# 百度日志分析技术分享

陈晓鸣
资深工程师
百度基础架构部
@陈晓鸣在百度
chenxiaoming@baidu.com

# 互联网公司技术架构系列资料

由  为您悉心整理

快简历

KuaiJianLi.com

/* 让工作重新关于成长和成就、关于快乐和分享、关于梦想和荣光 */

- LOG中自有黄金屋
- 日志分析基本过程
- 百度日志分析成长历程
- 深入LSP平台
- 深入DISQL语言
- 总结与问答

# 一条日志

- 46.70.93.94 - - [11/Nov/2011:11:11:11 -1100] "GET /book/1984.html HTTP/1.1 “404 2326 http://www.baidu.com/s?wd=1984&rsv_bp=0&rsv_spt=3&inputT=947 "Mozilla/5.0(iPad; U; CPU iPhone OS 3_2 like Mac OS X; en-us) AppleWebKit/531.21.10 (KHTML, like Gecko) Version/4.0.4 Mobile/7B314 Safari/531.21.10 ”

# 了解用户——地域来源

- 46.70.93.94 - -
- [11/Nov/2011:11:11:11 -1100]
- "GET /book/1984.html HTTP/1.1"
- 404
- 2326
- "http://www.baidu.com/s?wd=1984&rsv_bp=0&rsv_spt=3&inputT=947"
- "Mozilla/5.0(iPad; U; CPU iPhone OS 3_2 like Mac OS X; en-us) AppleWebKit/531.21.10 (KHTML, like Gecko) Version/4.0.4 Mobile/7B314 Safari/531.21.10 "

# 了解用户——跳转来源

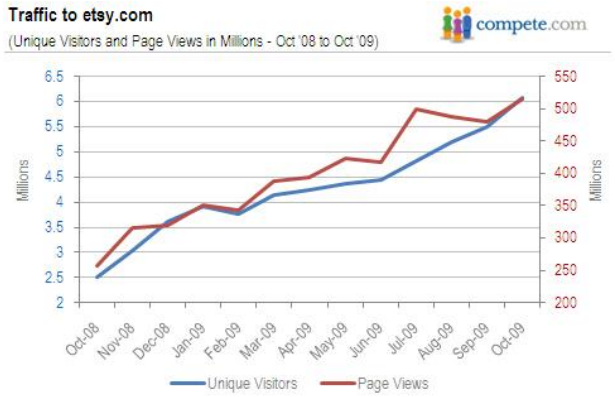
- 46.70.93.94 - -
- [11/Nov/2011:11:11:11 -1100]
- "GET /book/1984.html HTTP/1.1"
- 404
- 2326
- "http://www.baidu.com/s?wd=1984&rsv_bp=0&rsv_spt=3&inputT=947"
- "Mozilla/5.0(iPad; U; CPU iPhone OS 3_2 like Mac OS X; en-us) AppleWebKit/531.21.10 (KHTML, like Gecko) Version/4.0.4 Mobile/7B314 Safari/531.21.10 "

# 了解用户——使用终端

- 46.70.93.94 - -
- [11/Nov/2011:11:11:11 -1100]
- "GET /book/1984.html HTTP/1.1"
- 404
- 2326
- "http://www.baidu.com/s?wd=1984&rsv_bp=0&rsv_spt=3&inputT=947"
- " Mozilla/5.0(iPad; U; CPU iPhone OS 3_2 like Mac OS X; en-us) AppleWebKit/531.21.10 (KHTML, like Gecko) Version/4.0.4 Mobile/7B314 Safari/531.21.10"

# 了解自己——访问量

- 46.70.93.94 - -
- [11/Nov/2011:11:11:11 -1100]
- GET /book/1984.html HTTP/1.1
- 404
- 2326
- "http://www.baidu.com/s?wd=1984&rsv_bp=0&rsv_spt=3&inputT=947 "
- "Mozilla/5.0(iPad; U; CPU iPhone OS 3_2 like Mac OS X; en-us) AppleWebKit/531.21.10 (KHTML, like Gecko) Version/4.0.4 Mobile/7B314 Safari/531.21.10 "

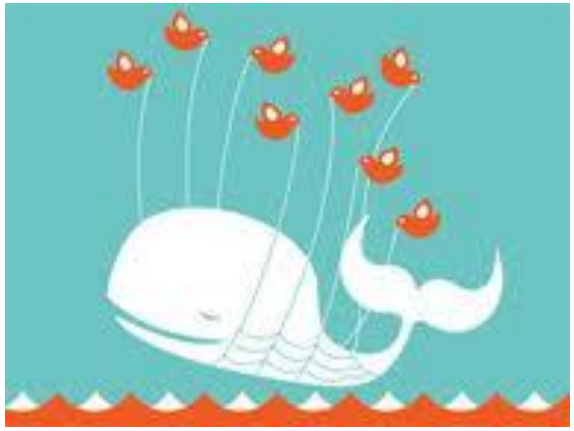



# 了解自己——易用性

- 46.70.93.94 - -
- [11/Nov/2011:11:11:11 -1100]
- "GET /book/1984.html HTTP/1.1"
- 404
- 2326
- "http://www.baidu.com/s?wd=1984&rsv_bp=0&rsv_spt=3&inputT=947"
- " Mozilla/5.0(iPad; U; CPU iPhone OS 3_2 like Mac OS X; en-us) AppleWebKit/531.21.10 (KHTML, like Gecko) Version/4.0.4 Mobile/7B314 Safari/531.21.10 "

# 了解自己——可用性

- 46.70.93.94 - -
- [11/Nov/2011:11:11:11 -1100]
- "GET /book/1984.html HTTP/1.1"
- 404
- 2326
- " http://www.baidu.com/s?wd=1984&rsv_bp=0&rsv_spt=3&inputT=947 "
- "Mozilla/5.0(iPad; U; CPU iPhone OS 3_2 like Mac OS X; en-us) AppleWebKit/531.21.10 (KHTML, like Gecko) Version/4.0.4 Mobile/7B314 Safari/531.21.10 "



# 目标——更好地服务

# 目标——更好地赚钱

# 日志分析基本过程

- LOG中自有黄金屋
- 日志分析基本过程
- 百度日志分析成长历程
- 深入LSP平台
- 深入DISQL语言
- 总结与问答

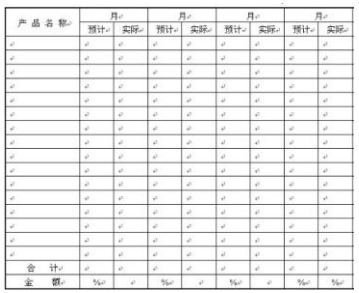
# 日志分析基本过程

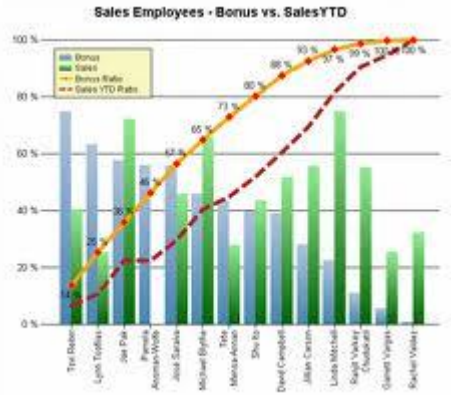
## 线上
- 提取
- 传输

## 预处理
- 解析
- 过滤

## 计算
- 预定义计算
- Ad hoc 查询

## 使用
- 报表
- 图表
- 回馈线上
- ……






# 百度日志分析成长历程

- LOG中自有黄金屋
- 日志分析基本过程
- 百度日志分析成长历程
- 深入LSP平台
- 深入DISQL语言
- 总结与问答

# 使用人数

LSP

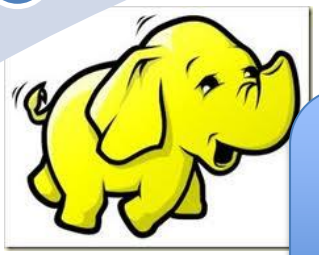
- 正在进行中...

