

Documentación del Proyecto de Haskell

Oswaldo Bayona
Rodrigo Castro
Jorge Vergara

18 de enero de 2014

1. El problema del Parsing de XML

Para resolver el problema del Parsing decidimos usar una estructura de datos llamada árbol n-ario, para cargar todos los datos del archivo y aprovechar la jerarquía que el árbol representa; la idea es la siguiente: El árbol se forma de muchos sub-árboles, cada sub-árbol representa una etiqueta del XML en donde su raíz contiene los atributos de dicha etiqueta y los niveles de anidamiento serán sub-ramas del árbol y así sucesivamente hasta que llegar al nivel más profundo que serán las hojas del árbol. Para entender mejor proporcionaré un ejemplo que se puede observar en la figura1.

Ahí se puede observar un fragmento de código XML, con 4 niveles de anidamiento y se puede apreciar cómo se organiza cada dato del texto. El árbol es un tipo de dato implementado en Haskell, se compone de un Nodo y una lista de árboles. En cada nodo almacenamos los datos usando una lista de duplas.

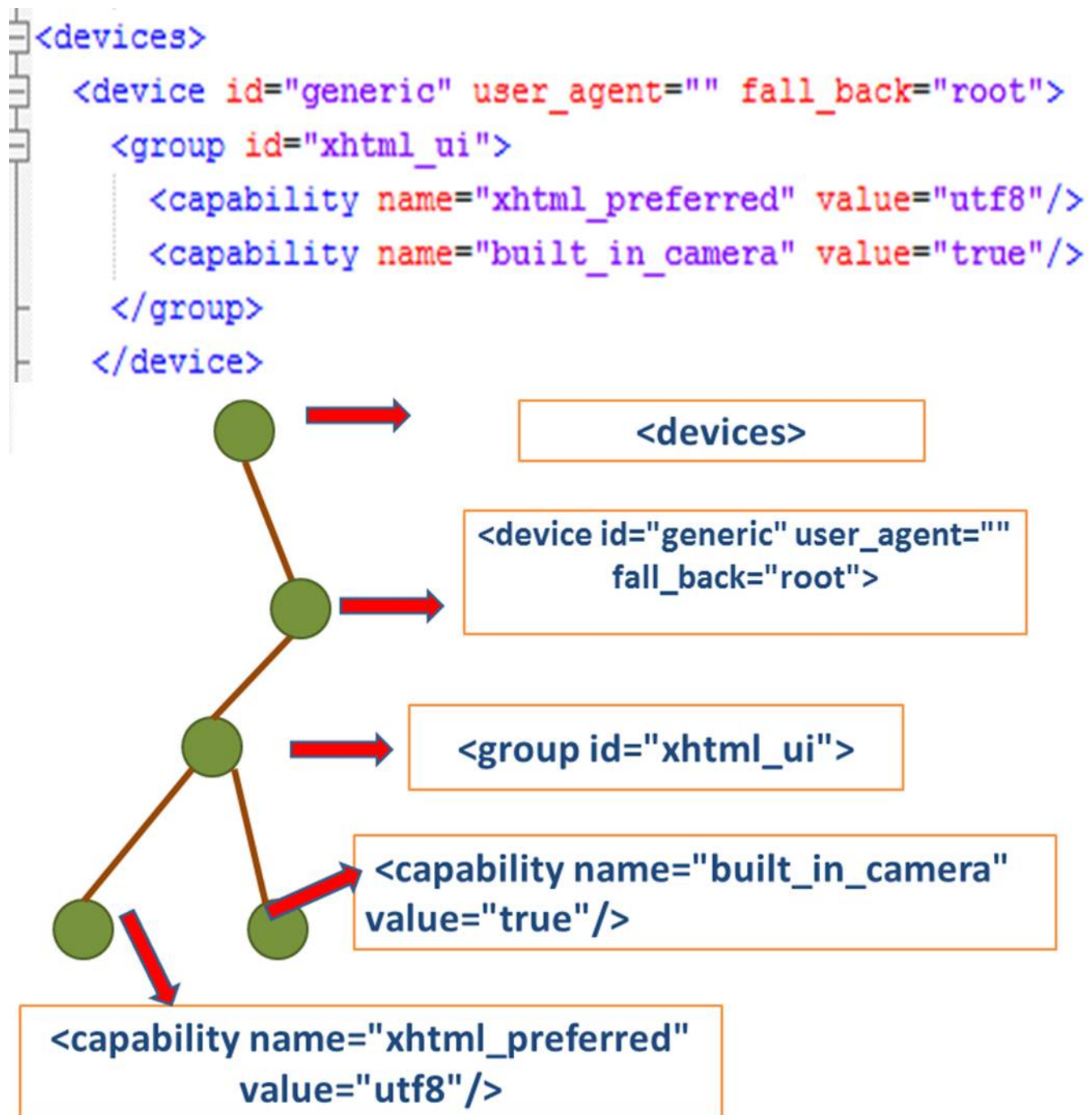


Figura 1

2. Opiniones acerca de Haskell

Oswaldo Bayona: “Haskell me pareció un lenguaje muy bueno para el manejo de listas, debido a que tiene facilidades sintácticas como el ajuste de patrones y las guardas, que mejoran la capacidad de escritura y de lectura. Tiene muchas ventajas como que las funciones no presentan efectos secundarios ya que no existen variables, además la recursión está optimizada, Haskell es perezoso; en mi opinión no haber introducido el concepto de puntero u otra forma de referenciar presentó una dificultad considerable, la forma en la que en Haskell se declaran nuevos tipos de datos tampoco me parece tan buena; pero la experiencia fue muy interesante y eso demuestra que es bueno conocer varios lenguajes para resolver diversos problemas de formas mucho más eficientes y elegantes.”

