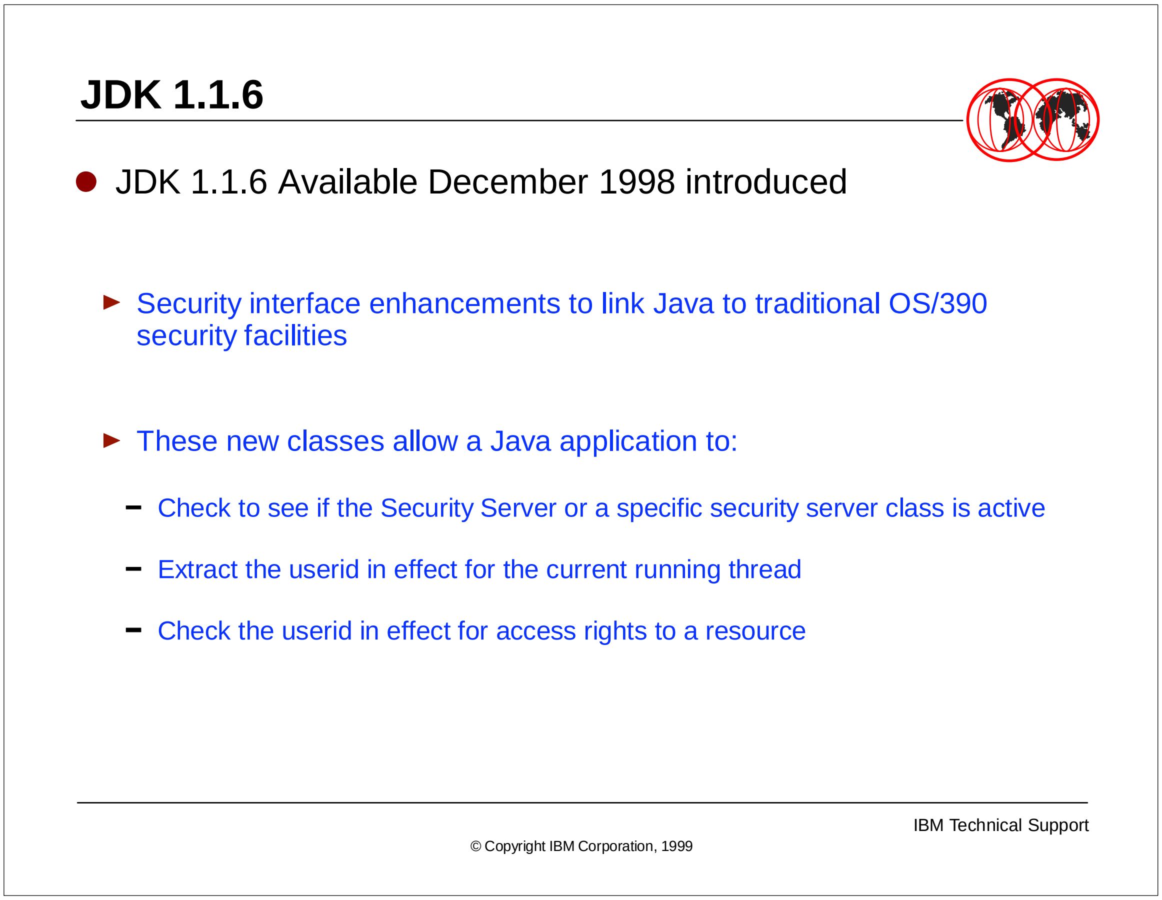
OS/390 Security Server and JAVA

Paul de Graaff ITSO Poughkeepsie S/390 Security



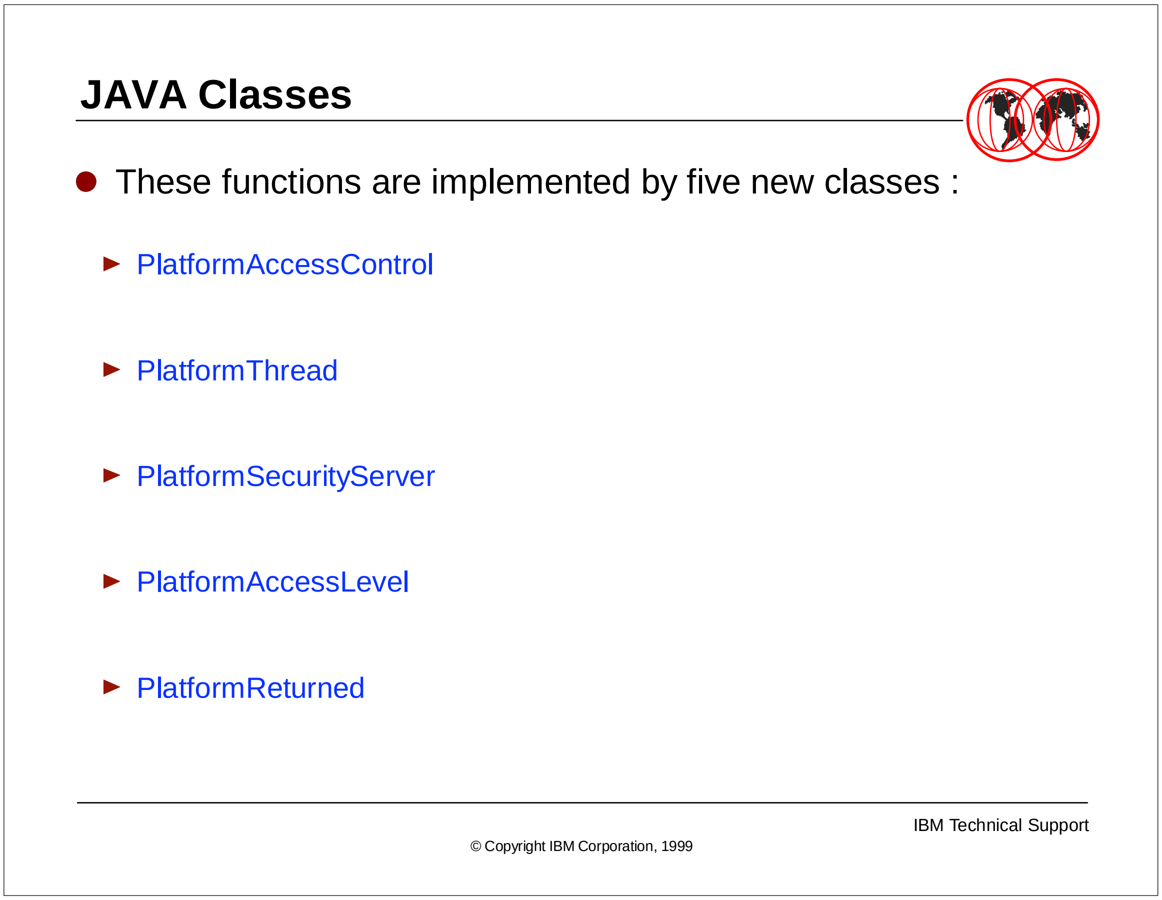
JDK 1.1.6

- JDK 1.1.6 Available December 1998 introduced
  - Security interface enhancements to link Java to traditional OS/390 security facilities
  - These new classes allow a Java application to:
    - Check to see if the Security Server or a specific security server class is active
    - Extract the userid in effect for the current running thread
    - Check the userid in effect for access rights to a resource

IBM Technical Support

- Java for OS/390 Security Services provide an additional set of security APIs. These APIs are available on Java for OS/390 at the JDK 1.1.6 level running on OS/390 Version 2 Release 4 or above. These APIs are implemented through Java classes wrapping OS/390 UNIX Services. The OS/390 UNIX Services are in turn handled by a Security Server for OS/390 that implements SAF interfaces (such as RACF).

- This initial release provides access to a basic set of existing OS/390 UNIX APIs that are required to implement principal based access control in a Java application, for example, an application that implements a Java SecurityManager class. Applications that use these APIs do not have to be APF authorized.

