3. Migracion de BD ISIS a MySql con campos json

3.1 **Implementación de la MIGRACIÓN de documentos**

1. Creación de la base de datos LILBDBI

* Crea tablas para cada entidad y tabla temporales para el procesamiento de datos enlazados
* Se ejecuta una sola vez
* Fichero: **lildbiDB.php**

1. Recuperar documentos ISIS y guardar datos originales en la tabla document

* Se utiliza el cliente OAI-PMH-PHP para conectarse a las BD ISIS y recuperar los documentos, con el método ListRecords.
* El resultado xml se convierte a array (fichero: xml2array.php)
* Los datos del registro original se guardan en la tabla *document*, campo *originalRecord* de tipo json manteniendo las mismas llaves (v1, v2, ….).
* La traducción de array a estructura de campo json se realiza con las funciones array\_to\_objStr y array\_to\_arrStr (fichero: array2json.inc).
* Se ejecuta periódicamente en búsqueda de nuevos registros y actualización de los existentes
* Se insertan los documentos nuevos llenando *originalRecord* y se actualiza ese mismo campo en aquellos que han sido modificados, reflejando fecha de inserción o modificación en dateCreated o dateTransfered
* Se informa la ctdad de registros nuevos y modificados
* Fichero: **importacion.php**

1. Migración de *originalRecord* a *data* con nuevo modelo de datos basado en espacios de nombres: schema y lilacs.

* Partiendo de originalRecord se traducen los campos y subcampos a propiedades de document, descritas en los espacios de nombres: schema.org y lilacs.sld.cu
* En las propiedades de document que son asociaciones (person, organization, document, descriptores), se asigna @type correspondiente
* Los resultados se guardan en el campo *data.* Se refleja la fecha de actualización del registro en el campo dateModified
* Se informa la ctdad de registros nuevos y modificados
* Se ejecuta después de cada importación si se insertaron o se modificaron registros
* Fichero: **migracion.php**

1. Descubrir y procesar datos enlazados

* Se recorren los registros de todas las tablas de entidades: document, organization, person, event, project
* En cada registro, en las propiedades que son una entidad (tienen @type, asignado en la migración) si es posible se le asigna su @id, quedando como dato enlazado
* Para cada tipo de entidad existe su tabla (Person, Organization, Document) y una tabla con datos temporales (tmp\_Person, tmp\_Organization, tmp\_Document)
* La entidad que se desea representar como dato enlazado, primero se busca (por name ) en la tabla de la entidad, si existe se representa como dato enlazado (se asigna @id de entidad), si no existe se busca en la tabla temporal. Si existe en la temporal y está asociada a una entidad de document se asigna @id, si no existe en la temporal se inserta, quedando como dato temporal.
* La búsqueda de coincidencias se hace de forma recursiva entrando en todas las propiedades del documento y en las propiedades que son entidad. Por ej: author es Person y su propiedad affiliation es Organization.
* Cuando la entidad coincide con una de la tabla definitiva o de la temporal, si tiene subcampos diferentes se agrega la información nueva a la entidad. Chequear y ver flujo de is\_valid
* En la tabla temporal para cada entidad se guarda el id de los documentos donde se hace referencia a ella.
* Las propiedades que se pudieron representar como datos enlazados se actualizan en el campo *document.* Se refleja la fecha de actualización del registro en el campo lastModified
* El proceso construye los datos enlazados y a la vez recopila, en la tabla temporal, las entidades descritas en el documento original. Las entidades temporales solo pasan a la tabla definitiva después de ser validadas.
* Se ejecuta después de cada migración o validación de datos temporales
* Fichero: **datos\_enlazados.php**

1. Validación de datos Temporales

* Revisionde datos de las tablas temporales y para cada entidad define si:
  + es correcta, asignando is\_valid = 1
  + es un error, asignando is\_valid = -1
  + es igual a otra de la tabla temporal (son diferentes porque no coinciden exactamente los nombres), asignando is\_valid = 1 y same\_as = al id de la entidad a la que es igual en la tabla temporal
  + es igual a otra de la tabla definitiva, asignando is\_valid = 1 y id\_person, id\_organization o id\_document = al id de la entidad a la que es igual en la tabla definitiva
  + debe crearse un informe con datos erróneos y same\_as para que sean corregidos en la BD original (ISIS)

