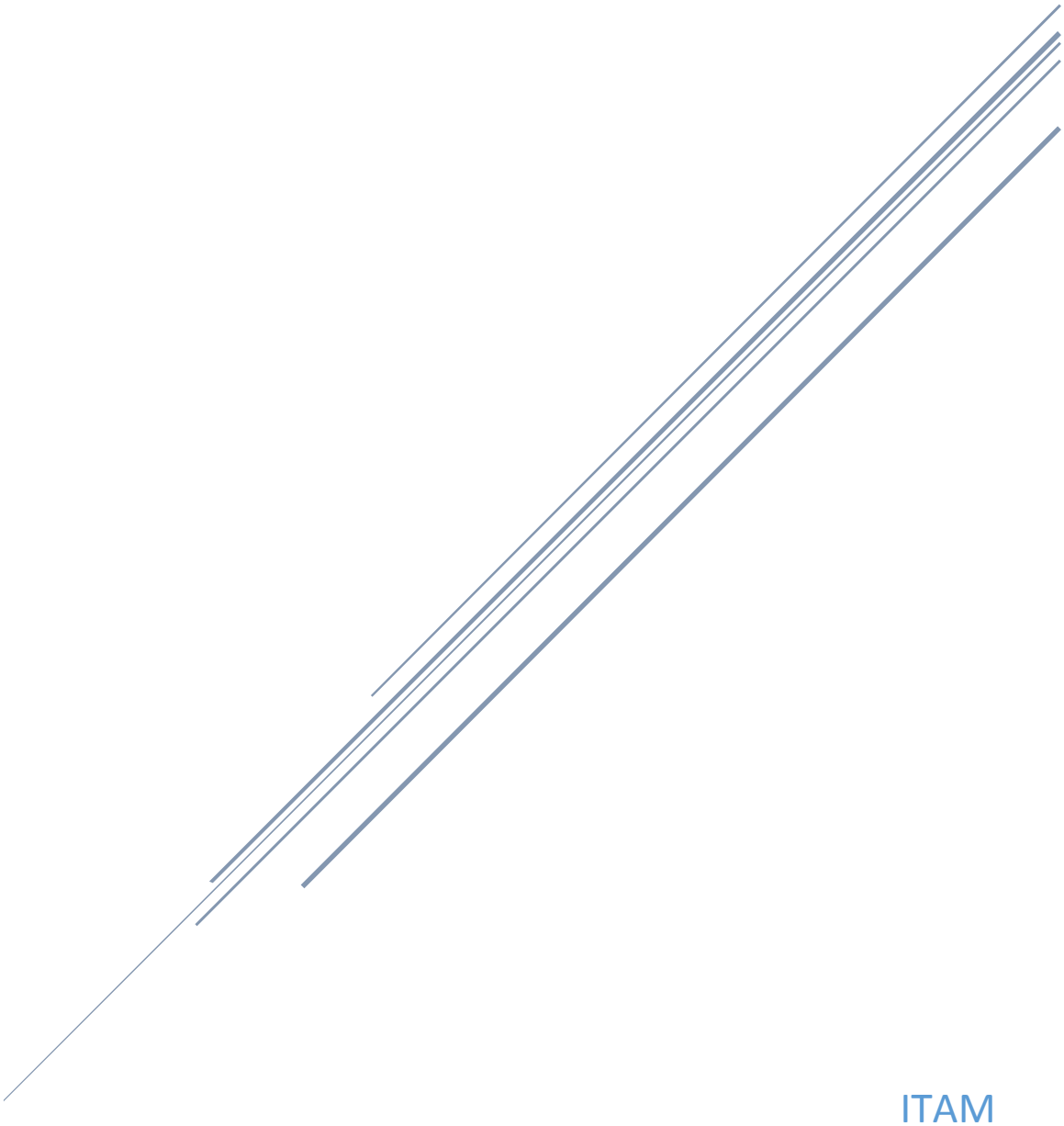
PROYECTO 2 – RDBM

Maestría en Ciencias de la Computación

Métodos de Gran Escala



ITAM
Gilberto Iglesias Roríguez - 98632

Índice

1. Introducción

Este documento contiene el detalle de la implementación del proyecto de la clase "Métodos de gran escala" relacionado al uso de Bases de Datos Relacionales. La base de datos utilizada para almacenar la información será Postgresql.

El objetivo de la implementación descrita en este documento es permitir a diferentes usuarios el consultar, a través de Internet, la información de personas, asociaciones y teatros relacionados al teatro en México durante el siglo XX (Teatro Contemporáneo Mexicano).

La información utilizada en el proyecto mencionado es de carácter público y está contenida en el libro "Diccionario Mexicano de Teatro – Siglo XX"[1].

