WIKIGrep
Búsquedas avanzadas en la Wikipedia

Introducción

- Wikipedia: enciclopedia libre
  - Entre los 10 sitios más visitados en Internet (Alexa.com)
  - Crecimiento exponencial
  - Actualmente: cerca de dos millones de artículos
    - Formato XML; llegan a pesar hasta 1TB
- Motores de búsqueda Web tradicionales no permiten búsquedas basadas en expresiones regulares
  - Consumen muchos recursos
- Problema: Imposible realizar búsquedas avanzadas (basadas en expresiones regulares) en la Wikipedia

Objetivos

- Utilizar técnicas de procesamiento masivo de datos para realizar búsquedas avanzadas de texto dentro de documentos
- Utilizar expresiones regulares para la búsqueda de texto dentro de documentos y de esta manera mostrar resultados más precisos
- Diseñar una interfaz Web que permita ingresar búsquedas basadas en expresiones regulares, y muestre los resultados obtenidos del procesamiento masivo de la misma, de una manera entendible al usuario

Dataset de la Wikipedia

Mantiene historial completo de las revisiones

Links representados por [[destination]]

La entidad <text> .. </text> contiene el contenido de los artículos


```xml
- <mediawiki xml:lang="en">
  - <page>
      <title>Page title</title>
      <restrictions>edit=sysop:move=sysop</restrictions>
      <revision>
        <timestamp>2001-01-15T13:15:00Z</timestamp>
      - <contributor>
          <username>Foobar</username>
        </contributor>
        <comment>I have just one thing to say!</comment>
        <text>A bunch of [[text]] here.</text>
        <minor/>
      </revision>
    - <revision>
        <timestamp>2001-01-15T13:10:27Z</timestamp>
      - <contributor>
          <ip>10.0.0.2</ip>
        </contributor>
        <comment>new!</comment>
        <text>An earlier [[revision]].</text>
      </revision>
    </page>
  - <page>
      <title>Talk:Page title</title>
    - <revision>
        <timestamp>2001-01-15T14:03:00Z</timestamp>
      - <contributor>
          <ip>10.0.0.2</ip>
        </contributor>
        <comment>hey</comment>
        <text>WHY YOU LOCK PAGE??!!!</text>
      </revision>
    </page>
  </mediawiki>
```

Diseño e implementación

Herramientas utilizadas

- Hadoop
  - Procesamiento distribuido
  - Gracias a grant de Amazon Web Services, posible levantar clústers bajo demanda usando los siguientes serivicios:
    - Elastic MapReduce (EMR), o
    - Elastic Computing Cloud (EC2)
- AWS Simple Storage Service (S3)
  - Almacenamiento escalable de datos

Análisis de las alternativas

- Los costos detallados en esta tabla fueron obtenidos de la Calculadora de Costos AWS, y están actualizados al 25 de agosto de 2009.

Número de Gigabytes almacenados en S3:	23
Costo por instancia en EC2	0,2
Recargo por uso de Elastic MapReduce, por hora*instancia	0,03
Data transfer in estimado (en GB)	1
Data transfer out estimado (en GB)	1
Duración de una consulta	1

Análisis de las alternativas (cont.)