1. Agregar datos validados a tabla definitiva de la entidad

* Se ejecuta antes de datos\_enlazados si hay entidades validadas que no están en la tabla definitiva
* Los datos de entidades marcadas como same\_as se unen a la entidad a la que son igual, su nombre se guarda en la propiedad alternateName
* Las entidades validadas se insertan en tabla definitiva (sin incluir en campo alternateName), el id que adquieren en la tabla definitiva se graba en el campo id\_person, id\_organization o id\_document de la tabla temporal.
* Las entidades no validadas no pasan a la tabla definitiva y por tanto no pueden representarse como datos enlazados. Se mantienen en el documento pero sin @id
* Entidades marcadas como error (-1) se debe revisar originalRecord y arreglar si es posible, cambiar también transfrToDB?
* Fichero: **tmp2entity.php**

3.2 Apuntes

1. En campos de autores, el grado de responsabilidad se traduce de la sgte forma

* sin grado de responsabilidad se asigna a author de CreativeWork
* con grado de responsabilidad:
  + aut -> autor

3.3 Problemas detectados en la migración de documentos

- *03* Localización del Documento:

* CUMED No se utiliza código de la institución (manual lilacs 7ma edición), sino su nombre completo, en algunos casos está mal el valor del campo y subcampos, aparece como un array con nombre institución como primer elemento y subcampos como segundo.

Ej: ["Centro Internacional de Restauración Neurológica", "CUE\n      <a>1998</a>\n   "];

debia ser

"Centro Internacional de Restauración Neurológica\n       <?>CUE<?>\n      <a>1998</a>\n   "???

* VIMED No aparece institución o esta en otro subcampo.

Ej: "\n <a>QS518.2</a>\n <b>Dov</b>\n <c>video, (VHS), 2004, 029</c>\n "

* TRAMED parece OK Ej: "CU1.1\n <a>WS 462</a>\n <c>237</c>\n "
* BIBMED están valores variados como la dirección electrónica , siglas, un Numero, un texto,

- *08 Direccion electrónica*: tiene subcampos diferentes en distintas BD

* En VIMED, TRAMED los subcampos se corresponden con doc de LILACS (Manual de Descripción Bibliográfica 7a. edición Julio 2008)
* En CUMED, BIBMED el valor del campo puede ser Internet y para la url el subcampo es <i>
* *10 Autor personal*: En VIMED nombre del autor aparece concatenado en el subcampo afiliación, hay que separarlo de forma manual. No es problema del ws, en otras BD lo devuelve OK, fueron introducidos mal los datos?.
* *14 Paginas (nivel monográfico)*: En CUMED, BIBMED aparece lo mismo "84-9" que “98-130”, no se puede indentifcar pag inicio y pag fin, además este no es el formato descrito <f>pag final <l> pagina inicial, tampoco [pinicial-pfinal]. En TRAMED esta OK

Decision? Entrevista con documentalista para tomar una decisión. Puede ser:

en 84-9 pageStart 84, pageEnd 84+9=93

en 98-130 pageStart 98, pageEnd 130

* 35 ISSN: si el documento no es una serie poner esta campo o incluirlo en isPartOf en la descripción de la serie
* 38 información descriptiva: tiene subcampos diferentes en distintas BD

-en CUMED arreglo con información general sobre el material ilustrativo (solo: ilus, tab, graf, mapas)

-en VIMED, TRAMED parece igual a lo descrito en el manual

* Llaves sin usar

v62 Editora (si es s.n.-sin nombre, no se llena),

v64 FECHA DE PUBLICACIÓN se usa la normalizada,

v84 Fecha de transferencia para la base de datos – no se guarda,

v91 FECHA DE CREACIÓN DEL REGISTRO ,

v98 Registro complementario -> FONTE? ,

v101 REGISTRO COMPLEMENTARIO (EVENTO) ->

"\n      <i>yanetl 36100639</i>\n      <t>201202061010</t>\n      <x>1328541017</x>\n   ",

v110 FORMA DEL ÍTEM,

v114,

v899,

v54 Evnto fecha – se usa la normalizada,

v500,

v66 Ciudad de publicación (s.l. no se llena),

v113