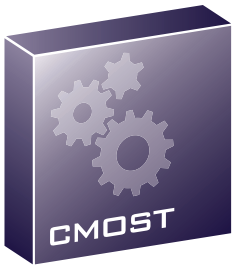
COMPUTER MODELLING GROUP LTD.



CMOST 用户手册

加强 & 促进
敏感性分析、历史拟合、
方案优化及不确定性分析

2015版　　李罡译

15. CM.M3

目 录

1 CMOST新功能（What' New in CMOST?）

1.1 CMOST 2015新功能

CMOST 2015版与之前版本差异总结如下：

1.1.1 CMOST 启动界面（Start Page）

打开CMOST时，会出现启动界面。它用来帮助创建或打开Project（项目）。为了方便起见，最近打开Project的详细内容也显著的呈现出来。通常，启动界面是默认打开的，然而，也可以将其隐藏。更多信息参考 Opening and Navigating CMOST。

1.1.2 在CMM文本编辑器中删除多个参数

可以更容易地删除单个或多个参数，具体例子参考To Delete Selected CMOSTParameters。

1.1.3 为OPAAT图输出数据

以表格的形式输出OPAAT数据到Excel 或Windows 粘贴板，更多信息参考 Exporting Data for OPAAT Analysis。

1.1.4 在Excel中打开输出的时间序列数据

可以在Excel电子表格中输出时间序列数据，详见Opening Exported Time Series Data in an Excel Spreadsheet。

1.1.5 利用时间序列数据在Excel中计算高级目标函数

时间序列数据可以写入Excel电子表格来计算高级目标函数，详见To use time-series data to calculate an advanced objective function第6步。

1.1.6 另存Project

可以另存当前的Project为一个新Project，并且指定一个新名字和目标文件夹。更多信息参考 Saving a CMOST Project as a New Project 。

1.1.7 参数界面文件选择按钮

在参数界面添加了一个文件选择按钮，对于离散文本参数可以方便地浏览并选择Inc文件，详细说明，请参考Discrete Text 。

1.1.8 选择Builder/Results Graph/Results 3D版本

如果在计算机上安装了多个版本的**Builder**、**Results Graph**或**Results 3D** ，可以通过CMOST菜单栏 **Tools | Builder, Graph and 3D Versions** 选择想要的版本。更多信息参考Builder, Graph and 3D Version Preferences。

1.1.9 将代理模型输出到Excel

可以将代理模型输出到Excel，然后利用 Excel，研究参数与目标函数之间的关系，更多信息，参考 Exporting the Proxy Model to Excel。

1.1.10 Sobol 和Morris方法

对敏感性分析，CMOST提供了Sobol和Morris方法，关于这两种方法的理论信息，请参考Sobol Method 和Morris Method。针对这两种方法产生的结果，可以参考Sobol Analysis 和 Morris Analysis。关于如何配置这两种方法，可参考Plotting Preferences。

1.2 CMOST 2014.10新功能

CMOST 2014.10版与之前版本差异总结如下:

1.2.1 帕累托前沿粒子群优化

将多个有冲突的参数作为优化的目标函数是一种挑战。为了处理好这类情况，现在CMOST 支持使用帕累托前沿粒子群优化。更多理论信息，参考Pareto Front Particle Swarm Optimization，配置信息参考 Particle Swarm Optimization (PSO) [Engine-Specific Settings]，详细信息参考 Pareto Front。

1.2.2 代理仪表盘功能加强

CMOST代理仪表盘功能增强，响应更加迅速，更易使用。优点如下:

- 响应更快(速度增加100倍).
- 普通克里金和RBF合并
- 比较RBF和多项式代理模型的能力
- 打开或关闭 QC假设场景曲线的能力。

- 通过代理模型预测目标函数结果。

更多信息参考 Proxy Dashboard 和Radial Basis Function (RBF) Neural Network 。

1.2.3 支持使用Python语言编辑公式

除了JScript语言，CMOST 2014支持使用Python语言，它是一种广泛使用的脚本语言，根据参数、约束条件和目标函数编写公式。现在，用户可以在CMOST中使用Python 或 JScript语言来编写公式。在 General Properties 中**Advanced Settings**选择使用Python作为编码语言，关于使用Python的更多信息参考 Using Python in CMOST。

1.2.4 交互数据分析

CMOST 2014支持交互数据分析，利用 scatter plot、scatter matrix、histogram和parallel coordinates，更好地查看n维数据，帮助工程师更好地理解CMOST结果，更多信息参考 CMOST Interactive Data Visualization Tool 。

1.2.5 合并时间序列曲线

CMOST 2014支持时间曲线合并，主要是用于快速显示时间曲线。为了实现该功能，CMG研发了高效的算法减少曲线和数据点的显示。主要分两步，第一步，当数据从 CMG SR2文件读入时应用筛选算法减少每条曲线上的数据点 。第二步，应用曲线合并算法将图上的曲线进行合并。

1.2.6 自动优化数据体I/O部分

CMOST自动优化数据体I/O部分来减少模拟器输出文件大小，使得CMOST运算更快，也更加稳定。当使用多个 CPU和内核进行CMOST运算时 ，由于I/O障碍，CMOST执行能力可能会受到限制，因为I/O请求不能足够快的处理，所以系统的运算能力就不会增加。为了减少该障碍对 CMOST执行能力的影响，我们研发了一个新功能，CMOST可以自动优化I/O部分来简化模拟输出结果（SR2, OUT, and Restart）的大小。考虑到使用动态编码执行功能需要使用网格数据，我们添加了一个独立的开关，用户可以微调OUT和SR2文件中重启动记录和网格记录。更多信息参考 Simulator Settings。输出模拟结果文件较小的另外一个优点是CMOST、Launcher和所有系统更加可靠，因为他们很少有机会遇到间歇性I/O错误。

1.2.7 曲线定制

用户可根据自己的习惯定制两种标准的图表。第一种是对所有的 CMOST Project 和图件通用的标准。更多信息参考Plot Preferences。第二种，可应用于个别图件，更多信息参考 Plot Settings。该标准对某些需要特别定制图表的油藏工程师非常有用，可根据自己的喜好编辑图表。

1.2.8 诊断Zip 文件

典型的CMOST文件运行后生成许多不同类型的文件。对于故障诊断来说，一个具有挑战性和耗时性的工作就是收集所有相关的数据文件。为了简化该过程，CMG研发了诊断数据的工具来自动收集需要的数据文件。该工具分为两步。第一步，当一个Project被保存时，CMOST自动生成一个XML文件（PDF文件），它包含了一系列Study和数据文件的信息。该步骤在私下完成。第二步，当用户联系CMG需求故障诊断帮助时，它们可以开启诊断数据搜集工具来搜集相关的数据。在该步，也可以选择需要诊断的一些Study。该工具将浏览Project/Study目录结构来查找诊断需要的文件。最后，该工具将所需文件压缩成一个zip文件，然后通过邮件或FTP发给CMG技术支持团队。更多信息，请参考 Using the CMG Diagnostic Tool。

1.2.9 帮助功能

通过按键 F1或工具栏中 来查找帮助信息，更多信息参考 Using the Online Help。对于交互数据可视化工具，其帮助信息可以通过菜单栏帮助按钮得到。

1.3 CMOST 2013.12新功能

CMOST 2013.12版与之前版本差异总结如下：

1.3.1 差分演化（DE）优化算法 (2013.12)

在历史拟合和优化任务中，引进了差分演化(DE)优化算法作为优化最大或最小目标函数的引擎。

DE是一种随机优化技术，由 Storn 和 Price (1995)研发，更多细节详见 Differential Evolution 。

1.3.2 复制参数数据到其他参数 (2013.12)

该功能允许用户从某个特定来源类型的参数复制数据到其他同样来源类型的参数，如下所示：

从.....复制数据	复制数据到.....	复制数据
连续实数	连续实数	数值范围设置、离散取样和先验概率设置
离散实数	离散实数	实数和先验概率
离散整数	离散整数	整数和先验概率
离散文本	离散文本	文本数值、数值和先验概率
公式	公式	JScript编码

更多信息参考Copying Parameter Data 。

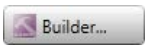
1.3.3 解决多个实验方案Reuse Pending(2013.12)

在 先前的版本，如果添加了新参数，需要为每个实验方案提供参数值来解决“reuse pending”实验方案。该版本发布后，只需提供一次参数数值 ，解决多个实验方案。更多信息，参考Experiments Table Columns 中**Status | Reuse Pending**。

1.3.4 输入部分节点（2013.12）

描述信息已经添加至输入部分节点：基础数据、参数化和目标函数。用户可以在这些页面子节点找到相关的解释。更多信息参考CMOST User Interface。

1.3.5 利用Builder编辑主文件 (2013.12)

Parameters界面 **Builder** 

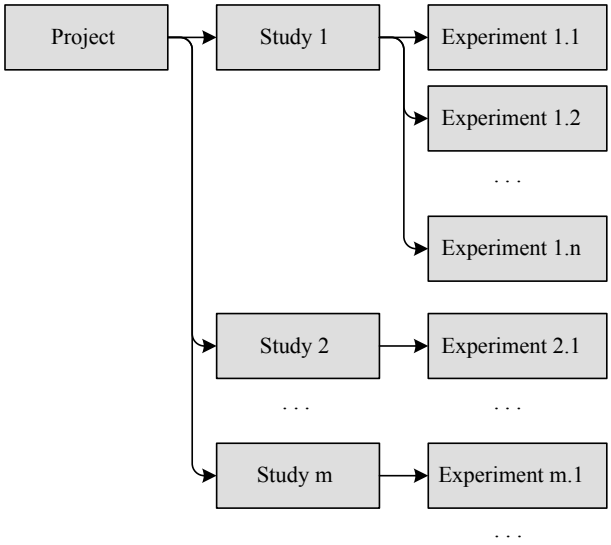
按钮可用于打开主文件（.cmm）。

1.4 CMOST 2013.11新功能

CMOST 2013.11版与之前版本差异总结如下:

1.4.1 基本概念(2013.11)

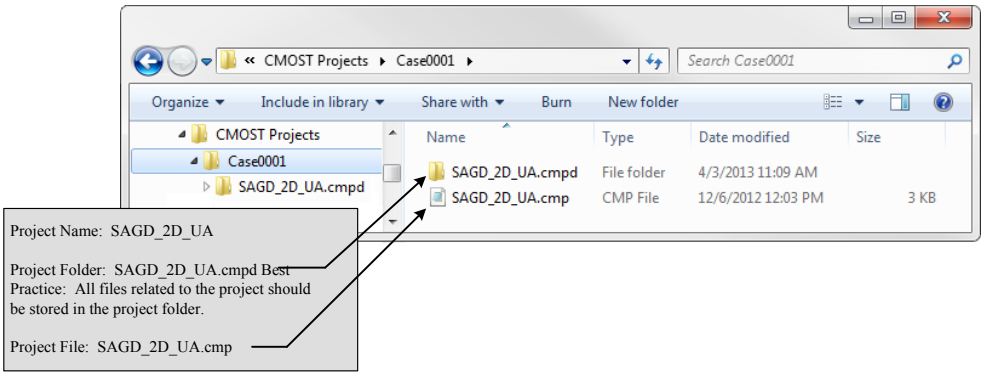
每个 CMOST 2013.11 Project可能包含多个Study，例如可能有敏感性分析、历史拟合、方案优化、不确定性分析及用户自定义Study，而每个Study都有自己的引擎设置。每个 Study都包含CMOST执行某个任务所需要的全部信息，并且这些信息在各个Study之间是可以相互复制的。 Study类型也可以随意转换。每个新的Study要尽可能使用先前Study信息。 Study由系列实验方案组成，每个实验方案都有一套独特的参数取值和目标函数。实验方案详细内容存储在 **Experiment**表格，详见Experiments Table。



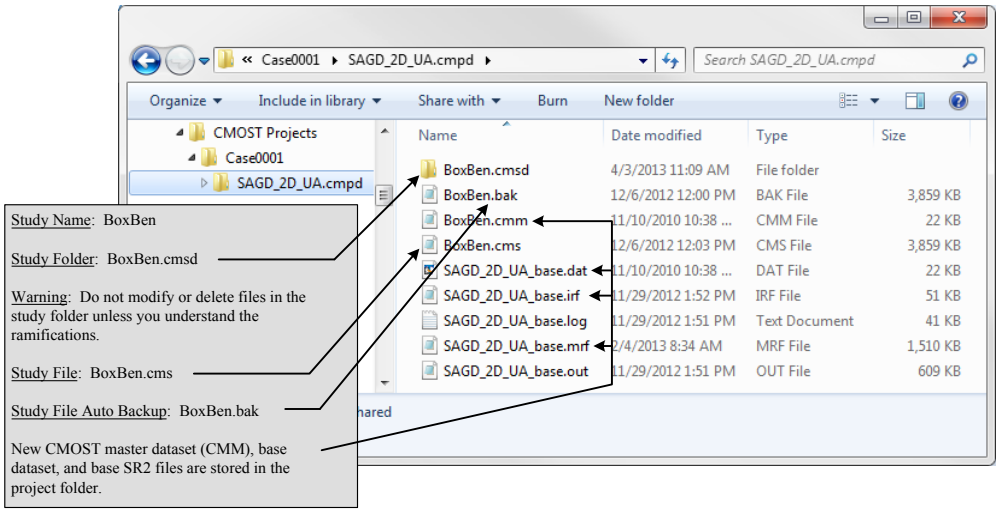
注意：用户可以灵活的定义project和study名称。

1.4.1.1 文件系统和文件夹构成(2013.11)

最高级别，CMOST Project 文件夹如下所示：

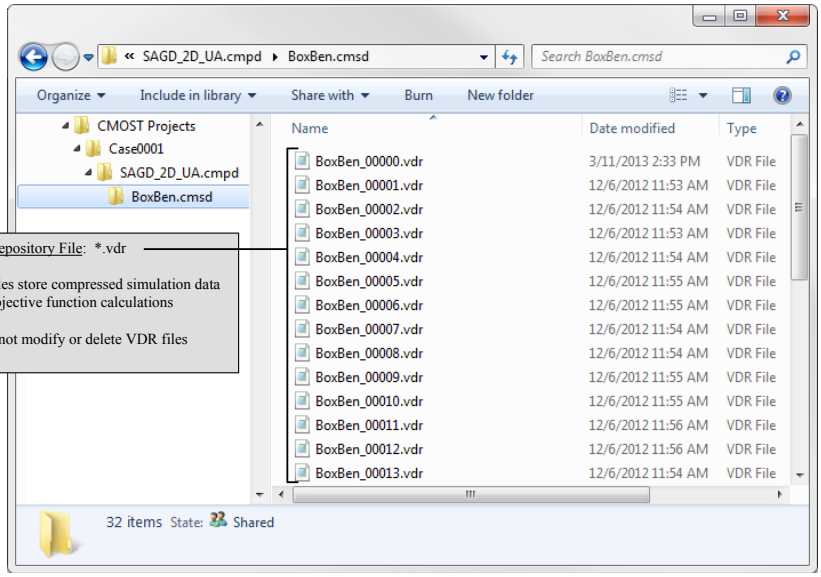


Project文件夹中的文件如下所示：



如果运行中出现错误，CMOST将试着保存后缀为 .bak的Study文件。该文件是最后的有效文件。

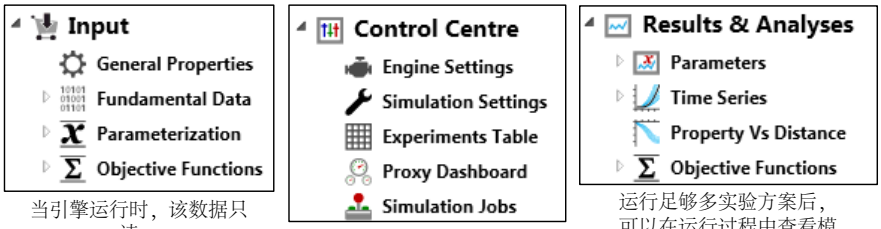
Study文件夹中的文件如下例所示：



VDR文件是压缩的模拟结果数据，用于计算目标函数。文件压缩后会减少磁盘空间和运行时间。

1.4.1.2 Study数据模型(2013.11)

CMOST是由不同等级的界面组成，每个界面上都有相关的信息。通过树状结构存取相关信息，如下所示：



当引擎运行时，该数据只读。

运行时可以改变部分数据，例如添加实验方案。

运行足够多实验方案后，可以在运行过程中查看模拟结果。

1.4.2 用户界面（2013.11）

CMOST 2013用户界面和之前的版本差异较大。

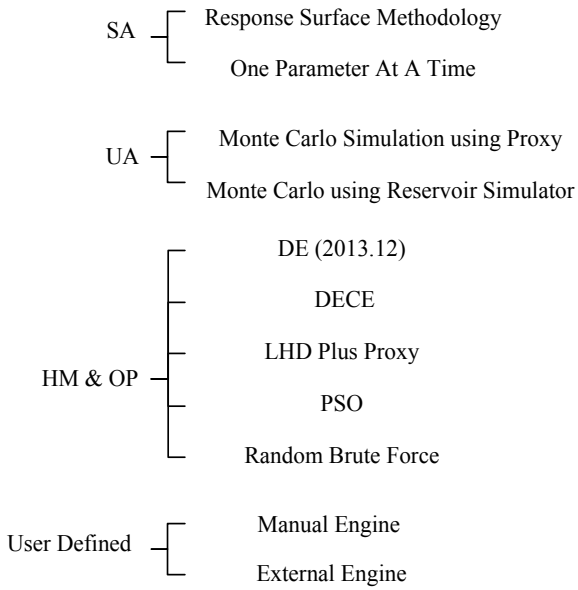
主界面是**Study Manager**标签，通过该界面可以创建、添加、加载、卸载、排除、导入以及复制Study。更多信息，参考Using the Study Manager。

除**Study Manager**之外，CMOST Project界面包含若干个Study，依据选择不同类型的节点，都有一个树状查看结构，包括其配置、状态以及结果界面。 树状节点有指示错误和警告的部分。更多信息参考Getting Started。

如上所述，如果确定了Study-setting 错误或问题，在相应的节点就会标记，这些错误或问题就会在Study标签底部的 **Validation** 用彩色编码，更多信息参考Validation tab。

1.4.3 Study 类型和引擎 (2013.11)

Study类型及可用引擎CMOST 2013.11如下所示:



通过**New Study** 对话框或**Engine Settings**界面指定Study类型及引擎。

1.4.3.1 所有引擎可用特性 (2013.11)

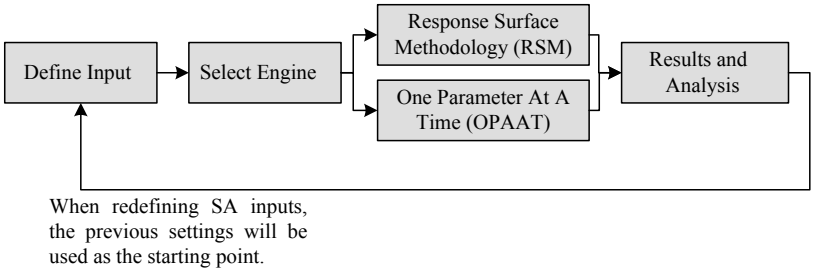
下面**Experiments Management** 设置可应用与所有引擎:

- 排除实验方案时运行任务失败数目;
- 每个异常实验方案允许的扰乱实验数目。这些实验方案会出现在*Perturbed experiment*的 **Experiments**。

使用任何引擎 (SA、HM、OP、UA以及用户自定义），在同一Study中可以使用连续和离散参数。

1.4.3.2 新敏感性分析（SA）工作流程（2013.11）

SA工作流程进行了修改，如下所示：



新SA工作流程优点如下所示：

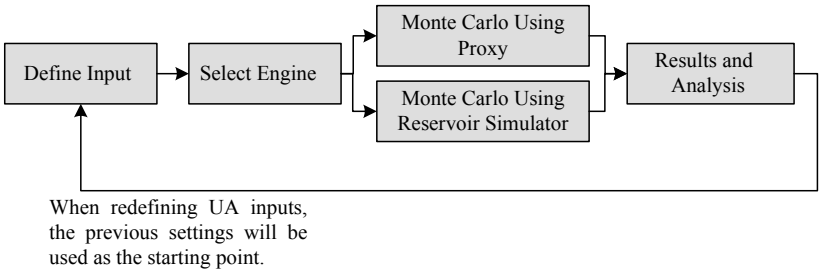
- 执行步骤顺序灵活。
- 可以添加参数及目标函数，并重新运算 SA Study。
- 合理并积极处理有问题的任务，得到可信赖的结果。

使用RSM引擎，可以指定：

- 要求精度，根据创建的引擎及运算的额必要。

1.4.3.3 新不确定性评价（UA）工作流程（2013.11）

UA工作流程进行了修改，如下所示：



新UA工作流程优点如下所示：

- 执行步骤顺序灵活。
- 可以添加参数及目标函数，并重新运算UA Study。
- 合理并积极处理有问题的任务，得到可信赖的结果。
- 可以定义参数关系式。某些参数，根据其物理属性，可能与其他参数相关。例如，渗透率和孔隙度，可以通过实验测得的数据得到孔渗相关关系式，然后通过**Parameter Correlations**界面定义。

CMOST算法调整关系式级别，更多信息参考Parameter Correlation。

当使用MCS-Proxy 引擎执行UA时，可以定义：

- 精度要求，根据引擎创建和运行的必要。

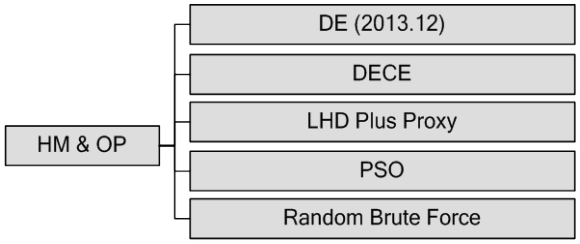
当使用MCS-Simulator引擎执行UA，注意以下几点：

- 引擎执行用户预先定义的蒙特卡洛模拟方案数量。

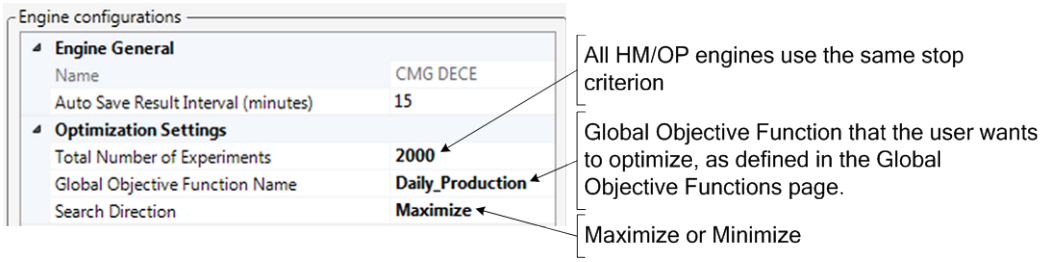
- 使用 MCS-Simulator方法，如果：
  - 需要验证MCS-Proxy结果。
  - 不能创建代理模型，例如，当使用多个地质模型或历史拟合模型。

1.4.3.4 修改HM和 OP算法 (2013.11)

HM和OP支持以下引擎：



HM和OP引擎支持以下优化算法：



DECE优化方法作以下修改：

- 目前DECE版本，如果遇到连续参数，用户可以自定义总实验方案数，以此作为模拟器停止运算的标准。对于离散参数，方参数案总数优先。

代理优化器作以下修改：

- 有能力一起处理连续参数和离散参数。
- 如果遇到连续参数，用户可以自定义总实验方案数，以此作为模拟器停止运算的标准。对于离散参数，方参数案总数优先。

PSO优化器作以下修改：

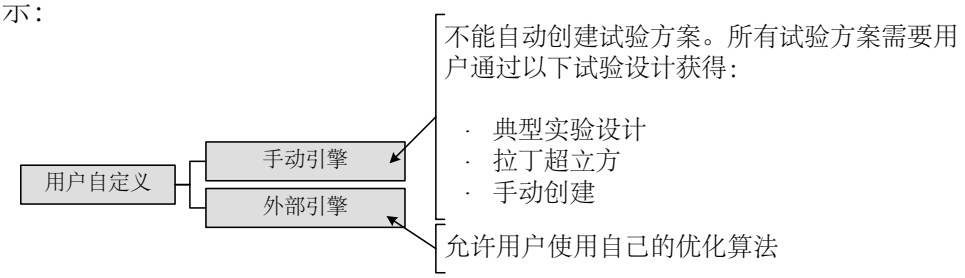
- 有能力一起处理连续参数和离散参数。
- 如果遇到连续参数，用户可以自定义总实验方案数，以此作为模拟器停止运算的标准。对于离散参数，方参数案总数优先。
- PSO可以利用先前所有实验方案的运算结果，帮助PSO快速找到最优方案。

DE优化器(2013.12)：

- 有能力一起处理连续参数和离散参数。
- 如果遇到连续参数，用户可以自定义总实验方案数，以此作为模拟器停止运算的标准。对于离散参数，方参数案总数优先。
- DE 可以利用先前所有实验方案的运算结果，帮助DE快速找到最优方案。

1.4.3.5 用户定义Study类型(2013.11)

CMOST 2013支持用户自定义Study类型，如下所示：



可以使用手动引擎：

- 为SA和UA，使用典型的实验方案设计
- 精确地控制拉丁超立方实验方案总数
- SA/UA/HM/OP运算完成后，运算额外添加的实验方案。

利用外部引擎执行自己的优化算法。更多信息参考External Engine。

1.4.4 创建和编辑输入数据 (2013.11)

1.4.4.1 矿场数据管理(2013.11)

矿场生产历史文件 (FHF)和测井数据导入后，才能被CMOST使用。

一旦导入，数据就被内部存储，定义历史拟合误差目标函数时可以使用。不需要担心包含的文件数据类型。

如果修改了原始FHF或测井文件，需要点击**Reload** 按钮重新加载。在重新加载期间，运行保持其权重。如果需要重新设置权重，需要清除所有的输入数据，然后再加载。

1.4.4.2 修改特殊属性(2013.11)

当原始类型为SPECIALS时，需要属性名称。该修改需要综合（SPECIALS）属性支持。

1.4.4.3 参数定义(2013.11)

连续参数

如果是连续参数，可以定义：

- 参数上下限，设置引擎创建试验方案的抽样范围。
- 离散级别数量，用于产生初始的实验方案总数。
- 参数先验概率分布，仅仅使用蒙特拉罗模拟（代理或模拟器）
- 在先验概率分布和数据范围设置之间自动同步，如果设置为*True*，修改先验概率分布后，会自动反应到参数取值范围，反之亦然。如果设置为*False*，修改之后，不会有影响。

离散参数

如果是离散参数，可以定义：

- 离散参数可以是实数、整数或文本。
- 在参数数值表格中，插入需要的离散数值号码。
- 为每个参数数值输入先验概率（仅仅UA）。
- 对每个离散文本数值，输入相应的数值。

更多信息，参考Parameters。

1.4.4.4 典型时间日期 (2013.11)

利用典型时间日期定义目标函数更容易，也不易出现错误。有三种类型的典型事件日期:

- 固定时间日期，是由基础模型自动输入的，例如模型起始和终止时间。• 确定时间日期，是由用户指定。
- 动态日期时间点，主要是基于数据体或用户定义的时间序列；例如某口井或某个井组的累产油超过一定数值。

更多信息参考Characteristic Date Times。

1.4.4.5 用户定义时间序列 (2013.11)

可以定义时间序列，从SR2文件中计算目标函数。一旦定义时间序列后，可以在图表中以曲线的形式与生产历史数据作对比。更多信息参考User-Defined Time Series 。

1.4.5 管理实验方案 (2013.11)

通过**Control Centre | Experiments Table**管理实验方案。实验设置和状态在**Experiments Table**显示。更多信息参考Experiments Table。

1.4.5.1 实验方案状态和结果状态(2013.11)

引擎启动后，实验方案状态会更新，详见 Experiment Status。

1.4.5.2 筛选实验方案(2013.11)

如果实验方案数量较大，可以使用 **Experiment Filter** 查看或输出感兴趣的实验方案。不会影响表格中的数据。例如，没有被筛选到的实验方案仍会隐藏在试验表格中，更多信息参考Experiment Filter 。

1.4.5.3 基础实验方案(2013.11)

缺省基础实验方案ID=0，是试验表格中的第一个方案。它使用缺省的实验参数值，但可以在任意时间点进行修改。

1.4.6 再利用和重启动(2013.11)

引擎运行结束后，可以回到输入部分重新修改。Study中的实验方案会被重新利用。

如果添加了新参数，实验方案状态改为“reuse pending”。如果在试验表格中有**reuse-pending**实验方案，需要先解决它，然后再运行引擎。更多信息参考 Resolve Reuse Pending。

