# 预抓取功能详细设计文档

## 概述

预抓取工作主要是为了华为天天浏览器agent服务，其主要功能是通过抓取并缓存用户访问量较大的url，以提高缓存命中率，更好的提高agent的性能。

## 业务简述

当前agent的缓存命中率为35%-40%，命中率较低，如果能提高缓存命中率，能降低服务器的压力，间接的提高服务器处理的性能，从而使用户体验更好。

## 目标描述

预置条件：url列表。

读取url列表，获取该列表下的url的内容，存入缓存，需要实现：

1）获取那些标签可以控制；

2）获取层级可以控制；

3）图片可以根据多次处理，以适配不同手机型号；

4）每一轮的抓取时间为180分钟，短于缓存失效时间；

5）多线程多台服务器同时抓取。

最终结果：缓存数据中有相应的内容，提高缓存命中率。

## 系统总体描述

## 系统上下文

* url列表

url列表由数据统计模块负责统计，在数据库中对应于表：S\_PV\_HOST

* Internet

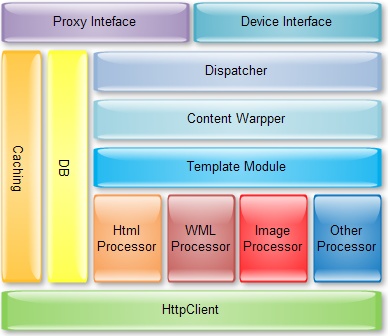
预抓取需要实现HTTP 和 HTTPS协议，以保证获取Internet资源。

## 系统功能模块列表

* 初始化模块
* 初始化线程池
* 根据访问次数读取所需抓取的url列表
* 初始化Berkeley DataBase数据库
* Berkeley DataBase URL队列的构建
* 模拟客户端访问的json数据
* 关闭PreCache的识别
* 过滤收集抓取页面的URL链接，并进行补全，将符合条件的url加入到url队列中
* 对图片实现多次处理，以匹配不同分辨率的手机和图片质量

## 功能模块详细设计

## 系统架构



简单描述：预抓取系统以当前agent为基础，在其基础上稍加改造，模拟了json数据的生成过程，所以其基本架构与agent相同，参考agent的系统架构。

Proxy Interface

浏览器引擎通过proxy interface与网络代理服务器联系，代理服务器通过此接口获取信息。浏览器欲访问资源，浏览器引擎携带的请求参数，设备信息等。

Device Interface

第三方应用通过此接口，提交设备信息。如华为产品有新增的设备，需要提交该设备相关信息至服务器，则调用此接口。

Caching

Caching模块用以缓存已处理资源和已处理资源索引，提高系统反应速度。且caching模块具有可配置型，满足缓存刷新的需要。另外caching模块也用以存储运行时状态，提高系统运行速度。

Dispatcher

分发模块主要用以适配不同的引擎版本和设备型号。

Content wrapper

此模块主要封装已处理资源，按照接口调用方的需要，进行封装。如封装块页面之间的跳转超链接等。

Processor

按照不同规范进行内容处理

HTTP Client

获取Internet资源的唯一途径，需要支持HTTP协议和HTTPS协议。

## 详细设计

[1] 通过嵌入式servlet容器，启动servlet服务器，初始化加载相关文件

[2] 初始化线程池，用来执行任务

[3] 读取URL，初始化要抓取的url

[4] 模拟json数据,相当于模拟客户端的请求

[5] 这里采用了责任链模式，将每个处理流程当成一个filter，加入到链中

[6] 将当前json数据传输给第一个filter，它将json字符串解析成pojo

[7] httpclient通过json数据，补齐相应的请求头，并且发起请求，得到响应的http头协议和html体

[8] imageHandle 根据从httpclient得到的信息判断是否为图片，如果是，则进行压缩

[9] 如果类型是图片，保存到cache中，并且输出

[10] 如果不是图片，将进入 html doc handle，进行xhtml标签补齐和去掉不支持标签以及过滤js、css操作，并且过滤当前文件中的url链接，将符合抓取条件的url加入到url队列中去

[11] 进行分块和权重排序，以及其他操作

[12] 最后将处理结果存入缓存，如果是图片的话，需要多次压缩处理。

## 初始化模块

[1] 概述

该模块主要负责加载agent工程所需要的一些信息，如配置文件的读取、预加载css和js文件、预加载广告模板、预加载风格模板。

[2] 详细设计

对于该模块，详见start.java文件内的load()方法。

## 初始化线程池

[1] 概述

预抓取工程需要以多线程实现多任务处理，该模块主要负责初始化线程池，线程池中包括任务队列，线程池大小等。

[2] 详细设计

对于线程池，在此采用jdk中自带的ThreadPoolExecutor类作为本模块的线程池，首先需要读取成功加载的minthreadpoolsize、maxthreadpoolsize、queuesize参数

ThreadPoolExecutor tp = new ThreadPoolExecutor(

minthreadpoolsize, maxthreadpoolsize, 1000 \* 60 \* 60, TimeUnit.MILLISECONDS,

new ArrayBlockingQueue<Runnable>(queuesize) );

其中minthreadpoolsize为最小线程数量，maxthreadpoolsize为最大线程数量，ArrayBlockingQueue为任务队列，queuesize为任务队列的大小。

## 初始化Berkeley DataBase数据库

[1] 概述

Berkeley DB Java Edition (JE)是一个完全用JAVA写的，它适合于管理海量的，简单的数据。能够高效率的处理1到1百万条记录，制约JE数据库的往往是硬件系统,而不是JE本身。多线程支持，JE使用超时的方式来处理线程间的死琐问题。Database都采用简单的key/value对应的形式。

JE要求在任何DATABASE操作前，要先打开数据库环境，就像我们要使用数据库的话必须得先建立连接一样。你可以通过数据库环境来创建和打开database，或者更改database名称和删除database。

[2] 详细设计

初始化BDB目录和 BDB环境，实例化BerkeleyDBUtil类。

## 根据访问次数读取所需抓取的url列表

该模块即是读取SPV表的模块，在读取SPV表的过程中实现多台预抓取服务器之间的协调，并且可以根据以下两个参数（预抓取服务器的数量serverNum，每台机器两个小时内所能抓取的URL的数量urlNum）调整最终返回的结果数量。

## Berkeley DataBase URL队列的构建

在完成BDB初始化后，新建URL队列，已实现Berkeley DataBase URL队列的 构建，代码详见UrlQueue类的实现。

## 关闭PreCache识别

新建一个Filter的子类，实现doProcess方法，抓取第一层方法体直接返回AttributeChain.HTTPCLIENT\_FILTER即可。然后在spring的配置文件beans.xml中，preCacheFilter改为该子类。

## 过滤收集抓取页面的URL链接

该模块实现通过重写原有的htmlElementFilter实现，过滤收集页面中的链接，将收集到的链接保存至BDB数据库中，等待抓取。

## 对图片实现多次压缩处理，以匹配不同分辨率的手机和图片质量

在EndCacheFilter中，对于是图片的链接，则进行多次处理，详见MultiEndCacheFilter。