

El sistema de gestión de traducciones SareBi

Fernando Quintana Hernández

Índice general

1. Introducción	7
2. Los metadatos de TEI	9
3. La arquitectura de SARE-Bi	13
4. Recuperación de objetos DeliTei	17
4.1. Qué es un ZCatalog	17
4.2. Filtrado de documentos	17
4.3. Búsqueda en segmentos	19
5. Control de versiones mediante XLIFF	21
6. Consideraciones finales	25

Índice de figuras

2.1. Fichero en formato TEI.	10
2.2. Tipología de la función comunicativa inquirir.	11
3.1. Diagrama de clases del producto TeiCorpus	14
3.2. Diagrama de la clase DeliCorpus	15
4.1. Criterios de selección	18
4.2. Resultados de la búsqueda	19
4.3. Resultados de una búsqueda de texto libre	19
5.1. Código Tei	22
5.2. Código XLIFF	23

Capítulo 1

Introducción

Uno de los inconvenientes más severos para incorporar la traducción automática a un entorno de publicación multilingüe es su incapacidad para detectar los factores pragmáticos que condicionan el tratamiento de los contenidos.

La mayor parte de los sistemas están diseñados como procesadores descontextualizados, insensibles a si el texto forma parte de una referencia bibliográfica, el nombre de una persona, una instrucción de programa, o una cita textual. En algunos casos, se preve la posibilidad de marcar parte del texto como no traducible y la mayoría toman en consideración el contenido temático, con el que se origina un cambio en la selección del diccionario de especialidad; pero poco más.

En marcado contraste con esta práctica, desde hace unos años existe la tendencia a utilizar la noción de metadato para agilizar el procesamiento de información en multitud de tareas (Kashyap y Sheth, 1998). Con la generalización de Internet y los problemas derivados de la publicación de contenidos multilingües, la noción de metadato asociada a la gestión de contenidos ha adquirido especial relevancia (Weibel, 1995; Decker et al., 2000).

El sistema Sare-Bi presenta un nuevo modelo de gestión de traducciones que tiene como principal característica la utilización de metadatos en todas las etapas del ciclo de vida documental. El aspecto más destacable es la estructuración de contenidos y las funciones de consulta y recuperación de documentos en un sistema de publicación en web (Zope Community, 2003) en el que se ha incorporado una estructura de metadatos basada en la propuesta TEI (TEI Consortium, 2003).

Por otra parte, la idea de que la traducción no es un eslabón aislado en la cadena documental, sino que forma parte de un amplio proceso, ha sido uno de los pilares de la recién desarrollada tecnología de localización. La localización se ocupa, en general, de la traducción y adaptación de contenidos en soporte electrónico y, en particular, de la creación y mantenimiento de webs multilingües (Localization Industry Standards Association, 2003; Esselink,

2000).

Entre las fases que suelen contemplarse están la revisión de los textos, las pruebas de validación de productos, las actualizaciones, así como las tareas de coordinación y dirección de proyectos. El estándar XLIFF es una propuesta reciente que tiene como objetivo permitir que todos los materiales utilizados en la elaboración de un producto que va a ser multilingüe puedan transportarse sin problemas de una fase a otra, a través de herramientas y plataformas distintas, sin que se produzca ninguna pérdida de información (Savourel, 2001). Por este motivo, XLIFF representa un excelente complemento a la gestión de contenidos basada en TEI.

En el capítulo 2 se expone la estrategia de catalogación del corpus de traducciones, cuyo esquema de descripción de contenidos está basado en la propuesta de cabecera de TEI. A continuación se introducen los aspectos más relevantes de la arquitectura del gestor, que debe la mayor parte de los rasgos de operatividad a las particularidades de diseño de Zope. En el capítulo 4 se muestran las funciones de actualización, filtrado y búsqueda de contenidos en la base documental, que está concebida como una base de datos orientada a objetos. De esta forma se comprobará la idoneidad de Zope como gestor de contenidos multilingües. Un aspecto que el modelo de metadatos basado en TEI todavía no resuelve adecuadamente es el control de versiones; por ello se analiza en el capítulo 5 la incorporación al sistema de un nuevo nivel de descripción basado en XLIFF.