可以在利用同一Project中其它Study中的数据，更多信息参考To Import Data from a Study。

修改完成后，在**Control Centre**界面点击点击**Start**  按钮，重启动引擎。

1.4.7 代理仪表盘(2013.11)

提供代理仪表盘 Proxy Dashboard，用户可以更加直观的评价生成的代理模型与模拟结果之间的关系，通过代理仪表盘，可以:

- 利用初始的代理模型预测油藏模拟。
- 研究改变参数数值对结果的影响。
- 定义和添加训练和验证实验方案。
- 在不同代理模型之间转换。

1.4.8 查看并分析结果 (2013.11)

计算机运行中动态创建模拟结果，存储在**Experiments Table**。

结果目标改变Study类型，例如，如果运行足够多的实验方案，HM和OP将自动拥有敏感性分析和代理模型分析结果 。

1.4.9 将旧版本CMOST文件转换成新版本CMOST文件（2013.11）

将CMOST任务（CMT）和结果（CMR）文件，转换为CMOST .cmp Project 文件。为了转换CMT文件，注意:

- 基础模型和SR2文件必须存在。如果没有，需要打开CMT文件，选择一个基础文件。
- 使用原始时间序列的目标函数不能转换。
- 公式需要手动核查，确保转换后是正确的。

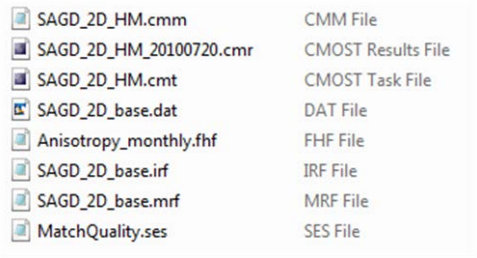
为了转换CMR文件，需要注意以下几点:

- CMOST 将首先查找CMT文件和CMR有相同的文件名称以及同一文件夹。

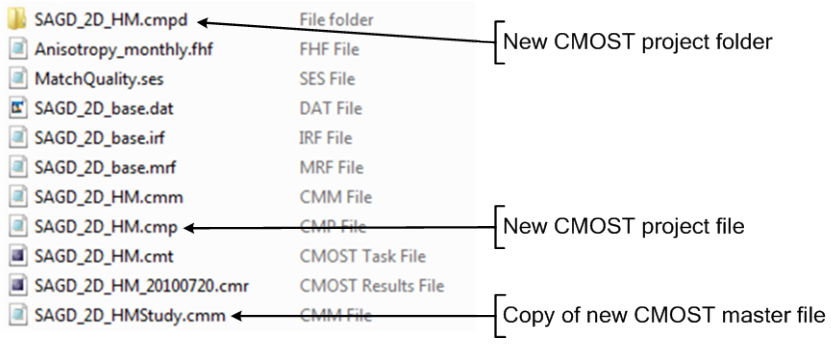
- 如果没有找到CMT文件。CMOST将根据CMR创建CMT文件。将其转到Project文件夹中。
- CMOST将从 CMR文件中导入结果。导入的实验方案用于代理仪表盘创建代理模型。

将旧版本CMOST任务文件转换成新CMOST project 和Study文件步骤如下例所示，下面我们将转换 SAGD_2D_HM.cmt 到新CMOST文件：

1. 转换前CMOST文件如下例所示：



2. 打开CMOST应用，显示CMOST界面。
3. 在菜单栏，选择**File | Convert CMT/CMR File**. A **Windows Explorer** 打开对话框。
4. 浏览并选择CMOST任务（CMT）文件，然后点击**Open**。任务和结果文件转换到新CMOST文件和文件夹，如下例所示：



CMOST Project文件夹包括基础模型的复制文件，文件如下图所示：

5. 用CMOST打开转换的Project文件。在树状结构浏览以下每部分的数据及设置：

   - **Gerenal Properties**：主文件、基础文件、基础session文件、历史文件被复制在Project文件并记录在该界面。不需要修改。

   - **Fundamental Data | Original Time Series**：在我们例子中，在该节点有个错误，因为**SPECIALS**中有个属性需要重新命名。为了消除该错误，在column中选择Stea-Oil ratio:SOR (Injector)/(PRODUCER) CUM。
   - **Parameterization | Parameters**：参数POR、 PERMH、PERMV、HTSORW和HTSORG 已经从就文件中导入过来。不需要修改。

   - **Objective Functions | Characteristic Date Times**：自动生成 *BaseCaseStart* 和BaseCaseStop。不需要修改。

   - **Objective Functions | History Match Quality**：历史拟合误差和原始时间序列项已经从文件中导入过来。不需要修改。

   - **Objective Functions | Global Objective Function Candidates**：总目标函数参考GlobalHmError 已经从文件中导入过来。不需要修改。

   - **Control Centre | Engine Settings**：设置已被导入。不需要任何修改。Settings have been imported from the

   - **Control Centre | Simulation Settings**：有个警告。点击LOCAL Scheduler前面的激活选项，消除警告。

   - **Control Centre | Experiments Table**：没有定义实验方案。在我们的例子中，启动CMOST 引擎后，会生成CMG DECE实验方案。

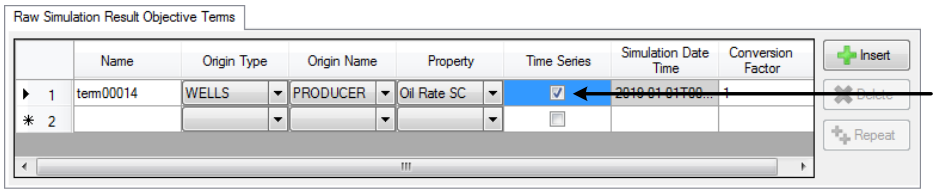
6. 在 **Control Centre**节点，点击**Start** 按钮，开始历史拟合。可以在 **Experiments Table**监测运行过程。 更多信息参考Experiments Table。

7. 运行完成后，通过 **Results & Analysis**节点查看模拟结果，更多信息参考Viewing and Analyzing Results。

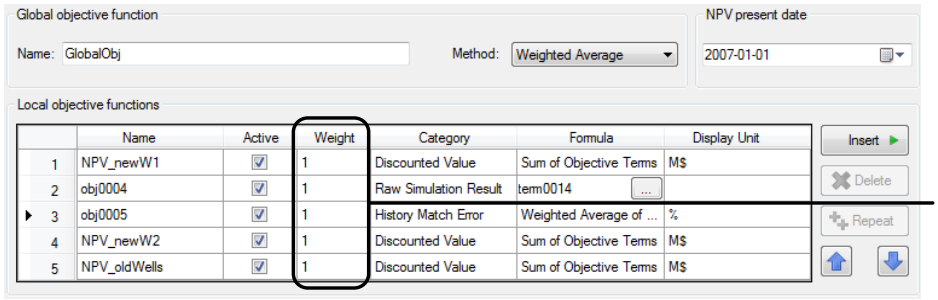
关于CMT转换器的重要信息：

旧版本CMOST任务结果文件中部分信息不能转换：

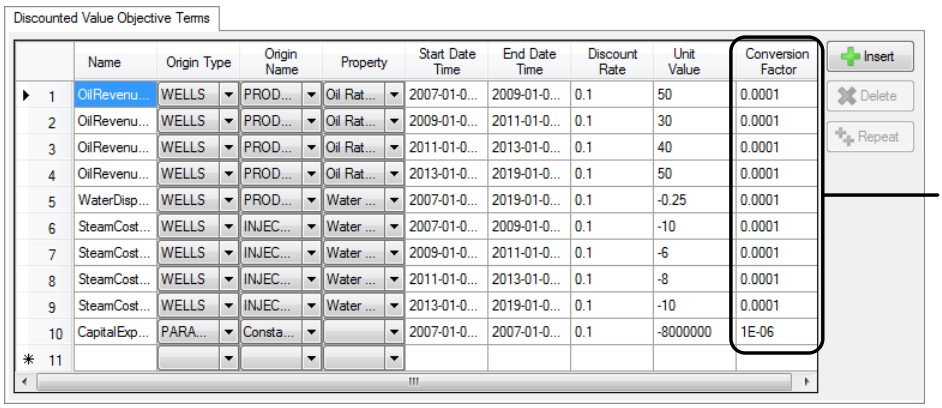
1. 如果目标函数包含时间序列类型为*Raw Simulation Result Objective Term*，考虑使用用户定义的时间序列目标函数。



2. 如果使用多类型局部目标函数计算总目标函数，每个局部目标函数的权重因子不能自动转换，建议检查转换的总目标函数。



3. 如果局部目标函数使用*Conversion Factor*，不需要修订转换目标函数的公式，因为转换因子项不能自动转换。



2 欢迎(Welcome)

2.1 简介

用户手册提供了如何使用CMOST的相关信息，并推荐其他CMG模块和可视化工具。

2.2 使用CMOST需要做什么

2.2.1 通用

为了进一步用好CMOST，需要用户更好的理解CMG油藏模型，尤其是理解想要调整的油藏参数以及调整这些参数对模型的影响。对Project本身而言，也应该有个明确的目标。

2.2.2 配置Launcher和CMOST

CMOST依赖 CMG Launcher或CMG任务服务器来运行任务。在使用CMOST之前，需要配置Launcher和CMG任务服务器。更多信息参考Configuring Launcher and CMOST to Work Together 。

2.2.3 计算机和许可

CMOST可以充分使用电脑和许可。一旦CMOST创建任务，它将自动提交给模拟器运算，定期检查它们的状态。运算完成后，CMOST将自动处理并生成相应的结果。

2.3 关于手册

CMOST用户手册旨在提高老用户的使用速度，并帮助新用户快速学习。对CMOST的熟练程度决定了查阅手册的路线。CMOST关键特征和重要知识点总结如下：

- Important Information for Existing Users 提供的信息用来帮助现有CMOST用户快速了解新版本。新用户忽略该章节。

- CMOST Overview 想要为新用户提供高层次的CMOST概述。

- Getting Started 为用户提供CMOST新版本的指导，尤其是为用户演示实现以下应用：

  - 打开CMOST应用，切换到用户界面。 - 打开已存在的Project，创建新Project。
  - 使用 CMOST Study管理器，创建、查看、重命名、添加、加载/卸载、排除、导入和复制。
  - 使用用户常规界面操作。
  - 关闭应用程序。

  同样地，该章节还提供了CMOST任务处理器的概述以及关于更多信息的链接。

- CMOST用户界面与树状结构图相应，树状结构图中的顺序按照配置、运行及分析结果排列：

  - Creating and Editing Input Data -
  Running and Controlling CMOST -
  Viewing and Analyzing Results

- 常规属性部分内容在General Operations中有描述。

- **Configuring Launcher and CMOST to Work Together**描述了关于CMOST和Launcher协同工作息。的信

- Troubleshooting 部分提供了解决CMOST常规问题的方法。在使用CMOST过程中，如果遇到问题尽量按照该章节提供的方法解决。

- Theoretical Background提供了理论信息。

- 超链接为你提供了快速找到手册相关内容的路径。
- 目录和索引帮助你快速找到需要的信息。

- glossary of terms帮助用户理解CMOST专业术语。

若用户反馈手册错误或提供建议，请联系CMG技术支持，详情见Getting Help。

2.4 使用CMG诊断工具

典型的CMOST运行任务会生成许多不同类型的文件。故障诊断是一个具有挑战性并且耗时的工作，需要搜集有关的所有数据文件。

为了简化该过程，CMG自主研发了一个诊断工具来自动搜集数据文件。在后台，CMOST自动生成XML文件（PDI文件），该文件包含了系列Study和相关数据文件信息。

当用户接触CMG排错帮助时，会被询问提交任务文件。为此：

1. 点击 **Help | Generate Diagnostic**启动诊断数据搜集工具。

2. 通过 **CMG Diagnostic Tool**，可以选择需要诊断的Study。该工具将浏览Project和Study文件夹找到需要的文件。

3. **CMG Diagnostic Tool**将压缩需要的文件，生成一个zip文件。

4. 通过邮件或FTP（如果文件太大的话），将zip文件发送至CMG技术支持团队。

更多信息点击**CMG Diagnostic Tool**对话框中的**Help**按钮。

2.5 使用在线帮助

CMOST用户指南中的技术信息也可以以在线方式得到帮助，具体如下：

- 点击菜单栏Help，然后在线选择CMOST用户指南中索引，查询或内容。索引，查询或内容在左面显示。
- 在对话框的工具栏中点击 或按 F1来打开帮助信息 。
- 从交互数据显示工具，帮助信息可以从菜单栏中帮助按钮得到。

也可以从**Help** 菜单中选择About来打开**About CMOST Studio**对话框，该部分提供了CMOST发布的相关信息，CMG网站链接以及联系CMG技术支持团队的邮箱地址。

2.6 获得其他帮助

关于CMOST应用或手册本身，如果需要额外帮助的话，请登录我们网站 www.cmgl.ca联系CMG 技术支持团队。

3 CMOST 概述(CMOST Overview)

3.1 简介

本章节提供以下内容的详细信息:

- CMOST 定义及用途
- CMOST 处理过程及一般工作流程
- CMOST 输入
- CMOST 概念及组成
- CMOST 用户界面
- CMOST 操作规范建议
- CMOST 算例

该章节的内容是假定用户已非常熟悉CMG软件，而对CMOST不熟悉的情况下介绍的。

3.2 CMOST定义

CMOST 是CMG软件的一个应用，它耦合了CMG油藏模拟器来执行敏感性分析、历史拟合、优化及不确定性评价等任务。

3.2.1 敏感性分析（SA）

敏感性分析用来判断模拟结果在不同油藏属性参数值变化下的影响程度，例如，确定那个参数对定义的目标函数影响最大（例如历史拟合误差）。敏感性分析用有限的参数组合来确定参数在整个变化范围内的影响程度，利用这一结果进一步进行历史拟合或优化，而历史拟合或优化可能需要更多的模拟方案。

3.2.2 历史拟合（HM）

历史拟合提供了有效的方法对模拟结果和生产历史数据进行拟合。CMOST利用基础模型，创建和运行试验方案。模拟任务运行完成后，CMOST分析结果来判断拟合精度。优化器来确定参数新的取值进一步运行模拟任务，当更多的模拟方案运行完成后，会得到一个最优的方案，其拟合精度满足用户要求。

3.2.3 方案优化（OP）

方案优化用来找到最优的开发方案和生产约束条件。方案优化可以优化最大值，也可以优化最小值。这些目标函数可以是物理量，例如累产油、采出程度和累积油汽比等。CMOST也允许将货币价值指定给这些物理量，因此方案优化时也可以使用净现值作为目标函数。

3.2.4 不确定性分析（UA）

不确定性分析用来判定由于残余不确定性导致模拟结果的变化，这些不确定性是由于历史拟合和方案优化之后余留的，由油藏参数的不确定性引起。不确定性分析包括以下：

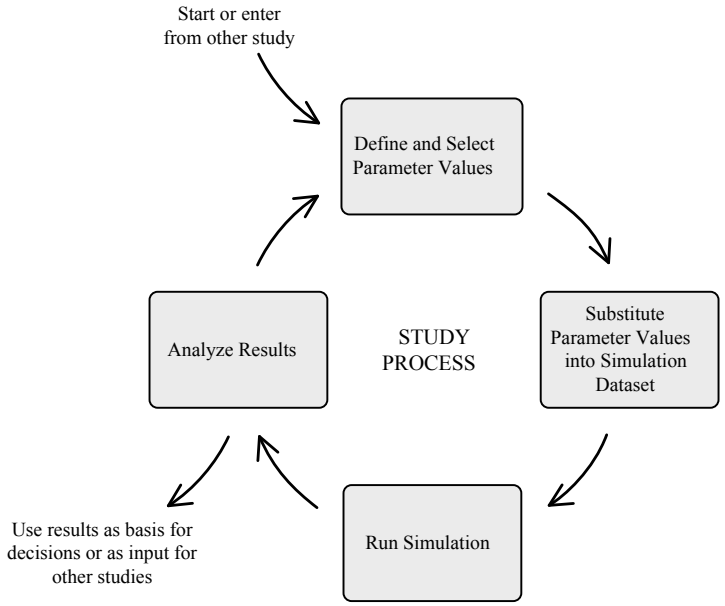
1. 利用有效的模拟结果为每个感兴趣的目标函数（例如NPV、CSOR和累产油）关于不确定性参数（例如孔隙度、渗透率、饱和度端点值和原油粘度）找到一个响应面（RS）。

2. 利用响应面，进行蒙特卡洛模拟，选择大量（成千上万）的变量组合，决定每个组合方案的目标函数值。不确定性分析的结果是每个目标函数的概率和累积概率密度函数。

除蒙特卡洛模拟结果以外，从不确定性Study中也可以得到效果评价和响应面结果，因此可以得到敏感性分析的结果。详细的响应面统计提供了关于使用响应面模型作为油藏代理模型的价值信息。

3.3 CMOST Study 一般处理过程

CMOST study 一般处理过程如下:



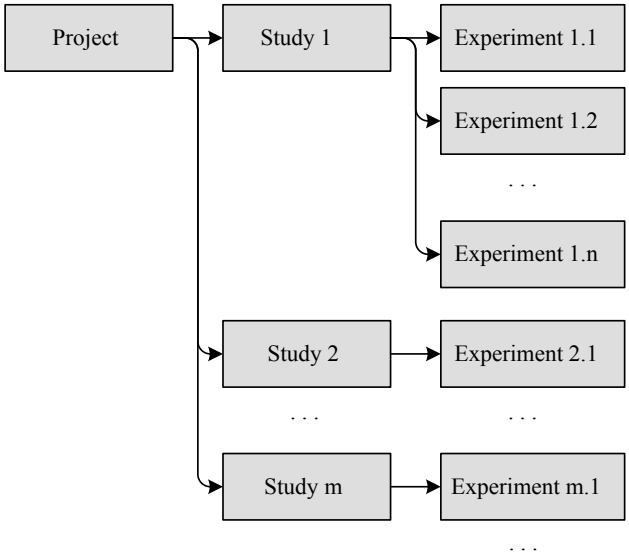
上图展示了CMOST如何来定义并选择参数，并将参数组合应用于Study实验方案，然后通过模拟器来进行运算。如有必要，依据Study类型，分析并使用模拟结果:

- 定义更多的实验方案
- 为后创建的Study提供基础
- 指定生产计划

3.4 CMOST 组成和概念

3.4.1 Project组成CMOST

Project组成等级如下：



CMOST Project包含一系列Study，可以包含敏感性分析、历史拟合、方案优化、不确定性分析以及用户自定义的Study。
通过以下设置定义Study：

- Study类型及输入数据的来源。
- 用于处理输入数据的CMOST 引擎。
- 生成的输出数据类型。

以上设置可以从某一个Study复制到另外一个Study。Study类型可以修改，在这种情况下，新的Study以再次使用先前Study类型的数据信息。

Study由系列实验组成，定义的每个实验包含一套截然不同的系列参数和目标函数。

3.4.2 基础文件

每个CMOST Study都有一个已完成运算的基础模型。CMOST需要从基础模型中读取信息。另外还可能会用到其他文件，例如历史拟合中需要生产历史文件，CMOST用到的基础文件如下所示：

3.4.2.1 基础数据文件

在配置CMOST之前必须有个基础数据文件，然后再运算CMOST实验方案。基础数据文件可以是任一模拟器的模型，用于生成基础SR2（模拟结果）文件。基础数据文件也用于创建 CMOST Master Dataset。

3.4.2.2 基础SR2文件

基础IRF（.irf，索引结果文件）为CMOST提供了基础信息，例如模拟器类型、井列表以及模拟起始和终止日期。基础MRF（.mrf，主要结果文件）是二进制文件，包含了模拟的所有数据。

CMOST可以从SR2文件得到并显示观察目标函数（模拟输出的结果），也能计算基础方案目标函数（用户想要得到的最小或最大的表达式或数值）。根据CMOST约定，基础方案的实验ID为0。

3.4.2.3 基础Ses文件

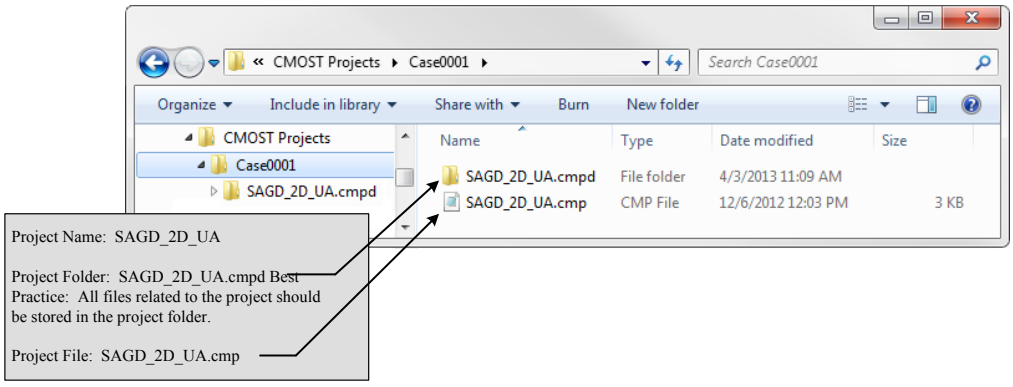
基础Ses文件可用于CMOST，但不是必需的。基础Ses文件是由CMG Results™ Graph 利用基础SR2文件生成的。CMOST使用基础Ses文件可以快速的显示需要的结果曲线。

3.4.2.4 生产历史文件

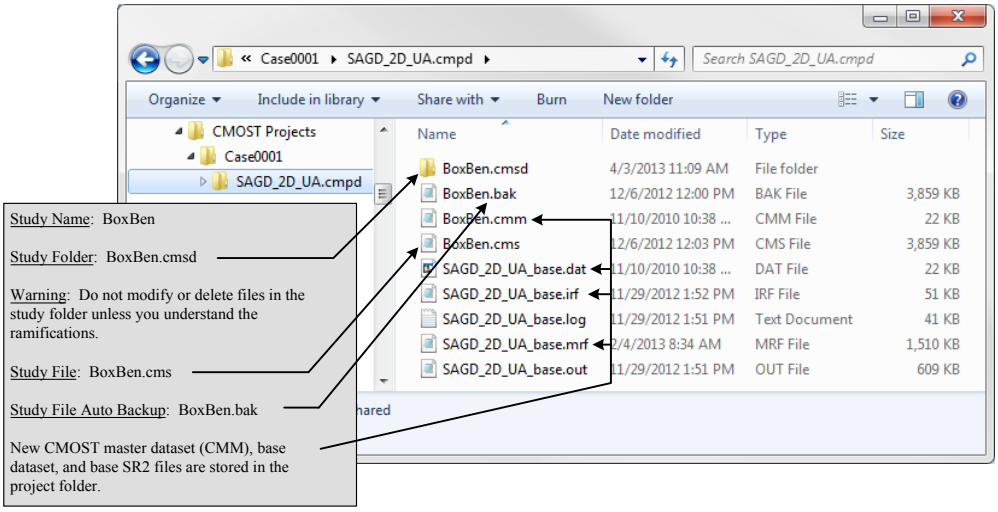
生产历史文件，例如生产历史拟合时，会用到矿场生产历史数据及测井文件数据。需要的文件主要依赖于历史拟合的类型。

3.4.3 文件系统

CMOST最高等级 Project文件夹包含如下例所示的文件，对于Project SAGD_2D_UA：

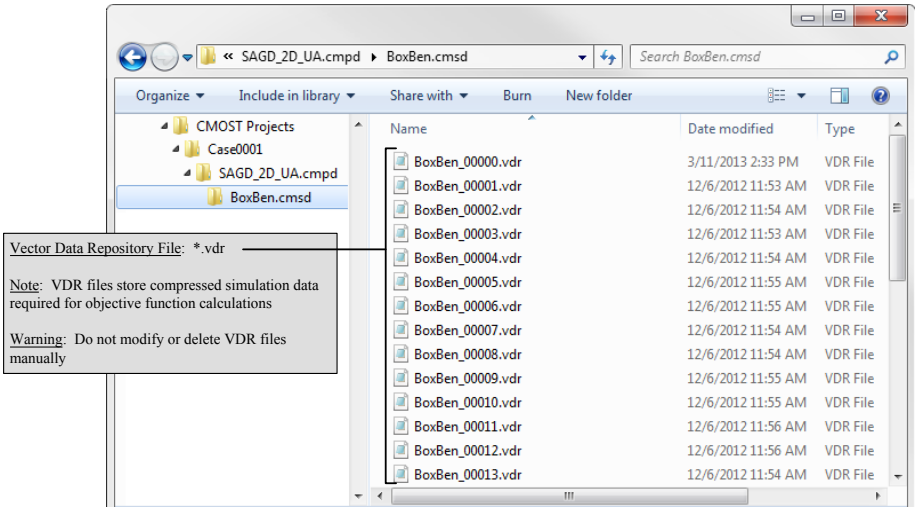


Project文件夹中的文件如下所示：



注意：如果运行时出现错误，CMOST将Study文件保存成后缀为 .bak 文件。.bak 文件是最终的有效文件，其格式和Study一致。

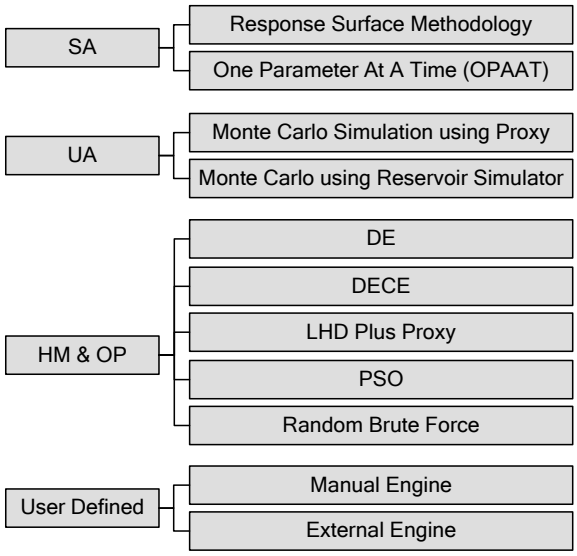
存储于cmsd文件夹中的文件如下所示：



注意：VDR文件是压缩文件，用于存储计算目标函数的模拟数据。文件压缩用来减少占用磁盘空间的大小和运行时间。

3.4.4 Study 类型和引擎

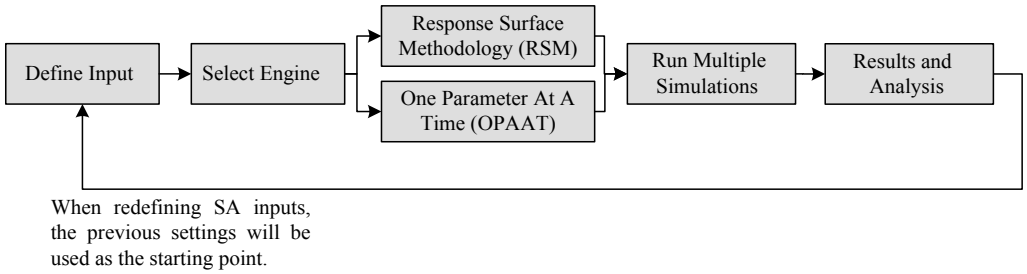
CMOST Study类型和引擎如下所示：



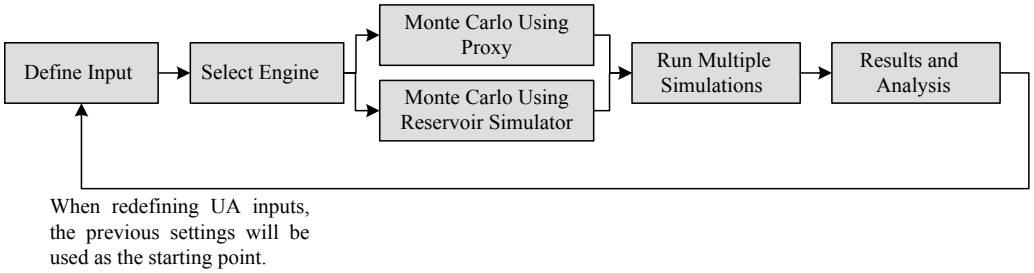
关于引擎的相关信息，可以在Theoretical Background 查看，另外关于配置引擎的相关信息，可以在 Engine Settings查看。