- 一站式的日志分析服务
- B-S模式访问
- 私有云
- 统一管理、统一运维

# 数据量的问题

DISQL

DISQL 2.0



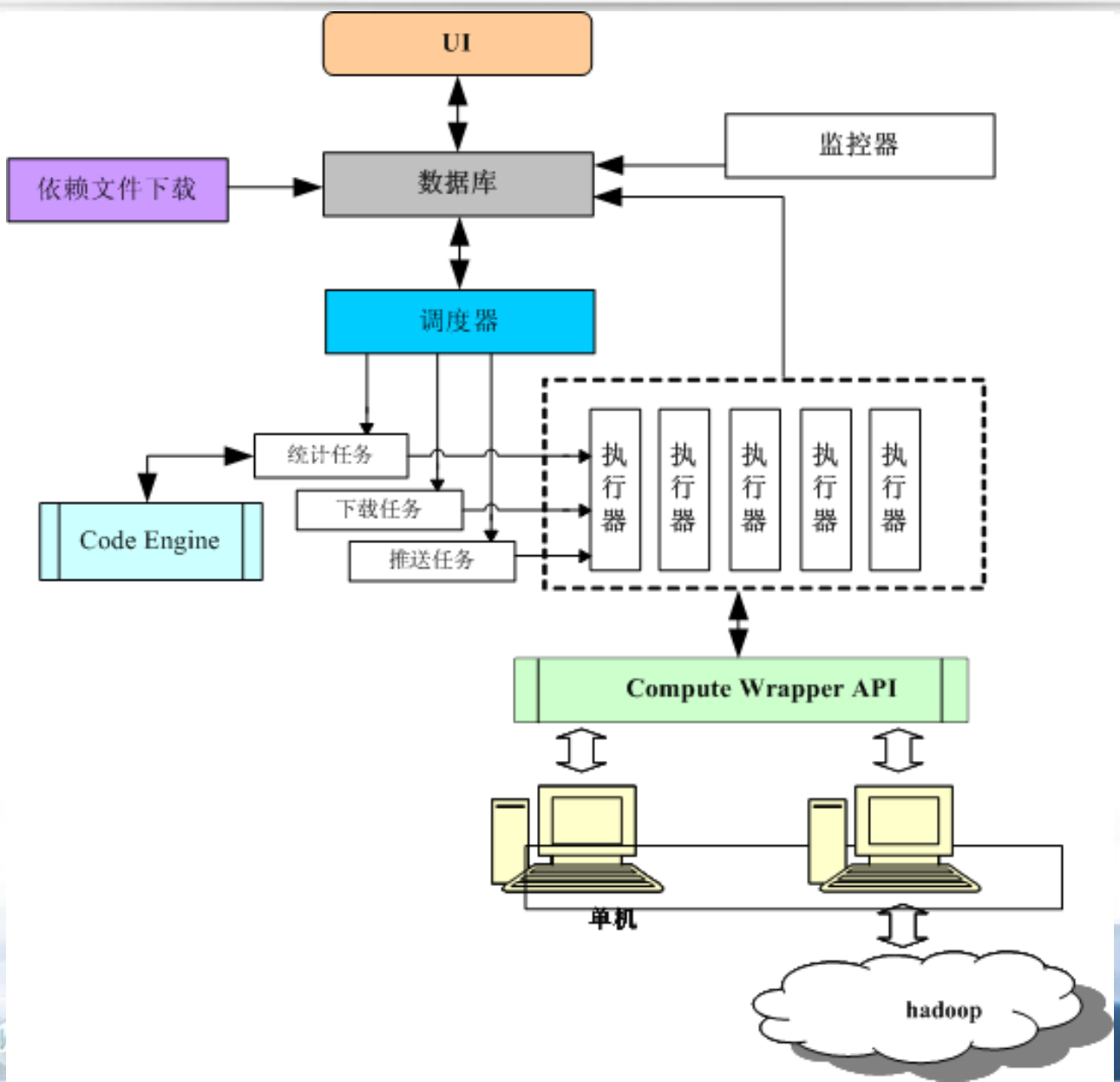
- 类SQL的描述式语言
- PHP + C 混合运行时
- 算子的分布式实现
- 用户自定义函数支持（PHP、.so）

- 全新的C++ 运行时
- C++自定义函数支持
- 更多算子支持
- Schema推导支持
- 大量的优化

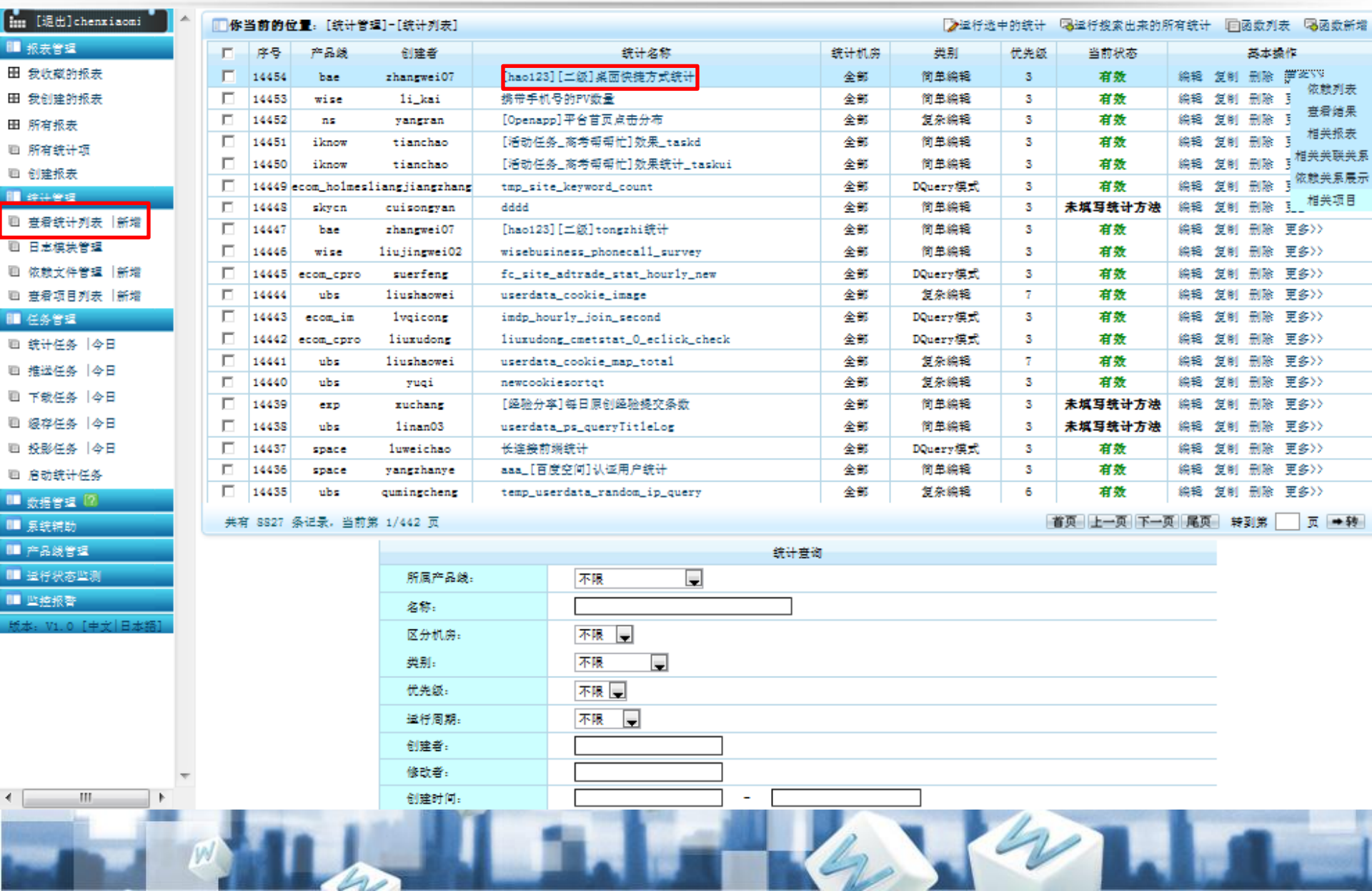
cron awk php

- LOG中自有黄金屋
- 日志分析基本过程
- 百度日志分析成长历程
- 深入LSP平台
- 深入DISQL语言
- 总结与问答

# 系统架构


UI
监控器
依赖文件下载
数据库
调度器
统计任务
下载任务
推送任务
Code Engine
执行器
执行器
执行器
执行器
执行器
Compute Wrapper API
单机
hadoop

