

Living on the Edge

Edge-Side Includes and  
Uncacheability

Chalu Kim

eGenius

## CACHE OR NO-CACHE

- In February 1999, it is estimated that the publicly indexable Web contains approximately 800 million pages, which is more than double their previous estimate in December 1997 of 320 million pages. A recent estimate in July 2000 reported that the current size of the Web is 2.1 billion unique pages and the growth rate is 7.3 million additional Web pages per day.
- During the 1998 World Cup between April and July, the site received 1,352,804,107 requests. During the collection period, there were 33 different World Cup HTTP servers at four geographic locations: Paris, France; Plano, Texas; Herndon, Virginia; and Santa Clara, California.
- On Monday, September 18, 2000, Olympics.com received 683 million hits.

## World-Wide Wait

- In this day of broadband speed, one of the most common end-user desires is for more speed.
- Some 30+% of online users are retrieving pages from Internet caches.
  - as of December 10, 1999, 17.5% of users are accessing the Web behind proxy servers (based on sample of 35M users). This represents an increase of over 15% since April 1999 (when approximately 15% of users were behind proxies).
- In July 2, 2004, Akamai had DNS break-down and the world experienced outage.

## Internet cache

- Internet cache (or reverse proxy) stores and forwards content to users.
- Caches store pages and images, content objects that are mostly static.
- Interesting pages are dynamically generated and are not cached.
- Personalized pages are dynamically generated and are not cached.

## Server-side caching

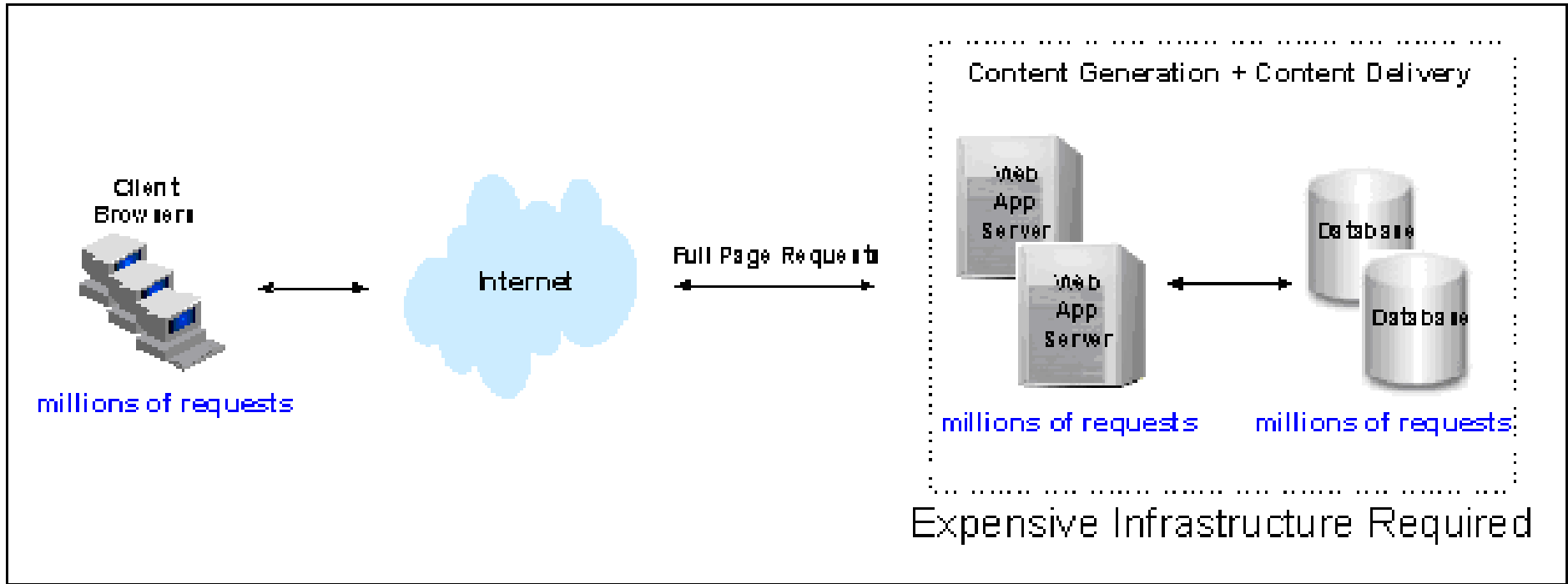
- Scripting languages like Active Server Pages and JavaServer Pages need to be executed serially.
- On the server-side, many application servers perform limited caching of templates, and HTML fragments, SQL query results.
- Zope uses disk caching, RAM caching, SQL caching and still performance is an issue when scaling to massive traffic.

