Pentaho 工具
使用手册

马腾，李洪宇版

本：1.0

名目

BI 介绍

1. BI 根底介绍

BI〔BusinessIntelligence〕即商务智能，它是一套完整的解决方案，利用数据仓库、数据挖掘技术对客户数据进展系统地储存和治理，并通过各种数据统计分析工具对客户数据进展 分析，供给各种分析报告，为企业的各种经营活动供给决策信息。其中的关键点是数据治理， 数据分析，支持决策。
依据要解决问题的不同，BI 系统的产出一般包括以下三种：

2. BI 系统的产出

2.1 固定格式报表

固定格式报表是 BI 最根本的一种应用，其目的是展现当前业务系统的运行状态。固定格式报表一旦建立，用户就不行以更改报表的构造，只能依据数据库的数据不断刷报表，以便 取得较的数据。在 pentaho 产品线中，我们使用 pentaho report designer 来实现固定格式报表的需求。

2.2OLAP 分析

OLAP 分析是指创立一种动态的报表展现构造，用户可以在一个 IT 预定义的数据集中自由选择自己感兴趣的特性和指标，运用钻取，行列转换等分析手段实现得到学问，或者验证假设的目的。在 pentaho 产品线中，我们使用 Saiku 来实现 OLAP 分析的需求。

2.3 数据挖掘

数据挖掘是 BI 的一种高级应用。数据挖掘是指从海量数据中通过数据挖掘技术得到有用的学问，并且以通俗易懂的方式表达学问，以便支持业务决策。在pentaho 产品线中，我们使用 weka 来实现数据挖掘的需求。

Pentaho 产品介绍

1. 产品介绍

Pentaho 是世界上最流行的开源商业智能软件，以工作流为核心的、强调面对解决方案而非工具组件的 BI 套件，整合了多个开源工程，目标是和商业BI 相抗衡。它是一个基于java 平台的商业智能套件，之所以说是套件是由于它包括一个web server 平台和多个工具软件：报表，分析，图表，数据集成，数据挖掘等，可以说包括了商业智能的方方面面。

2. Pentaho 架构图

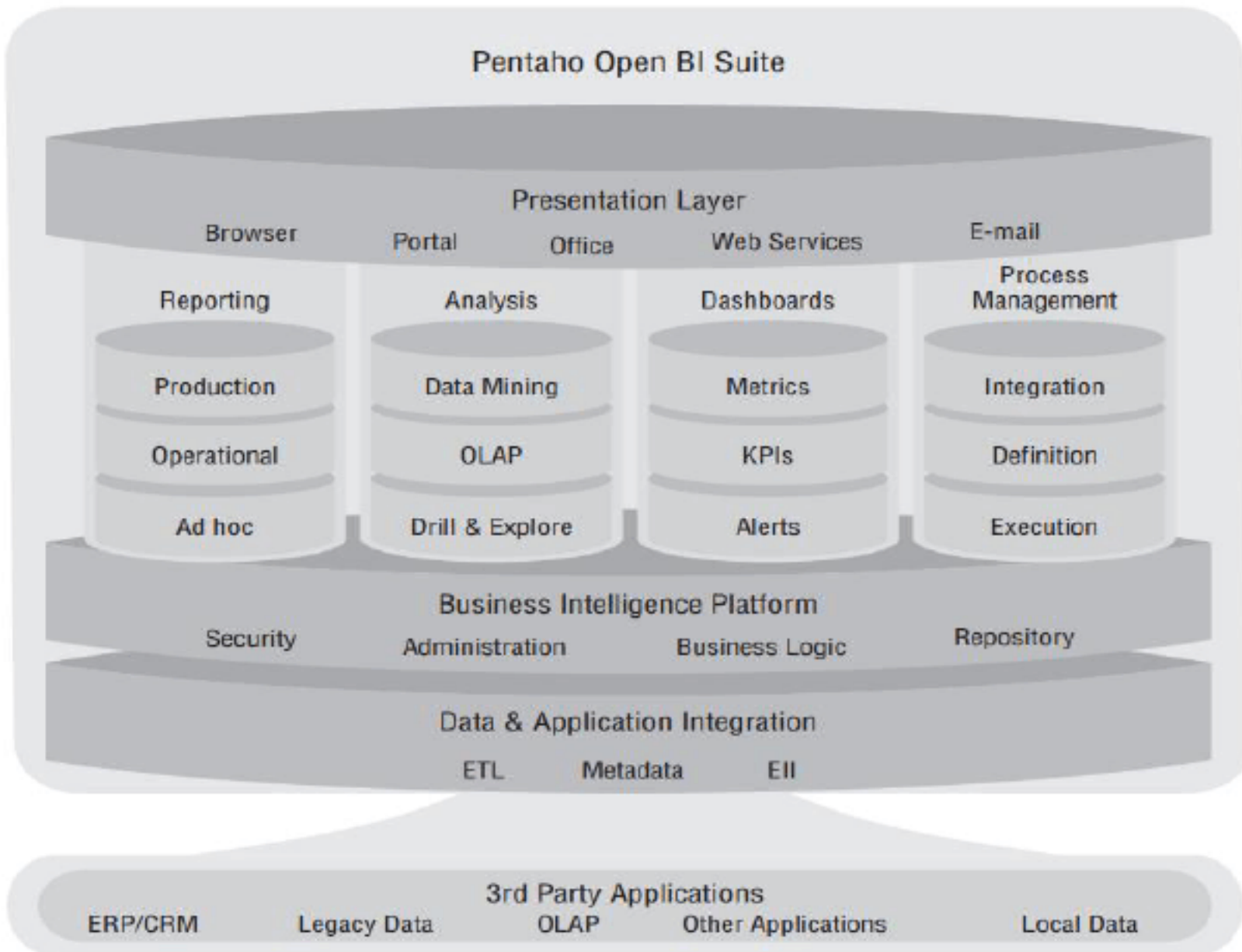
Pentaho 的架构图如下，简要解释如下：

3rd party applications 指交易系统，也就是数据仓库的原系统。

Data & Application Integration 主要指定义数据仓库的元数据，在数据仓库构造设计完毕后，通过 ETL 过程将原系统数据送入数据仓库。

Business Intelligence Platform 指 pentaho 供给的 BI 平台，在这个平台上可以进展平台安全设置，平台治理之类的工作，这个平台也是BI 效劳的根底。Reporting，Analysis，Dashboards，Process Management 是基于 BI 平台上 Pentaho 可以实现的效劳，比方报表，分析，仪表盘，效劳自动掌握等。