3.4.5 Study 工作流程

Study工作流程依赖于Study类型和选择的引擎而变化；例如，下面是敏感性分析的简化流程：



简化的不确定性分析工作流程：



更多关于一般工作流程的信息，请参考 CMOST User Interface。

3.5 CMOST 主文件（.cmm）

CMOST主文件是必需的文件组成，它是在基础模型文件的基础上，通过嵌入CMOST语句修改而来的，主要用于运行时告诉CMOST提交不同参数值到数据文件。

可以通过以下几种方式创建CMOST主文件：

- CMOST CMM 文本编辑器
- Builder™ *（更多信息参考Builder Users Guide 中 “Setting Up Datasets for CMOST”章节）*
- 文本编辑器，例如记事本

该部分提供了概述和使用CMM文本编辑器插入CMOST参数的例子。CMM File Editor 提供了更多信息和细节来添加CMOST参数。

为了说明嵌入主文件中的CMOST语句，参考下面的例子，我们已经为参数Porosity、PERMH_L1、PERMH_L2 和KvKhRatio嵌入了CMOST语句：

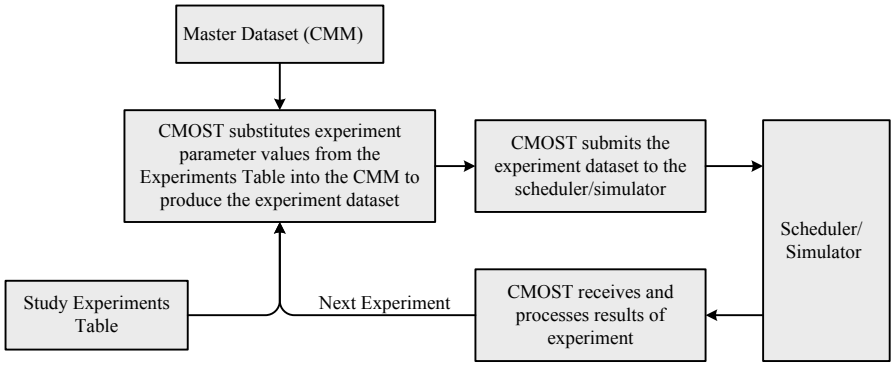
```
WPRN ITER 0
OUTSRF GRID PERMI PRES SG SO SW TEMP VPOROS

**  ==============  GRID AND RESERVOIR DEFINITION  =================

*GRID *RADIAL 13 1 4 *RW 0.3

**  Radial blocks:  small near well;  outer block is large
*DI *IVAR   9*15 20 40 80 100

*DJ *CON 360  **  Full circle

*DK *KVAR 25 25 20 10
**$ Property: NULL Blocks  Max: 1  Min: 1
 **$  0 = null block, 1 = active block
 NULL CON            1

*POR *CON <CMOST>this[0.3]=Porosity</CMOST>
*PERMI *KVAR <CMOST>this[3500]=PERMH_L1</CMOST> <CMOST>this[800]=PERMH_L2</CMOST> <CMOST>this[340]=
PERMJ EQUALSI
PERMK EQUALSI * <CMOST>this[0.3]=KvKhRatio</CMOST>
**$ Property: Pinchout Array  Max: 1  Min: 1
 **$  0 = pinched block, 1 = active block
 PINCHOUTARRAY CON           1

*END-GRID
ROCKTYPE 1
```

CMOST parameters added

CMOST需要提交一个值或文本到主文件，应该输入CMOST语句。CMOST语句可以出现在主文件的任何位置，然而，每个CMOST语句都要在一行内写完。

注意：主文件中的第一个日期/时间关键字必须是*DATE。如果使用*TIME 关键字将会报错。

下图展示了CMOST如何创建并将实验方案提交到模拟器，然后接收并处理结果的整个过程（对单核处理器）：



3.5.1.1 主文件语法

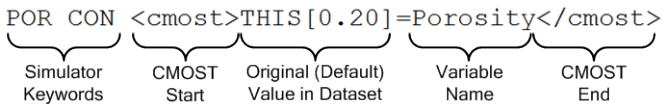
主文件语法如下例所示：

例 1：

原始数据文件中，POR定义如下：

```
POR CON 0.20
```

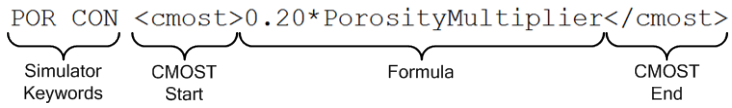
在主文件中，我们想改变孔隙度的数值，因此需要输入CMOST语句，如下所示：



注意：CMOST部分不允许空格存在，参数名称区分大写。

例 2：

公式也可用于一个或多个变量，如下：



上述例子中，模拟运行之前参数 PorosityMultiplier 将乘0.2。

注意：缺省值是可选的，通常等于基础文件中的原始值 。

例 3：

油藏部分中的参数值可以通过关键字MOD 来修改，如下例所示：

```
POR CON 0.20
MOD
1:5 2:8 1:10 * <cmost>this[1]=PorosityMultiplier1</cmost>
6:10 2:8 1:10 * <cmost>this[1]=PorosityMultiplier2</cmost>
```

Block Ranges
I:I J:J K:K

3.5.1.2 CMOST 公式

所有输入到主文件中的公式都需要嵌入一个起始标志<cmost> 和结束标</cmost>。可以选择使用this[OriginalValue]=来开始一个公式。原始值表示基础方案中使用的数据。同样，如果使用this[OriginalValue]= 语法，变量this 可用于公式来引用数据文件中的原始值。如果原始值是个字符串（文本），需使用双引号。例如，CMOST可使用下面的语句准确处理：

```
INCLUDE  '<cmost>this["por50.inc"]=PORINC</cmost>' PERMI CON
<cmost>this[5000]=PERMH</cmost>
```

如果CMOST原始值不应用于上述公式，那么上述公式可简化，省略this[ ]= ，如下：

```
INCLUDE  '<cmost>PORINC</cmost>'
PERMI CON <cmost>PERMH</cmost>
```

CMOST不能处理下面的语句，因为原始文本文件没有使用双引号：

```
INCLUDE  '<cmost>this[por50.inc]=PORINC</cmost>'
```

CMOST公式语法和可用的内置函数描述，详见 Formula Editor。

3.5.1.3 CMOST 公式示例

下面的例子中详细说明了将不同函数公式插入主文件中。

例1：将参数varA的值直接提交到主文件：

```
<cmost>this[1.0]=varA</cmost>
<cmost>varA</cmost>
```

例 2：将+参数varB 之后的值赋给原始值

```
<cmost>this[1]= this + varB</cmost>
```

例3：提交 varA–varB+5的结果：

```
<cmost>this[26]=varA-varB+5</cmost>
<cmost>varA-varB+5</cmost>
```

例 4：提交 $179.79 \times \left(\frac{\text{var A}}{\text{var B}} \right)^{0.248}$ 的值:

```
<cmost>this[203.9]=179.79*POWER(varA/varB, 0.248)</cmost>
```

例 5：如果参数varA 乘以参数 varB 的值大于1200，那么将这两个参数的乘积提交给主文件，否则将1200提交：

```
<cmost>this[1800.0]=MAX(varA*varB, 1200)</cmost>
```

例6：如果参数 varA 大于或等于 600，那么将OPEN提交至主文件；否则提交 CLOSED：

```
<cmost>this["OPEN"]=IF(varA>=600, "OPEN", "CLOSED")</cmost>
```

例 7：如果 varB 与第一组系列的某值相匹配，则对应使用第二组系列中的相应值；例如，如果 varB 使用7.3，那么将188.75提交至主文件：

```
<cmost>this[402.57]=LOOKUP(varB, {3.0, 5.0, 7.3}, {524.62, 402.57, 188.75})</
cmost>
```

例 8：如果 varA 与第一组系列中的某文件相匹配，则对应使用第二组系列中的相应文件，例如varA 使用文件"porMid.inc"，那么需将文件"permMid.inc" 提交至主文件。

```
<cmost>this["permMid.inc"]=LOOKUP(varA, {"porLow.inc",
"porMid.inc", "porHigh.inc"}, {"permLow.inc", "permMid.inc", "permHigh.inc"})</
cmost>
```

例9：为varA设置可接受的值。如果 varA 小于 0，则 0提交至主文件。如果 varA 大于1，则 1被提交至主文件。如果varA 介于0和1之间，则提交 varA 值:

```
<cmost>this[0.68]=MAX(MIN(varA, 1), 0)</cmost>
```

3.5.1.4 将 Include 文件插入主文件

如果需要提交数组数据，可能使用Include文件会比较简单；例如，油藏模型中的每个网格都有一个孔隙度值。为每个网格的孔隙度创建一个参数是不现实的。可以使用多个Include文件，每个文件包含所有网格不同的孔隙度值。

Include文件可以应用于任意主文件位置。主文件中使用Include文件的语法如下所示：

```
*INCLUDE '<cmost>ArrayIncFile</cmost>'
```

参数 ArrayIncFile 通过Parameters 界面定义为一个 Text 参数作为参考值。

Include文件中包含网格文本，将其提交至主文件。例如，油藏模型中的孔隙度可以应用于以下Include文件 ，其尺寸 ni = 10, nj = 3, nk = 2：

```
*POR *ALL
.08  .08  .081 .09  .12  .15  .09  .097 .087 .011 .15  .134 .08  .087 .157 .14
                                5 .12  .135 .18  .092
.074 .12  .12  .154 .167 .187 .121 .122 .08  .08 .095 .13  .12  .157 .17  .1
8  .184 .122 .084 .09 .11  .12  .134 .157 .157 .18  .18  .098 .09  .09 .08  .
09  .144 .143 .123 .16  .165 .102 .10  .10
```

其他Include文件也可以使用相似的语法来创建，但需要输入不同值。

3.5.1.5 参考主文件处理文件

CMOST使用下面的逻辑来处理路径和相关的文件，包括INCLUDE、BINARY_DATA和 FILENAMES INDEX-IN关键字：

如果是绝对路径：

CMOST将不会复制相关的文件到相应的Study文件夹。CMOST不会修改路径； 例如， CMOST将保持下面的语句，不做修改：

FILENAMES INDEX-IN '\\computer8\d\Test\punq-history.irf'

因为它是绝对路径， CMOST将不会复制其参考文件（.irf和.mrf）。

如果路径仅包含文件名（没有目录信息）：

CMOST将复制相关的文件到相应的Studu文件夹。CMOST将不修改路径；例如，CMOST将保持下面的语句，不做修改：

INCLUDE 'PorLow.inc'

CMOST将从原始目录复制文件 ‘PorLow.inc’到Study文件夹。

如果路径是相对路径：

CMOST将不会复制相关文件到相应的Study文件夹。当创建数据模型时，CMOST将以相对路径为依据；例如，CMOST将以下面的相对路径为基础：

INCLUDE 'incfiles\PorLow.inc' 到Study文件夹后，其路径修改为：

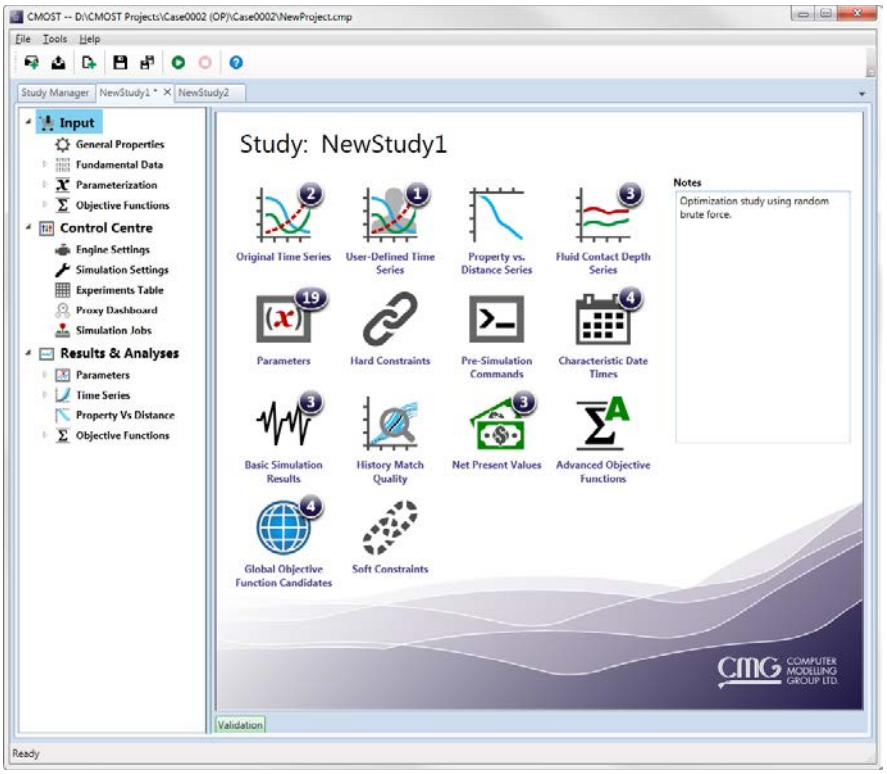
INCLUDE '..\incfiles\PorLow.inc'

因为是相对路径，CMOST将不会复制相应的Include file文件。

注意：即使BINARY_DATA后面没有跟一个模拟器可接受的路径名，CMOST也不允许，因为它需要为每个CMOST创建的模型复制一个二进制文件。 这样创建.cmgbin 文件的目的是用来节省空间。

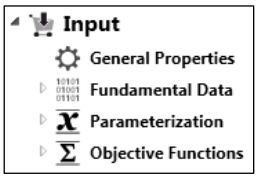
3.6 CMOST 用户界面

CMOST用户界面的例子如下所示：

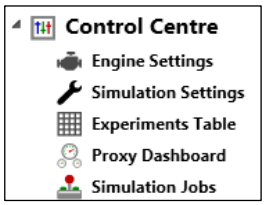


CMOST用户界面的设计使得功能强大、访问复杂的CMOST简单化，也更好理解。常用的操作在菜单栏和工具栏都可以方便的找到。有关CMOST用户界面的基础信息请参考 Getting Started。

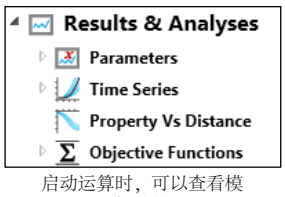
CMOST用户界面通过左边的树状目录按照Study工作流程分层次显示。通过树状节点进行配置及访问结果界面，如下所示：



当引擎启动运算时，该界面只能阅读，不能修改。

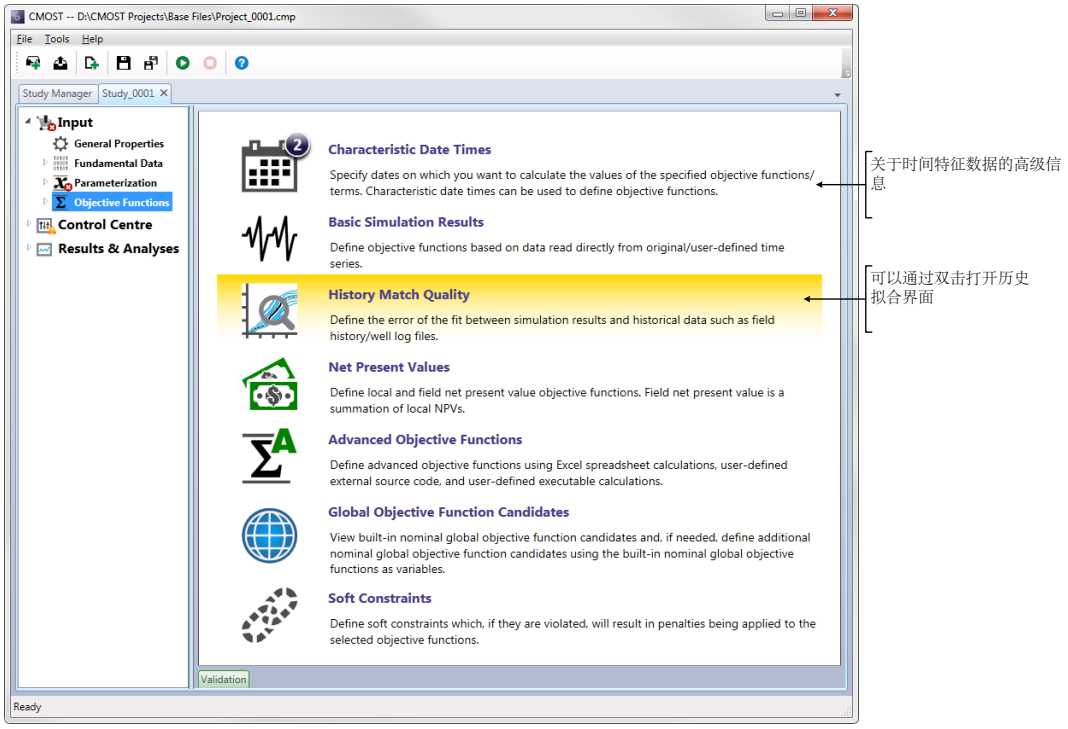


引擎启动运算时，可以修改该部分的某些数据，例如，添加实验方案。



启动运算时，可以查看模拟结果

在**Input**部分，每个主要节点及子节点的信息，如下例所示：



通过**Input**节点，可以：

- 定义文件的输入，包括基础文件、主文件、ses文件和历史数据文件。更多信息参考**General Properties**。
- 定义基础数据（时间、距离或深度数据），这些数据是从SR2文件或计算得到的，可用于敏感性分析、历史拟合或优化。参考Fundamental Data。
- 定义和指定在CMOST Study实验中使用的变量参数，包含嵌入到主文件中的参数。参考Parameterization。
- 定义想要优化最小和最大的目标函数。例如，如果是历史拟合，则要优化历史数据和模拟结果之间的最小误差，如果是方案优化，则想要优化最大的净现值 。参考Refer to Objective Functions。

通过**Control Centre**节点配置，可以：

- 定义和配置Study和引擎类型。参考 Engine Settings.
- 指定和配置模拟设置。参考Simulation Settings.
- 指定实验方案，一旦引擎启动，检测运行进展，如果有必要，则进行调整。参考Experiments Table。

- 启动、停止、暂停和监测CMOST引擎。参考 Control Centre.
- 监控代理模型的状态，如果有必要，对试验方案进行调整。参考Proxy Dashboard。
- 监测模拟任务的过程。参考Simulation Jobs。通过Results & Analyses节点，可以:
- 查看和分析Study结果。一旦引擎启动，就可以动态查看运行结果。根据不同的Study类型，显示不同的结果信息。例如，如果使用One-Parameter-At-A-Time引擎执行敏感性分析时，将生成一张 OPAAT图。更多信息参考 Viewing and Analyzing Results。

3.7 使用CMOST的最优方法

配置主文件I/O 控制部分

配置.cmm 文件**Input/Output Control** 部分，使得模拟输出的结果文件（.irf, .mrf, .rst, .out）尽可能小。模拟结果太大运算比较缓慢，同时也经常引发I/O问题，导致模拟任务运行失败。详细细节，请参考相应手册部分关键字WRST、OUTSRF、WSRF、OUTPRN和WPRN 。通常，不需要在每个DATE/TIME 时间点写重启动。

多个远程任务同时运行

如果想同时运算多个远程任务（≥ 5），CMG推荐使用Windows 2003或2 008文件服务器来存储CMOST 输入/ 输出文件（. cms、 .vdr、模拟输入/ 输出文件）。这是因为对于Windows 操作系统（Windows XP、Windows Vista及 Windows 7）的工作站类型，远程计算机允许的最大任务数量为1 0。该限制包括所有传输和资源共享协议的总和。因此，当多于4 个远程任务同时允许时，CMOST文件服务器对于Windows工作站 的使用可能都是不适合的。更多关于Windows XP、Windows Vista和Windows 7入站连接限制的信息，请查看微软知识基础文献Q314882。

检测可用的磁盘空间

为所有Windows计算节点不定期检测 C盘可用空间。如果 C盘可用空间较小，删除文件夹C:\ProgramData\CMG\CopyLocalJobs (Windows 2008、Windows Vista及Windows 7或C:\Documents and Settings\All Users\Application Data\cmg\CopyLocalJobs (Windows）2003和Windows XP)中不想保留的文件。对于Linux计算节点，检测/tmp文件夹中的可用空间。将不想保留的模拟结果输出文件（.irf、.mrf、.rst及.out）从/tmp文件夹中删除。

使用累产速率数据作为历史拟合的目标函数

速率（例如产油速率、产水速率或产气速率）一般不建议作为历史拟合的目标函数，因为这些参数都是不连续的。阶梯函数的本质意味着速率值每个间隔边界不好确定（不一致）。这种阶梯函数的不一致可能会引起历史拟合误差计算的不精确性。因为累产（累产油、累产水等等）在时间上都是连续的，建议作为历史拟合的目标函数。如果整条曲线拟合的较好，就保证了速率曲线也拟合的较好。同样，这也适用于其他的瞬时变量，例如含水率或瞬时GOR或SOR。

为敏感性分析使用历史拟合误差

敏感性分析时，不推荐使用历史拟合误差作为目标函数，通常，使用物理量，例如累产油、压力或温度作为目标函数。原因有二，第一，历史拟合的目标函数从线性函数转化为非线性函数，第二，油藏模拟基础理论支持使用直接的物理量应用于敏感性分析。

3.8 CMOST算例文件

下面的算例文件展示了CMOST系列特征、Study和引擎类型以及模拟器。在大多数算例中，提供了两个zip文件：一是CMOST运行前Project和Study文件，另外一个是运行后的文件。

文件夹/Project	Study	类型	主要特征	引擎	模拟器
FluidContact \ gmsmo014_3D	gmsmo014_3D	HM	• 计算流体界面 • 历史拟合误差	Random Brute Force	GEM
NPV_Excel \ SAGD_2D_OP_Excel	SAGD_2D_OP_ Excel	OP	• Excel类型和高级目标函数 • 软性约束条件	PSO	STARS
PUNQ_infill \ punq_infill	punq_infill	OP	• 公式 • Include文件 • NPV目标函数	DECE	IMEX
Relative Perm Corey \ 1_Insert_To_Dataset	RelPermMatch_ InsertToDataset	HM	• 公式 • 历史拟合误差目标函数	DECE	IMEX

文件夹/Project	Study	类型	主要特征	引擎	模拟器
Run Builder Silently \ AQU_40ACRES_ Builder	AQU_40ACRES_ Builder	HM	• Run Builder Silently type pre-simulation command	DECE	IMEX
RunGOCAD \ PUNQS3_HM_SA	PUNQS3_HM_SA	SA	• Run GOCAD type user-defined study • 确定时间日期	RSM	IMEX
SAGD 100 Realizations \ SAGD_3Pairs_ 100Realization	SAGD_3Pairs_ 100Realization	User-Defined	• 用户自定义的手动引擎类型 • 运算所有参数组合方案	Manual	STARS
SAGD DynaGrid \ SAGD_2D_DynaGrid_ Optimization	SAGD_2D_ DynaGrid_ Optimization	OP	• 历史拟合误差目标函数 • 用户定义编码类型 • 交互图中查看帕特洛前沿	DECE	STARS
SAGD Numerical Tuning \ SAGD_NT	Tuning	OP	• 特性 • 用户自定义编码类型	DECE	STARS
SAGD_SA_HM_OP \ SAGD_Pair1_Brown	SA	SA	• 确定日期时间 • 连续参数 • 用户定义时间序列 • 属性vs. 距离关系 • 历史拟合误差目标函数	RSM	STARS
	HM	HM	• 确定时间日期 • 连续参数 • 用户自定义时间序列 • 属性vs. 距离关系 • 历史拟合误差目标函数	DECE	STARS

文件夹/Project	Study	类型	主要特征	引擎	模拟器
	Optimization	OP	• 确定时间日期 • 动态时间日期 • 连续参数 • 属性 vs.距离 • NPV 目标函数	DECE	STARS
SAGD_SA_OP_UA \ SAGD_Pair1_Green	Sensitivity study	SA	• 用户自定义时间序列 • 确定时间日期 • 动态时间日期 • 公式 • 连续参数 • NPV 目标函数	RSM	STARS
	Optimization study	OP	• 用户自定义时间序列 • 确定时间日期 • 动态时间如期 • 公式 • 连续参数 • 硬性约束条件 • NPV 目标函数	DECE	STARS
	Base-case uncertainty	UA	• 用户自定义时间序列 • 确定时间日期 • 动态时间日期 • 公式 • 连续参数 • NPV目标函数	Monte Carlo	STARS

文件夹/Project	Study	类型	主要特征	引擎	模拟器
	Optimal-case uncertainty	UA	• 用户自定义时间序列 • 确定时间日期 • 动态时间日期 • 公式 • 连续参数 • NPV 目标函数	Monte Carlo	STARS
Shale Gas \ Shale Gas	HM_New	HM	• 差分演化 • Run Builder Silently	DE	IMEX
STARS-ME \ stmef008_Date_SA	stmef008_Date_SA	SA	• STARS-ME • Special dictionary	RSM	STARS -ME
Well Testing \ WellTesting	HmStudy	HM	• 连续参数	DECE	GEM

4 入门指南(Getting Started)

4.1 简介

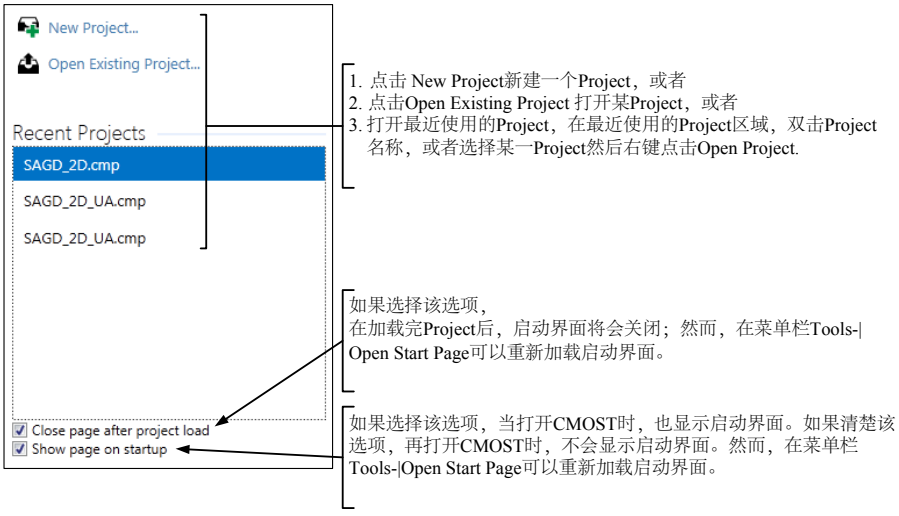
本章节详细介绍CMOST基本内容：

- 打开并操作CMOST
- 打开CMOST Project
- 创建CMOST Project
- 另存CMOST Project
- 使用Study管理器
- 常规界面操作及约定
- 退出CMOST

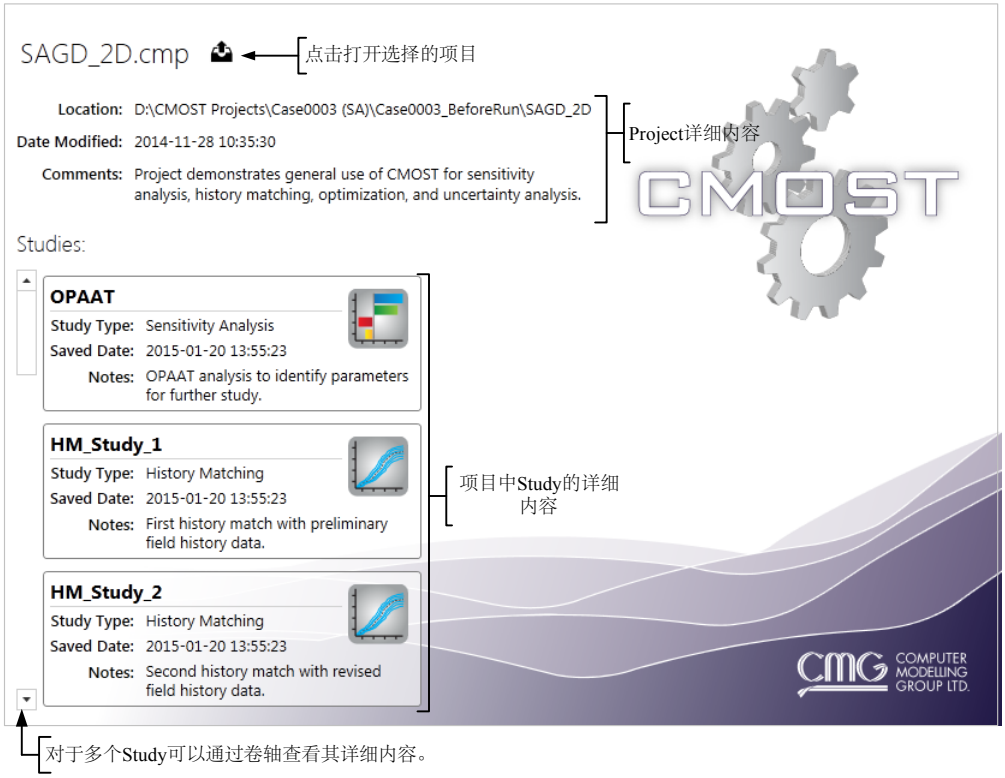
4.2 打开并操作CMOST

为了打开 CMOST：

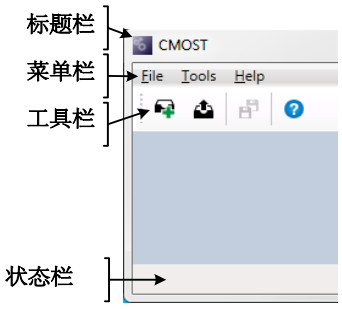
打开Launcher，然后双击CMOST 图标。出现CMOST启动画面，同时显示正在加载应用，随后呈现CMOST启动界面。下面详细介绍CMOST启动界面：



