

Session: 2004 St. Louis User Group



# iSeries Access Programming Overview



**Bleeker**  
bleek@us.ibm.com

© Copyright IBM Corporation, 2004. All Rights Reserved.  
This publication may refer to products that are not currently available in your country. IBM makes no commitment to make available any products referred to herein.

*IBM @server iSeries*

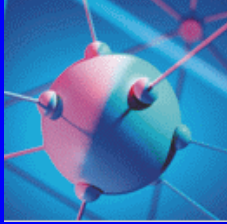
**TITLE: iSeries Access Programming Overview**  
**SESSION ID#: 404391**  
**AGENDA KEY: 33CC**

**SPEAKER 1 Troy Bleeker**

Evaluation comments are important to improving COMMON sessions

# Agenda

- Introduction
  - API
  - Strategy
  - Where to start?
- Traditional APIs
  - Industry Standards
  - iSeries Specific
  - Client Management
- Next Generation APIs
  - ActiveX automation objects and controls
  - OLE DB Provider
  - Java tools
- iSeries Navigator
  - Plug-ins
- Where to start
- What's new



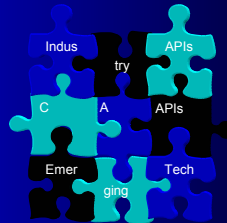
## API – a definition

- A set of method calls
- An object supporting data flows or a service from program to program
- A software development kit
  - Visual such as Visual Basic
  - Non-visual such as ODBC
- Any combination of the above

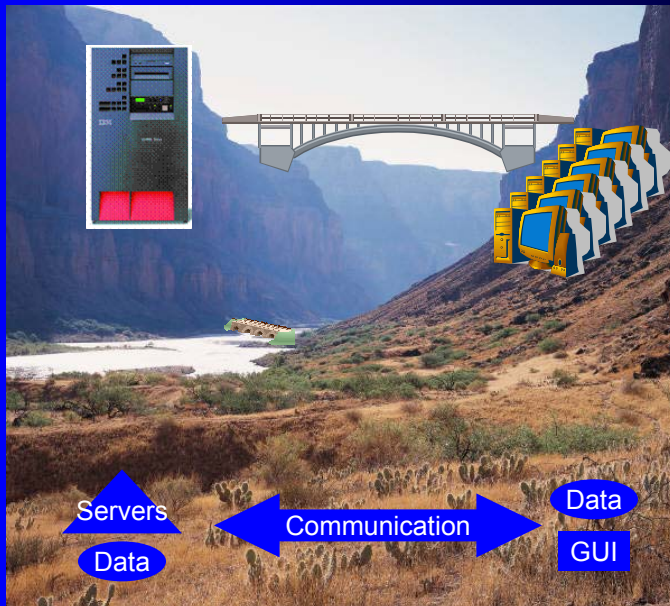
■ "The interface (calling conventions) by which an application program accesses operating system and other services. An API is defined at **source code level** and provides a **level of abstraction** between the application and the kernel (or other privileged utilities) to ensure the portability of the code. An API can also provide an interface between a high level language and lower level utilities and services which were written without consideration for the calling conventions supported by compiled languages. In this case, the API's main task may be the translation of parameter lists from one format to another and the interpretation of call-by-value and call-by-reference arguments in one or both directions." From Denis Howe's FOLDOC.