# 平台UI


[退出]chenxiaomi
报表管理
我收藏的报表
我创建的报表
所有报表
所有统计项
创建报表
统计管理
查看统计列表 |新增
日志模块管理
依赖文件管理 |新增
查看项目列表 |新增
任务管理
统计任务 |今日
推送任务 |今日
下载任务 |今日
缓存任务 |今日
投影任务 |今日
启动统计任务
数据管理
系统辅助
产品线管理
运行状态监测
监控报警
版本：V1.0 [中文|日本語]
你当前的位置：[统计管理]-[统计列表]
运行选中的统计
运行搜索出来的所有统计
函数列表
函数新增
序号 产品线 创建者 统计名称 统计机房 类别 优先级 当前状态 基本操作
14454 bae zhangwei07 [hao123][二级]桌面快捷方式统计 全部 简单编辑 3 有效 编辑 复制 删除
14453 wise li_kai 携带手机号的PV数量 全部 简单编辑 3 有效 编辑 复制 删除
14452 ns yangran [Openapp]平台首页点击分布 全部 复杂编辑 3 有效 编辑 复制 删除
14451 iknow tianchao [活动任务_高考哥哥忙]效果_taskd 全部 简单编辑 3 有效 编辑 复制 删除
14450 iknow tianchao [活动任务_高考哥哥忙]效果统计_taskui 全部 简单编辑 3 有效 编辑 复制 删除
14449 ecom_holmes liangjiangzhang tmp_site_keyword_count 全部 DQuery模式 3 有效 编辑 复制 删除
14448 skycn cuisongyan dddd 全部 简单编辑 3 未填写统计方法 编辑 复制 删除
14447 bae zhangwei07 [hao123][二级]tongzhi统计 全部 简单编辑 3 有效 编辑 复制 删除 更多>>
14446 wise liujingwei02 wisebusiness_phonecall_survey 全部 简单编辑 3 有效 编辑 复制 删除 更多>>
14445 ecom_cpro suerfeng fc_site_adtrade_stat_hourly_new 全部 DQuery模式 3 有效 编辑 复制 删除 更多>>
14444 ubs liushaowei userdata_cookie_image 全部 复杂编辑 7 有效 编辑 复制 删除 更多>>
14443 ecom_im lvqicong imdp_hourly_join_second 全部 DQuery模式 3 有效 编辑 复制 删除 更多>>
14442 ecom_cpro liuxudong liuxudong_cmetstat_0_eclick_check 全部 DQuery模式 3 有效 编辑 复制 删除 更多>>
14441 ubs liushaowei userdata_cookie_map_total 全部 复杂编辑 7 有效 编辑 复制 删除 更多>>
14440 ubs yuqi newcookiesortqt 全部 复杂编辑 3 有效 编辑 复制 删除 更多>>
14439 exp xuchang [经验分享]每日原创经验提交条数 全部 简单编辑 3 未填写统计方法 编辑 复制 删除 更多>>
14438 ubs linan03 userdata_ps_queryTitleLog 全部 简单编辑 3 未填写统计方法 编辑 复制 删除 更多>>
14437 space luweichao 长连接前端统计 全部 DQuery模式 3 有效 编辑 复制 删除 更多>>
14436 space yangzhanye aaa_[百度空间]认证用户统计 全部 简单编辑 3 有效 编辑 复制 删除 更多>>
14435 ubs qumingcheng temp_userdata_random_ip_query 全部 复杂编辑 6 有效 编辑 复制 删除 更多>>
依赖列表
查看结果
相关报表
相关关联关系
依赖关系展示
相关项目
共有 8827 条记录，当前第 1/442 页
首页 上一页 下一页 尾页 转到第 页 转
统计查询
所属产品线：不限
名称：
区分机房：不限
类别：不限
优先级：不限
运行周期：不限
创建者：
修改者：
创建时间：

# 编辑模式

# 简单模式


[退出]chenxiaoming
报表管理
我收藏的报表
我创建的报表
所有报表
所有统计项
创建报表
统计管理
查看统计列表 |新增
日志模块管理 |新增
依赖文件管理 |新增
查询下载记录
任务管理
统计任务 |今日
下载任务 |今日
缓存任务 |今日
推送任务 |今日
启动统计任务
调度器管理
数据管理
数据推送 |新增
推送记录查询
统计项保存时间设置
关联关系列表 |新增
运行状态监测
统计任务监控
报表监控
您当前的位置：ecom_cpro - test
切换编辑模式
添加数据源 显示高级选项
1. 日志: ecom_pb_log 删除
统计编辑区
计数 Top统计 去重计数 数据过滤 更多 全部收起 帮助
1. 分地域计数
*分城市计数变里名 _CityAdCnt *中文名 分城市广告展现
*分省份计数变里名 _ProvAdCnt *中文名 分省份广告展现
*IP字段: 请选择...
添加限制条件 使用公共限制条件组
1. _Url 正则匹配
2. _Time 数字大于
为该统计添加注释(选填):
加和
去重列表
分时段计数
分时段加和
分时段去重
分组计数
分地域计数
分地域去重
新增记录统计
重复记录统计
Transmit统计
公共限制条件组
添加公共限制条件组 全部收起
1. 删除
添加限制条件
检查 测试 提交
日志行实例 上传文件测试
我要测试
测试返回信息 取消滚动
:) 您还没进行测试!

# DQuery模式

您当前的位置：cpro - doris_log_dcharge_sf生成字典　　切换编辑模式

统计数据源列表　　添加数据源　显示高级选项

类型：日志　数据源：doris_log_dcharge_s　日期：当日　删除

字段列表

原始记录集的所有字段列表：_Balance　_Balance

代码编辑区　　API手册　帮助　查看流程图

```
1  DQuery::input()
2  ->filter( array( array( '_Cmatch', '===', 202) ) )
3  ->select( array( '_Srchid', '_Descid', '_Bid' ) )
4  ->group( array('_Srchid', '_Descid' ) )
5  ->each(
6      DQuery::count('*', '_Click' ),
7      DQuery::sum('_Bid', '_Gain' )
8  )
9  ->outputAsFile('doris_log_dict','doris日志生成字典',
   'StorerUtils::phpLine',true);
10
```

检查　测试　提交

日志行实例　上传文件测试

```
101      1947467 2055624
11184222
269167736        周星驰
206344   28176565
0.68     0.54
32918.62
```

