

10.- Líneas de continuación

10.1.- Líneas en el ámbito de la aplicación

10.1.1.- Mayor funcionalidad de la librería

En la implementación de las clases de la librería se ha contemplado únicamente que posean los atributos determinados por los estándares seguidos para su definición, así como una estructura bean. Todas ellas se pueden completar con nuevos métodos como pudiera ser isEmpty() (comprobación de clase vacía) o clear() (eliminación de los contenidos) que dieran mayor funcionalidad a la librería.

Por otro lado, la norma sobre las listas de datos establece el tipo de los elementos internos a un único tipo que depende de la clase en cuestión. En la implementación realizada, cualquier lista se implementa con el tipo **List**, que contiene un **ArrayList** de **Object**, y no se hace ninguna comprobación de que el tipo contenido sea el especificado en la norma. Dicho aspecto se debería mejorar en revisiones posteriores, bien creando métodos de comprobación del tipo, o bien, lo que probablemente sea más factible, estableciendo **List** como un tipo del que heredan implementaciones particulares para cada una de las listas de datos.

Así mismo existe una tercera parte dentro del estándar Health informatics - prEN 14822-2:2003, denominado *Health informatics – General purpose components for messages – Clinical*, que contiene una serie de especificaciones sobre datos clínicos que harían completa la librería de tipos CEN.

10.1.2.- Aumento de las especificaciones de búsqueda

Actualmente la localización de los atributos se realiza a través de un nombre simple que no identifica a qué clase completa se refiere. Es decir, un atributo *name* sabemos que se corresponde a una lista de **EntityName**, pero no se ha especificado si *name* es un atributo de *Person*, *StandarPatientInformation*, o cualquier otra clase que contenga entre sus atributos a *name*: **List<EntityName>**. Esto se mejoraría estableciendo como nombre del trait la cadena XPath que nos lleva a dicho atributo. De éste modo se aumentaría la especificación de la información buscada y podrían mejorarse las condiciones de recuperación de datos.

Por otro lado, parte de la interfaz *ProfileAccess*, y otras interfaces más, no han sido implementadas por el servidor de partida de modo que tampoco se han incluido como interfaces XML en el presente proyecto. Deberían implementarse para un completo desarrollo del PID.

10.2.- Líneas fuera del ámbito de la aplicación

10.2.1.- Servicio web

El objetivo de transformar los datos obtenidos de la base de datos a XML tiene un claro propósito final, que es permitir que la información sea transmitida a través de una red, como pueda ser Internet, permitiendo que ambas partes, emisor y receptor, sean capaces de reconocer los datos enviados. Se trata pues de que esta aplicación sirva de soporte a los servicios web que sería necesario desarrollar para implementar la lógica del sistema de identificación de personas de CORBamed.

De esta manera el diagrama que explicaría el proceso queda como sigue:

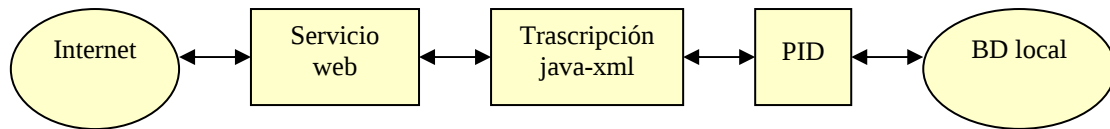


Figura 10.1: Conexión de procesos para acceso a red con BD local con definiciones federadas

10.2.2.- Amplitud de base de datos

Los objetos tratados en el proyecto cumplen el estándar europeo y por tanto se corresponden al modelo de información común a todos los sistemas que vayan a formar una federación. El problema puede estar en que una base de datos local haya sido implementada sin que se sigan las normas mencionadas, y contendrá, por tanto, los datos estructurados de una forma diferente, con diferentes denominaciones e incluso diferente codificación. En este caso sería necesario incluir entre el sistema local y la aplicación presente un sistema que permita la traducción de los datos, tanto sus nombres como sus tipos; se permitiría pasar de los traits locales a los traits federados y posteriormente a documentos XML.

Además para una mayor reutilización del código sería necesario que el elemento encargado de las transformaciones de federado a local y viceversa, al cual llamaremos ProxyPID, sea independiente de la base de datos local y use para explorarla una ontología que contenga las relaciones entre los traits locales y los federados. De éste modo el proceso a seguir sería el siguiente:

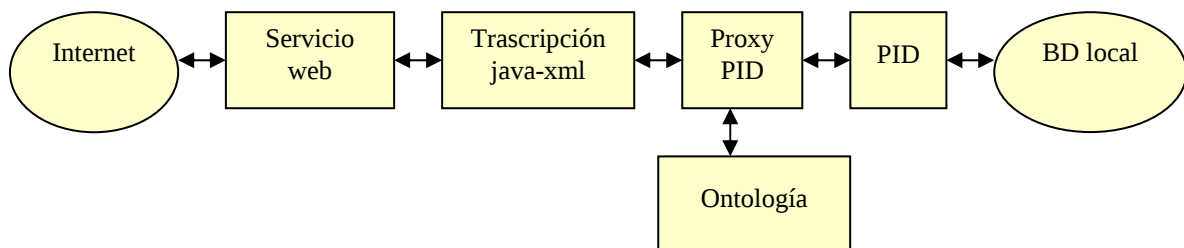


Figura 10.2: Conexión de procesos para acceso a BD local con definiciones locales