

[今日课程大纲]

分析门户中大广告需求

Redis 集群安装

完成门户大广告显示

完成后台大广告缓存数据同步

[知识点详解]

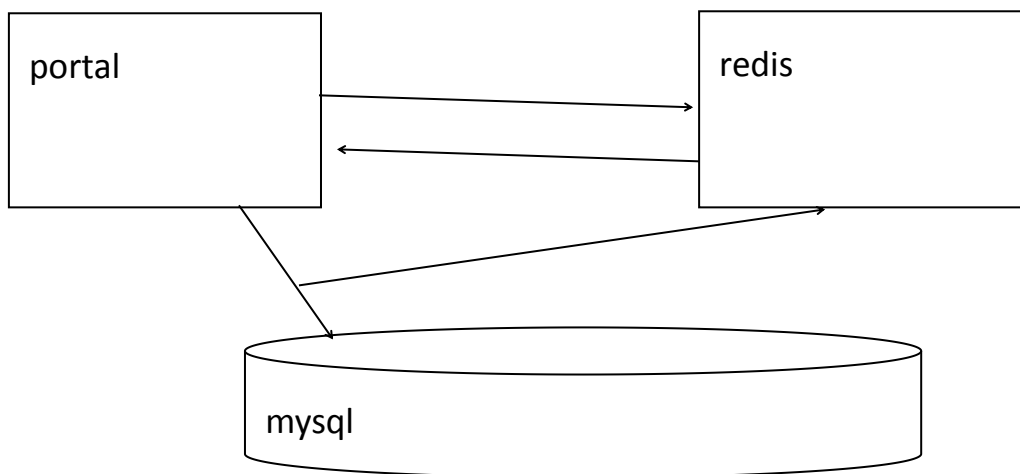
一.分析门户中大广告需求

1. 在控制器中通过 request 作用域按照固定的 json 格式把数据传递给

jsp

```
}  
//var data = [{"srcB":"http://image.ego.com/  
var data = ${ad1};
```

2. 缓存运行图



二.实现门户中大广告显示(添加缓存)

1. 在 ego-service 的 TbContentService 及实现类添加

```
/**
 * 查询出最近的前 n 个
 * @param count
 * @return
 */
List<TbContent> selByCount(int count, boolean
isSort);

@Override
public List<TbContent> selByCount(int count, boolean
isSort) {
    TbContentExample example = new TbContentExample();
    //排序
    if(isSort){
        example.setOrderByClause("updated desc");
    }
    if(count!=0){
        PageHelper.startPage(1, count);
        List<TbContent> list =
        tbContentMapper.selectByExampleWithBLOBs(example);
```

```
        PageInfo< TbContent > pi = new PageInfo<>(list);

        return pi.getList();

    }else{

        return

tbContentMapper.selectByExampleWithBLOBs(example);

    }

}
```

2. 新建 ego-redis 项目

3. 在 ego-redis 的 pom.xml 中添加

```
<dependencies>

    <!-- jedis -->

    <dependency>

        <groupId>redis.clients</groupId>

        <artifactId>jedis</artifactId>

    </dependency>

    <!-- spring -->

    <dependency>

        <groupId>org.springframework</groupId>

        <artifactId>spring-webmvc</artifactId>

    </dependency>

</dependencies>
```

4. 在 ego-redis 中添加 applicationContext-redis.xml,除了原有内容添加注解扫描

```
<context:component-scan  
base-package="com.ego.redis.dao.impl"></context:component-scan>
```

5. 在 ego-redis 中添加 redis.properties

```
redis.bigpic.key = bigpic
```

6. 在 ego-redis 中新建 com.ego.redis.dao.JedisDao 和实现类

```
public interface JedisDao {  
  
    /**  
     * 判断 key 是否存在  
     * @param key  
     * @return  
     */  
    Boolean exists(String key);  
  
    /**  
     * 删除  
     * @param key  
     * @return  
     */  
    Long del(String key);  
}
```

```
/**
 * 设置值
 * @param key
 * @param value
 * @return
 */
String set(String key,String value);

/**
 * 取值
 * @param key
 * @return
 */
String get(String key);
}

@Repository
public class JedisDaoImpl implements JedisDao{

    @Resource

    private JedisCluster jedisClients;

    @Override

    public Boolean exists(String key) {
```

```
        return jedisClients.exists(key);
    }

    @Override
    public Long del(String key) {
        return jedisClients.del(key);
    }

    @Override
    public String set(String key, String value) {
        return jedisClients.set(key, value);
    }

    @Override
    public String get(String key) {
        return jedisClients.get(key);
    }
}
```