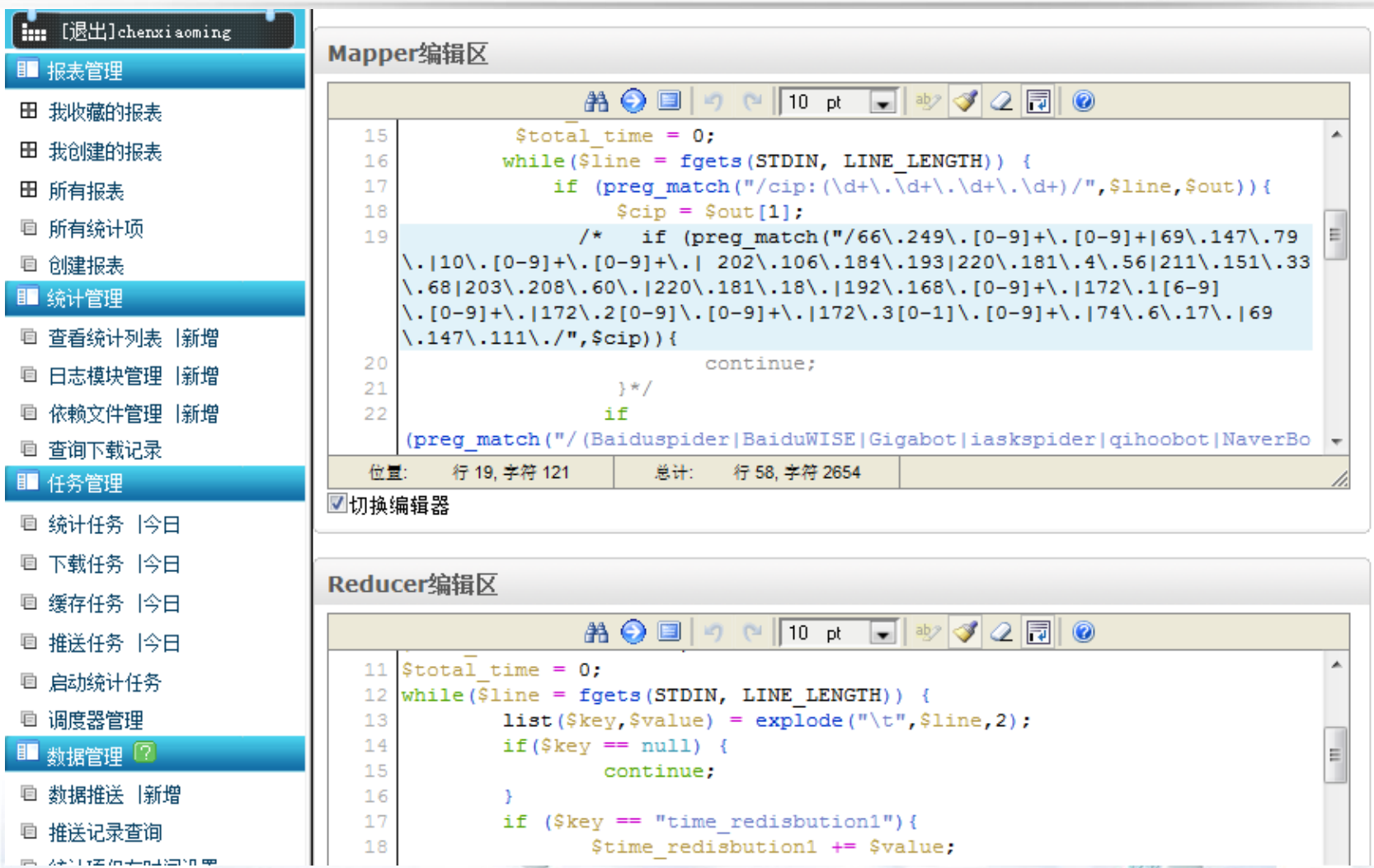
我要测试

测试返回信息

**doris_log_dict:**

a:5:{s:7:"_Srchid";s:16:"5bef9a5e

# 复杂模式


[退出]chenxiaoming
报表管理
我收藏的报表
我创建的报表
所有报表
所有统计项
创建报表
统计管理
查看统计列表 |新增
日志模块管理 |新增
依赖文件管理 |新增
查询下载记录
任务管理
统计任务 |今日
下载任务 |今日
缓存任务 |今日
推送任务 |今日
启动统计任务
调度器管理
数据管理
数据推送 |新增
推送记录查询
Mapper编辑区
10 pt
$total_time = 0;
while($line = fgets(STDIN, LINE_LENGTH)) {
if (preg_match("/cip:(\d+\.\d+\.\d+\.\d+)/",$line,$out)){
$cip = $out[1];
/* if (preg_match("/66\.249\.[0-9]+\.[0-9]+|69\.147\.79
\.|10\.[0-9]+\.[0-9]+\.| 202\.106\.184\.193|220\.181\.4\.56|211\.151\.33
\.68|203\.208\.60\.|220\.181\.18\.|192\.168\.[0-9]+\.|172\.1[6-9]
\.[0-9]+\.|172\.2[0-9]\.[0-9]+\.|172\.3[0-1]\.[0-9]+\.|74\.6\.17\.|69
\.147\.111\./",$cip)){
continue;
}*/
if
(preg_match("/(Baiduspider|BaiduWISE|Gigabot|iaskspider|qihoobot|NaverBo
位置: 行 19, 字符 121
总计: 行 58, 字符 2654
切换编辑器
Reducer编辑区
10 pt
$total_time = 0;
while($line = fgets(STDIN, LINE_LENGTH)) {
list($key,$value) = explode("\t",$line,2);
if($key == null) {
continue;
}
if ($key == "time_redisbution1"){
$time_redisbution1 += $value;

- LOG中自有黄金屋
- 日志分析基本过程
- 百度日志分析成长历程
- 深入LSP平台
- 深入DISQL语言
- 总结与问答

# 例：新闻站点访问量和广告量统计

- 执行步骤
  - 读取日志数据
  - 选取出_Url、_Res(广告数)两列
  - 编写一个函数，从_Url中抽取出_Site
  - 用正则表达式过滤出新闻站点的数据
  - 按站点分组，每组做两件事：
    - 计算访问量
    - 将广告数求和
  - 输出数据，每行是一个JSON数据

# 例：DQuery代码

```
function get_site($rec){
    $parts = explode( '/', $rec['_Url'] );
    $rec['_Site'] = $parts[0];
    return $rec;
}

$temp_view = DQuery::input()
->select( array( '_Url','_Res' ) )
->select( 'get_site' )
->filter( array( '_Site', 'match', '/news\.[^.]+\.cn/' ) );

$temp_view
->group( array('_Site') )
->each(
    DQuery::count( '*', '_QueryCnt' ),
    DQuery::sum( '_Res', '_ResSum' )
)->outputAsFile( 'query_ad_shows', '分站点访问量与广告量统计
','StorerUtils::jsonLine' );
```

- 读取日志数据
- 选取出_Url、_Res(广告数)两列
- 编写一个函数，从_Url中抽取出_Si
- 用正则表达式过滤出新闻站点的数
- 按站点分组，每组做两件事：
  - 计算访问量
  - 将广告数求和
- 输出数据，每行是一个JSON数据

# PHP-Callback

```php
function expand($fields){
    $ret=array();
    for($i=0;$i<count($fields['_Dis'][0]);$i++)    {
        if($fields['_Dis'][0][$i]>0 && $fields['_Cn'][0][$i]!='baidu_fc_gusuan'
   &&$fields['_Cn'][0][$i]!='baidufcidear_pg' && $fields['_Cn'][0][$i]!='baiduadrquery_pg')
{
            for($j=0;$j<$fields['_Dis'][0][$i];$j++){
                $r=array();
                $r['_Srchid']=$fields['_S'];
                $r['_Cmatch']=$fields['_Im_apres'][27][$i][$j];
                $r['_Rank']=(string)$fields['_Absrk'][0][$i][$j];
                $r['query']=$fields['_Query'];
                $r['ip']=$fields['_Ip'];
                $r['cn']=$fields['_Cn'][0][$i];
                $r['bd']=$fields['_Bd'];
                $r['dis']=$fields['_Dis'][0][$i];
                $r['wd']=$fields['_Wd'][0][$i][$j];
                $r['pid']=$fields['_Pid'];
                $r['cid']=$fields['_Cid'];
                $r['tim']=$fields['_Tim'];
                $r['term']=$fields['_Term'][0][$i][$j];
                $r['pres_1']=$fields['_Pres_1'];
                $r['pn']=$fields['_Pn'];
                $r['eq']=$fields['_Eq'];
                $r['qs']=$fields['_Extra']['qs'];
                $r['const1']=1;
                global $valfields;
                foreach($valfields as $val)
                {
                    if($r[$val]==''||$r[$val]==null)
                        $r[$val]='-';
```

