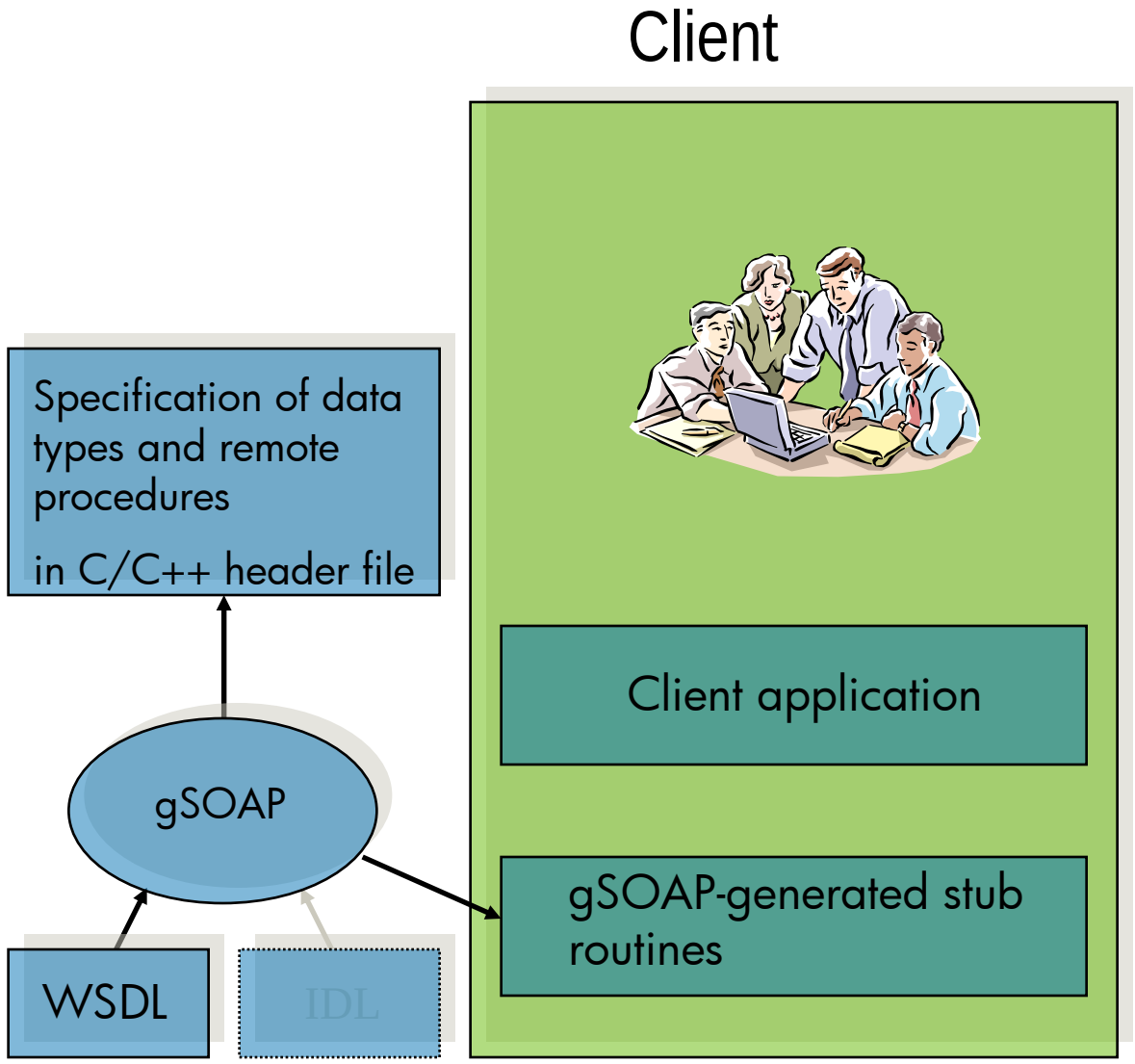


Supplied header file is processed by soapcpp2 to generate stub and skeleton routines that are linked with user-written application code. Note that soapcpp2 can optionally generate WSDL.

Adapted from http://www.cs.fsu.edu/~engelen/soaptalk.ppt

gSOAP development

An existing WSDL can be used to develop a gSOAP client (or server) application. The `wsdl2h` tool is used to convert the WSDL into a header file, which can be processed by `soapcpp2` to generate stub and skeleton routines that are linked with user-written application code.



Adapted from http://www.cs.fsu.edu/~engelen/soaptalk.ppt

OpenVMS port

- Major components ported to OpenVMS Alpha and IA64
  - `soapcpp2.exe`
    - Stub and skeleton compiler
    - Generates proxies (and RPC stubs)
    - Generates the C/C++ Web service skeletons
    - Can optionally generate WSDL and XSDs
  - `wsdl2h.exe`
    - WSDL parser
    - Converts WSDL into gSOAP header file specifications of Web services
  - Object libraries (runtime libraries)
    - Provide a transport layer with an HTTP stack on top of TCP/IP

OpenVMS port - extensions

- Additional API functions to facilitate COBOL (and languages other than C/C++)
  - GSOAP$TO_CSTRING
  - GSOAP$INIT
  - GSOAP$DESTROY
  - GSOAP$END
  - GSOAP$DONE
  - GSOAP$CHECK_ERROR
  - GSOAP$PRINT_FAULT
  - GSOAP$SET_PROXY
  - GSOAP$SET_AUTH
  - GSOAP$TCPIP_SERVER (multi-threaded server)
  - More to come…
- ACMS gateway
  - Multi-threaded ACMS agent
    - GSOAP$ACMS_AGENT
    - GSOAP$ACMS_CALL
- Apache support via mod_gsoap
- Automated wrapping of ACMS applications
  - Tool to generate gSOAP interface from STDL file
  - Still some work to be done on this…
- gSOAP and RTR
  - Essentially provides SOAP over RTR
    - GSOAP$RTR_SERVER
    - GSOAP$RTR_CLIENT_INIT
    - GSOAP$RTR_CLIENT_DONE
  - Approximates WS-ReliableMessaging
  - Very fast, very scalable…
  - Facilitates development of RTR applications using gSOAP development model

gSOAP Summary

- High-performance Open Source Web services engine
- Simple and flexible development model
- Can be used on OpenVMS to implement Web services
  - Integrates well with "legacy" 3GL code (language integration issues aside)
  - No need for Java, ODS5
  - Can be used in conjunction with other technologies such as WSIT and Apache to provide excellent levels of:
    - Scalability
    - Fault tolerance
    - Performance
- Can be used on OpenVMS to call Web services from "legacy" 3GL code
- Flexibility
  - Can be readily adapted to work with other technologies
    - WSIT
    - RTR
    - ACMS
    - …
- Availability
  - Contact Brett Cameron (brett.cameron@hp.com) or John Apps (john.apps@hp.com)