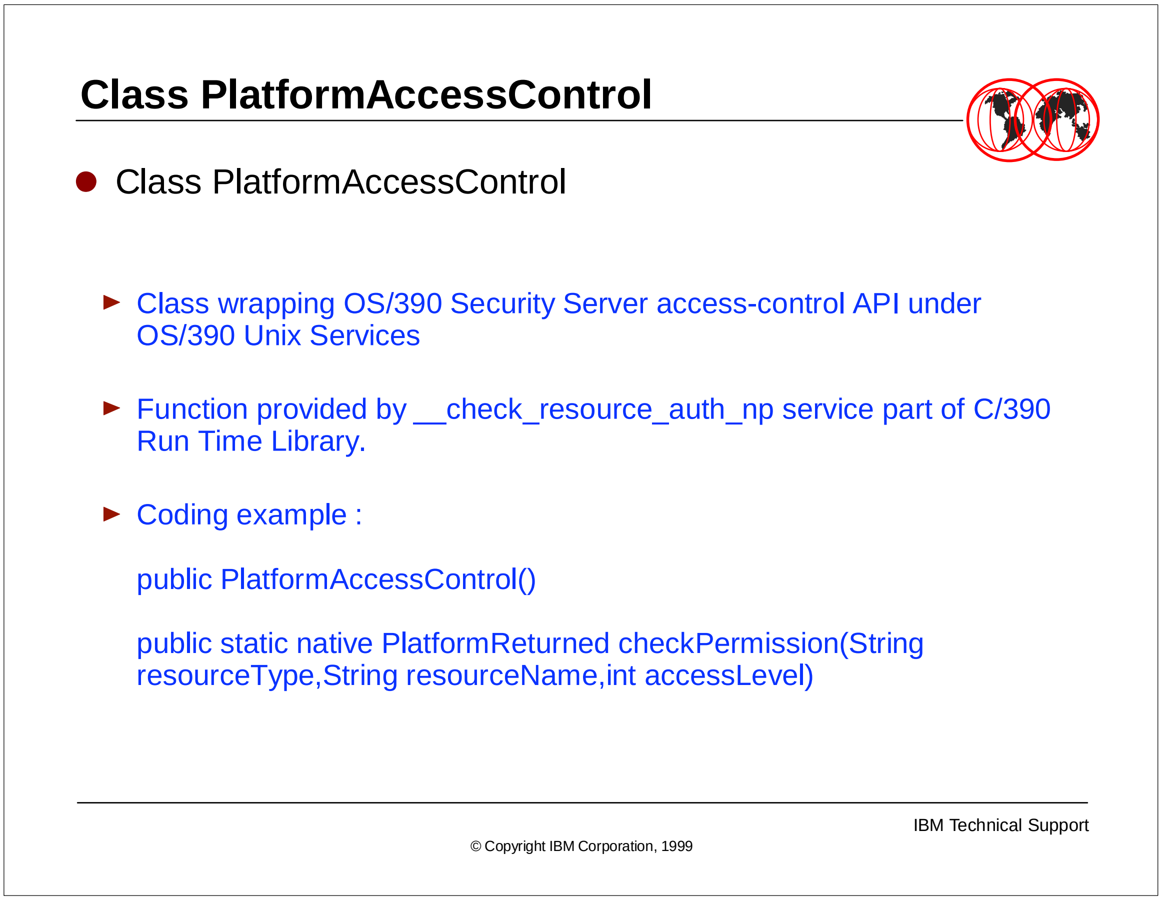


JAVA Classes

- These functions are implemented by five new classes :
  - PlatformAccessControl
  - PlatformThread
  - PlatformSecurityServer
  - PlatformAccessLevel
  - PlatformReturned

IBM Technical Support

- These new classes are provided with JDK 1.1.6
- The RACF.JAR that contains the new classes is located in /usr/lpp/java16/J1.1/lib/



Class PlatformAccessControl

- Class PlatformAccessControl
  - Class wrapping OS/390 Security Server access-control API under OS/390 Unix Services
  - Function provided by __check_resource_auth_np service part of C/390 Run Time Library.
  - Coding example :