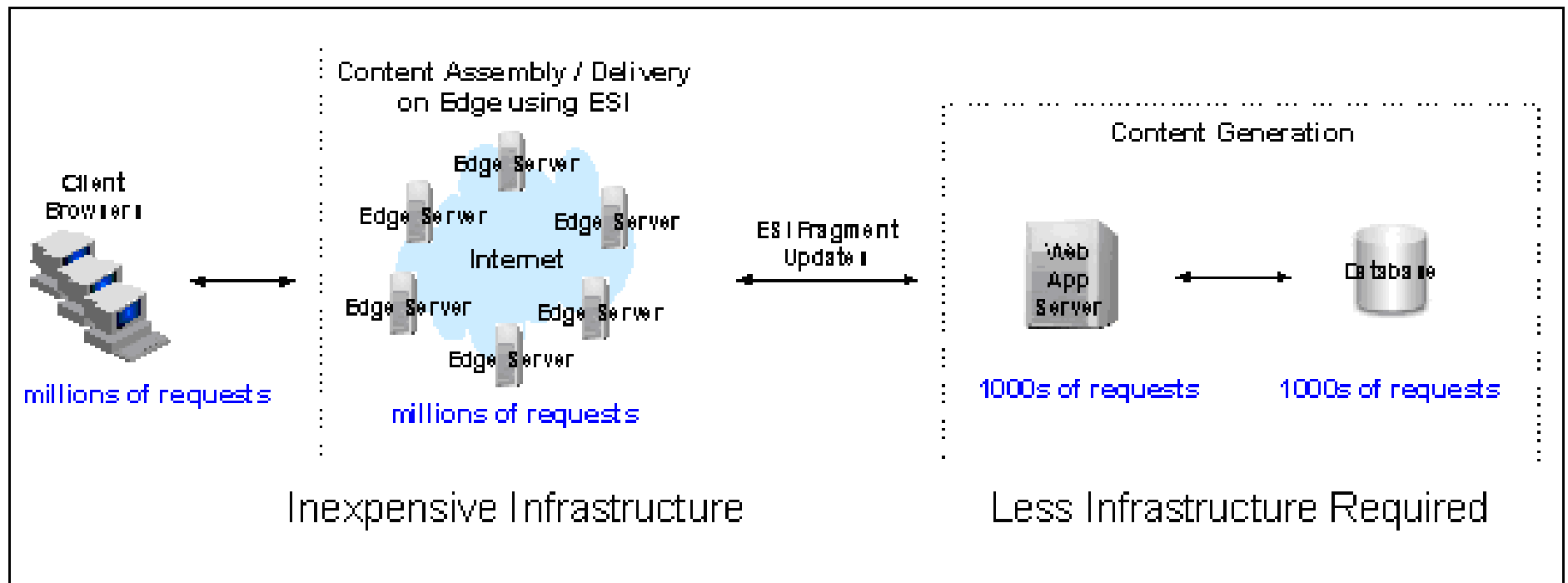
## External Caching Service

- Using Web server plug-in with ESI
- proprietary fragment accelerator
- proprietary page cache

## Enter Edge-Side Includes

- Around 2001, companies like Akamai and Oracle endorsed and implemented ESI.
- ESI provides additional ways for the programmer to tell Internet caches (rather sophisticated ones) what and how to cache.
- Instead of creating your own expensive web farm, you use other expensive web farms like Akamai.
- In July 2, 2004, Akamai had DNS break-down and the world experienced outage bringing down the expensive web farms. Akamai became a single point of failure.