Capítulo 2

Los metadatos de TEI

El estándar para la anotación estructural de textos TEI posee como principal virtud un diseño de cabecera `<teiHeader>` con una gran capacidad para recoger información con valor documental, lo que le asemeja a los sistemas de catalogación de objetos bibliográficos (Caplan, 2001; Wittenburg y Broeder, 2002). Una cabecera TEI tiene cuatro partes:

1. la descripción del archivo `<fileDesc>`, que contiene una descripción bibliográfica completa que permita su citación;
2. una descripción sobre la codificación `<encodingDesc>`, en la que se precisan las incidencias en el momento de la transcripción;
3. perfil del texto `<profileDesc>` que resuelve aspectos contextuales (fecha de creación, categorización, etc.);
4. historial de revisiones `<revisionDesc>` que permite llevar un registro de los cambios realizados sobre la versión electrónica.

Por este motivo se puede decir que TEI aporta un metalenguaje sumamente adecuado para describir documentalmente un corpus textual. La cabecera de un documento en formato TEI se muestra en la figura 2.1.

Hay que señalar que la información ofrecida en la cabecera `<teiHeader>` se complementa con la aportada en la sección `<front>` del propio texto, en la que se recogen datos explícitamente contenidos en el documento, normalmente en la parte superior de la versión impresa. En este caso, los nuevos datos recogidos son la fecha original del documento `<docDate>`, el nombre de la persona que lo firma `<docAuthor>` (que en nuestro corpus es reemplazado por el cargo del firmante) que por defecto coincide con el `<titleStmt:author>`, el lugar en el que se firma y la filiación. Los códigos 1400 y 1406 de los atributos `DepSup` y `DepInf` identifican respectivamente a la Facultad de Filosofía y Letras y al Departamento de Filología Inglesa. En la actualidad existen 93 códigos de centros (facultades, institutos, departamentos, órganos de gobierno, servicios, etc.).

```

<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE TEI.2 SYSTEM "teixlite.dtd">
<TEI.2>
  <teiHeader>
    <fileDesc>

      <titleStmt>
        <title>Reunión de Departamento</title>
        <author>Director de Departamento de Filología Inglesa</author>
        <funder>Proyecto PI1999-72, Gobierno Vasco</funder>
        <principal>Proyecto XML-Bi; grupo DELi</principal>
      </titleStmt>

      <publicationStmt>
        <publisher>Universidad de Deusto</publisher>
        <availability><p>Uso interno. Universidad de Deusto</p></availability>
      </publicationStmt>

      <notesStmt>
        <note type="validated" resp="REDACTOR">No</note>
      </notesStmt>

    </fileDesc>

    <encodingDesc>
      <projectDesc>XML-Bi</projectDesc>
      <classDecl>
        <taxonomy id="DELi_SareBi_Tipologia_4">
          <bibl>http://www.deli.deusto.es/AboutUs/Projects/XML-Bi/Admin/XML-Bi_I2_an1_tipologia4.txt</bibl>
        </taxonomy>
      </classDecl>
    </encodingDesc>

    <profileDesc>
      <creation>
        <date>2003/04/14 16:40:49.366 GMT+2</date>
      </creation>
      <langUsage> <language id="es">espa` nol</language> </langUsage>
      <textClass>
        <classCode scheme="DELi_Tipologia_4">21504</classCode>
      </textClass>
    </profileDesc>

  </teiHeader>

  <text>
    <front>
      <docDate>1999/06/23</docDate>
      <docAuthor>Director de Departamento de Filología Inglesa</docAuthor>
      <head type="place">Bilbao</head>
      <head type="DepSup">1400</head>
      <head type="DepInf">1406</head>
    </front>
    <body> ... </body>
  </text>
</TEI.2>

```

Figura 2.1: Fichero en formato TEI.

```
31000/inquirir
  31400/instancia
    31401/inscripción pruebas mayores 25 años
    31402/solicitud de adaptación de planes de estudio
    31403/solicitud de convalidación asignaturas
    31404/solicitud de reconocimiento complementos
    31405/solicitud de reconsideración admisión
    31406/solicitud de título
    31407/solicitud de traslado expediente
    31408/solicitud cambio de asignaturas optativas y libre elección
```

Figura 2.2: Tipología de la función comunicativa inquirir.