Presentation Layer 指展现层，在这一层，我们可以把其下层做好的报表等分析结果通过门户网站，Email 等各种方式展现给用户。



Pentaho 产品线设计

1. 产品线设计

Pentaho 作为一个开源的 BI 套件，商业版与社区版加起来共有几十种产品。考虑到恒信实际业务开展的状况，以及将来可能的需求，确定产品线如下。

BI Function	**Product**
ETL	Kettle
Metadata Management	Pentaho Metadata Editor (PME)
OLAP	Saiku + Schema Workbench
Report tools	Fixed report：Pentaho report designer
	Ad-hoc report：Saiku
	Dashboard：CDE
Data Mining	Weka
BI platform	Pentaho BI Platform
R language	R
Big Data	Pentaho for Big Data

产品线的设计并非一成不变，随着需求的增加，当某些需求无法利用现有的产品线实现时，可以连续添加组件，以便形成更为完善的BI 体系。

Pentaho BI Platform 安装

1. 安装步骤

将下载下来的 biserver-ce-X.X.X-stable.zip 文件解压到 D:\下，将会产生 administration-console 和 biserver-ce 两个文件夹，前者是 pentaho 掌握台，后者是 pentaho BI 效劳器。
默认时，Pentaho BI 平台会使用内置的JRE，它位于 D:\biserver-ce\jre 位置。假设用户机器上安装了 JDK，并设置了 JAVA_HOME，则 Pentaho BI 平台会使用用户指定的 JDK。运行 D:\biserver-ce>下的“start-pentaho.bat”批处理脚本能够启动 Pentaho BI 效劳器，它运行在 Apache Tomcat容器中，并承受了 HSQLDB 数据库〔〕。

2. 启动/停顿 BI server

现在，翻开扫瞄器，并访问 ://localhost:8080/pentaho，则将看到登录界面，

当 joe/password 用户登录后，BI 效劳器的主界面将呈现在眼前。

假设需要停顿 Pentaho BI 效劳器，则于 D:\biserver-ce 名目下运行“stop-pentaho.bat”批处理脚本即可。它将同时停顿 Pentaho BI 效劳器和 HSQLDB 数据库。

3. 启用/停顿 Pentaho 治理掌握台

于 D:\administration-console 名目运行如下“start-pac.bat”批处理脚本能够启动 Pentaho 治理掌握台。默认时，它宿主在 Jetty Web 容器中。将扫瞄器定位到 ://localhost:8099/网址后，并输入默认的 admin/password 用户，即可登录到 Pentaho 治理掌握台中。Pentaho 治理掌握台是整个 BI 平台的重要后端软件，系统治理员通过它能够完成各类操作，比方维护用户及角色信息、注册的业务库〔数据库连接〕、掌握 BI 效劳器中的各种敏感信息、使用调度效劳等。

假设要停顿 Pentaho 治理掌握台，则于 D:\administration-console 名目下运行“stop-pac.bat”批处理脚本即可。

4. HSQLDB 迁移到 MySQL DB

4.1 迁移缘由

Pentaho BI 效劳器的很多重要信息存储在数据库中，其默认使用 HSQLDB 数据库，即借助它存储自身的资料库，比方 Quartz 调度信息、业务资料库连接信息〔数据源〕等。HSQLDB 是不能够支撑真实的企业应用的，生产环境必需替换它，因此我们需要将 HSQLDB 迁移至 MySQL。

4.2 创立 MySQL 数据库

分别执行下面加粗的 sql 脚本。先后挨次不限。运行方法是多种的，可以通过 MySQL Workbench 导入工具实现。我们设定导入的 MySQL 数据库地址为 jdbc:mysql://localhost:3307，用户名 root，密码 root。

biserver-ce\data\mysql5\create_quartz_mysql.sql

biserver-ce\data\mysql5\create_repository_mysql.sql

biserver-ce\data\mysql5\create_sample_datasource_mysql.sql

其中

1. create_repository_mysql.sql

创立 hibernate 数据库，用于存储用户授权认证，solution repository 以及数据源。

2. create_sample_datasource.sql

为 sample 数据添加 pentaho 全部根本的实例数据源。

3. create_quartz_mysql.sql

为 Quartz 打算任务器创立资源库。

4.3 配置 Pentaho

1．给 pentaho 添加 JDBC 文件

下载 MySQL 的 JDBC 驱动：MySQL—mysql-connector-java-x.x.x.jar

将其拷贝至 biserver-ce\tomcat\lib 和 administration-console\jdbc 下，以便 BI service 和 administration console 访问 MySQL 数据库。

2．修改以下文件

biserver-ce\pentaho-solutions\system\applicationContext-spring-security-jdbc.xml

biserver-ce\pentaho-solutions\system\applicationContext-spring-security-hibernate.properties

biserver-ce\pentaho-solutions\system\hibernate\hibernate-settings.xml

biserver-ce\pentaho-solutions\system\hibernate\mysql5.hibernate.cfg.xml

biserver-ce\tomcat\webapps\pentaho\META-INF\context.xml

以上文件主要是替换 SQL 驱动，SQL 用户名与密码等信息。修改详情如下，红色局部代表文

件名，黑体代表更改点。

applicationContext-spring-security-jdbc.xml

<bean id=“dataSource“

class=“org.springframework.jdbc.datasource.DriverManagerDataSource“

>

<property name=“driverClassName“ value=“**com.mysql.jdbc.Driver**“ />

<property name=“url“

value=“**jdbc:mysql://localhost:3307/hibernate**“ />

<property name=“username“ value=“**root**“ />

<property name=“password“ value=“**root**“ />

</bean>

applicationContext-spring-security-hibernate.properties

jdbc.driver=**com.mysql.jdbc.Driver**

jdbc.url=**jdbc:mysql://localhost:3307/hibernate**

jdbc.username=**root**

jdbc.password=**root**

hibernate.dialect=**org.hibernate.dialect.MySQL5InnoDBDialect**

hibernate-settings.xml

<config-file>**system/hibernate/mysql5.hibernate.cfg.xml**</config-file>

mysql5.hibernate.cfg.xml

<property name=“connection.driver_class“>**com.mysql.jdbc.Driver**</property>

<propertyname=“connection.url“>**jdbc:mysql://localhost:3307/hibernate**</property>