AXIS2

- Axis2 for OpenVMS replaces the SOAP Toolkit for OpenVMS
- Apache Axis2 is the core engine for web services. It is a complete re-design and re-write of the widely used Apache Axis SOAP stack.
- Port of Apache Axis2 1.3
- Runs on OpenVMS Integrity server Version 8.2 and higher, and OpenVMS Alpha Version 7.3-2 and higher.

Web Services and Integration (WSIT)

- Wraps an OpenVMS 3GL application as a JavaBean object as part of implementing an integration solution.
- Designed to call non-Java (e.g. C, Cobol, BASIC, Fortran) applications from Java applications.
- Using WSIT, a developer can wrap older application libraries and expose them as Java Classes. Rewriting older non-Java applications is difficult, time consuming and expensive.
- Easy to use tools
  - 100% OpenVMS-based
  - Leverage open source technology and standards including XML, Java, and Apache Velocity

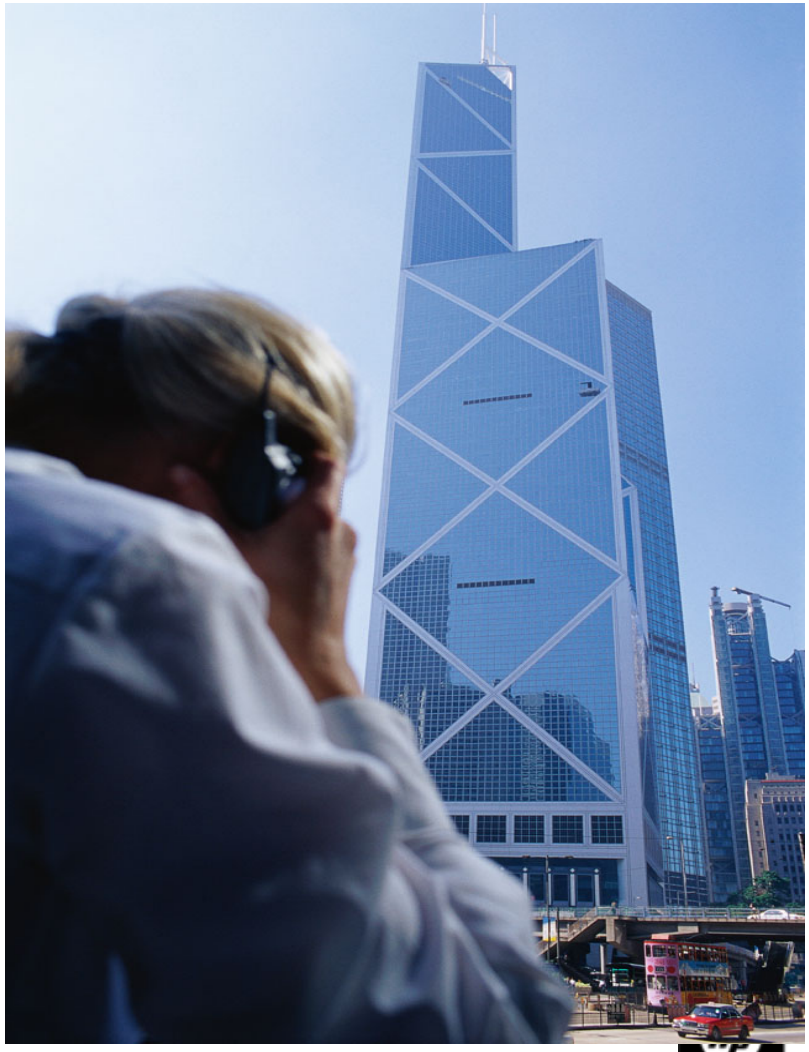
Web Services and Integration (WSIT)

- WSIT V3.0 available on OpenVMS Integrity and Alpha
  - Supports Integrity and Alpha
  - Supports C, BASIC, COBOL, FORTRAN, ACMS
  - New application tracing feature is supported
    - WSI$APPTRACING
  - Log file generation for process applications to print to a log file
    - WSI$LOGFILE
  - Documentation for debugging out-of-process features
  - Automatic stack expansion for single threaded applications
  - Support for the generation of sample AXIS2 web services
    - Not intended for production environments
  - Option to install the WSIT runtime only
  - Generated web services have login and logout methods
    - Authentication is specified using the –l switch on the generator tool (idl2code.jar)

WSIT - development steps

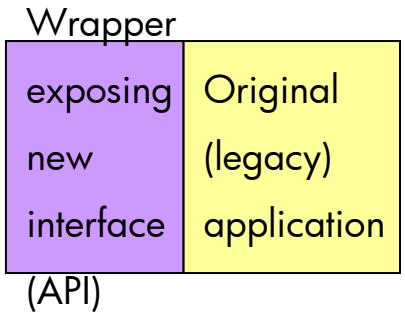
- Developing with WSIT is made up of the following steps:
  - Preparing the existing application
  - Describing the existing application to WSIT
  - Generating the new application
  - Client development

The primary goal of WSIT is to expose existing 3GL code as objects (Java beans) that can then be used as part of implementing a complete integration solution