Un aspecto destacable de la cabecera es la taxonomía de categorías documentales. En la versión actual, que es la cuarta, se han contemplado tres niveles de categorización atendiendo a los criterios de función comunicativa, género y tema. Estos criterios están basados en la propuesta de clasificación tipológica de (Trosborg, 1997). En esta versión existen 282 categorías, divididas de manera jerárquica entre tres funciones comunicativas (reglamentar, informar e inquirir), 25 géneros y 256 temas.

La tipología fue uno de los desarrollos del proyecto XML-Bi (Abaitua et al., 2001) y en su elaboración desempeñaron un destacado papel el equipo de traductoras de la Universidad. Un ejemplo del género instancia dentro de la función comunicativa inquirir se recoge en la figura 2.2.

Tanto la tipología textual, como la relación de centros, son listas abiertas que deben actualizarse periódicamente. La asignación de categorías textuales así como de códigos de centro es una de las principales tareas que facilita el gestor documental desarrollado en Zope.

Capítulo 3

La arquitectura de SARE-Bi

SARE-Bi surgió de la necesidad de disponer de un mecanismo de gestión eficaz del corpus de traducciones. La elección de un servidor de publicaciones para web como Zope, frente a otras alternativas como intervenir directamente sobre el código en XML o utilizar paquetes de gestión documental, radica sobre todo en las ventajosas prestaciones que este sistema ofrece para la gestión de contenidos en general, y multilingües en particular, mediante el módulo Localizer (J. David Ibáñez Palomar, 2003). Es además software libre.

Zope está desarrollado casi en su totalidad en Python y su arquitectura incluye una base de datos orientada a objetos ZODB (Zope Object Data Base) como almacén de información. Todos los elementos de una aplicación Zope se elaboran como objetos Python de distintas clases para su publicación en la web (en cualquiera de los formatos HTML, XHTML, XML, u otro). El acceso a estos objetos se realiza de manera habitual por medio del protocolo HTTP (aunque FTP, WebDAV y XML-RPC también están disponibles). Una consulta HTTP será atendida por el objeto de publicación `ZPublisher`, que determinará el método que debe invocarse y los parámetros que le corresponden. Para ello, `ZPublisher` recuperará los objetos de la base de datos ZODB.

Todos los módulos que amplían Zope se denominan productos (por ejemplo, el antes mencionado Localizer). Para crear un nuevo producto se diseñan nuevas clases (`ZClass`) que heredan de clases predefinidas en Zope según la funcionalidad que se les quiera otorgar.

El producto `TeiCorpus` es el motor del sistema SARE-Bi, basado en tres clases (`Delitei`, `Delilang` y `Deliseg`) que heredan de las siguientes clases predefinidas:

1. `ZObject`: clase base de la que heredan todas las clases `ZClass`.
2. `ZCatalogAware`: permite que los objetos se creen, borren y se actualicen automáticamente en un `ZCatalog` (elemento adicional de Zope que permite la indización de información, así como las subsiguientes búsquedas, filtrados, etc, sobre la misma).

