

**Universidade Federal de Santa Catarina**  
**Departamento de Informática e Estatística**  
**Bacharelado em Ciências da Computação**

**Rafael Luís Vasel**

**Um Sistema de Extração e Publicação de Informações**  
**Georreferenciadas em um Domínio Turístico**

Trabalho de Conclusão de Curso submetido à  
Universidade Federal de Santa Catarina como parte  
dos requisitos para a obtenção do grau de Bacharel em  
Ciência da Computação.

Orientador: Prof. Ronaldo dos Santos Mello, Dr.

Florianópolis, julho de 2007

# **Um Sistema de Extração e Publicação de Informações Georreferenciadas em um Domínio Turístico**

**Rafael Luís Vasel**

Trabalho de conclusão de curso apresentado como parte dos requisitos para obtenção do grau de Bacharel em Ciência da Computação.

Banca examinadora:

---

Prof. Ronaldo dos Santos Mello, Dr.  
Orientador

---

Prof. Frank Augusto Siqueira, Dr.

---

Prof. Mário Antônio Ribeiro Dantas, Dr.

---

Marco Antônio Floriano de Oliveira, M.Sc

*We live in a society exquisitely dependent on science and technology, in which hardly anyone knows anything about science and technology.*

---

Carl Sagan

## Sumário

Lista de Figuras.....	vi
Lista de Tabelas.....	vii
Resumo.....	viii
Abstract.....	ix
1 Introdução.....	1
2 Conceitos e Tecnologias Utilizadas.....	3
2.1 Métricas de similaridade.....	3
2.1.1 Levenshtein.....	3
2.1.2 Needleman-Wunch.....	4
2.1.3 Jaro.....	4
2.1.4 Jaro-Winkler.....	4
2.1.5 Matching.....	5
2.1.6 Dice.....	5
2.1.7 Overlap.....	5
2.2 Extração de dados na Web de forma automática.....	6
2.3 Google Earth.....	6
2.3.1 KML.....	6
2.4 Web Feed e Web Syndication.....	8
2.5 Serviços Web.....	8
2.6 Padrões Geográficos.....	8
2.6.1 Sistema Geodésico 1984.....	8
2.6.2 ISO 3166-1.....	9
3 Trabalhos Relacionados.....	10
4 Sistema Proposto.....	11
4.1 Arquitetura.....	11
4.1.1 Extrator.....	13
4.1.2 Geocodificador.....	14
4.1.3 Publicador.....	16
4.2 Implementação.....	17

4.3 Testes.....	19
5 Considerações finais.....	24
Referências Bibliográficas.....	25
Anexo I – Código Fonte do Sistema.....	xxvii
Anexo II – Artigo.....	xxviii

## Lista de Figuras

Figura 1. Exemplo de um arquivo KML.....	7
Figura 2. Exemplo de visualização de um arquivo KML no Google Earth.....	7
Figura 3. Arquitetura do Sistema .....	12
Figura 4. Interface de busca por membros no Hospitality Club.....	12
Figura 5. Exemplo de retorno do web service Geonames para consulta “Jaraguá do Sul” .....	15
Figura 6. Exemplo de retorno do web service Geonames para consulta por latitude e longitudes dos resultados de “Jaraguá do Sul”.....	15
Figura 7. Diagrama de Classes do Sistema.....	17
Figura 8. Interface de configuração do sistema.....	18
Figura 9. Tabelas do banco de dados do sistema.....	18
Figura 10. Gráfico do número de cidades casadas por métrica/limiar sem considerar a região.....	19
Figura 11. Gráfico do número de cidades casadas por métrica/limiar considerando a mesma métrica/limiar para a região.....	20
Figura 12. Gráfico do número de cidades casadas por métrica/limiar considerando a mesma métrica/limiar para a região, incluindo os resultados sem região.....	20
Figura 13. Comparativo entre os três testes anteriores para a métrica Jaro-Winkler.....	21

## **Lista de Tabelas**

Tabela 1. Trabalhos Relacionados.....	10
Tabela 2. Os 15 termos mais frequentes nos nomes das regiões.....	22
Tabela 3. Os 15 termos mais frequentes nos nomes das cidades.....	23

## **Resumo**

Este trabalho descreve a arquitetura de um sistema de georreferenciamento de dados no domínio turístico que visa a facilitar o planejamento de viagens. Coordenadas geográficas são adicionadas aos dados extraídos de um portal na Internet e são exibidos sobre mapas existentes. Diferentemente dos sistemas existentes, este sistema inverte a ordem de navegação na busca pela informação, ou seja, a informação é encontrada através da navegação em imagens geográficas ao invés da informação ser visualizada textualmente e posteriormente ser visualizada em um mapa.

Palavras-chave: Integração de dados, Georreferenciamento, Sistemas de Informação Geográficos, Turismo, Planejamento de viagens.



## **Abstract**

This work describes the architecture of a geotagging system on the touristic domain which is designed to make easier to plan trips. Geographic coordinates are added to the data extracted from a web site and exhibited upon existing maps. Different from existing systems, this system inverts the navigation's sequence on searching for information, e.g., information is found out through the navigation on geographic images instead of information be visualized after it is found out.

Key words: Data Integration, Geo Tagging, Gis, Tourism, Trip planning



# 1 Introdução

O planejamento de uma viagem pode ser visto como um problema de síntese e organização de dados de diversas fontes, uma vez que consultas a hotéis, meios de transporte, lugares etc. são feitas para estabelecer as escolhas e ações a serem tomadas durante a viagem.

Veit Kuehne propôs com a criação do Hospitality Club [1] um modelo de viagem diferente do convencional. Tal modelo propõe que moradores de uma determinada cidade se organizem em um clube baseado em um *site* na Internet e ofereçam suas casas para acomodação de viajantes que, por sua vez, também façam o mesmo em suas cidades de origem, promovendo assim o intercâmbio de pessoas de diferentes culturas. O principal problema de planejar uma viagem nesse modelo é conseguir organizar um roteiro que contenha cidades onde existam membros dispostos a hospedar viajantes. Quanto maior o número de cidades nesse roteiro, mais demorado é o processo de encontro de informações a ele relacionada.

O presente trabalho pretende facilitar este processo, integrando dados dispersos em várias fontes na Internet em um único lugar de fácil acesso. Propõe-se a criação de um Sistema de Georreferenciamento que extraia a informação de sites como o Hospitality Club, georreferencie esta informação e disponibilize-a através do Google Earth. Com a utilização do sistema proposto, minimiza-se o problema ao reunir, já na fase de escolha do roteiro as informações relevantes ao planejamento da viagem.

O Google Earth [11] é um *software*, desenvolvido inicialmente pela empresa Keyhole, adquirida em 2004 pelo Google, que permite a visualização de imagens de satélite e aéreas de todo o globo terrestre. A maioria das cidades é visualizada em uma resolução de 15 metros por *pixel*, porém algumas podem ser visualizadas em até 15 cm por *pixel*. Ele permite que os próprios usuários disponibilizem facilmente informações referenciadas nas imagens a outros usuários do *software*. Sua escolha como camada para visualização se deve à sua disponibilidade, por possuir uma interface intuitiva e de fácil uso e ser popular tanto entre entusiastas quanto usuários típicos de sistemas de informação geográfica (SIG), como por exemplo agentes do poder público .