# iSeries Access Strategy

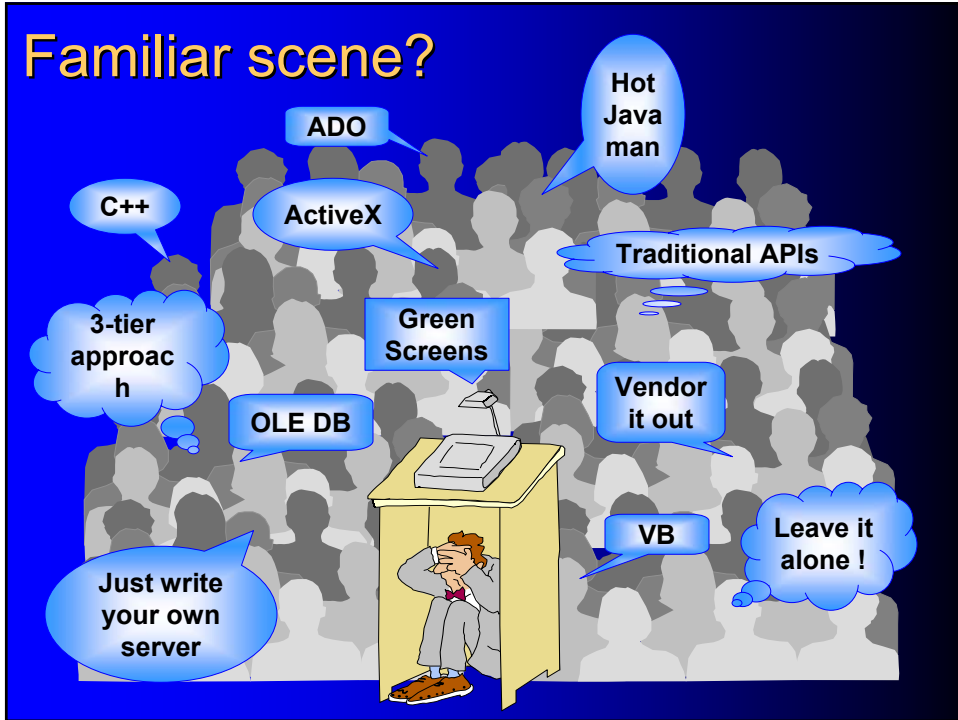
- Support industry APIs where it makes sense
- Supplement with iSeries Access specific APIs
  - ❖ Provide consistent interfaces
  - ❖ Create a complete set of client/server APIs
  - ❖ Enable client management
- Support emerging technologies
  - ❖ Java
  - ❖ ActiveX
  - ❖ OLE DB



## Where to start?



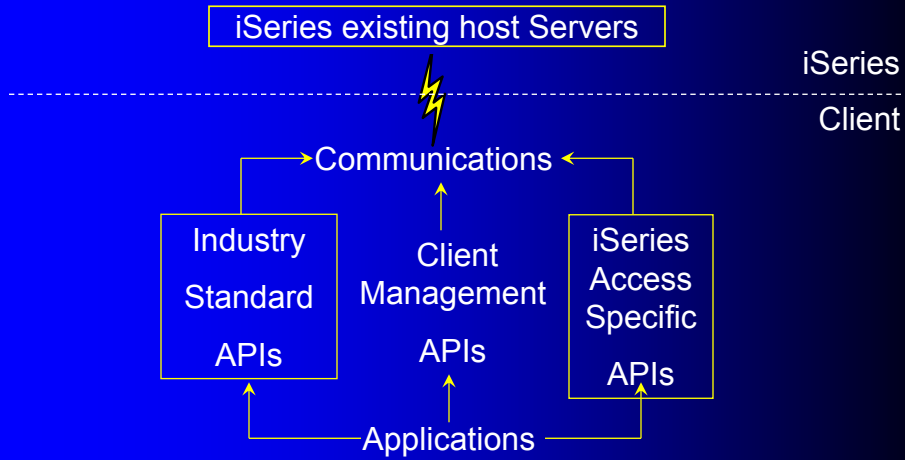
## Familiar scene?



