



集算器

创新大数据计算引擎

银行业大数据量清单报表

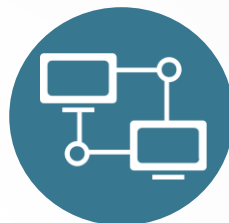
润乾软件出品



项目背景



大清单报表是银行业务中常见需求，单次检索数据量可达百万、甚至千万级别；客户后台采用了HBase做数据存储层，使得开发这类报表面临了很多难题！



硬编码难度高

后台采用hbase做数据存储，要通过api实现大报表分页，硬编码实现难度高



开发工作量大

hbase实现分页代码量大，每个报表需求平均开发周期为7天，且无复用价值



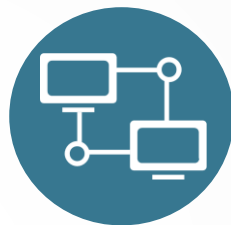
功能存在残缺

若报表系统实现独立分页后，打印和导出等需求依然无法解决



常见方案的不足

增加前置数据库，通过ETL工具定时从HBase同步数据到前置库，然后利用关系数据库的分页机制满足需求；但增加前置库不仅造成重复建设，且分页方案并不理想！



结构变化大

重复的开发工作量，以增加数据库提升软件成本，让管理和维护变的更为复杂



翻页效率差

页码小时，感觉不明显；页码较大时，翻页会有明显的等待感



游标与SQL混用

向后翻页时用游标，一旦需要向前翻页，就重新执行取数SQL，解决问题不彻底

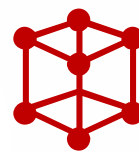


如何从根本上解决问题？

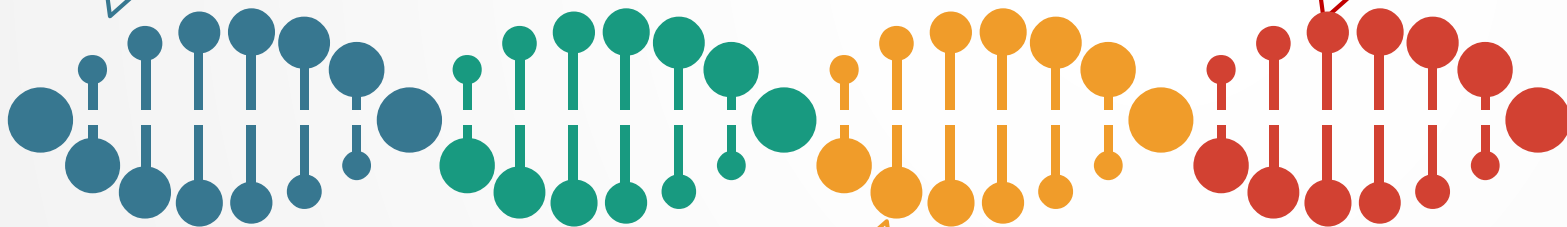
取数和呈现做成两个异步线程，边取数边呈现，达到快速响应；且不依赖于数据库的分页取数能力！