En el desarrollo mencionado se contempla (i) procesamiento de patrones reconocidos automáticamente para diferenciar información relevante de diferente naturaleza en cadenas de texto plano; (ii) definir la estructura de una base de datos que permita almacenar la información relevante; (iii) procesos de almacenaje de estructuras de datos reconstruidas en una base de datos relacionada; (iv) definición adicional de índices que permitan realizar funciones de "*full-text search*" sin la necesidad de instalar paquetería adicional como Solr.

2. Caracterización de la información

2.1 Fuentes de información

La información utilizada en este proyecto fue provista por una asociación civil dedicada a la investigación en temas relacionados con el teatro. Sin embargo, dado que la información a utilizar fue exportada directamente desde un archivo utilizado para la impresión, la opción más viable era la exportación a archivos de texto plano. En dicho archivo, cada renglón corresponde a una entidad (personas, asociaciones, teatros u otros).

A continuación se muestra un pequeño extracto de uno de los archivos originales.

```
ABBUD, Yolanda / n. Chih. Actriz y escritora. Egresada de la UACH. Inicia su carrera a finales de los setenta. Participa en diversos festivales y recibe premiaciones como mejor ac-triz. Actúa en la CNT del INBA en Felipe Ángeles, de Elena Garro (1999) y Santa Juana de los Mataderos, de Brecht (2001). En el Centro Dramático de Michoacán participa en las obras: El alma buena de Se-Chuan, de Brecht; La dama boba, de Lope de Vega y El inspector, de Gogol. Otras obras en las que ha destacado son: Una luna de pinole, Pancho Villa y los niños de la bola, Mara o De la noche sin sueño y Mamá corazón de acero. Fundadora y actual integrante de Carretera 45 Teatro. Becaria del FONCA y del Instituto Chihuahuense de la Cultura.
ABC / Periódico de arte y variedades, fundado en 1912 en Fresnillo, Zac. Se publicaba semanalmente como órgano del Partido Católico de Zacatecas, bajo la dirección de Arcadio Delgado. Reseñaba acontecimientos teatrales de ese estado. Languideció con la Revolución mexicana hasta desaparecer.
ABRAHAM, Jorge / Actor y cantante. Incursiona en diversas comedias musicales, entre las que destacan Jesucristo Superestrella, de Rice y Webber (1975); El loco, de Gibrán Jalil Gibrán (1977); La pandilla; Godspell, de Schwartz y Tebelak (1980, 1987) y El Full Monty… o todo el paquete, de McNally y Yazbek (2002).
```

[1] Escrito por el Lic. Edgar Ceballos y publicado por Escenología Ediciones en 2013.

Por cuestiones internas a la A.C., la exportación de información se hizo en etapas, ocasionando que se tuviesen múltiples de archivos que deberían ser procesados secuencialmente.

2.2 Variables

Las variables identificables dadas las fuentes de información son:

- Nombre de la entidad (cadena de texto).
- Lugar de nacimiento (cadena de texto, opcional): corresponde al estado o país (en caso de ser extranjero) donde la persona nació. Esta información no aplica para asociaciones ni para teatros, siendo incluso opcional para personas.
- Lugar de muerte (cadena de texto, opcional): corresponde al estado o país (en caso de ser extranjero) donde la persona murió. Esta información no aplica para asociaciones ni para teatros, siendo incluso opcional para personas.
- Año de nacimiento (cadena de texto, opcional): corresponde al año de nacimiento de la persona. Esta información no aplica para asociaciones ni para teatros, siendo incluso opcional para personas.
- Año de muerte (cadena de texto, opcional): corresponde al año de muerte de la persona. Esta información no aplica para asociaciones ni para teatros, siendo incluso opcional para personas.
- Tipo de entidad (cadena de texto, obligatorio): corresponde al tipo de entidad (persona, asociación, teatro o sin clasificar). Esta información no está contenida en los archivos de entrada, sino que es inferida por características de información asociada a la entidad.
- Categorías de la entidad (relaciones a catálogo externo, opcional): esta variable incluye la clasificación de la actividad(es) de la persona, teatro o asociación. Por ejemplo, una persona podría categorizarse como actor, empresario, director, etc.

Por cuestiones relacionadas al workflow esperado del sistema web, se definen variables adicionales:

- Obras de teatro relacionadas a entidades (relaciones a catálogo externo, opcional): se considera la posibilidad de definir las obras de teatro en las que participaron personas o asociaciones, así como aquellas que fueron albergadas por diferentes teatros.
- Validación (booleano, obligatorio): se utiliza una variable booleana

2.3 Tamaño del problema

El tamaño del proyecto consiste en cargar en una base de datos toda la información relacionada con el teatro mexicano contemporáneo. La carga de información se realizará por etapas:

i. Información digitalizada que se encuentra contenida en el libro mencionada en el apartado 1.
ii. Información digitalizada en diferentes medios que no pudo ser incluida en la edición del libro por cuestiones de tamaño físico del mismo.
iii. Información no digitalizada (como registros de periódicos, revistas, etc.) que serán capturados en el sistema por personal de manera manual.

