Trabajo de Grado
Migración de Código en Ambientes Distribuidos: Plataforma Aglets

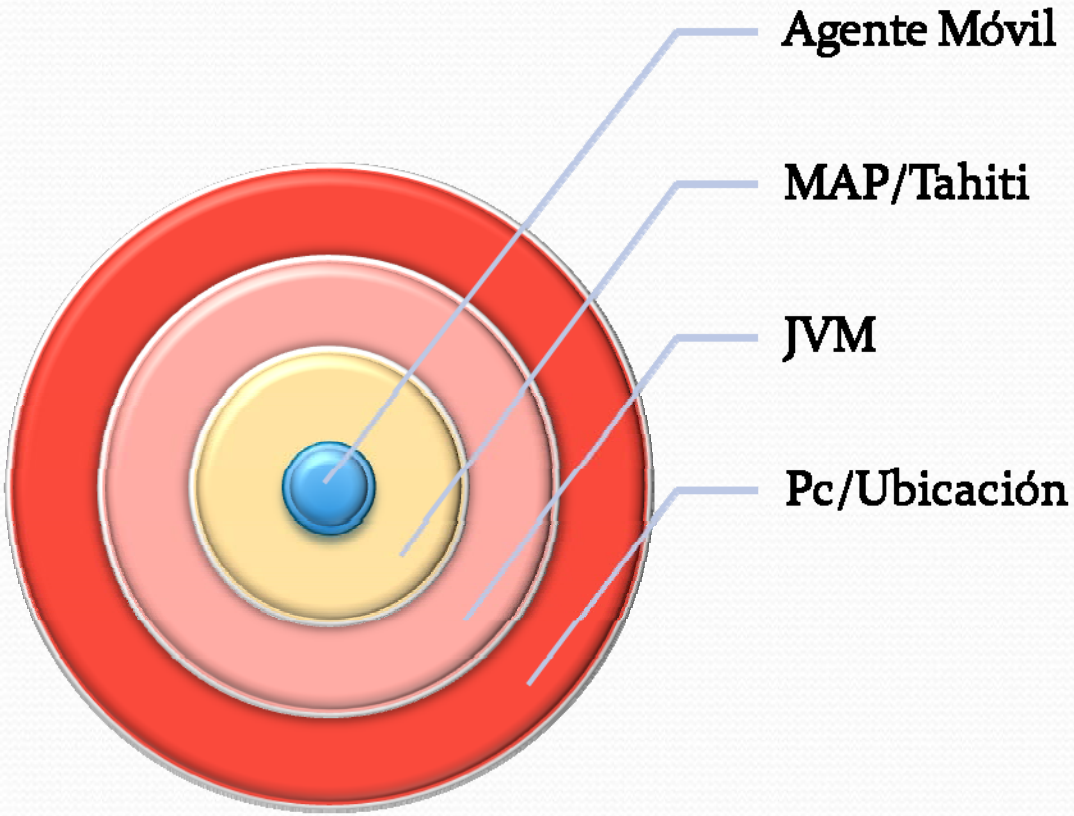
Trabajo de Grado
Migración de Código en Ambientes Distribuidos: Plataforma Aglets