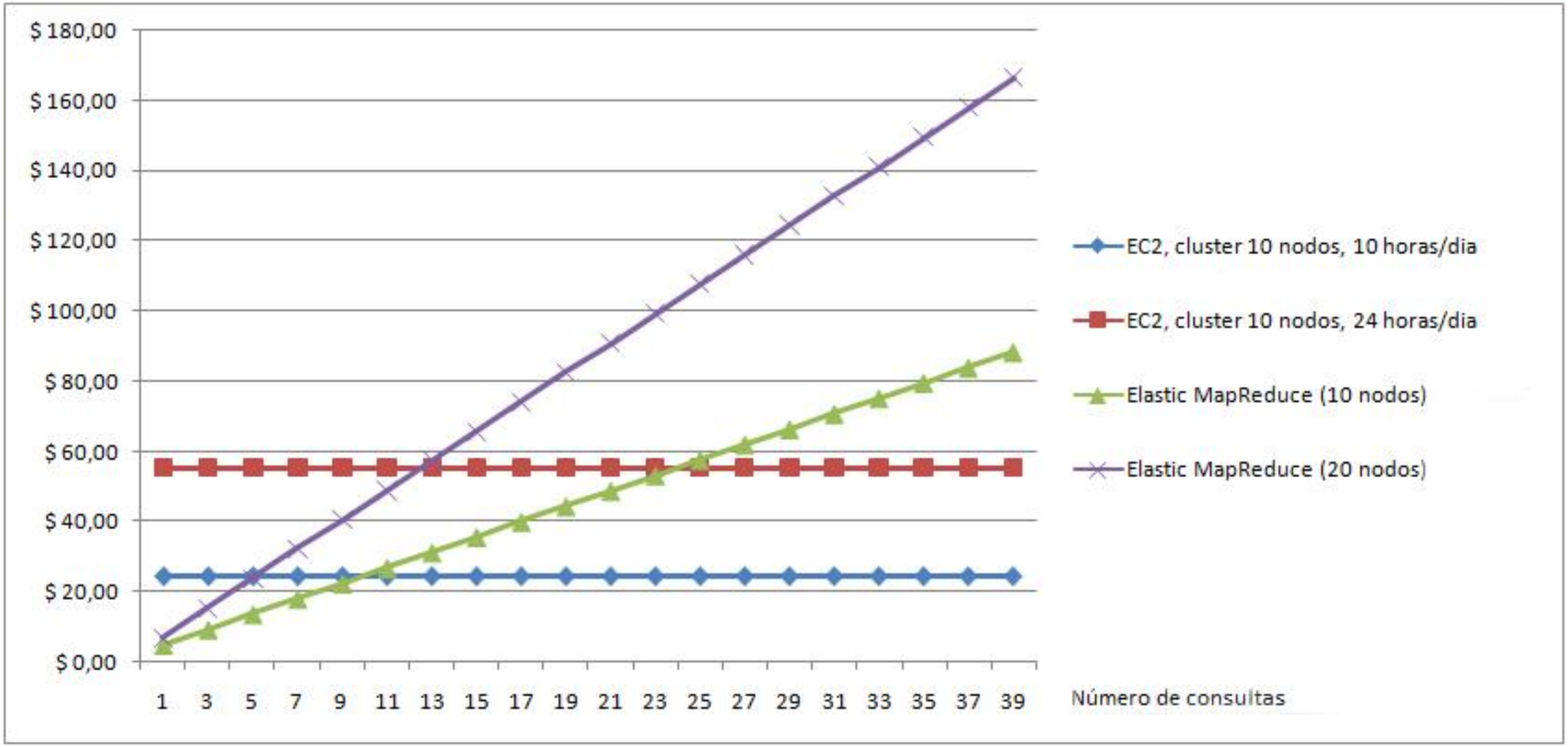


Diagrama de la solución