Para efectos de este documento nos enfocaremos únicamente a la primera etapa, en cuyo caso hablamos de más de 20,000 fichas hemerográficas sin homogeneizar.

Asimismo, la información recibida se encuentra sin procesar ni homogeneizar, por lo que los archivos recibidos tienen un tamaño total aproximado de 2.5Gb (incluyendo imágenes).

2.4 Definición de la base de datos

Se definieron las siguientes tablas:

- escenelogia_entidades: alberga el catálogo de entidades (personas, asociaciones y teatros).
- escenologia_entidades_categorias: contiene el catálogo de categorías de una entidad.
- escenologia_entidades_categorias_rel: contiene la relación de entidades con las categorías.
- escenologia_obras: contiene el catálogo de obras de teatro.
- escenologia_obras_relacionadas: contiene la relación de las obras de teatro con entidades.

Para definir la base de datos se tomaron las siguientes decisiones de diseño:

- Dado que las categorías de las entidades pueden variar en el tiempo, se creó una tabla para contener dicho catálogo.
- Dado que los tipos de entidades no van a cambiar significativamente en el tiempo, no se creó una tabla específica para ello, sino que se definió como un campo de la tabla de entidades que acepta 4 valores: persona, asociación, teatro y sin_clasificar.
- Dado que una de las principales funcionalidades de la aplicación es permitir búsquedas de texto sobre los nombres, se requiere definir índices especiales que agilizan la ejecución de dichas búsquedas.
- Dado que se va a realizar un proceso de carga de información manual, se incluyeron campos de información adicionales que son relevantes como dirección y datos de contacto. Asimismo, dado que los campos adicionales son pocos y por facilidad del uso de un framework específico (ver sección 2.5) se decidió no fragmentar la tabla en dos, sino conservar todos los campos en una tabla horizontal.
- En la tabla de escenologia_obras_relacionadas se podría haber utilizado únicamente un campo para relacionar dicha tabla con escenologia_entidades, sin embargo por cuestiones del framework utilizado se decidió utilizar dos campos, facilitando así la obtención de diferentes reportes ara diferentes tipos de entidades.

La definición de los esquemas de las tablas son:

escenelogia_entidades

```
CREATE TABLE escenologia_entidades
(
  id serial NOT NULL,
  website character varying, -- Pagina de internet
  validado boolean, -- Entidad validada?
  ciudad character varying, -- Ciudad
  tipo_entidad character varying, -- Tipo de entidad
  lugar_nacimiento character varying(100), -- Lugar de nacimiento
```

```
  lugar_muerte character varying(100), -- Lugar de muerte
  anio_nacimiento character varying(100), -- Anio de nacimiento
  anio_muerte character varying(100), -- Anio de muerte
  estado character varying, -- Estado
  dato_profesion character varying(200), -- Datos sobre su profesion
  telefono_fijo character varying, -- Telefono fijo
  name character varying(200) NOT NULL, -- Nombre de la entidad
  colonia character varying, -- Colonia
  calle character varying, -- Calle
  codigo_postal character varying(24), -- Zip
  num_ext character varying, -- Numero exterior
  num_int character varying, -- Numero Interior
  create_uid integer, -- Created by
  email character varying, -- Email
  telefono_celular character varying, -- Telefono celular
  dato_exp text,-- Resenia general
  CONSTRAINT escenologia_entidades_pkey PRIMARY KEY (id),
)
WITH (
  OIDS=FALSE
);
CREATE INDEX escenologia_entidades_tipo_entidad_index
  ON escenologia_entidades
  USING btree
  (tipo_entidad COLLATE pg_catalog."default");

CREATE INDEX idx_escenologia_entidades_trgm_gin_name
   ON escenologia_entidades USING gin (name gin_trgm_ops);
```

escenologia_entidades_categorias

```
CREATE TABLE escenologia_entidades_categorias
(
  id serial NOT NULL,
  name character varying(200) NOT NULL, -- Nombre de la categoria
  descripcion text, -- Descripcion de la categoria
  CONSTRAINT escenologia_entidades_categorias_pkey PRIMARY KEY (id),
)
WITH (
  OIDS=FALSE
);
```