Autor: García Martín, Santiago
Director: Tinetti, Fernando

Índice

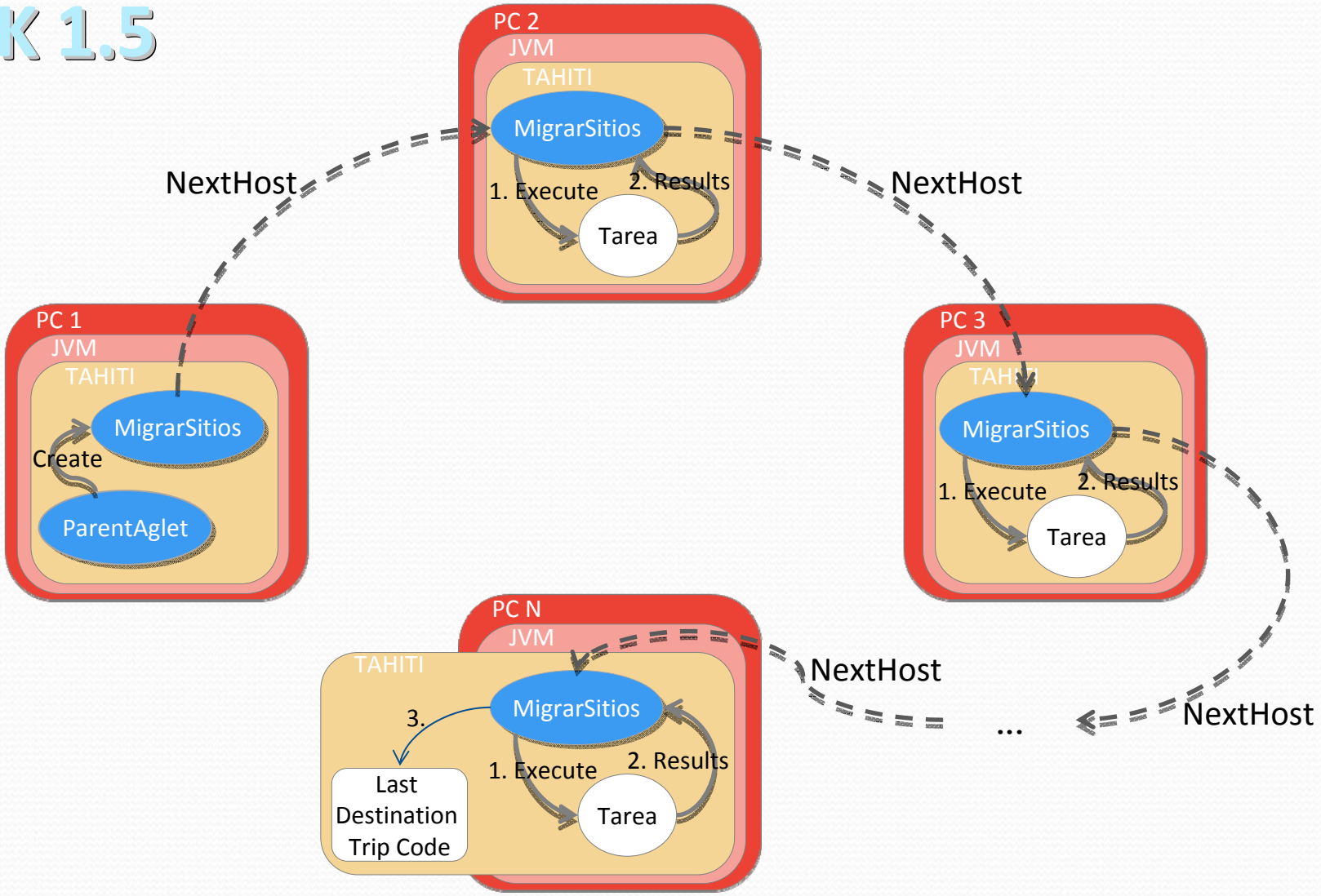
Información Adicional

Composición de software para la ejecución



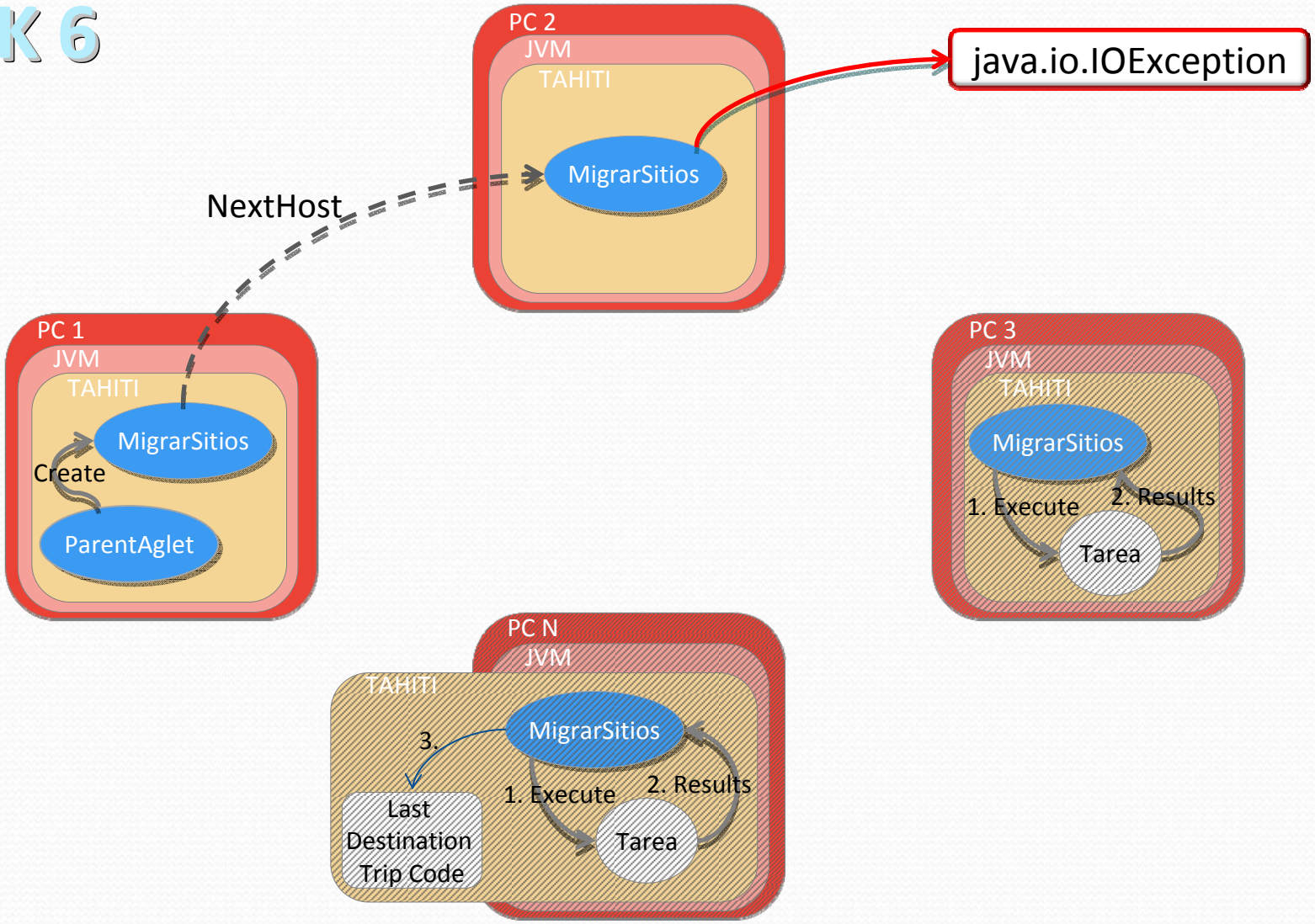
I. Definición del Problema

JDK 1.5



I. Definición del Problema

JDK 6



I. Definición del Problema

Importancia de Resolver el Problema

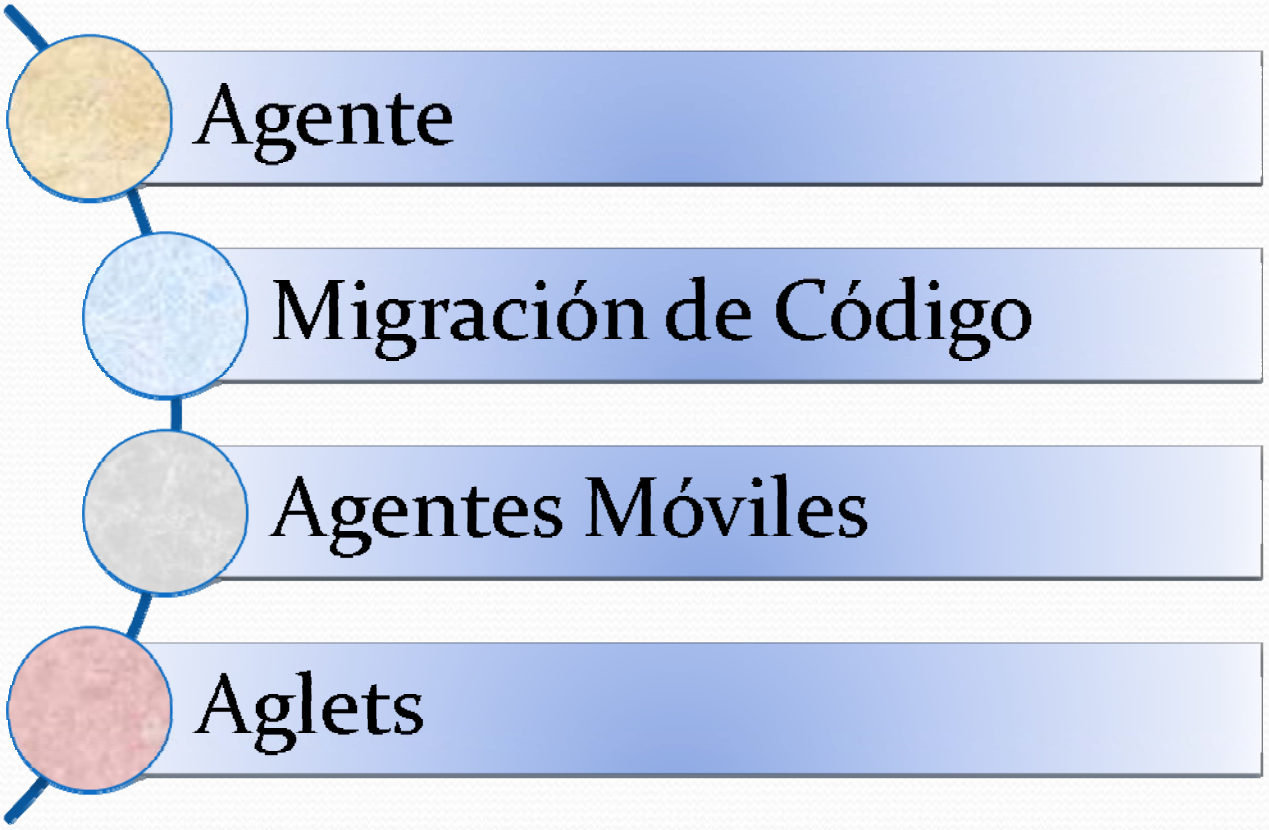
Utilización de Aglets c/ JDK 6

- 34 Paquetes
- 305 Clases

Resolver carga dinámica de clases en JDK 6

Índice

II. Conceptos Generales

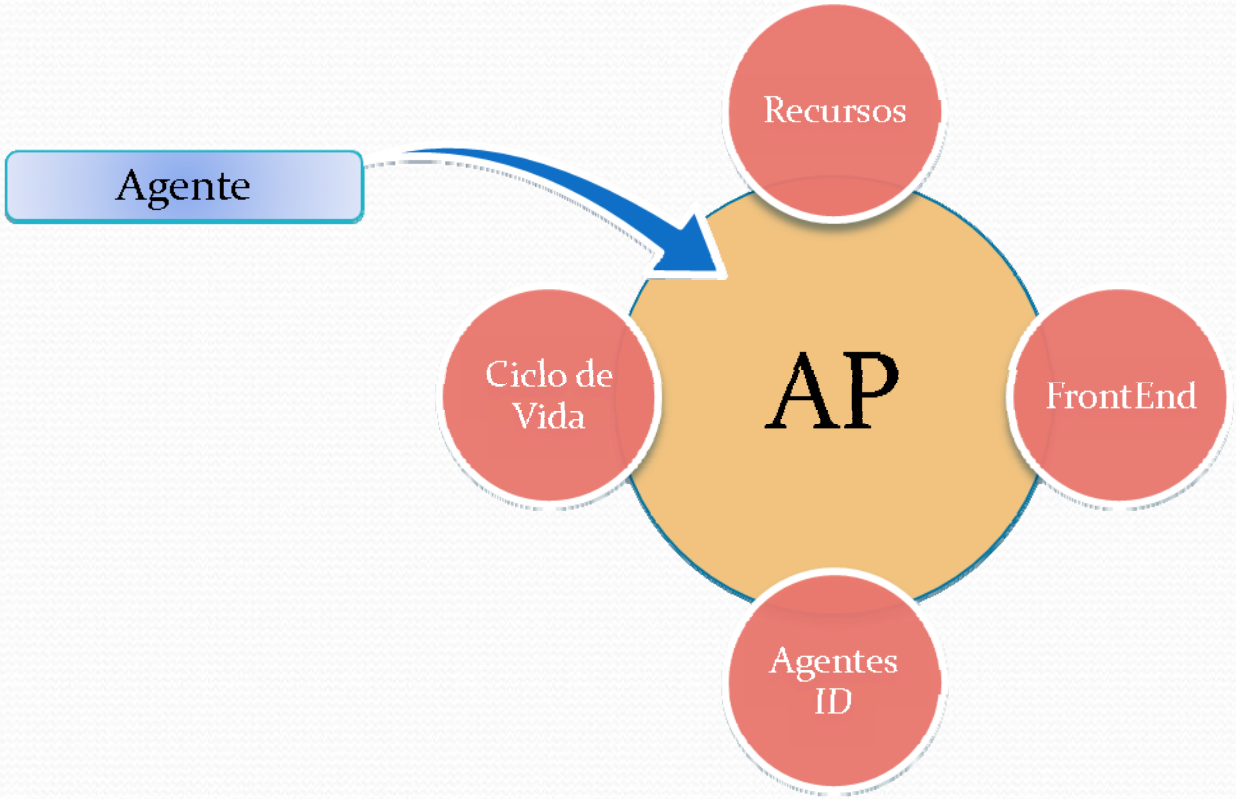


II. Conceptos Generales

- Agente
- Entidad Social
- Módulo Activo
- Pieza de Código Ejecutable

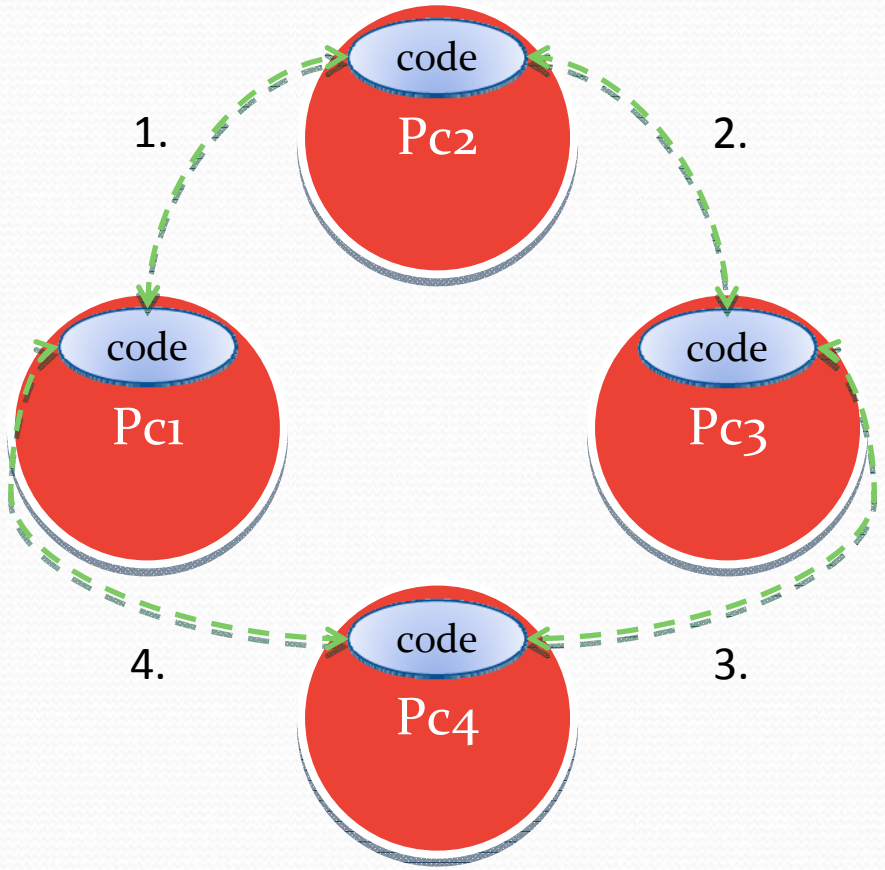
II. Conceptos Generales

Agente



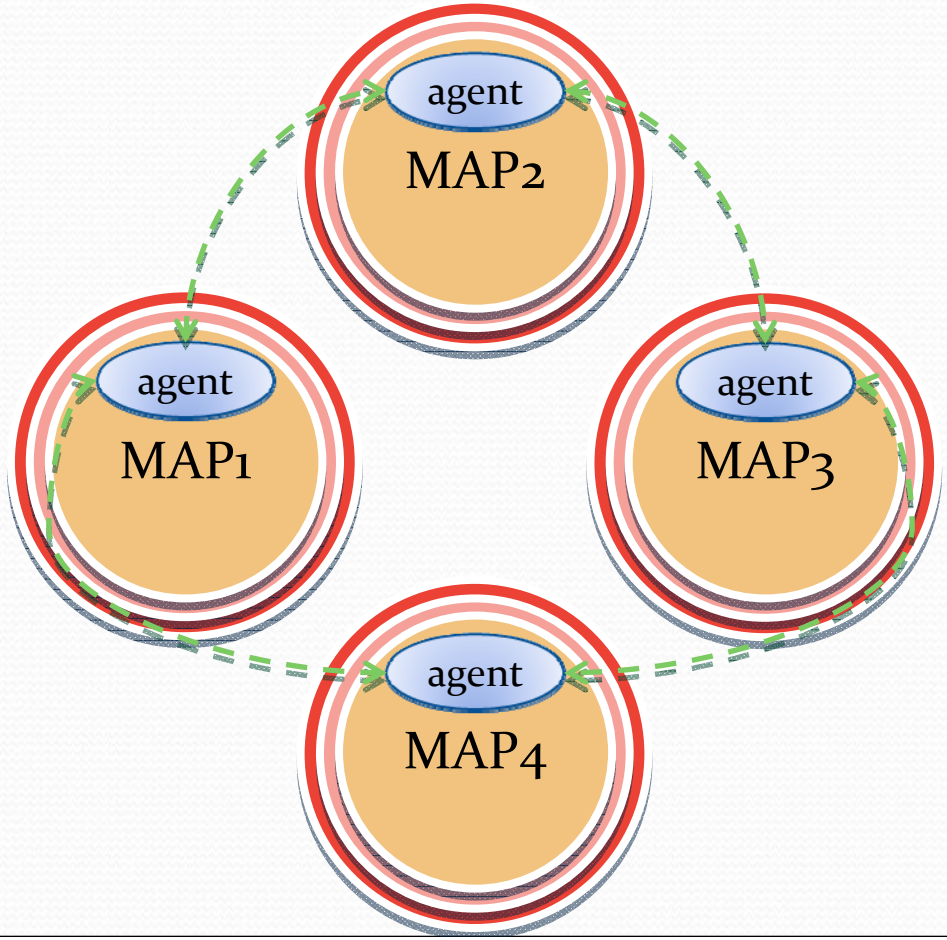
II. Conceptos Generales

Migración de Código



II. Conceptos Generales

Agentes Móviles

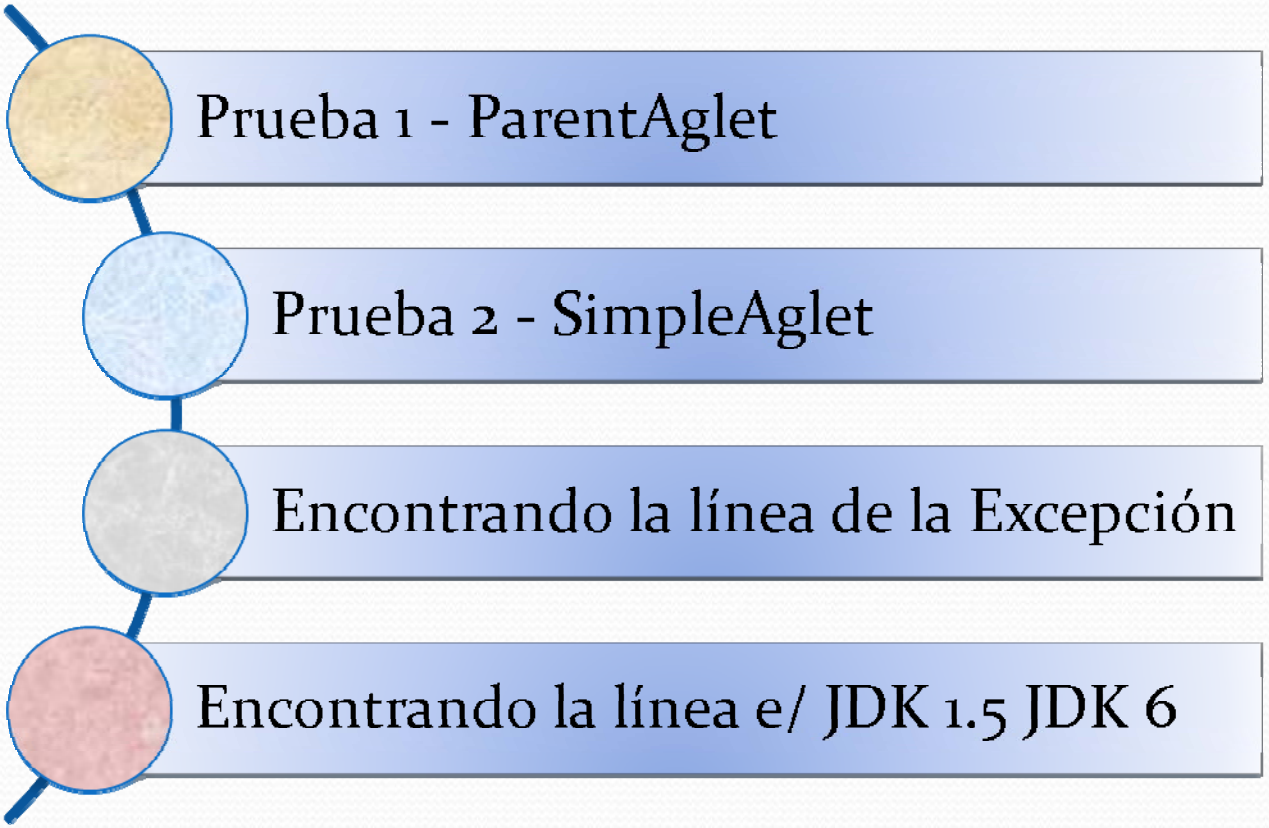


II. Conceptos Generales

- Aglets
  - Aglets MAP
  - Tahiti
  - Aglets SDK - ASDK

Índice

III. Debug


Prueba 1 - ParentAglet
Prueba 2 - SimpleAglet
Encontrando la línea de la Excepción
Encontrando la línea e/ JDK 1.5 JDK 6

III. Debug

Prueba 1 - ParentAglet

```
public class MigrarSitios extends Aglet
{
    Sla
    pub
    {
    }
    pub
    {
    class Tarea extends Task
    {
}
```

```
public class ParentAglet extends Aglet
{
    {variables privadas de la clase}
    public void onCreation(Object init)  { … }
    public boolean handleMessage(Message msg)  { … }
}
```

```
if(slaveTrip.atLastDestination())  { … }
public void execute(SeqItinerary i)  { … }
```

III. Debug

Prueba 1 - ParentAglet

Consola localhost:4500

[Warning: The hostname seems not having domain name.
Please try -resolve option to resolve the fully qualified hostname
or use -domain option to manually specify the domain name.]
***** Addr: atp://localhost:5000 place:
No integrity check because no security domain is authenticated.
java.io.IOException: FileNotFound:
/home/santi/Desktop/java/aglets/public/[Ljava/lang/Object;.class
...
{Líneas de excepción}
...
java.io.IOException: FileNotFound: /home/santi/Desktop
/java/aglets/public/[B.class
...
{Líneas de excepción}
...
Padre Finalizado

Consola localhost:5000

