

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 557 109**

51 Int. Cl.:

G06F 11/36 (2006.01)

G06F 11/34 (2006.01)

G06F 9/455 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **19.01.2011 E 11000381 (1)**

97 Fecha y número de publicación de la concesión europea: **30.09.2015 EP 2365434**

54 Título: **Método para acceder a elementos de interfaz de usuario de aplicaciones de terceros en ejecución en dispositivo móvil basado en Symbian**

30 Prioridad:

18.02.2010 IL 20405010

45 Fecha de publicación y mención en BOPI de la traducción de la patente:

22.01.2016

73 Titular/es:

**DEUTSCHE TELEKOM AG (100.0%)
Friedrich-Ebert-Allee 140
53113 Bonn, DE**

72 Inventor/es:

**MIMRAN, DAVID y
SHAPIRA, BRACHA**

74 Agente/Representante:

ISERN JARA, Jorge

Observaciones :

Véase nota informativa (Remarks) en el folleto original publicado por la Oficina Europea de Patentes

ES 2 557 109 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín europeo de patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre concesión de Patentes Europeas).

DESCRIPCIÓN

Método para acceder a elementos de interfaz de usuario de aplicaciones de terceros en ejecución en dispositivo móvil basado en Symbian

5 Campo de la invención

La presente invención se refiere al campo de consultar aplicaciones en ejecución en dispositivos móviles. Más particularmente, la invención se refiere a consultar elementos de datos de aplicaciones actualmente en ejecución en dispositivos que usan un sistema operativo Symbian.

10

Antecedentes de la invención

En los últimos años el número y diversidad de las aplicaciones soportadas mediante los dispositivos digitales ha aumentado significativamente. Además del hambre de los clientes de nuevas aplicaciones estimulantes, los operadores y proveedores de servicios comparten un fuerte deseo de que los clientes aprovechen las aplicaciones que se les proporciona. El beneficio de los operadores y los proveedores de servicio es directamente proporcional a la frecuencia de uso de sus aplicaciones. Por lo tanto, una caída en uso generalmente da como resultado una caída en beneficios.

15

La calidad de una aplicación se determina a menudo mediante el uso de ella del cliente. Típicamente, la calidad de una aplicación se evalúa en términos de cómo de bien está diseñada, si está libre de defectos, y cómo de duradera permanece en uso por los clientes. Aunque estos son componentes críticos para la evaluación de calidad, es importante comprender que si un cliente considera un producto como difícil de usar o no "amigable", probablemente no lo usará lo suficiente para averiguar cómo de bien está diseñado, o explorar todas las opciones que proporciona. Un término que se usa a menudo para describir la calidad de un producto desde la perspectiva del usuario, particularmente con respecto a productos informáticos, es el término "amigable para el usuario".

20

Existen varios métodos para evaluar la calidad de una aplicación. Típicamente estos métodos generan un modelo que describe los objetivos soportados por la aplicación y analiza secuencias de acciones ejecutadas por los usuarios para evaluar la calidad de la interacción usuario-sistema. Cuando el usuario abre una aplicación en un dispositivo digital, normalmente tiene una tarea en mente que desea completar - por ejemplo, enviar un Mensaje Instantáneo o añadir un contacto a la lista de contactos telefónica. Para realizar tales tareas, el usuario pasa interactivamente a través de una serie de pantallas hasta que la tarea se completa. Cada pantalla está compuesta de elementos - algunos de los cuales son Elementos de Entrada de Pantalla (SIE) que se usan para entrada de datos y avanzar al usuario a otras pantallas, mientras otros son Elementos de Pantalla (SE) que se presentan en la pantalla. Todo el grupo de elementos presentaos en el dispositivo del usuario se denominan en lo sucesivo como Elementos de Interfaz de Usuario (UIE).

30

En muchos casos los UIE no están disponibles para una aplicación de terceros. Por lo tanto, no hay manera de recopilar información con respecto a las etapas de usuario y los elementos que se le presentan en la pantalla de dispositivo. Uno de estos casos son dispositivos que operan usando el Sistema Operativo (OS) Symbian. El Symbian OS tiene medidas de control ajustadas que evitan a una aplicación de terceros acceder al espacio de memoria u otras propiedades de una aplicación actualmente en ejecución visible en el Symbian OS. Por lo tanto, las secuencias de usuario y otros datos con respecto a la interacción de usuario-sistema no pueden recopilarse.

40

Es un objeto de la presente invención proporcionar un método para monitorizar y recopilar datos desde aplicaciones en ejecución actualmente visibles en dispositivos móviles basados en Symbian.

