

文章

[Frank Ma](#) · 五月 24, 2022 阅读大约需 5 分钟

我们如何将报告生成时间减少28倍

同事们，大家好

在这篇文章中，我将告诉你我们如何将报告生成时间从28分钟减少到1分钟。让我告诉你我们是如何实现这一目标的

我希望，如果有必要，你将能够为自己重现同样的结果。这篇文章里有一些有用的链接，所以要读到最后。

让我们开始吧。

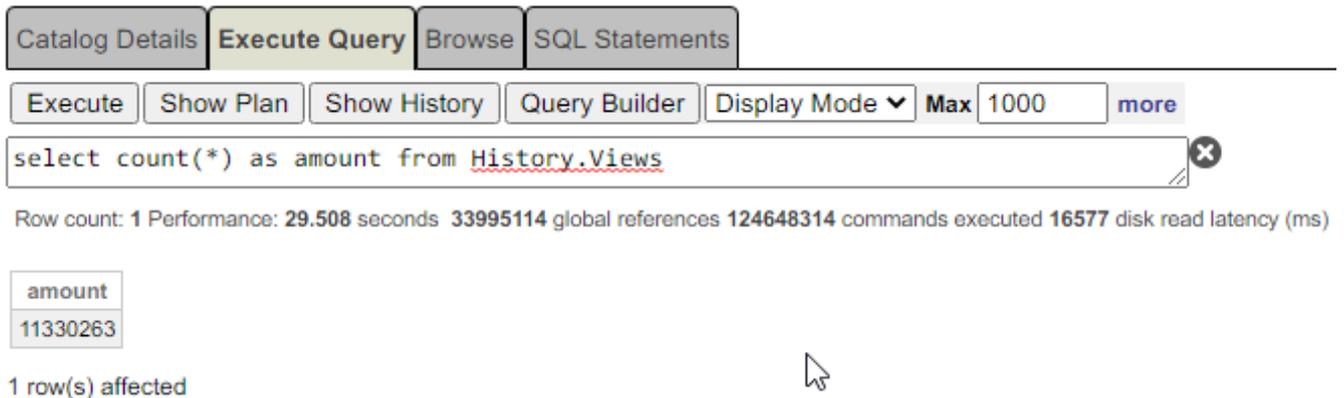
报告

我们使用Adaptive Analytics和InterSystems Reports Server为一家公司做报告。以前，这个报告是以DeepSee的屏幕截图形式生成的。总的来说，这并不坏，但它花费了大量的时间，而且看起来不是很可读。该报告本身由12页组成，为PDF格式。

一般来说，数据不是太大，不会使报告的生成花费很多时间

源数据

然而，在撰写本文时，有一个表包含11,330,263行。不是那么关键，但它造成了延迟。即使是计算行数的查询也需要近30秒

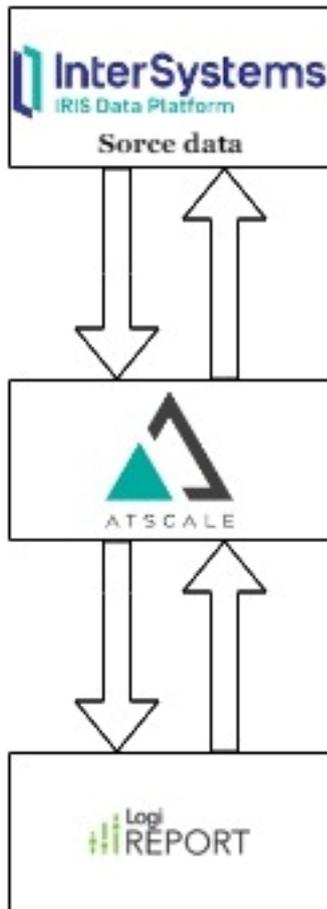


The screenshot shows the InterSystems Reports Server interface. At the top, there are tabs for 'Catalog Details', 'Execute Query', 'Browse', and 'SQL Statements'. Below these are buttons for 'Execute', 'Show Plan', 'Show History', 'Query Builder', and 'Display Mode'. A dropdown menu is set to 'Max' with a value of '1000' and a 'more' link. The query text area contains: `select count(*) as amount from History.Views`. Below the query, the execution results are displayed: 'Row count: 1 Performance: 29.508 seconds 33995114 global references 124648314 commands executed 16577 disk read latency (ms)'. A table shows the result:

amount
11330263

. Below the table, it says '1 row(s) affected'. A mouse cursor is visible over the table.