[Warning: The hostname seems not having domain name.
Please try -resolve option to resolve the fully qualified hostname
or use -domain option to manually specify the domain name.]
java.io.IOException: atp://localhost:4500/[Ljava/lang/Object;.class
...
{Líneas de excepción}
...
java.io.IOException: atp://localhost:4500/[B.class

III. Debug

Prueba 2 - SimpleAglet

```
public class SimpleAglet extends Aglet
{
                    boolean migrado = false;
    String[] lista = null;

    public void run()
    {
        System.out.println("Empieza Run()");
        if (migrado)
        {
            lista = new String[2];
            lista[0] = "String0-1";
            lista[1] = "String1-1";
            for (String s : lista)
            {   System.out.println(s);   }
        }
```

```
        else
        {
            migrado = true;
            lista = new String[2];
            lista[0] = "String0-0";
            lista[1] = "String1-0";
            for (String s : lista)
            {   System.out.println(s);   }
            try
            {   dispatch(new java.net.URL("atp://localhost:5000"));   }
            catch (Exception e)
            {   System.out.println("Failed dispatch to 5000");e.getMessage();   }
        }
    }
}
```

III. Debug

Prueba 2 - SimpleAglet

Consola localhost:4500

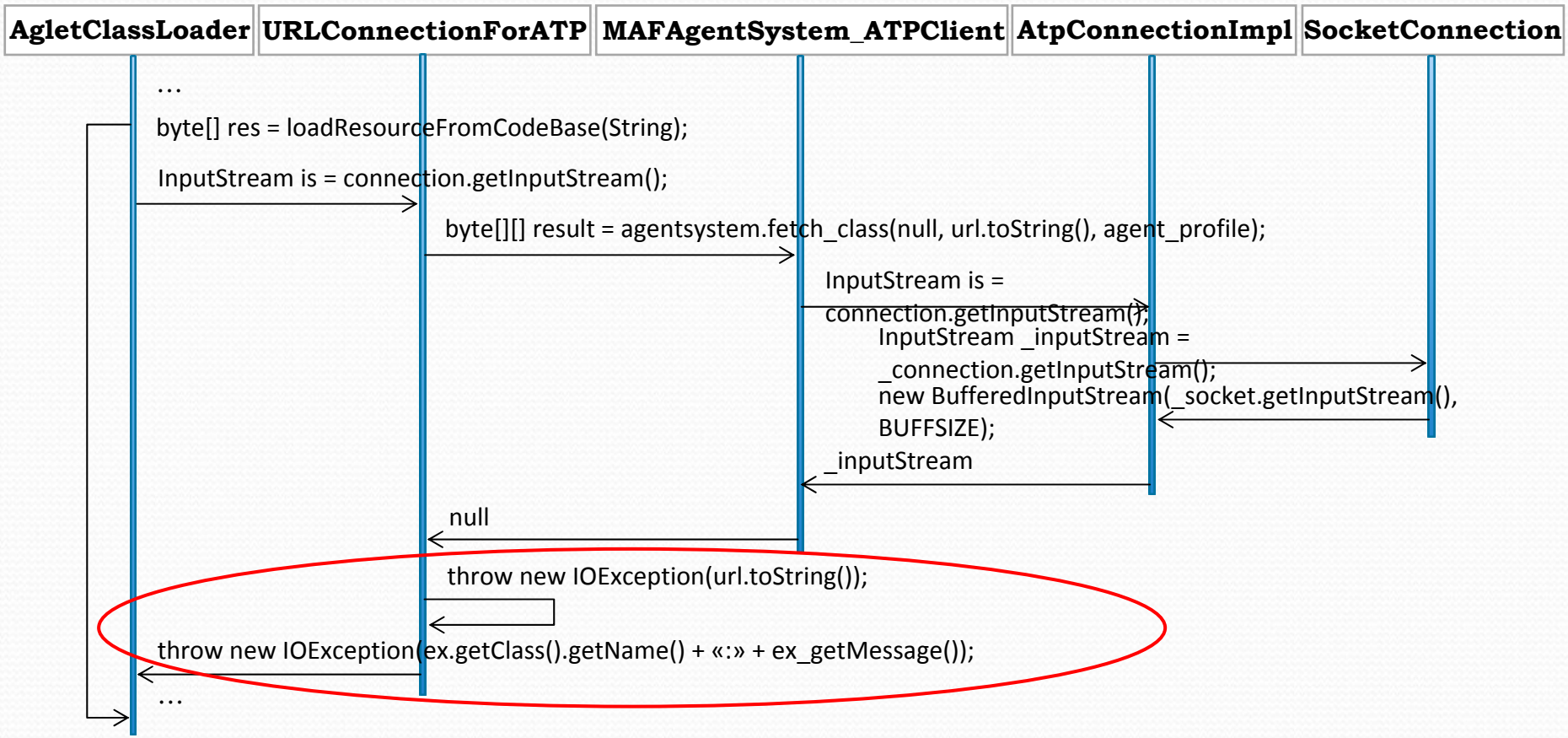
[Warning: The hostname seems not having domain name.
Please try -resolve option to resolve the fully qualified hostname
or use -domain option to manually specify the domain name.]
Empieza Run()
String0-0
String1-0
***** Addr: atp://localhost:5000 place:
No integrity check because no security domain is authenticated.
java.io.IOException: FileNotFound:
/home/santi/Desktop/java/aglets/public/[Ljava/lang/String;.class
...
{Líneas de excepción}
...
Failed dispatch to 5000

Consola localhost:5000

[Warning: The hostname seems not having domain name.
Please try -resolve option to resolve the fully qualified hostname
or use -domain option to manually specify the domain name.]
java.io.IOException: atp://localhost:4500/[Ljava/lang/String;.class
...
{Líneas de excepción}
...
java.lang.ClassNotFoundException: [Ljava.lang.String;
...
{Líneas de excepción}
...