50 Es otro objeto de la presente invención permitir a terceros acceder al espacio de proceso en ejecución actualmente visible, en dispositivos móviles basados en Symbian.

Es otro objeto más de la presente invención proporcionar un método para consultar datos de aplicaciones y ventanas en ejecución actualmente visibles en dispositivos móviles basados en Symbian.

55

Otros objetos y ventajas de la invención se harán evidentes a medida que la descripción continúa.

Sumario de la invención

La invención se refiere a un método para monitorizar y acceder a elementos de Interfaz de Usuario de una aplicación de terceros actualmente en ejecución en un dispositivo basado en sistema operativo (OS) Symbian, que comprende:

60

(a) cargar un segmento de código a través de una Biblioteca Dinámicamente Cargada de Procesador Frontal (FEP DLL) en el espacio de proceso de dicha aplicación actualmente en ejecución;

65

(b) ejecutar, por medio de dicho segmento de código, una llamada de Interfaz de Programación de Aplicación (API)

de Symbian para encontrar el manejador de objeto del Elemento de Interfaz de Usuario (UIE) que tiene el foco en la pantalla de dispositivo actualmente visible;

5 (c) atravesar la cadena parental para alcanzar el control parental de nivel superior que es el manejador de ventana superior de la pantalla actualmente visible; y

(d) realizar las etapas (b) y (c) para cada pantalla creada y usar dicho manejador para capturar el manejador de objeto del Elemento de Interfaz de Usuario (UIE) que tiene el foco superior de la pantalla actualmente visible.

10 Se proporcionan adicionalmente realizaciones de la invención en las que el segmento de código se usa para acceder al Elemento de Interfaz de Usuario capturado de la pantalla actualmente visible para posibilitar al menos una de las siguientes etapas:

15 (a) identificar la estructura de pantallas de aplicación de terceros;

(b) modificar pantallas en tiempo de ejecución como parte de una arquitectura de Interfaz de Usuario (UI) flexible;

(c) proporcionar ayuda de contexto; y

20 (d) grabar interacción de usuario con la aplicación actualmente en ejecución.

Breve descripción de los dibujos

En los dibujos:

25 -La Figura 1 es un diagrama de flujo que describe el proceso para recopilar datos desde aplicaciones en ejecución actualmente visibles; y
 - La Figura 2 ilustra esquemáticamente el entorno operativo del proceso.

30 Descripción detallada de la invención

Varios sistemas operativos, tales como Symbian evitan que terceros recopilen información con respecto a los pasos de un usuario y los elementos que se le presentan a través de sus pantallas del dispositivo. La presente invención proporciona medios para monitorizar y recopilar tal información desde aplicaciones que se ejecutan en un dispositivo digital que son actualmente visibles para el usuario. Aprovechando el Procesador Frontal (FEP) disponible en dispositivos basados en Symbian se carga un código en la memoria de la aplicación en ejecución. El código ejecuta una llamada de interfaz de programación de aplicación (API) de Symbian y proporciona acceso al UIE presentado en la pantalla actualmente visible.

40 La presente invención usa el mecanismo FEP existente en el sistema operativo Symbian. El mecanismo FEP está diseñado para permitir la extensión de capacidades de entrada de terceros. Por ejemplo pueden ampliarse capacidades de entrada tales como teclados virtuales o pantallas táctiles para activarse en aplicaciones de primer plano. El FEP es un tipo especial de Biblioteca Dinámicamente Cargada (DLL). Esta biblioteca se carga en el espacio de proceso de cada aplicación que se ejecuta en Symbian con una interfaz de usuario. El FEP se carga una vez por creación de proceso y se llama a una función específica en el FEP durante la creación de cada ventana de aplicación.

50 La Figura 1 es un diagrama de flujo que describe el proceso para recopilar datos desde aplicaciones en ejecución actualmente visibles. En la primera etapa 101 se carga un segmento de código de un tercero en un proceso actualmente en ejecución. El segmento de código se carga a través del mecanismo FEP al subproceso de creación de una nueva ventana. En la etapa 102 el segmento de código espera mientras se construye una nueva ventana en la que se añaden todos los UIE en la pantalla a la pila de control de ventana superior. Para cada nueva pantalla creada en la aplicación objetivo se realiza una llamada de función personalizada FEP (etapa 103). Esta función contiene el código de terceros para tomar control sobre la aplicación objetivo. Como un ejemplo de métodos que proporcionan el control de FEP durante la ejecución de terceros, de acuerdo con una realización la función que se usa para tomar el control sobre la aplicación objetivo es "HandleChangeInFocus". Se llama a este evento cada vez que un elemento de ventana recibe el foco de la entrada de usuario. Como se ha mencionado anteriormente en el presente documento, se llama a esta función únicamente después de que se rellena el control de la pila. Cuando se llama a esta función el código que se llama en la función se usa para tomar el control sobre la aplicación objetivo.