启动界面右面包含以下部分：



CMOST 用户界面：



CMOST主界面如下所示：

- 菜单栏：
  - 通过菜单栏-**File**，可以添加以下指令：

命 令	描 述
New | Project	创建一个新CMOST Project。参考 Creating a CMOST Project。
New | Study	若打开了一个新Project，在该部分可以添加一个新Study。参考To Create a New Study。
Open | Project	打开一个已经存在的CMOST Project。参考 Opening a CMOST Project.
Open | Existing Study	若打开了一个Project，在该部分可以打开一个已经存在的Study。
Save | Current Study	保存当前选择的 CMOST Study。
Save | Save All Save	保存所有的Studies。
Project As	另存当前的Project为一个新Project。参考 Saving CMOST Projects as New Projects.
Convert CMT/CMR File	将CMT 和 CMR文件转换为 CMOST CMP文件。参考Converting old CMOST Files to new CMOST Files.
Exclude Existing Studies	从Project中排除一个存在的Study。参考To Exclude a Study.
Close Project	关闭当前的CMOST Project。将会提示保存当前的修改。
Recent Files	打开一个近期浏览过的 Project。
Exit	关闭CMOST。将会被提示保存当前的修改。

- 通过**Tools**菜单， 可以启动以下选项：

命 令	描 述
Plotting Preferences	根据个人偏爱，编辑CMOST图形，包括颜色，数据点类型、 条以及线。
Open Start Page	打开CMOST启动界面。
Builder, Graph & 3D Versions	如果电脑中安装了多个版本的Builder, Graph & 3D，CMOST会选择使用缺省的版本。
Update Results and Analyses	如果做了修改，例如添加了一个参数或一个目标函数，点击Update Results & Analyses，可以刷新所有的结果。更多信息参考Update Results and Analyses。 Update Results and Analyses 。

- 通过**Help**菜单，你可以启动以下选项：

命 令	描 述
Index	打开CMOST帮助信息中的索引窗口。
Search	打开CMOST帮助信息中的查找窗口。
Contents	打开CMOST帮助信息中的内容窗口。
Generate Diagnostic File	打开CMG诊断工具，将CMG文件压缩成zip文件，发送给CMG技术支持员工来解决问题。
About	查看该版本关于CMOST的信息，在此可以连接到CMG官方网站，还可以得到CMG技术支持邮箱。

- **Status bar**：显示CMOST状态信息，根据选择的页面变化。

- **Toolbar**工具栏提供了以下按钮选项，当打开或新建Study时，有些按钮才会出现或被激活。

按 钮	描 述
New Project	打开该对话框，创建新的Project。如果打开了另外一个Project，将会建议关闭Study引擎，保存文件。

按 钮	描 述
Open Project	打开已经存在的Project。如果打开了另外一个Project，将会建议关闭Study引擎，然后保存文件。
New Study	打开Project后，可以点击该按钮，用来新建Study。
Save Current Study	保存当前Study设置及结果，但不关闭该Project
Save All	保存当前所有的Study设置及结果，但不关闭。
Start Engine	启动CMOST引擎，如果其他Study正在运行或者 没有做好运行准备，则不能使用该按钮。
Pause Engine	暂停CMOST引擎，该按钮只有在有CMOST运行时才可以使用。
Stop Engine	停止Stop引擎 ，该按钮只有在CMOST运行时才可以使用。
Help	打开CMOST帮助页面。

4.3 打开CMOST Project

如上所述，可以通过两种方式打开CMOSTProject，一种是在启动界面，点击**Open Project** ，另外一种是通过菜单栏，如下：

1. 点击**File | Open Project**。

注意：如果还打开了一个Project，建议先保存。

2. 浏览Project文件夹，然后点击**Open**。该Project将打开**Study Manager** 文件夹。

注意：

a. 如果打开的Project中包含一个或多个有问题的Study，会出现一个警告信息。如果点击**OK**，Project会打开，但是有问题的Study不会被加载。

b. 如果要打开的Project已经被其他人打开，在编辑和保存时会被限制，如下所述：

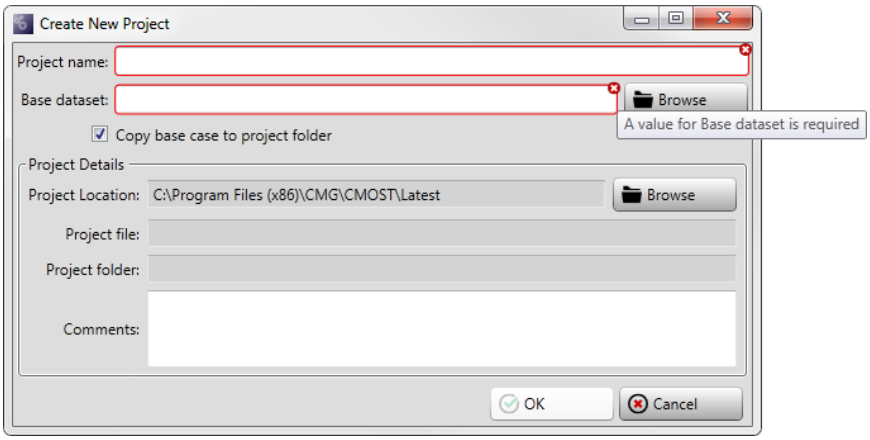
注意：可以通过以下方式打开CMOST文件：

1) 将 .cmt 或 .cmr 文件拖到CMOST桌面图标。通过这种方式只能打开一个.dat文件。
2) 将.cmp文件拖到桌面CMOST图标。
3) 右键点击.cmp文件，然后选择**Open with | CMG.CmostPlus.Studio.View**.

4.4 创建 CMOST Project

为了创建一个CMOST Project：

1. 在启动界面，点击**New Project**或在菜单栏点击 **File | New | Project**。出现 **Create New Project** 对话框：

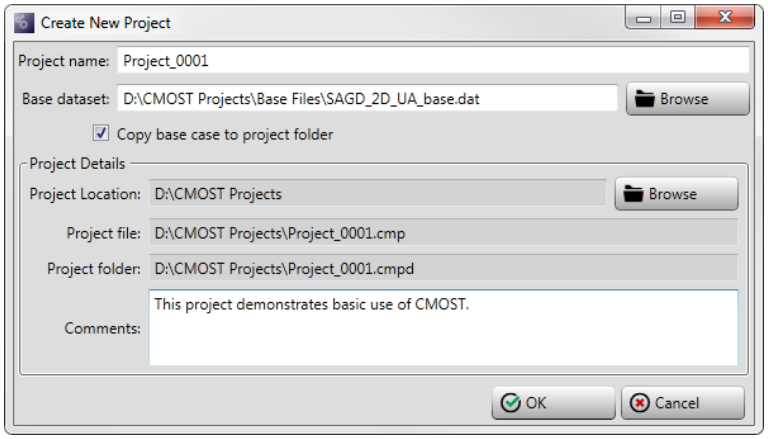


注意：如上所述，红线框是必填部分。把鼠标移到到有红叉部分，查看提示信息。

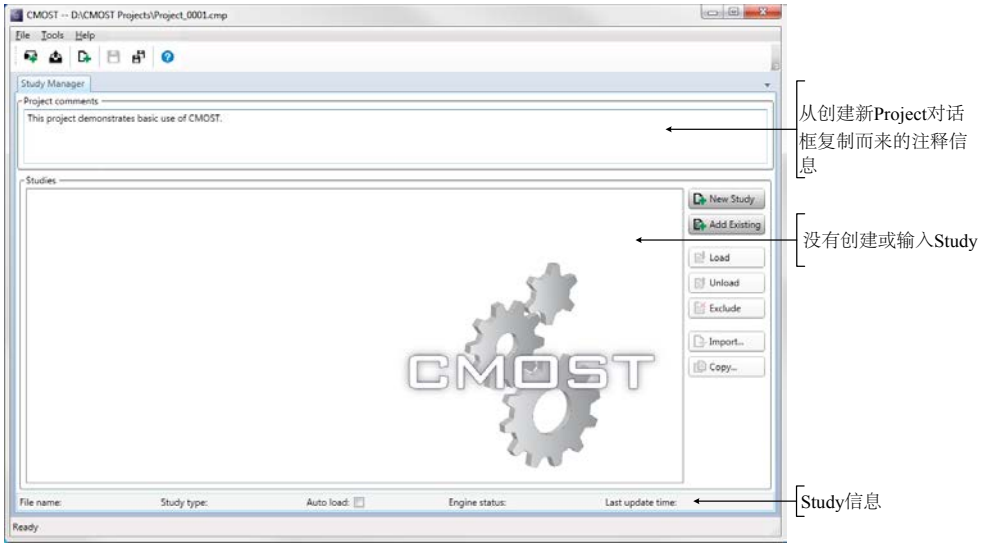
2. 在**Create New Project** 对话框，输入以下信息：

   - **Project name**（必需）：需要输入Project名称，可以是任意字符的组合，包括空格。
   - **Base Dataset**（必需）：点击空白处右面的 **Browse**，浏览并选择需要的基础文件，然后点击**Open**。在该路径下，会出现基础文件及结果文件。
   - **Copy base case to project folder**：如果选择该选项，基础文件将会被复制到Project.cmpd文件夹。如果没有选择该选项， 基础文件还是存放在原来的路径。
   - **Project Location**：点击**Browse**，浏览并选择文件夹，然后点击Open。
   - **Project file**：依据Project名称和位置，将会自动输入到.cmp文件。
   - **Project folder**：依据Project名称和位置，将会自动输入到 .cmpd文件夹。
   - **Comments**：如果有必要，可以输入Project注释，这些注释会出现在启动界面。

输入必要的文件后，就会激活**OK** 按钮，如下所示：

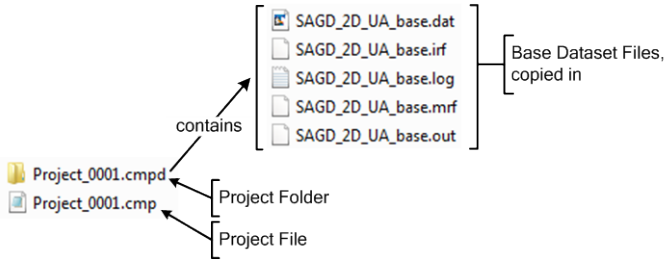


3. 点击 **OK**创建并保存CMOST文件及文件夹，出现如下所示的界面：



如上所述，将**Create New Project** 对话框输入的注释复制到启动界面相应区域。

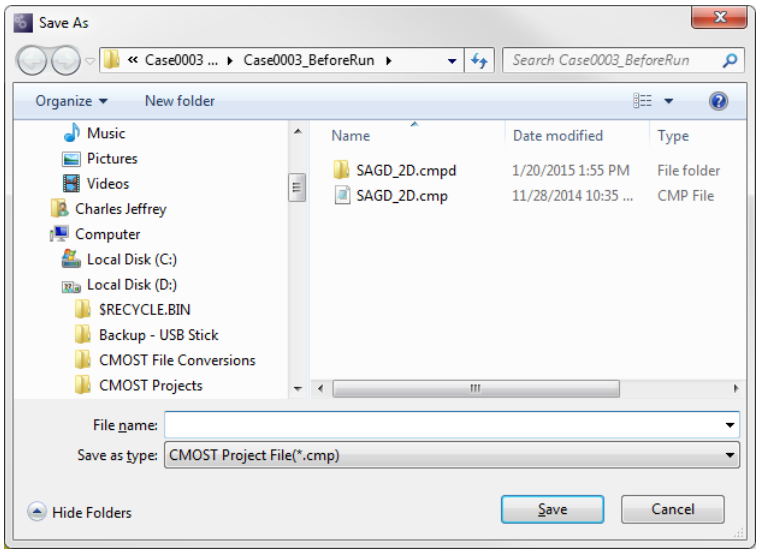
基础文件被复制到指定的.cmpd 文件夹，下面的例子表示创建CMOST Project后的文件及文件夹：



4.5 保存CMOST Project为新 Project

保存一个已经存在的CMOST Project到新Project，包括所有的Study、设置以及结果。这样做的目的，或许想考查该Project的操作方案，但又想保留原始Project中的数据。

1. 打开需要另存的Project。
2. 在菜单栏，点击**File | Save Project As**。出现一个另存的对话框。



3. 在另存对话框，选择保存路径，输入Project名称，然后点击保存。在该路径下会创建新Project文件夹。

4.6 使用Study Manager

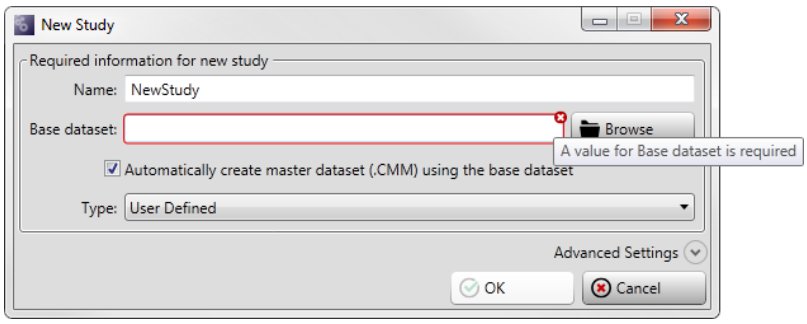
通过**Study Manager**选项，可以：

- 创建Study
- 查看Study详细信息
- 修改Study名称
- 添加已经存在的Study到某个Project
- 加载/卸载Stud
- 移除Study
- 从Study中输入数据
- 复制Study

注意：某些**Study Manager**功能，例如创建Study或添加已经存在的Study都可以通过文件菜单栏实现。然而，实现该部分功能的程序需要点击**Study Manager** 选项中的按钮。

4.6.1 创建新Study

1. 在Study Manager 右侧，点击 New Study 按钮，或者在菜单栏点击**File | New | Study**。出现对话框**New Study**：



注意：如上所述，需输入红色部分。将鼠标移动到红叉 标志处，查看此处关于输入的提示信息

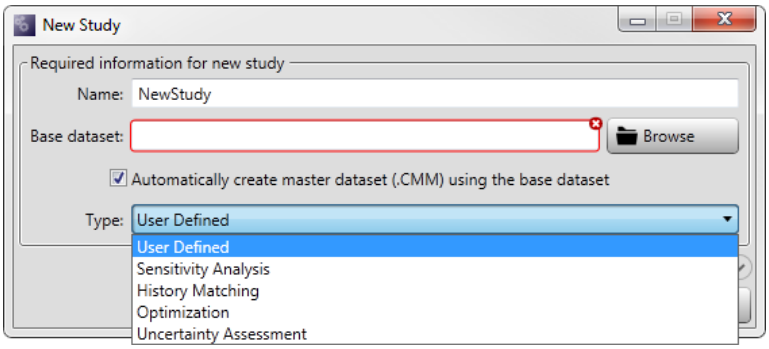
在空白处输入以下信息：

- **Name**：为**Study**输入名字。点击**OK**后，**Study**标会出现在**Study Manager**。

- **Base dataset**：浏览并选择基础文件。

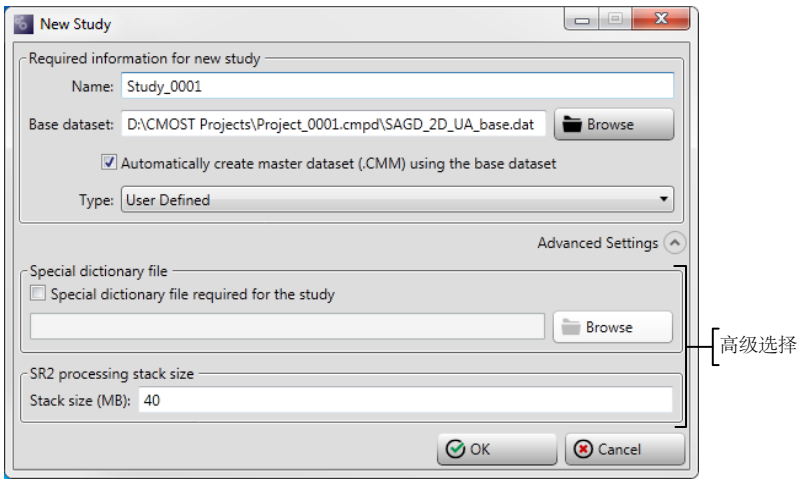
- **Automatically create master dataset (.CMM) using the base dataset:** 如果选择了该复选框，就会自动创建CMM文件。CMOST主文件会和基础文件保存在相同的文件夹。如果没有选择该选项，需要在**General Properties**定义CMOST主文件。

- **Type**：从下拉菜单中选择Study类型，如下图：



注意：在**Engine Settings**中还可以重新修改Study类型。

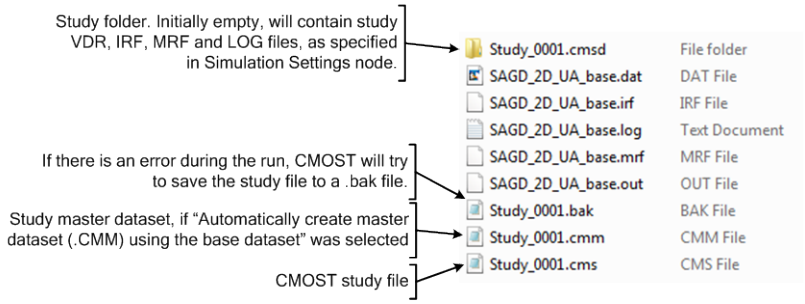
如果点击**Advanced Settings**中 按钮，则进入以下设置：



- **Special dictionary file**：CMOST支持特殊的模拟器版本，例如STARS-ME™。如果想用特殊的模拟器来处理SR2文件，需要选择输入特殊处理器文件。浏览并选择该文件。

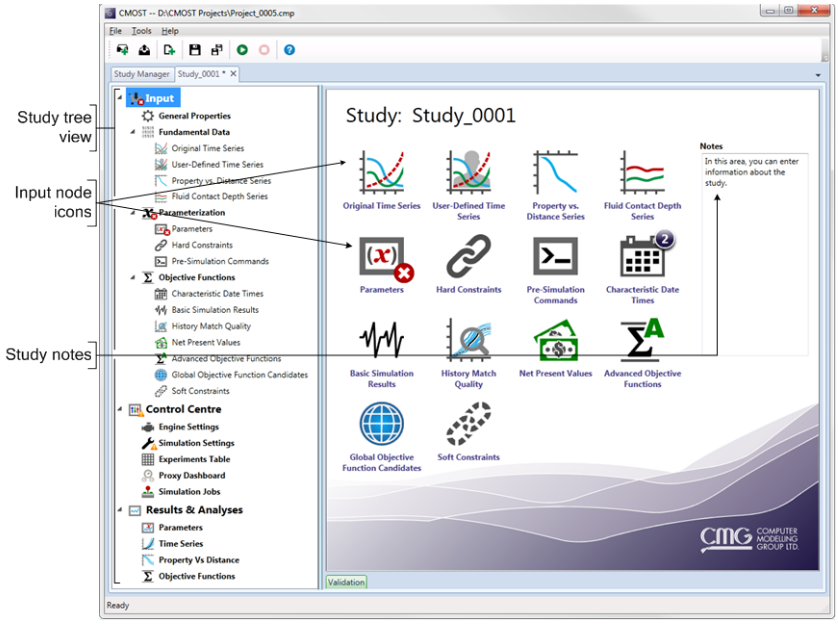
- **SR2 processing stack size**：处理SR2文件堆栈大小。SR2阅读器利用堆栈大小读取SR2文件。默认值是40MB。

2. 点击**OK**，创建新Study，如下：



注意：如上所述，在运行过程中，如果出现错误，CMOST会保存备份文件.bak。该备份文件是最后的有效文件，和Study文件有相同的格式。

- 下面的例子中，已经添加了新的Study，**General Properties**窗口内容如下所示：



Study树状结构包括以下几个节点：

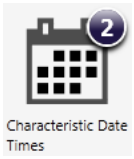
Input：通过该节点及子节点，可以创建并修改Study数据。

Control Centre：通过该节点及子节点，可以配置、运行、监测以及控制CMOST Study。

Results and Analyses：通过该节点及子节点，查看并分析CMOST Study结果。

为了打开Study节点页面，点击树状结构图中该节点及子节点，在Input界面，如上所示，点击子节点。

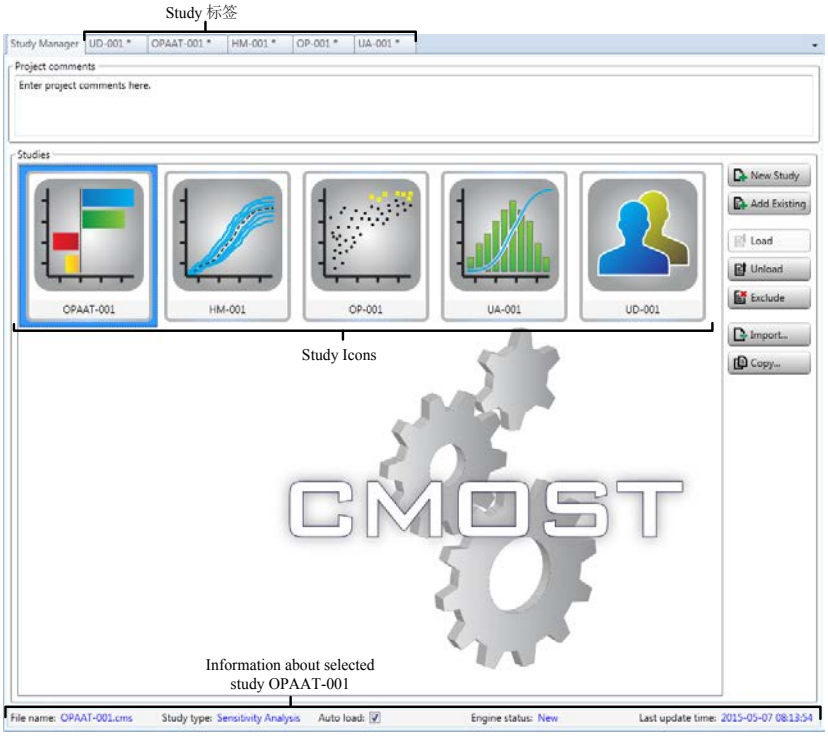
在Input界面，数字信息在该节点下是重叠的。下面的例子表示在Characteristic Date Times定义了两个时间点：



在Study树状图中，每个节点和子节点状态图标叠加在图标之上。

图 标	意 义
Error	表示该节点或子节点有错误。点击页面底部Validation图标了解更多信息。
Warning	某节点下设置及配置有警告信息。在Simulation Settings节点下，如果没有选择scheduler，会出现警告。点击页面底部Validation图标了解更多信息
Information	该页面包含的信息可能对用户非常有用，点击页面底部Validation图标了解更多信息。
No errors or warnings	当前的节点或子节点没有错误或警告。

- Study图标添加至Study Manager，如下面列子所示，包含五个Study，一个敏感性分析（SA），两个历史拟合（HM），一个优化（OP）以及一个不确定性分析（UA）：



4.6.2 查看Study

点击Study图标，在页面底部查看该Study的详细信息。

注意：如果选择**Auto load**，当打开Project时，不管先前是什么状态，Study会自动被加载。如果没有选择**Auto load**，Study恢复为**Load/Unload**状态。

双击Study图标，或右键并选择**Go to Study**，查看Study的详细内容。

4.6.3 修改Study名字

点击，然后编辑Study的名字，但不改变与之相关的文件及文件夹的名字。

4.6.4 添加一个已经存在的Study到当前的Project

点击**Add Existing**按钮或者在菜单栏点击**File | Open | Existing Study**打开Windows窗口，在Project文件夹中浏览已经存在的Study文件，然后**OPEN**。

4.6.5 加载/卸载Study

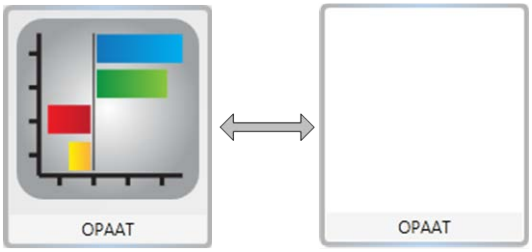
在当前的Project中，可以卸载或删除Study，出现以下情况：

- 需要减少Project文件所占内存的大小。
- Project包含已经运行的Study并且不需要再次运行。
- ·Project中包含已经过期的Study或失效的Study。.