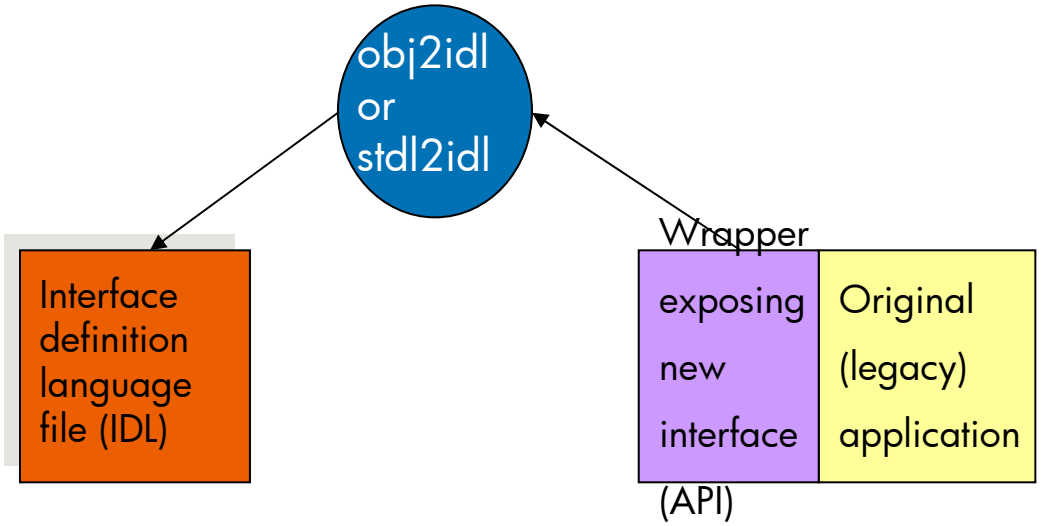
Prepare Application

- Prepare the application
  - Identify business logic that you want to reuse and expose
  - Ensure that the code be linked as a shareable image
  - Write a wrapper to cleanly expose that business logic
    - This will become the application's new interface



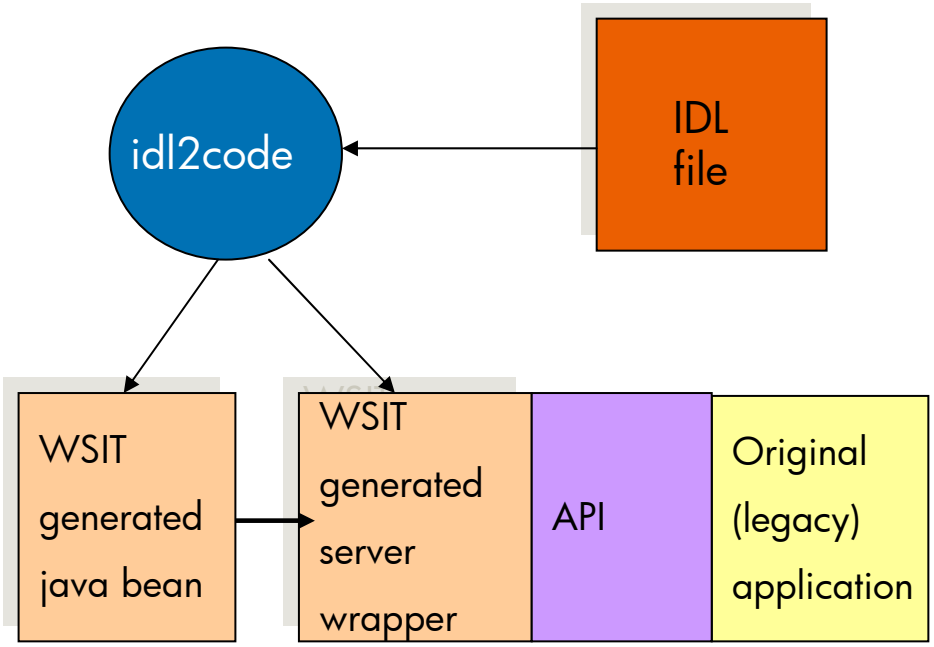
Define Application

- Define the application to WSIT
  - Using the tools obj2idl and stdl2idl, create a WSIT IDL file describing the application's interface…