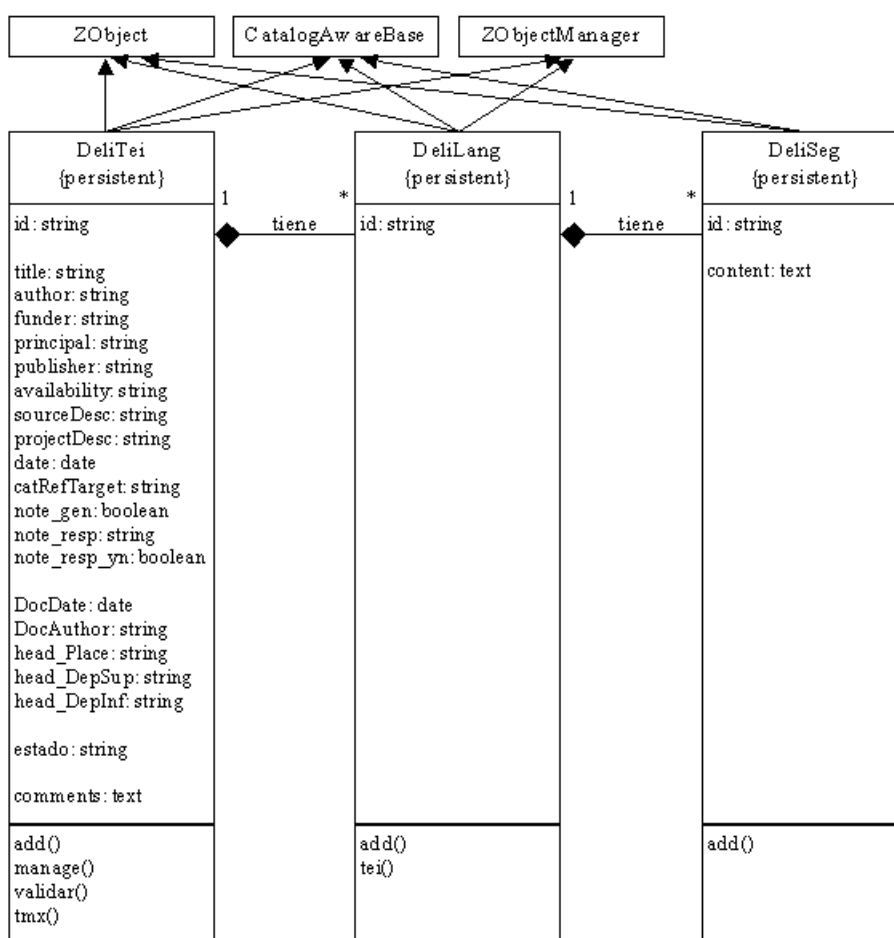


Figura 3.1: Diagrama de clases del producto TeiCorpus

3. **ZObjectManager**: otorga a un objeto las propiedades necesarias para contener otros objetos.

El diagrama de clases en UML (Booch, Rumbaugh, y Jacobson, 1999) del producto **TeiCorpus** se recoge en la figura 3.1.

Las tres clases (**DeliTei**, **DeliLang** y **DeliSeg**) no tienen relación jerárquica (herencia) entre sí, sino de composición (relación “todo/parte” o “tiene”): un **DeliTei** (o documento multilingüe) tiene varios **DeliLang** (subdocumento en una lengua), cada uno de los cuales tiene varios **DeliSeg** (segmentos, en nuestro caso, párrafos, como se sabe).

Los atributos de la clase **DeliTei** sirven para aportar los metadatos al documento multilingüe, y reproducen el esquema de etiquetado TEI recogido anteriormente en la sección 2. Para la funcionalidad del sistema, los atributos más relevantes son el título, la fecha, la clasificación según la tipología documental, el centro y los que conforman el registro de versiones

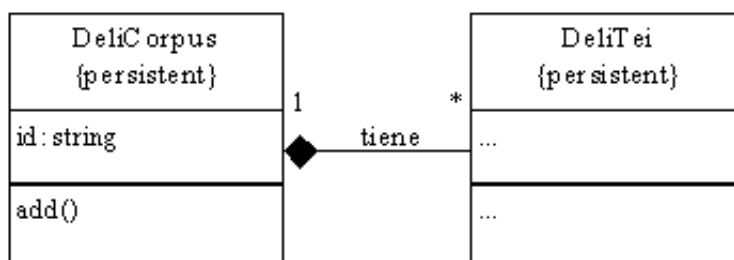


Figura 3.2: Diagrama de la clase `DeliCorpus`

del documento. Las operaciones (o métodos) de la clase `DeliTei` son las habituales: un constructor (`add`), un modificador general (`manage`) y un modificador específico para el control de validación (`validar`). Por motivos de seguridad, de momento no se permite acceso de usuario a un posible método destructor (debe hacerse en todo caso desde la administración del sistema Zope). Finalmente, se incorpora el método `tmx`, que exporta el documento multilingüe al formato TMX de memorias de traducción.