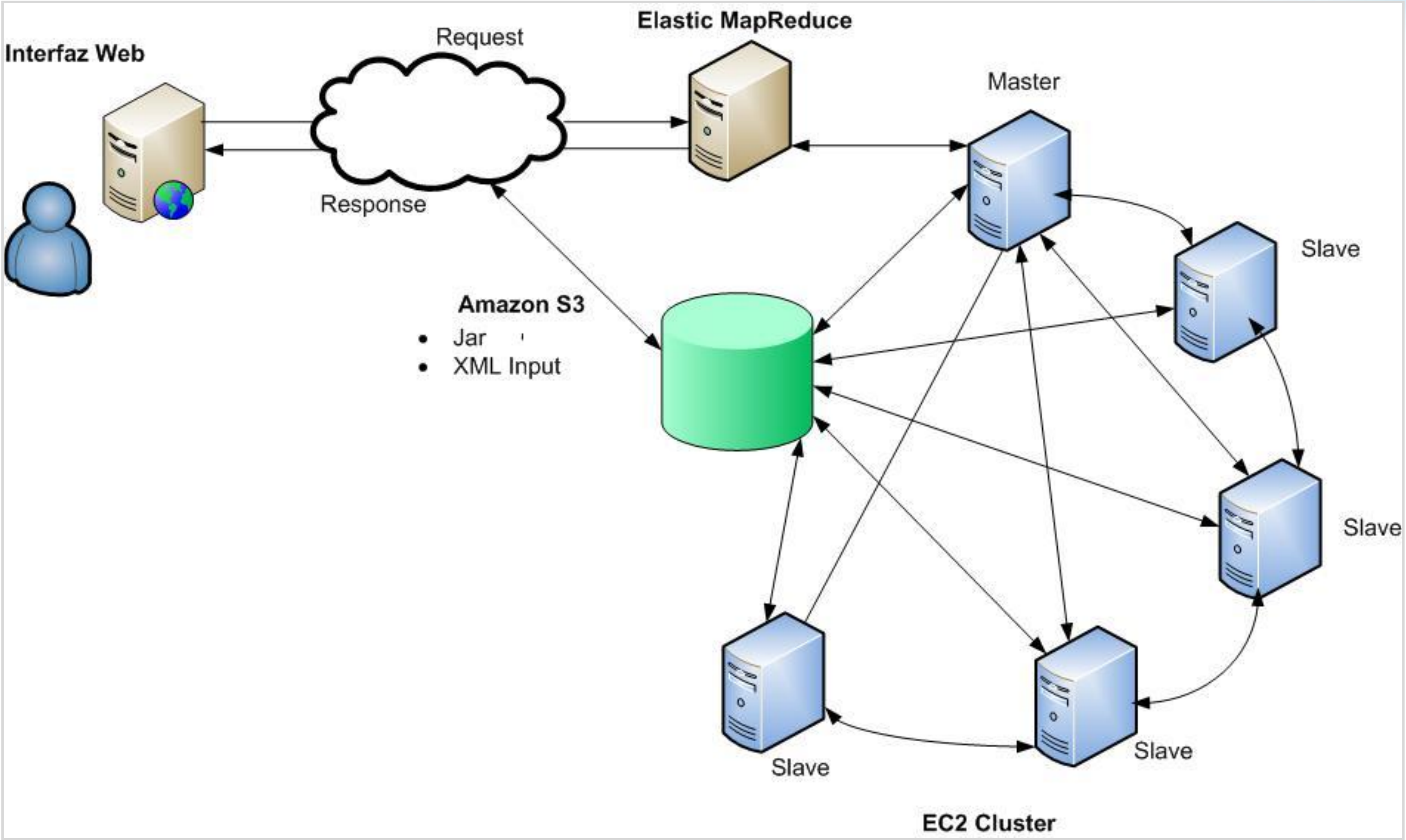
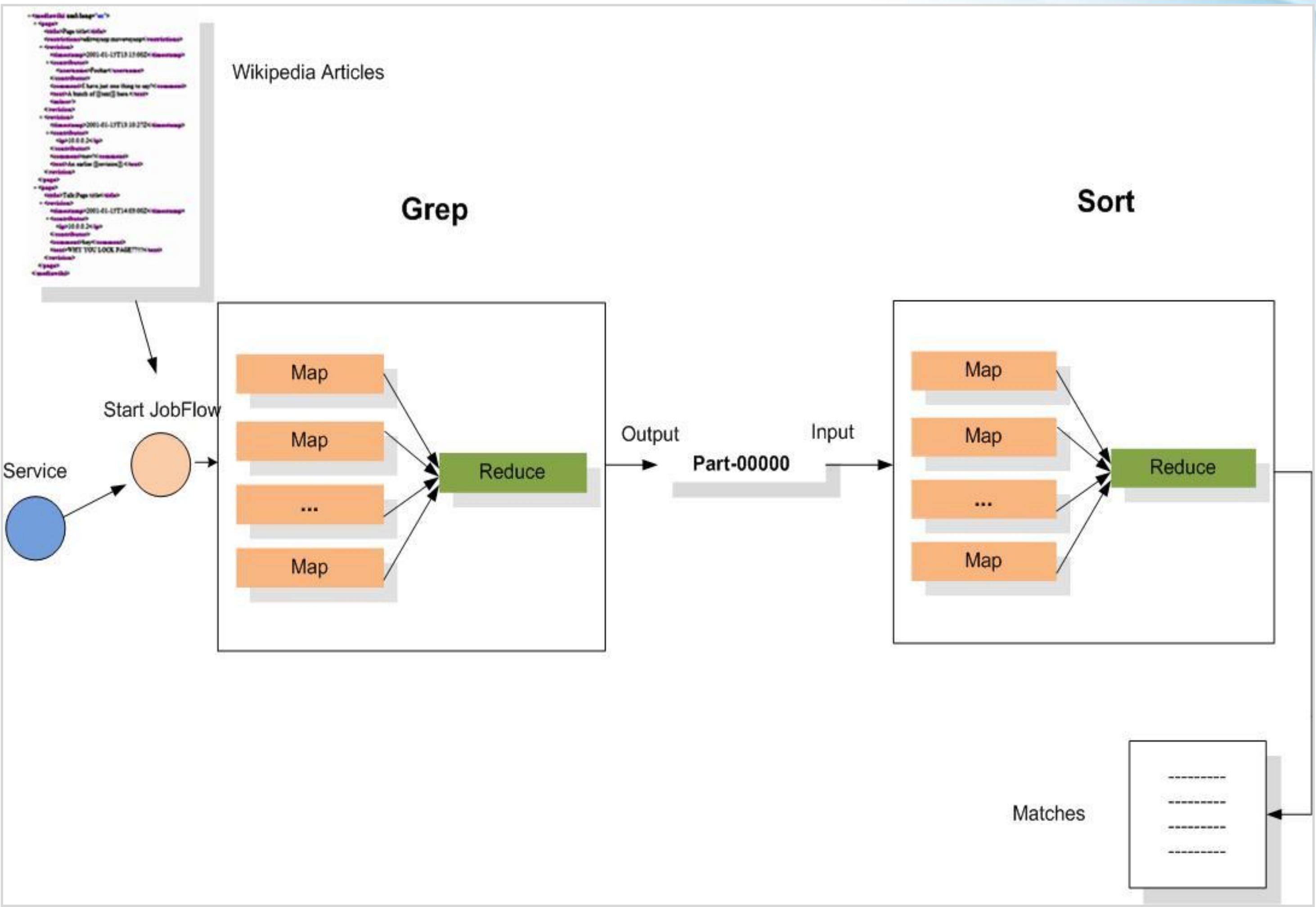