<property name=“dialect“>org.hibernate.dialect.MySQL5InnoDBDialect</property>

<property name=“connection.username“>**root**</property>

<property name=“connection.password“>**root**</property>

context.xml

<Resource name=“jdbc/Hibernate“ auth=“Container“ type=“javax.sql.DataSource“

factory=“org.apache mons.dbcp.BasicDataSourceFactory“ maxActive=“20“

maxIdle=“5“ maxWait=“10000“ username=“**root**“ password=“**root**“

driverClassName=“**com.mysql.jdbc.Driver**“

url=“**jdbc:mysql://localhost:3307/hibernate**“ validationQuery=“**select 1**“ />

<Resource name=“jdbc/Quartz“ auth=“Container“ type=“javax.sql.DataSource“

factory=“org.apache mons.dbcp.BasicDataSourceFactory“ maxActive=“20“

maxIdle=“5“ maxWait=“10000“ username=“**root**“ password=“**root**“

driverClassName=“**com.mysql.jdbc.Driver**“ url=“**jdbc:mysql://localhost:3307/quartz**“

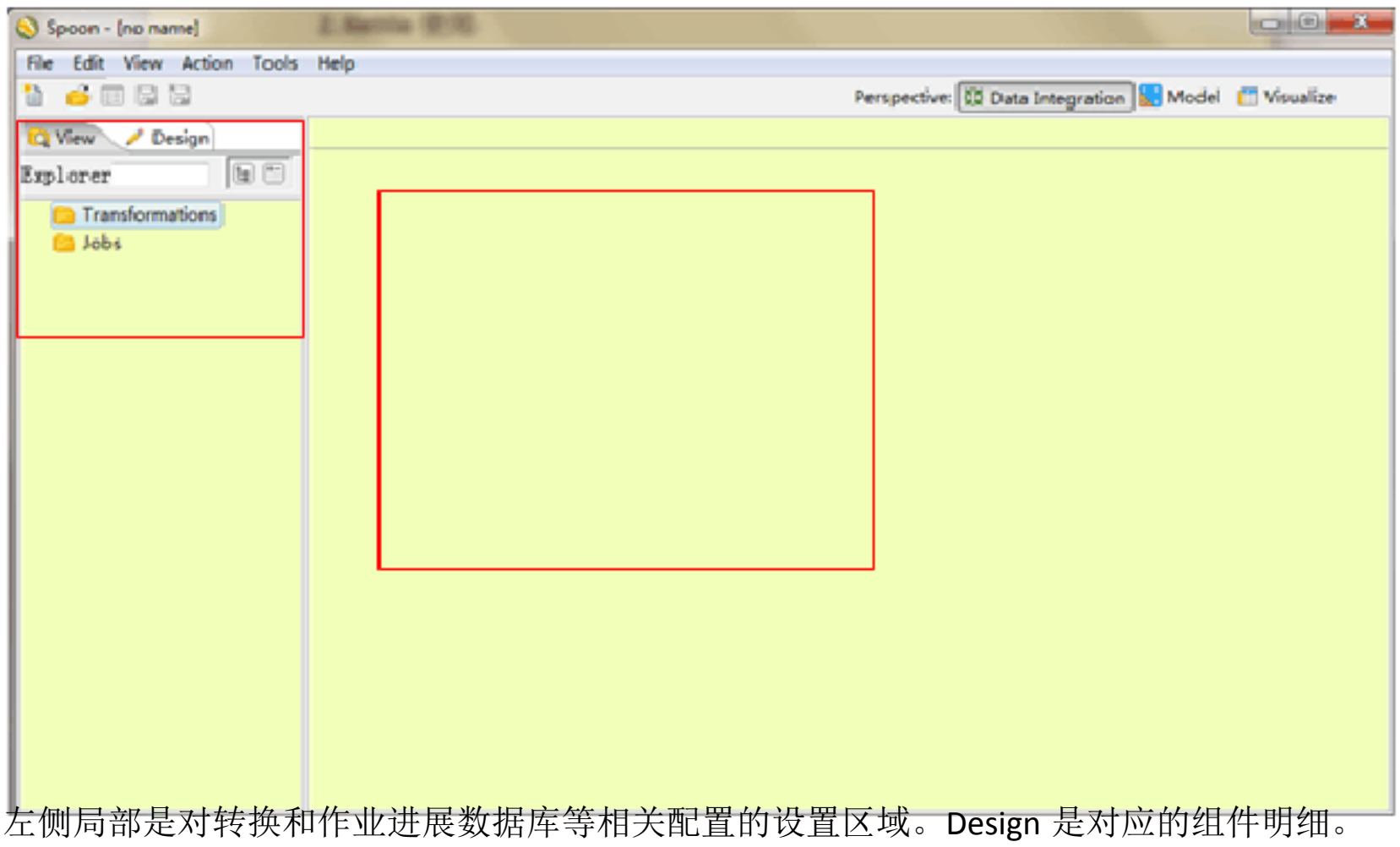
validationQuery=“**select 1**“/>

现在可以启动 pentaho 效劳了。可以看到 BI 环境预备就绪。

PentahoDataIntegration ------Kettle

1. Kettle 安装

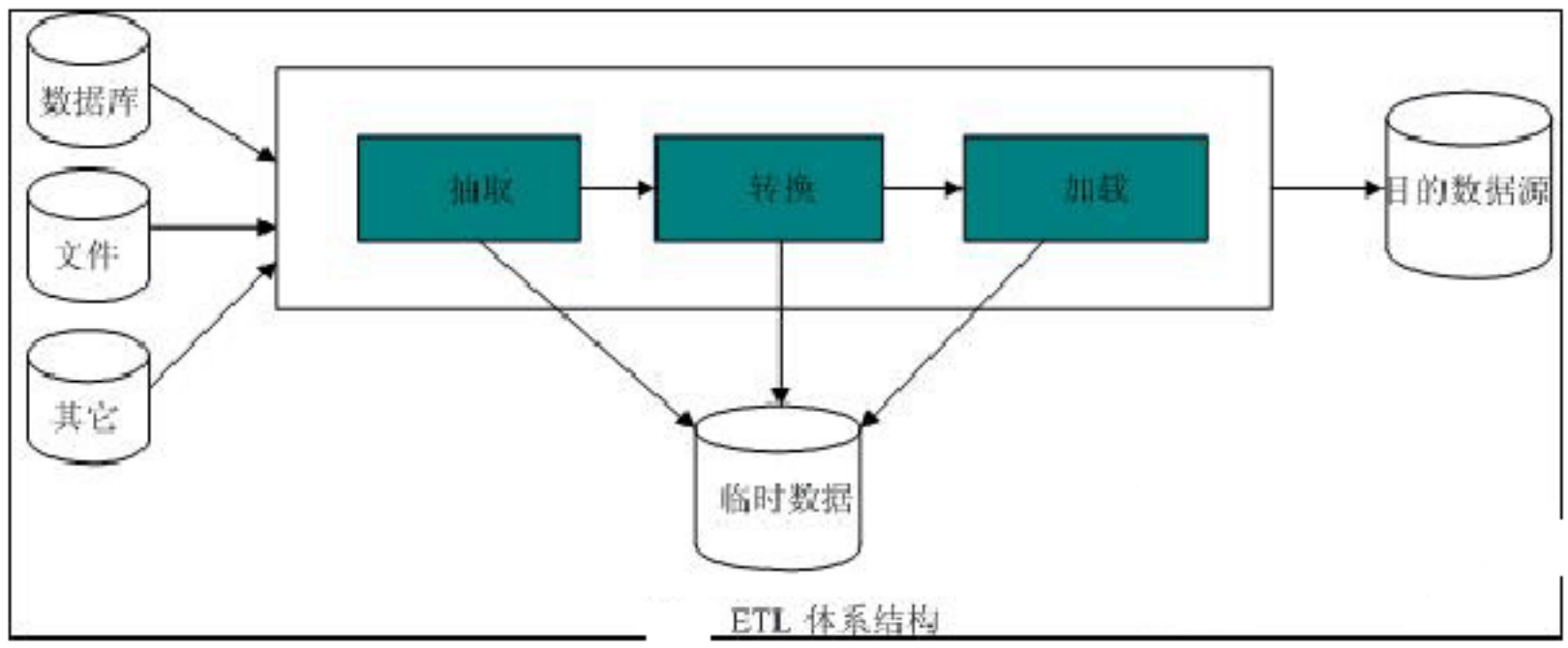
要运行此工具你必需安装 Sun 公司的 JAVA 运行环境 1.4 或者更高版本。Kettle 的下载可通过 。我们将下载的 pdi-ce-4.4.0-stable.zip 解压到想要放置的路径，并执行这一名目中 Spoon.bat 文件，Kettle的主界面将呈现在我们面前。



左侧局部是对转换和作业进展数据库等相关配置的设置区域。Design 是对应的组件明细。右边局部是 ETL 的主界面，我们需要把 Design 页面中相关组件在上面设计展现。