Muitas fontes de dados turísticos já disponibilizam alguma forma de visualização da informação em mapas / imagem geográfica, porém sempre de forma individual. A contribuição principal do trabalho será integrar estas fontes em um só local, de forma incremental, ou seja, a arquitetura do sistema possibilita que posteriormente outras fontes de informação possam ser adicionadas, não necessariamente no mesmo domínio (turístico) originalmente escolhido. Outra contribuição é inverter a ordem de navegação para se chegar à informação. Enquanto comumente se encontra a informação e então a visualiza no mapa/imagem, o trabalho propõe que a informação seja encontrada através da navegação na imagem, não exigindo desta forma conhecimentos prévios da localização geográfica da informação.

O trabalho é organizado da seguinte forma: no capítulo dois são apresentadas as tecnologias e conceitos utilizados. No capítulo três, são listados os trabalhos relacionados e um quadro comparativo com o trabalho proposto. No capítulo quatro é explicada detalhadamente a arquitetura do sistema, contendo a descrição dos componentes envolvidos e o relacionamento entre eles. No capítulo cinco, são apresentadas as considerações finais.

## 2 Conceitos e Tecnologias Utilizadas

Este capítulo apresenta uma breve revisão bibliográfica das tecnologias envolvidas no desenvolvimento do sistema proposto.

### 2.1 Métricas de similaridade

As seções seguintes apresentam as métricas de similaridade do pacote *Simmetrics*, uma biblioteca disponível sob uma licença livre que provê algoritmos de similaridade entre duas *strings*, utilizados no componente Geocodificador para encontrar em um conjunto de resultados a *string* de maior similaridade com o nome da cidade. Essa biblioteca oferece tanto o valor normalizado em ponto flutuante entre 0 e 1, sendo 0 a desigualdade e 1 a igualdade quanto o valor não normalizado, que pode variar muito entre uma métrica e outra, não permitindo a comparação entre elas.

#### 2.1.1 Levenshtein

A distância Levenshtein ou distância básica de edição entre duas sequências de caracteres (*strings*) é dada pelo número mínimo de operações necessárias para transformar a *string* A na *string* B. As operações são as seguintes:

- copiar um caracter da string A para a string B (custo 0)
- excluir um caracter da string A (custo 1)
- substituir um caracter por outro (custo 1)

O nome advém do cientista russo Vladimir Levenshtein, que considerou esta distância já em 1965. [23]

Exemplo: Para as *strings* Curitiba e CTBA, o resultado não normalizado é igual a 7, enquanto que o normalizado é igual a 0,125 (7 operações para 8 caracteres). Para as *strings* Jaraguá do Sul e Jaraguá, os resultados são 7 e 0,5, respectivamente.

### 2.1.2 Needleman-Wunch

Similar à distancia básica de edição (Levenshtein), esta adiciona um custo variável à cada operação. Em outras palavras, a distância Levenshtein pode ser vista como uma simplificação da Needleman-Wunch com custo constante igual a um.

O algoritmo foi proposto em 1970 por Saul Needleman and Christian. [15] É utilizado para alinhamento de sequências de proteínas do DNA.

Exemplo: Para as *strings* Curitiba e CTBA, com um custo de exclusão/substituição igual a dois, o resultado não normalizado é igual a 8, enquanto que o normalizado é igual a 0,5. Para as *strings* Jaraguá do Sul e Jaraguá, os resultados são 13 e 0,53, respectivamente.

### 2.1.3 Jaro

A métrica Jaro calcula o número de correspondências e transposições dividido pelo tamanho das *strings*. A fórmula é a seguinte:

$$d_j = \frac{m}{3a} + \frac{m}{3b} + \frac{m - t}{3m}$$

onde  $m$  é o número de correspondências entre caracteres,  $t$  é o número de transposições (quantidade de posições em que o caracter da *string* A não corresponde ao caracter da *string* B),  $a$  e  $b$  são os tamanhos das duas *strings*. [24]

Exemplo: Para as *strings* Curitiba e CTBA, o resultado é 0,458 ( $m=4, a=8, b=4, t=9$ ). Para as *strings* Jaraguá do Sul e Jaraguá, o resultados é 0,83 ( $m=7, a=14, b=7, t=0$ ).

### 2.1.4 Jaro-Winkler

A métrica *Jaro-Winkler*, derivada da métrica Jaro, é calculada da seguinte forma:

$$d_{jw} = \frac{m}{3a} + \frac{m}{3b} + \frac{m - t}{3m} + (l * p * (1 - \frac{m}{3a} + \frac{m}{3b} + \frac{m - t}{3m}))$$

onde  $m$  é o número de correspondências entre caracteres,  $t$  é o número de transposições,  $a$  e  $b$  são os tamanhos das duas *strings*,  $l$  é o tamanho do prefixo comum entre as duas *strings* e  $p$  é um fator de ajuste. [17]

Exemplo: Para as *strings* Curitiba e CTBA, o resultado é 0,51. Para as *strings* Jaraguá do Sul e Jaraguá, o resultado é 0,9 .

### 2.1.5 Matching

O algoritmo Matching é calculado pelo número de termos comuns entre as duas *strings*. Por exemplo, para as *strings* Curitiba e CTBA, não há termos comuns. Logo, para essa métrica, o resultado é zero. Para as *strings* Jaraguá do Sul e Jaraguá, há um termo comum, logo, o resultado é um, sendo normalizado para 0,333 (um termo igual entre os três existentes).

### 2.1.6 Dice

A similaridade Dice é medida pelo dobro do número de termos comuns dividido pelo número total de termos em ambas as strings. [25] Para as *strings* Curitiba e CTBA, o resultado é igual a zero, pois não há termos comuns. Para as *strings* Jaraguá do Sul e Jaraguá, o resultado é 0,5, pois há um termo comum (Jaraguá) e quatro termos somando-se as duas strings.

### 2.1.7 Overlap

Essa métrica define se uma das strings é um subconjunto da outra, resultando em 1 em caso positivo e em zero em caso negativo. Para as *strings* Curitiba e CTBA, o resultado é 0, enquanto que para as *strings* Jaraguá do Sul e Jaraguá, o resultado é 1.

## 2.2 Extração de dados na *Web* de forma automática

A extração de dados publicados em páginas na *Web* de forma automática se dá geralmente através de programas denominados *Web Crawlers*, um tipo de agente de software muito utilizado por mecanismos de busca para copiar páginas inteiras publicadas na Internet, com o objetivo de posterior indexação. Em geral, é iniciado com uma lista de URLs a visitar, chamados de sementes, que são percorridas pelo *Crawler* em busca de outras URLs, que por sua vez também são inseridas nesta lista. [5]

## 2.3 Google Earth

O Google Earth é um programa que disponibiliza um modelo tridimensional do globo terrestre através de fotos aéreas e de satélites. O usuário tem a liberdade de rotacionar esse globo e aproximar-se ou distanciar-se da superfície. Ele possui diversas funcionalidades, tais como a medição de distâncias e a possibilidade de adição de várias camadas. A sua versão mais simples, embora limitada em funcionalidades como a integração com aparelhos de GPS portáteis, é gratuita. Seu grande atrativo é possibilitar que os próprios usuários criem e publiquem informações georreferenciadas. Ele é compatível com os sistemas operacionais *Windows XP*, *Mac OS X* e *Linux*.

### 2.3.1 KML

O KML (*Keyhole Markup Language*) é um formato de dados baseado em XML utilizado para mostrar dados no Google Earth e no Google Maps. Os arquivos KML são geralmente compactados e distribuídos como arquivos com extensão KMZ. O elemento mais comum é o *Placemark*, elemento que marca uma posição na superfície terrestre, usando um círculo amarelo como ícone. A forma mais simples desse elemento inclui somente um elemento `<Point>`, que especifica a localização do ícone. É possível ainda especificar um nome, uma descrição e uma série de outros atributos. A definição completa do esquema do KML, com todos os elementos e atributos possíveis pode ser encontrada na documentação *on-line* da linguagem. [18] O exemplo da Figura 1 ilustra a estrutura de um arquivo KML, composto por um elemento `<Placemark>` contido em uma estrutura de diretório composta por país, região e cidade. A Figura 2 mostra a



visualização deste arquivo quando aberto no Google Earth. Os dados exibidos são de uma participante do Hospitality Club chamada Anja Tabery, que mora na cidade de Nuremberg, na região de Bayern, Alemanha.