一旦卸载：

- 在Study Manager中的Study图标将会发生变化，表明Study已被卸载，如下图所示。
- Study被删除，不能查看对应信息。
- Study不能被复制到新的Study。
- 数据也不能被输出。

为了卸载Study，点击该图标，右键选择**Unload**，或者点击Study选择**Unload**按钮。Study图标会改变，表示已被卸载：



Study Loaded　　Study Unloaded

加载Study，右键Study图标，然后点击**Load**。

4.6.6 移除Study

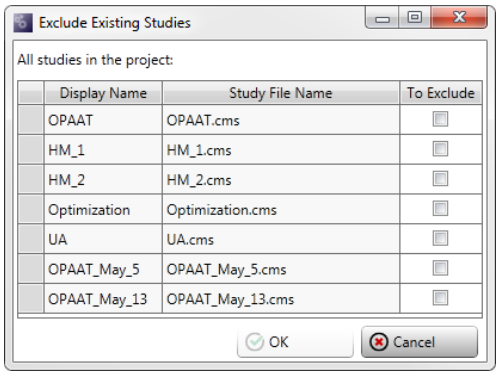
可以从Project中移除一个Study，移除后，在Project窗口中将看不到该Study，并且也不会占用计算机内存。例如，或许已经创建好一个Study并完成了运算，查看其运算结果，现在不需要再查看其运算结果。

1. 右键Study，选择**Exclude**或者点击Study图标，然后点击启动界面右面**Exclude**按钮。无论哪种情况，都会被询问是否要删除选择的记录。

2. 点击**Yes**。Study图标从启动界面中删除，其文件及文件夹不会被删除，因此还可以把移除的Study文件重新加载回来。

也可以将多个Study从菜单栏排除，如下：

1. 点击 **File | Exclude Existing Studies**。**Exclude Existing Studies** 对话框如下：

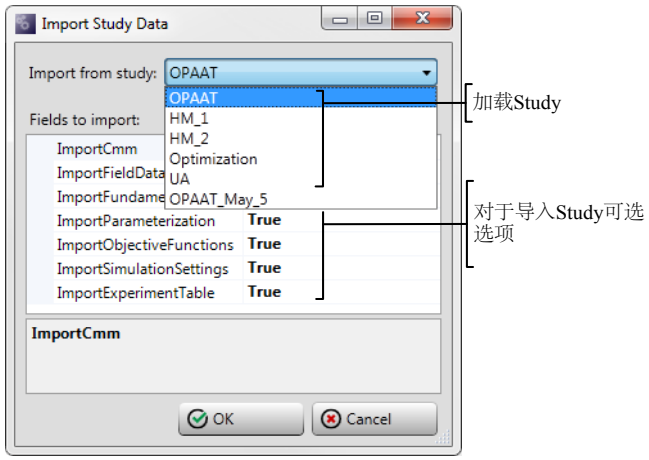


2. 在**To　Exclude**列，选择想要移除的Study，然后点击**OK**。Study从Project中移除，但其文件夹及文件没有被删除。在启动界面中，通过**Add Existing**，随时可以将Study添加到Project中。

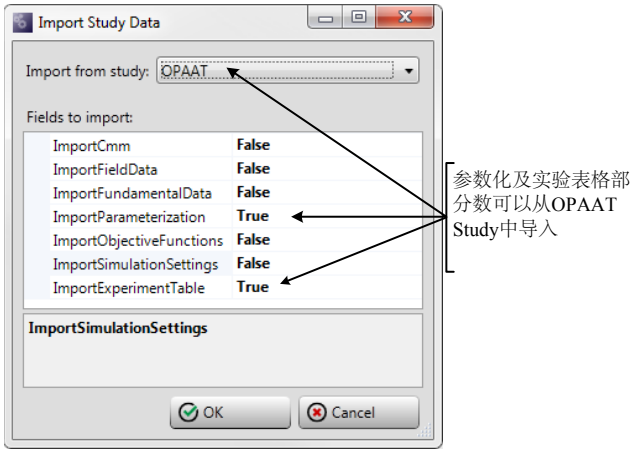
4.6.7 从Study导入数据

可以将数据从一个Study导入另外一个Study（在同一个文件夹加载）：

1. 右键Study，选择想要导入的Study数据，然后选择**Import**。**Import Study Data**对话框如图所示，可以选择想要加载的Study及对应数据部分：



2. 选择想要导入的Study。
3. 选择想要导入的数据，True表示导入，False表示不导入，如下例子所示：

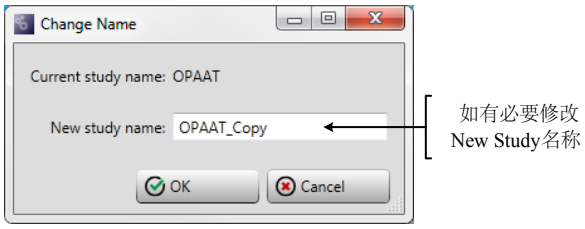


4. 4点击OK。在上面的例子中，参数数据及实验表格数据从OPAAT导入到选择的Study。

4.6.8 复制到新Study

可以复制（加载）Study到新Study，如下：

1. 点击想要复制的Study，右键选择**Copy to New Study**。出现修改名字的对话框，默认的名字是在原先Study名字的基础上，添加“_Copy”，如下图所示：



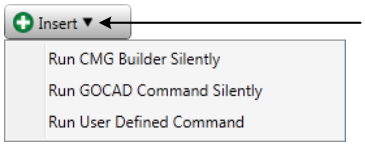
2. 如有需要，可以修改Study名字，然后点击**OK**。这样就创建了一个新的Study，和原来的Study一样。新Study图标添加至启动界面。

4.7 常规操作及约定

点击不同的Study节点，会显示不同的信息，然而对整个窗口而言，有些信息是通用的。

4.7.1 按钮及图标

当点击标签旁边的 ▼ 按钮时，会出现一个下拉菜单选项，如下所示

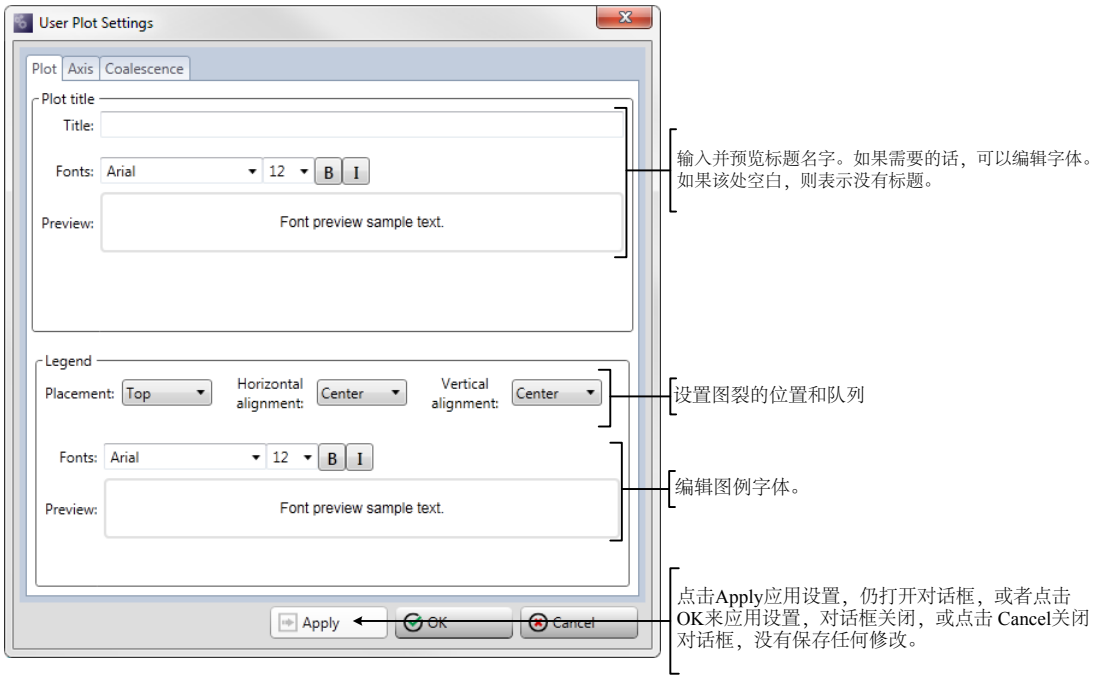


4.7.2 图形偏爱设置

参考Tools部分Plot Preferences信息，使用**User PlotPreferences**对话框来设置图片风格、线颜色、数据点以及其他选项。

4.7.3 图形设置

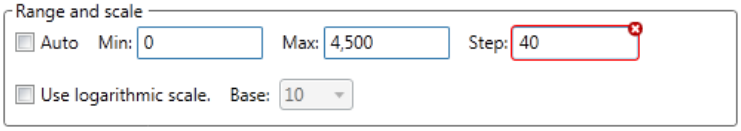
在**Results & Analysis**节点，右键CMOST任意一条曲线，选择**Properties**打开**User Plot Settings**对话框：



注意：

1. 当点击时间序列曲线时， **Coalescence**（合并）键也将打开。根据图形类型来呈现该设置。

2. CMOST不允许输入无效的图形设置参数，例如对于某一取值范围，最小值就不能大于最大值。如下面例子所示，CMOST会自动标注无效输入，因为输入的step大小为40，这样就会导致产生多于（大于100）CMOST正常显示的步数。



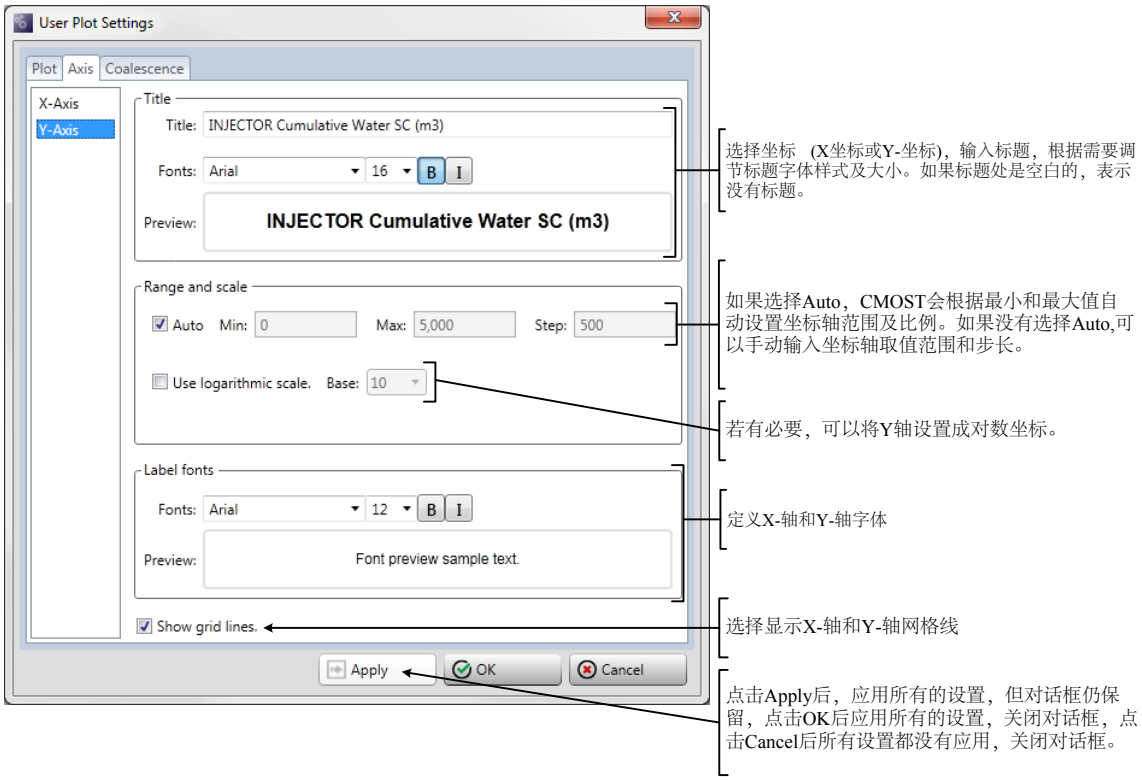
同样，如果输入无效数据，对话框中的OK键也不会被激活。

4.7.3.1 Plot 选项

通过Plot选项，可以输入图形曲线名称，编辑字体，另外还可以自定义图例位置以及字体大小等。

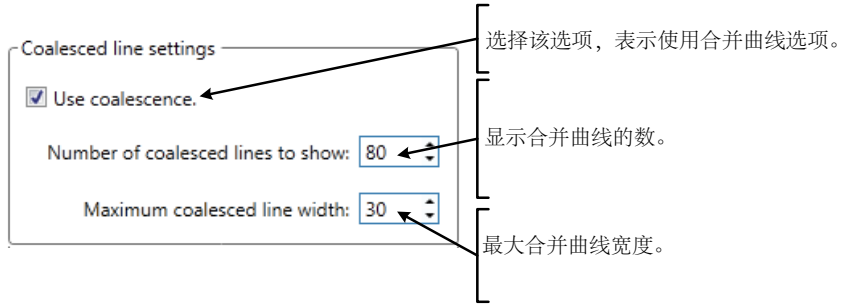
4.7.3.2 Axis 选项

通过**Axis**选项，如下图所示，对X轴和Y轴自定义坐标标题、形式、取值范围、比例及步长大小。依据图形的类型，可以选择使用对数坐标。下面的例子显示的是Y轴的设置，X轴的设置与之类似。



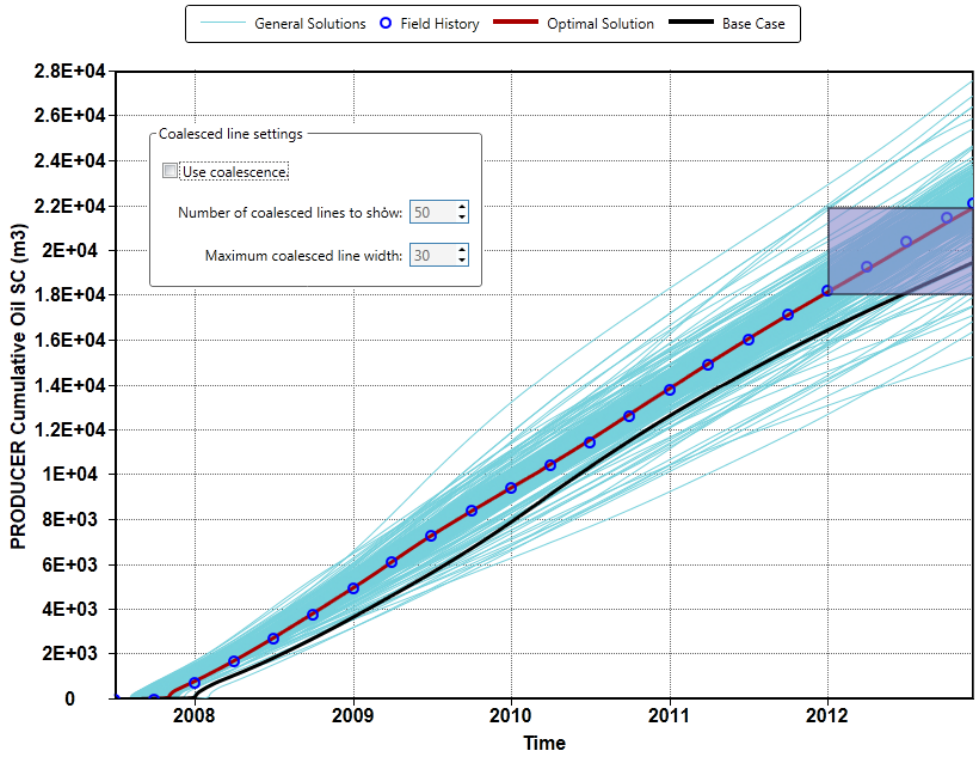
4.7.3.3 Coalescence 键

在时间序列图和参数vs. 距离图中，能够使用**Coalescence**（合并）键。在实验方案较多的情况下，为了加快图形的显示速度，CMOST会自动将曲线类型相似的方案合并。缺省设置如下所示：

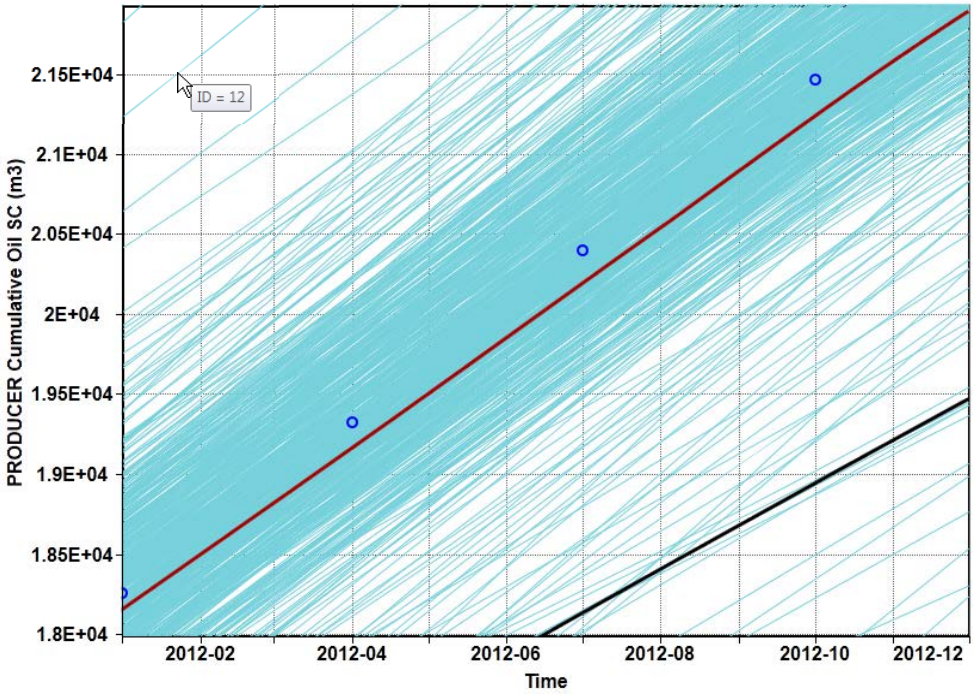


合并曲线的宽度表示实际曲线被合并的数目。两条曲线的合并由以下因素决定。第一，两条曲线对应数据点之间的平均距离，将平均距离最小的曲线合并。CMOST会按照这个方法继续寻找，达到允许显示的合并曲线数为止。当两条曲线合并后，CMOST会自动随机选择显示其中一条。当三条或三条以上曲线合并后，CMOST会自动应用数学算法找到其中某一条曲线作为代表显示出来。

在下面的例子中，我们没有选择使用合并曲线选项。因此显示的是全部曲线。我们可以局部放大部分曲线：

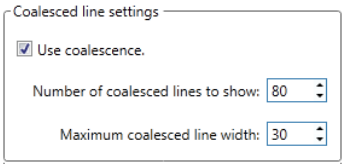


这是放大后的曲线部分：

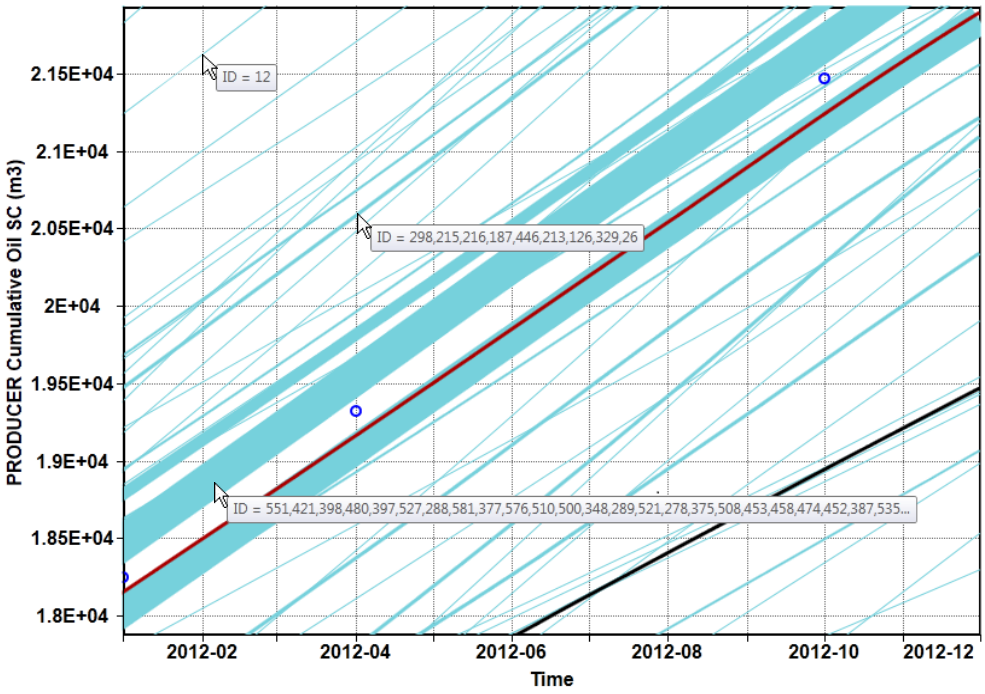


如上，可以利用鼠标选择其中一条试验曲线。上面的例子，没有进行合并，所以每一条曲线代表一个单独的实验方案，该图中选择的是ID=12的实验方案曲线。

现在，我们使用合并曲线选项，并且使用缺省值：



如果放大和之前相同的曲线部分，可以看出，合并后的曲线变得稀疏。将鼠标放在合并后的曲线上，可以显示该曲线是由哪些试验方案合并而来，如下：



4.7.4 图片操作

4.7.4.1 将图片复制到粘贴板

为了将图片复制到一个文件：

1. 右键图片，选择**Copy Image to Clipboard**.
2. 在目标应用文件（例如Word）点击 **Paste**（或 CTRL+V）。

4.7.4.2 保存图片

为了将曲线保存成图片文件：

1. 右键图片，选择**Save Image**.
2. 在对话框选择需要的文件类型，浏览文件夹，然后点击**Save**。

4.7.4.3 关于数据点和曲线

呈现在graph中的数据点和曲线使用Plotting Preferences中的缺省值。可以根据需要改变Plotting Preferences中的缺省值。如果将鼠标移动到某一数据点，该点就会变成红色，并且会显示该点的数值。如果点击该数据点，它就会变成黑色加粗。

4.7.4.4 高亮显示试验方案

为了高亮显示某一试验方案，右键数据点或曲线，然后选择**Highlight the Experiment**。如果在某一张图中高亮显示了某一实验方案曲线，那么在其他图中也会高亮显示该曲线，允许在多个图中查看该高亮显示的曲线。若想取消高亮显示，右键清除**Highlight the Experiment**。

在有些情况下，或许需要点击 来刷新图片。也可以通过Experiments Table来高亮显示试验方案。

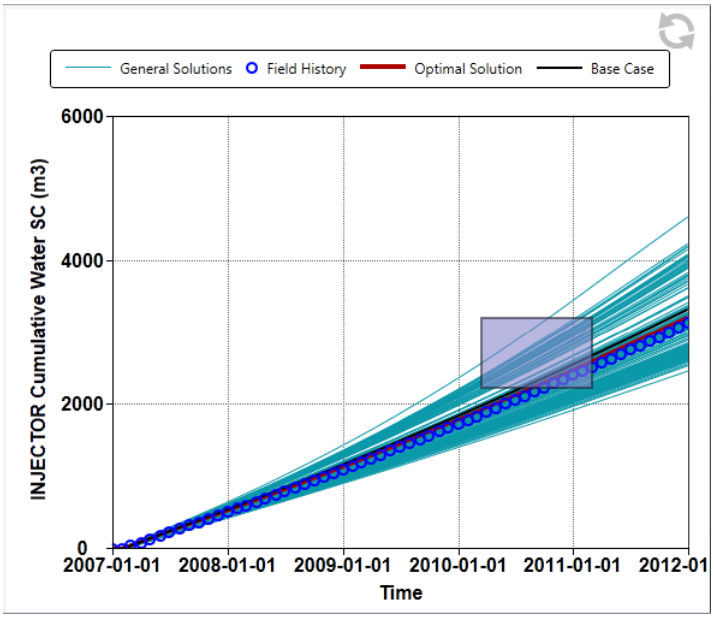
4.7.4.5 放大和缩小曲线

当通过CMOST Result Observers查看结果时，可以放大曲线的任何区域，如下：

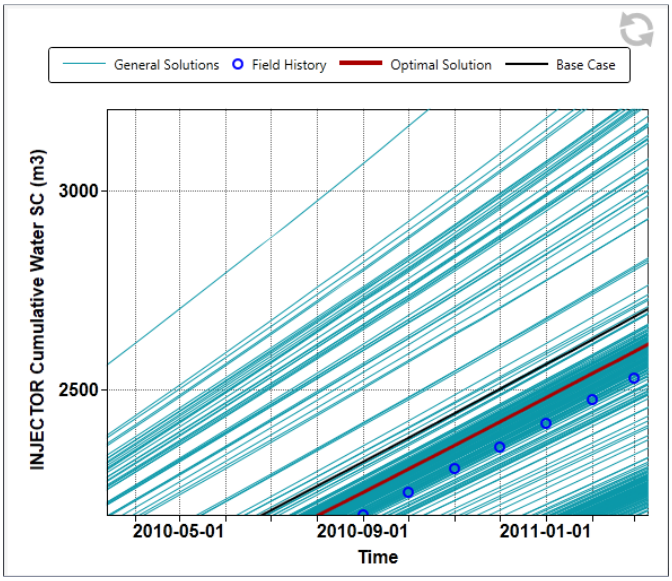
放大：

放大有两种方式。第一种是定义想要修改的区域：

1. 在图中选择想要放大的区域，从左下角拉到右上角，例如：



2. 释放鼠标，放大区域如下所示：



第二种方式是通过鼠标滑轮实现。选中放大区域，保持x 和 y坐标不变，滑动滑轮，放大目标区域。

恢复至原始尺寸：

右键曲线图，选择 **Un-zoom to 100%**。

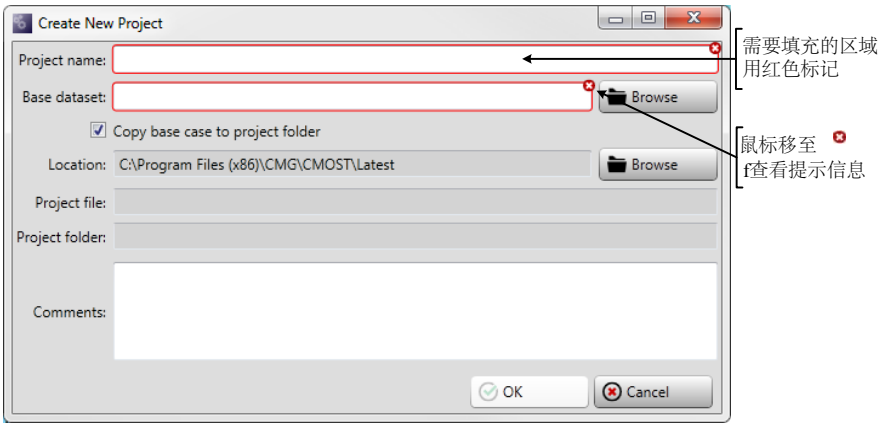
4.7.5 名称

必须为参数、目标函数、目标函数项等定义名称。需要满足以下几点：

- 第一个字符必须是字母。名称中的其他字符可以是字母、数字及下划线等。
- 名称区分大小写。
- 不允许出现空格。出现字符分开时，可以使用下划线，例如*perm_h and perm_v*。
- 模拟器关键字可以用来作为参数名称。对于熟悉模拟器关键字的用户来说，更容易理解定义的CMOST参数。
- 不要使用CMOST内部常用的关键字：this, Status, Dataset, Scheduler, Computer, Pattern, Source, Average, Maximum, Minimum, and Target。
- 名字必须是唯一的。如果某一个名字已经用于定义某个参数，那就不可以用来定义目标函数或其他参数。

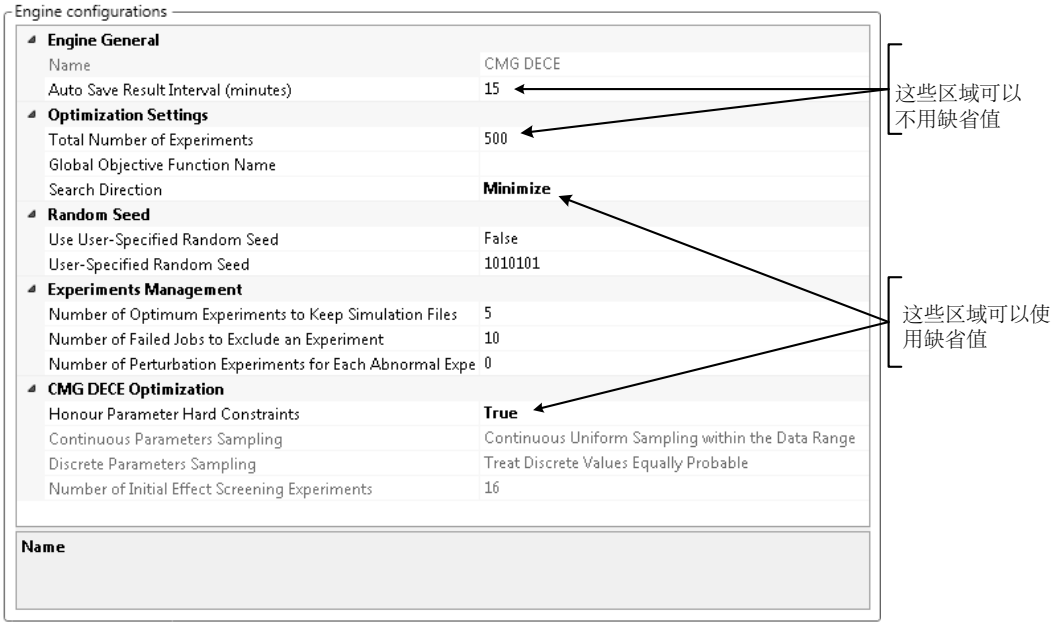
4.7.6 需要填充区域

对话框中，红色区域是必须填充的区域，例如：



4.7.7 缺省值

在缺省表格，当数据使用缺省值时，会加载缺省名字，如下所示：

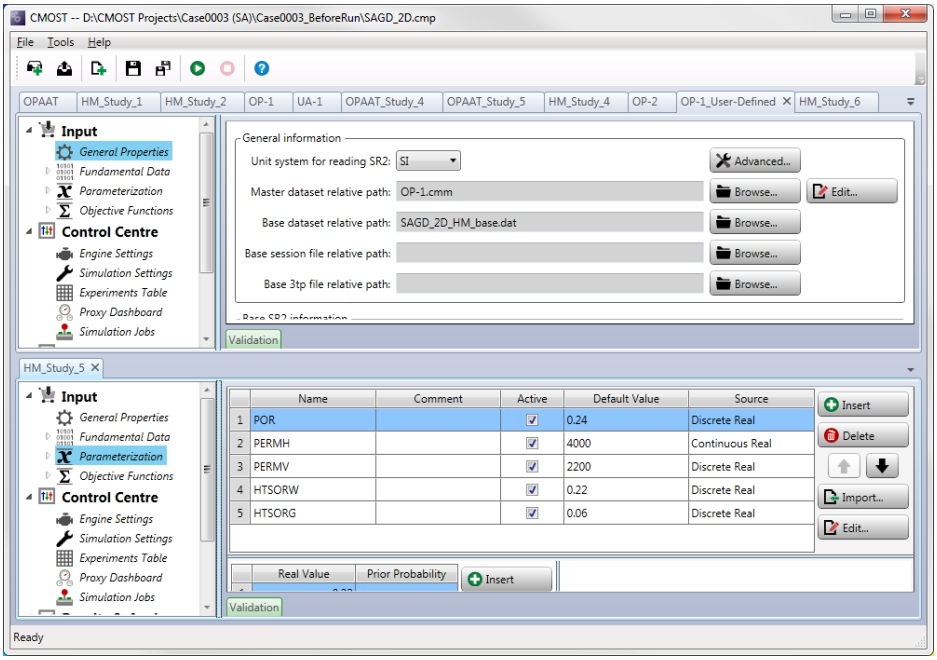


4.7.8 按钮显示

右键点击**Study Manager**或**Study**时，可以使用以下命令：

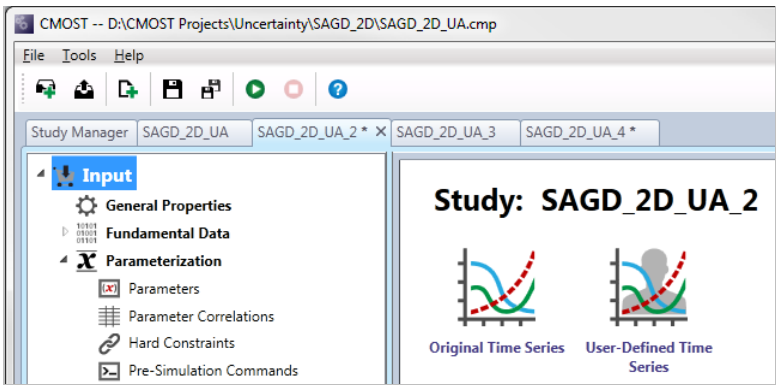
- **Close**: 不能关闭 Study Manager，但是可以关闭Study选项。通过关闭Study选项来关闭当前显示的Study，但不是删除该Study。为了重新打开该Study，在Study Manager下，双击该Study图标，或者右键该图标选择Go to Study。