Diagrama del Algoritmo MapReduce



Pseudocódigo MapReduce

GrepMapper

(docid,wikipediaPages)->[(match,[docid,1])]

GrepReducer

(match,[docid1,docid2,docid1…])->[(match,[docid1,n1]),

(match,[docid2,n3]),…..]

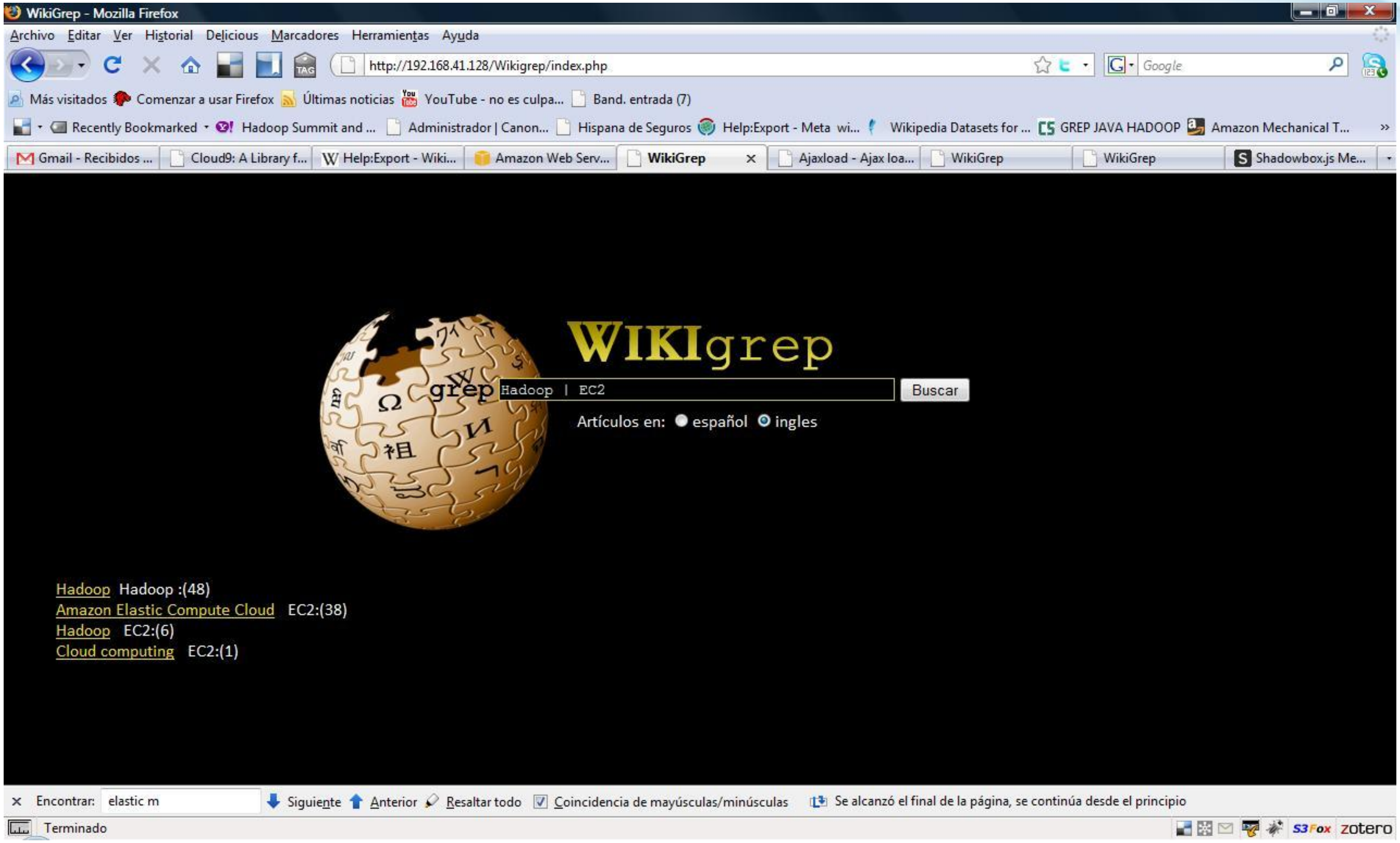
SortMapper

(docid,n) -> [(n,docid)]

SortReducer

(n,docid) -> [(docid,n)]

Diseño de la Interfaz



Resultados

Patrones de Búsqueda

- Se seleccionó tres expresiones regulares
  - Interpretados usando el paquete java.util.regex

#	Expresión	Descripción	Coincidencias
1	“(\\d\\d\\d\\d)-(\\d\\d\\d\\d)”	Hallar fechas dentro de un rango	Antonio Lucio Vivaldi (1678)-(1741)
2	“[^\\s]*(less|ness|able)”	Hallar palabras con los sufijos less, ness y able.	Sleepless, capable, greatness
3	“(.)(.).\\2\\1”	Hallar palabras palíndromas de 5 letras	Radar, kayak, level

Resultados

Expresión	# nodos	# de Mappers	# de Reducers	GB	Tiempo Ejecución
1	10	16	1	23	10 min
2	10	16	1	23	11 min
3	10	16	1	23	10 min

#	Expresión	Descripción	Coincidencias
1	“(\\d\\d\\d\\d)-(\\d\\d\\d\\d)”	Hallar fechas dentro de un rango	Antonio Lucio Vivaldi (1678)-(1741)
2	“[^\\s]*(less|ness|able)”	Hallar palabras con los sufijos less, ness y able.	Sleepless, capable, greatness
3	“(.)(.).\\2\\1”	Hallar palabras palíndromas de 5 letras	Radar, kayak, level

Análisis Comparativo

Expresión	Google	Wikipedia Search	Wikigrep
“(\\d\\d\\d\\d)-(\\d\\d\\d\\d)”	No es posible	No es posible	Si
“[^\\s]*(less|ness|able)”	No es posible	Es posible por el uso de wildcards ejm: *less, *ness, *able lo que representa un total de 3 consultas	Si
“(.)(.).\\2\\1”	No es posible	No es posible	Si

- Wikigrep es capaz de procesar las 3 consultas anteriores en un tiempo de 10 minutos para un dataset de 23 GB