Para publicar conteúdo dinâmico, utiliza-se o elemento `<NetworkLink>`, que contém um elemento `<Link>`, onde é possível especificar um documento KML ou um *site* que gere um documento KML. Pode-se, assim, gerar conteúdo dinamicamente, tanto em uma rede local, quanto na Internet.

```

<?xml version="1.0" encoding="UTF-8"?>
<kml xmlns="http://earth.google.com/kml/2.1">
  <Folder>
    <name>HC</name>
    <Folder>
      <name>Germany</name>
      <Folder>
        <name>Bayern</name>
        <Folder>
          <name>Nurnberg</name>
          <Placemark>
            <name>tapechen</name>
            <description>Anja Tabery
            you are the sun, you are the only one, you are so cool...you are so
            ROCKN'N ROLL! come to Nürnberg and visit this wonderful town! I love meeting and helping people from all over the world- you are
            welcome!</description>
            <Point>
              <coordinates>11.0683333,49.4477778,0</coordinates>
            </Point>
          </Placemark>
        </Folder>
      </Folder>
    </Folder>
  </kml>

```

Figura 1. Exemplo de um arquivo KML

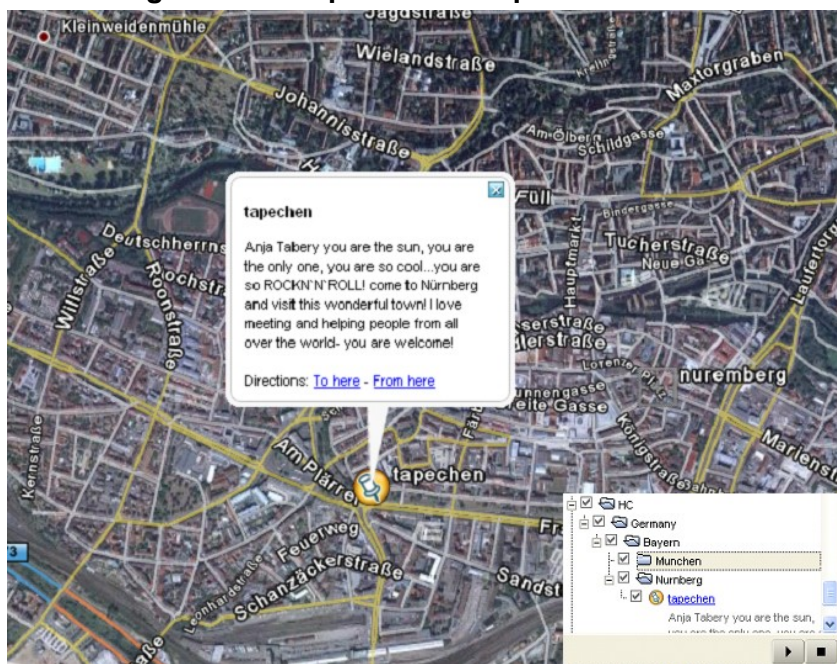


Figura 2. Exemplo de visualização de um arquivo KML no Google Earth.

## 2.4 Web Feed e Web Syndication

*Web syndication* é o termo utilizado para designar a disponibilização de partes de um *site* provedor de conteúdo a terceiros. Geralmente, isto ocorre através da criação de *web feeds*, documentos baseados em XML que podem ser lidos por outros *sites* ou por *softwares* agregadores de conteúdo. Originalmente, foi utilizado em *blogs* e *sites* de notícias e, posteriormente, adotado por vários outros tipos de conteúdo na *web*: músicas, jornais, vídeos, lojas virtuais etc.

## 2.5 Serviços Web

Os serviços *web* são uma tecnologia utilizada na interoperabilidade de aplicações na Web. As aplicações envolvidas trocam mensagens em XML sobre o protocolo HTTP em padrões abertos padronizados por órgãos como a World Wide Web Consortium (W3C) e a Organization for the Advancement of Structured Information Standards (OASIS). A vantagem sobre tecnologias semelhantes (como Corba, RMI e DCOM) é a possibilidade de aplicações desenvolvidas em diferentes linguagens e diferentes sistemas operacionais poderem interagir entre si. A utilização de um *web service* geralmente ocorre de forma transparente, bastando uma chamada a uma função na própria linguagem que encapsula os parâmetros em XML, faz uma requisição ao *web service* e retorna o resultado da computação. O transporte dos dados entre as aplicações é realizado normalmente através do protocolo HTTP, tornando mais fácil em relação às outras tecnologias o fluxo entre *firewalls* das diferentes redes que hospedam as aplicações.

## 2.6 Padrões Geográficos

As seções a seguir apresentam os padrões geográficos utilizados.

### 2.6.1 Sistema Geodésico 1984

Considerando que a Terra não é uma esfera perfeita, é necessária uma técnica para transformar latitudes e longitudes obtidas por satélites ou cartógrafos para uma posição

real na face da Terra. Essa técnica é definida por um Sistema Geodésico. O Google Earth e o Geonames utilizam o Sistema Geodésico Mundial, revisão de 1984 (WGS84), criado pelo departamento de defesa estadunidense para a utilização no Sistema de Posicionamento Global (GPS).

Um outro é o Sistema Geodésico Brasileiro (SGB) [4], elaborado pelo IBGE por meio do decreto nº 5334/2005 e atribuído como o padrão para a cartografia nacional.

### 2.6.2 ISO 3166-1

O padrão ISO 3166-1 é parte da norma ISO 3166 e provê códigos para os nomes de países e territórios dependentes. Foi publicado em 1974 pela International Organization for Standardization (ISO) e define três códigos diferentes para cada país ou território:

- ISO 3166-1 alpha-2: define códigos com 2 letras. Possui diversas aplicações para localização e internacionalização de *softwares*. É o código utilizado para a nomenclatura de domínio na Internet;
- ISO 3166-1 alpha-3: define códigos com 3 letras. É utilizado principalmente na identificação de passaportes;
- ISO 3166-1 numeric: define códigos de 3 dígitos. Foi criado pela divisão de estatística das Nações Unidas. É utilizado principalmente em alfabetos não latinos.

A tabela completa está disponível *on-line* no *site* da ISO. [13].

### 3 Trabalhos Relacionados

O próprio *site* Hospitality Club disponibiliza a informação em mapas, utilizando-se das funções de programação do Google Maps [10], porém esta visualização ainda não está disponível para todos os países e não disponibiliza a possibilidade de navegação para outros locais a partir do próprio mapa.

Uma outra iniciativa semelhante é o *EarthBooker* [7], um *site* que disponibiliza informações no Google Earth sobre 80 mil hotéis localizados ao redor do mundo. Entretanto, se comparado ao sistema apresentado, o *EarthBooker* não extrai informação de forma automática. As partes necessitam fazer um acordo pago com o sistema e então as informações são disponibilizadas ao usuário final.

Além dos diferenciais existentes com relação aos trabalhos relacionados, uma característica particular do sistema apresentado é a possibilidade de futuramente agregar outras fontes de dados através da incorporação de informações relevantes dentro deste contexto em um único local geográfico.