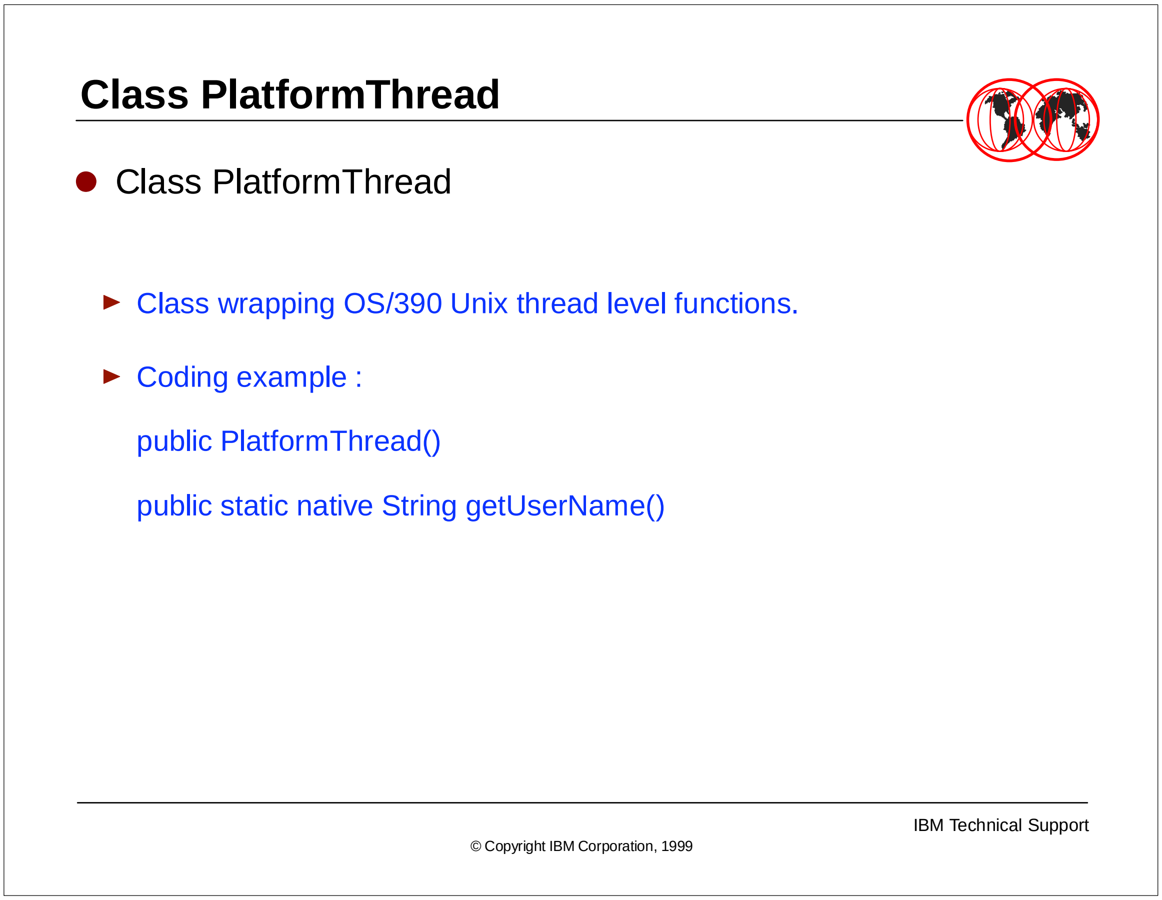
    public PlatformAccessControl()

    public static native PlatformReturned checkPermission(String resourceType,String resourceName,int accessLevel)

IBM Technical Support

- The Method "checkPersmission" is used to check "user in effect" permission to a resource. If the current platform thread has a security context the thread userid is used in an access control check. If not the userid of the Process is used in an access control check.

- Parameters:
  - resourceType, - a String with resource type (i.e. FACILITY).
  - resourceName, - a String with resource name (i.e. BPX.SERVER).
  - accessLevel, - an integer denoting acccess level Possible values for this parameter are listed in PlatformAccessLevel interface class.
- Returns:
  - If authorized, a null object is returned If NOT authorized, an instance of the PlatformReturned class is returned with



Class PlatformThread

- Class PlatformThread
  - ► Class wrapping OS/390 Unix thread level functions.
  - ► Coding example :