escenologia_entidades_categorias_rel

```
CREATE TABLE escenologia_entidades_categorias_rel
(
  partner_id integer NOT NULL,
  category_id integer NOT NULL,
  CONSTRAINT escenologia_entidades_categorias_rel_category_id_fkey FOREIGN KEY (category_id)
      REFERENCES escenologia_entidades_categorias (id) MATCH SIMPLE
      ON UPDATE NO ACTION ON DELETE CASCADE,
  CONSTRAINT escenologia_entidades_categorias_rel_partner_id_fkey FOREIGN KEY (partner_id)
      REFERENCES escenologia_entidades (id) MATCH SIMPLE
      ON UPDATE NO ACTION ON DELETE CASCADE,
  CONSTRAINT   escenologia_entidades_categorias_rel_partner_id_category_id_key   UNIQUE   (partner_id,
category_id)
)
WITH (
  OIDS=FALSE
);

CREATE INDEX escenologia_entidades_categorias_rel_category_id_index
```

```
  ON escenologia_entidades_categorias_rel
  USING btree
  (category_id);

CREATE INDEX escenologia_entidades_categorias_rel_partner_id_index
  ON escenologia_entidades_categorias_rel
  USING btree
  (partner_id);
```

escenologia_obras

```
CREATE TABLE escenologia_obras
(
  id serial NOT NULL,
  validado boolean, -- Obra validada?
  name character varying(200) NOT NULL, -- Nombre de la obra de teatro
    resenia_obra text, -- Resenia de la obra
  CONSTRAINT escenologia_obras_pkey PRIMARY KEY (id),
)
WITH (
  OIDS=FALSE
);
```

escenologia_obras_relacionadas

```
CREATE TABLE escenologia_entidades_obras_relacionadas
(
  id serial NOT NULL,
  funcion_persona character varying(200), -- Nombre de la obra de teatro
  teatro_id integer, -- Teatros deonde se monto la obra
  fechas_montaje character varying(100), -- Fechas de montaje
  obra_id integer, -- Obras de teatro
  persona_id integer, -- Participantes en la obra
  CONSTRAINT escenologia_entidades_obras_relacionadas_pkey PRIMARY KEY (id),
  CONSTRAINT escenologia_entidades_obras_relacionadas_obra_id_fkey FOREIGN KEY (obra_id)
      REFERENCES escenologia_obras (id) MATCH SIMPLE
      ON UPDATE NO ACTION ON DELETE SET NULL,
  CONSTRAINT escenologia_entidades_obras_relacionadas_persona_id_fkey FOREIGN KEY (persona_id)
      REFERENCES escenologia_entidades (id) MATCH SIMPLE
      ON UPDATE NO ACTION ON DELETE SET NULL,
  CONSTRAINT escenologia_entidades_obras_relacionadas_teatro_id_fkey FOREIGN KEY (teatro_id)
      REFERENCES escenologia_entidades (id) MATCH SIMPLE
      ON UPDATE NO ACTION ON DELETE SET NULL,
  )
WITH (
  OIDS=FALSE
);
```

2.5 Observaciones adicionales

En la implementación original se utilizó el sistema ODOO (ERP open-source), el cual utiliza Open-Object (que es un MVC open-source). Por tal motivo la defnición de la base de datos se hizo mediante a definición de los objetos que se iban a utilizar dentro del framework. Sin embargo, fue necesario incluir algunas especificaciones adicionales dentro de la base de datos para mejorar su funcionamiento.

Algunos de los cambios en el diseño original del MCV fueron:

- Instalación de extensiones adicionales en la base de datos:
    - Fuzzystrmatch: permite utilizar funciones de búsqueda de texto como la distancia de Levenshtein.

- Pg_trgm: permite utilizar funciones de búsqueda de texto como la obtención de similitud entre cadenas por trigramas.
- Unaacent: permite normalizar la información guardada en la base de datos para quitar acentos y caracteres especiales.

3. Procesamiento de información

3.1 Limpieza de metadatos

En este caso, no fue necesario renombrar variables provenientes ni limpiar metadatos de la fuente de información original dado que ésta no proveía de ningún esquema.

3.2 Transformación de variables

Se tuvieron que realizar los siguientes procesos de transformación de variables:

- Al momento de reconocer el nombre de la entidad, éste se encontraba en formato "Apellidos, Nombre (Apodos)". Por ello, se realizó un proceso para transformar los nombres al formato "Nombres Apellidos (Apodos)".
- Al momento de obtener la sección con información de datos de natalidad y mortandad, ésta venía en formato "n. lugar_nacimiento m. lugar_muerte (año_nacimiento-año_muerte)". Por ello, se realizó un proceso de transformación para obtener cuatro variables separadas de tipo integer y string (lugar_nacimiento, lugar_muerte, año_nacimeinto, año_muerte).
- Después de la sección de datos de natalidad y mortandad aparecía una sección con las principales actividades de la entidad. Es decir, se describía en forma de prosa cuales eran las funciones de las entidades dentro de las obras de teatro en las que participaron (actor, escritor, coreógrafo, etc.).
    - Se realizó un proceso de reconocimiento de texto contra un diccionario definido manualmente para obtener las palabras claves. Una vez reconocidas las palabras claves se procedió a guardar en memoria las categorías reconocidas para las entidades en cuestión (una vez reconstruida la entidad completa se enviaba a la base de datos simulando un proceso de streaming).