**Tabela 1. Trabalhos Relacionados.**

Trabalho	Informação publicada no Google Earth ?	Navegação pelo mapa ?	Informação	Extração automática de Georreferenciamento?
Mapas no Hospitality Club	Não	Não	Membros do HC	Não
Earthbooker	Sim	Sim	80 mil hotéis espalhados pelos 5 continentes	Não
Trabalho proposto	Sim	Sim	Membros do HC + possibilidade de adição de outras	Sim

## 4 Sistema Proposto

### 4.1 Arquitetura

Os componentes da arquitetura do sistema são os seguintes: Extrator, Geocodificador e Publicador. Um diagrama do relacionamento entre estes componentes é apresentado na Figura 3. Eles atuam isoladamente, compartilhando um mesmo banco de dados. As tabelas do banco de dados são apresentadas na Figura 9. A utilização das tabelas por cada componente é descrita nas próximas seções.

Até o momento, a única fonte dos dados é o Hospitality Club, um clube internacional fundado no ano 2000 e baseado em um *site* na Internet que propõe uma filosofia de viagem que promove intercâmbio entre pessoas. Para fazer parte deste clube é necessário somente um registro gratuito no próprio *site*, onde se informam dados pessoais, preferências e papéis no clube. Os papéis que um membro pode assumir são anfitrião (oferecendo sua residência para hospedagem de outros membros) ou hóspede (acomodando-se em residências oferecidas por outros membros). A Figura 4 mostra a interface do Hospitality Club.

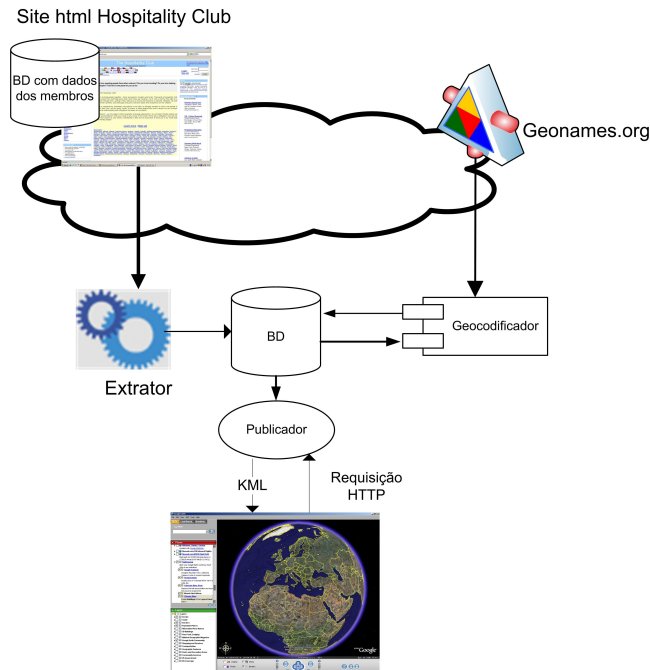


Figura 3. Arquitetura do Sistema

A arquitetura do sistema proposto permite que se compreenda o seu funcionamento. O Extrator busca as informações no *site* do Hospitality Club e as insere no banco de dados. O Geocodificador lê estes dados e associa-lhes a informação da latitude e longitude, utilizando como fonte o Geonames. O Publicador apresenta o resultado deste processamento no Google Earth. As próximas seções detalham cada um destes componentes.

Hospitality Club  
...bringing people together!

Menu > All countries > Venezuela > Anzoátegui > Puerto La Cruz

City Volunteers

Active members in Puerto La Cruz, Anzoátegui, Venezuela

Photo:	User name:	Name:	Accommodation:	Profile Summary:
	caraballo01	Oscar Caraballo	yes	Para personas sencillas, honestas y cordiales.
	xamby	Oscar Caraballo	yes	

Info about Puerto La Cruz, Anzoátegui, Venezuela

Introduction

Top Things to See and Do

Nearby cities/suburbs:

Who is coming?

Want more guests?

Hospitality Club Meetings

Last postings about Puerto La Cruz Anzoátegui Venezuela from other users

caraballo01 wrote:

Canaima, El salto Angel, Los Roques, Mochima,

Figura 4. Interface de busca por membros no Hospitality Club

#### 4.1.1 Extrator

O Extrator é o componente que tem como objetivo a coleta das informações no *site* alvo (no caso, o Hospitality Club). Essa informação compreende tanto a estrutura geográfica quanto a informação dos membros (nome, telefone, endereço etc.). A estrutura geográfica em que se navega no *site* apresenta três níveis: *país*, *região* e *cidade*. Como a entrada dos dados durante o cadastro no *site* é livre, algumas inconsistências que tiveram que ser tratadas, como o fato de que várias cidades nessa estrutura estão em mais de uma região. Na sequência, é comentado como esta inconsistência é tratada.

O funcionamento do Extrator é o seguinte: uma requisição à página inicial que apresenta apontadores para todos os países onde há membros do clube é efetuada. Cada apontador possui o nome do país e o código deste. Esses dados são gravados na tabela *Países* do banco de dados do sistema. Na sequência, uma requisição à página de regiões é feita, utilizando o código de cada país, e os dados da região são gravados. O mesmo procedimento é repetido para cada cidade e para os membros. Quando uma cidade está contida em mais de uma região (utilizando o mesmo código), a segunda ocorrência é ignorada, impedindo de haver duplicidade de registro de cidades. A única consequência dessa decisão é que cidades que estão relacionadas a mais de uma região (por erro ou para facilitar a navegação), somente aparecem na árvore mostrada no canto inferior direito da Figura 5 na região da primeira ocorrência. Independente disto, a cidade continua a ser georreferenciada corretamente, ou seja, continua a ser apresentada sempre nas mesmas coordenadas geográficas, na sua região real.

Ao final da execução do Extrator, o banco de dados está populado com toda a estrutura geográfica do *site* e com a informação dos membros.

### 4.1.2 Geocodificador

O Geocodificador é o componente responsável por adicionar a informação de latitude e longitude à cada registro de cidade. A sua execução requer a configuração prévia dos seguintes parâmetros:

- o nome do algoritmo de similaridade a ser utilizado para a comparação entre o nome da cidade e as ocorrências dessa cidade no serviço de georreferenciamento utilizado. Neste sistema, estão disponíveis os algoritmos da biblioteca de código livre *Simmetrics*;
- um limiar entre 0 e 1 para aceitação do resultado obtido (*threshold*).

O funcionamento do Geocodificador se divide em 3 partes, apresentadas nas seções a seguir.

#### **Consumo do *Web Service Geonames***

O Geonames é um *Web Service* disponível gratuitamente na Internet. A partir do nome de uma cidade, este serviço retorna a latitude e longitude da mesma, no formato XML. Como a entrada de dados é livre no Hospitality Club, muitas vezes ocorrem erros de digitação no nome da cidade. O retorno do Geonames, nestes casos, são várias cidades com nome semelhante ao requerido. A Figura 5 apresenta o retorno do Geonames quando consultada a cidade *Jaraguá do Sul*. O Geonames não retorna nesses resultados a região à qual a cidade faz parte. A fim de evitar que ocorrências de cidades homônimas, dentro do mesmo país, porém de regiões distintas, para cada ocorrência retornada é realizada uma nova consulta, através de georreferenciamento reverso a partir dos dados de latitude e longitude. Desse modo, é possível adicionar a informação da região a cada registro dos resultados do Geonames. Um exemplo dessa requisição reversa pode ser vista na Figura 6. As latitudes e longitudes requisitadas correspondem às retornadas na Figura 5. Percebe-se assim que o primeiro registro de “Jaraguá do Sul” corresponde ao estado de Santa Catarina, enquanto que o segundo localiza-se no estado do Mato Grosso do Sul.



```

<geonames>
  <totalResultsCount>2</totalResultsCount>
  <geoname>
    <name>Jaraguá do Sul</name>
    <lat>-26.4833333</lat>
    <lng>-49.0666667</lng>
    <geonameId>3460102</geonameId>
    <countryCode>BR</countryCode>
    <fcl>P</fcl>
    <fcode>PPL</fcode>
  </geoname>
  <geoname>
    <name>Jaraguá</name>
    <lat>-20.4666667</lat>
    <lng>-54.7666667</lng>
    <geonameId>3460105</geonameId>
    <countryCode>BR</countryCode>
    <fcl>P</fcl>
    <fcode>PPL</fcode>
  </geoname>
</geonames>