  - Tiempo puede ser reducido a menos de 5 minutos si se utiliza un clúster EC2 (en lugar de EMR) ya que al usar EMR se debe levantar y bajar el clúster por cada consulta

Conclusiones y Recomendaciones

Conclusiones

- El uso de computación distribuida se vuelve cada vez más popular
  - Sin embargo, subir los datos a la nube es aún un problema
    - Limitaciones como el ancho de banda del usuario
- Al trabajar con clústers para el procesamiento masivo de datos, se puede llegar a reducir los tiempos de procesamiento de los mismos considerablemente
  - En nuestro caso, búsquedas tomaron menos de 15 minutos vs. un grep tradicional que hubiera tomado varias horas de procesamiento

Conclusiones

- El desarrollo del Wikigrep distribuido fue un éxito y atribuimos este suceso ha distintas razones:
  - El diseño de la solución usando como EMR facilita levantar un clúster EC2
  - La librería cloud9 que facilito el uso y manipulación de los artículos de la Wikipedia
  - La optimización realizada mediante la investigación en el uso del número adecuado de mappers y reducers
- Uso de expresiones regulares permite buscar patrones exactos en un documento y su buen uso puede llegar a ser una gran herramienta para el usuario
  - Servicios de cloud computing facilitan el desarrollo de herramientas de este tipo, a bajo costo

Recomendaciones

- Recomendamos el uso de datasets de la Wikipedia comprimidos mejorando así los tiempos transmisión
- Al momento, EMR no soporta el uso de los dataset públicos de la Wikipedia (disponibles de manera gratuita gracias a Amazon), por lo que podría usarse EC2 que sí los soporta
  - Evitaría subir los datos a S3
- Utilizar algún algoritmo como Page Rank que ordernar los resultados en base a relevancia, importancia, etc.

¿Preguntas?