Kettle 中有两种脚本文件，transformation 和 job，transformation 完成针对数据的根底转换，job 则完成整个工作流的掌握。

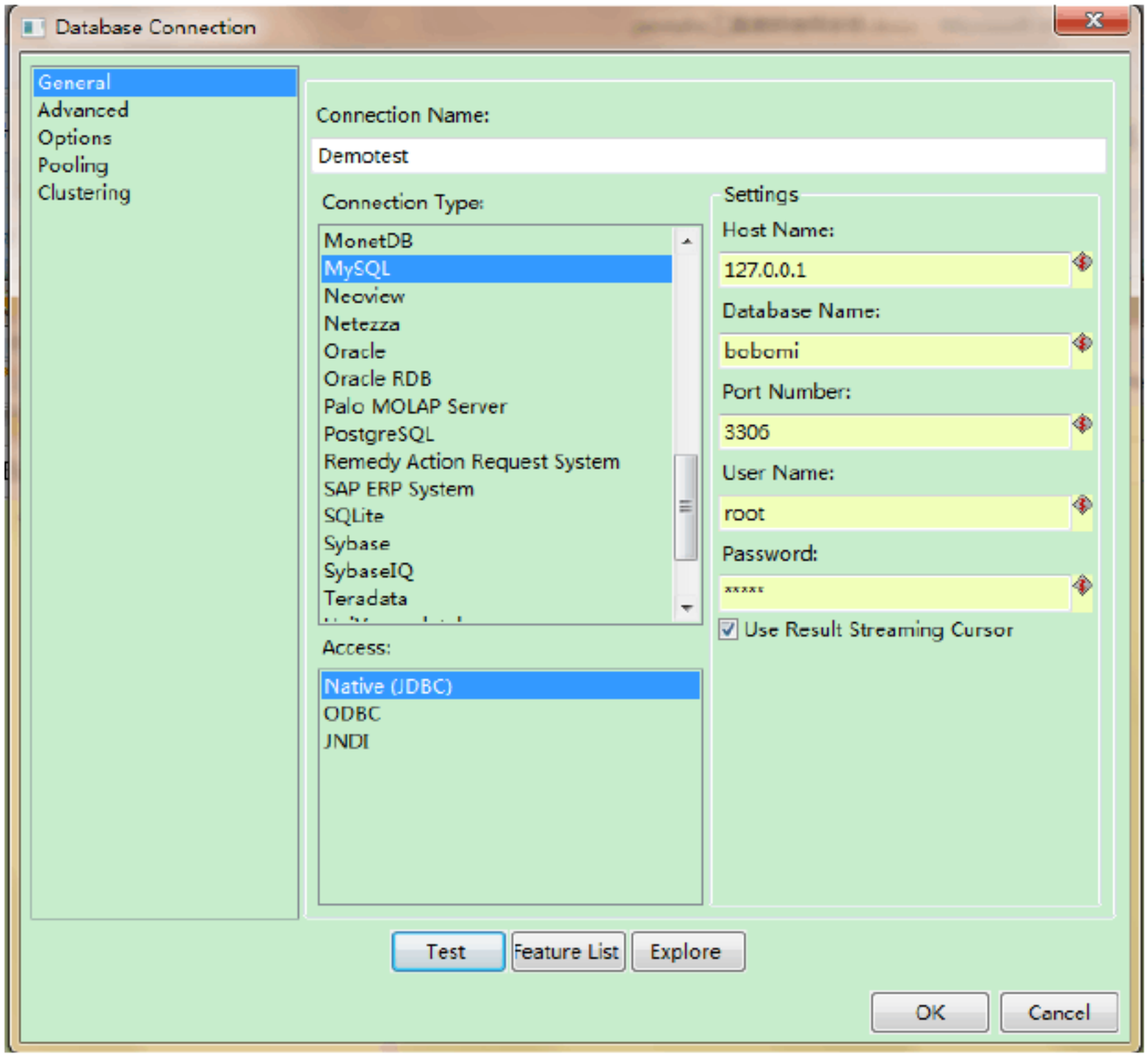
Kettle 的体系构造：



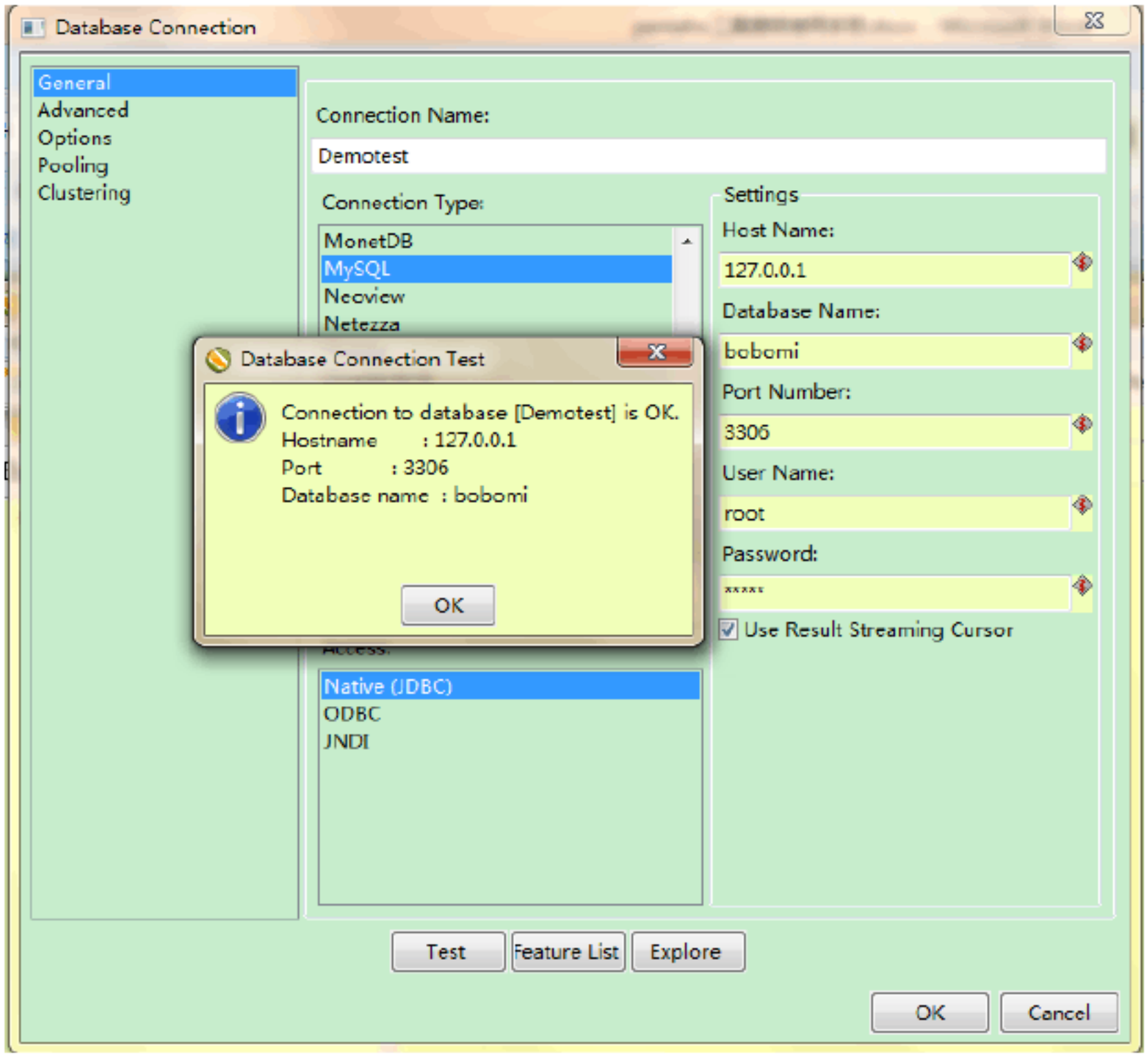
2. Kettle 使用

2.1 数据库连接

使用 kettle 进展数据抽取和转换之前必需连接数据库，你可以同时创立几种不同的数据库连接，如：Oracle、sql server、MySQL 等。以下图是对本地 mysql 数据库建立连接。