60 En la siguiente etapa 104, se realiza un acceso sencillo a una memoria y a espacio de proceso de la aplicación objetivo. El código ejecutado en la etapa 103 puede obtener el control debido al hecho de que se ejecuta en el espacio de memoria de la aplicación objetivo. Una vez que el código obtiene el control de la aplicación, se ejecuta la etapa 105. En la etapa 105 un manejador de entorno de aplicación objetivo, que es un id único que recibe cada pantalla y proceso en Symbian cuando se crea, se usa para consultar diferentes datos en la aplicación y ventanas en ejecución. El manejador de entorno básico que es accesible se usa para encontrar el contenedor de aplicación

objetivo, que es un objeto asignado para cada aplicación de Symbian específica. El contenedor de aplicación objetivo soporta datos detallados sobre la aplicación objetivo.

5 En la etapa 106 se ejecuta una llamada API de Symbian convencional para encontrar el manejador de objeto del control que tiene el foco superior, es decir, un elemento de ventana tal como entrada, texto, botón, etc., en la pantalla. Una vez que se recibe el manejador de objeto del control que tiene el foco superior en la pantalla, se aprovecha la relación natural entre los controles y ventanas padres en Symbian. Esta relación es una relación padre-hijo que asegura que el control tiene controles padre hasta el padre superior que no tiene un atributo padre. Este control realmente es el manejador de objeto de ventana de nivel superior.

10 Otra manera de describir lo que tiene lugar en la etapa 106 es la siguiente: las pantallas Symbian se construyen como un árbol donde los contenedores de nivel superior son intersecciones de árbol superiores y los elementos internos son cualquiera de una hoja o una rama. Para encontrar la rama de nivel superior del árbol el mecanismo de la invención captura un único elemento en la pantalla y atraviesa el árbol hasta que alcanza la rama de nivel superior disponible.

15 En la etapa 107 se alcanza el manejador de objeto de ventana de nivel superior atravesando la cadena parental desde el control que tiene el foco superior al control padre superior. El control padre superior es el manejador de ventana superior en la pantalla actualmente visible.

20 Desde este punto el segmento de código ejecutado (en la etapa 108) puede acceder al UIE de la pantalla objetivo de un número de maneras que están disponibles para un desarrollador de aplicación como si el segmento de código estuviera escrito por el propio desarrollador. Por ejemplo, las funciones de segmento de código proporcionan acceso al UIE capturado de la pantalla actualmente visible. Tal acceso puede usarse para identificar estructura de pantallas de aplicación de terceros, modificar pantallas en tiempo de ejecución como parte de una arquitectura de UI flexible, que proporciona en su lugar servicios tales como ayuda de contexto, y grabar interacción de usuario con la aplicación actualmente en ejecución.

25 La Figura 2 ilustra esquemáticamente una realización del entorno operativo del proceso. En el principio del proceso Symbian nativo 201, se carga un código FEP 202 desde la DLL de FEP 203 en el espacio de proceso. Tras la creación de cada nueva ventana, el subproceso para crear una nueva ventana 204 llama a las funciones de código desde el código de FEP 202. Una función de código de FEP 205 se carga en el subproceso de creación de una nueva ventana. La función de código de FEP se ejecuta y proporciona acceso a la pila que almacena todos los UIE de la ventana disponible actualmente visible en la pantalla del dispositivo.

30 Aunque se han descrito algunas realizaciones de la invención a modo de ilustración, será evidente que la invención puede llevarse a cabo con muchas modificaciones, variaciones y adaptaciones, y con el uso de numerosos equivalentes o soluciones alternativas que están dentro del alcance de los expertos en la materia, sin alejarse del alcance de las reivindicaciones adjuntas.