- **New Horizontal Tab Group, New Vertical Tab Group**：对不同的Study进行编组，可以选择垂直分组和水平分组，如下所示：



可以有水平和垂直标签分组的组合，也可以在不同组之间来回切换。点击 **Active Files** 按钮，选择其中的一项。不论那一项，在Project 启动界面中显示所有的Study。

如果对Study做了某些改变，在其名字对应的标签处会出现一个星号，如下例， 对SAGD_2D_UA_2 and SAGD_2D_UA_4做了一些修改，选择需要保存的Study，而仅仅需要保存这些已做修改的Study。



4.7.9 表格

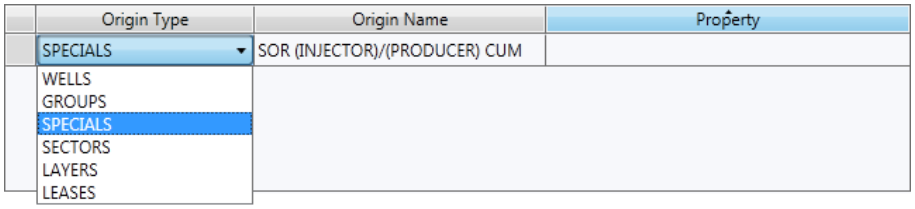
4.7.9.1 插入、删除及重复行

通常，点击**Insert**，在选择的行下面插入一行。点击 **Delete** 用来删除某一行。点击 **Repeat**，用来重复该行，但是可以使用不同的**Origin Type**。

4.7.9.2 输入选项框数据

根据选项框表格，通过以下几种方式输入数据：

- 在某些情况下，选项框中的内容不能直接修改，例如文件中已经输入的技术参数。
- 在某些情况下，点击选项框时，会出现一个下拉菜单，选择需要的选项。



- 在某些情况下，可以直接输入某些参数。

4.7.9.3 调节表格栏宽度

在表格中，通过拖拉调节表格所在列的宽度。

4.7.9.4 对表格行排序

点击表格表头对行进行排序，例如：

Name 显示行顺序是原始输入的顺序。

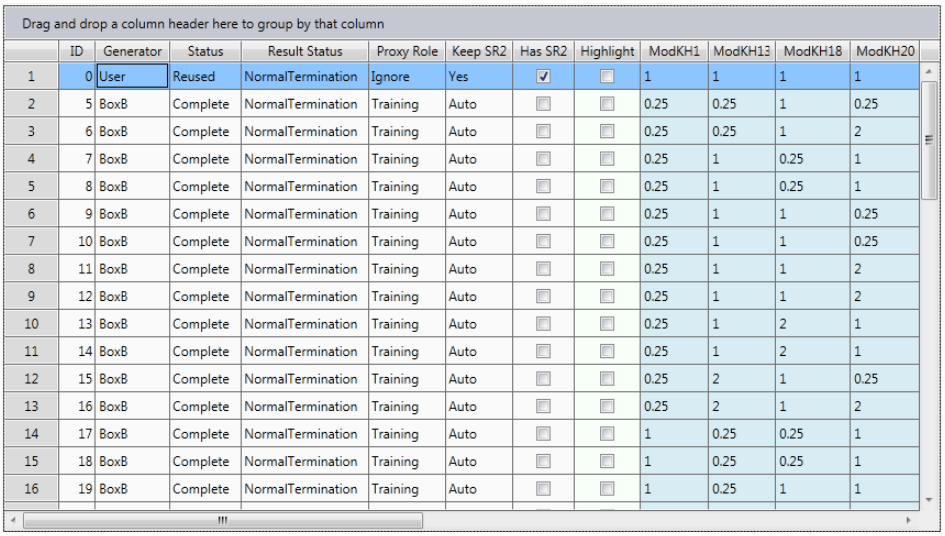
Name 显示行按升序排序。

Name 显示行按降序排序。

4.7.9.5 整理表格的行和列

表格支持按照一列或多列进行分类，说明如下：

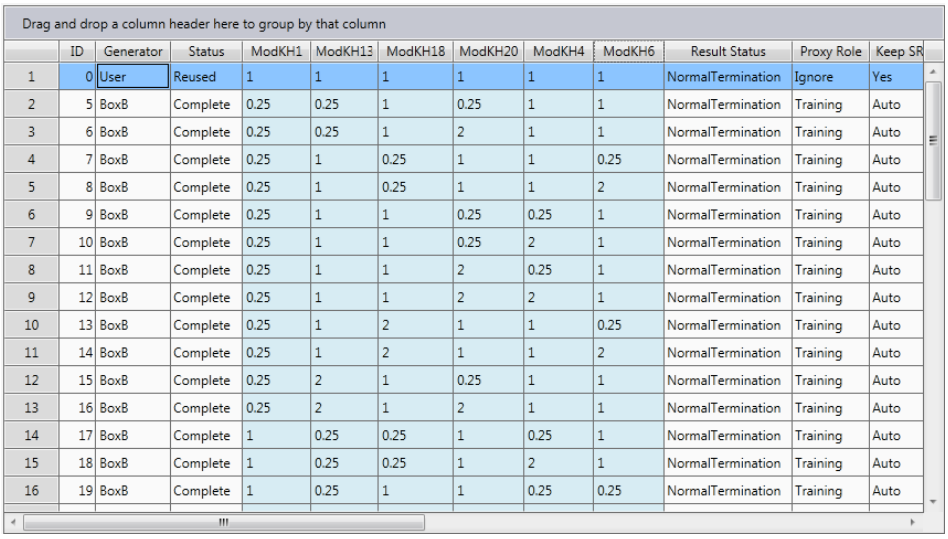
1. 打开某一Study的**Experiments Table**：



Drag and drop a column header here to group by that column

	ID	Generator	Status	Result Status	Proxy Role	Keep SR2	Has SR2	Highlight	ModKH1	ModKH13	ModKH18	ModKH20
1	0	User	Reused	NormalTermination	Ignore	Yes	☑	☐	1	1	1	1
2	5	BoxB	Complete	NormalTermination	Training	Auto	☐	☐	0.25	0.25	1	0.25
3	6	BoxB	Complete	NormalTermination	Training	Auto	☐	☐	0.25	0.25	1	2
4	7	BoxB	Complete	NormalTermination	Training	Auto	☐	☐	0.25	1	0.25	1
5	8	BoxB	Complete	NormalTermination	Training	Auto	☐	☐	0.25	1	0.25	1
6	9	BoxB	Complete	NormalTermination	Training	Auto	☐	☐	0.25	1	1	0.25
7	10	BoxB	Complete	NormalTermination	Training	Auto	☐	☐	0.25	1	1	0.25
8	11	BoxB	Complete	NormalTermination	Training	Auto	☐	☐	0.25	1	1	2
9	12	BoxB	Complete	NormalTermination	Training	Auto	☐	☐	0.25	1	1	2
10	13	BoxB	Complete	NormalTermination	Training	Auto	☐	☐	0.25	1	2	1
11	14	BoxB	Complete	NormalTermination	Training	Auto	☐	☐	0.25	1	2	1
12	15	BoxB	Complete	NormalTermination	Training	Auto	☐	☐	0.25	2	1	0.25
13	16	BoxB	Complete	NormalTermination	Training	Auto	☐	☐	0.25	2	1	2
14	17	BoxB	Complete	NormalTermination	Training	Auto	☐	☐	1	0.25	0.25	1
15	18	BoxB	Complete	NormalTermination	Training	Auto	☐	☐	1	0.25	0.25	1
16	19	BoxB	Complete	NormalTermination	Training	Auto	☐	☐	1	0.25	1	1

2. 通过拖拉列标题栏向左或向右，重新对列进行排序。在下面的例子中，将参数（**ModKH1**、**ModCH13**等等）移至了左边。

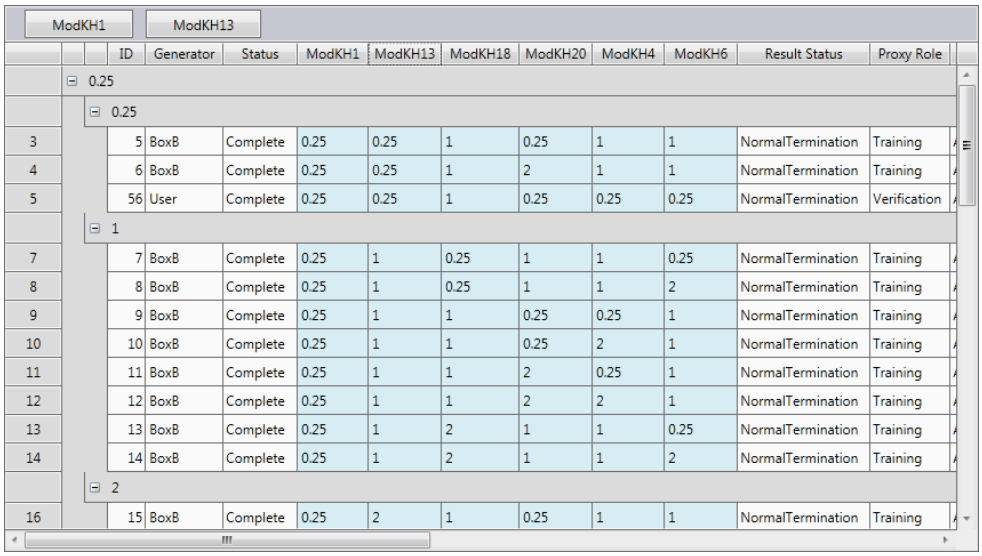


Drag and drop a column header here to group by that column

	ID	Generator	Status	ModKH1	ModKH13	ModKH18	ModKH20	ModKH4	ModKH6	Result Status	Proxy Role	Keep SR
1	0	User	Reused	1	1	1	1	1	1	NormalTermination	Ignore	Yes
2	5	BoxB	Complete	0.25	0.25	1	0.25	1	1	NormalTermination	Training	Auto
3	6	BoxB	Complete	0.25	0.25	1	2	1	1	NormalTermination	Training	Auto
4	7	BoxB	Complete	0.25	1	0.25	1	1	0.25	NormalTermination	Training	Auto
5	8	BoxB	Complete	0.25	1	0.25	1	1	2	NormalTermination	Training	Auto
6	9	BoxB	Complete	0.25	1	1	0.25	0.25	1	NormalTermination	Training	Auto
7	10	BoxB	Complete	0.25	1	1	0.25	2	1	NormalTermination	Training	Auto
8	11	BoxB	Complete	0.25	1	1	2	0.25	1	NormalTermination	Training	Auto
9	12	BoxB	Complete	0.25	1	1	2	2	1	NormalTermination	Training	Auto
10	13	BoxB	Complete	0.25	1	2	1	1	0.25	NormalTermination	Training	Auto
11	14	BoxB	Complete	0.25	1	2	1	1	2	NormalTermination	Training	Auto
12	15	BoxB	Complete	0.25	2	1	0.25	1	1	NormalTermination	Training	Auto
13	16	BoxB	Complete	0.25	2	1	2	1	1	NormalTermination	Training	Auto
14	17	BoxB	Complete	1	0.25	0.25	1	0.25	1	NormalTermination	Training	Auto
15	18	BoxB	Complete	1	0.25	0.25	1	2	1	NormalTermination	Training	Auto
16	19	BoxB	Complete	1	0.25	1	1	0.25	0.25	NormalTermination	Training	Auto

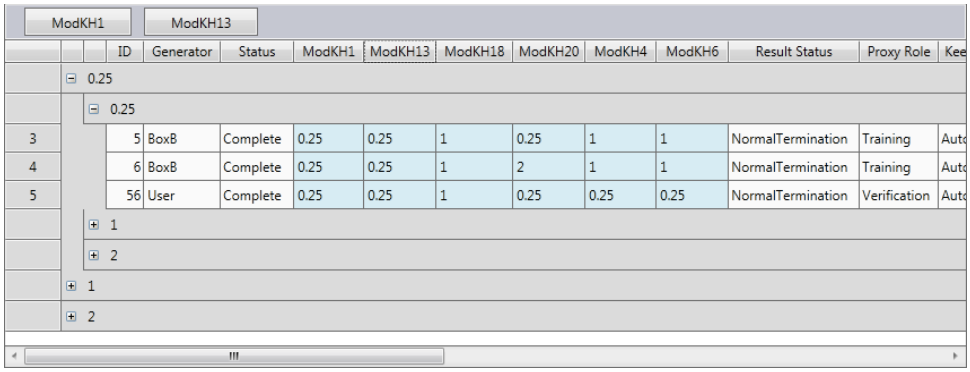
注意：如果输出**Experiments Table**到Excel，列的顺序将会保留。

3. 拖拉列标题到表格的上面区域来重新编排试验方案。在下面的例子中，分别依据参数**ModKH1**和**ModKH13**进行分类：



		ID	Generator	Status	ModKH1	ModKH13	ModKH18	ModKH20	ModKH4	ModKH6	Result Status	Proxy Role
	0.25											
	0.25											
3		5	BoxB	Complete	0.25	0.25	1	0.25	1	1	NormalTermination	Training
4		6	BoxB	Complete	0.25	0.25	1	2	1	1	NormalTermination	Training
5		56	User	Complete	0.25	0.25	1	0.25	0.25	0.25	NormalTermination	Verification
	1											
7		7	BoxB	Complete	0.25	1	0.25	1	1	0.25	NormalTermination	Training
8		8	BoxB	Complete	0.25	1	0.25	1	1	2	NormalTermination	Training
9		9	BoxB	Complete	0.25	1	1	0.25	0.25	1	NormalTermination	Training
10		10	BoxB	Complete	0.25	1	1	0.25	2	1	NormalTermination	Training
11		11	BoxB	Complete	0.25	1	1	2	0.25	1	NormalTermination	Training
12		12	BoxB	Complete	0.25	1	1	2	2	1	NormalTermination	Training
13		13	BoxB	Complete	0.25	1	2	1	1	0.25	NormalTermination	Training
14		14	BoxB	Complete	0.25	1	2	1	1	2	NormalTermination	Training
	2											
16		15	BoxB	Complete	0.25	2	1	0.25	1	1	NormalTermination	Training

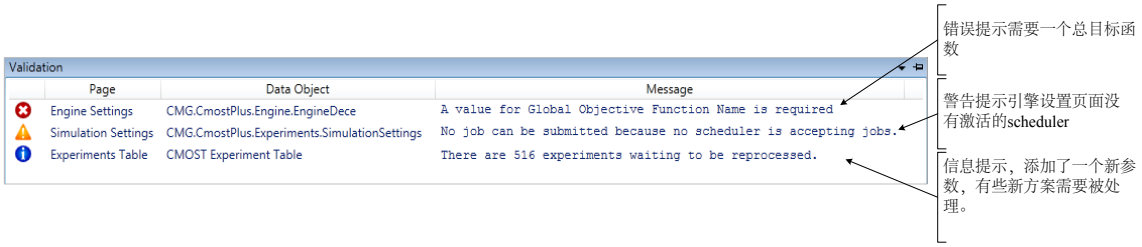
4. 闭分组的行。在下面的例子中，我们仅仅打开的实验方案行为**ModKH1**=*0.25*和**ModKH13**=*0.25*：



		ID	Generator	Status	ModKH1	ModKH13	ModKH18	ModKH20	ModKH4	ModKH6	Result Status	Proxy Role	Kee
	0.25												
	0.25												
3		5	BoxB	Complete	0.25	0.25	1	0.25	1	1	NormalTermination	Training	Aut
4		6	BoxB	Complete	0.25	0.25	1	2	1	1	NormalTermination	Training	Aut
5		56	User	Complete	0.25	0.25	1	0.25	0.25	0.25	NormalTermination	Verification	Aut
	1												
	2												
	1												
	2												

4.7.10 Validation 选项

Validation 选项提供了典型错误和警告的详细信息，如下所示：

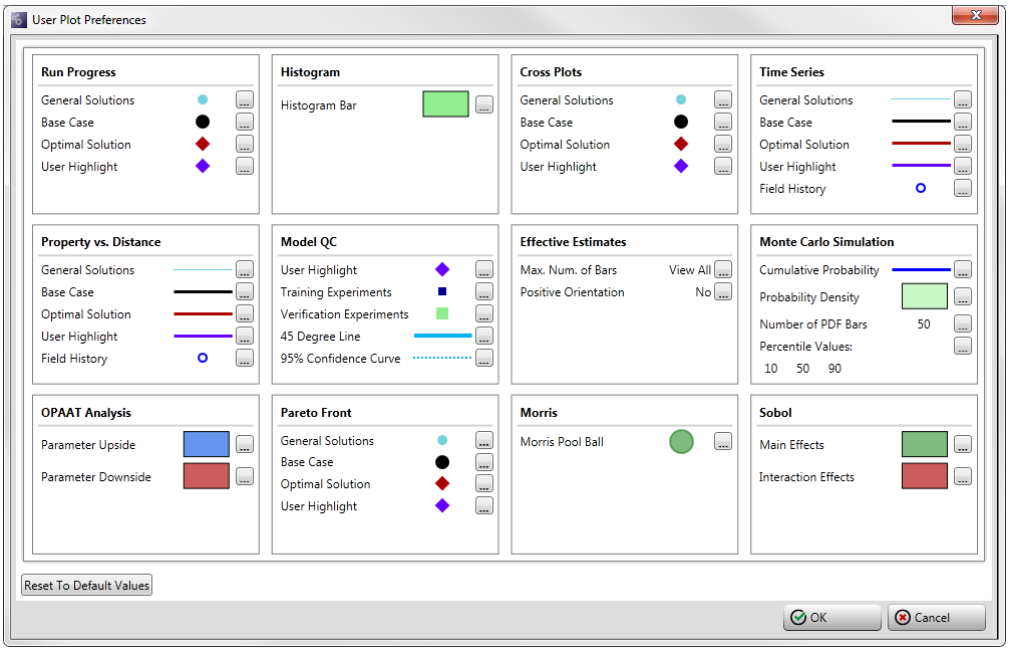


注意：在CMOST引擎运行前，必须解决错误，警告可以忽略。

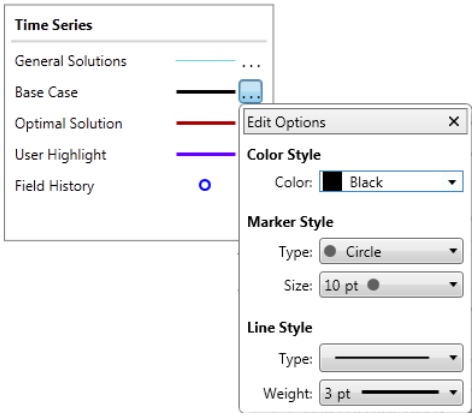
4.8 工具

4.8.1 画图偏好设置

点击**Tools | Plotting Preferences**打开 **User Plot Preferences** 对话框，通过此对话框设置图形样式、曲线颜色、数据点以及其他选项。这些设置可以应用到用户自己所有Project中：



在相关联的区域，点击想要自定义参数旁边的 ，出现对话框**Edit Options**，例如：

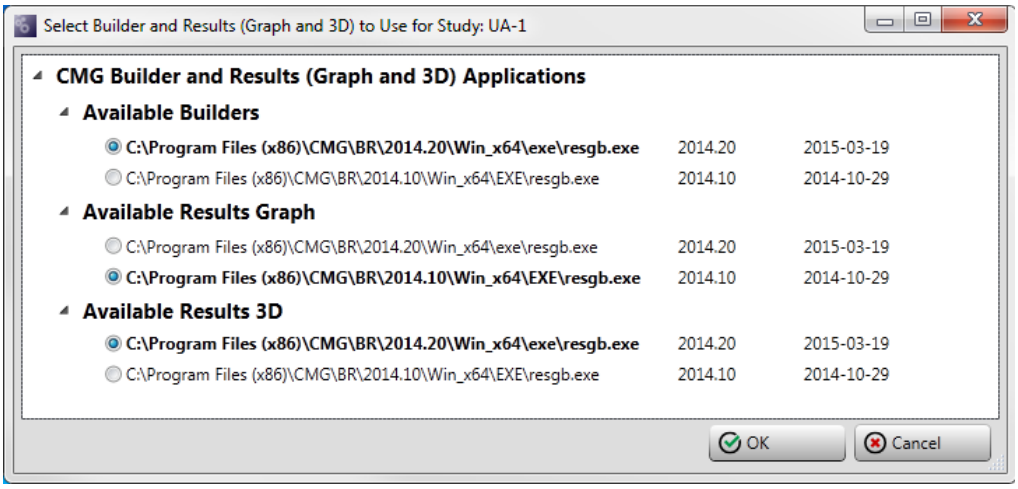


按照要求对参数进行配置，然后点击OK。在本手册中，所有曲线的设置都使用缺省值。

注意：在对话框**User Plot Preferences**左下角，点击**Reset to Default Values** 按钮来恢复所有缺省值。

4.8.2 Builder、Graph 和3D 版本偏好

如果用户计算机安装了多个版本的Builder、 Results Graph 和Results 3D，可以通过菜单栏**Tools | Builder, Graph & 3D Versions**，选择想在CMOST中使用的版本。这或许是有用的，因为最新的版本或许没有某一特殊的功能。



4.8.3 更新CMOST分析结果

当参数和目标函数改变时，通过 **Update Results and Analyses** 工具，可以将这些改变应用到CMOST节点、表格等结果：

1. 如果改变了参数名字（**Parameters**表格）或目标函数，然后点击**Tools | Update Results & Analysis**，CMOST将更新所有的CMOST节点参数及目标函数名称，包括**Results & Analyses** 节点以及所有出现参数及目标函数名称的节点。即使在CMOST中显示的参数名称和CMM文件中不一致，这个工具也可以使用。

2. 如果添加了一个参数，首先需要解决检测到试验*reuse pending*状态（详细信息参考*Resolve Reuse Pending*）。解决完reuse pending后，点击 **Tools |Update Results & Analysis**。新的一列和子节点将会添加至表格。

3. 如果修改了一个参数或目标函数的属性值，你需要解决*reuse pending*，但不需要使用**Tools |Update Results & Analysis**.

4.9 退出CMOST

1. 任何时间都可以点击 **Save**来保存CMOST文件。
2. 如果想退出CMOST，点击主页面右上角 Close  按钮或在菜单栏点击 **File | Exit** 。会被询问是否保存当前的文件，点击 **Yes**。

5 创建和编辑输入数据 (Creating and Editing Input Data)

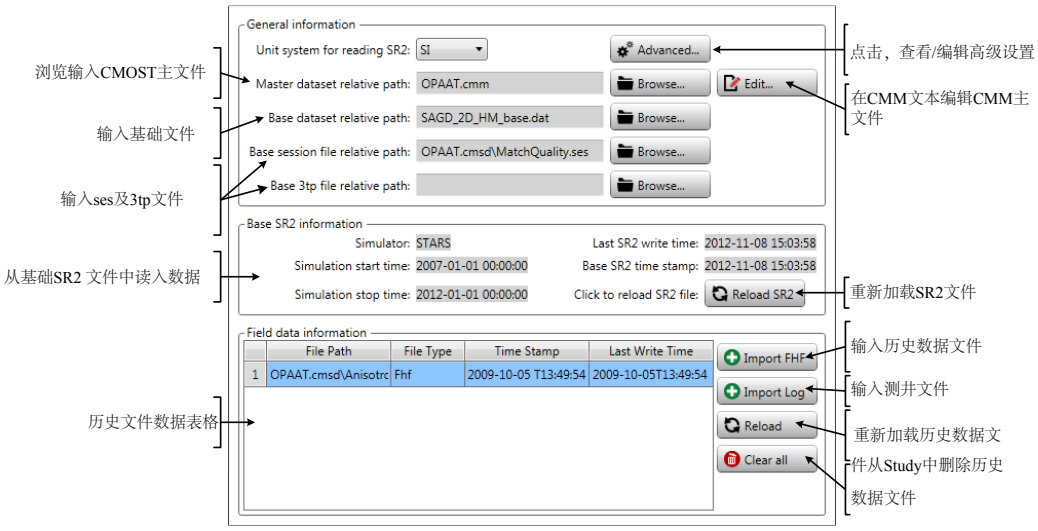
5.1 简介

该部分介绍了如何利用CMOST界面来创建和编辑输入数据，为CMOST运行做准备。

5.2 常规属性

通过 General Properties 界面，可以输入Project中用到的常用数据及文件。在所有类型的Study中都需要填写该界面。

General Properties界面如下：



5.2.1 常规信息部分

- **Unit system for reading SR2**：定义CMOST输出SR2数据单位制，它不会影响模拟器输入和输出单位制，因此CMOST单位制仅仅影响Study中定义的目标函数及observers。例如，如果选择的单位制是国际单位SI，那么产油速率将是 m3/day ，原油价格NPV 是$/m3。同样，如果单位制是矿场单位Field，原油价格将是 $/bbl。

- **Master dataset relative path**：当创建新的Study或选择某个已经存在的主文件时会自动创建主文件。Study主文件是在基础文件上修改而来的，该文件指导CMOST输入参数值到数据文件。**Master dataset relative path**可以手动或通过**Browse**浏览输入。如果该文件不在Study文件夹内，CMOST会将文件自动复制到文件夹内，主文件是CMOST必要组成部分。

  在CMM Editor 使用Edit按钮打开主文件来进行编辑。参考 Master Dataset 部分，了解更多关于主文件的信息。

- **Base dataset relative path**：当创建Study时，需要输入基础文件。基础文件是CMOST必要组成部分。**Base dataset relative path**可以手动输入或者通过**Browse**浏览输入。参考 Base Dataset部分，了解更多关于主文件的信息。

注意：修改基础文件不会反映到主文件，每次都必须单独编辑。

- **Base session file relative path**：基础ses文件和3tp文件是非必须组成部分，这些文件对分析结果非常有用，因此它们可以作为展示Results Graph 和 Results 3D结果的依据 。详细信息查看 Base Session File 部分。

5.2.2 基础SR2信息部分

该部分的信息只能用来阅读，是由CMOST基础文件提供，主要包括使用的模拟器类型，模型运算起始和终止时间，还包括了模拟器运行终止时间点等。

当Study文件创建完成，若需要修改主文件，而SR2文件没有变化，如果是这样的话，就需要重新加载新的SR2文件。

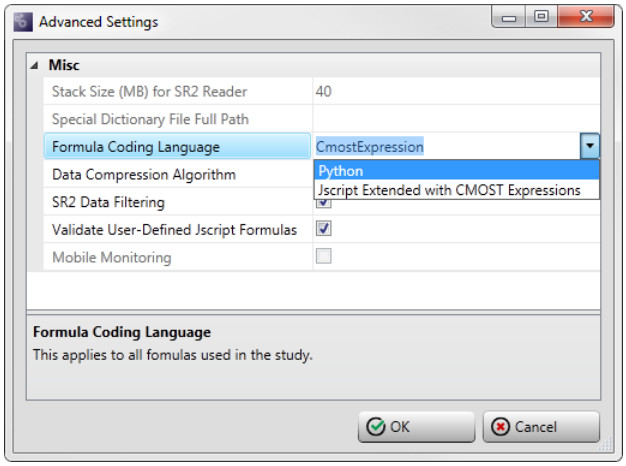
例如，Study文件创建完成，需要在主文件中添加一口井，如果用户想让CMOST使用带有新井的结果，就需要将新SR2文件重新加载。那么就要重新运行添加完新井的模型，然后点击**Reload SR2**按钮，将SR2文件重新加载。