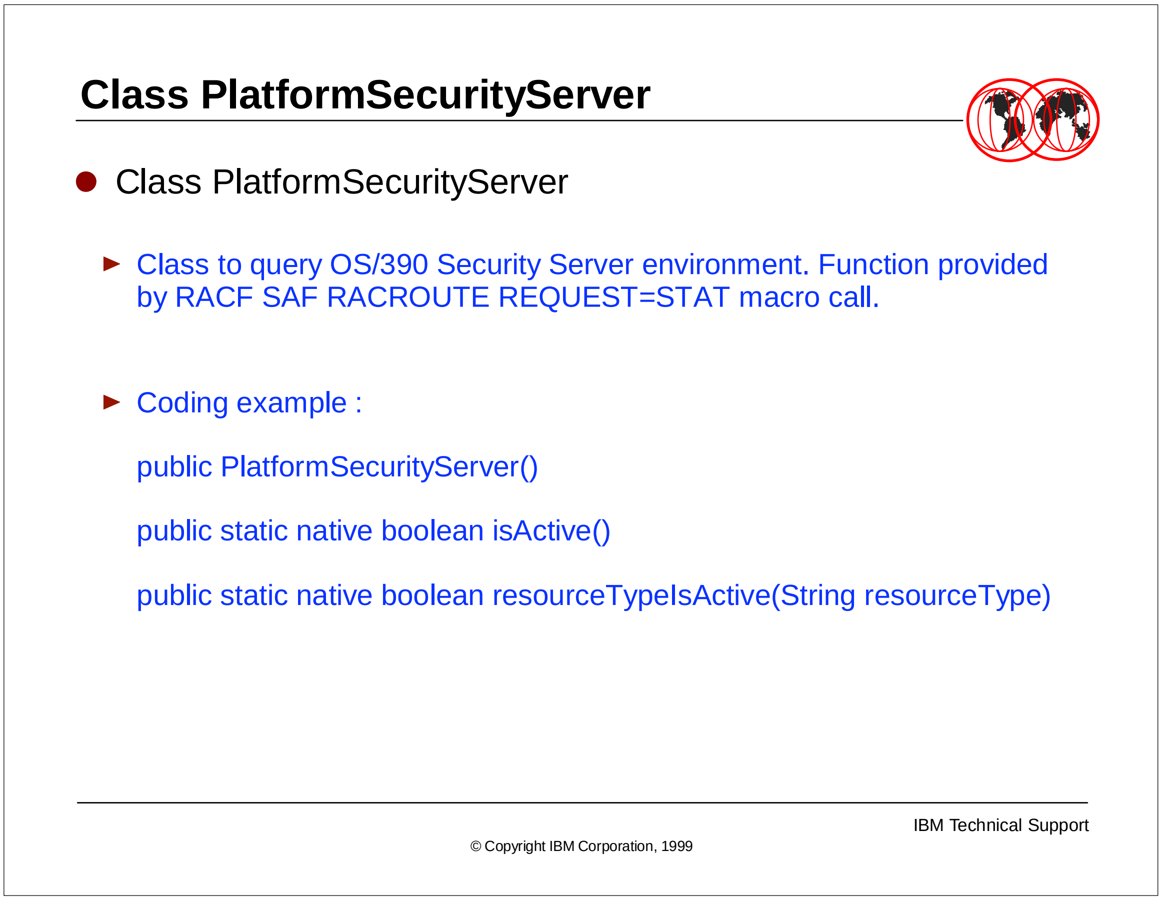
    public PlatformThread()

    public static native String getUserName()

IBM Technical Support

- ► The method getUserName extracts userName associated with the current platform-thread. This method wraps Asm/390 BPX1ENV Unix callable service.

- ► The method returns a String containing the OS/390 user name.



Class PlatformSecurityServer

- Class PlatformSecurityServer
  - Class to query OS/390 Security Server environment. Function provided by RACF SAF RACROUTE REQUEST=STAT macro call.
  - Coding example :

    public PlatformSecurityServer()

    public static native boolean isActive()

    public static native boolean resourceTypeIsActive(String resourceType)