位置: 行 1, 字符 26 | 总计: 行 41, 字符 1521

# C-callback

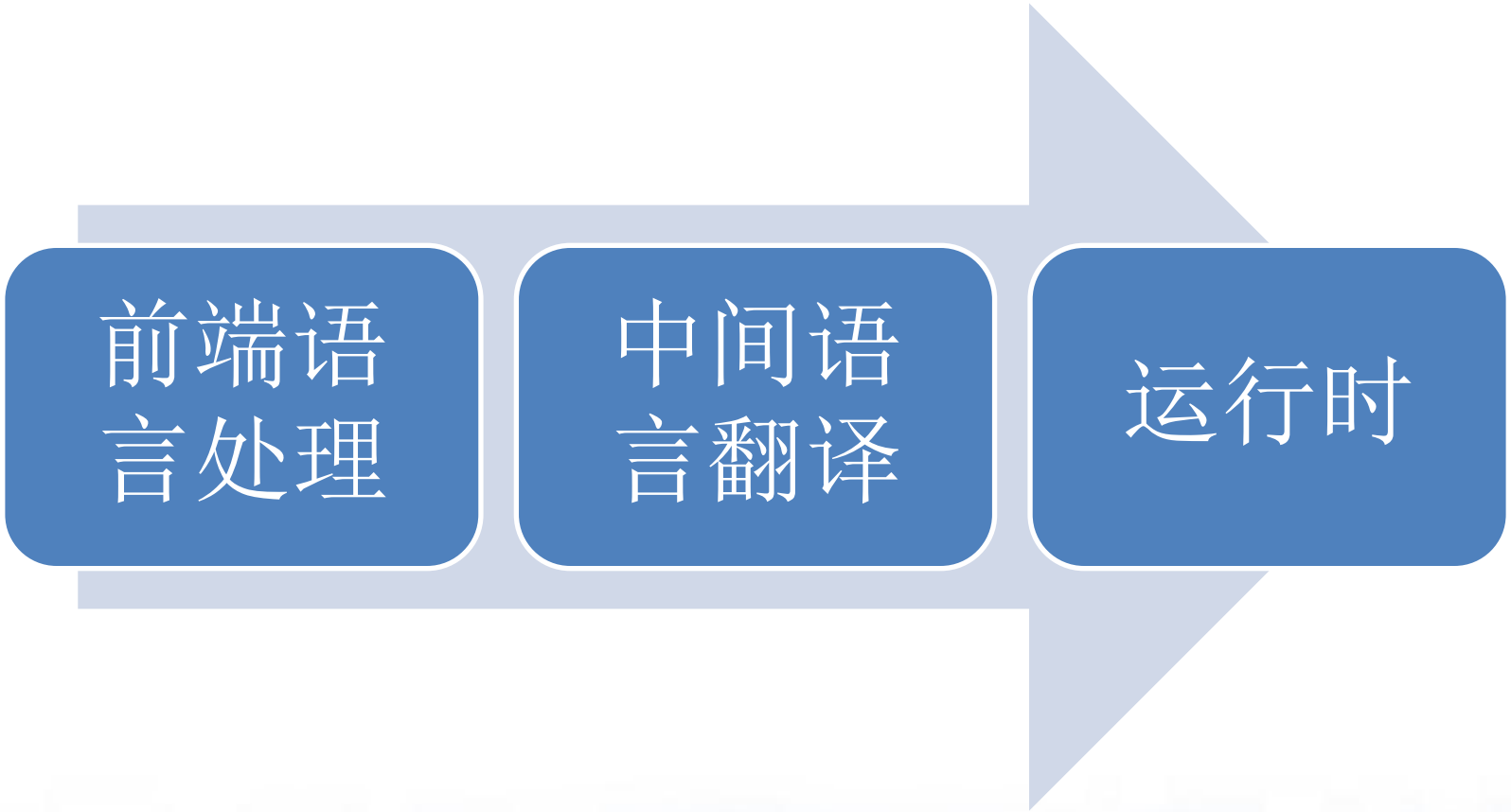
```
#include "disql_udf.h"
#include "disql_udf_schema.h"

extern "C" void expand(
    bsl::var::LValue& fields, bsl::var::LValue& result, disql::CallbackContext& context) {
    int k = 0;
    if (!fields[ASP_LOG::_Dis].is_array()) return;
    if (!fields[ASP_LOG::_Dis][0].is_array()) return;
    int dis0_size = fields[ASP_LOG::_Dis][0].size();
    for (int i = 0; i < dis0_size; i++) {
        int dis0i;
        const char * cn;
        if (fields[ASP_LOG::_Dis][0][i].is_null() || strcmp(fields[ASP_LOG::_Dis][0]
[i].c_str(), "") == 0) dis0i = 0;
        else dis0i = fields[ASP_LOG::_Dis][0][i].to_int32();
        if (fields[ASP_LOG::_Cn][0][i].is_null()) cn = "";
        else cn = fields[ASP_LOG::_Cn][0][i].c_str();
        if (dis0i > 0 && strcmp("baidu_fc_gusuan", cn) != 0 && strcmp("baidufcidear_pg", cn)
!= 0 && strcmp("baiduadrquery_pg", cn) != 0){
            for (int j = 0; j < dis0i; j++) {
                result[k][EXPAND::_Srchid]      = fields[ASP_LOG::_S];
                result[k][EXPAND::_Cmatch]      = fields[ASP_LOG::_Im_apres][27][i][j];
                result[k][EXPAND::_Rank]        = fields[ASP_LOG::_Absrk][0][i][j];
                result[k][EXPAND::query]        = fields[ASP_LOG::_Query];
                result[k][EXPAND::ip]           = fields[ASP_LOG::_Ip];
                result[k][EXPAND::cn]           = fields[ASP_LOG::_Cn][0][i];
                result[k][EXPAND::bd]           = fields[ASP_LOG::_Bd];
                result[k][EXPAND::dis]          = fields[ASP_LOG::_Dis][0][i];
                result[k][EXPAND::wd]           = fields[ASP_LOG::_Wd][0][i][j];
                result[k][EXPAND::pid]          = fields[ASP_LOG::_Pid];
                result[k][EXPAND::cid]          = fields[ASP_LOG::_Cid];
```

位置: 行 1, 字符 23 总计: 行 50, 字符 2520

# 几个特点

- 用PHP表达的类SQL逻辑(非常简约)
  - 封装所有SQL算子的M/R分布式实现：
    - 分组、表连接、行列过滤、集合操作、输入输出格式转换
- 通过连续函数调用表达DAG数据流
  - 自动翻译为一轮或多轮MapReduce
  - 也可翻译为单机计算或数据流图
- 用逻辑顺序而非SQL顺序表达逻辑
- 支持PHP自定义函数（简洁）
- 支持C++自定义函数（同样简洁+高效）和C-Runtime NEW!
  - 全自动高效内存管理 （RAII + 内存池）
  - 廉价对象复制（Copy On Write）
  - 字段操作翻译为数组操作，无字典查找 （schema推导）
  - C++的性能，PHP的开发代价！

# 处理阶段



# 前端处理

- 把用户编写的计算逻辑翻译为便于编译程序理解的中间码（语法树、数据流图）
- 前端代码运行一遍，产生结果是中间码
- 相当于编译技术中的parser
- 中间码用JSON表示

```
[
  {
    "cmd": "load",
    "path": null
    "using": "SchemaReader"
    "from": 17
    "options": {"max_item_in_mem": 100000}
    "include": [25]
  }
   , {"cmd":"filter"……}, {"cmd":"join"……},…… ……
]
```