Las otras dos clases del sistema, `DeliLang` y `DeliSeg`, son comparativamente mucho más simples. `DeliLang` no necesita atributos (salvo la clave que identifica la lengua del subdocumento) y, aparte del método constructor, solo se habilita la exportación al formato TEI (que conceptualmente corresponde a esta clase, ya que TEI sirve en principio para codificar documentos en una lengua y, a pesar de que accede a los atributos del objeto contenedor, que es de la clase `DeliTei`). Por su parte, la clase `DeliSeg` incorpora un único atributo (`content`), que es el que almacena el contenido textual de cada segmento.

Aunque no está reflejado en el diagrama de clases de la figura 3.1, se puede decir que existe una clase contenedora, `DeliCorpus`, que puede contener varios documentos multilingües (de la clase `DeliTei`), tal como se refleja en la figura 3.2. En tal caso, se puede decir que el sistema SARE-Bi es un conjunto de objetos de la clase `DeliCorpus`. Esta separación de la base documental en distintos corpora no sería a priori estrictamente necesaria, pero posibilita un grado adicional de organización estructural, que además de su claro propósito, ha facilitado enormemente la transición desde la fase de desarrollo del sistema (en que se crearon pequeños corpora de prueba) a la fase de explotación definitiva.

Capítulo 4

Recuperación de objetos DeliTei

4.1. Qué es un ZCatalog

El `ZCatalog` es un objeto Zope que proporciona indización y búsquedas en una base de datos Zope. Las estructuras de datos utilizadas por los índices proporcionan búsquedas rápidas sin consumir mucha memoria. `ZCatalog` soporta términos de búsqueda booleana, búsquedas por proximidad, sinónimos y “stopwords”

Las tareas principales del `ZCatalog` son dos: indizar información y realizar búsquedas, siendo la primera la más costosa. Estas tareas se pueden llevar a cabo de dos maneras:

1. Si un objeto pertenece a la clase `ZCatalogAware`, automáticamente actualiza el índice cuando es añadido, editado o directamente borrado.
2. Otros objetos pueden indizarse manualmente recorriendo la base de datos en busca de los que cumplan un determinado criterio, registrándolos en el catálogo.

Existen varios tipos de índices. Los `FieldIndex` tratan los datos de forma atómica. Todo el contenido de una propiedad indizada por un `FieldIndex` es tratada como una unidad. Con un índice `TextIndex`, una propiedad es dividida en palabras que son indizadas individualmente. Este índice también se le conoce como índice full-text.

4.2. Filtrado de documentos

El filtrado de documentos es conceptualmente equivalente a una consulta (*query*) a la base de datos, con la salvedad de que se utilizan en todo caso

algunos atributos de la clase `Delitei` (explícitos o implícitos) como criterios de selección, y no es posible concretamente usar el contenido textual de los segmentos (clase `Deliseg`) como criterio adicional (aunque para una próxima revisión del sistema SARE-Bi, se está considerando esta posibilidad). Mediante un formulario se establecen los criterios de selección, que se recogen gráficamente en la figura 4.1.

Por medio de filtros

val:

categoría:

centro:

corpus:

ordenar por:

inverso:

Figura 4.1: Criterios de selección

Los tres primeros criterios de selección (`val`, `categoría` y `centro`) se refieren a los atributos de los objetos de la clase `Delitei`, mientras que el cuarto criterio (`corpus`) se refiere precisamente al grado adicional de estructura que ofrece la separación de la base documental en un conjunto de corpora individuales. Las dos últimas opciones no son criterios de selección, sino de presentación de resultados: pueden ordenarse por `centro`, `categoría`, fecha o `corpus` al que pertenecen, y tanto en orden normal (ascendente) como inverso.

Los resultados se muestran en una tabla, como se ve en la figura 4.2 desde la cual el usuario puede acceder a las operaciones de visualización y modificación de cada documento.

Técnicamente, la implementación ha consistido en configurar el componente `ZCatalog` de SARE-Bi de manera que exista un `FieldIndex` por cada atributo de la clase `Delitei` que puede ser usado en la consulta.