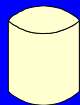
## Traditional APIs

- Industry Standards
- Client Access specific
- Client Management

# Traditional API structure



# Industry Standard APIs



## DATABASE

- ODBC
- JDBC
- OLE DB



## APPLICATIONS

- EHLLAPI
- DDE



## COMMUNICATIONS

- Sockets



FILES



PRINT

# iSeries Specific APIs



Data Queue



Remote Comand

Distributed Program Call



File Transfer



Multimedia

# Client Management



Security



Service



Update



Data Transformation

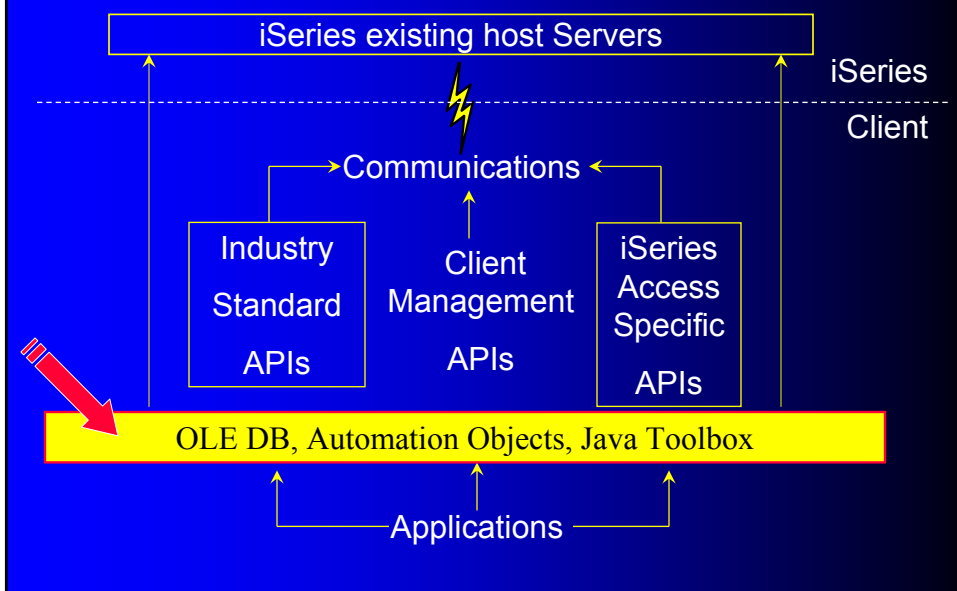


NLS

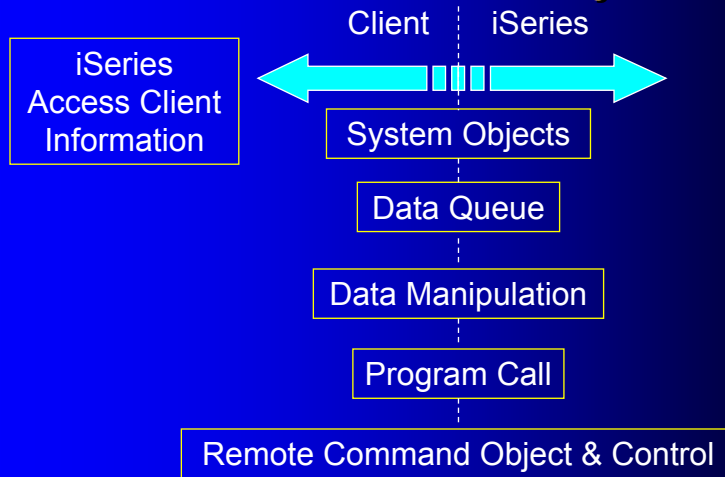
## Next Generation APIs

- ActiveX Automation objects and custom controls
- OLE DB Provider & Wizards
- IBM Toolbox for Java
- .Net
- iSeries Navigator Plug-ins

## Next Generation API structure



# ActiveX Automation Objects



## OLE DB Provider

**1** Middleware layer provides interface to iSeries data and services (runtime)

- ActiveX Data Objects (ADO)
- Can manipulate records directly, not through SQL
- Easy access to
  - ✓ Keyed record level access
  - ✓ SQL stored procedures
  - ✓ Data Queues
  - ✓ Programs
  - ✓ SQL statements
  - ✓ CL commands