```

**Figura 5. Exemplo de retorno do web service Geonames para consulta “Jaraguá do Sul”**

<pre> - &lt;geonames&gt;   - &lt;countrySubdivision&gt;     &lt;countryCode&gt;BR&lt;/countryCode&gt;     &lt;countryName&gt;Brazil&lt;/countryName&gt;     &lt;adminCode1&gt;11&lt;/adminCode1&gt;     &lt;adminName1&gt;Mato Grosso do Sul&lt;/adminName1&gt;   &lt;/countrySubdivision&gt; &lt;/geonames&gt; </pre>	<pre> - &lt;geonames&gt;   - &lt;countrySubdivision&gt;     &lt;countryCode&gt;BR&lt;/countryCode&gt;     &lt;countryName&gt;Brazil&lt;/countryName&gt;     &lt;adminCode1&gt;26&lt;/adminCode1&gt;     &lt;adminName1&gt;Santa Catarina&lt;/adminName1&gt;   &lt;/countrySubdivision&gt; &lt;/geonames&gt; </pre>
--	--

**Figura 6. Exemplo de retorno do web service Geonames para consulta por latitude e longitudes dos resultados de “Jaraguá do Sul”**

### Definição do Grau de Similaridade dos Resultados

A cada registro do retorno da consulta ao Geonames é adicionado um grau de similaridade determinado pela métrica escolhida previamente, tanto para o nome da cidade quanto para o nome do estado.

### Filtragem dos Resultados

No uso do Geonames, três situações podem ocorrer:

a) *Nenhuma ocorrência semelhante foi encontrada.* Neste caso, a cidade fica sem a informação geográfica e outro serviço de georreferenciamento deverá ser utilizado. No

trabalho atual, apenas é relatado que não houve êxito no georreferenciamento desta cidade; Trabalhos futuros podem incluir outras fontes dessa informação nesses casos.

*b) Ao menos uma ocorrência com similaridade de 100 % foi encontrada.* Neste caso, a informação geográfica é então associada ao registro da cidade no banco de dados;

*c) Várias ocorrências foram encontradas, porém sem similaridade de 100%.* Neste caso, se existir algum registro com similaridade superior ao limiar pré-configurado, adota-se o mesmo procedimento da situação “b”. Caso não exista, adota-se o procedimento da situação “a”.

#### 4.1.3 Publicador

O Publicador é o componente responsável pela interação do usuário com o sistema, ou seja, é a interface para visualização dos dados georreferenciados. Ele consiste em um *site* que é invocado pelo Google Earth sempre que o usuário altera a área de visualização do mapa em que está navegando. O Google Earth envia a latitude e longitude do canto superior esquerdo e do canto inferior direito. Deste modo, é possível publicar a informação sob demanda. Um exemplo da informação publicada pode ser vista na Figura 2.

A cada alteração da área de visualização e conseqüente requisição da página são percorridas todas as cidades com latitudes e longitudes entre as latitudes e longitudes enviadas pelo Google Earth, montando assim uma estrutura semelhante à apresentada na Figura 4, contendo uma árvore com país, região e cidade seguida da lista dos membros destas cidades, com o resumo que este membro cadastrou no Hospitality Club. Há a possibilidade de se configurar a partir de que ponto que o Publicador deve montar esta lista, a fim de que se evite, por exemplo, que ele monte uma lista com todas as cidades de um determinado continente ao mesmo tempo, o que pode ser muito demorado dependendo da velocidade de conexão. Essa configuração é pré-determinada no servidor através do fornecimento de um número inteiro positivo que representa a diferença, em módulo, entre a latitude inicial e final da área de visualização. Por exemplo, se este parâmetro for 30, seria mostrado dados de uma área correspondente à

todo o Brasil. Quando configurado em 3, mostraria uma área equivalente ao do estado de Santa Catarina.

## 4.2 Implementação

O Extrator e Geocodificador foram desenvolvidos em C#, utilizando-se do .NET Framework. O Publicador foi desenvolvido em ASP.net. A Figura 7 apresenta o diagrama de classes do Extrator e Geocodificador. As classes Extrator e Geocodificador derivam da classe Base, que fornece atributos básicos a ambas as classes e métodos de comunicação entre essas classes e as classes da interface gráfica, provendo métodos como o de atualização do progresso das tarefas e comunicação de erro ao usuário. A classe Extrator apresenta métodos comuns a um extrator genérico, a qual pode ser utilizada futuramente no desenvolvimento de um extrator de outra fonte. A classe ExtratorHC é específica para o Hospitality Club.

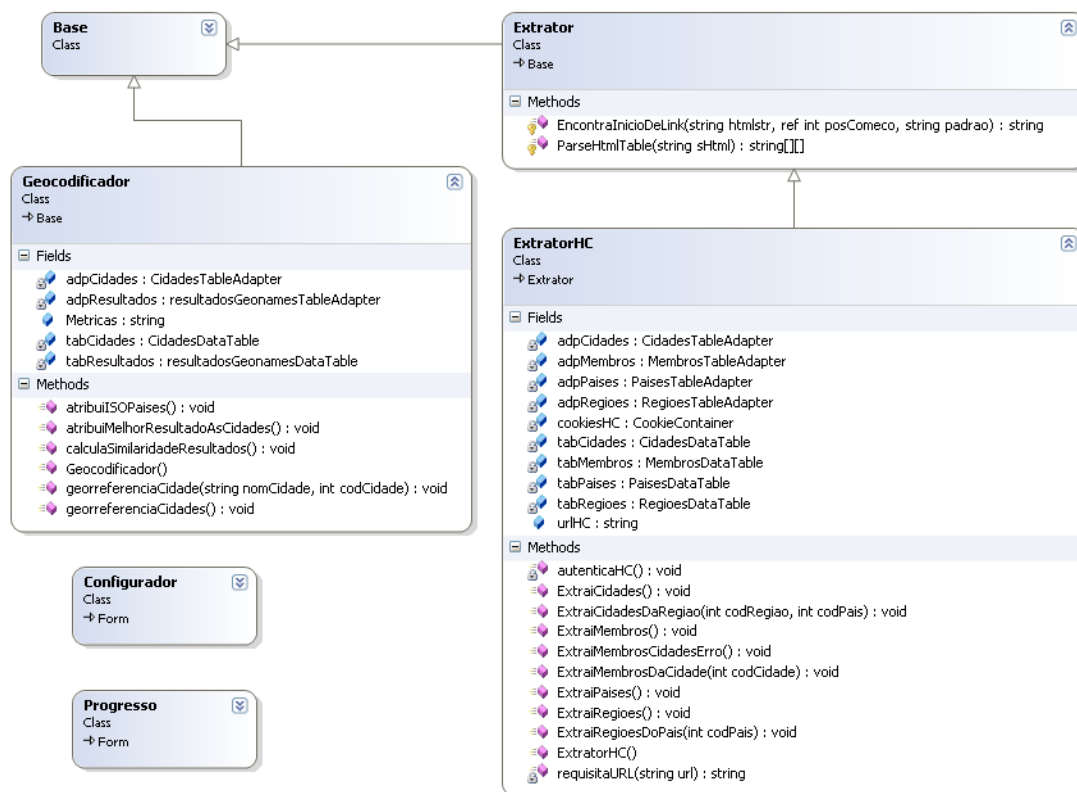


Figura 7. Diagrama de Classes do Sistema.