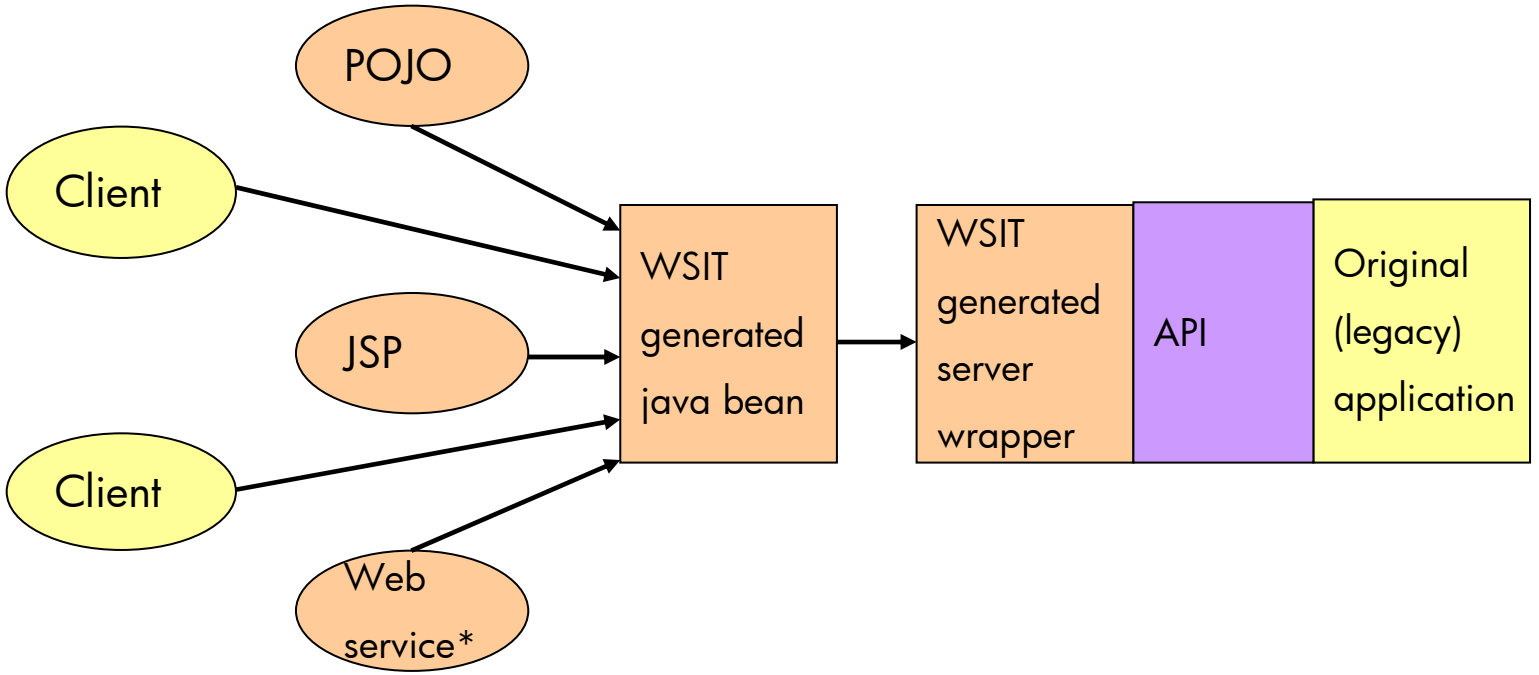
Generate New application

- Generate the new application
  - Using the tool idl2code, generate the WSIT wrapper code



Develop clients

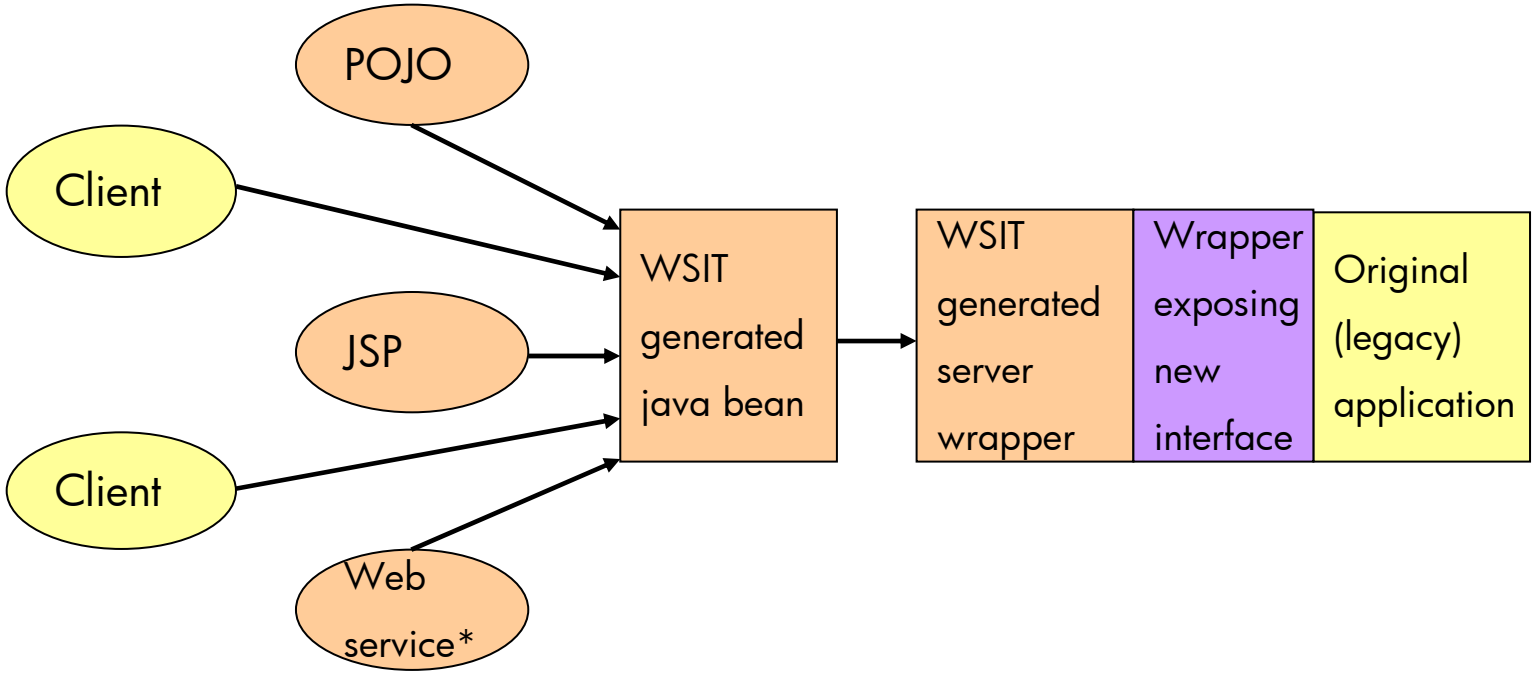
- Develop clients for the new application
  - Use the clients that WSIT optionally creates as a starter
  - Write your own using WS, Java, J2EE, JSPs, Servlets, and so on