# SQL语言接口？

```
query: SELECT fiel_list FROM form_list WHERE
clause
fiel_list: fiel_list ',' fiel_name | fiel_name |
'*';
fiel_name: ID|FID;
form_list: form_list ',' form_name|form_name;
form_name: ID;
clause: clause AND clause | clause OR clause |
clause_express| '('clause')';
clause_express: element compopera element;
element: element mathopera element | element1 |
'('element')';
compopera: '<' | '>' | '=' | '>=' | '<
=';
mathopera: '+' | '-' | '*' | '/';
element1: ID|ICON|FID|'"'STRI '"';
```

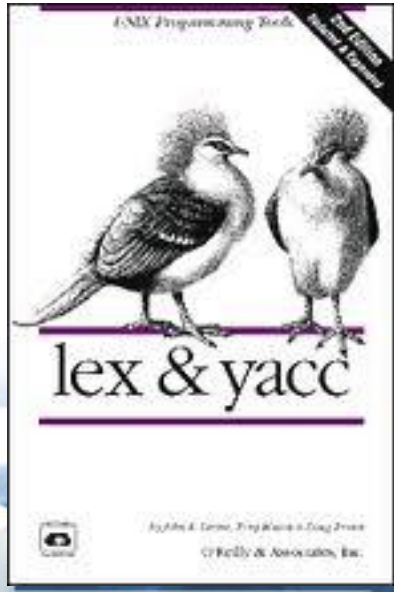
**语言定义（词法、语法分析）**

**动作（生成中间码）**

```
%%
SELECT |yylval.str=yytext;return(SELECT);|
";" |yylval.str = yytext;return(ENDMARK);|
/*;为结束符*/
.....
|alpha||alnum| * |yylval.str = yytext; return
(ID);| /*变量名或字段各*/
|digit| + |yylval.str = yytext; return(I-
CON);| /*常数*/
|alpha||alnum|[.]|alpha||alnum| * |yyl-
val.str=yytext;return(FID);|
/*带表或视图名的字段名*/
|alnum| + |yylval.str = yytext;return(STRI);|
/*字符串*/
[/t/n]; /*去除空格、制表符和换行*/
%%
```

```
[
  {
    "cmd": "load",
    "path": null
    "using": "SchemaReader"
    "from": 17
    "options":
    {"max_item_in_mem":
    100000}
    "include": [25]
  }
  , {"cmd":"filter"……},
  {"cmd":"join"……},…… ……
]
```

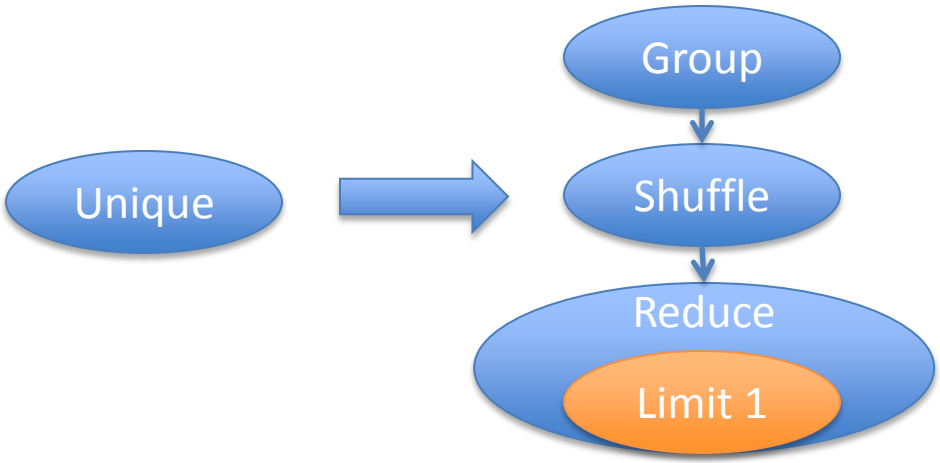
**中间码**

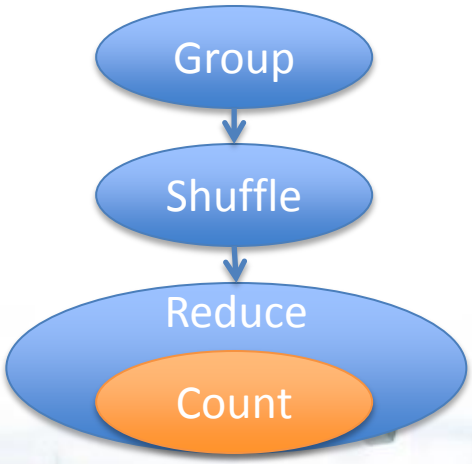


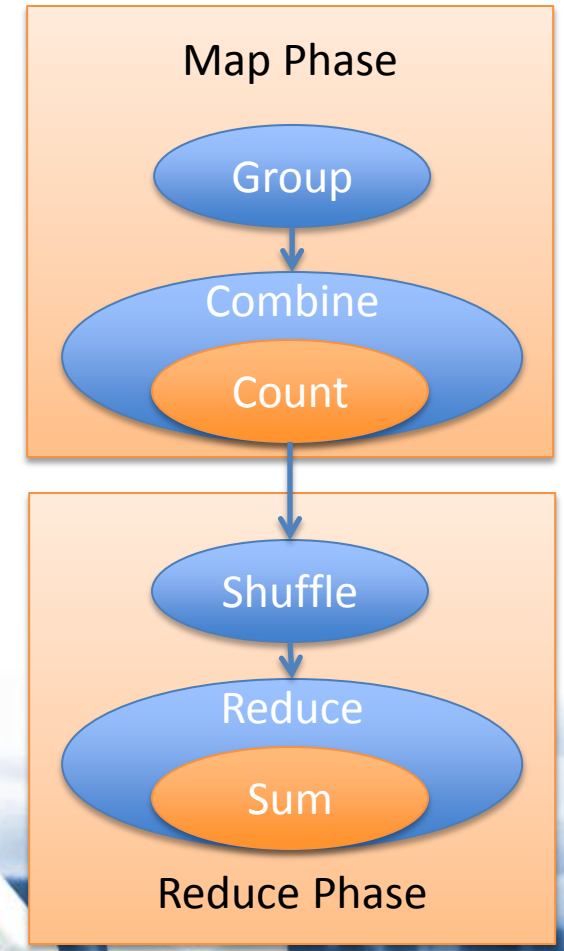
# 中间语言处理

- 对数据流图作多次等价变换
  - 正规化
    - 将数据流图变成完整的方便后续处理的数据流图
  - 算子替换
    - 将实现复杂的算子等价替换成多个简单算子
  - 优化
    - 对数据流图进行各种优化，使执行效率提高
  - 阶段划分(可选）
    - 划分为多个MapReduce执行阶段
  - Schema推导、字段偏移量推导
    - 推导每一算子产出的表schema，以及字段偏移量
  - 代码生成（C++、PHP、DOT）
    - 生成真正可执行的代码

# 算子替换


Unique
Group
Shuffle
Reduce
Limit 1

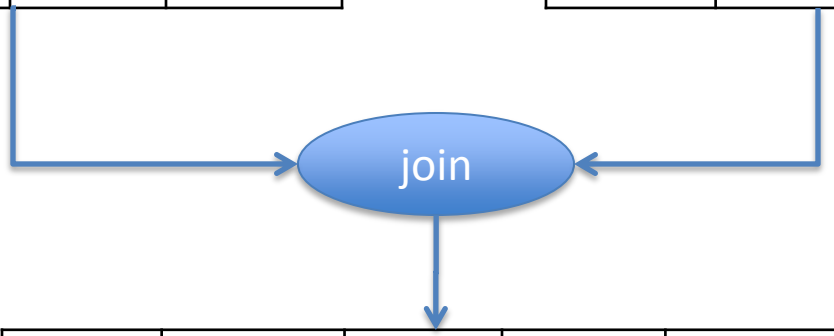

Group
Shuffle
Reduce
Count


Map Phase
Group
Combine
Count
Shuffle
Reduce
Sum
Reduce Phase

# Schema推导、下标推导

| field | ID | name | age |
|---|---|---|---|
| type | uint64 | string | int32 |
| index | 2 | 5 | 9 |

| field | ID | score |
|---|---|---|
| type | uint64 | double |
| Index | 0 | 1 |