**2** Programmer's Toolkit

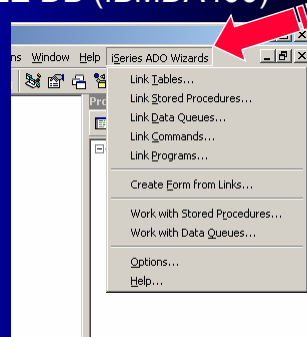
- Wizards generate code in a Visual Basic project
- Provides interfaces to manage Stored Procedures and Data Queues
- Documentation and samples



# OLE DB Tool Integration

## 1 Any OLE Compliant Tool can use OLE DB (IBMDA400)

- Visual Basic,
- Delphi,
- PowerBuilder,
- VBScript



## 2 Wizards are for Visual Basic only

## 3 Fast keyed record level access to DB2/400

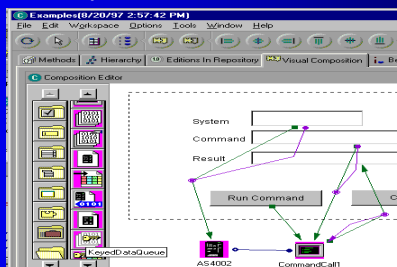
## 4 Easier to build Client/Server Applications

# Toolbox for Java

## 1 Interfaces for Java programs to access iSeries data and resources:

- ✓ Infrastructure to communicate with the iSeries
- ✓ User Space access
- ✓ User ID and password
- ✓ Command call
- ✓ Jobs, messages, objects, system values and info
- ✓ Program call
- ✓ Data queues
- ✓ Code page and data type conversion
- ✓ JDBC for SQL
- ✓ Integrated File System access
- ✓ Record-Level database access (keyed)
- ✓ Network print

## 2 Provides beans for the public interfaces



# Toolbox for Java ME



Interfaces for Java programs to access iSeries data and resources:

- ✓ Available in JTOpen 3.0
- ✓ Access to iSeries from wireless devices with a JVM
- ✓ Small footprint on device
- ✓ CMDs
- ✓ PGMs
- ✓ DQs
- ✓ JDBC

# .Net



Interfaces for .Net programs to access iSeries database:

- ✓ iSeries Access .Net managed provider    `IBM.Data.DB2.iSeries`
- ✓ Bridge to ODBC drivers                    `Microsoft.Data.Odbc`
- ✓ Bridge to OLE DB providers              `System.Data.OleDb`

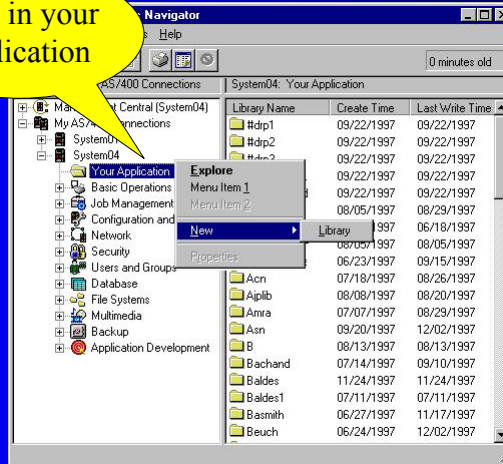


IBM creating a native provider (i.e. no bridge) for the fastest possible and most reliable data access from the .Net framework