As classes Configurador e Progresso implementam as interfaces gráficas do sistema. A Figura 8 apresenta o formulário de configuração do sistema. A execução do sistema é acompanhada através do formulário Progresso, programado em 2 *threads* a fim de possibilitar o acompanhamento gráfico da execução.

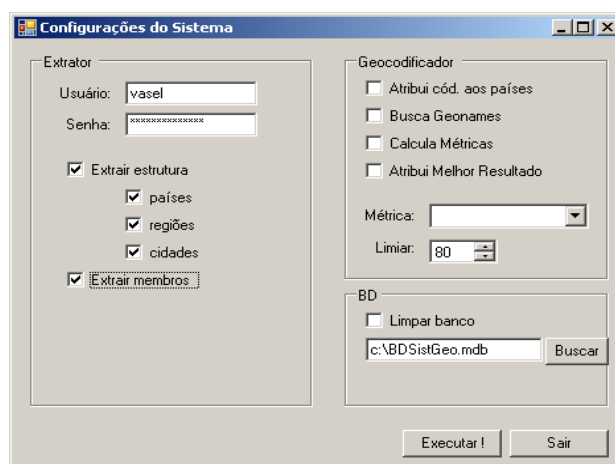


Figura 8. Interface de configuração do sistema.

O banco de dados foi implementado em Microsoft Access. As tabelas e seus relacionamentos podem ser vistos na Figura 9.

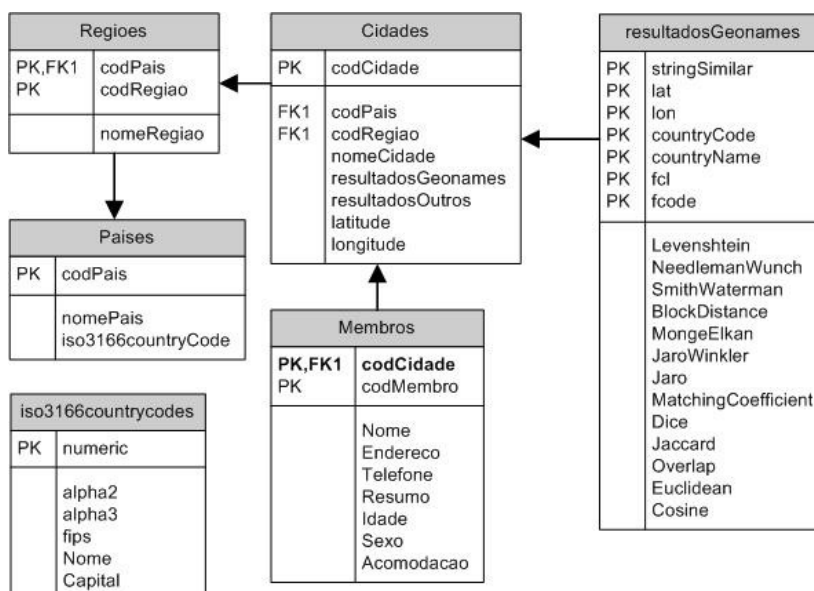
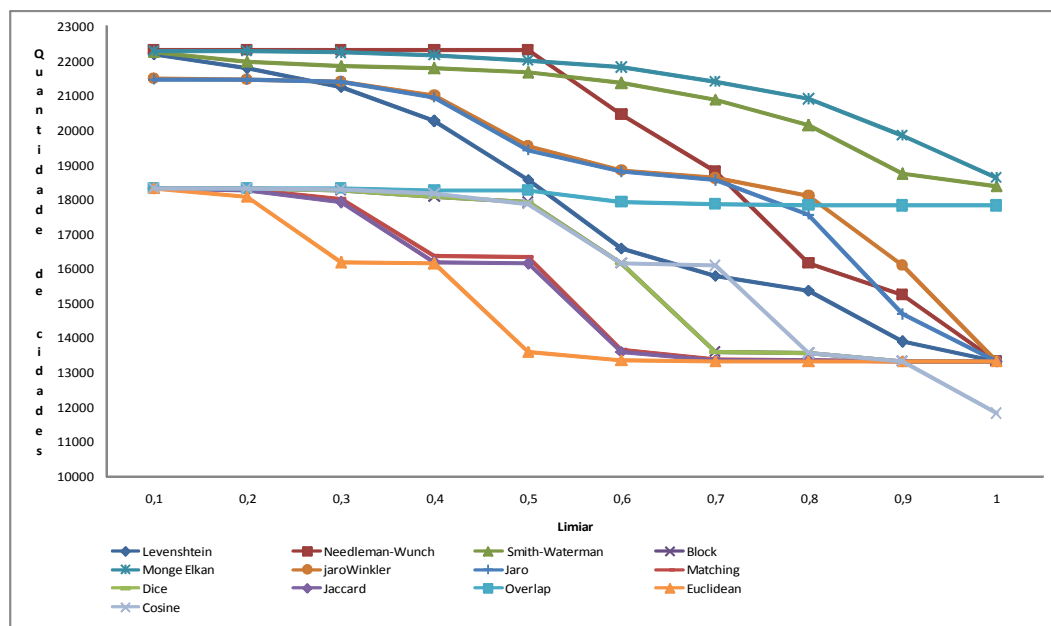


Figura 9. Tabelas do banco de dados do sistema.

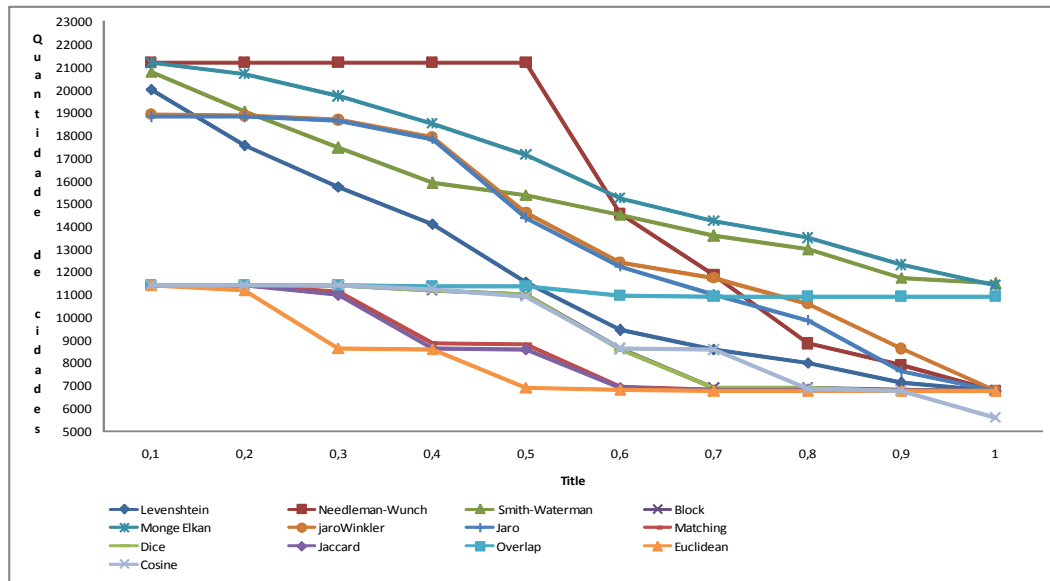
### 4.3 Testes

Após a execução do Extrator, o banco de dados ficou populado com 213 registros de países, 1908 registros de regiões, 25855 cidades e 93621 membros. O número máximo de membros por cidade foi limitado propositalmente em 50 membros, objetivando não onerar a performance nos testes.