最初，系统的交互方案是这样的：



Atscale创建了自己的数据缓存，这导致了性能的提高。

Logi使用自己的数据缓存，这稍微加快了报告的开发速度。

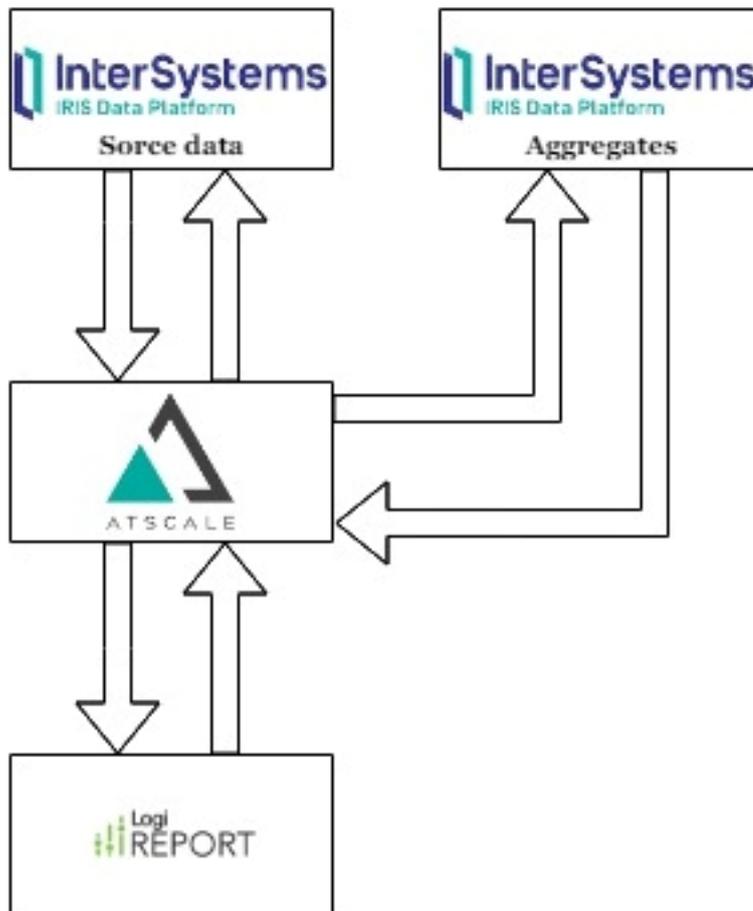
但总的来说，这仍然导致了报告在28分钟内形成的事实。

鉴于报告只有12页，这个速度还是很慢的。

我们甚至故意添加了新的标签，并将报告中的小部件复制到那里，以便在开发或调试时不需要生成整个报告。也许这就是在Logi上开发时的一种日常技巧，或者说是一种正常的开发方法。总的来说，我们在工作中使用了它

在报告生成时，有数百个请求从Logi到Atscale，还有一些请求从Astcale到IRIS。一些单独的查询长达4分钟。有几次，请求一般都是在超时的情况下发出的。

然后我们意识到，这种情况不能再继续下去了。于是我们把自适应分析的主要功能连接起来，方案变成了这样：



这是一个UDAF功能，换句话说，这是个用户定义的聚合函数。实际上这是些汇总表，根据要求，定期地将需要测量的聚合值存储在其中。而最有趣的是，这些汇总表是由Atscale自动创建和更新的。

为了实现聚合，我们编译了isc-aa-udaf包，该包目前在一个私人仓库中，因为根据Atscale的使用条款，它不能被免费分发。

在打开汇总表后，创建服务表花了几分钟，汇总表被计算，总的来说，数据库被加载。但随后真正的解脱开始了。系统开始像它应该的那样工作。以前需要4分钟才能形成的请求，开始在5秒内形成。缓存变得更快。

结果，原先花了28分钟才形成的报告现在开始在1分钟内形成。

值得注意的是，这样的增长更多的是稳定的、生产系统的特征，在这些系统中，立方体是相对稳定的，聚集物被越来越多的收集。当立方体结构发生变化时，在对变化后的立方体的第一次请求中，聚合被重构并重新创建。