# iSeries Navigator

Plug in your application



## iSeries Navigator Plug-ins

1 Allows you to add your own applications to iSeries Navigator

2 Allows you to be part of the iSeries user interface of choice

3 Use iSeries Access install to distribute your application

4 Applications currently using the support

✓ Domino server

✓ Java VM



Use Plugin Support to add your application to Operations Navigator

# Third Party Tools

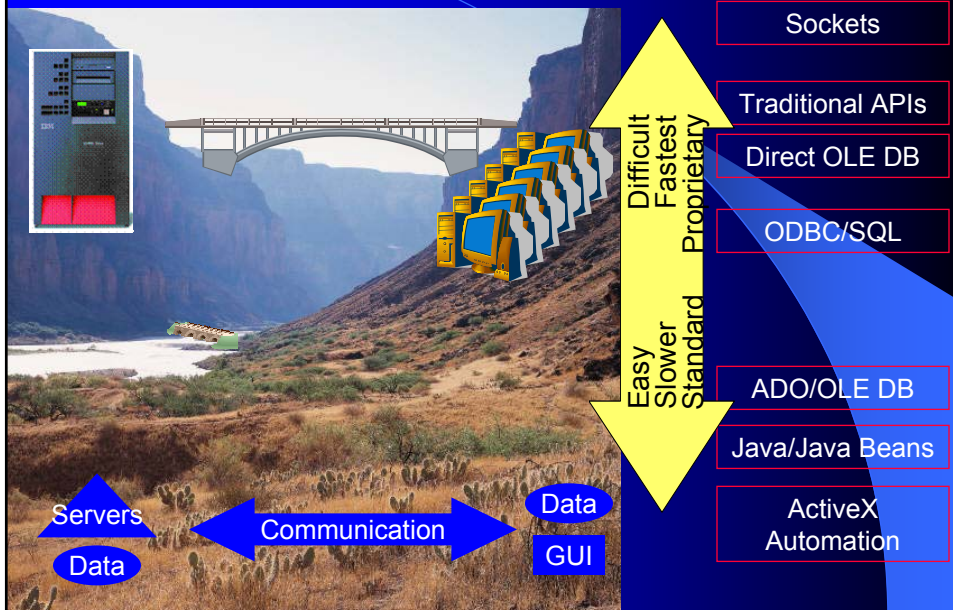
## 1 Examples

- Seagull
- Delphi/400
- ASNA
- Jacada
- HiT
- ...

## 2 Characteristics

- ✓ Often have their own server
- ✓ Often claim better performance than others
- ✓ Usually proprietary

# Where to start?



# Functions supported

	C/C++ APIs (traditional)			ActiveX		Java
	Non-IBM	IBM	ODBC	Automation Objects	OLE DB	Java Toolbox
Record level DB2/400 database access					X	X18
SQL Statements		X	X	X10	X6	X1, 18
Call Stored Procedure		X	X	X10	X	X1, 18
Call AS/400 Program (parameter I/O)		X		X6	X	X18
Call AS/400 Command		X		X3, 13	X	X18
Data Queue (FIFO, LIFO, Keyed)		X		X3, 13	X2	X18
Data Area						X
Integrated File System						X18
User Space						X
Data Manipulation (code page, data type)		X	9	X12, 19	4	X12
Network Print		X				X
Multimedia (USF)		X				
Security		X				X
Service		X				X
Directory Update		X				X
Client Access Client Information		X		X		
AS/400 Connection management		X		X		X
List of AS/400 systems		X		X13		
Jobs		X6				X18
Messages		X6				X18
Manipulate print resources		X6				X18
Object Authorities						X
System Values						X
System Status information						X
Users and groups						X
Operations Navigator Plug-In		X7				
Object Lists						X
Operations Navigator Control				X		
PCS250 Emulator control (HACL)		X15, 16		X15, 16		
EHLAPI (and PCSAPI)		X				
ENPTUI						X6
Code generation or Wizards					X17	
DDE in PCS250 Emulator	X					
File access	X14					
LDAP	X					
Sockets						X

- Using JDBC
- Reads all entries on the queue even for keyed data queues
- Automation objects require developer to transform host data
- OLE DB transforms data automatically, but complex types are not supported
- Has wrapper classes
- No commitment control
- Class implementation that can be in Java, C++, or Visual Basic
- This is the same as System Object Access
- ODBC transforms data automatically, but only SQL types are supported
- These are simple wrappers of the C/C++ APIs
- If the API supports the new System Object
- Complex data types such as structures and arrays supported
- Custom control still available
- Regular file I/O and 95/NT Network Drives to AS/400s configured with Netserver (don't need CA for this)
- This library is like EHLAPI, but has more capability
- C++ language only
- Code generating wizards for Visual Basic
- Java GUI class is available here
- Not in Lotus script