40

REIVINDICACIONES

1. Un método para monitorizar y acceder a elementos de Interfaz de Usuario de una aplicación de terceros actualmente en ejecución en un dispositivo basado en Sistema Operativo (OS) Symbian, que comprende:

- 5
- (a) cargar un segmento de código a través de una Biblioteca Dinámicamente Cargada de Procesador Frontal (DLL de FEP) en el espacio de proceso de dicha aplicación actualmente en ejecución;
 - (b) ejecutar, por medio de dicho segmento de código, una llamada de Interfaz de Programación de Aplicación (API) de Symbian para encontrar el manejador de objeto del Elemento de Interfaz de Usuario (UIE) que tiene el foco superior en la pantalla de dispositivo actualmente visible;
 - 10 (c) atravesar la cadena parental para alcanzar el control parental de nivel superior que es el manejador de ventana superior en la pantalla actualmente visible; y
 - (d) realizar las etapas (b) y (c) para cada pantalla creada y usar dicho manejador para capturar el manejador de objeto del Elemento de Interfaz de Usuario (UIE) que tiene el foco superior de la pantalla actualmente visible.
- 15

2. Un método de acuerdo con la reivindicación 1, en el que el segmento de código se usa para acceder al Elemento de Interfaz de Usuario capturado de la pantalla actualmente visible para posibilitar al menos uno de lo siguiente:

- 20
- (a) identificar la estructura de pantallas de aplicación de terceros;
 - (b) modificar pantallas en tiempo de ejecución como parte de una arquitectura de Interfaz de Usuario (UI) flexible;
 - (c) proporcionar ayuda de contexto; y
 - (d) grabar interacción de usuario con la aplicación actualmente en ejecución.

