Varnish作为缓存服务提供者，本身没有内容，所有内容都是从后端的服务器或其它缓存服务器上取过来的。当varnish收到这些内容之后，需要根椐一定的策略确定是否缓存它，如果需要缓存，还需要确定可以缓存的时间。  
  
1、varnish缓存策略  
  
缺省是根椐后端返回的http状态码决定是否缓存。可以缓存的状态码如下：  
200     
203     
300     
301     
302     
410     
404     
  
varnish现在还不支持ranges请求，所以不会缓存206状态的结果。  
  
管理员可以在vcl的配置文件中的vcl\_fetch部分，加入自已的缓存策略，方法是修改beresp.cacheable变量。下面的例子，看后端返回中如果设置了cookie，则不缓存：  
  
sub fetch {  
  
    if (beresp.http.Set-Cookie) {  
        set beresp.cacheable = false;  
    }  
}  
  
2、缓存时间的计算方法  
  
缺省的缓存时间计算方法如下：  
  
. 先根椐max-age确定缓存时间  
  
看返回头中的Cache-Control头中，有没有指定s-maxage或max-age信息，另外，如果后端是一个缓存服务器，它还会指定一个age头，表明已经缓存过的时间，这样需要减去age头指定的时间，计算出对应的缓存时间。  
  
例1 只有max-age，没有age  
  
CacheControl: max-age=86400  
  
缓存时间：86400秒，即一天  
  
例2 同时有max-age和age信息  
  
CacheControl: max-age=86400  
Age: 6400  
  
缓存时间：86400 - 6400 = 80000秒  
  
. 如果没有max-age信息，则根椐Expires头部计算缓存时间  
  
先看有没有Date头，  
   如果没有Date头  
      看expires指定的时间是否小于收到数据的时间  
         小于，表明从后端取得内容已经过期，缓存时间为0，不缓存  
         >=，则缓存到Expires指定的时刻再过期  
   如果有Date头  
      看Expires指定的时间是否小于Date指定的时间  
         小于，取到已经过期的内容，缓存时间为0，不缓存  
         >=，则看Date时间跟取到内容的时间差别(判断本机和后端的时钟差异)  
           小于clock\_skew(缺省是10秒，管理员可以设置别的数值)，以本机时间为准  
              看Expires指定的时间是否小于收到数据的时间  
                 小于，表明从后端取得的内容已经过期，缓存时间为0，不缓存  
                 >=，则缓存到Expires指定的时刻再过期  
           >= clock\_skew，以后端的时间为准  
             缓存时间为Expires - Date的差值  
  
. 如果既没有max-age信息，又没有指定Expires过期信息  
  
则按default\_ttl(缺省是2分钟，管理员可以设置成别的数值)设置缓存时间。  
  
  
管理员可以在vcl的配置文件中的vcl\_fetch部分，手工指定缓存时间，方法是修改beresp.ttl变量。下面的例子，将缓存时间设置成20分钟(1200秒)：  
  
set beresp.ttl = 1200s;  
  
3、给用户返回已经过期的对象  
  
varnish内部有一个grace模式。当后端不可用，或者已经向后端发过更新请求的情况下，别的客户再请求这个对象时，会收到已经过期的版本，当然过期的时间在可接受的范围内。  
  
varnish缺省可接受的过期时间范围是不超过10秒，可以修改params中的default\_grace参数。修改这一数值，但这样的改法相当于改了全局的参数，适应于所有请求。  
  
如果想只修改特定的url对应的对象的grace时间范围，则需要修改vcl配置，改两个地方。首先需要修改vcl\_recv过程，在收到用户请求时指定可以接受过期对象的过期时间范围。其次需要修改vcl\_fetch过程，从后端取到内容时，设置对象过期后还可以保留的时间，这样以便在过期后不会立即被后台线程清理掉。  
  
下面的例子针对特定的请求，将接受对象的grace时间设置为2分钟。  
  
       sub vcl\_recv {  
         set req.grace = 2m;  
       }  
       sub vcl\_fetch {  
         set beresp.grace = 2m;  
       }  
  
4、varnish对象过期之后的更新说明  
  
对象过期之后，过了grace时间，就会被后台线程清理掉。下一次用户再去访问的时候，会从后端重新抓取该对象。更合适的方法是发一个ims请求，如果后端对应的对象没有更新就不抓。这一点squid处理得比varnish好。希望以后的版本能改进这一块。