III. Debug

Encontrando la línea de la Excepción



III. Debug

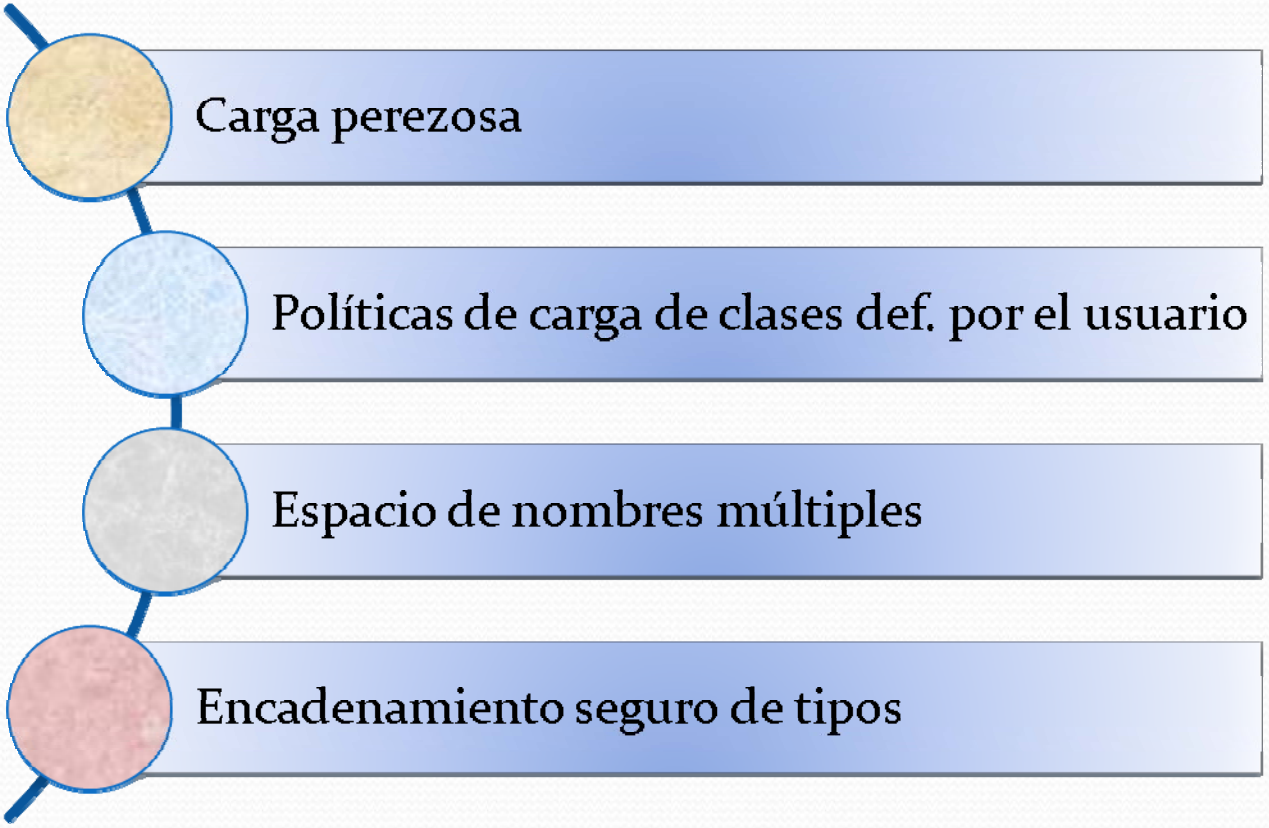
Encontrando la línea e/ JDK 1.5 JDK 6

```
596:    try {
597:            Class cl = findResolvedClass(name);
598:
599:            if (cl != null) {
600:                log.debug("Using class " + name + " in resolved cache");
601:                return cl;
602:            }

{...}

681:    try {
682:            clazz = findSystemClass(name);
                                        /* Línea donde difiere JDK 1.5 y JDK 6*/
683:            if (clazz != null)
684:            {
685:                log.debug("Loading " + name + " from System");
686:                return clazz;
687:            }
688:     }
689:     catch (ClassNotFoundException ex) {}
```