5.2.3 矿场数据信息部分

该部分，可以将生产历史数据及测井文件输入到Study文件夹。通常在做历史拟合时，会经常用到，在做敏感性分析时，偶尔也会用到。 如果这些文件做了修改，可以重新加载，另外还可以从Study中删除这些文件。

5.2.4 高级设置

点击**Advanced** 按钮，打开**Advanced Settings** 对话框：



注意：在**Advanced Settings**表格中显示了所有设置的信息，如上面的例子所示，当创建新Study后，就会自动缺省这些设置。

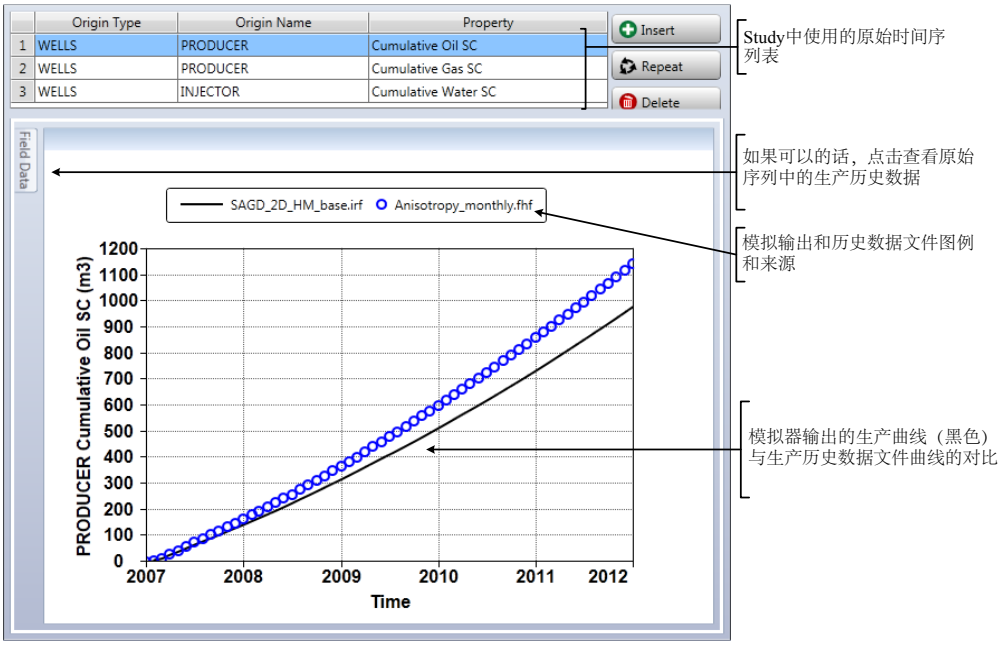
- **Stack Size (MB) for SR2 Reader**：SR2阅读器利用堆栈大小读取SR2结果文件。默认值是40MB。
- **Special Dictionary File Full Path**：如果需要特殊的处理器读取SR2文件时，需要定义该选项，否则保留空白选项。
- **Formula Coding Language**：该选项应用于Study中的所有公式。JScript 和 Python是当前CMOST版本支持的两种语言。这些代码语言用于编写公式。
- **Data Compression Algorithm**：无论是压缩和非压缩，CMOST利用数据压缩算法来反对序列化和反序列化。
- **SR2 Data Filtering**：选择该选项时，对SR2数据进行过滤，减少数据冗余。
- **Validate User-Defined JScript Formulas**：选择该选项时，当引擎启动后，CMOST将验证用户定义的脚本公式。若有错误出现，引擎将会停止。

5.3 基础数据

通过 **Fundamental Data** 界面，定义模拟器输出的系列数据。

5.3.1 原始时间序列

Original time series 是直接从SR2文件中产生的时间系列数据。在Study中，输入原始时间序列，选择**Fundamental Data | Original Time Series**节点。**Original Time Series** 界面如下所示：



- 原始时间序列表

  - 添加原始时间序列表，点击 **Insert**按钮插入一行，然后在下拉菜单中选择 **Origin Type**，**Origin Name** 和**Property**。如果表格中已经有了一行，点击 **Insert**后会在下面再添加一行。

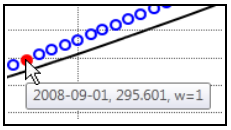
  - 了删除原始时间序列曲线，选中一行，点击 **Delete** 删除该行。可以利用SHIFT和CTRL键删除多行。当提示是否删除时，点击**Yes** ，表示删除，点击**No**，表示不删除。

  - 为了重复某一行，首先选中该行，然后点击**Repeat**。只有在原始类型下有多个原始名字时才可以使用该选项。通常会提示同样的类型并且拥有同样的属性参数，不是表格中出现的属性参数 。

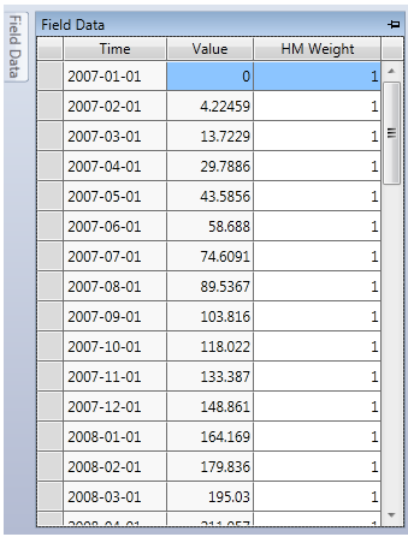
- **基础方案和矿场数据曲线:**

  -于选择的原始时间序列曲线，对比模拟器输出的结果曲线（基于SR2文件）与生产历史数据曲线。这张图表示了模拟器模拟结果与生产历史数据之间的误差。

  -如果将鼠标移动到生产历史数据某一数据点，该数据点就会变成红色，并且会显示数值以及权重因子，如下图所示:



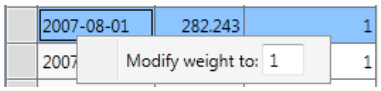
- **Field Data tab**：点击 **Field Data**键 ，如下所示:



如上所示，上面的表格中，每个数据点包含**HM Weight**，默认值都是1。**HM Weight**用于计算历史拟合误差（History Match Error.）。如果在Field Data 表格中选中一个点，图中就会高亮显示该点。如果降低该点的 **HM Weight** ，那么图中该点也会相应减小。这对于历史数据提供了更加直观的视觉效果。

通过**Field Data** 表格，使用以下方法可以对所有历史数据点编辑**HM Weight**:

- 在表格中，直接修改权重因子下面的数值即可。
- 在表格中，选中一行或多行，然后点击右键，会出现提示修改 **Modify weight** 对话框:



输入**HM Weight to**，然后点击**Enter**。表格和图中的**HM Weight** 将会更新。

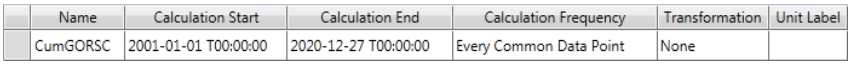
5.3.2 用户自定义时间序列

通过 **User-Defined Time Series** 界面，可以自定义时间序列，该序列不是从SR2文件直接得到，而是来源于SR2数据文件。

为了说明该问题，参考以下例子，例子中定义了累积GORSC 时间序列：：

1. 选择 **Input | Fundamental Data | User-Defined Time Series**.
2. 在**User-Defined Time Series** 表格顶部，点击 

插入一行。
3. 选择新的一行，然后定义新的时间序列，如下所示：



Name	Calculation Start	Calculation End	Calculation Frequency	Transformation	Unit Label
CumGORSC	2001-01-01 T00:00:00	2020-12-27 T00:00:00	Every Common Data Point	None	

- **Name**：定义新时间序列的名称，上面的例子中新时间序列名称为CumGORSC。
- **Calculation Start, Calculation End**：定义新时间序列起始和终止时间。
- **Calculation Frequency**：想要在时间序列中定义的时间，如下：

Every Common Data Point	时间序列中想要计算的时间点可以是 **Calculation Start** 和 **Calculation End**之间任一时间点
Every Minute, Every Hour, and so on	时间序列中想要计算的时间点可以是 **Calculation Start** 和 **Calculation End**之间任一时间点。因此原始时间序列在所有的时间点可能不可用，计算时可能要依据 **Transformation**的设置。

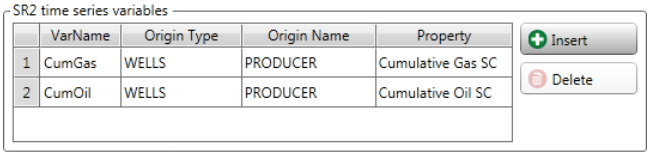
- **Transformation**如果所有要求的数据点不可用，则需要通过以下方法转换后得到：

None	如果日期数据是有效的，那么仅仅需要计算时间序列数据。

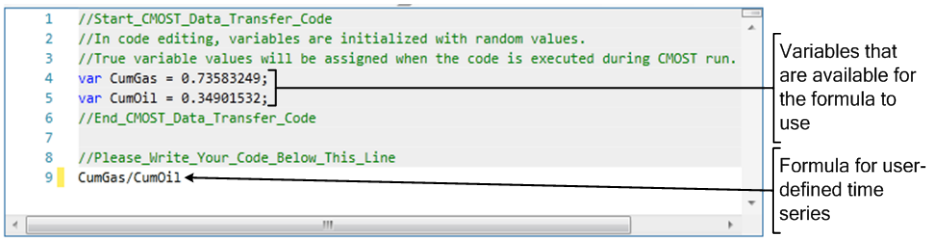
Numerical Integration	用户自定义的时间序列数据利用数值积分法来决定，在这种情况下，需要定义数值积分法选项**Numerical Integration Option**（*Backward Rectangle*、*Forward Rectangle*或 *Trapezoidal Rule* 种的一种）和时间间隔单位**Time Interval Unit**。
Numerical Differentiation	用户自定义的时间序列数据利用数值微分法，在这种情况下，需要定义数值微分选项**Numerical Integration Option** （*Backward Difference*、*Forward Difference*或*Central Difference*）和 **Time Interval Unit**.
Moving Average	用户自定义的时间序列数据利用移动平均法，在这种情况下，需要定义**Moving Average Window**。

- 单位标签（**Unit Label**）：用户定义的时间序列单位。这些单位显示在Base Case and Field Data Plot Preview。在我们的例子中，单位是ft3/bbl。

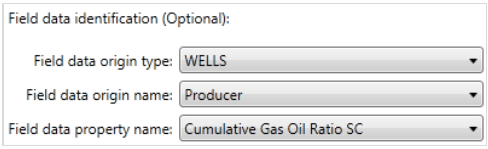
4. 通过定义原始时间序列来计算用户自定义的时间序列，通过下拉菜单选择**Origin Type**、 **Origin Name**以及 **Property**， 在**VarName**下输入自定义时间序列名称。该名称可以用于公式中的变量。例如，我们插入了两个时间序列， *Cumulative Gas SC* (变量名称- *CumGas)* 和 *Cumulative Oil SC* (变量名称 *CumOil*)，如下所示：



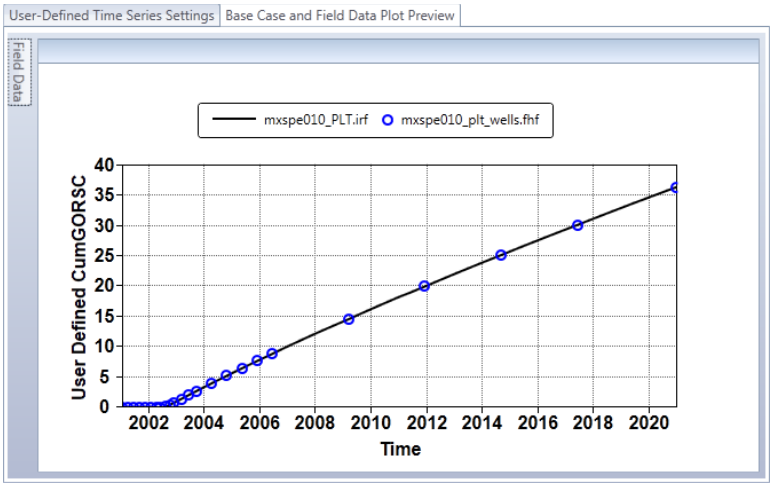
5. 在公式编辑界面，为用户自定义的时间序列输入公式。参考 Formula Editor，我们的例子如下所示：



6. 利用下拉菜单定义与用户自定义时间序列相关的历史数据**Origin Type**，**Origin Name**和**Property Name**，如下例所示：



7. 为了查看用户自定义的时间序列与基础方案及历史数据之间的误差，点击**Base Case** 和 **Field Data Plot Preview**键，如下：



该图表示用户自定义的时间序列，SR2文件数据与历史数据之间的对比。

8. 如果点击**Field Data** 键，打开历史数据表格，与用户自定义的时间序列进行对比：

Time	Value	HM Weight
2001-01-26	0.00145017	1
2001-04-11	0.00146581	1
2001-06-25	0.00146768	1
2001-09-08	0.00146959	1
2001-11-22	0.001471	1
2002-02-05	0.00147269	1
2002-04-21	0.00147371	1
2002-06-10	0.00147436	1
2002-08-24	0.1515736	1
2002-10-13	0.4275312	1
2002-12-02	0.7397569	1
2003-03-12	1.387368	1
2003-06-20	2.045953	1
2003-09-28	2.695072	1

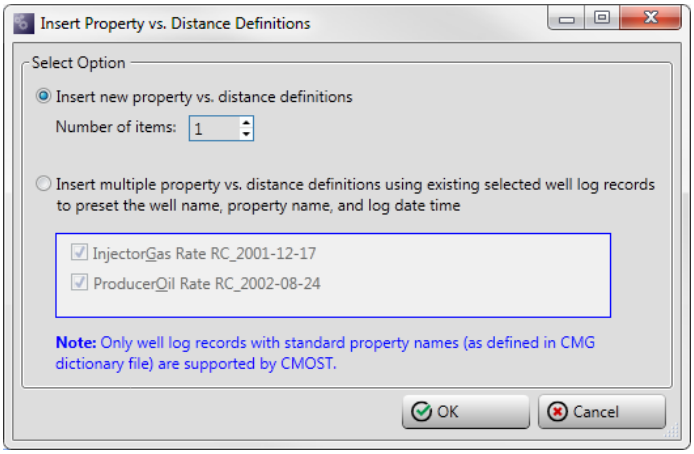
通过这个表格，可以为每一个数据点设置HM Weight （在0~1之间），用于计算历史拟合误差，详细描述在 History Match Error。

5.3.3 属性vs.距离序列

属性vs. 距离的时间序列可用于计算历史拟合误差。属性vs. 距离的时间序列可以通过SR2文件和测井文件反演得到。计算出模拟器模拟值与生产历史数据之间的误差。

创建属性vs. 距离序列：

1. 点击**Fundamental Data | Property vs. Distance Series**.
2. 在**Property vs. Distance Series** 界面，点击 Insert 插入一个或多个属性vs.距离，**Insert Property vs. Distance Definitions**对话框如下所示：



3. 选择必要的选项，其中：

   - **Insert new property vs. distance definitions**选择想要插入的number of items，然后点击 OK。每行都有缺省值。需要输入井名称和属性参数，然后根据需要调整其他设置。

   -如果选择**Insert multiple property vs. distance definitions using existing selected well log records…**，显示测井数据。对话框注意部分提示，CMOST仅支持标准的测井数据（按照CMG格式定义的测井数据）。选择测井数据，然后点击OK。

4. 在表格中配置如下：

   - **Name**为property vs. distance数据输入名称。例如，GasRateRC_2001_12_17，它是由属性参数及日期合并而来。

- **Well Name**：在wells 的下拉菜单中，选择一口井。

- **Property**：选择属性参数。不是所有下拉菜单中的属性都有和距离之间的关系。如果该数据不存在，将不会陈列在图中。

- **Log Date Time**：指定想要定义property vs. distance series的时间。再次确认某一特定的时间点是可用的。

- **Data Path**：从Well log，Along linear path，Along well path或Along trajectory这些方法中指定从SR2得到property vs. distance 数据的方法。如果选择Along trajectory，则 必须定义轨迹的来源。点击Trajectory按钮，查看 To import a trajectory file 关于如何导入井轨迹的信息。如果没有可用的井轨迹文件，将会显示缺省的选项Well path 。

- **TVD or MD**：定义TVD (垂直距离) 或 MD (测深) 作为距离坐标。

- **Use Block Center**：表示是否只在网格中心读取空间属性信息。如果没有选择Use Block Center，那么读取网格进和出对应点的空间属性信息。

- **Use Accumulation Flow**：指定沿着井眼轨迹从最深处向上的流体体积是否累积计算。

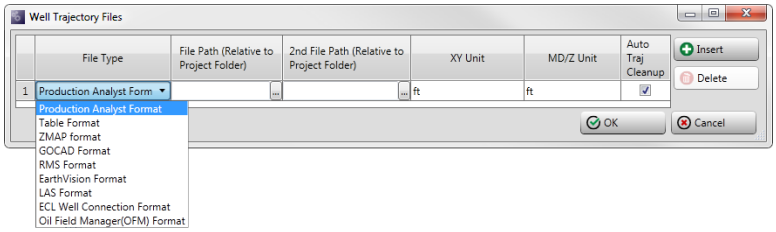
- **Use Normalized Flow**：指定沿着井眼轨迹从最深处向上的流体体积是否累积，然后进行归一化。

- **Smoothing Method**选择平滑property vs. distance 数据的方法，*one of None*、*Weighted Moving Average*、*Aitken Interpolation*，*Akima Interpolation*或者 *Cubic Spline*。

导入井轨迹文件：

1. 点击 Trajectory 按钮，出现**Well Trajectory Files**对话框。

2. 点击**Insert**，在表格中插入一行。



3. 输入井轨迹数据，如下所示，然后点击OK。

- **File Type**：在下拉菜单中选择井轨迹文件类型。
- **File Path (Relative to Project Folder)**：输入文件或路径和井轨迹文件名称。

- **2nd File Path (Relative to Project Folder)**：井轨迹文件有两种形式，XY文件和Deviated文件。在空白处输入全路径或相对路径。

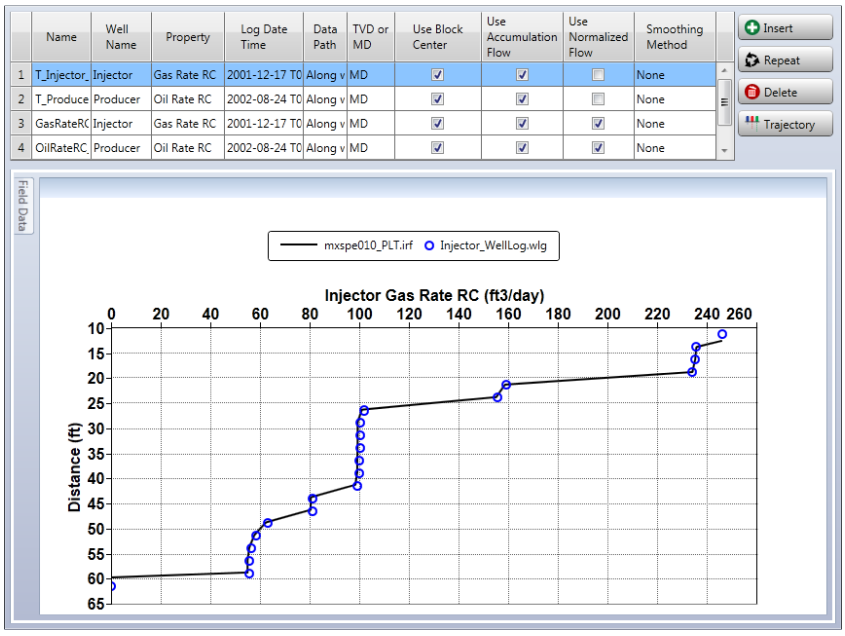
- **File Type**：选择轨迹文件类型。

- **XY Unit**：选择轨迹文件中使用的单位来定义x和 y 坐标。

- **MD/Z Unit**：选择轨迹文件中使用的单位来定义MD（测深）和z坐标。

- **Auto Traj Cleanup**：当出现保留背离数据时，选择删除多余的井轨迹节点。

下面展示了关于**Property vs. Distance Series**界面的例子，如下：



5.3.4 流体界面序列

如果SR2包含流体饱和度数据，CMOST能够根据井的位置来计算gas-oil、water-oil和water-gas 流体界面。

为了建立流体界面时间序列：

1. 选择**Fundamental Data | Fluid Contact Depth Series** 界面。**Fluid Contact Property**表格包括用户选择的流体界面类型：

	Calculate	Fluid Contact Property	Saturation Property	Porosity Type	Method	N Smoothing Points	Threshold	First/Last	MD/TVD	Data Path
1	☐	Well Gas Oil Contact Depth		Matrix	Max Derivatives over Moving Average	3	0	First Occurrence	MD	Along Perforation
2	☐	Well Water Oil Contact Depth		Matrix	Max Derivatives over Moving Average	3	0	First Occurrence	MD	Along Perforation
3	☐	Well Water Gas Contact Depth		Matrix	Max Derivatives over Moving Average	3	0	First Occurrence	MD	Along Perforation

2. 在 Calculate 列，选择流体界面参数或想要计算的属性参数。

3. 设置的流体界面属性参数，如下：

   - **Calculate**：选择该选项，CMOST将计算井所在位置的界面。

   - **Fluid Contact Property**：仅仅能查看流体界面的名字，用于预览和观察图形。

   - **Saturation Property**定义饱和度类型，包括*SG*、*SO*和*SW*。利用饱和度类型，再结合一定的算法来确定沿着井的长度是否存在过渡带。

   - **Porosity Type**：定义孔隙度类型，*Matrix* 或者*Fracture*。

   - **Method**选择计算方法，*one of Predefined Threshold*，*Max Derivatives over Moving Average*或者 *Max Saturation Difference on Average*。对于这些计算方法的详细描述，可以参考*Results Graph User Guide*中的*Using Results Graph | Working with Curves | Adding Curves | To create a fluid contact depth parameter*部分。

   - **N Smoothing Points**这个参数用于计算*Max Derivatives over Moving Average*和 *Max Saturation Difference on Average*。

   - **Threshold**：这个参数用于计算*Predefined Threshold*。可以在0和1之间进行取值。

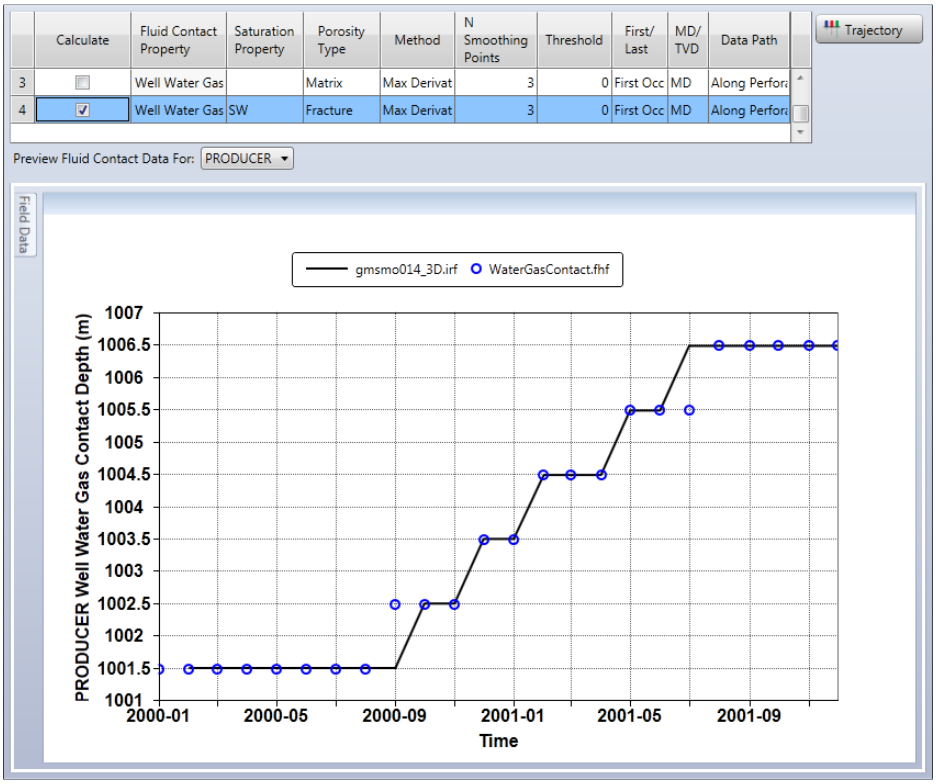
   - **First/Last**：如果该空白处设置为First Occurrence，那么符合标准的第一个点作为界面深度。如果该空白处设置为Last Occurrence，那么符合标准的最后一个点作为界面深度。

   - **MD/TVD**：选择垂深 (TVD) 或测深 (MD)。

   - **Data Path**：通过网格定义路径，其中：
   Along Perforation：如果数据文件包括井射孔信息，可以使用该选项。如果井 LAYERXYZ数据可用，可以通过射孔进网格和出网格来定义井路径。如果井LAYERXYZ不可用，井路径是通过射孔网格中心连接而成。距离与第一个射孔点的深度有关。

Along Trajectory：如果井轨迹文件可用的话，可以使用该选项。点击 按钮打开井轨迹对话框，输入该文件。更多信息请参考 To import a well trajectory file 。

4. 在**Preview Fluid Contact Data For**旁边，选择想要查看流体界面的井，例如：

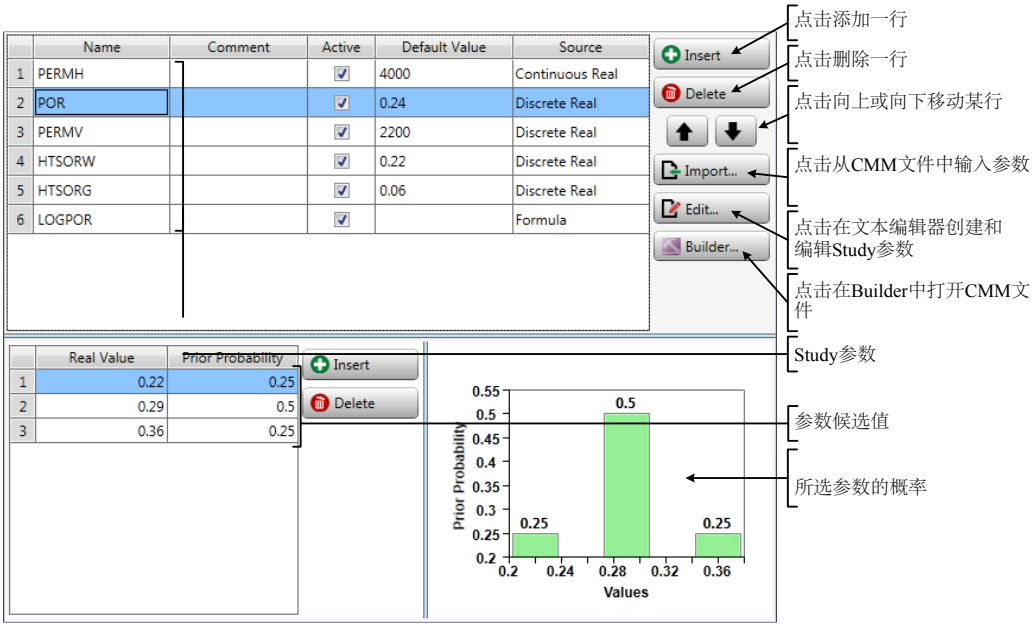


如上所示，预测的流体界面数据与历史数据的对比。如果你点击左面**Field Data** 键，可以为每个数据点调节**HM Weight** 。

5.4 参数化

5.4.1 参数

通过Parameters界面，如下图，输入参数并且定义它们的属性。首先将参数输入到主文件，然后输入到CMOST（参考 To import a parameter from the master dataset）；在特殊情况下，可以通过中间参数来定义参数，这些中间参数不会输入到CMM文件（参考To add an intermediate parameter ）。



注意：关于利用Builder打开和编辑CMM文件的更多信息，请参考Builder用户手册 “Setting up CMOST Master Datasets” 章节。

5.4.1.1 添加新参数

从主文件输入一个参数：

1. 利用CMM文本编辑器（CMM File Editor）输入参数及其缺省值。在CMOST查看Names 更多信息。
2. 保存CMM文件。
3. 点击Import  按钮，将参数从主文件导入**Parameters**表格。如果主文件较大的话，CMOST读取并找到所有的参数可能会需要几分钟的时间。更多信息参考Importing Parameters from the Master Dataset。