Template



Fragments



```
< table >
< tr >
< td colspan= "2" >
< esi:try >
< esi:attempt >
< esi:include src= "http:// myserver/ news"
                onerror= "continue" / >
< / esi:attempt >

< esi:except >
< !- - esi commented out esi - - >
< / esi:except >
< / esi:try >
< / td > < / tr >
< / table >
```

## Squid and ESI

- In 2002, ESI development on Squid was funded by Zope.
- Initial and "nasty" part of ESI implementation is done.
- Squid is a popular reverse proxy as well as accelerator.
- Squid is used by numerous free network of caches such as the IRCache Mesh.

## Squid – open-source caching software

- a full-featured Web proxy cache
- designed to run on Unix systems
- free, open-source software
- the result of many contributions by unpaid (and paid) volunteers
- proxying and caching of HTTP, FTP, and other URLs
- proxying for SSL
- cache hierarchies
- ICP, HTCP, CARP, Cache Digests
- transparent caching
- WCCP (Squid v2.3 and above)
- extensive access controls
- HTTP server acceleration
- SNMP
- caching of DNS lookups

## Uncacheability

- Uncacheability is a term to describe what portion of your web site is not cachable. It is also a factor to multiply into your log count because hits recorded are uncached hits directly by users or Internet caches.
- Cacheability test

# Forced caching

```
refresh_pattern \.gif$ 10080 90%  
refresh_pattern \.xml$ 10080 50%  
refresh_pattern \.zip$ 10080 90%
```

# ESI Language Specification 1.0

Namespace - `http://www.edge-delivery.org/esi/1.0`

```
<esi:try> Invalid markup here <esi:attempt>
```

```
  <esi:include ... >
```

```
  This line is valid and will be processed.
```

```
  </esi:attempt> Invalid markup here
```

```
<esi:except>
```

```
  This HTML line is valid and will be processed.
```

```
  </esi:except> Invalid markup here
```

```
</esi:try>
```

## ESI Language Specification 1.0

```
<esi:include src="http://example.com/1.html"  
alt="http://bak.example.com/2.html"  
onerror="continue"/>
```

```
<esi:include src="http://example.com/  
search?query=$(QUERY_STRING{query})"/>
```

```
<esi:inline name="URI" fetchable="{yes | no}">  
    fragment to be stored within an ESI processor  
</esi:inline>    Note inline is not available in Squid 3.
```

## ESI Language Specification 1.0

```
<esi:choose>
  <esi:when test= "$ (HTTP_COOKIE{group })
== 'Advanced'">
    <esi:include
src= "http://www.example.com/advanced.html"/ >
  </esi:when>
  <esi:when test= "$ (HTTP_COOKIE{group })= 'Basic
User'">
    <esi:include
src= "http://www.example.com/basic.html"/ >
  </esi:when>
  <esi:otherwise>
    <esi:include
src= "http://www.example.com/new_user.html"/ >
  </esi:otherwise>
</esi:choose>
```

# ESI Language Specification 1.0

```
<esi:try>
  <esi:attempt>
    <esi:comment text="Include an ad"/>
    <esi:include
src="http://www.example.com/ad1.html"/>
  </esi:attempt>
  <esi:except>
    <esi:comment text="Just write some HTML
instead"/>
    <a
href=www.akamai.com>www.example.com</a>
  </esi:except>
</esi:try>
```

## ESI Language Specification 1.0

```
<esi:comment text="the following animation will have  
a 24 hr TTL." />
```

```
<esi:include  
src="http://www.example.com/ad.html"/>  
<esi:remove>  
  <a  
href="http://www.example.com">www.example.com<  
/a>  
</esi:remove>
```

# ESI Language Specification 1.0

```
<esi:vars>
```

```
  
```

```
</esi:vars>
```

```
<!-- esi
```

```
  <p><esi:vars> Hello, $(HTTP_COOKIE{name})!
```

```
</esi:vars></p>
```

```
-->
```

## ESI Variables

ESI 1.0 supports the following read-only variables, which are based on the client's HTTP request line and headers:

Variable names are always uppercase. To reference a variable, surround the name with parenthesis and append a dollar sign (\$).