Resultados de la búsqueda

Elementos encontrados: 4

Nº	valor	título	tamaño	lenguas	categoría	centro	corpus	actualizado	fecha doc.	
1	borrador	Festival audiovisual	13	es eu	invitación	ConsejoGob	XML-Bi02	2003/02/27	2002/06/29	Editar
2	borrador	Inauguración exposición	7	es eu	invitación	ConsejoGob	TMMtore	2003/04/01	2001/07/28	Editar
3	completo	Invitación a conferencia	6	es eu	invitación	ConsejoGob	XML-Bi02	2003/03/31	2001/10/23	Editar
4	validado	Decreto de constitución de centro	20	es eu	23700	ConsejoGob	Corpus-2003	2003/04/11	2003/04/08	Editar
actualizar										

Figura 4.2: Resultados de la búsqueda

Resultados de la búsqueda en segmentos

Con la búsqueda de libro, se han encontrado 3.

1 - Reclamanto sobre fotocopias

es 813 SÓLO SE PODRÁ REPRODUCIR UN 5% DEL TOTAL DE LAS PÁGINAS DEL LIBRO

2 - Reclamanto sobre fotocopias

es 806 Título del Libro o Revista:

3 - Invitación a presentación de libro

es 001 El Instituto de Derechos Humanos Pedro Arrupe de la Universidad de Deusto le invita a la presentación del libro "El caso Awas Tingni contra Nicaragua: nuevos horizontes para los derechos humanos de los pueblos indígenas" que tendrá lugar el próximo martes 6 de Mayo en la Sala de Conferencias de la Universidad de Deusto a las 7 de la tarde y contará con la presencia de James Anaya, catedrático de Derecho Internacional de la Universidad de Arizona y asesor legal de la comunidad Awas Tigni, y de Mikel Berraondo, investigador del Instituto de Derechos Humanos Pedro Arrupe.

eu 001 Deustuko Unibertsitateko Pedro Arrupe Giza Eskubideen Institutuak "El caso Awas Tingni contra Nicaragua: nuevos horizontes para los derechos humanos de los pueblos indígenas" liburuaren aurkezpena gonbidatzen zaitu. Ekitaldia maiatzaren 6an, asteartean, izango da arratsaldeko 7etan Deustuko Unibertsitateko Hitzaldi Aretoan eta James Anaya, Arizonako Unibertsitateko Nazioarteko Zuzenbideko katedraduna eta Awas Tigni komunitatearen lege aholkularia, eta Mikel Berraondo, Pedro Arrupe Giza Eskubideen Institutuko ikertzailea izango dira bertan.

Figura 4.3: Resultados de una búsqueda de texto libre

4.3. Búsqueda en segmentos

La otra operación de selección posible en el sistema SARE-Bi es la búsqueda de cadenas de texto en los segmentos de los documentos (es decir, en los valores de los atributos `content` de los objetos de clase `DelSeg`). La única posibilidad adicional es que se puede especificar la lengua de los documentos a revisar.

En cuanto al texto de búsqueda, hay que indicar que en esta primera versión de SARE-Bi las posibilidades son bastante limitadas, pues solo se admiten cadenas literales. Para futuras ampliaciones del sistema, este aspecto constituye una de las mejoras con mayor prioridad. Los resultados se muestran en una lista, que permite como en el filtrado acceder a cada documento que contiene el texto buscado tal como se observa en la figura 4.3.

Técnicamente, el contenido de los segmentos (el atributo `content` de los objetos `DelSeg`) se cataloga en el `ZCatalog` bajo un `TextIndex`, de manera que cada palabra del documento resulta indizada.

Capítulo 5

Control de versiones mediante XLIFF

En la versión actualmente operativa de SARE-Bi se contemplan cinco fases en el proceso de publicación de los documentos. Las tres primeras representan estados escalonados y necesarios, es decir, describen la evolución natural de los contenidos y se aplican a todos los documentos introducidos en el gestor. En cambio, las dos segundas son selectivas y opcionales. Las tres fases primeras representan los estados de borrador , completo y validado. Los estados opcionales son el normativo y el público.