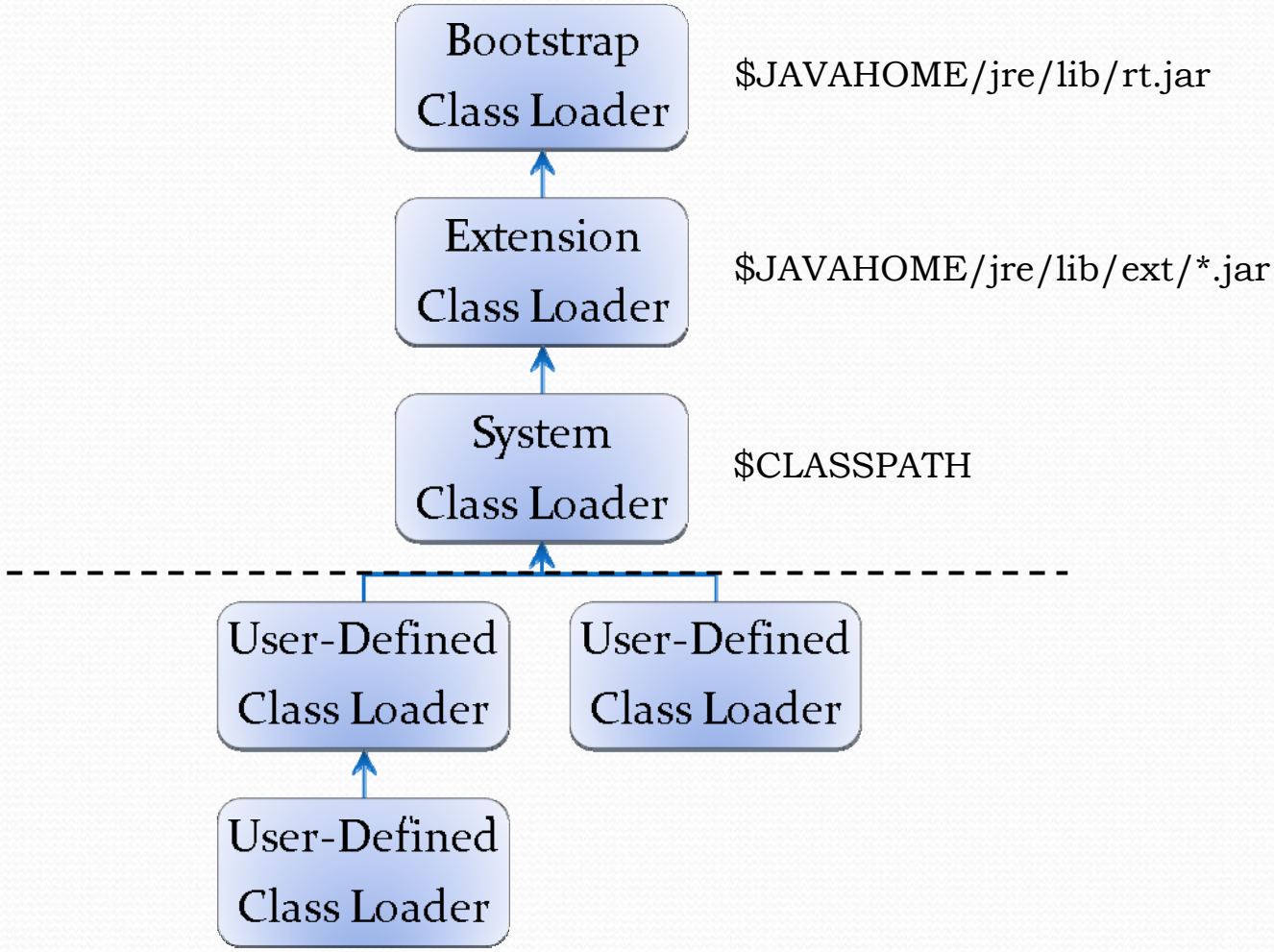
Índice

IV. Sistema de carga de clases en Java

Classloader

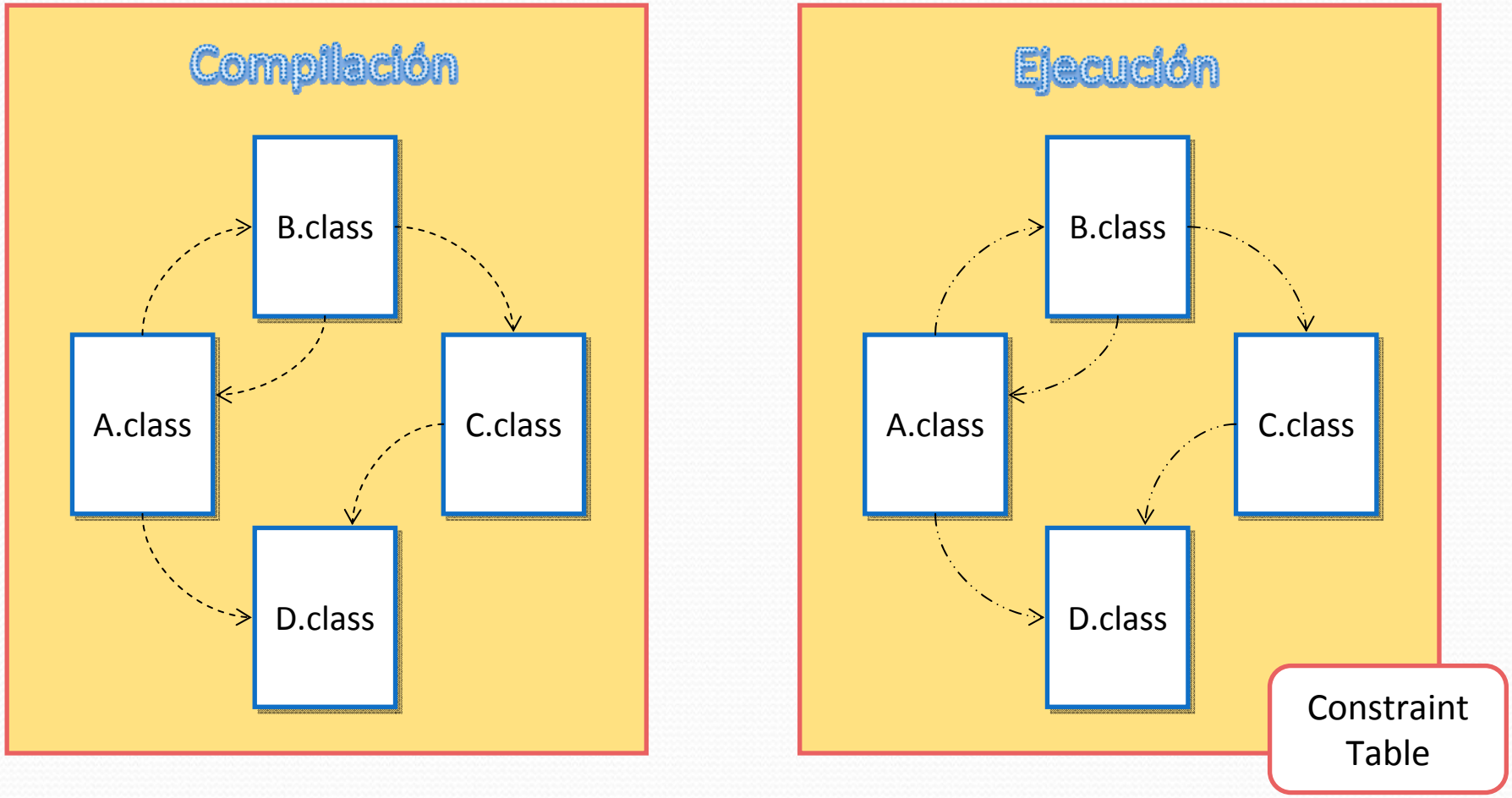


El Modelo de Delegación

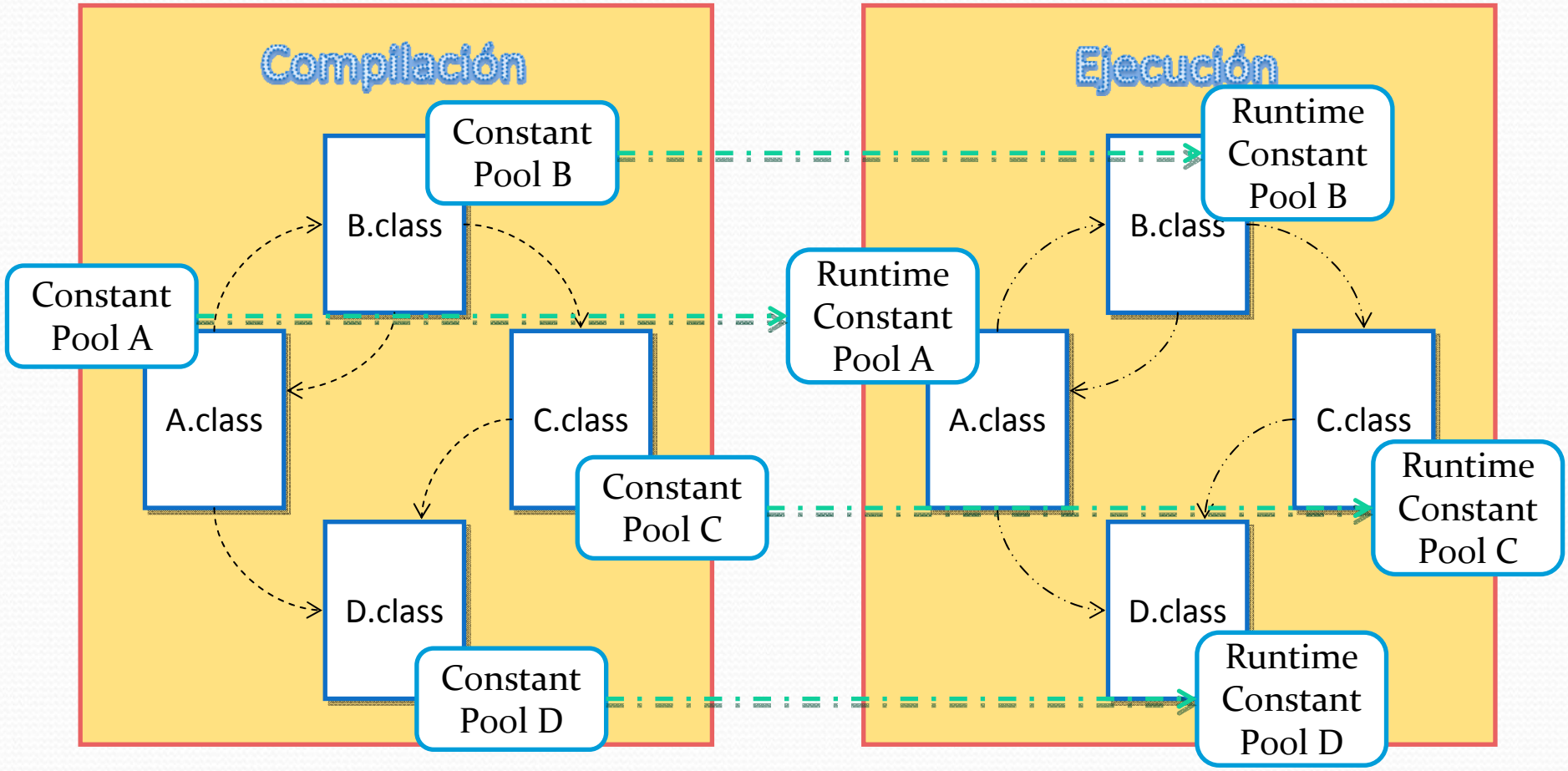


Encadenamiento seguro de tipos

Loader Constraints



Constant Pool - Runtime Constant Pool



Runtime Constant Pool

Método Class.getName() - Ejemplo

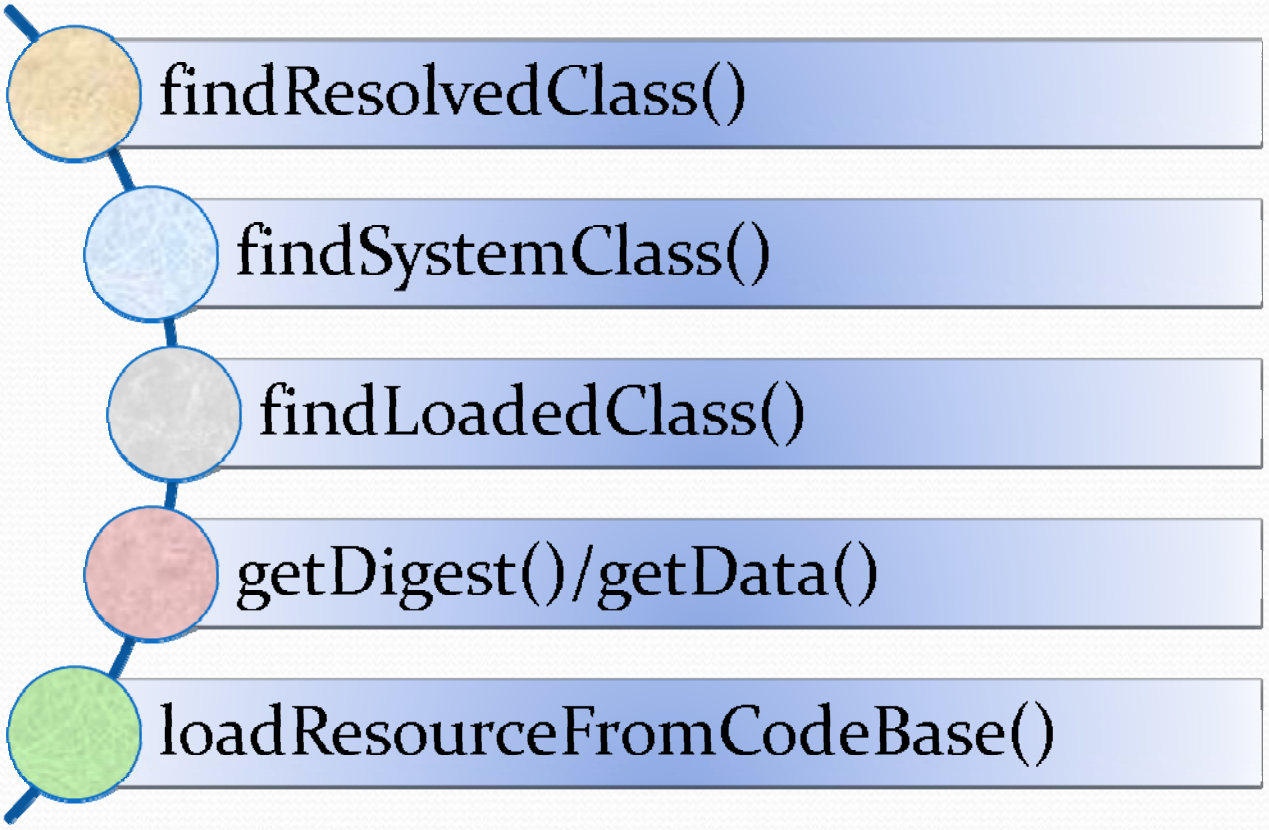
java.lang.String[] → [...

Descriptores de Campo	
Tipo de Elemento	Codificación
Boolean	Z
Char	C
...	...
class o interface	L<classname>;

→ [Ljava.lang.String;

Índice

Método loadClass()



Índice

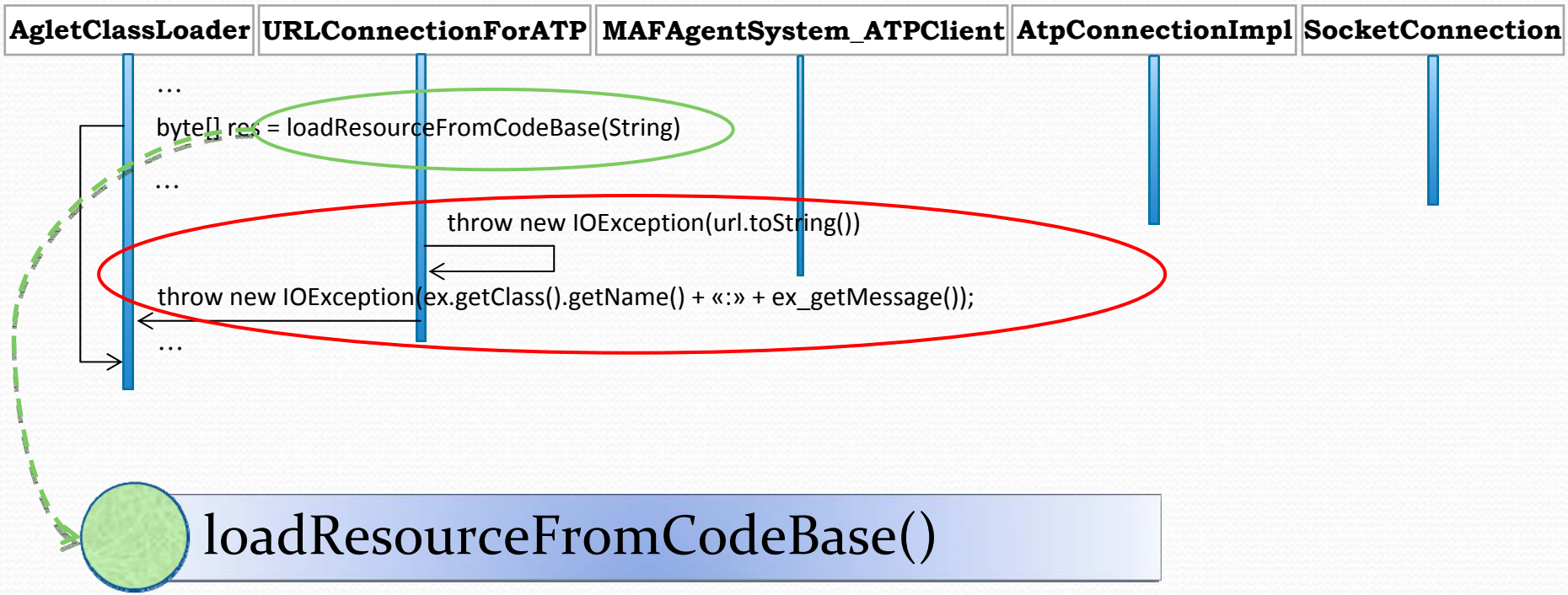
VI. Entendiendo la Excepción

java.io.IOException:
atp://localhost:4500/[Ljava/lang/String;.class

1. atp://localhost:4500/
2. [Ljava/lang/String;.class

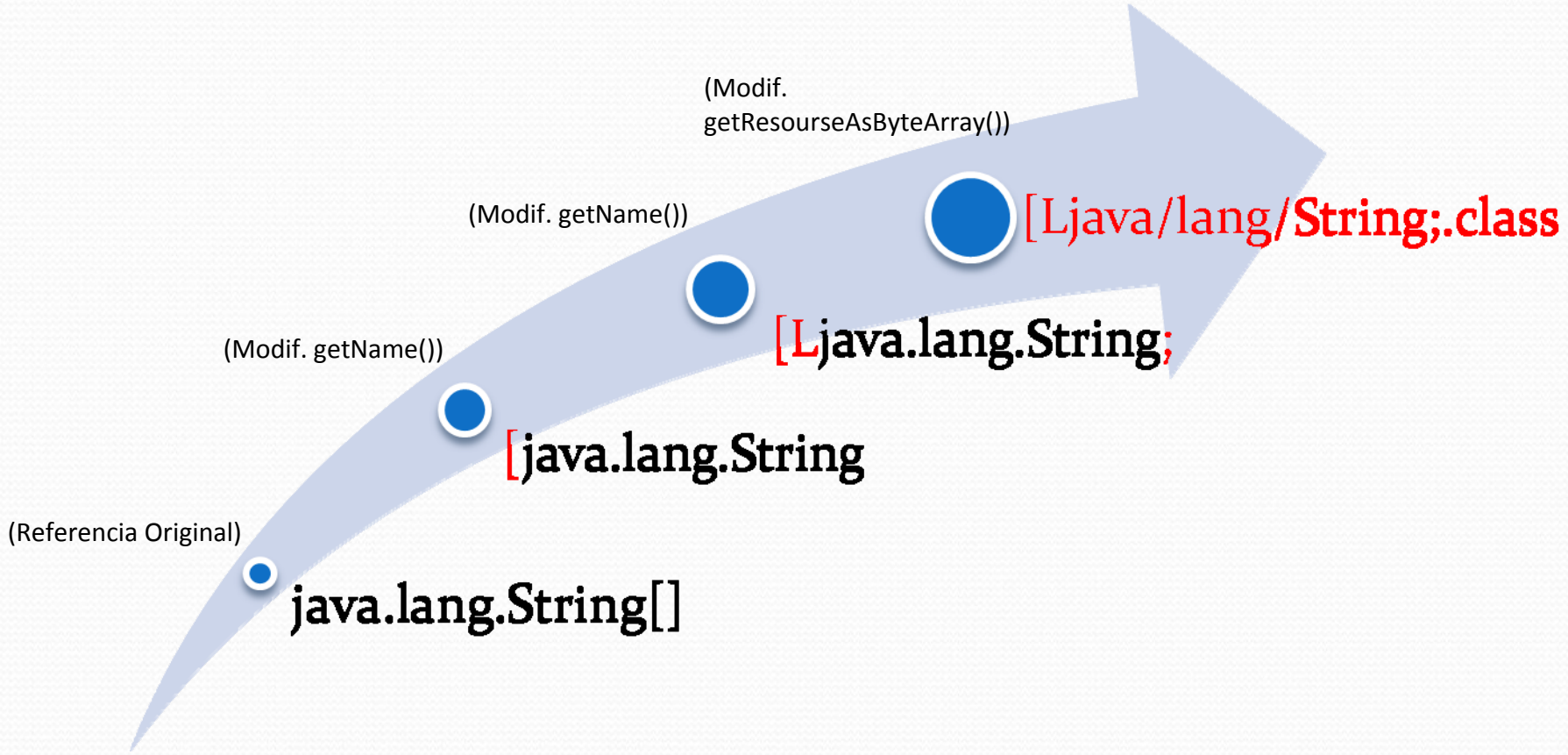
VI. Entendiendo la Excepción

1. atp://localhost:4500/



VI. Entendiendo la Excepción

2. [Ljava/lang/String;.class



Índice

Excepciones hasta el momento

java.io.IOException: FileNotFound: /home/santi/Desktop/java/aglets/public/[Ljava/lang/Object;.class