For example:

`$(HTTP_HOST)`

## Variable Substructure Access

By default, ESI variables are evaluated in a string context. However, some that represent more complex data will make automatically parsed and typed data available.

To access a variable's substructure, the variable name should be appended with braces containing the key which is being accessed. For example,

```
$(HTTP_COOKIE{username})
```

- Dictionary, List, Default

## ESI Expressions

Conditional elements use expressions (in their test attributes) to determine how to apply the contained elements. Expressions consist of operators, variables and literals, and evaluate to true or false.

Single quotes are used within an expression to delimit literals. For example,

```
$(HTTP_HOST) == 'example.com'
```

Whitespace is optional around all operators, except `,` which must be surrounded by whitespace.

## Edge Architecture Specification

Surrogate-Control: no-store, content="ESI/ 1.0"

Surrogate-Control: max-age=60;abc, max-age=300

Surrogate-Control: content="ESI/ 1.0";abc,  
content="ESI-Inline/ 1.0";def

Surrogate-Control: max-age=60, no-store-remote;abc

# ESI Invalidation Protocol 1.0

ESI Invalidation protocol allows invalidation of ESI cache objects. This allows flushing cached objects at will.

## Request Example

```
POST /x-invalidate HTTP/1.0
Authorization: Basic aW52YWxpZGF0b3I6aW52YWxpZGF0b3I=
Content-Length: 217
<?xml version="1.0" ?>
<!DOCTYPE INVALIDATION SYSTEM "invalidation.dtd">
<INVALIDATION VERSION="WCS-1.0">
<OBJECT>
<BASICSELECTOR URI="/cache.htm" />
<ACTION />
</OBJECT>
</INVALIDATION>
```

# ESI Invalidation Protocol 1.0

## Response Example

```
HTTP/1.1 200 OK
Date: Sun, 22 Apr 2001 07:54:09 GMT
Allow: GET, HEAD
Server: Webserver/2.0.0.0.0
Content-Type: text/html
Content-Length: 284
<?xml version="1.0"?>
<!DOCTYPE INVALIDATIONRESULT SYSTEM "invalidation.dtd">
<INVALIDATIONRESULT VERSION="WCS-1.0">
<OBJECTRESULT>
<BASICSELECTOR URI="/cache.htm"/>
<RESULT ID="1" STATUS="SUCCESS" NUMINV="1"/>
</OBJECTRESULT>
</INVALIDATIONRESULT>
```

# ESI Invalidation Protocol 1.0

Advanced

URIPREFIX (required) , URIEXP , HOST , METHOD , BODYEXP , COOKIE ,  
HEADER , OTHER

The advanced selector is more sophisticated than the basic selector. Its descriptive capability in URLs is as powerful as regular expression itself. In fact, a basic selector can be expressed in the form of the advanced selector.

For example, suppose the URI in basic selector is " / p1/ p2/ p3/ file.htm ", then in the equivalent advanced selector, the URIPREFIX is " / p1/ p2/ p3/ " and the URIEXP is " ^/ p1/ p2/ p3/ file.htm\$ ". If you choose to specify " / " for the URIPREFIX, it is still okay. Since the regular expression is done against the set of page objects containing the common URIPREFIX path prefix, it is obvious that the smaller the set, the more efficient the invalidation.