Un documento de borrador es el que contiene habitualmente una primera versión del texto, a veces incompleta, y generalmente sin traducir. El estado completo debe incluir la traducción y todos los datos necesarios para alcanzar la siguiente fase de revisión. Una vez revisado, el documento se cataloga como validado. Se considera que un documento validado cumple todos los requisitos de calidad para su difusión interna. El estatus normativo se reserva para aquellos documentos que se desean proponer como modelo: formularios, fichas, instancias, certificados y otros géneros afines. Sólo alcanzan la categoría de público aquellos documentos que no contienen información privada o de uso exclusivamente interno. Los documentos públicos son los únicos accesibles en la versión de SARE-Bi abierta a Internet.

Directamente relacionado con el control de versiones son los roles de usuario del sistema. Se han definido cuatro roles: invitado , redactor , validador y administrador , que re ejan una lógica progresión de los permisos de acceso al sistema. Los usuarios invitados sólo pueden acceder en modo lectura a los documentos con estado público. Los redactores tienen acceso a todos los documentos en modo lectura, pero en modo escritura sólo a los que ellos mismos generan (en estado de borrador). Los validadores tienen acceso de lectura y escritura para todos los documentos, pero con limitaciones en la gestión general de archivos y carpetas, cuyo acceso completo se concede únicamente al rol administrador.

```

<revisionDesc>
<change id="complete"><date value="2003/03/12 11:30:23.243 GMT+2"></date>
  <respStmt><name key="decanato.fil">Decano de Letras</name>
  <resp>REDACTOR</resp></respStmt>
</item></item>
</change>
<change id="validated"><date value="2003/03/13 10:22:19.118 GMT+2"></date>
  <respStmt><name key="sara">Euskal Irakasletoa</name>
  <resp>VALIDADOR</resp></respStmt>
</item></item>
</change>
</revisionDesc>

```

Figura 5.1: Código Tei

La figura 5.1 ilustra el código TEI correspondiente al control de versiones. El elemento que se encarga en el `<teiHeader>` de recoger esta información es `<revisionDesc>`. Como se ve en la figura, este elemento contiene atributos para la identificación del cambio (de estado del documento), la fecha, el autor (para el estado borrador por defecto se atribuye el mismo valor que el de `<docAuthor>`), así como el rol del responsable del cambio (por defecto validador para todos los estados salvo para los de borrador y completo).

Este mecanismo de control de versiones no es completamente satisfactorio. El principal inconveniente de los modelos de metadatos TEI para un sistema de gestión de traducciones es que están pensados para la descripción de documentos monolingües.

Por otro lado, hay otras cuestiones que tienen que ver con otros procesos más complejos de localización (como son el control de formatos, de rutinas de programación, de objetos en código binario, etc.) que serían imposibles de abordar en TEI, aunque no son relevantes en nuestro caso ya que, por el momento, SARE-Bi sólo trata con documentos de texto simple.

En cualquier caso, la solución a la mayor parte de estos problemas, y concretamente al control de versiones de documentos multilingües, la aporta la propuesta XLIFF. En la figura 5.2 aparece un ejemplo en XLIFF, equivalente al fragmento de TEI, con textos originales y traducciones.

```

<header>
<phase-group>
<phase phase-name="complete"
process-name="publication">
contact-email="decanato@fil.deusto.es"
date="2003/03/12 11:30:23.243 GMT+2"/>
<phase phase-name="validated"
process-name="publication">
contact-email="sara@irakaslego.deusto.es"
date="2003/03/13 10:22:19.118 GMT+2"/>
</phase-group>
</header>
...
<trans-unit id="3">
<source xml:lang="es">
  Que este Departamento procurará que la investigadora encuentre un entorno
  de trabajo adecuado y que pondrá a su disposición los medios necesarios
  para garantizar una labor formativa óptima.
</source>
<target xml:lang="eu">
  Sail honek ahalegina egingo duela ikertzaileak behar dituen lan-giro eta
  baliabideak izan ditzan bere prestakuntza aurrera eramateko.
</target>
<alt-trans>
<source xml:lang="es" phase-name="complete">
  Que este Departamento procurará que el investigador encuentre el clima
  y los medios que precise para llevar adelante su labor formativa como
  investigador.
</source>
<target xml:lang="eu" phase-name="complete">
  Sail hau saiatuko da ikertzaileak bere prestakuntza lana burutzeko
  behar dituen giro eta baliabideak ematen.
</target>
</alt-trans>
<alt-trans>
<source xml:lang="eu" phase-name="validated">
  Que este Departamento procurará que la investigadora encuentre un entorno
  de trabajo adecuado y que pondrá a su disposición los medios necesarios
  para garantizar una labor formativa óptima.
</source>
<target xml:lang="eu" phase-name="validated">
  Sail honek ahalegina egingo duela ikertzaileak behar dituen lan-giro eta
  baliabideak izan ditzan bere prestakuntza aurrera eramateko.
</target>
</alt-trans>
</trans-unit>