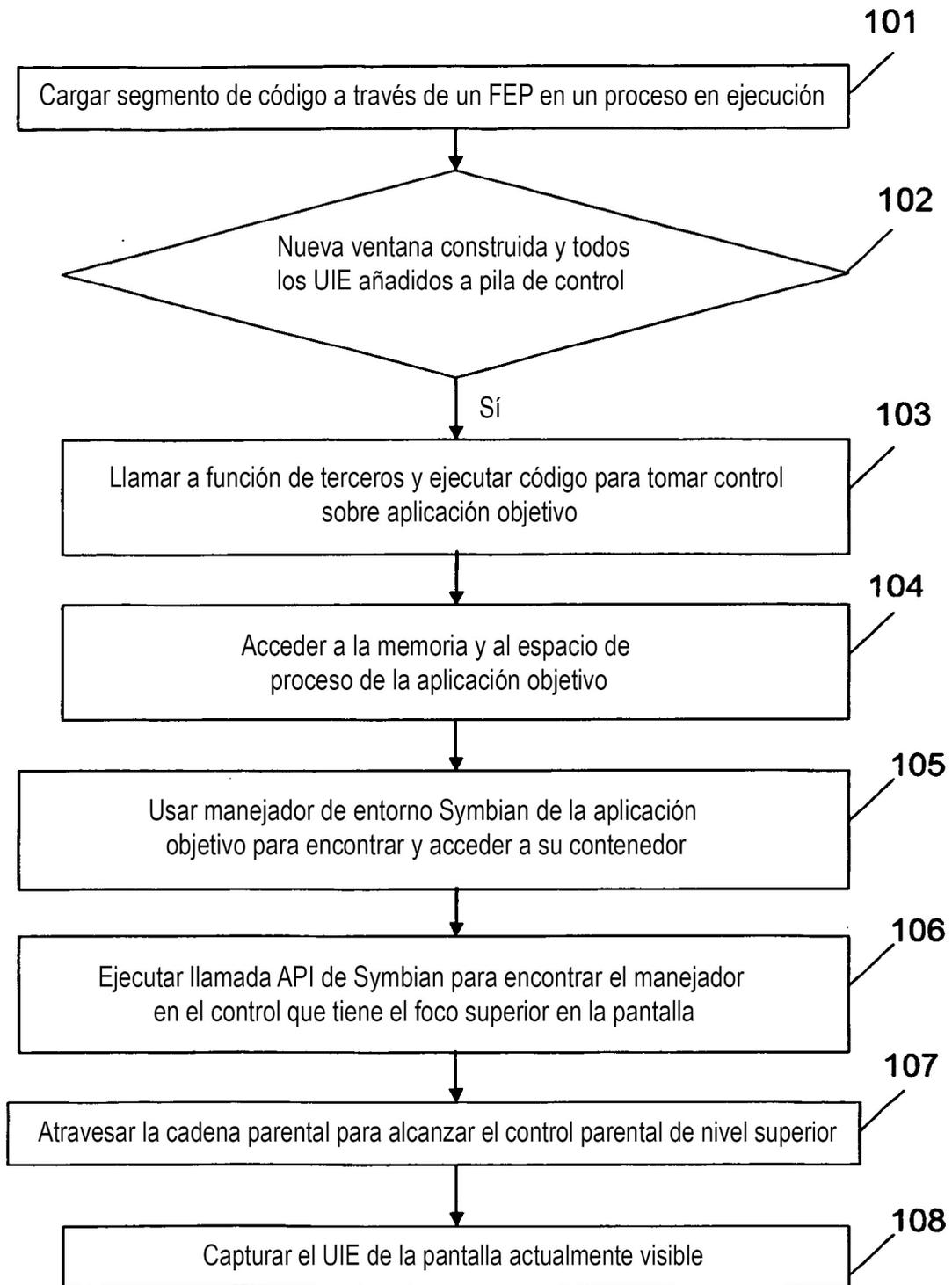


Fig. 1

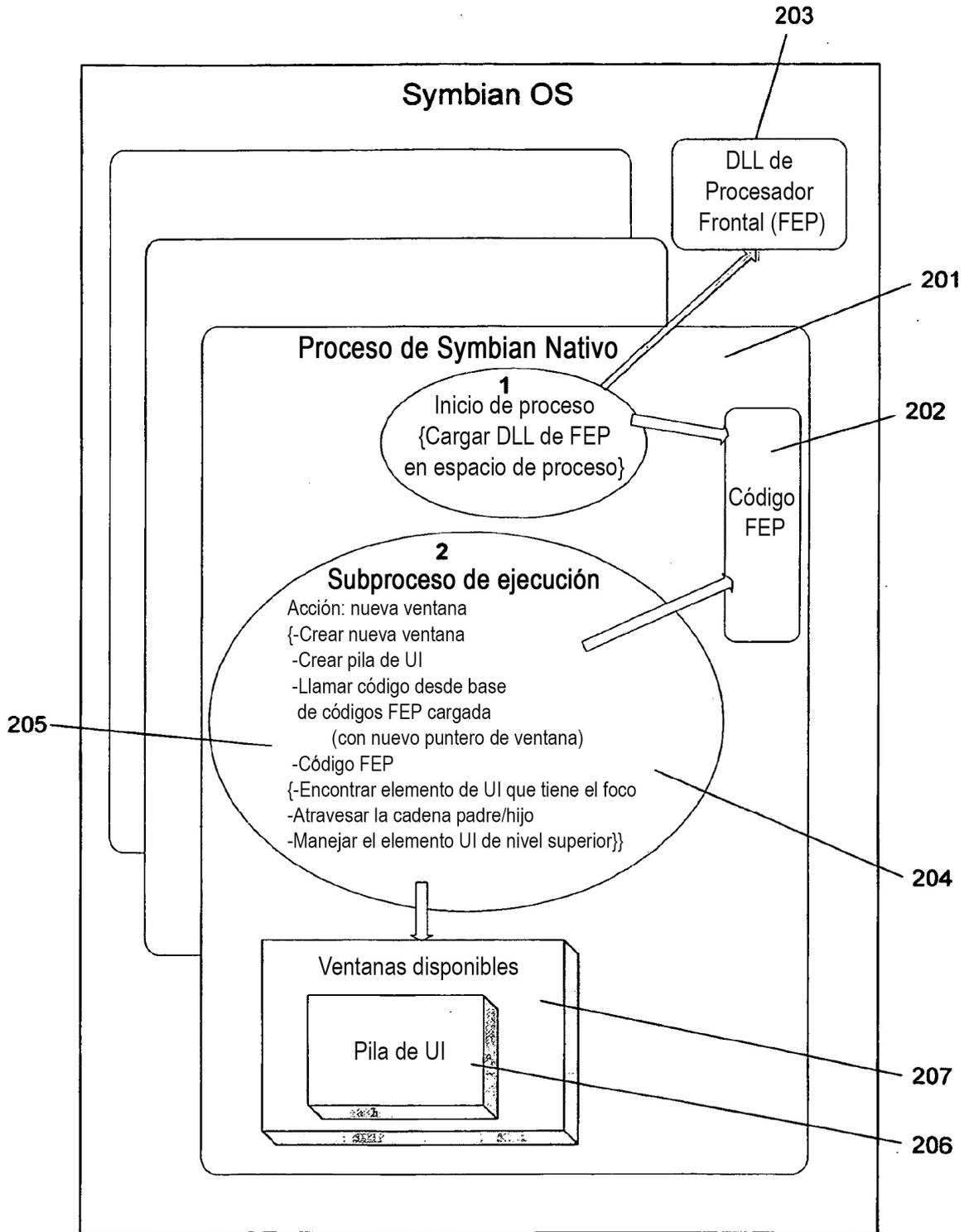


Fig. 2