A Figura 10 apresenta o total de cidades georreferenciadas, quando utilizada a métrica e o limiar indicados nos eixos. O comportamento obtido era o esperado : aumentado-se a exigência do limiar mínimo para aceite do resultado, a quantidade de cidades com referenciamento válido diminui. O gráfico apresentado nessa figura desconsidera a região. Já na Figura 11, a região é considerada, o que ocasiona uma redução no número de resultados, porém evitando o georreferenciamento incorreto de cidades homônimas dentro do mesmo país em regiões diferentes.

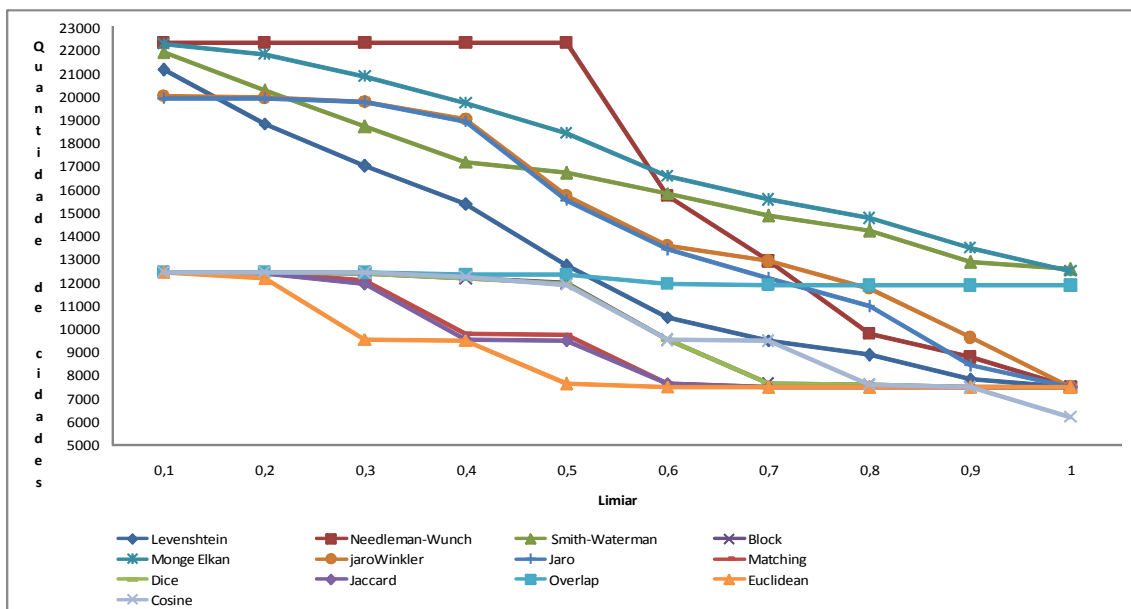


**Figura 10. Gráfico do número de cidades casadas por métrica/limiar sem considerar a região.**

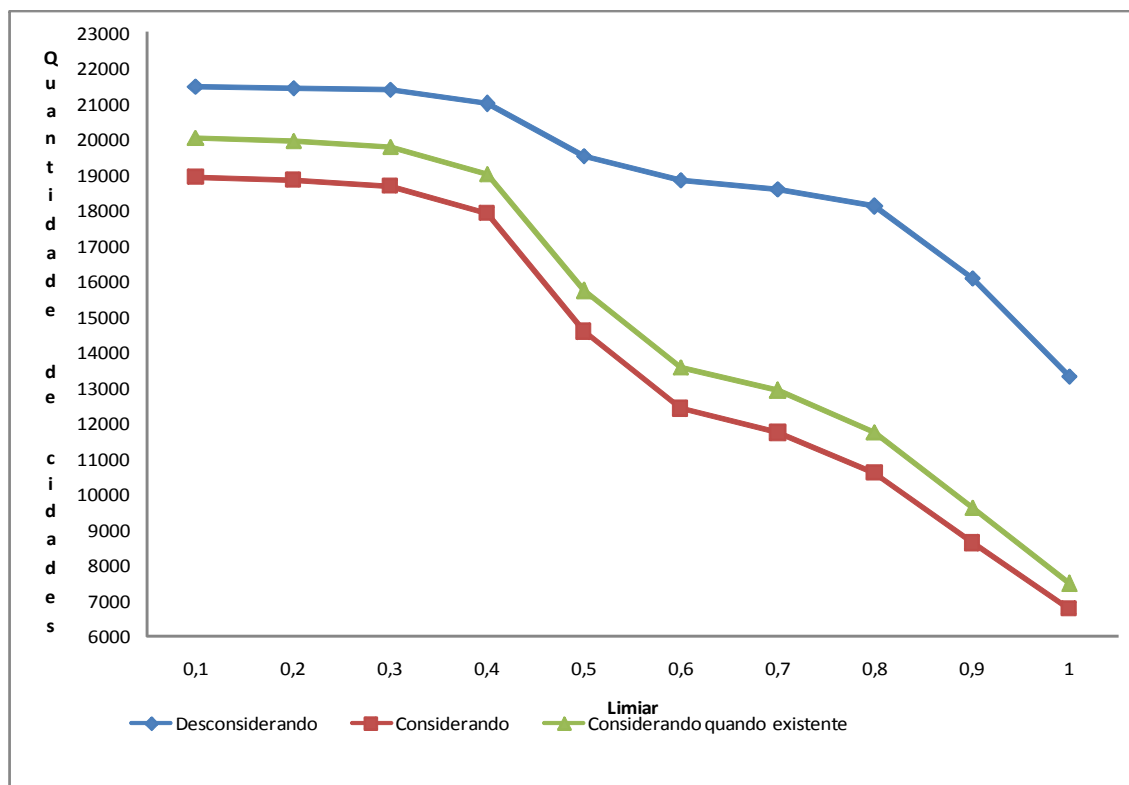


**Figura 11. Gráfico do número de cidades casadas por métrica/limiar considerando a mesma métrica/limiar para a região.**

A Figura 12 apresenta os resultados considerando a similaridade da região sempre que esta estiver presente, ao contrário do teste anterior que na ausência desta, considerava similaridade zero para esse caso. Dessa forma, evita-se os nomes de cidades homônimas, com uma redução menor no número de resultados.



**Figura 12. Gráfico do número de cidades casadas por métrica/limiar considerando a mesma métrica/limiar para a região, incluindo os resultados sem região.**



**Figura 13. Comparativo entre os três testes anteriores para a métrica Jaro-Winkler.**

A Figura 13 apresenta num mesmo gráfico os dados obtidos quando utilizada uma métrica em particular (Jaro-Winkler).

Dados conclusivos sobre a precisão do Geocodificador não foram alcançados. Foram realizados testes a partir de uma amostra de 50 cidades, tomadas ao aleatório e distribuídas pelo mundo e que representam somente 0,2 % do total de cidades. Os testes mostraram que, utilizando a métrica Jaro-Winkler, a partir do limiar 0,6 todas elas foram referenciadas corretamente. Utilizando-se o limiar 0,5, 47 delas foram referenciadas corretamente. O mesmo número foi obtido com limiar 0,4. Com limiar 0,3 45 foram referenciadas corretamente. A metodologia utilizada para esse teste não foi automatizada, razão pela qual a amostragem foi pequena.

