

Apéndice A

A.1. Extensión de la Herramienta IDA

La herramienta Identifier Analyzer (IDA) tiene una característica extra, puede recibir como parámetro de entrada la ruta de un archivo XML¹ existente en el disco. Este archivo debe contener información asociada a ids, literales y comentarios; similar a la que es provista por el Analizador Sintáctico que IDA posee (Módulo de Extracción de Datos - Ver Capítulo 4). Luego si los datos están correctamente leídos, IDA ejecuta directamente los algoritmos de análisis de ids en base a la información provista por el archivo XML. Por último, los resultados de la ejecución se escriben en otro archivo XML que será creado en la misma ruta que el archivo leído como entrada (Ver Figura A.1).

La ventaja de que IDA soporte interacción con archivos XML, permite un intercambio de datos estándar con otras aplicaciones, lo que conlleva a la compatibilidad entre aplicaciones para compartir información. De esta forma, IDA puede formar parte de un proceso de análisis más extenso que involucre otras herramientas asociadas a la comprensión de sistemas.



Figura A.1: Arquitectura de la Extensión de IDA

¹Extensible Markup Language

Para que IDA procese un archivo XML como entrada, simplemente se realiza a través del archivo JAR¹ correspondiente a la herramienta por medio de la siguiente orden en la línea de comandos del sistema operativo²:

```
java -jar IDA.java <argumento>
```

En `<argumento>` se coloca la ruta donde se encuentra ubicado el archivo XML a procesar (un ejemplo en linux es `/home/entrada.xml`³). Este argumento no es obligatorio, y en caso de no pasarlo, se ejecuta la interfaz normal de IDA que fue descrita en el capítulo 4. A continuación, se describe como debe estar estructurado el archivo XML de entrada.

Archivo XML de Entrada

El archivo xml que se ingresa, el comienzo debe marcarse con `<entrada>` y el fin con `</entrada>`, en su interior puede contener las siguientes listas de elementos propios de una aplicación JAVA:

Lista de Identificadores: Lista de ids que van a ser analizados esta enmarcada por `<lista_ids>` y `</lista_ids>`; cada elemento de esta lista se indica con `<id>` y `</id>`; dentro de cada uno de estos elementos se aprecia el nombre del id `<nombre>nmId</nombre>`, y el número de línea `<linea>45</linea>`.

Lista de Frases: Listado de frases (asociadas a comentarios y literales) el inicio y fin de esta lista se indica con `<lista_frases>` y `</lista_frases>`; cada elemento de esta lista se indica con `<frase>` y `</frase>`; dentro de cada uno de estos elementos de la lista se aprecia la frase correspondiente `<texto>Hello World!</texto>`, y el número de línea `<linea>25</linea>`.

Lista de Clases: Este listado se corresponde a las clases que posee el programa, esta enmarcado con `<lista_clases>` y `</lista_clases>`; cada elemento de este listado se indica con `<clase>` y `</clase>`; dentro de cada

¹JAVA Archive.

²Se recomienda utilizar los sistemas operativos Windows o UNIX

³No es necesario que se llame entrada, pero si que tenga extensión xml.

elemento se halla el nombre de la clase con `<nombre>Person< /nombre>`, el numero de línea donde comienza la clase `<linea_inicio>12< /linea_inicio>`, y la línea donde finaliza la clase `<linea_fin>38< /linea_fin>`.

Lista de Métodos: Similar al listado anterior pero para métodos, esta enmarcado con `<lista_metodos>` y `< /lista_metodos>`; cada elemento de este listado se indica con `<metodo>` y `< /metodo>`; dentro de cada elemento se halla el nombre del método con `<metodo>getPerson< /metodo>`, el numero de línea donde comienza el método `<linea_inicio>20< /linea_inicio>`, y la línea donde finaliza la clase `<linea_fin>25< /linea_fin>`.

Importante:

En los ítems anteriores, el listado de frases y el listado de clases son obligatorios, mientras que el resto colabora con el análisis de los ids. También en los cuatro ítems antedichos, cuando se refieren a número de líneas, hace referencia a la ubicación dentro del código, sin embargo esté dato no es obligatorio dado que están para colaborar con el algoritmo de expansión de abreviaturas.

Un ejemplo del archivo XML de entrada se puede apreciar en la figura A.2. La herramienta IDA procesa el archivo XML ingresado ejecutando los algoritmos de análisis de ids que tiene implementados (Greedy, Samurai y Expansión Básica). Luego de ejecutar estos algoritmos, IDA escribe los resultados de cada ejecución en un nuevo archivo XML de salida que se describe en la próxima sección.

```
<entrada>
  <lista_clases>
    <clase>
      <nombre>Minesweeper</nombre>
      <linea_inicio>7</linea_inicio>
      <linea_fin>202</linea_fin>
    </clase>
  </lista_clases>
  <lista_metodos>
    <metodo>
      <nombre>win</nombre>
      <linea_inicio>152</linea_inicio>
      <linea_fin>159</linea_fin>
    </metodo>
  </lista_metodos>
  <lista_ids>
    <id>
      <nombre>bttns</nombre>
      <linea>10</linea>
    </id>
    <id>
      <nombre>min_mtrx</nombre>
      <linea>12</linea>
    </id>
  </lista_ids>
  <lista_frases>
    <frase>
      <texto>buttons</texto>
      <linea>9</linea>
    </frase>
    <frase>
      <texto>mines matrix</texto>
      <linea>79</linea>
    </frase>
  </lista_frases>
</entrada>
```

Figura A.2: Ejemplo de Archivo XML de entrada.

Archivo XML de Salida

El archivo de salida se crea en la misma ubicación que el archivo XML pasado por entrada (siguiendo con el ejemplo de la sección anterior se creará en /home/salida.xml¹). Este archivo de salida indica el inicio con `<salida>` y el fin con `</salida>`, en su interior posee la siguiente lista:

Lista de Identificadores Analizados: Es el listado de los ids incluyendo el análisis realizado en cada uno, `<lista_analisis_ids>` señala el comienzo a la lista y `</lista_analisis_ids>` indica el fin; cada elemento de la lista se indica con `<id>` y `</id>`; dentro de cada elemento de la lista se encuentra el nombre del id analizado indicado con `<nombre>nmId</nombre>`, la división greedy del id se ubica entre `<div_greedy>nm-id</div_greedy>`, la división samurai del id entre `<div_samurai>nm-id</div_samurai>`, la expansión desde greedy entre `<exp_greedy>name identifier</exp_greedy>`, y la expansión desde samurai entre `<exp_samurai>name identifier</exp_samurai>`.

Un ejemplo del archivo XML de salida con los ids analizados por las distintas técnicas, es el mostrado en la figura A.3.

```
<salida>
  <lista_analisis_ids>
    <id>
      <nombre>nmId</nombre>
      <div_greedy>nm-id</div_greedy>
      <div_samurai>nm-id</div_samurai>
      <exp_greedy>name identifier</exp_greedy>
      <exp_samurai>name identifier</exp_samurai>
    </id>
    <id>
      <nombre>fs</nombre>
      <div_greedy>fs</div_greedy>
      <div_samurai>fs</div_samurai>
      <exp_greedy>file system</exp_greedy>
      <exp_samurai>file system</exp_samurai>
    </id>
  </lista_analisis_ids>
</salida>
```

Figura A.3: Ejemplo de Archivo XML de salida.

¹Si ya existe un archivo con el nombre salida.xml, el mismo se sobrescribirá.