3.3 Recodificación

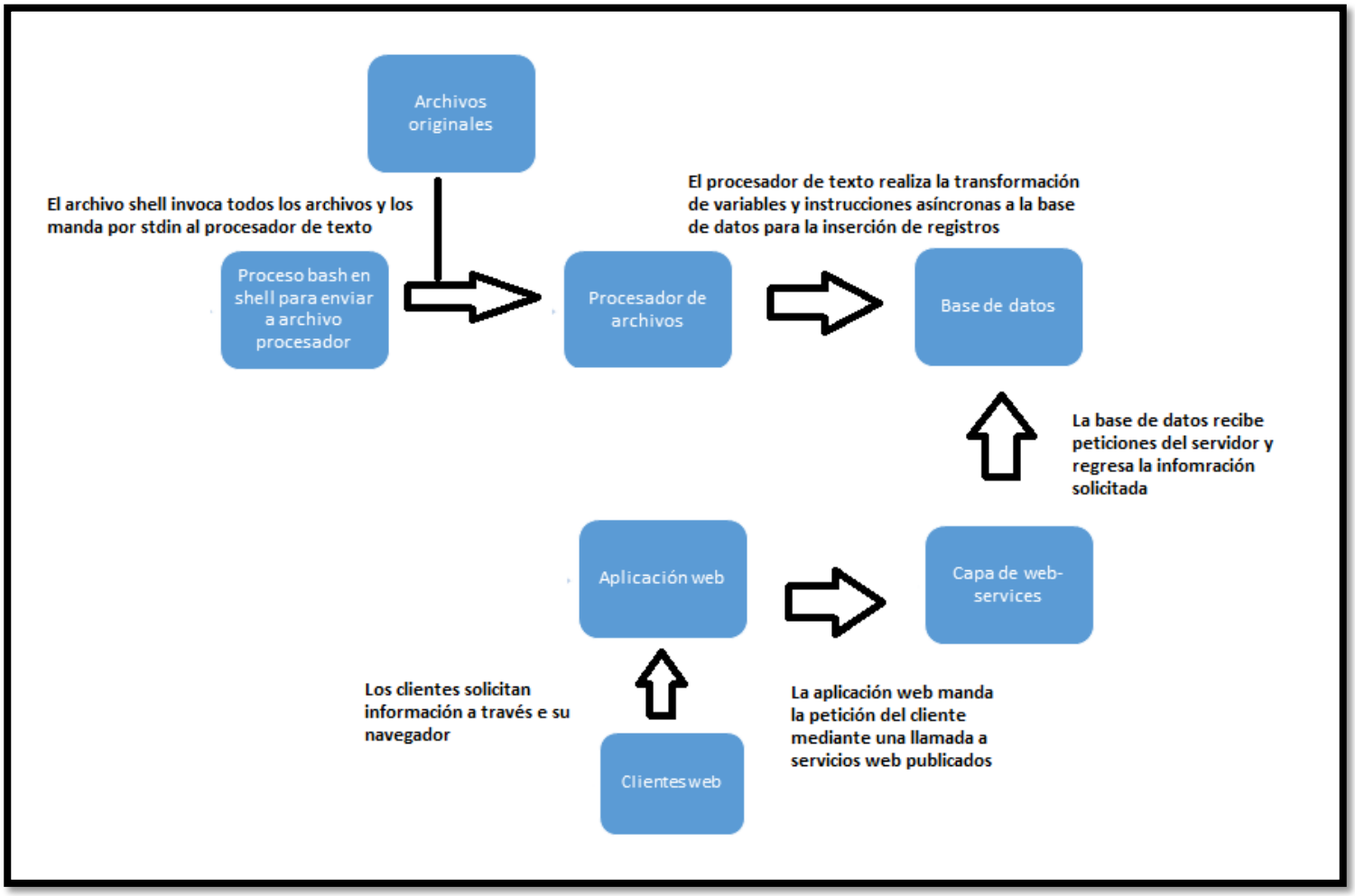
Se requirió realizar los siguientes procesos de recodificación tanto durante el proceso de carga de información como en el proceso de transmisión de la base de datos y la aplicación.

- Se debieron cambiar todas las variables de tipo texto al formato UTF-8, que es el definido en la base de datos.
- Al momento de extraer la información de la base de datos para enviarla a la aplicación se requirió realizar un proceso de decodificación de UTF-8. Lo anterior se debe a que se estructuró una capa intermedia entre el frontend y el backend que transmite la información mediante web services, y no se puede enviar json de cadenas codificadas en UTF-8.

3.4 Variables a ignorar

En este caso no tenemos la necesidad de ignorar información de la fuente de datos. Lo anterior se justifica por el hecho de que se permitirá realizar consultas analíticas sobre todos los campos.