- ParentAglet

java.io.IOException: atp://localhost:4500/[B.class

- ParentAglet

java.io.IOException: FileNotFound: /home/santi/Desktop/java/aglets/public/[Ljava/lang/String;.class

- SimpleAglet

Bug ID

Bug ID: 6434149
Synopsis (cl) ClassLoader.loadClass() throws java.lang.ClassNotFoundException: [Ljava.lang.String; in JDK 6.0
Description Consider this simple test program:

```
public class test {

    public static void main(String[] args) throws Exception {

        String[] s = new String[] { "123" };
        String clName = s.getClass().getName();
        test.class.getClassLoader().loadClass(clName);
    }
}
```

This runs fine on JDK 1.5, but throws this exception on JDK 6.0:

```
Exception in thread "main" java.lang.ClassNotFoundException: [Ljava.lang.String;
        at java.net.URLClassLoader$1.run(URLClassLoader.java:200)
        at java.security.AccessController.doPrivileged(Native Method)
        at java.net.URLClassLoader.findClass(URLClassLoader.java:188)
        at java.lang.ClassLoader.loadClass(ClassLoader.java:306)
        at sun.misc.Launcher$AppClassLoader.loadClass(Launcher.java:276)
        at java.lang.ClassLoader.loadClass(ClassLoader.java:251)
        at test.main(test.java:7)
```

Posted Date : 2006-06-05 21:54:58.0

Work Around Change classLoader.loadClass(className) into Class.forName(className, false, classLoader).

¿ findSystemClass() = loadClass()?

Implementación findSystemClass()

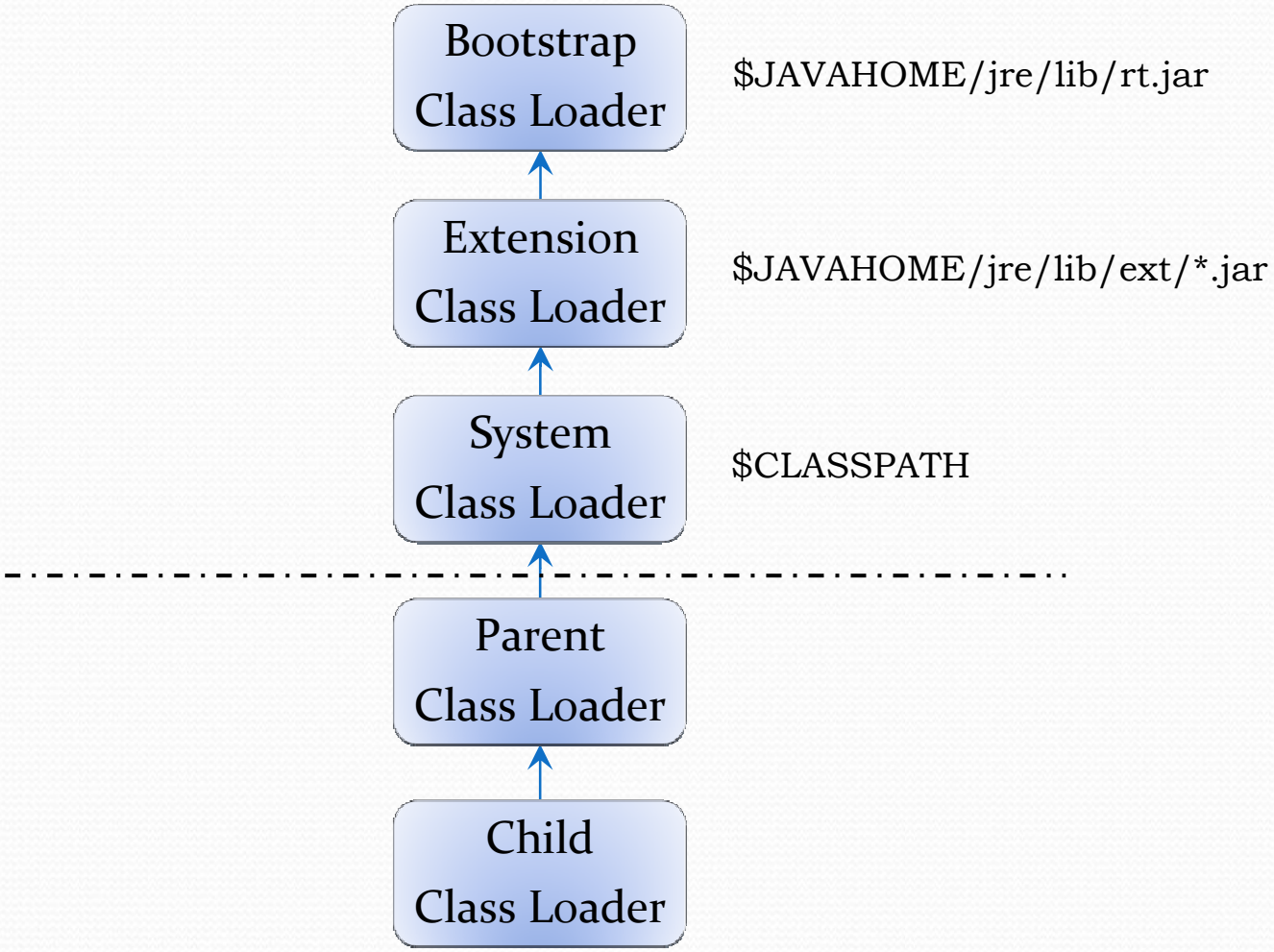
```
protected final Class<?> findSystemClass(String name)
                                         throws ClassNotFoundException
{
    check();
    ClassLoader system = getSystemClassLoader();
    if (system == null)
    {
        if (!checkName(name))
            throw new ClassNotFoundException(name);
        return findBootstrapClass(name);
    }
    return system.loadClass(name);
}
```

IV. Debug

Prueba 2 - SimpleAglet

```
596:    try {
597:            Class cl = findResolvedClass(name);
598:
599:            if (cl != null) {
600:                log.debug("Using class " + name + " in resolved cache");
601:                return cl;
602:            }

{...}

681:    try {
682:            clazz = Class.forName(name,false,this);
                                        /* Línea donde difiere JDK 1.5 y JDK 6*/
683:            if (clazz != null)
684:            {
685:                log.debug("Loading " + name + " from System");
686:                return clazz;
687:            }
688:     }
689:     catch (ClassNotFoundException ex) {}
```

Prueba 4 - LinkageErrorProof



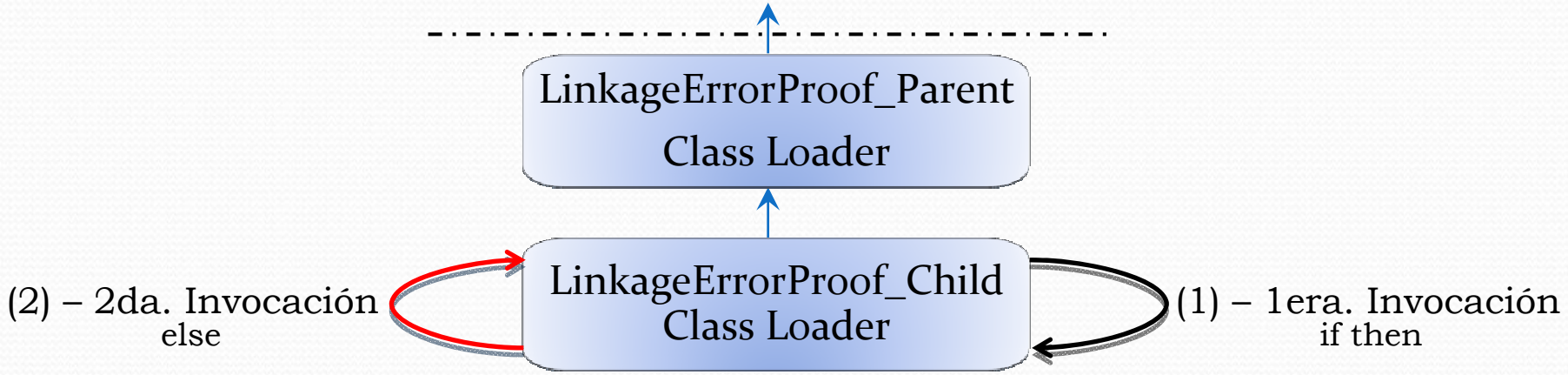
Prueba 4 - LinkageErrorProof

```
class LinkageErrorProof_Child extends ClassLoader
{
  int i= 0;
  public LinkageErrorProof_Child(ClassLoader cl)
  {   super(cl);   }

  protected synchronized Class loadClass(String name, boolean resolve)
                                                    throws ClassNotFoundException{
     Class clazz = null;
     try{
        if(i==0){
            i++;
            (1) clazz=Class.forName(name,false,???);
        }
       else{
            (2) clazz=Class.forName(name,false,???);
        }
     }
     catch(Error ex){e.printStackTrace();}
     finally{return clazz;}
  }

  public void test(String n,boolean b) throws ClassNotFoundException{
      ((LinkageErrorProof_Parent)this.getParent()).setChild(this);
      this.loadClass(n,b);
  }
}
```

Prueba 4 - LinkageErrorProof

1.

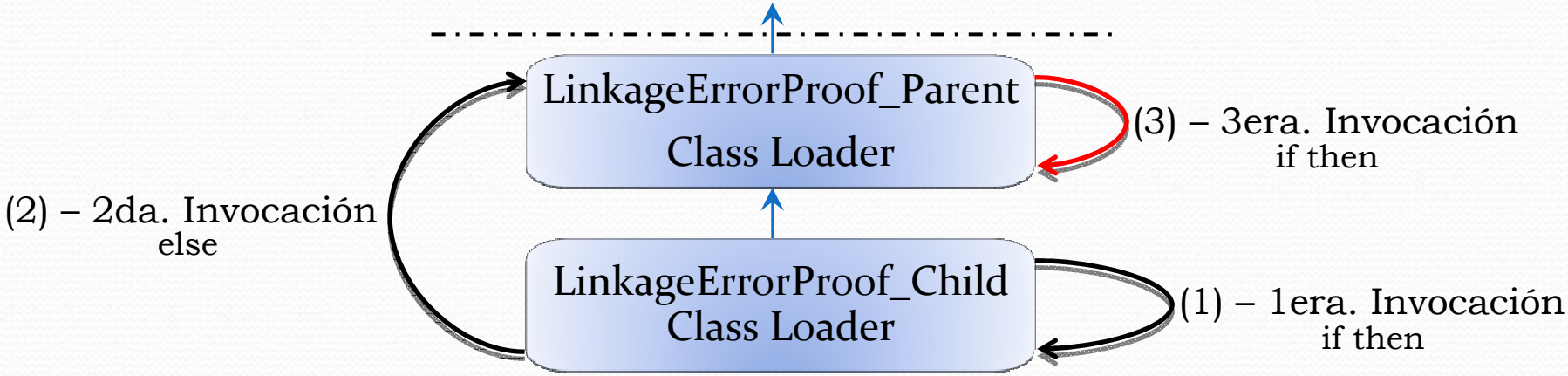
(1) clazz=Class.forName(name,false,this); (LinkageErrorProof_Child)
(2) clazz=Class.forName(name,false,this); (LinkageErrorProof_Child)
(3) Indistinto ya que el error se debería haber producido.
(4) Indistinto ya que el error se debería haber producido.



