

Salesforce Apex Batch 接口介绍

27 APRIL 2016 on Salesforce, Apex

在 Salesforce 里如果需要处理成千上万的数据，但 Apex SOQL 每次最多只能查询 50,000 条数据，DML 可以操作的数据更少只有 10,000 条，所以在要处理更多的数据时，我们就可以使用 Apex 中的 **Batch** 来实现。顾名思义，Batch 就是将数据分块处理。这次在这里详细介绍一下 Batch 几个接口的使用。

Datebase.Batchable

这是创建 Batch 必须实现的接口，该接口封装了三个方法：

(1) start 方法

该方法用于查询数据，将数据封装到 Database.QueryLocator 或 Iterable，并传给 execute 方法。

```
global (Database.QueryLocator | Iterable<sObject>) start(Data
```

需要注意的是，使用 Database.QueryLocator 最多可以查询 五千万 条数据，Salesforce 对 SOQL 查询条数的限制将不起作用，而使用 Iterable，限制依然存在。

(2) execute 方法

对从 start 方法获得的数据进行处理，例如增删改查等操作。execute 方法采用异步的方式执行，提高了数据处理速度。需要注意的是，每一个 execute 方法里 Apex 的 limits 和正常的 transaction 是一样的。

```
global void execute(Database.BatchableContext BC, list<P>){}
```

每个 execute 方法执行都是一个独立的 transaction。例如，如果一个 Batch 需要执行 1000 条数据，并且没有对 scope 进行限制，则默认是每次执行 200 条数据，五个 transaction 同时执行。如果第一个 transaction 成功，但是第二个失败了，第一个 transaction 对数据的更新操作将不会回退。scope 最大是 2,000 条数据。

(3) finish 方法

所有的 execute 执行完毕后，执行一次 finish 方法，我们可以在里做发邮件通知，汇总数据处理结果等。

```
global void finish(Database.BatchableContext BC){}
```

Database.BatchableContext

在 Database.Batchable 接口提供的方法中，都需要

Database.BatchableContext 对象作为参数，我们可以通过该对象的 getJobId() 方法查询 AsyncApexJob，对 Batch job 的进程进行追踪。

```
global void finish(Database.BatchableContext BC)
{
    AsyncApexJob a = [SELECT Id, Status, NumberOfErrors,
                          JobItemsProcessed, TotalJobItems, CreatedBy.Email
                      FROM AsyncApexJob WHERE Id = :BC.getJobId()];
}
```

Database.AllowsCallouts

如果需要在 Batch 中进行 HTTP 请求或者使用 Webservice 方法，则要实现 `Database.AllowsCallouts` 接口。

```
estBatch implements Database.Batchable<sObject>, Database.Allow
```

这个接口对于做系统集成是有帮助的，当提示 callout 错误的时候，可以检查下有没有实现这个接口。

Database.Stateful

如果需要实例变量在每次 execute 方法执行后不清空，可以实现 `Database.stateful` 接口。需要注意的是，只有实例变量适用，静态变量依然会重置回原始值。

例如，有的时候我们需要对 batch 操作的数据进行一些简单的求和处理。首先我们声明了一个整数类型的变量，初始值为 0，然后在 execute 方法中对变量进行累加操作，最后在 finish 方法中对结果进行一些处理。如果没有实现 `Database.Stateful` 接口，finish 方法中打印的变量值依然为 0。

```
global class SummarizeAccountTotal implements Database.Batchable
{
    global final String Query;
    global Integer Summary;

    global SummarizeAccountTotal(String queryStr)
    {
        Query = queryStr;
        Summary = 0;
    }

    global Database.QueryLocator start(Database.BatchableContext BC)
    {
        return Database.getQueryLocator(Query);
    }

    global void execute(Database.BatchableContext BC, List<sObject> batch)
    {
        Summary += batch.size();
    }

    global void finish(Database.BatchableContext BC)
    {
        // Print the summary
    }
}
```

```
{
    for(sObject s : scope)
    {
        Summary = Integer.valueOf(s.get('Total__c'))+Summary;
    }
}

global void finish(Database.BatchableContext BC)
{
    system.debug(Summary);
}
}
```

通过上面的介绍，希望大家对 Batch 的接口能有一个大致的了解。当然，batch 还有很多其他的功能，例如使用 `System.scheduleBatch` 方法在特定的时间执行 batch，使用 `Apex Flex Queue` 对 Batch Job 进行排序等，有兴趣的可以深入研究一下，下次再继续为大家介绍。

Vicky Zhang

Share this post

Read [more posts](#) by this author.

0条评论

Meginfo Blog

1 登录 ▾

♥ 推荐

🔗 分享

评分最高 ▾



开始讨论...

通过以下方式登录

或注册一个 DISQUS 帐号 (?)

姓名

来做第一个留言的人吧！

📧 订阅 在您的网站上使用 Disqus 添加 Disqus 添加 🔒 隐私

READ THIS NEXT

怎样解决 Case.Link 无法在 Visualforce Email Template 中使用的问题

首先我们来看一下 Case.Link 的描述，Case.Link 可以返回 Case 在 Salesforce 内部访问的 URL，无论这条记录是在 Sandbox 还是在 Production，它都可以返回正确的

YOU MIGHT ENJOY

使用 Apex 获取审 批记录

为什么要写这篇文章呢？因为本人在实际的项目中确实遇到了这样的问题，并花了一些时间来寻找解决办法。在此拿出来和大家分享一下。这篇文章涉及到了 Salesforce 的 Approval Processes 功能，在此我们只做简单的概述，不做过多的介绍。Approval Processes 是

访问地址。 第一步：创建一个文
本...

Salesforce 中的审批工作流, 用它
可以在 Salesforce...

[联系我们](#) · [关于我们](#) · [加入我们](#) · [隐私策略](#) · [Blog](#) · [AppExchange](#)

电话：010-67713956 邮箱：contactus@meginfo.com



© 2016 文睿创新信息科技（北京）有限公司



官方微信 扫码关注