我们做了什么

我们所做的基本如下：

1. 安装IRIS Adaptive Analytics UDAF
2. 在IRIS中创建名称空间，用于存储聚合数据
3. 连接IRIS 到 Atscale
4. 定制功能安装模式=在数据仓库中启用定制管理功能
5. Adaptive Analytics已经用数据创建了一个立方体
6. 在Logi中创建报告
7. 测试和测量

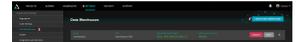
这里的细节：

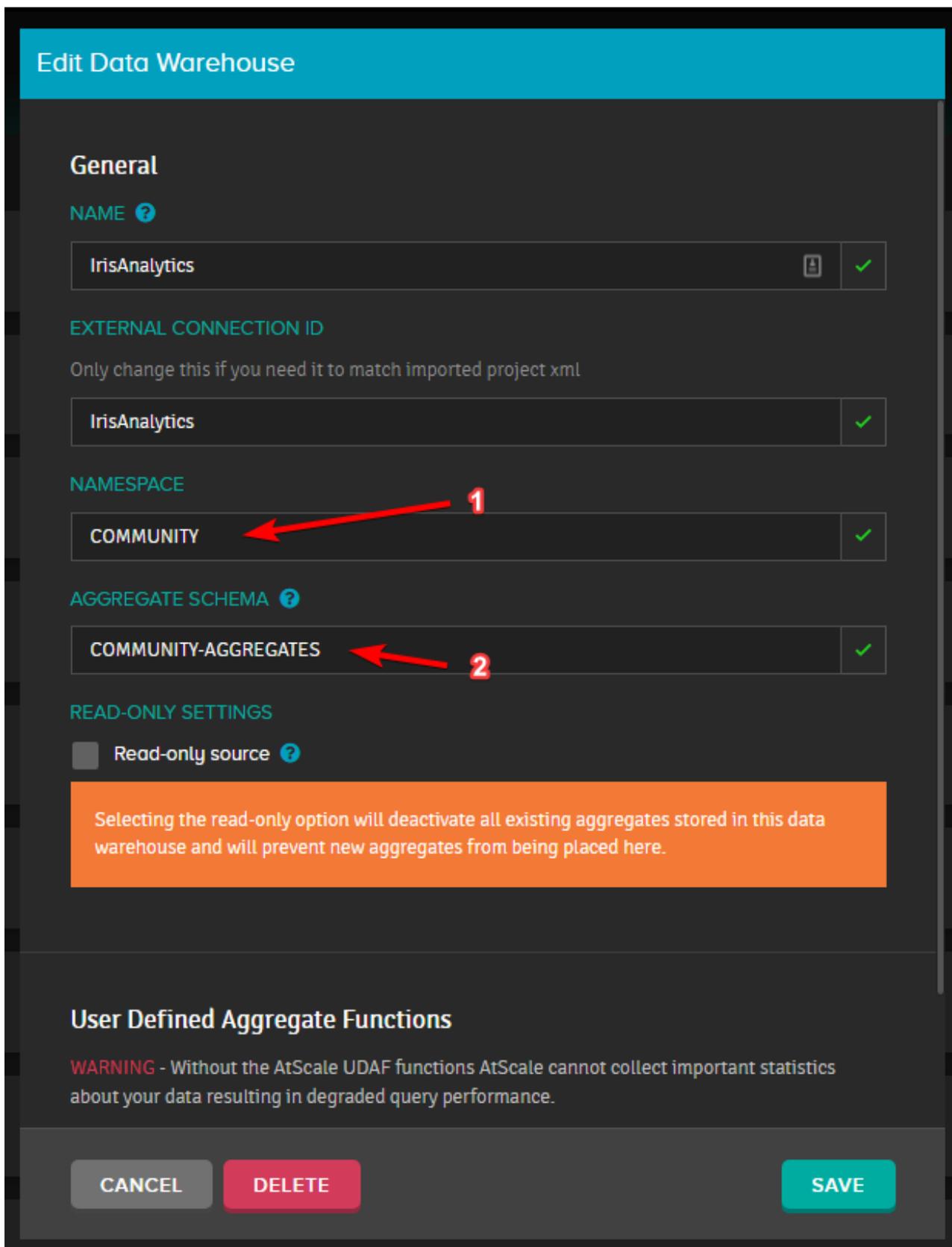
1. InterSystems IRIS已经为UDAF的工作安装了必要的类。我们已经将它们打包在一个名为isc-aa-udaf的ZPM包中，该包存储在zpm注册表pm.intersystems.com中。这个注册表对InterSystems的官方客户是可用的。