 New or changed significantly for Client Access Express

## To Use or Not to Use?

	C/C++ APIs (traditional)		ActiveX		Java
	IBM interfaces	ODBC	Automation objects	ADO / OLE DB	Java Toolbox
Performance	Best performance (if coded properly)	Best performance for SQL			
Language	Difficult to use in anything except C/C++	Most languages/tools add function to support ODBC calls	Language independent (must support COM)	Language independent (must support COM)	Must use the Java language
OS	Windows only	ODBC SQL on many platforms	Windows only	Windows only	Platform independent
Standards?	Different coding for each type of data or service access	Standard way to access relational data	IBM defined the interface	Microsoft defined the interface	IBM defined the interface (except JDBC)
Binding		Automatically binds data (translation and conversion)	Requires developer to translate and convert data (API provided)	Automatically binds data (translation and conversion)	Requires developer to translate and convert data (API provided)
Complex data		SQL data types only	Handles complex data types (call system APIs)	All but complex data types (i.e. structures)	Handles complex data types (call system APIs)
Coding model	Many API calls for one kind of data or service access	ODBC SQL common across platforms	Each data or service access has its own interface	One common object model for all data and service access	Each data or service access has its own interface
Record level access				Keyed record-level access	Keyed record-level access
Code generation tools				VB wizards to get you started in ADO	XML Framework for GUI building and program call
Connecting	Connections in iSeries Access can occur on the fly	Must configure Datasource connections	Connections in iSeries Access can occur on the fly	Connections in iSeries Access can occur on the fly	Connections can occur on the fly
Coverage	Broad functional coverage	Most mature/robust (older and highly used)			Broad functional coverage
Support					More IBM resource behind Java



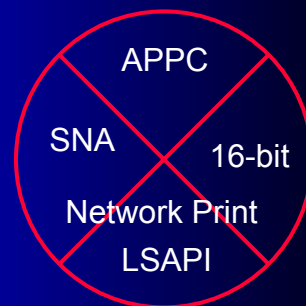
## What's New in V5R1?

- 1 System object (across all functions)
- 2 Automation objects
  - ✓ Program, Command, Data Queue, Data Transfer
- 3 Java Toolbox
  - ✓ Uses iSeries Client Access as a ship vehicle
  - ✓ Two new Java tools: GUI Builder and Resource Script Converter
- 4 Support additional data types
  - ✓ BIGINT
  - ✓ LOBs



## What's New in V5R1?

- 5 iSeries Navigator Plug-In
  - ✓ Support Java and VB for Plug-Ins
- 6 Unicode support
- 7 ODBC now supports:
  - ✓ MTS 2.0
  - ✓ Wide API set
  - ✓ Static cursor





## What's New in V5R1?

### Programmer's Toolkit – Pulling it Together

- ❖ Documentation
- ❖ Headers, type libraries
- ❖ Sample Programs

#### API and Tech Reference

- ❖ API Descriptions
- ❖ Typical Usage



#### Programmer's Toolkit

- ❖ Help center
- ❖ SDK in Toolkit
- ❖ Data Transfer ActiveX Automation
- ❖ Samples on web
- ❖ API Tech Ref online only



## What's New in V5R2?

**1** Kerberos support for any iSeries Access connection

**2** Automation objects

- ✓ Lotus 123 V9 format for Data Transfer

**3** Java Toolbox

- ✓ Wireless API's (com.ibm.as400.micro)
- ✓ Clustered Hashtables
- ✓ Record Format Markup Language (RFML)
- ✓ iSeries System Debugger
- ✓ JDBC 3.0 Support
- ✓ JTOpen 3.2 has been released
- ✓ Support for JSSE