**Sample Web Server client is new for WSIT V2.0*

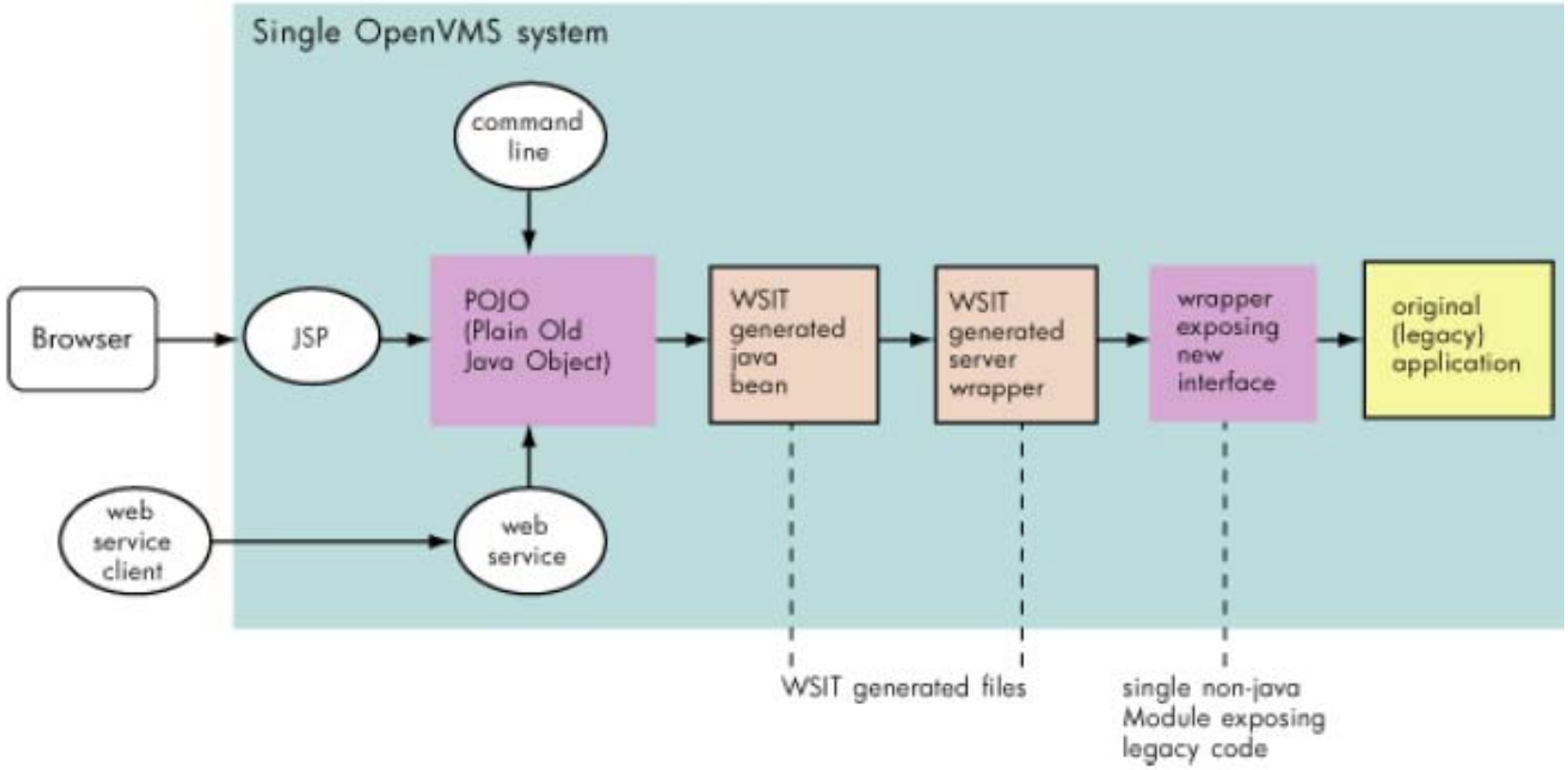
Develop Client - cont

- Developing with WSIT is intended to be quick and simple
  - You worry about your application, WSIT does the rest
  - New interface opens the door to new technologies



*Sample Web Server client is new for WSIT V2.0

WSIT – In a Nutshell



WSIT – Further Readings

- Further information on usage can be found in OpenVMS Technical Journal article by David Sullivan

  http://h71000.www7.hp.com/openvms/journal/v7/reusing_openvms_applications_from_java.pdf

XML Technology

- XML Technology based on Apache Xerces and Xalan.
- XML Parsers:
  - Based on Apache Xerces
  - XML Java Technology Version 2.0 for OpenVMS Alpha and OpenVMS I64 is based on Apache Xerces-Java Version 2.3.0 and Apache Xalan-Java Version2.4.1
  - XML C++ Technology Version 3.0 for OpenVMS Alpha and OpenVMS I64 is based on Apache Xerces C Version 2.7.0 and Apache Xalan C Version 1.10.
  - Support for Features like:
    - the Xerces Native Interface (XNI)
    - A complete framework for building parser components and configurations

XSLT

- XSLT Stylesheet Processor
  - Based on Apache Xalan
  - XSLT processor for transforming XML documents into HTML, text, or other XML document types.
  - Implements
    - XSL Transformations (XSLT) Version 1.0
    - XML Path Language (XPath) Version 1.0
  - Can be used from :
    - The command line
    - In an applet or a servlet,
    - As a module in other program.

SOAP Toolkit

- SOAP Toolkit Version 2.0 for OpenVMS (based on Apache Axis 1.1)
- Apache Axis is XML based protocol that consists of three parts:
  - An envelope that defines a framework for describing the contents of a message and how to process it
  - A set of encoding rules for expressing application-defined datatypes
  - A convention for representing remote procedure calls and responses
- The SOAP Toolkit :
  - Java-based
  - Provides development tools to create SOAP clients or to implement server-side SOAP accessible services using HTTP as the transport protocol.
  - Provides the ability to invoke SOAP RPC services available elsewhere, in addition to features for sending and receiving SOAP messages.
  - Mechanism to write new RPC or message accessible services
- AXIS2 is the next gen

UDDI Client Toolkit

- UDDI- Universal Description Discovery and Integration is service discovery protocol for Web Services.
- UDDI is the building block which enables businesses:
  - Quickly, easily and dynamically discover each other
  - Define how they interact over the Internet
  - Share information in a global registry architecture.
- The UDDI Client Toolkit:
  - Based on the UDDI4J open source implementation.
  - Provides a Java class library that provides an API to interact with UDDI registry.