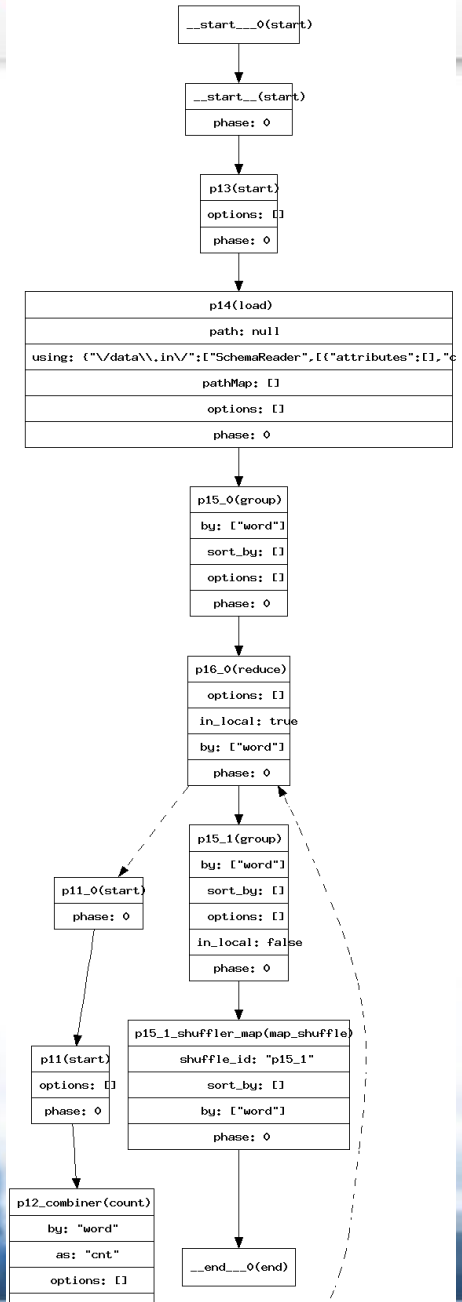


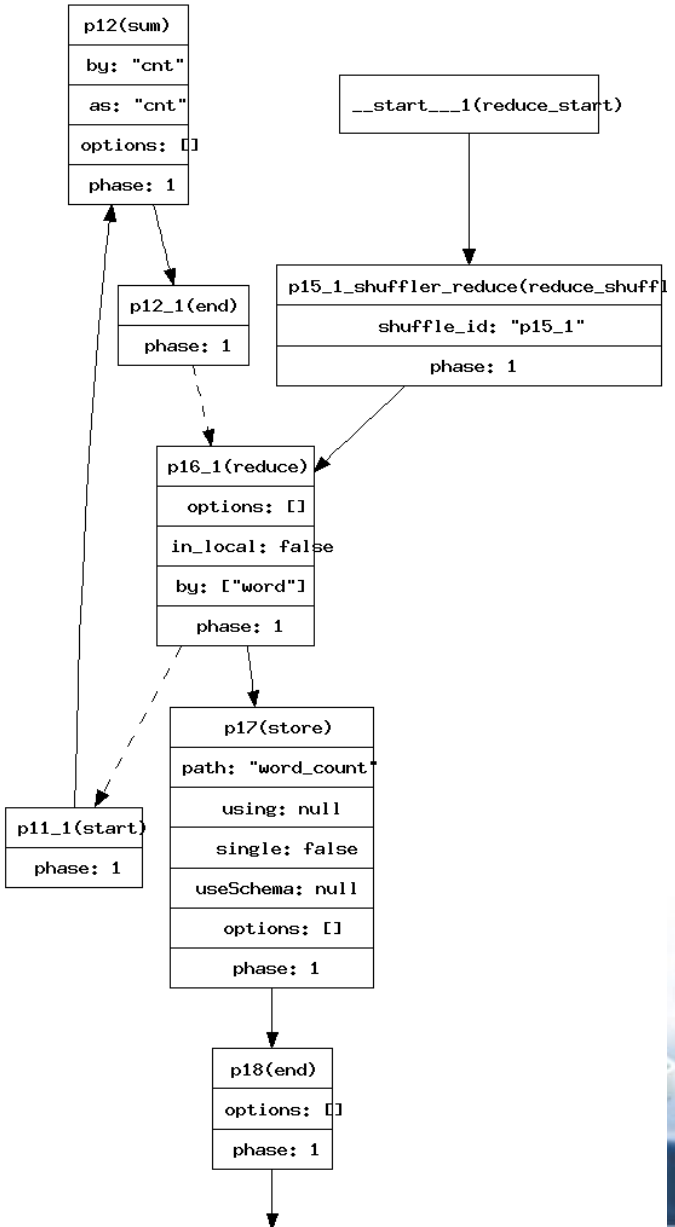

| Field | ID | name | age | Score |
|---|---|---|---|---|
| Type | Uint64 | string | int32 | double |
| Index | 2 | 5 | 9 | 10 |

# 优化

- 多任务合并
- 等价算子合并
- Combiner优化
- Cached Combiner优化
- 同key Join合并优化
- 公共子表达式提取
- 核心思想
  - 减少作业轮数、减少I/O、减少重复计算

# 数据流图生成


__start___0(start)
__start__(start)
phase: 0
p13(start)
options: []
phase: 0
p14(load)
path: null
using: {"\/data\\.in\/":["SchemaReader",[{"attributes":[],"
pathMap: []
options: []
phase: 0
p15_0(group)
by: ["word"]
sort_by: []
options: []
phase: 0
p16_0(reduce)
options: []
in_local: true
by: ["word"]
phase: 0
p11_0(start)
phase: 0
p15_1(group)
by: ["word"]
sort_by: []
options: []
in_local: false
phase: 0
p11(start)
options: []
phase: 0
p15_1_shuffler_map(map_shuffle)
shuffle_id: "p15_1"
sort_by: []
by: ["word"]
phase: 0
p12_combiner(count)
by: "word"
as: "cnt"
options: []
__end___0(end)


p12(sum)
by: "cnt"
as: "cnt"
options: []
phase: 1
__start___1(reduce_start)
p12_1(end)
phase: 1
p15_1_shuffler_reduce(reduce_shuffl
shuffle_id: "p15_1"
phase: 1
p16_1(reduce)
options: []
in_local: false
by: ["word"]
phase: 1
p17(store)
path: "word_count"
using: null
single: false
useSchema: null
options: []
phase: 1
p11_1(start)
phase: 1
p18(end)
options: []
phase: 1