7. 在 ego-commons 中添加工具类 JsonUtils
8. 修改 ego-portal 的 applicationContext-spring.xml

```
<context:property-placeholder
```

```
location="classpath*:*.properties"/>
```

9. 在 ego-portal 中新建 com.ego.portal.service.TbContentService 及实现类

```
public interface TbContentService {  
  
    /**  
     * 显示大广告  
     * @return  
     */  
    String showBigPic();  
}  
  
@Service  
public class TbContentServiceImpl implements  
TbContentService {  
    @Reference  
    private TbContentDubboService  
    tbContentDubboServiceImpl;  
  
    @Resource  
    private JedisDao jedisDaoImpl;  
  
    @Value("${redis.bigpic.key}")  
    private String key;  
  
    @Override  
    public String showBigPic() {
```

//先判断在 redis 中是否存在

```
if(JedisDaoImpl.exists(key)){  
    String value = JedisDaoImpl.get(key);  
    if(value!=null&&!value.equals("")){  
        return value;  
    }  
}
```

//如果不存在从 mysql 中取,取完后放入 redis 中

```
List< TbContent> list =  
tbContentDubboServiceImpl.selByCount(6, true);  
  
List<Map<String,Object>> listResult = new  
ArrayList<>();  
  
for (TbContent tc : list) {  
    Map<String,Object> map = new HashMap<>();  
  
    map.put("srcB", tc.getPic2());  
    map.put("height", 240);  
    map.put("alt", "对不起,加载图片失败");  
    map.put("width", 670);  
    map.put("src", tc.getPic());
```



```
        map.put("widthB", 550);

        map.put("href", tc.getUrl() );

        map.put("heightB", 240);

        listResult.add(map);
    }

    String listJson =
JsonUtils.objectToJson(listResult);

    //把数据放入到 redis 中
    JedisDaoImpl.set(key, listJson);

    return listJson;
}

}
```

10. 在 ego-portal 中新建 TbContentController

```
@Controller

public class TbContentController {

    @Resource

    private TbContentService tbContentServiceImpl;

    @RequestMapping("showBigPic")

    public String showBigPic(Model model){
```

```
        model.addAttribute("ad1",  
tbContentServiceImpl.showBigPic());  
  
        return "index";  
    }  
}
```

11. 在 ego-portal 中修改 PageController 如下

```
@RequestMapping("/")  
  
    public String welcome(){  
  
        return "forward:/showBigPic";  
    }  
}
```

三. 后台项目同步大广告数据

1. 在 ego-manage 的 pom.xml 依赖 ego-redis
2. 修改 ego-manage 的 applicationContext-spring.xml 属性文件扫描
3. 修改 ego-manage 的 TbContentServiceImpl

```
@Service  
  
public class TbContentServiceImpl implements  
TbContentService {  
  
    @Reference  
  
    private TbContentDubboService  
tbContentDubboServiceImpl;
```

```
@Resource

private JedisDao jedisDaoImpl;

@Value("${redis.bigpic.key}")
private String key;

@Override

public EasyUIDataGrid showContent(long categoryId,
int page, int rows) {

    return
tbContentDubboServiceImpl.selContentByPage(categoryId
, page, rows);
}

@Override

public int save(TbContent content) {

    Date date = new Date();

    content.setCreated(date);

    content.setUpdated(date);

    int index =
tbContentDubboServiceImpl.insContent(content);

    //判断 redis 中是否有缓存数据
```

```
if(jedisDaoImpl.exists(key)){  
    String value = jedisDaoImpl.get(key);  
    if(value!=null&&!value.equals("")){  
        List<HashMap> list =  
JsonUtils.jsonToList(value, HashMap.class);  
  
        HashMap<String, Object> map = new HashMap<>();  
  
        map.put("srcB", content.getPic2());  
        map.put("height", 240);  
        map.put("alt", "对不起,加载图片失败");  
        map.put("width", 670);  
        map.put("src", content.getPic());  
        map.put("widthB", 550);  
        map.put("href", content.getUrl() );  
        map.put("heightB", 240);  
  
        //保证六个  
        if(list.size()==6){  
            list.remove(5);  
        }  
        list.add(0, map);  
    }  
}
```

```
        jedisDaoImpl.set(key,  
        JsonUtils.objectToJson(list));  
    }  
}  
  
    return index;  
}  
}
```