取数线程发出命令后就不断取出数据并缓存到本地存储中，呈现线程根据页数计算出行数到本地缓存中去获取数据显示



只要已经取过的数据就能快速呈现，不会有等待感，还没取到的数据需要等待一下也是正常可理解的

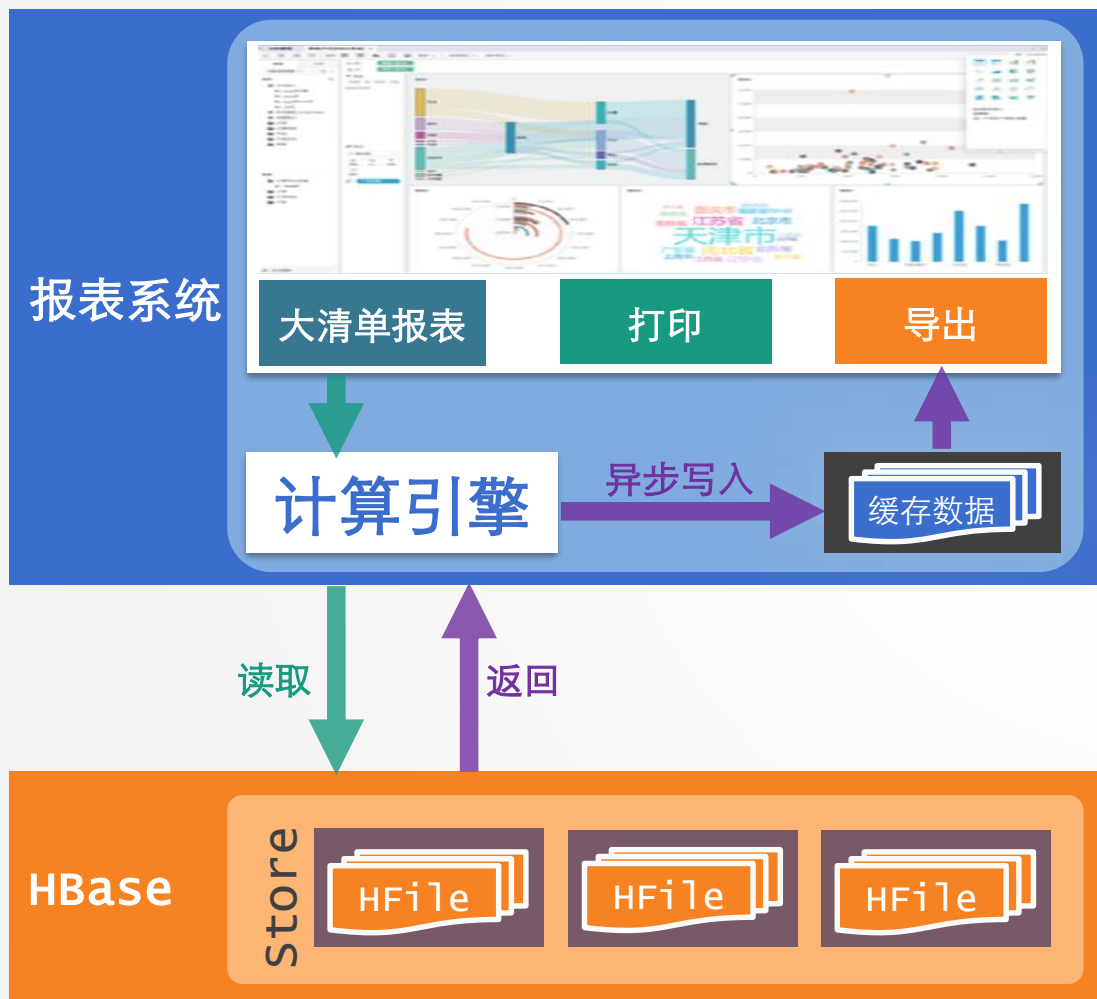


取数线程所涉及到的命令，在数据源中保持同一个事务，也不会有不一致的问题

引入数据计算层



经过需求分析和产品选型，最终确定采用润乾报表加上润乾集算器的方式来应对批量大清单报表开发难得诸多问题！



简单开发

无须复杂环境配置与应用层代码引用，简化了大清单报表后端编码难度，降低对开发人员要求



完备计算能力

可以直接连HBase实施计算，弥补了非关系型数据库计算能力的不足



高性能存储

具备高性能缓存功能，可以按行号随机访问记录，而不用每次通过遍历查找数据，极大地改善了用户体验



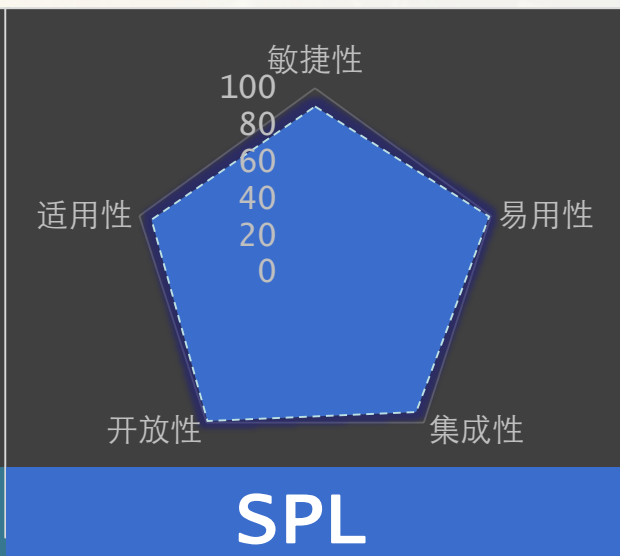
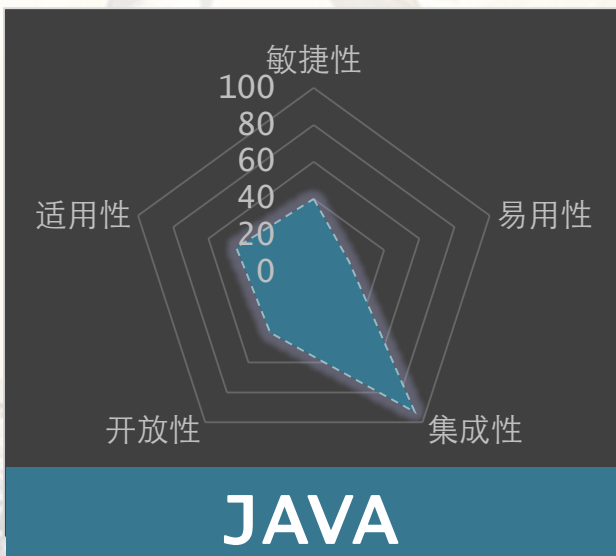
报表模块独立化

算法外置于脚本，文件形式易于管理；解释执行的语言，功能模块热插拔，实现独立维护

实测：集算器在各方面的工作量评估

对比指标	JAVA	集算器(SPL)	提升
代码量 (单一场景)	200多行	5行	平均40倍
工作量	41人天	8人天	平均5倍
可优化	无	容易	质变
耦合性	高	低	模块化、工具化
维护性	较差	好	易管理、热切换

提升开发效率
40倍!



集算器(SPL)伪代码风格举例：

	A	B	C
1	=hbase_open("hdfs://192.168.0.8", "192.168.0.8")	/创建hbase数据库连接	
2	=hbase_filter("ColumnCountGetFilter",3)	/通过hbase过滤器筛选数据	
3	=hbase_scan@c(A1,"emp";filter:A2)	/查询hbase数据库中的数据，结果值返回游标	
4	return A3	/提供给报表使用	
5	>hbase_close(A1)	/关闭hbase数据库连接	SPL

为什么集算器能如此提高效率



敏捷语法体系

降低算法实现复杂度，极大地提高开发效率



开发工具化

不必做复杂的环境配置，可编写简单代码实现复杂计算



多样性数据源

丰富的多样性数据源接口，直接取数计算

创新技术 推动应用进步!