Prueba 4 - LinkageErrorProof

2.

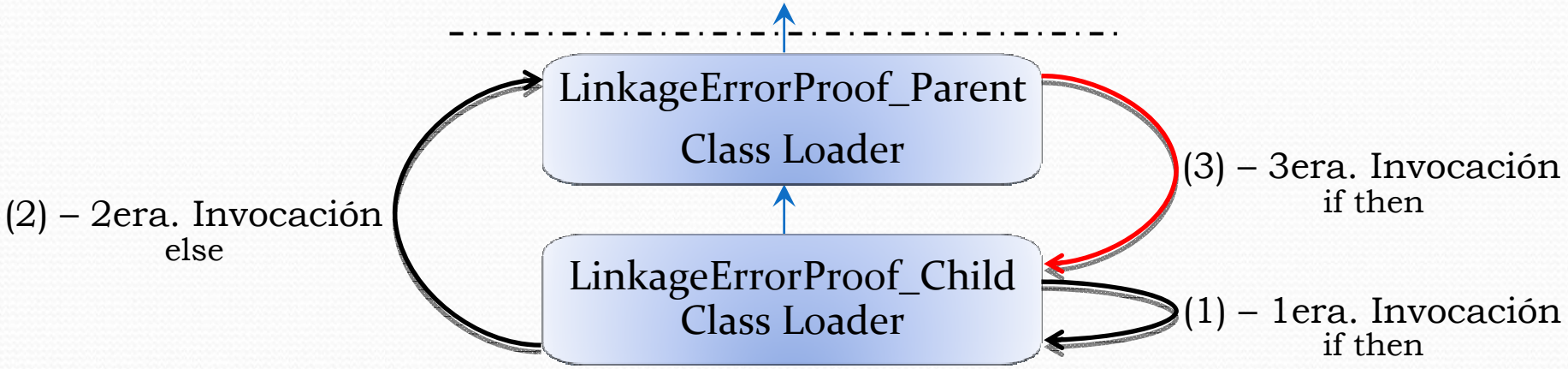
(1) clazz=Class.forName(name,false, this); (LinkageErrorProof_Child)
(2) clazz=Class.forName(name,false, this.getParent()); (LinkageErrorProof_Child)
(3) clazz=Class.forName(name,false,this); (LinkageErrorProof_Parent)
(4) Indistinto ya que el error se debería haber producido.



Prueba 4 - LinkageErrorProof

3.

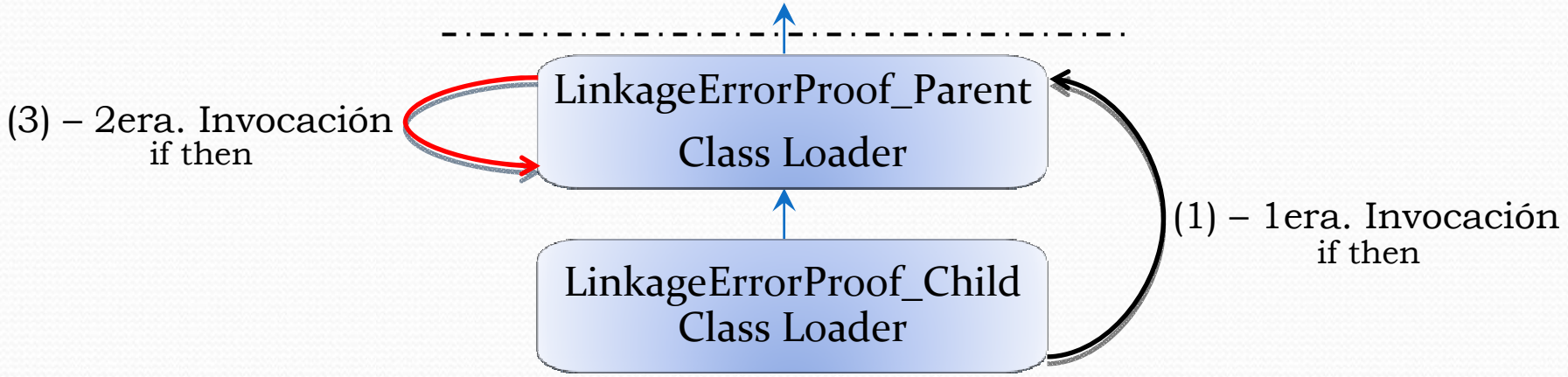
(1) clazz=Class.forName(name,false, this); (LinkageErrorProof_Child)
(2) clazz=Class.forName(name,false, this.getParent()); (LinkageErrorProof_Child)
(3) clazz=Class.forName(name,false,this.getChild()); (LinkageErrorProof_Parent)
(4) Indistinto ya que el error se debería haber producido.



Prueba 4 - LinkageErrorProof

4.

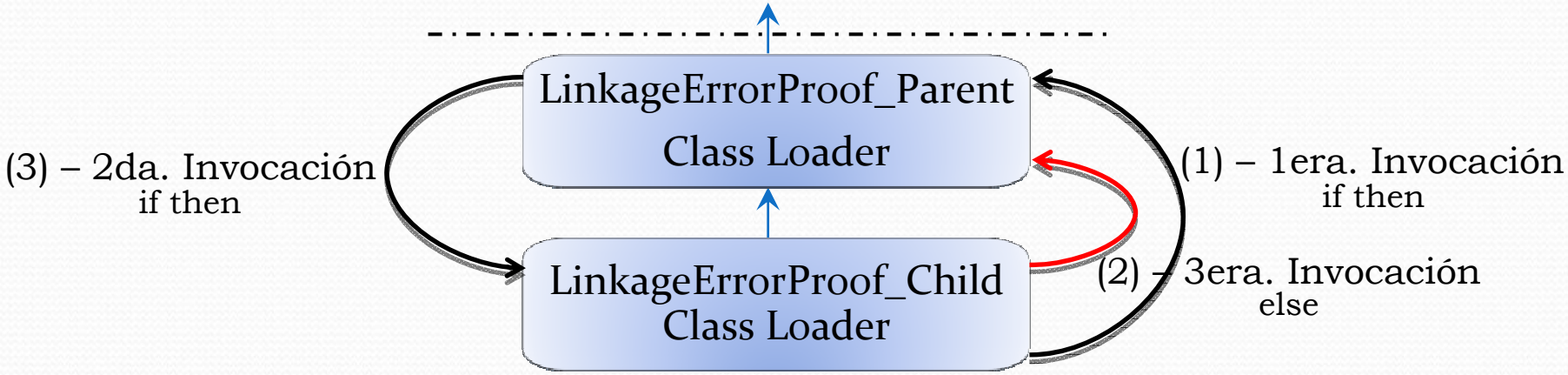
(1) clazz=Class.forName(name,false, this.getParent()); (LinkageErrorProof_Child)
(2) Indistinto ya que el error se debería haber producido.
(3) clazz=Class.forName(name,false, this); (LinkageErrorProof_Parent)
(4) Indistinto ya que el error se debería haber producido.



Prueba 4 - LinkageErrorProof

5.

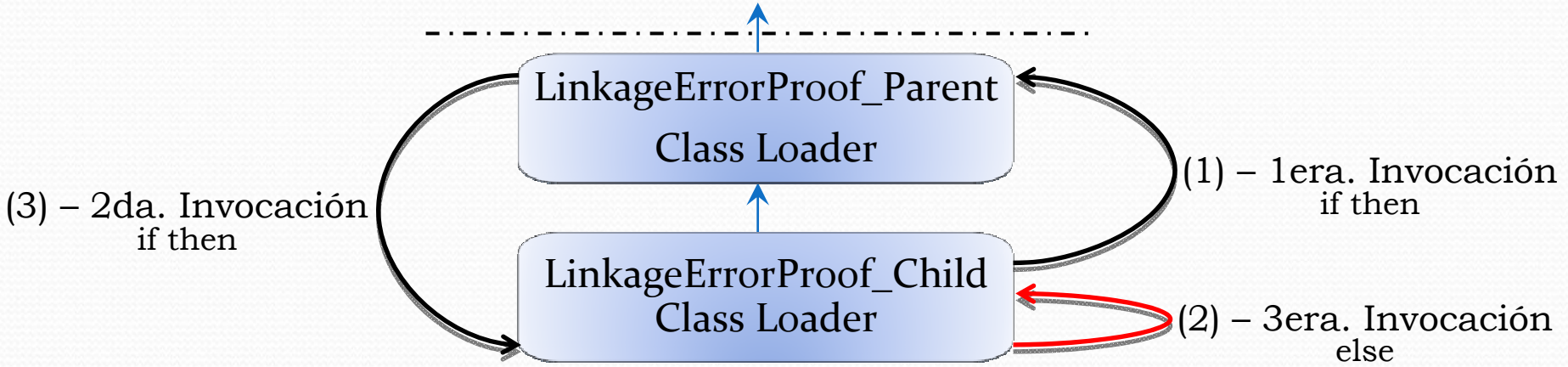
(1) clazz=Class.forName(name,false, this.getParent()); (LinkageErrorProof_Child)
(2) clazz=Class.forName(name,false, this.getParent());
(LinkageErrorProof_Child)
(3) clazz=Class.forName(name,false, this.getChild()); (LinkageErrorProof_Parent)
(4) Indistinto ya que el error se debería haber producido.



Prueba 4 - LinkageErrorProof

6.

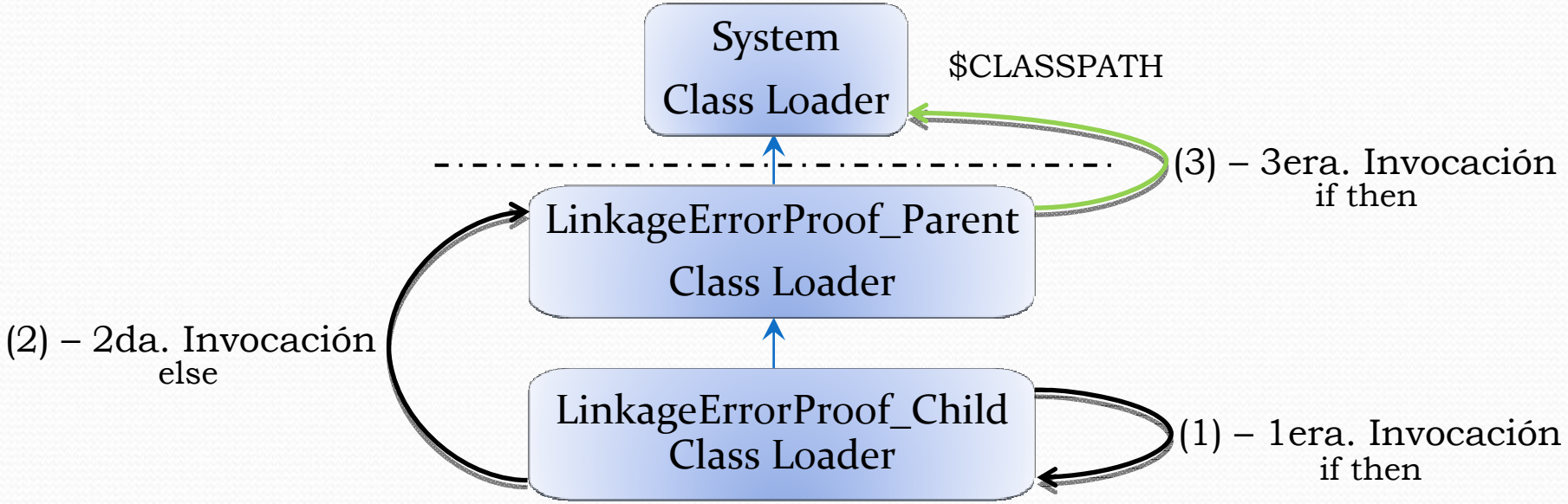
(1) clazz=Class.forName(name,false, this.getParent()); (LinkageErrorProof_Child)
(2) clazz=Class.forName(name,false, this); (LinkageErrorProof_Child)
(3) clazz=Class.forName(name,false, this.getChild()); (LinkageErrorProof_Parent)
(4) Indistinto ya que el error se debería haber producido.