**4** Additional data types

- ✓ ROWID



## What's New in V5R2?

- 5 64K SQL statement size
- 6 ODBC now supports meaningful information from the `SQLTablePrivileges` and `SQLColumnPrivileges`
- 7 ODBC and OLE DB now support additional base table information
- 8 OLE DB now supports updatable SQL cursors
- 9 Multiple databases on the same system (IASPs)

## References

- AS/400 Client/Server
  - <http://www.ibm.com/servers/eserver/series/access/>
- Client Access Toolkit
  - <http://www.ibm.com/servers/eserver/series/access/toolkit>
- Client Access OLE DB Provider
  - <http://www.ibm.com/servers/eserver/series/access/oledb>
- Java
  - <http://www.ibm.com/servers/eserver/series/toolbox/>
  - <http://www.software.ibm.com/ad/vajava>
- iSeries Navigator
  - <http://www.ibm.com/servers/eserver/series/navigator/>
- Online documentation
  - <http://www.ibm.com/servers/eserver/series/library/>
  - <http://as400bks.rochester.ibm.com>



## Books

- Client Access Reference Manuals
  - SC41-5507-02 Client Access/400 for Windows Setup - V5R1M0
  - SC41-3513 Client Access/400 for Windows 95 API and Technical Reference
- Client Series Redbooks
  - GG24-4027 AS/400 Client Series - Products and Positioning
  - GG24-4285 AS/400 Client Series Handbook
  - SG24-5191 AS/400 Client Access Express for Windows: Implementing V4R4M0
  - SG24-5183 A Fast Path to AS/400 Client/Server Using AS/400 OLE DB Support
- Other References
  - SC41-3554 Client Access/400 PC5250 Programmer's Guide
  - SC41-3652 Ultimedia System Facility Programming Guide
  - GG24-3070 Bibliography of International Technical Support Organization Technical Bulletins
  - Microsoft Open Database Connectivity Software Development Toolkit Version 3.0: Programmer's Reference

## Summary

- Use the correct API set that suits your business needs.
- Take advantage of the iSeries Information Center

# Trademarks & Disclaimers

© IBM Corporation 1994-2003. All rights reserved.

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

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

AS/400	IBM
AS/400e	IBM (logo)
eServer	iSeries
 server	OS/400

Lotus and SmartSuite are trademarks of Lotus Development Corporation and/or IBM Corporation in the United States, other countries, or both.

MMX, Pentium, and ProShare are trademarks or registered trademarks of Intel Corporation in the United States, other countries, or both.

Microsoft and Windows NT are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

SET and the SET Logo are trademarks owned by SET Secure Electronic Transaction LLC.

C-bus is a trademark of Corollary, Inc. in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product or service names may be trademarks or service marks of others.

Information is provided "AS IS" without warranty of any kind.

All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer.

Information in this presentation concerning non-IBM products was obtained from a supplier of these products, published announcement material, or other publicly available sources and does not constitute an endorsement of such products by IBM. Sources for non-IBM list prices and performance numbers are taken from publicly available information, including vendor announcements and vendor worldwide homepages. IBM has not tested these products and cannot confirm the accuracy of performance, capability, or any other claims related to non-IBM products. Questions on the capability of non-IBM products should be addressed to the supplier of those products.

All statements regarding IBM future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. Contact your local IBM office or IBM authorized reseller for the full text of the specific Statement of Direction.

Some information in this presentation addresses anticipated future capabilities. Such information is not intended as a definitive statement of a commitment to specific levels of performance, function or delivery schedules with respect to any future products. Such commitments are only made in IBM product announcements. The information is presented here to communicate IBM's current investment and development activities as a good faith effort to help with our customers' future planning.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance 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 or performance improvements equivalent to the ratios stated here.

Photographs shown are of engineering prototypes. Changes may be incorporated in production models.