```

Figura 5.2: Código XLIFF

Capítulo 6

Consideraciones finales

SARE-Bi, con las funcionalidades descritas, es un sistema que está operativo en la actualidad, aunque en una escala todavía experimental. Puede ser accedido via Internet, con permisos de consulta para los objetos catalogados como públicos. El sistema está pasando por una fase de optimización, sobre todo en los aspectos de usabilidad y eficacia de la gestión para los usuarios con rol validador. El mecanismo de control de versiones actual no contempla el registro histórico que se propone en XLIFF. Su incorporación a SARE-Bi está prevista como función opcional para el control de flujo de trabajo, en las distintas fases de publicación, entre los responsables de redacción, validación y publicación.

Bibliografía

- Abaitua, Joseba, Arantza Casillas, Thomas Diedrich, Arantza Domínguez, Alberto Garay, Carmen Isasi, Inés Jacob, Idoia Madariaga, Raquel Martínez, y José Luis Ramírez. 2001. XML-Bi: procedimientos para la gestión de flujo documental multilingüe sobre TEI/XML. *Procesamiento de Lenguaje Natural*, (27):291–292.
- Booch, Grandy, James Rumbaugh, y Ivar Jacobson. 1999. *El Lenguaje Unificado de Modelado*. Addison Wesley.
- Caplan, Priscilla. 2001. International Metadata Initiatives: Lessons in Bibliographic Control. En *Proceedings of the Bicentennial Conference on Bibliographic Control for the New Millennium*. <http://lcweb.loc.gov/catdir/bibcontrol/>.
- Decker, Stefan, Sergey Melnik, Frank van Harmelen, Dieter Fensel, Michel C. A. Klein, Jeen Broekstra, Michael Erdmann, y Ian Horrocks. 2000. The semantic web: The roles of XML and RDF. *IEEE Internet Computing*, 4(5):63–74.
- Esselink, Bert. 2000. *A Practical Guide to Localization John Benjamins*. John Benjamins.
- J. David Ibáñez Palomar. 2003. Localizer home. <http://www.j-david.net/localizer/>.
- Kashyap, Vinay y Amit P. Sheth. 1998. Semantic heterogeneity in global information systems: The role of metadata, context and ontologies. En G. Schlageter (comp) M. P. Papazoglou, editor, *Cooperative Information Systems: Current Trends and Directions*. Academic Press, páginas 139–178.
- Localization Industry Standards Association. 2003. LISA Web Site. <http://www.lisa.org/>.
- Savourel, Yves. 2001. *XML Internazionalization and Localization*. Sams Publishing.
- TEI Consortium. 2003. Text Encoding Initiative. <http://www.tei-c.org>.
- Trosborg, Anna. 1997. Text Typology: Register, Genre and Text Type. En Anna Trosborg, editor, *Text Typology and Translation*. John Benjamins, páginas 3–23.
- Weibel, Stuart. 1995. Metadata: the foundations of resources description. *D-Lib Magazine*.

Wittenburg, Peter y Daan Broeder. 2002. Metadata Overview and the Semantic Web. En ELDA, editor, *Proceedings of the International Workshop on Resources and Tools in Field Linguistics*, Las Palmas.

Zope Community. 2003. Zope Home Site. <http://www.zope.org>.