Jorge Vergara: “Haskell en mi opinion es un buen lenguaje para resolver cierto tipo de problemas, ciertas cosas como hacerlo peroso y la mejora de la recursion lo hacen rapido en ciertos aspectos que otros lenguajes, la utilizacion de la recursion en los problemas hace que el codigo sea legible y elegante, la implementacion de los ajustes de patrones y las guardas hacen que las funciones recursivas sean mucho mas faciles de implementar, el chequeo de tipos es muy riguroso al principio es un poco complicado acostumbrese ha esto pero es muy bueno por que evita errores al momento de la ejecucion, con respecto a la implementacion de tipos me gusto la parte que permite el manejo de errores en un tipo como el tipo Maybe, trabajar con arboles es un poco complicado por que todo en Haskell es constante y para modificar un arbol hay que volver a construirlo todo esto es un poco tedioso, los creadores de Haskell han creado muchas contruucciones para el manejo d ela listas esto me parece muy bueno por que hacen que el lenguaje sea muy facil de utilizar en este aspecto. En conclusion haskell me parece un muy buen lenguaje que permite crear codigo legible y elegante”

Rodrigo Castro: Resolver problemas de forma legible y compacta utilizando recursion es una de las características que mas me gusto de este lenguaje. Me parecio interesante tambien el sistema de inferencia de tipos que realiza Haskell que en algunos ambitos nos brinda un poco mas de flexibilidad como cuando definimos una funcion sin necesidad de declarar los parametros de entrada y la salida pero en otros casos esto puede hacer al lenguaje algo restrictivo, por ejemplo se me presentaron errores de tipo cuando implementaba funciones en donde la definicion de la funcion no se ajustaba a los parametros que Haskell esperaria. Me gustaron muchas de las construcciones que nos brinda Haskell como las guardas y el ajuste de patrones. Tambien me gusto el especial tratamiento que existe sobre las listas, en donde tenemos muchas facilidades para resolver problemas usando este tipo de estructuras como por ejemplo la listas intencionales que nos brinda una gran ventaja para poder definir problemas desde el punto de vista matematico, ademas de las funciones estandares como para filtrar, mapear listas, etc.”