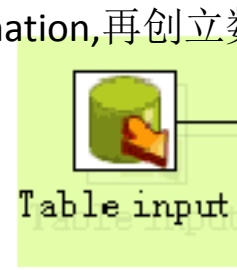
点击 test 按钮进展数据库连接测试



1.1 建一个转换 Transformation(Ctrl+N)

eg 要求：将数据库中交易表的数据按时间增量抽取并过滤输出到目标数据库中的另一张表中。

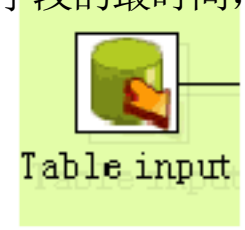
第一步要先创立一个 transformation,再创立数据库连接；



其次步从输入中找到【表输入】，拖到主窗口释放鼠标。接下来双击表输入写

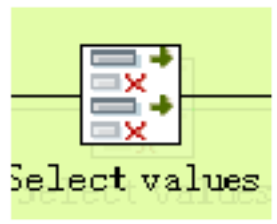
```
SELECT
    case when max( 商品交易时间) is null then makedate(2000,1) else max(日期) end
FROM bobomi_test
```

查询语句；【猎取目标表中对应字段的最时间，没有就给个初始时间。】



第三步从输入中找到【表输入】，拖到主窗口释放鼠标。按住 shift 键，用鼠标点中其次步的表输入与第三步的表输入进展连接；在sql 中从点击猎取 sql 查询语句中选择需要进展增量操作的表，然后确认需要显示列名，消灭没有where 条件的 sql 语句，然后自

已在 sql 中增加 where 条件用？代表从上一个步骤中传过来的变量，在下面替换sql 语句里的变量，打勾，确保到时问号符号能用被替换，从步骤插入数据的下拉菜单中选择上一步操作，在执行每一行上打勾。

第四步从输入中找到【字段选择】 

，拖到主窗口释放鼠标。

按住 shift 键，用鼠标点中表输入与字段选择进展连接；

第五步双击【字段选择】，可在依据需求选择保存字段；

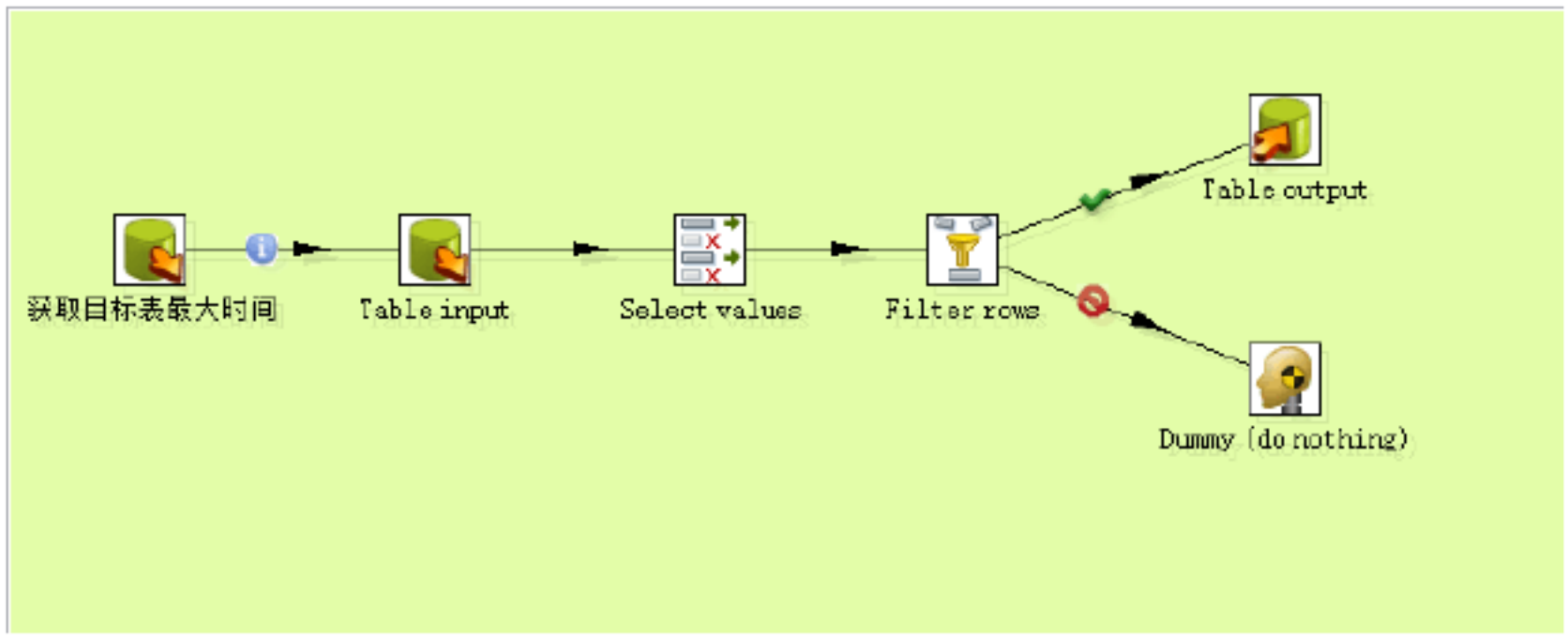
第六步是找到【过滤记录】 

连接方法同上，双击过滤记录，在<field>里面进展字段选择，并在<value>中键入值。确定保存。

最终找到【表输出】 

并对其连接，设置好输出表名等信息后。点击保存，然后点击运行可进展转换。

整个构造图如下：

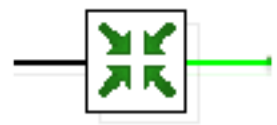


2.2 建一个作业 Jobs(Ctrl+ALT+N)

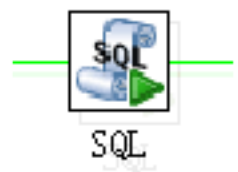
eg 要求：将多个转换依据处理挨次保存到执行打算中。并可以对其进展定时执行。

第一步要先创立一个 transformation,再创立数据库连接；

其次步从通用组件中找到【开头】 ，拖到主窗口释放鼠标。接下来双击可以进展定时设置.