Algumas otimizações podem ser desenvolvidas futuramente. A retirada de termos como *city*, *state*, *oblast*, *township* e outros pode melhorar a precisão e a quantidade de cidades referenciadas corretamente. Porém, tal otimização implica em tornar o georreferenciador semi-automático, pois a avaliação de quais desses tokens devem ser retirados é de difícil automatização e necessitam da intervenção de um usuário

especialista. A Tabela 2 apresenta os quinze termos mais frequentes nos nomes das regiões. Alguns deles, tais como *State* ou *County* poderiam ser retirados objetivando melhorar a precisão. Outros termos porém, são os próprios nomes das regiões e não poderiam ser retirados. Tal avaliação teria que ser realizada por um usuário especialista. A mesma análise deve ser feita para os quinze termos mais frequentes nos nomes das cidades, apresentados na Tabela 3.

**Tabela 2. Os 15 termos mais frequentes nos nomes das regiões.**

Termo	Ocorrências
of	9277
State	7932
New	1676
Bavaria	1347
de	1072
South	1029
North	978
Baden-Württemberg	882
Minnesota	816
County	806
Illinois	796
Dakota	746
Kansas	734
Pennsylvania	714
Michigan	635



**Tabela 3. Os 15 termos mais frequentes nos nomes das cidades.**

Termo	Ocorrências
of	9425
Township	5421
de	2681
County	1887
City	1809
Town	1616
Borough	743
San	734
District	679
La	410
Departamento	398
Arrondissement	390
Provincia	380
Rayon	371
New	288

## 5 Considerações finais

O sistema descrito facilita o planejamento de viagens ao apresentar dados em mapas existentes e possibilitar a navegação neste mesmo mapa. Apresenta significativa vantagem sobre os recursos e trabalhos relacionados atualmente disponíveis por inverter a ordem de navegação na busca pela informação.

Uma deficiência do sistema é a dependência de extrações regulares da lista de membros de cada cidade a fim de manter os dados apresentados sempre atualizados. Essas extrações são sempre completas por não ter sido encontrado até o momento uma forma de ler somente as informações novas.

A partir desse sistema, diversos outros trabalhos futuros podem ser realizados. Uma possibilidade é a adição de outras fontes de dados à arquitetura, dentro desse mesmo domínio, como por exemplo, dados de atrações turísticas, serviços de suporte ao viajante (*cybercafés*, escritórios de informação turística, empresas de transporte etc.) ou mesmo de outros domínios. Para realizar tal tarefa, poderia-se estudar a criação de um extrator genérico, que não esteja atrelado a estrutura específica de cada *site*, obrigando a criação de um extrator para cada novo *site* acrescentado à arquitetura. Uma métrica ou um método para avaliar a precisão de um geocodificador seria outro trabalho interessante. A adaptação a outras interfaces de visualização completaria o ciclo, possibilitando ao usuário final visualizar dados de diversas fontes na forma que preferir.

Um artigo resumindo este trabalho de conclusão de curso foi aceito e apresentado na terceira Escola Regional de Banco de Dados, realizada em abril de 2007 em Caxias do Sul. Os componentes aqui descritos estão disponíveis em <https://svn.inf.ufsc.br/vasel>, licenciados sob a Licença Pública Geral (GNU GPL).

## Referências Bibliográficas

- [1] Hospitality Club. Disponível em < [http://www.hospitalityclub.org/bpt/Hospitality\\_Exchange/hospitality\\_exchange.html](http://www.hospitalityclub.org/bpt/Hospitality_Exchange/hospitality_exchange.html) >. Acessado em: 12/07/2006 .
- [2] Especificação do formato de arquivo KML. Disponível em: < <http://earth.google.com/kml/> >. Acessado em: 12/07/2006 .
- [3] Tags KML. Disponível em: < [http://earth.google.com/kml/kml\\_tags\\_21.pdf](http://earth.google.com/kml/kml_tags_21.pdf) > . Acessado em: 12/07/2006 .
- [4] Sistema Geodésio Brasileiro. Disponível em: < [http://www.ibge.gov.br/home/geociencias/geodesia/default\\_sgb\\_int.shtm](http://www.ibge.gov.br/home/geociencias/geodesia/default_sgb_int.shtm) >. Acessado em: 12/07/2006.
- [5] Verbete Web Crawler na Wikipedia. Disponível em: < [http://en.wikipedia.org/wiki/Web\\_crawler](http://en.wikipedia.org/wiki/Web_crawler) >. Acessado em: 12/07/2006.
- [6] Cimiano, Philip e Staab, Steffen. **Learning by Googling**. Institute of Computer Science. Universidade de Karlsruhe, Alemanha.
- [7] Earth Booker. Disponível em: <<http://www.earthbooker.net/>>. Acessado em: 04/08/2006.
- [8] Global FreeLoaders. Disponível em: < <http://globalfreeloaders.com> >. Acessado em: 10/08/2006.
- [9] Projeto CouchSurfing. Disponível em: < <http://www.couchsurfing.com>>. Acessado em: 10/08/2006.
- [10] Google Maps Api. Disponível em: < <http://www.google.com/apis/maps> >. Acessado em: 10/08/2006.
- [11] Google Earth. Disponível em: < <http://earth.google.com/> > . Acessado em: 18/08/2006 .
- [12] MultiMap. Disponível em: < <http://www.multimap.com> >. Acessado em: 10/08/2006.
- [13] ISO 3166-1 alpha-2. Disponível em: <http://www.iso.org/iso/en/prods-services/iso3166ma/02iso-3166-code-lists/index.html> . Acessado em: 10/06/2007.
- [14] Tabela dos códigos do Geonames. Disponível em:<http://www.geonames.org/export/codes.html>. Acessado em: 15/06/2007.
- [15] Needleman, S.B. e Wunsch, C. D. A general method applicable to the search for similarities in the amino acid sequence of two proteins, *Journal of Molecular Biology* 48, 1970, p. 443-453
- [16] Smith, T. F. and Waterman, M. S. Identification of common molecular subsequences. *Journal of Molecular Biology* 147, 1981, p. 195-197.

- [17] Winkler, W. E. The state of record linkage and current research problems. Statistics of Income Division, Internal Revenue Service Publication R99/04, 1999.
- [18] Especificação do formato de arquivo Keyhole Markup Language. <http://earth.google.com/kml>, último acesso em: 05/01/2007.
- [19] Geonames Web Service. <http://www.geonames.org>, último acesso em: 05/01/2007.
- [20] Google Maps. <http://maps.google.com>, último acesso em: 05/01/2007.
- [21] Earth Booker. <http://www.earthbooker.net>, último acesso em: 05/01/2007.
- [22] Biblioteca livre de algoritmos de similaridade Simmetrics. <http://sourceforge.net/projects/simmetrics>, último acesso em: 05/01/2007.
- [23] V. I. Levenshtein, Binary codes capable of correcting spurious insertions and deletions of ones, Problemy Peredachi Informatsii, vol. 1, no. 1, p. 12-25, 1965.
- [24] Jaro, M. A. Advances in record-linkage methodology as applied to matching the 1985 census of Tampa, Florida. Journal of the American Statistical Association 84, p. 414–420, 1989.
- [25] Kondrak, G., Marcu, D. and Knight, K. Cognates Can Improve Statistical Translation Models. Proceedings of HLT-NAACL 2003: Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, p. 46-48, 2003

## **Anexo I – Código Fonte do Sistema**

(O código fonte completo do sistema está anexado/incluído separadamente em um arquivo compactado chamado FontesTCCVasel.rar)

## **Anexo II – Artigo**

(O artigo este está anexado / incluído no CD-ROM separadamente.)