## **JESI - Tags for Edge-Side Includes in JSP**

`<jesi:template>`  
`<jesi:include>`  
`<jesi:fragment>`  
`<jesi:codeblock>`  
`<jesi:control>`  
`<jesi:invalidate>`  
`<jesi:personalization>`

# JESI - Tags for Edge-Side Includes in JSP

```
<html>
  <body>
    <jesi:include page="stocks.jsp" flush="true"
  />
    <p>
  <hr>
    <jesi:include page="/weather.jsp"
  flush="true" />
    <p>
  <hr>
    <jesi:include page="../sales.jsp" flush="true"
  />
  </body>
</html>
<%
  else {
    // a known customer; trying to retrieve
  recommended products from profiling
    String recommendedProductsDescPages[]
  = ProfileUtil.getRecommendedProductsDescUR
  L(customerId);
    for (int i=0;
  i<recommendedProductsDescPages.length;
  i++) {
    %>
    <jesi:include page="<%
  =recommendedProductsDescPages[i]%>" />
    }
  }
  %>
  %>
  String
```

## JESI - Tags for Edge-Side Includes in JSP

```
<html>
<jesi:control cache="no" />
<jesi:include page="a.jsp?p1=v1" />
<h3>hello ...</h3>
<jesi:include page="b.jsp" />
<h3>world ...</h3>
<jesi:include page="c.jsp?p1=v2"
copyparam="true" />
</html>
```

```
<jesi:template expiration="3600">
HTML #1
<% // some Java code got always executed
Java-code-#1 %>
<jesi:fragment expiration="60">
JSP code block #1
</jesi:fragment>
HTML #2
<jesi:fragment>
JSP code block #2 using default expiration
</jesi:fragment>
HTML #3
<jesi:fragment expiration="600">
JSP code block #3
</jesi:fragment>
HTML #4
</jesi:template>
```

# JESI - Tags for Edge-Side Includes in JSP

1. Invalidate a single object in the default ESI processor:

```
<jesi:invalidate  
  <jesi:object uri= "/ images/ logo.gif"/ >  
</jesi:invalidate>
```

2. Invalidate all objects in the default ESI processor:

```
<jesi:invalidate  
  <jesi:object uri= "/" prefix= "yes"/ >  
</jesi:invalidate>
```

3. Invalidate a single object but allow it to be served stale for upto 30 mins:

```
<jesi:invalidate  
  <jesi:object uri= "/ images/ logo.gif"  
maxRemovalDelay= "1800"/ >  
</jesi:invalidate>
```

4. Invalidate a multi-version object:

```
<jesi:invalidate  
  <jesi:object uri= "/ page.htm">  
    <jesi:cookie name= "user_type"  
value= "Customer"/ >  
  </object>  
</jesi:invalidate>
```

# Zope page template

esi\_header – python script

```
request = container.REQUEST
RESPONSE = request.RESPONSE
RESPONSE.setHeader('Surrogate-
Control','max-age=30+60,content="ESI/1.0"')
```

Page template

```
<table tal:define="dummy
here/set_esi_header" >
<tr>
<td colspan="2">
<esi:try>
<esi:attempt>
<esi:include
src="top.html"
tal:attributes="src string:$
{here/absolute_url}/top.html"
onerror="continue" />
</esi:attempt>

<esi:except>
<!-- esi
This spot is reserved for something
-->
</esi:except>
</esi:try>
</td></tr>
<tr><td>esi test1</td></tr>
</table>
```

# Reference

- <http://www.web-caching.com/proxy-caches.html>
- <http://www.ircache.net/>
- <http://www.david-guerrero.com/papers/squid>
- <http://vancouver-webpages.com/CacheNow/>
- <http://www.web-caching.com/cdns.html>
- <http://www.measurementfactory.com/>
- <http://www.squid-cache.org/related-software>
- <http://www.eecs.harvard.edu/~vino/web/user>
- <http://www.squid-cache.org>
- <http://www.linofee.org/~jel/proxy/Squid/icp->
- <http://www.esi.org>
- <http://www.egenius.com/Members/chalu/squid>
- <http://www.egenius.com/Members/chalu/zope>
- [http://www.joachim-bauch.de/tutorials/squid  
- ESI on Squid 3 using libxml](http://www.joachim-bauch.de/tutorials/squid-ESI-on-Squid-3-using-libxml)