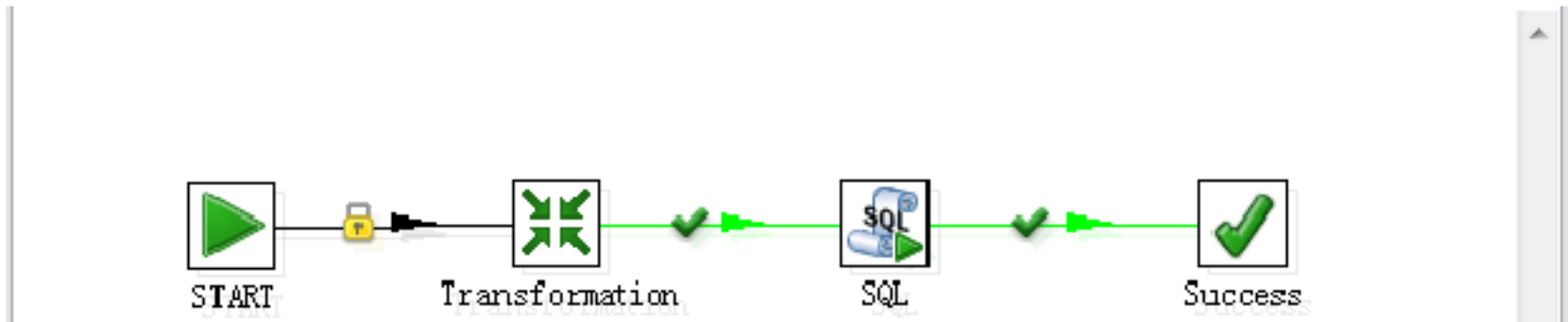


第三步从通用组件中找到【转换】 ，拖到主窗口释放鼠标，按住 shift 键，用鼠标点中开头与转换进展连接；



最终从脚本中找到【SQL】 并对其连接，设置执行的 SQL 后。点击保存，然后点击运行可依据 start 定义的方式进展执行。

整个构造图如下：



由于组件种类繁多，关于 Design 中其他组件的使用方法可以参考附件中的官方文档 [**Pentaho_Data_Integration_4_Cookbook.pdf]**

Pentaho Report Designer

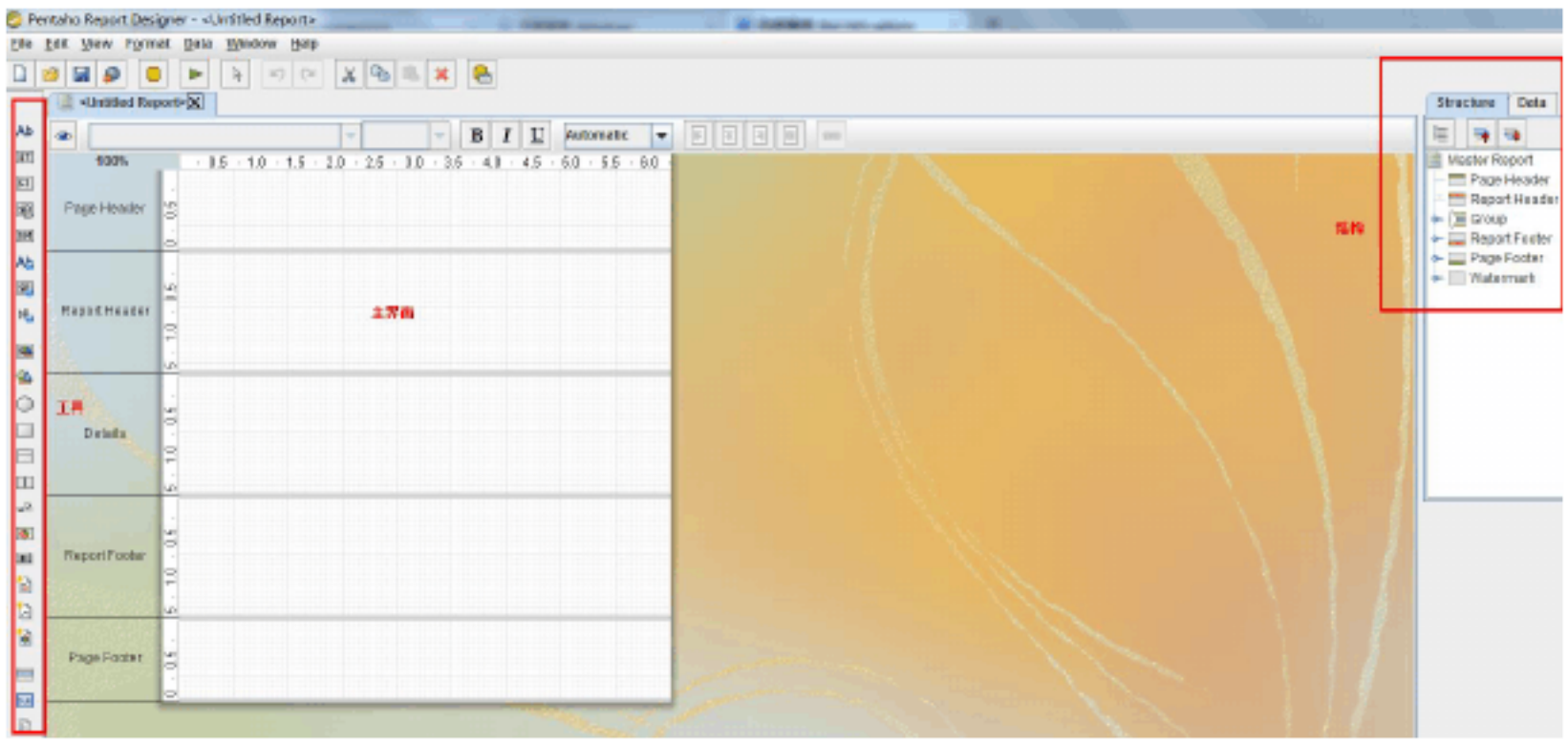
1. Pentaho Report Designer 安装

开发者可通过 PRD，我们将下载的prd-ce-3.9.1-GA.zip 解压到想要放置的路径，并执行这一名目中的 report-designer.bat 批处理文件，PRD 的主界面将呈现在我们面前。