# 代码生成

0.php (~/doc/disql/tech_salon/word_count) - VIM

0.php (~/doc/disql/tech_salon/word_count) - VIM 86

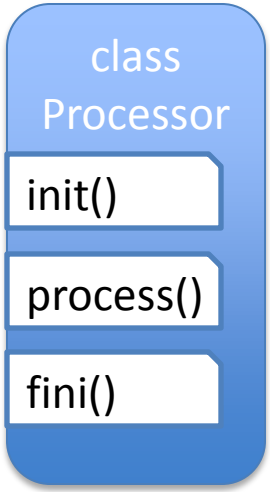
```php
/***** vertex section *****/
$__start__ = new Starter;
$__start__->set_id('__start__');

$p13 = new Starter;
$p13->set_id('p13');

$p14 = new Loader(NULL,array (
  '/data\\.in/' =>
  array (
    0 => 'SchemaReader',
    1 =>
    array (
      0 =>
      array (
        'attributes' =>
        array (
        ),
        'children' =>
        array (
          0 =>
          array (
            'attributes' =>
            array (
            ),
            'name' => 'word',
            'type' => 'string',
          ),
        ),
        'name' => 'res',
        'type' => 'struct',
      ),
    ),
  ),
));
$p14->set_id('p14');

$p15_0 = new Grouper(array (
  0 => 'word',
));
$p12_combiner->set_to('p12'

$p12_0 = new Ender;
$p12_0->set_id('p12_0');

$p15_1 = new Grouper(array
  0 => 'word',
));
$p15_1->set_id('p15_1');

$p15_1_shuffler_map = new M
  0 => 'word',
),NULL,array (
));
$p15_1_shuffler_map->set_id

$__start___0 = new Starter;
$__start___0->set_id('__sta

$__end___0 = new Ender;
$__end___0->set_id('__end_

/***** edge section *****/
$__start__->follow($__start
$p13->follow($__start__);
$p14->follow($p13);
$p15_0->follow($p14);
$p16_0->follow($p15_0);

$p11->follow($p11_0);
$p12_combiner->follow($p11
$p12_0->follow($p12_combine
$p15_1->follow($p16_0);
$p15_1_shuffler_map->follow

$__end___0->follow($p15_1_s
/***** child section *****/
$p16_0->set_children($p11_0
/***** excution section ***
$__start___0->init(0);
```

0.cpp (~/doc/disql/tech_salon/word_count) - VIM

0.cpp (~/doc/disql/tech_salon/word_count) - VIM 78x41

```cpp
#include"Starter___start__.h"
#include"Starter_p13.h"
#include"Loader_p14.h"
#include"Grouper_p15_1.h"
#include"MapShuffler_p15_1_shuffler_ma
#include"Starter___start___0.h"
#include"Ender___end___0.h"
#include<disql/types.h>
#include<disql/Context.h>
#include<disql/Dumper.h>
using namespace disql;

int main(int argc, char *argv[]){
Context::initialize(argc, argv);

bsl::Exception::set_line_delimiter("\n
try{
/*****  vertex section  *****/
Starter___start__ *__start__ = new Sta
Starter_p13 *p13 = new Starter_p13(1);
Loader_p14 *p14 = new Loader_p14(2);
Grouper_p15_1 *p15_1 = new Grouper_p15
MapShuffler_p15_1_shuffler_map *p15_1_
Starter___start___0 *__start___0 = new
Ender___end___0 *__end___0 = new Ender

/*****  child section  *****/

/*****  edge section  *****/
__start__->follow(*__start___0);
p13->follow(*__start__);
p14->follow(*p13);
p15_1->follow(*p14);
p15_1_shuffler_map->follow(*p15_1);

/*****  edge section  *****/
p12->follow(*p11_1);
p12_1->follow(*p12);

p16_1->follow(*p15_1_shuffler_r
p17->follow(*p16_1);
p18->follow(*p17);
__end__->follow(*p18);
p15_1_shuffler_reduce->follow(*

__end___1->follow(*__end__);

/*****  exec section  *****/
bsl::var::IFactory& factory = C
Value main_record(factory);
__start___1->init(0);
__start___1->process(0, main_re
__start___1->fini(0);
/*****  clean section  *****/
delete p12;
delete p12_1;
delete p11_1;
delete p16_1;
delete p17;
delete p18;
delete __end__;
delete p15_1_shuffler_reduce;
delete __start___1;
delete __end___1;
} catch(bsl::Exception &e){
    fprintf(stderr, "BSL EXCEPT
} catch(std::exception &e){
    fprintf(stderr, "STD EXCEPT
} catch(...){
```

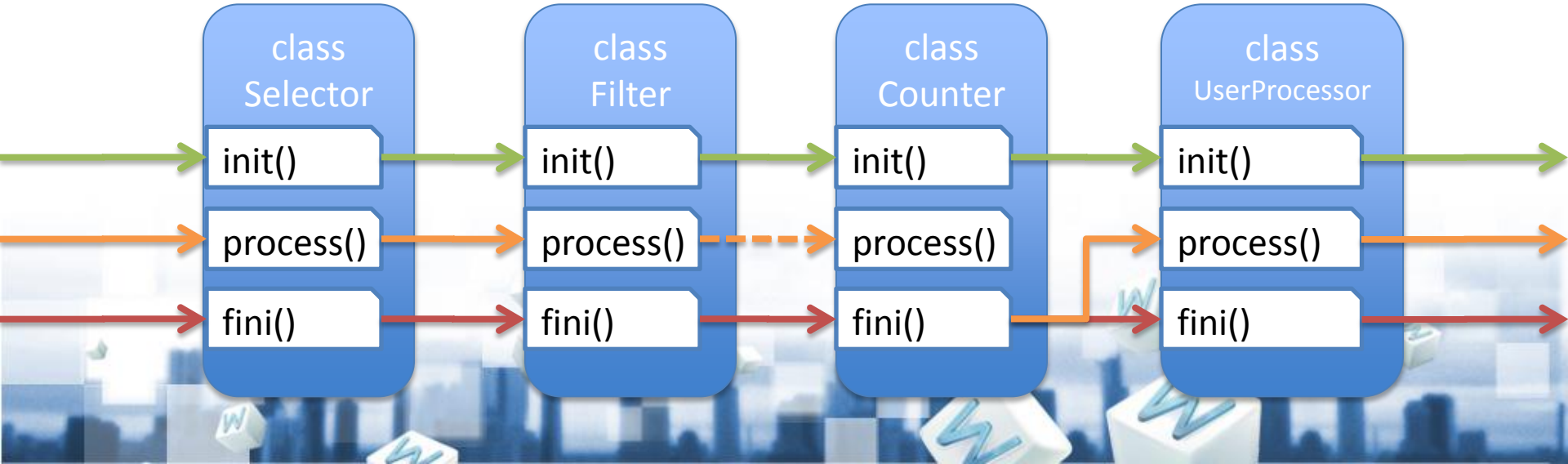
# Processor模型——Pipes & Filter模式




初始化一组数据处理

处理一组中的一条数据记录（多次调用）

结束一组数据处理



## 分析程序数量增长

| 程序类型 | 4月1日 | 10月27日 | 增长 | 增长百分比 |
|---|---|---|---|---|
| 简单编辑 | 3540 | 4761 | 1221 | +34.5% |
| DQuery模式 | 1153 | 3359 | 2206 | +191% |
| 复杂编辑 | 1569 | 2963 | 1394 | +88.9% |

## 分析程序输入量

| 程序类型 | 占比 |
|---|---|
| 简单编辑 | 24% |
| DQuery模式 | 43% |
| 复杂编辑 | 33% |

} 67%

## LSP平台用户数量

| 角色 | 数量 | 占比 |
|---|---|---|
| PM | 1352 | 47.4% |
| RD | 1174 | 41.2% |
| OP | 190 | 6.66% |
| 其他 | 136 | 4.77% |
| 总数 | 2852 | 100% |

# 总结与问答

- LOG中自有黄金屋
- 日志分析基本过程
- 百度日志分析成长历程
- 深入LSP平台
- 深入DISQL语言
- 总结与问答

# 总结与问答

- 日志的价值
  - 了解用户 ● 了解自己
- 日志分析基本过程
  - 提取与传输 ● 解析与过滤 ● 计算 ● 使用（报表、图表、回馈线上...）
- 百度日志分析成长历程
- 深入LSP平台
  - 平台架构 ● 平台UI ● 三种编辑模式
- 深入DISQL语言
  - 一个例子 ● 几个特点 ● 前端处理 ● 中间语言翻译 ● 运行时
- 采用情况
- 问题
  - 亦可通过微博(@陈晓鸣在百度)或邮箱(chenxiaoming@baidu.com)提问
  - 请关注Hadoop in China大会12月2日2时20分：《DISQL2.0》

我们还有很多**非常有挑战、非常有用、非常好玩**的问题急需解决……

如果**你**恰好也喜欢解决这些问题……

请发邮件到chenxiaoming@baidu.com

加入**百度**，让我们一起来把它们解决！

谢谢！