Application Development Tools

- Java™ Standard Edition Development Kit (JDK)
- Distributed NetBeans
- gSOAP
- Axis2
- Web Services Integration Toolkit (WSIT)
- Simple Object Access Protocol (SOAP) Toolkit (Apache Axis)
- XML Technology (parsers and stylesheet processors) (Apache)
- UDDI Client Toolkit (UDDI4J)
- GNV
- ANT

GNV

- GNV – **G**NU is **N**ot **V**MS
- Delivers Unix tools and utilities
- Implements Unix BASH shell
- Provides many typical Unix tools and utilities for:
  - General purpose
  - Command manipulation
  - Program creation
  - File manipulation
  - Text processing
  - Printing
  - Networking
- Current version – 2.1
- Future – 2.1-3 in works

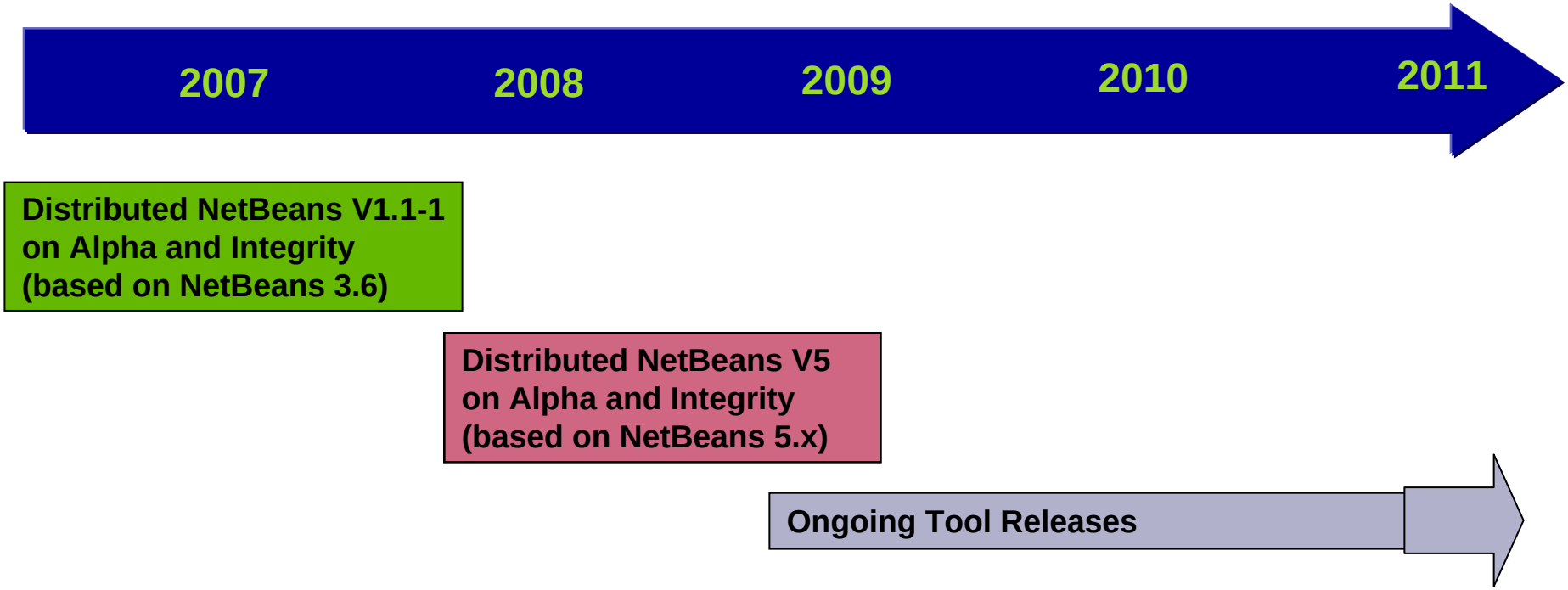
Application Development Tools

- Java™ Standard Edition Development Kit (JDK)
- Distributed NetBeans
- gSOAP
- Axis2
- Web Services Integration Toolkit (WSIT)
- Simple Object Access Protocol (SOAP) Toolkit (Apache Axis)
- XML Technology (parsers and stylesheet processors) (Apache)
- UDDI Client Toolkit (UDDI4J)
- GNV
- ANT

ANT

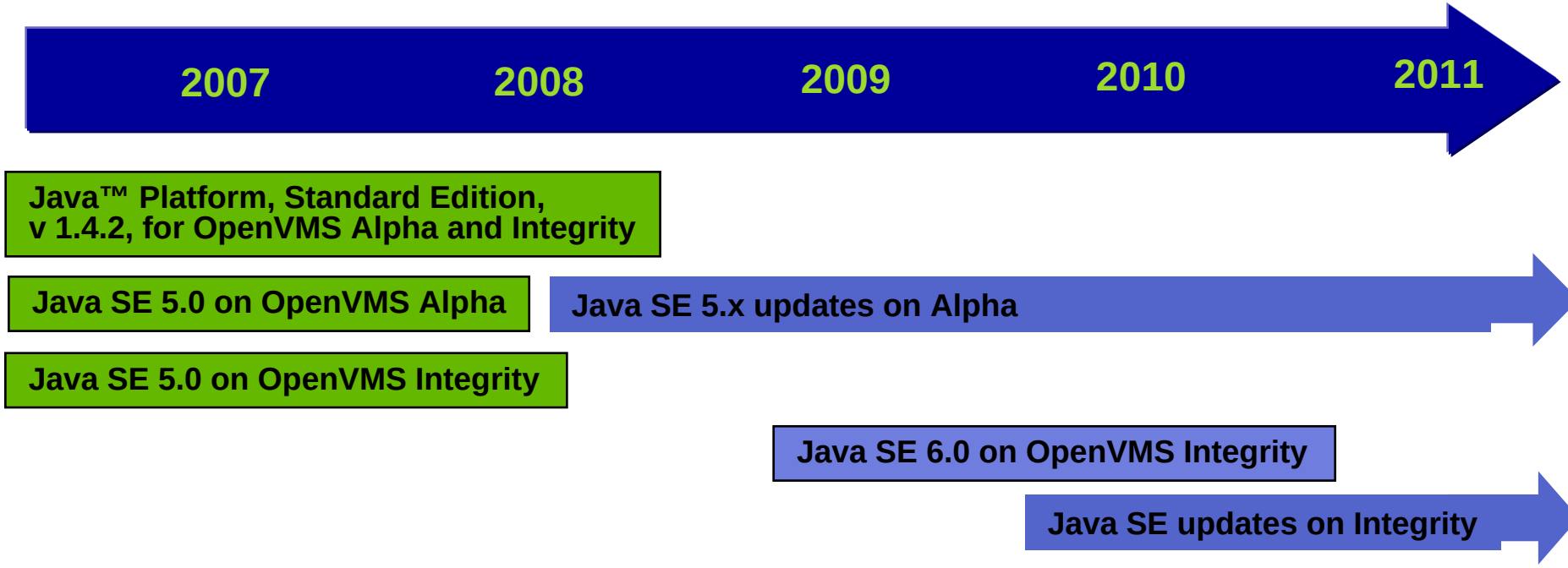
- ANT v1.7
- Runs on OpenVMS Integrity server Version 8.2 and higher, and OpenVMS Alpha Version 8.2 and higher.
- Java-based build tool, similar to “make”.
- Port of Apache Ant 1.7 to the OpenVMS environment.