2. PRD 使用例子

2.1 建一个报表(Ctrl+N)

对报表界面做简要介绍：



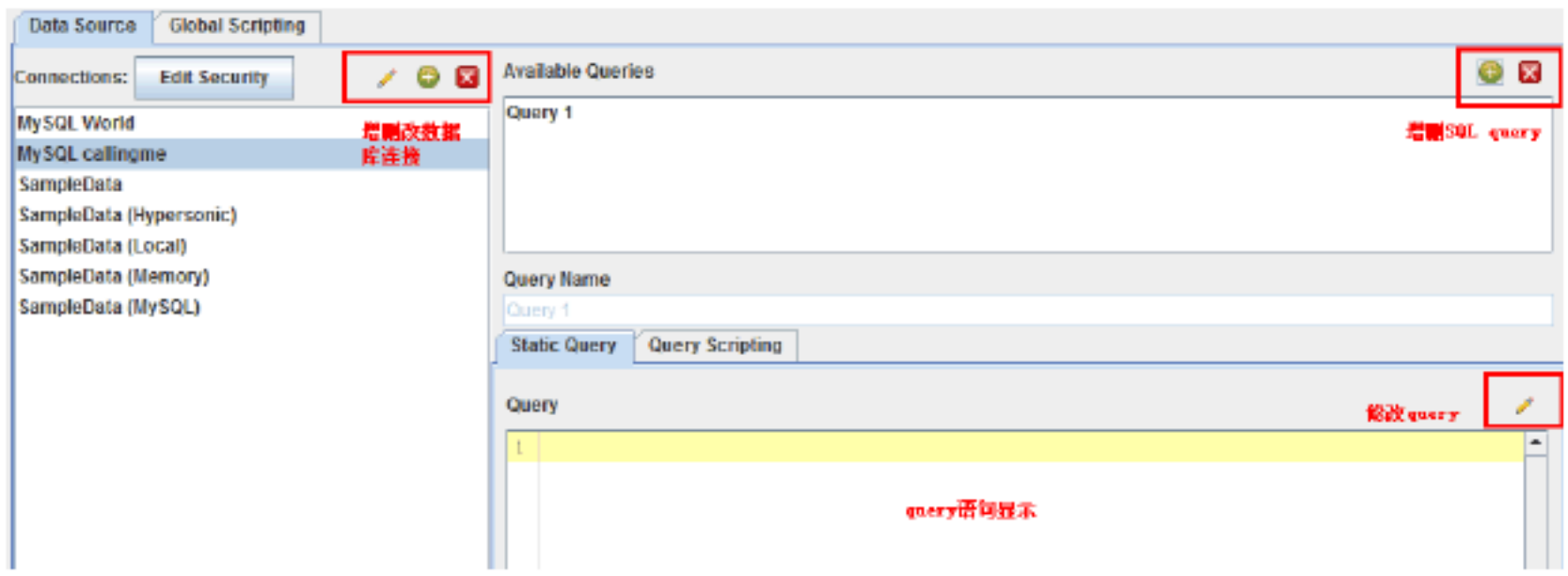
左侧的竖条展现了我们在设计报表时可能用到的工具。

中间的局部是报表的主界面，我们需要把报表结果在主界面上排版展现

右边的标签Structure可以看到报表的构造，Data标签里有全部要展现的数据。包括报表query的结果，以及各种函数。

2.2 创立 query

在 Data 标签下右击 data sets，选择 JDBC 连接，以下界面将会跳出。



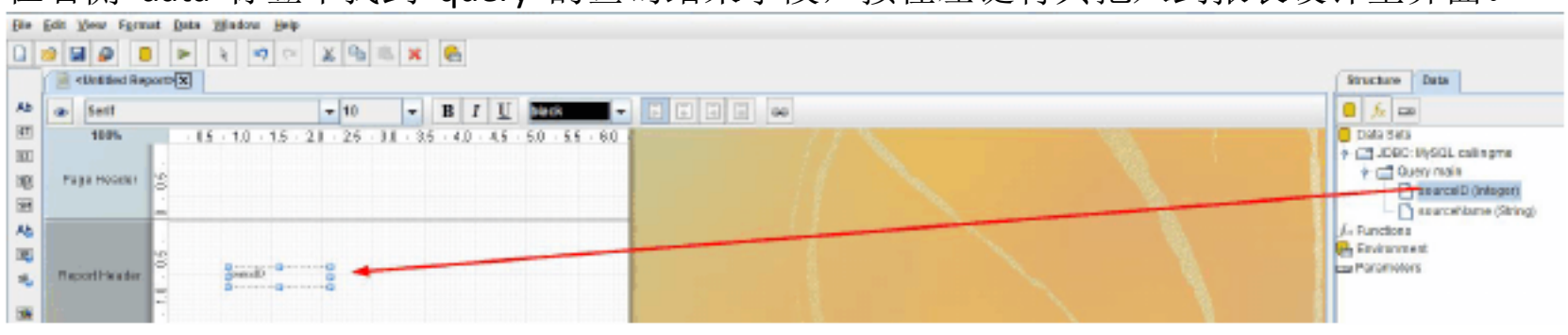
左侧的框格显示了全部已存在的数据库连接，我们可以点击框格上方的按钮来增删改数据库连接。

右上侧的框格展现了我们对应于某个数据库连接有哪些已存在的query。同样可以通过右上角的按钮来增删 query。

右下侧的框格是 query 的主题局部，可以点击铅笔图标进入图形化SQL 编辑器，也可以直接在显示的 query 语句中编辑 SQL 语句。

2.3 设计 query 字段的展现。

在右侧 data 标签中找到 query 的查询结果字段，按住左键将其拖入到报表设计主界面。



切换右上角的标签页到 Structure 标签，单击报表设计主界面上的对象，在右下角的Style 和