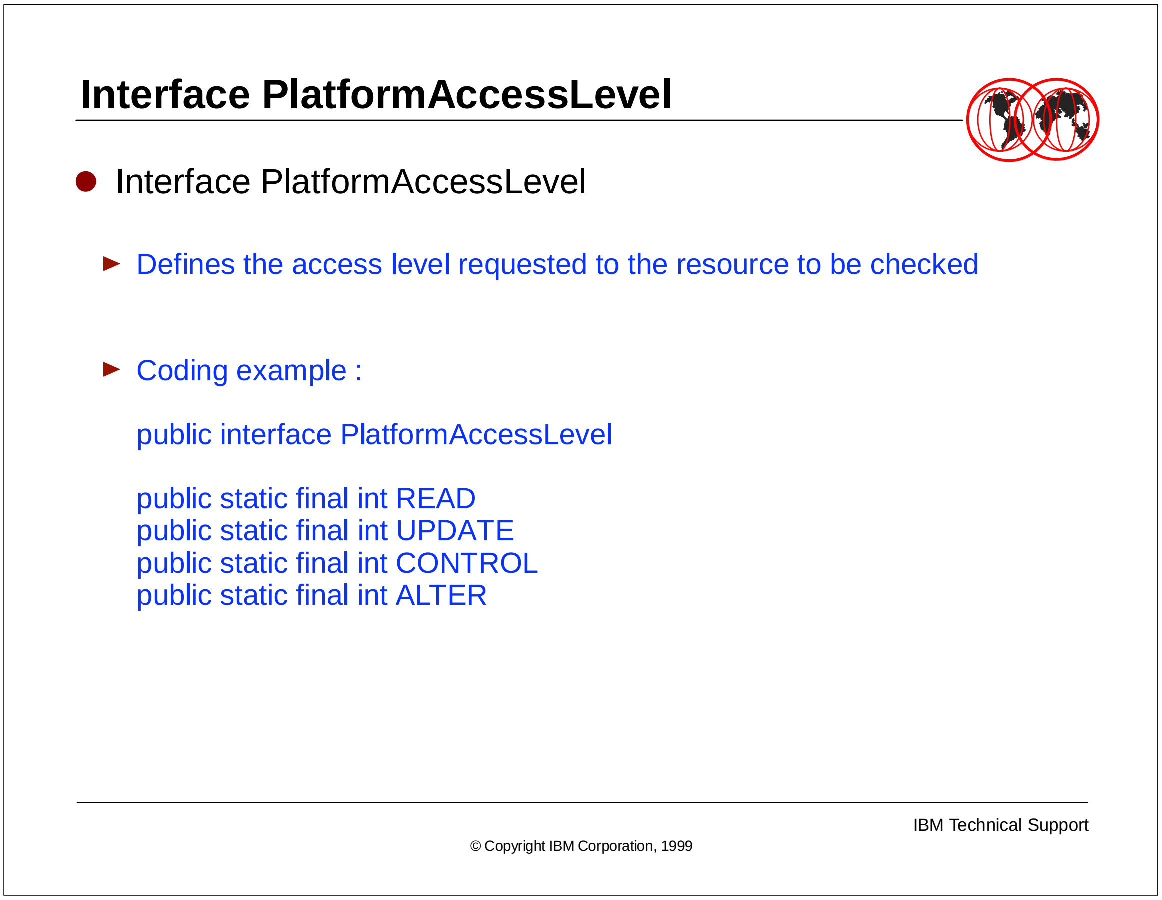
IBM Technical Support

- The class has two methods :
  - isActive()

    Method to check if a Security Server (i.e. RACF) is active. The method returns true or false.

  - resourceTypeIsActive(string)

    Method to check if a resource type (RACF class) is active. The methos returns a boolean true or false.



Interface PlatformAccessLevel

- Interface PlatformAccessLevel
  - Defines the access level requested to the resource to be checked
  - Coding example :