iris 命令:

```
zpm " install isc-aa-udaf "
```

2. 一个旨在优化资源使用的可选项：我们为IRIS添加了一个专门的命名空间，它将存储预先计算的聚合值。这个命名空间的名称将在下一步中用到。
3. 将IRIS连接到Atscale作为立方体（Cube）的数据源。转到设置（Setting），然后是数据仓库（Data Warehouses），最后是创建数据仓库（CREATE DATA WAREHOUSE）。





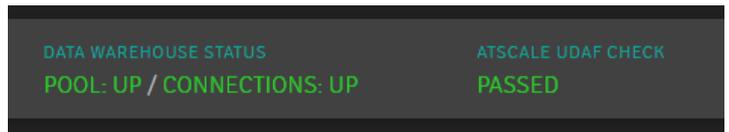
在字段号1中指定了解析后的数据所存储的命名空间

在字段号2中，我们指定了在步骤2中创建的用于存储聚合的命名空间。这个值可以与1的值相同，在这种情况下，集合体将被存储在数据的旁边。在这种情况下，数据源不能是只读的。

4. 我们将定制功能安装模式 (CUSTOM FUNCTION INSTALLATION)

MODE) 设置为用户管理 (Customer Managed)，因为我们之前在第一步安装了UDAF。如果你指定无 (None) 模式，那么即使安装了UDAF，这些功能也不会被使用，也不会有性能上的提高。

如果一切操作正确，那么UDAF检查显示绿色。



5. 使用创建的数据仓库创建了一个项目和一个立方体。这是一个漫长而令人兴奋的过程。我不会在这里详细谈论这个问题，已经有好几篇关于它的文章，包括我的文章《[如何轻松开始在Adaptive Analytics + InterSystems Reports中工作](#)》。
6. Atscale上发布的项目为Logi报告创建了数据源连接。在之前的文章中，我也介绍了如何创建报告。链接为：《[如何轻松开始在Adaptive Analytics + InterSystems Reports中工作](#)》。
7. 测试和测量。这里很有趣。我最初设计的报告没有启用UDAF。正因为如此，一些请求被执行了4分钟或更长时间。由于该报告由12页组成，完成报告的时间平均为28分钟。

启用UDAF后，Atscale系统在自动模式下用一段时间加载数据源。她会自己计算将在报告中使用的实际查询，并为它准备预先计算的数值。此外，InterSystems报告是基于计算出来的参数，InterSystems报告本身对这些参数进行了部分缓存，AtScale系统给出了额外的优化，它缓存了执行相同查询的结果并即时返回，而不是重新发送到数据源。

在所述的捆绑工作包中，还有一个有趣的点：报告生成的频率越高，制作报告的时间就越短。

根据所有操作和几次测量的结果，生成12页报告的时间开始是60秒，也就是1分钟。

差别是28倍。

同时，类似的报告，在结构上完全相同，但从其他数据库中获取数据，其构建速度也有类似的提高。

基于我们所看到的，我们做出了一个明确的结论，推荐在所有未来的项目中使用这个捆绑包。它可以提高开发速度，提升调试速度，并减少向这些报告的商业消费者交付报告的时间。

我希望将来我们能够从IRIS - AtScale -

Logi捆绑系统中提取更多的性能，并能够与你分享我们发现的解决方案。

如果你也有这类提升工具链性能的经验和我们分享，我将非常感激。

[#自适应分析 \(Adaptive Analytics \)](#) [#InterSystems Reports \(Logi\)](#) [#InterSystems IRIS](#)

源

URL:

<https://cn.community.intersystems.com/post/%E6%88%91%E4%BB%AC%E5%A6%82%E4%BD%95%E5%B0%86%E6%8A%A5%E5%91%8A%E7%94%9F%E6%88%90%E6%97%B6%E9%97%B4%E5%87%8F%E5%B0%9128%E5%80%8D>