Application Development and Deployment – Tools



These timelines are only an indication of the future and may be subject to change

Application Development and Deployment - Tools…



These timelines are only an indication of the future and may be subject to change

For more information

E-Business Technology Web Site

http://h71000.www7.hp.com/ebusiness/technology.html

Agenda

- eBusiness and Integration Strategy
- Application Modernization and Integration Technology - Overview
- Web Servers and Browsers
- Application Development Tools
- Open Source Tools
- Case Studies
- Q&A

Open Source Tools

- **Open Source Tools Provided by HP**
  - Freeware CD
  - GTK+
  - libIDL
  - Majordomo
  - Stunnel
- **Open Source Tools Provided by our Contributors**
  - CERN HTTPD server
  - ht://Dig
  - MySQL
  - OSU HTTP server
  - Python
  - SWISH-E query interface
  - WASD VMS Hypertext Services
  - Webware for Python

Open Source Tools

- **Source Code Kits**
  - CD Record –v1.1
  - CDSA –v2.3
  - Kerberos – v3.1 – Based on Kerberos V5 Release 1.4.1
  - SSL – v1.3 – based on OpenSSL 0.9.7e
- **Other Tools** (Contact John Apps or Brett Cameron)
  - FastCGI
  - zeroMQ
  - OpenAMQ
  - Memcached
  - Gearman
  - Tokyo Cabinet
  - Libevent
  - Lua scripting Language

Agenda

- eBusiness and Integration Strategy
- Application Modernization and Integration Technology - Overview
- Web Servers and Browsers
- Application Development Tools
- Open Source Tools
- Case Studies
- Q&A

A couple of project examples

- Community College in the US
- A New Zealand Government Department

Community College in the US – project goals

- Goals of the project many fold:
  - Migration of applications from Alpha to Integrity
  - Discovery and exposure of functionality in the applications as services for processing by any software capable of invoking Web Services or .NET methods
  - Creation of RSS feeds providing campus, curriculum and other data
  - Integration of data and functionality across multiple platforms including Windows, SAP, OpenVMS, Oracle 10g, Oracle Rdb, Microsoft SQL Server