3.5 Flujo de procesamiento



Los componentes incluidos en el diagrama anterior se detallan a continuación:

- Proceso bash en Shell: es un script que ejecuta algunos comandos Shell secuencialmente para la concatenación de archivos fuente y los envía por stdin al procesador de archivos.
- Procesador de archivos: es un script PHP que recibe la información de los archivos fuente por stdin. A continuación ejecuta los procedimientos descritos en las secciones anteriores. El resultado de este script puede verse reflejado de dos maneras:

  - Inserciones directas a la base de datos: esto simula un efecto de streaming, ya que recibe la información de procesos anteriores, realiza un proceso ETL y envía el resultado a la base de datos.
  - Escritura de resultados a un archivo de texto con sentencias SQL: se guardan todas las sentencias SQL a ejecutarse en un archivo de texto. El archivo generado podrá ser importado directamente a la base de datos.
- Base de datos: se utiliza el motor de bases de datos relacionales Postgresql.
- Capa de web-services: está compuesta por scripts PHP que hacen llamadas a scripts en Python (mediante un protocolo RPC). Los scripts Python son los encargados de hacer las consultas a la base de datos.
- Aplicación web: está compuesta por archivos PHP que hacen llamadas Ajax a la capa de web-services para consultar la información.

4. Resultados

Se tiene 2 tipos de resultados:

- Estadísticas de performance de la aplicación en su conjunto.
- Estadísticas analíticas sobre la información de la base de datos.

En relación a las estadísticas de performance, tenemos que el servidor (ver sección Apéndice: Ambiente de desarrollo) soporta una carga de 300 consultas por segundo sin ver afectado la velocidad de ejecución.

Se utilizó esa métrica porque la aplicación va a estar en Internet de manera libre, ocasionando que su mayor carga sea a través de peticiones web.

A pesar de que no es probable que se tengan 300 visitas por segundo, se llegó a estresar el servidor con 1000 peticiones por segundo durante un minuto, no reduciendo éste su velocidad de ejecución.

A continuación se presentan algunas estadísticas analíticas interesantes obtenidas a partir de la base de datos.

Resultado 1. Datos generales sobre la base de datos:

- Número total de entidades: 3934
- Número total de personas: 3206
- Número de personas con fecha de nacimiento: 2010
- Número de personas con fecha de muerte: 987
- Número de personas con lugar de nacimiento: 2084
- Número de personas con lugar de muerte: 1018

Resultado 2. Top 10 de lugares de nacimiento de actores:

Ranking	Valor	Conteo
1	Distrito Federal	906
2	España	126

3	Jalisco	93
4	Yucatán	75
5	Veracruz	74
6	Nuevo León	65
7	Guanajuato	48
8	Puebla	48
9	Argentina	45
10	Michoacán	44

No es inesperado el hecho que el Distrito Federal sea el que más personas relevantes aporte al teatro mexicano del siglo XX. Sin embargo, es importante destacar que España y Argentina aportan más personas relevantes que muchos estados de la República Mexicana. En especial, es importante destacar que España es la segunda entidad que más personas relevantes aporta.

Resultado 3. Top 10 de lugares de muerte de actores:

Ranking	Valor	Conteo
1	Distrito Federal	805
2	EUA	26
3	Jalisco	19
4	Morelos	19
5	España	18
6	Yucatán	18
7	Puebla	15
8	Estado de México	8
9	Nuevo León	8
10	Veracruz	7

No es inesperado el resultado que la mayoría de las personas relevantes mueran en el Distrito Federal, pero si es un poco inesperado que muchos actores se mueran en EUA.

Resultado 4. Top 10 de años de nacimiento de personas:

Ranking	Valor	Conteo
1	1944	41
2	1932	41

3	1934	36
4	1943	35
5	1937	35
6	1954	35
7	1940	34
8	1924	33
9	1930	33
10	1945	33

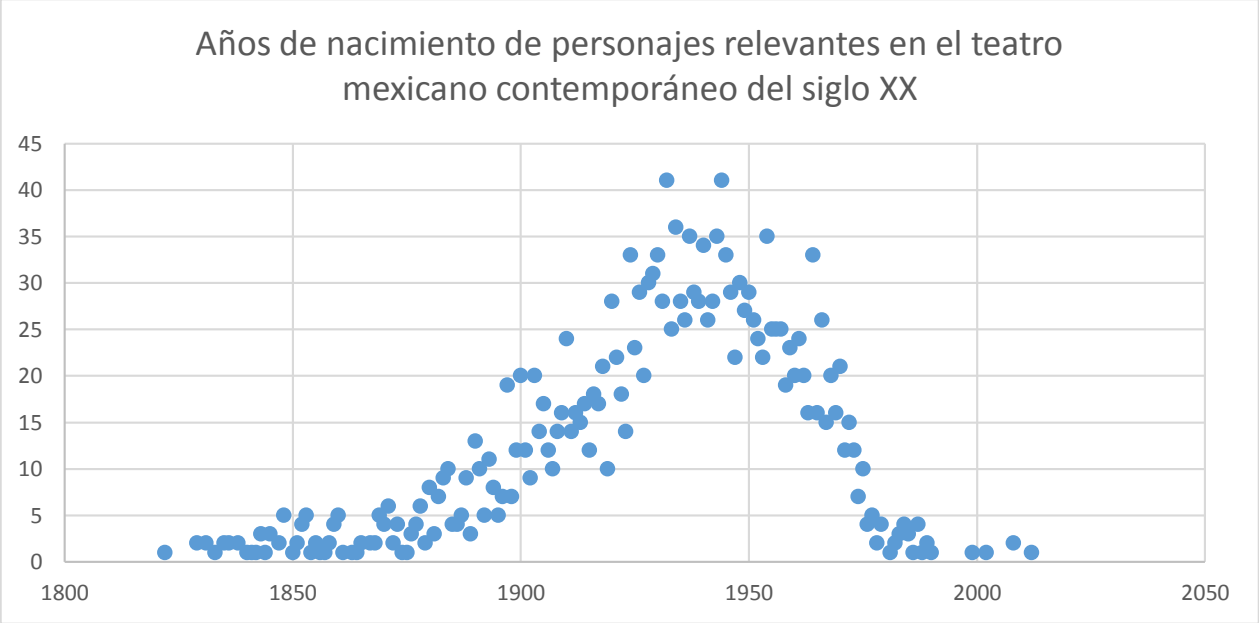
Este resultado es interesante porque podemos ver una tendencia de que muchos de los personajes relevantes del teatro mexicano contemporáneo del siglo XX crecieron en un ambiente de la posguerra (a mediados del siglo XX).

Resultado 5. Top 10 de años de muerte de personas:

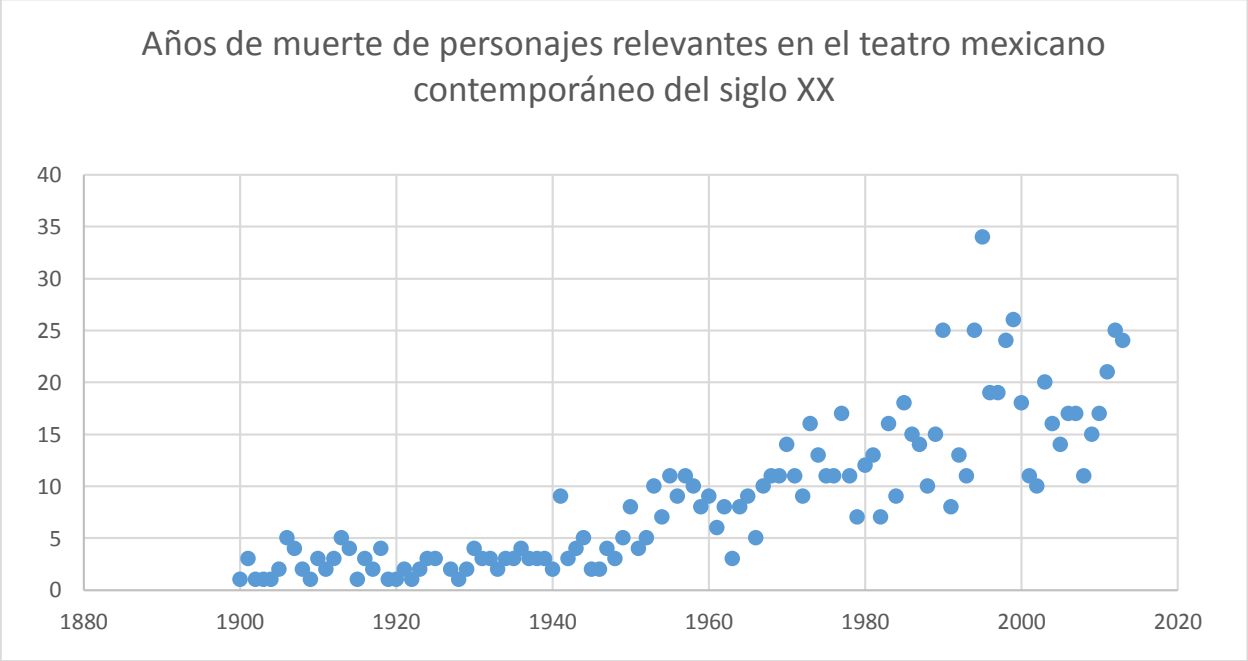
Ranking	Valor	Conteo
1	1995	34
2	1999	26
3	1990	25
4	1994	25
5	2012	25
6	1998	24
7	2013	24
8	2011	21
9	2003	20
10	1997	19

Este resultado es interesante por el hecho que la mayoría de los decesos son muy cercanos a la actualidad. Lo anterior podría explicarse por dos motivos (i) realmente muchos de los decesos acaban de ocurrir, o (ii) se registraron únicamente los decesos recientes.