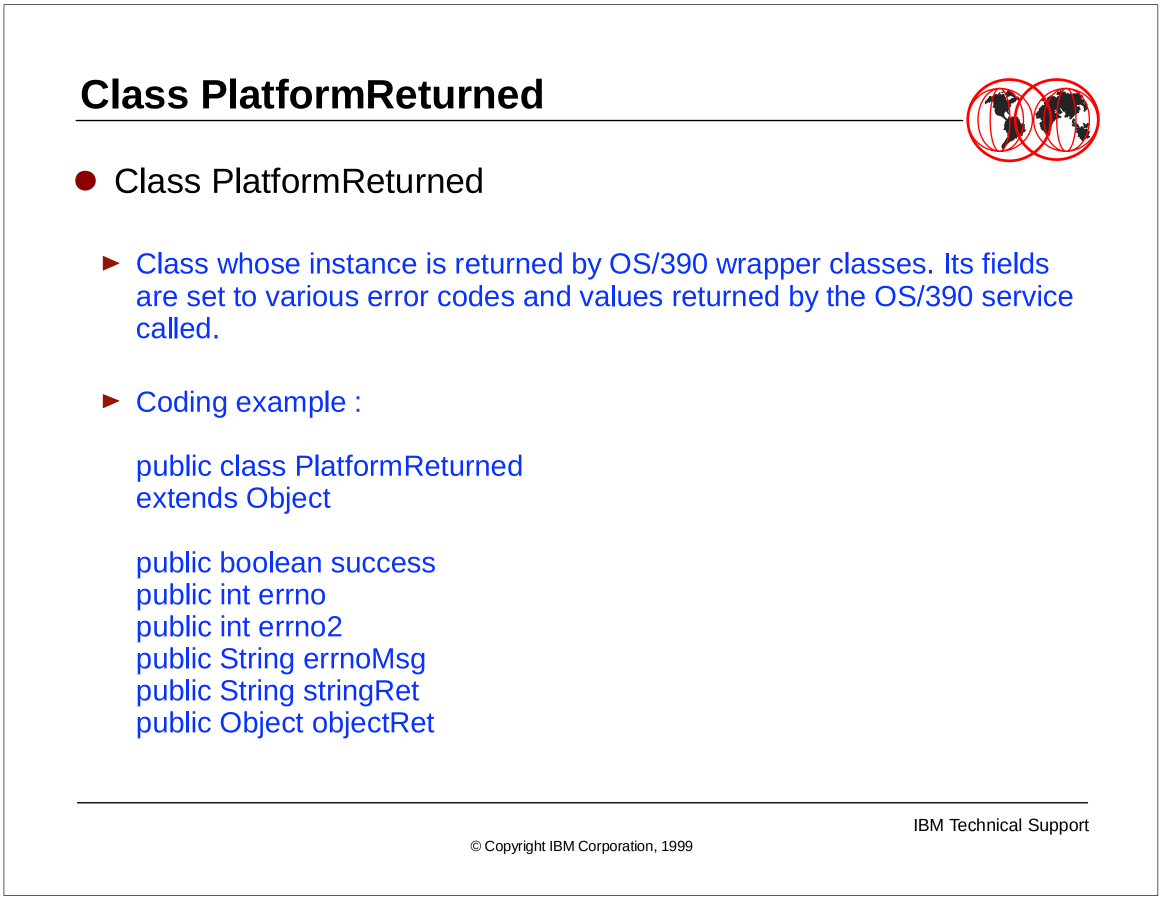
```
public interface PlatformAccessLevel

public static final int READ
public static final int UPDATE
public static final int CONTROL
public static final int ALTER
```

IBM Technical Support

- The PlatformAccessLevel is a interface rather then a class. Although it is similiar to a java class, except there is no data associated with the interface. The primary difference between a class and an interface is that the variables in an interface must be final, and the methods are only declarations.

- Place-holder for named constants used by accessLevel parameter of methods in PlatformAccessControl class. Java interface used as emulation of C enum definition.

  With OS/390 Security Server (RACF) permissions to resources are granted to a resource along with granularity specification of one of READ/UPDATE/CONTROL/ALTER levels.



Class PlatformReturned

- Class PlatformReturned
  - Class whose instance is returned by OS/390 wrapper classes. Its fields are set to various error codes and values returned by the OS/390 service called.
  - Coding example :

```
public class PlatformReturned
extends Object

public boolean success
public int errno
public int errno2
public String errnoMsg
public String stringRet
public Object objectRet
```

IBM Technical Support

- The variable index shows :
  - errno Instance variable to denote service C errno - first error field.
  - errno2 Instance variable to denote service C errno2 - secondary error field.
  - errnoMsg Instance variable to denote message string associated with errno.
  - objectRet Reference variable to an object returned by the Platform.
  - stringRet Reference variable to a String object returned by the Platform.
  - success Instance variable to denote service Success/Failure.

Coming with JDK 1.1.8

- Soon to be release JDK 1.1.8
  - New Classes for OS/390 Security Server
    - Perform a logon from a JAVA program !

- For more Information on JAVA on OS/390 see :

  www.s390.ibm.com/java/security.html