Prueba 4 - LinkageErrorProof

7.

(1) clazz=Class.forName(name,false, this); (LinkageErrorProof_Child)
(2) clazz=Class.forName(name,false, this.getParent()); (LinkageErrorProof_Child)
(3) clazz=Class.forName(name,false,this.getSystemClassLoader());(LinkageErrorProof_Parent)
(4) Indistinto ya que la clase ha sido encontrada.

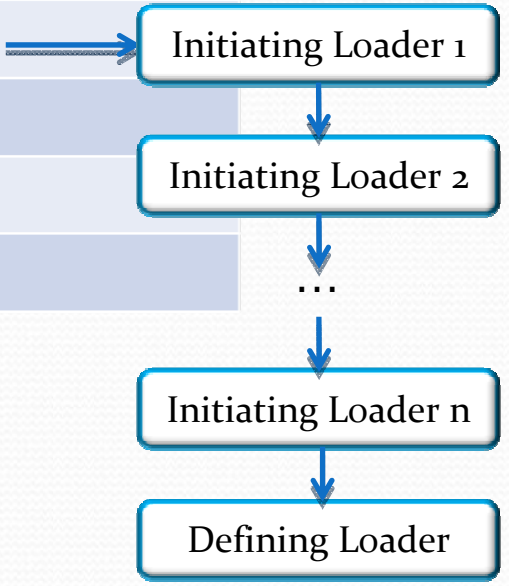


Entendiendo la Excepción

java.lang.LinkageError

Estructura de la Constraint Table

Constraint Table		
Nombre de Clase	Objeto de Clase Resuelto	Lista de ClassLoaders Asociados



Ejecución SimpleAglet

1era. Invocación

forName("java.lang.String",false,AgletClassLoader)

Constraint Table		
Nombre de Clase	Objeto de Clase Resuelto	Lista de ClassLoaders Asociados
java.lang.String	null	→ AgletClassLoader

Ejecución SimpleAglet

2da. Invocación

forName("java.lang.String",false,AgletClassLoader)

Constraint Table		
Nombre de Clase	Objeto de Clase Resuelto	Lista de ClassLoaders Asociados
java.lang.String	null	AgletClassLoader

(1era. Invocación)

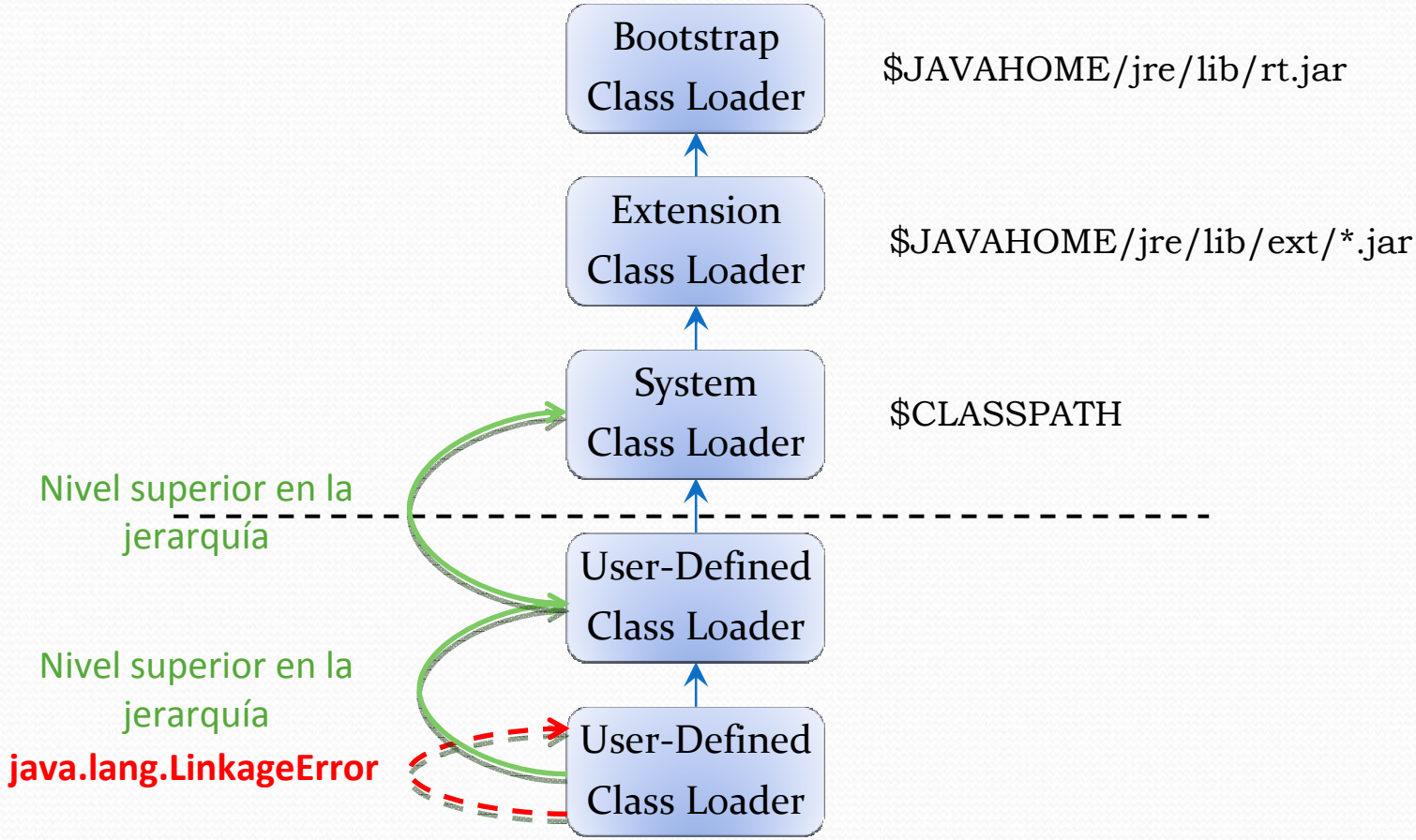
AgletClassLoader

(2da. Invocación)

LinkageError

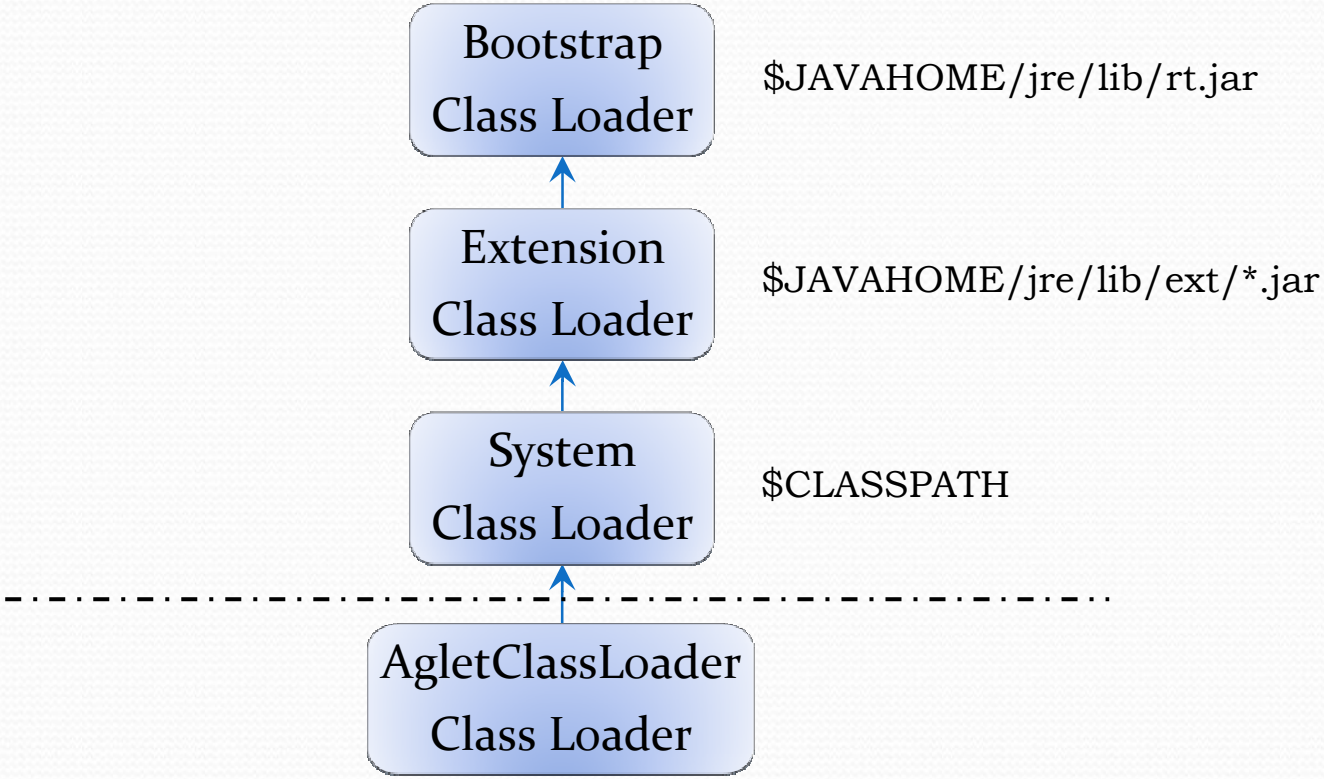
Solución al LinkageError

Correcto uso de forName(String,boolean,ClassLoader)



Solución al LinkageError

Jerarquía de classloaders – Plataforma Aglets



Solución al LinkageError

Invocación correcta del método forName()

```
596:    try {
597:            Class cl = findResolvedClass(name);
598:
599:            if (cl != null) {
600:                log.debug("Using class " + name + " in resolved cache");
601:                return cl;
602:            }

{...}

681:    try {
682:            clazz = Class.forName(name,false, this.getSystemClassLoader());
                                        /* Línea donde difiere JDK 1.5 y JDK 6*/
683:            if (clazz != null)
684:            {
685:                log.debug("Loading " + name + " from System");
686:                return clazz;
687:            }
688:     }
689:     catch (ClassNotFoundException ex) {}
```

Solución al LinkageError

Prueba 2 - SimpleAglet

Consola localhost:4500

[Warning: The hostname seems not having domain name.
Please try -resolve option to resolve the fully qualified hostname
or use -domain option to manually specify the domain name.]
Empieza Run()
String0-0
String1-0
***** Addr: atp://localhost:5000 place:
No integrity check because no security domain is authenticated.

Consola localhost:5000

[Warning: The hostname seems not having domain name.
Please try -resolve option to resolve the fully qualified hostname
or use -domain option to manually specify the domain name.]
Empieza Run()
String0-1
String1-1

Solución al LinkageError

Prueba 1 - ParentAglet

Consola localhost:4500

[Warning: The hostname seems not having domain name.
Please try -resolve option to resolve the fully qualified hostname
or use -domain option to manually specify the domain name.]
***** Addr: atp://localhost:5000 place:
No integrity check because no security domain is authenticated.
Padre Finalizado

Consola localhost:5000

[Warning: The hostname seems not having domain name.
Please try -resolve option to resolve the fully qualified hostname
or use -domain option to manually specify the domain name.]
Trip Finalizado Exitosamente

Índice

VIII. Conclusión