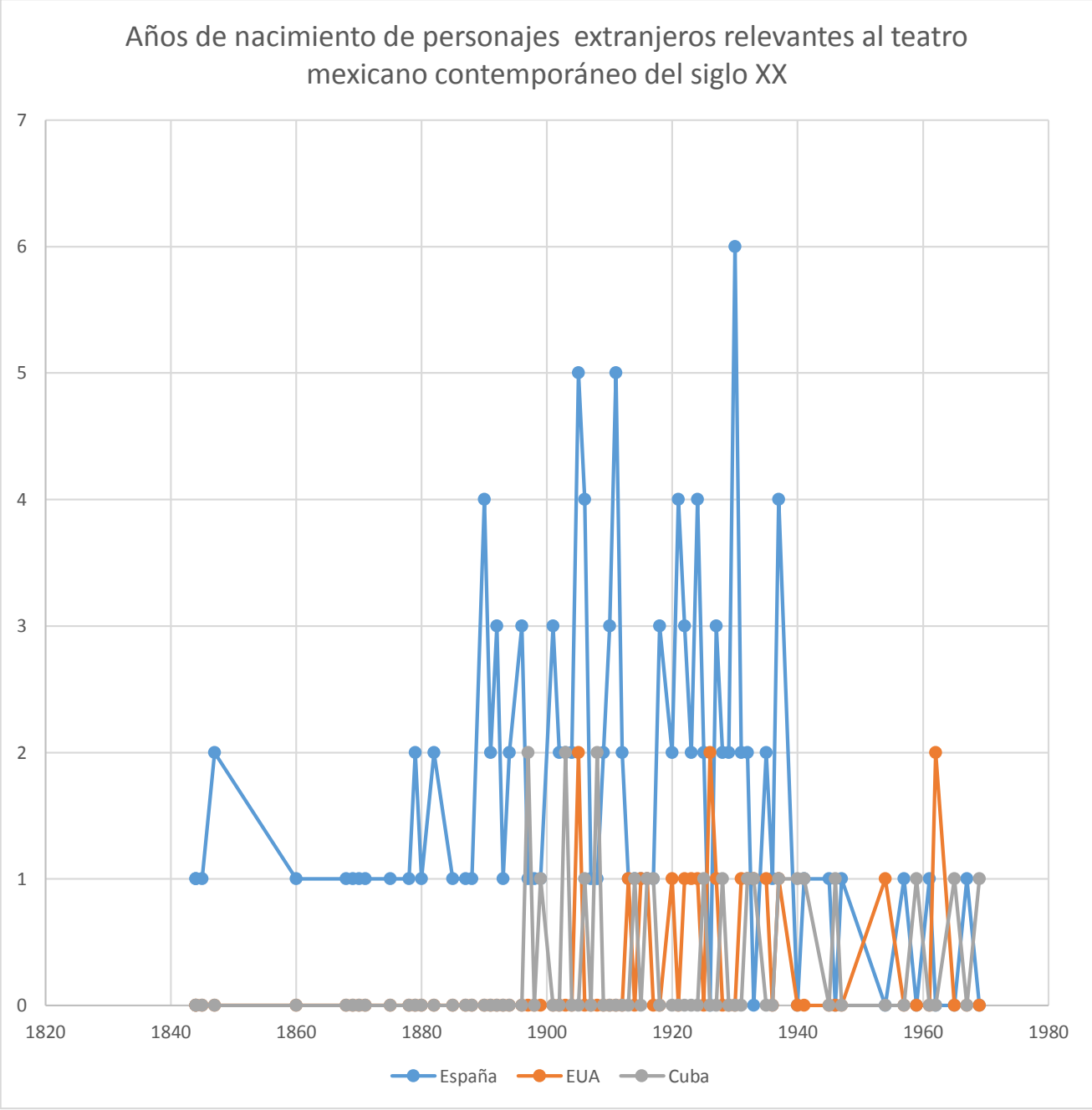
Resultado 6. Serie de tiempo de años de nacimientos:



Resultado 7. Serie de tiempo de años de muerte.



Resultado 8. Progresión de años de nacimientos de extranjeros llegados a México.



Apéndice: Ambiente de desarrollo

El ambiente de desarrollo consta de las siguientes características:

- Sistema Operativo Linux Ubuntu 14.04 Server x64
- Servidor Apache 2.4.7 for Ubuntu
- PHP versión 5.5.9
- Python 2.7.6
- Postgresql 9.3.6