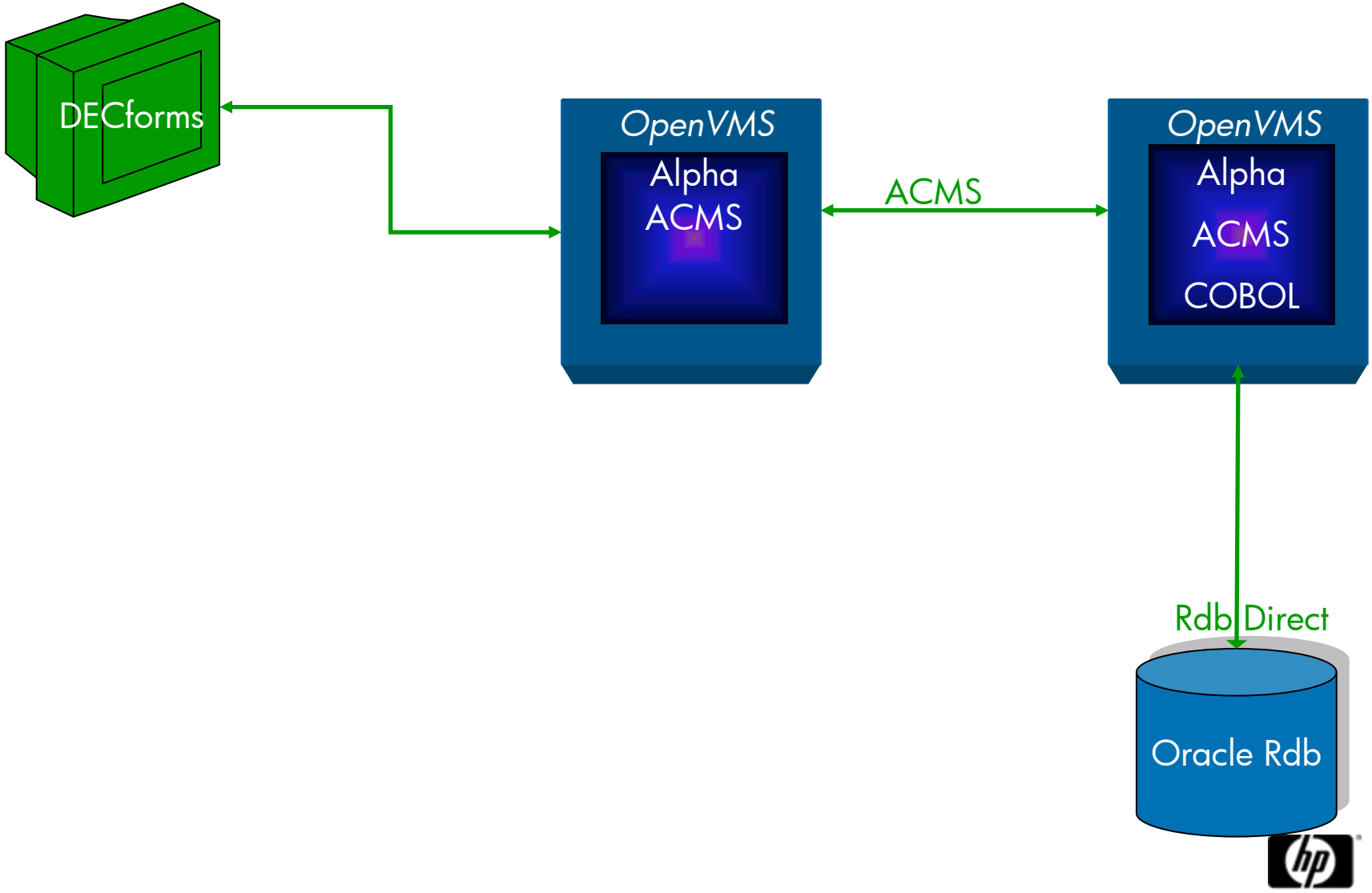
Community College in the US – end state

Front-end

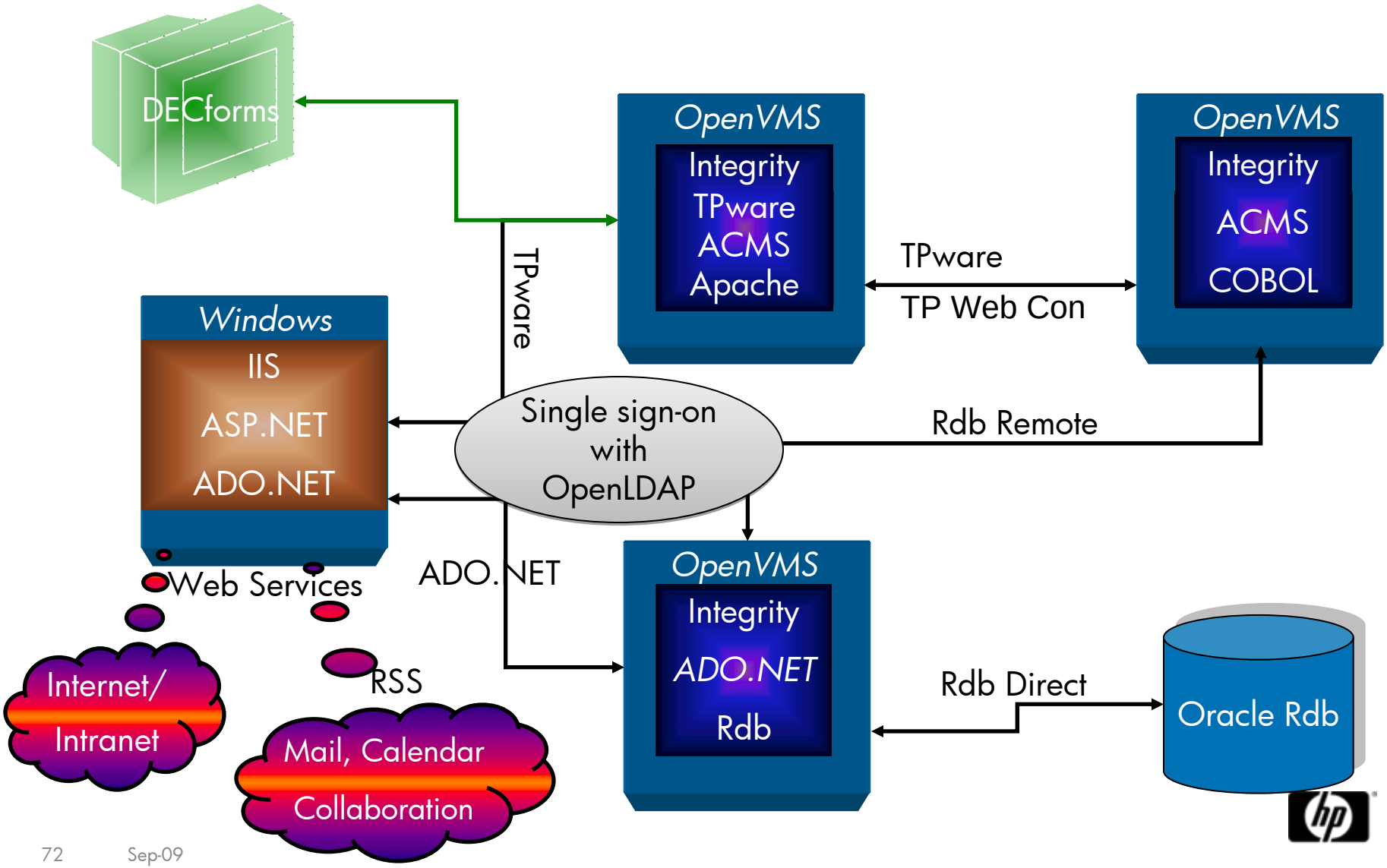
- Web browser-based user interface (replace/supplement existing DECforms)
- Web Services for WS-enabled applications
  - Integration with SAP, college applications, and so on
- Really Simple Syndication (RSS) feeds

- Middle tier
  - .NET Web pages, .NET Web Services, RSS feeds
  - Rdb .NET driver to access the Oracle Rdb databases for some functions
  - Windows 2003 Server middle tier with HP TPware to facilitate communication with back-end ACMS application components
- Back-end
  - OpenVMS with ACMS, COBOL and Oracle Rdb
    - OpenVMS applications migrated from Alpha to Integrity
  - Current user population of > 2,000
- Database tier
  - Oracle Rdb on OpenVMS Integrity servers (migrated from OpenVMS Alpha)

What are our customers doing?



What are our customers doing?



A New Zealand Government Department

- Incremental modernization (and enhancement)
- Initial application suite developed by DEC in 1988 and still maintained and supported by HP
  - Rally, COBOL, C, RDB
- Migration from VAX to Alpha (1997)
- Original Rally "green screens" replaced with Visual Basic GUI front end (1997)
  - Custom-written HTTP-based RPC client-server interface
- FreeTDS used to access SQL Server from OpenVMS environment
- Further introduction of Web technologies (ongoing)
  - Looking to introduce gSOAP for Web services
    - Have developed a facility to generate Web services interface layer from existing RPC interface definition (no changes to application code)
  - Replace existing middleware solution with Web services
  - Replace Visual Basic screens with .NET-based user interface components
  - Web service-based integration with other systems

Contacts

Rohini Madhavan
OpenVMS e-business Product Manager
Rohini.Madhavan@hp.com

Q & A