   在**Parameters**表格中显示输入的参数名称及缺省值，与主文件中的一致。仅仅通过主文件来修改这些参数。

4. 在**Parameters**表格编辑参数：
   - **Name**：从主文件输入参数名称。不需要改变参数名称。
   - **Comment**若有必要，该选项可记录了关于设置、假设及基本原理等内容。
   - **Active**：当从主文件导入参数时，默认选择Active选项，选项Active

决定了CMOST是否给参数指定候选值。如果选择Active选项，CMOST会指定候选值到参数。如果不选择Active选项，CMOST会指定缺省值到参数。

- **Default Value**：参数的缺省值是从主文件中得到的。缺省值应该是输入到基础文件中的原始值。只有当没有选择该选项时才仅仅使用缺省值。

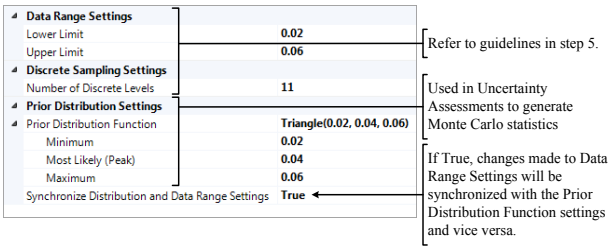
如果想编辑缺省值，点击Edit 

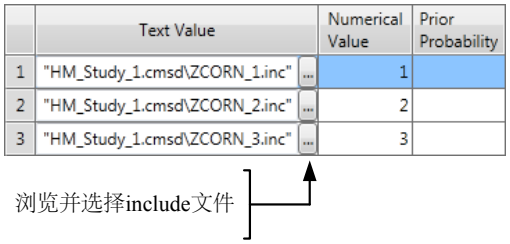
按钮，在主文件中编辑缺省值，然后导入新参数的缺省值。如果CMOST使用了较为复杂的公式，缺省值可能不等于原始值，例如，如果将下述公式输入到主文件：

<cmost>this[1]=LOG10(Keq)</cmost>缺省值1= LOG10(Keq)。

- **Source**：该列告诉CMOST如何为参数指定值。

  Source可以是下面的一种：

Continuous Real	如果是这种情况，需要在**Candidate Values**页面左下方，配置以下信息：  如果做了任何修改，都会反映在先前概率分布函数上面。
Discrete Real	如果是这种情况，将输入离散候选值的表格，并且为每个离散值定义概率。查看下面的注意项。
Discrete Integer	需要输入整数候选值的表格并且为每个整数值定义概率。查看下面的注意项。

Discrete Text	需要输入文本文件的表格，每个文本都是唯一的数值，并且为每一个文本文件定义概率。查看下面的注意项。一个文本选择按钮来帮助浏览并选择include文件： 
Formula	利用Formula Editor公式编辑器为变量输入公式。

注意：先验概率分布用来产生不确定性分析的数值，例如，会根据这个分布产生不确定分析需要的参数值。如果输入了先验概率值，则在页面右下角产生概率分布图。

考虑好使用的来源类型后，接着考虑下面的选项：

Continuous or Discrete Real	所使用的参数可以取带有小数的数值，例如孔隙度或渗透率。这是缺省的来源类型。
Discrete Integer	所使用的参数不可以取带有小数的数值，例如网格坐标或岩石类型。
Discrete Text	所使用的参数是文本文件或include文件，这些取值通常在双引号之内（“OPEN”或“ZCORN_1.inc”）。对任何参数都自动指定 Discrete Text 类型，其缺省值在主文件内用双引号标示。
Formula	可以为参数输入公式，会出现编辑公式的界面 。其他某些参数也会出现在公式中。 更多信息请参考Formula Editor 。

5. 在使用离散、整数和文本文件的情况下，**Parameters** 页面左下角会出现**Candidate Values** 表格。通过**Candidate Values**表格输入想要CMOST交给主文件的参数值。

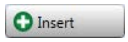
选择的参数候选值依赖Study类型，概述如下：

Sensitivity Analysis	对敏感性来说，其主要影响是线性影响，因为每个参数只有两个取值。也可以考虑交互影响和非线性影响，但每个参数至少需要三个取值。所有的取值都应该在合理的取值范围内，并且这些取值都是不同的。
History Matching and Optimization	对于历史拟合和优化来说，**Candidate Values** 可以输入无限多个候选值，或者输入一个取值范围（最小和最大值），但对优化来说，这将会花费更多的时间找到最优解。
Uncertainty Assessment	对不确定性分析来说，如果要考虑交互作用和非线性（二次方）的影响，那么每个参数需要三个候选值。其中低值应该在参数取值范围下限边缘，高值应该在参数取值范围上限边缘，中间值应该在参数取值范围的中间附近 。

注意：如果参数类型是离散的文本，那么需要相应的数值添加至候选值。 如果某个数值和该文本值相匹配，应该输入该数值。

添加中间参数：

注意：从 CMOST 2013开始， 参数必须 Active 后，用于中间参数。

1. 在**Parameters** 表格点击**Insert** 

输入一行。
2. 输入中间参数的名字，选择**Source** 为*Formula*。
3. 为中间参数定义公式。
4. 为参数或依赖中间变量的参数定义公式。

以此说明，假定在主文件定义了*Parameter_A* 和 *Parameter_B*。让我们假定参数*Parameter_B* 是中间变量C的函数，如下：

$$Parameter\_B = Intermediate\_Parameter\_C^2$$

而*Intermediate_Parameter_C* 是*Parameter_A*的函数，如下：

$$Intermediate\_Parameter\_C = 4 * \log$$

($Parameter\_A$) 上面的公式在Parameters 表中如下：

	Name	Comment	Active	Default Value	Source
1	Parameter_A		☑	0.24	Discrete Real
2	Intermediate_Parameter_C		☑		Formula
3	Parameter_B		☑		Formula

其中*Intermediate_Parameter_C* 通过公式编辑器编辑：

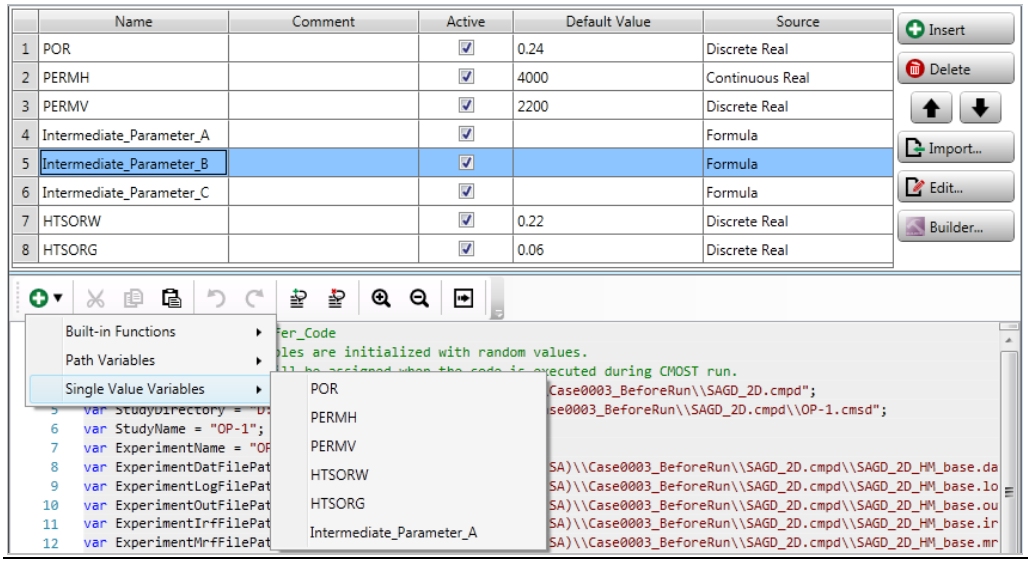
```
17    4*LOG10(Parameter_A)
```

Parameter_B 通过公式编辑器编辑

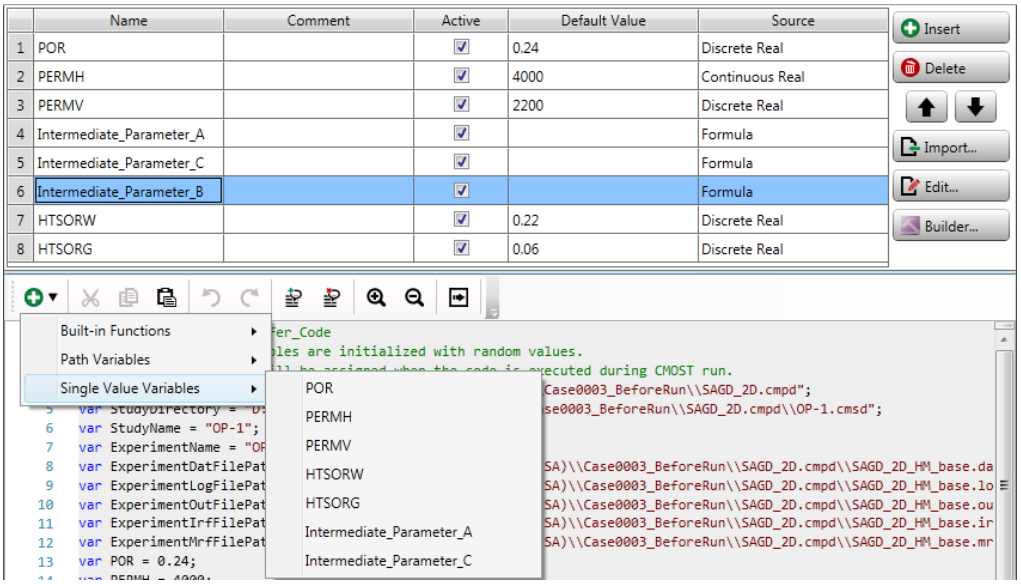
```
18    POWER(Intermediate_Parameter_C,2)
```

参数之间的关系越复杂，越能体会该特征的优势。

注意：当为中间参数定义公式时，只有上面出现的中间变量才可用于公式。在下面的例子中，*Intermediate_Parameter_A* 可用于公式*Intermediate_Parameter_B*中，但是 *Intermediate_Parameter_C*不可用



如果参数Intermediate_Parameter_C 移到Intermediate_Parameter_B上面，它可以用于参数Intermediate_Parameter_B 公式中，如下所示：



5.4.1.2 先验概率分布函数

先验概率分布函数仅仅作用于连续实数。它表示了所有参数可能出现的数值概率。CMOST里面有以下几种不同的分布类型：

- 未指定
- 确定性分布
- 均匀分布
- 三角分布
- 正太分布
- 对数正太分布
- 自定义

每个参数都需要填充所有先验概率分布函数配置项。更多信息关于每个先验概率分布函数类型查看 Probability Distribution Functions 部分。

5.4.1.3 删除参数

在Parameters 表格，选中某一行，然后点击 Delete  按钮，如果删除某个参数，那么对应试验表格中的列也将会删除，最后需要手动删除CMM文件中的参数。

可以从表格中删除多个参数，点击参数名称左边的空白处，然后向上或向下移动鼠标。也可以通过SHIFT和 CTRL 来删除多个参数。Delete  按钮或 DELETE 键可以用来删除行。

5.4.1.4 在表格中移动参数

在Parameters表格中，为了移动参数所在的行，在表中选择想要移动的行，点击Up 或 Down按钮。参数可以任意顺序排列。

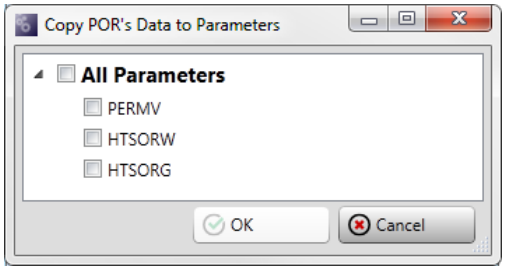
注意：点击表头按照升序或降序进行排序

5.4.1.5 复制参数数据

复制一个参数的数据到另外一个参数，如下：

1. 在参数**Parameters** 表格，右键选择想要复制的参数。

2. 选择 **Copy Parameter Data to Other Parameters** 用来打开Copy Data to Parameters 对话框，如下面的例子所示：



如上，仅有对话框中的参数可以用来复制。

3. 选择想要复制的参数，然后点击OK。数据复制总结如下：

从.....复制	复制到.....	复制数据
连续实数	连续实数	设置数据范围，离散抽样和先验分布设置
离散实数	离散实数	实数和先验分布概率

从.....复制	复制到.....	复制数据
离散整数	离散整数	整数和先验分布概率
离散文本	离散文本	文本文本，数值和先验分布概率
公式	公式	JScript /Python 编码语言

5.4.1.6 编辑主文件

通过Parameters 界面打开主文件并且查看或编辑参数。为了在CMM File Editor打开主文件， Edit 

按钮。

5.4.1.7 从主文件输入参数

CMOST可以自动复制主文件中的所有的参数到Study文件。点击**Import** 

按钮 (CMOST可能会需要几秒钟的时间来读取庞大的主文件数据)。如果主文件使用 *this[OriginalValue]* 语法，通常会将缺省值复制到文件。更多信息参考Adding New Parameters。当参数从主文件导入时，CMOST假定缺省值等于主文件中的原始值 。应该选用导入的缺省值，因为不一定总是出现这种情况。

参数来源类型也应该被选上，否则会出现错误。当导入数据时，对于*Discrete Text* 文件，CMOST会自动用引号标示其原始值。其他所有的参数设置为*Continuous Real*，因此在导入数据后，参数类型需要手动修改。

当导入一个参数时，**Active** 复选框会被自动勾选，然后需要设置Prior Probability Distribution Functions。

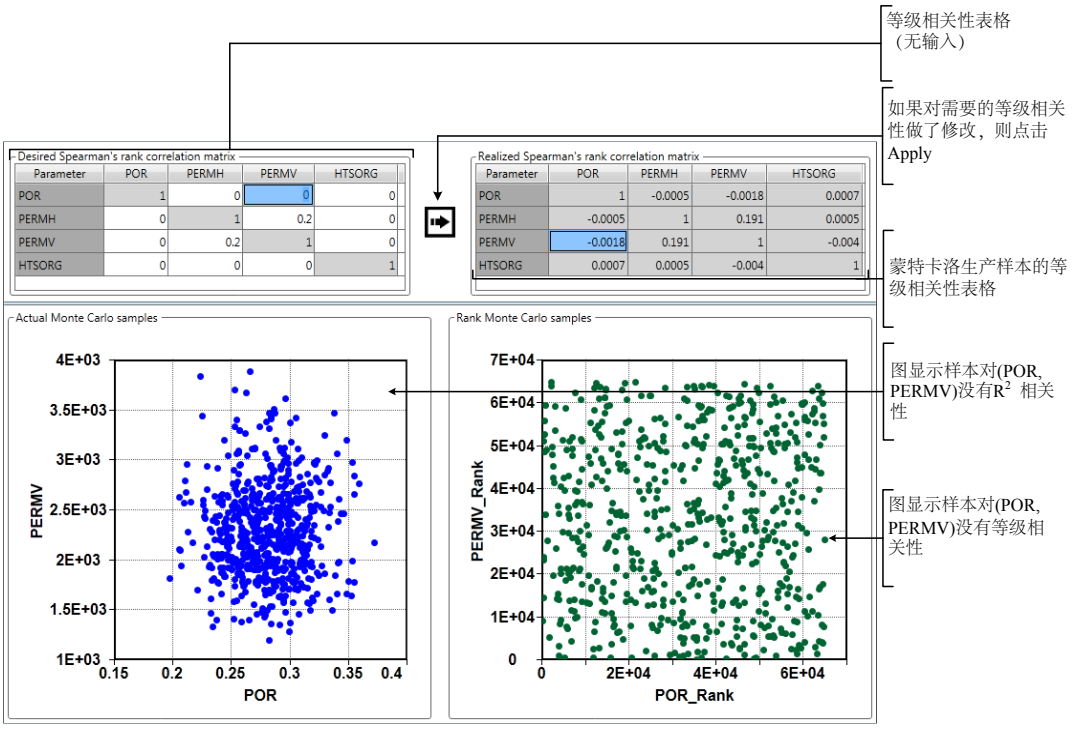
5.4.2 参数相关性

使用蒙特卡洛模拟方法，CMOST抽取的实验样本应用于不确定性分析，通过**Parameters**页面设置*Continuous Real* 参数并定义概率分布函数。当通过代理模型计算目标函数时（例如没有使用模拟器），实验方案数量是巨大的。在CMOST中，UA应用**Monte Carlo Simulation Using Proxy**实验方案数为65000。

通过**Parameter Correlations**界面，CMOST可以通过算法调整蒙特卡洛模拟生成的系列参数相关性的等级，因此它们尊重所需的等级相关设置（伊曼和科诺菲尔1982）。这就要求与等级相关的表格中所有输入的参数有先验概率分布。

更多信息，参考Parameter Correlation。

如果还没有定义等级相关数值，Parameter Correlation 界面类似下图：



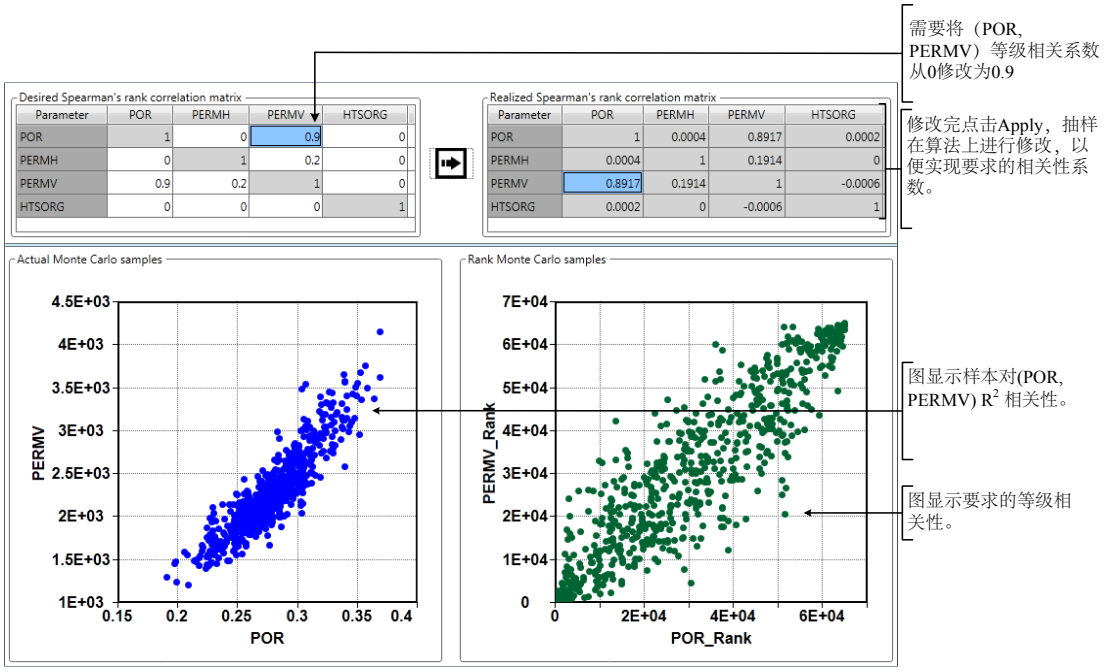
在上面的例子中：

- 在Realized Rank Correlation 表格右侧选择(POR, PERMV)空白处 。
- 目前的相关性等级与期待的相关性等级非常接近。当对期待的相关性等级做修改时，点击Apply Changes 按钮。点击该按钮后，表格中的等级相关性或相关的图形会重新刷新。

- 在我们的例子中，POR先验概率分布属于正太分布，均值是0.28，标准差是0.03。PERMV先验概率分布也属于正太分布，均值是2400，标准差是387。这些先验概率与图Actual Monte Carlo Samples一致。

- 已经被指定没有被需要的（POR，PERMV）等级相关性，这都从Rank Monte Carlo Samples图中反映出来。

如果需要定义（POR和PERMV）的相关性可根据UA取样上的算法等级。在下面的例子中详细说明：



在上面的例子：

- 按照算法调整相关等级后符合要求的相关性等级。
- 样本设置为不确定性分析做准备。

5.4.3 硬性约束条件

可以为历史拟合和优化定义硬性约束条件和软性约束条件。硬性约束条件的目的阻止运行不必要的实验方案。关于软性约束条件的更多信息参考Soft Contraints。

如果有许多实验方案需要运行，那么使用硬性约束条件或许更合适。如果违反了硬性约束条件，实验方案就不会运行，因为实验方案运行前，CMOST会根据硬性约束条件对其进行判断。

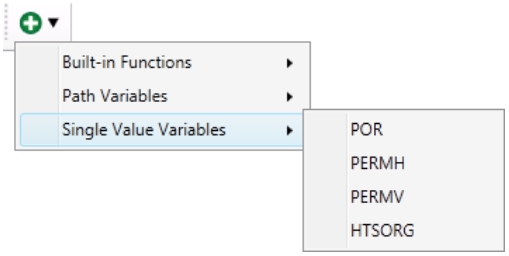
例如，如果一个SAGDProject要利用CMOST进行方案优化，事先应该知道生产井之间的距离不能小于某一数值：

$$W1\_I - W2\_I > 40$$

对于这个例子，W1_I 和 W2_I 分别表示井‘W1’ 和 ‘W2’在‘I’方向的网格坐标。上面的公式表示在W1 和W2在‘I’方向的网格至少有40个。如果遇到 ‘W1_I –W2_I <= 40’这种情况，实验方案停止运行。‘W1_I’ and ‘W2_I’作为CMOST主文件中被指定修改的参数，通过下面的步骤说明该例子：

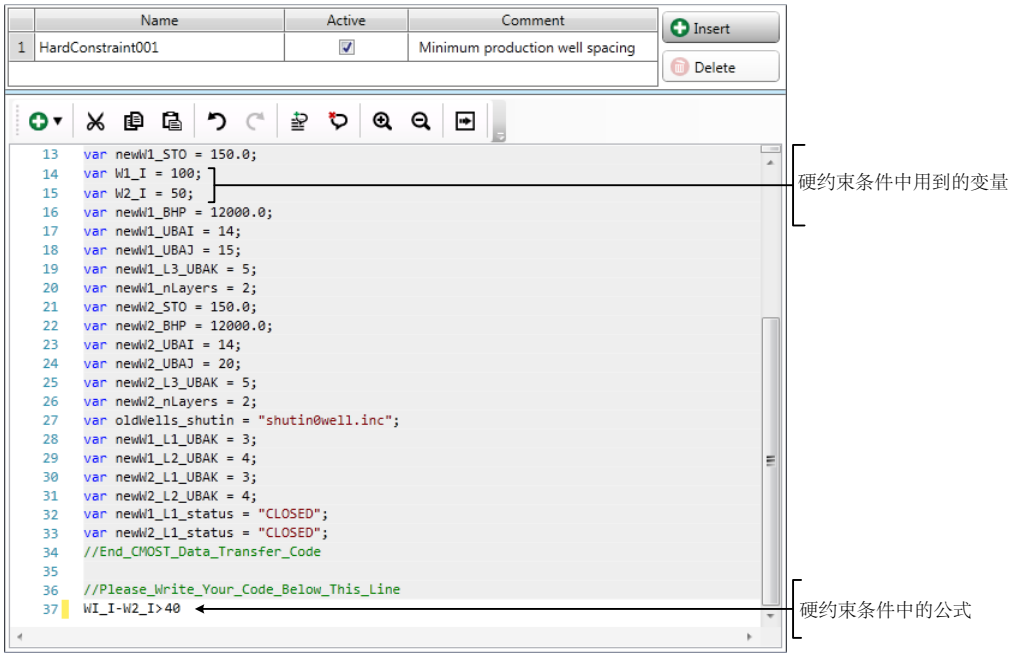
定义硬性约束条件：

1. 打开**Parameterization | Hard Constraints**界面。
2. 点击按钮，在Constraints 表格中输入一个硬性约束条件。如果需要的话，可以编辑名称及发表评论。选择Active 命令，CMOST会在每个实验方案运行前检查是否违反了硬约束条件。
3. 创建一个硬性约束条件时，会打开CMOST Formula Editor 部分。为硬性约束条件输入公式。定义一个硬约束条件可能需要不止一行。输入公式时可以使用变量公式和参数按 ，例如：



列表中的参数都是先前在主文件中定义的参数。一个替代的方法是手动输入约束公式。

例子：



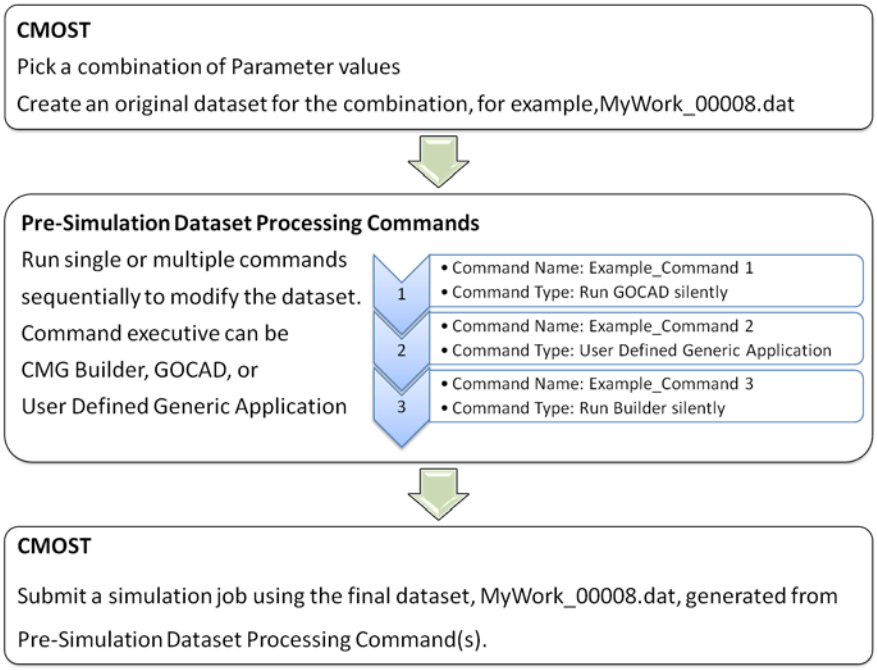
输入，*HardConstraint001*:

W1_I – W2_I > 40

必须满足这个条件，否则实验方案不能运行。

5.4.4 运行前指令

有时候，在CMOST将实验方案提交给模拟器之前，用户需要修改这些实验方案。例如，用户在做历史拟合时，可能需要调整变差函数。在这种情况下，可能需要额外的地质建模软件包，例如 GOCAD 为CMOST创建孔隙度或渗透率等地质模型。之后，CMOST使用修改过的基础模型提交模拟任务。具体过程如下表所示：



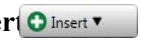
运行前命令用于修改模拟方案：

1. CMOST创建好最初的方案后，将会执行命令。
2. 在开始下个命令之前，CMOST将等待每个退出命令。
3. 执行完所有的命令，CMOST将最终的模拟方案提交运行。

关于运行前命令的相关重要信息：

- 如果相对路径用于定义可执行的路径，它应该基于Project文件的目录。
- 当CMOST执行运行时，工作目录将被设置为Study目录。
- 命令的执行顺序由**Pre-Simulation Dataset Processing Commands (Commands)**表格中命令顺序决定。

5.4.4.1 Adding a New Pre-Simulation Dataset Processing Command

点击 **Insert** 

在**Commands**表格中来插入一条新命令，可以添加三种类型的命令：**Run CMG Builder Silently**，**Run GOCAD Silently** 和 **Run User Defined Command**。

名称

需要为每个命令定义名称。如果有多个命令存在时，那么每个命令的名称必须是唯一的，并且命令的名字是区分大小写。

可执行的类型

有三种执行前命令类型：

- Run CMG Builder Silently
  运行Builder来执行在基础模型中定义的公式，或更新相渗曲线表。
- Run User Defined Command
  执行用于定义的命令，修改基础文件的来源，产生一个job名字标签。可以从CMOST或者从先前的处理命令得到产生的源文件。
- Run GOCAD Silently
  利用GOCAD脚本和工作流程（可选的）文件来执行计算。GOCAD 输出文件可用于基础文件中的include文件。

激活

如果选择**Active**选项框。CMOST会执行运行前命令，否者不会执行。

最大执行时间

CMOST将在下个命令运行前等待一个退出命令。可以为每个命令设置最大执行（等待）时间（按分钟计算）。如果一个命令在最大执行时间内没有运算完成，CMOST引擎将会停止，并且在**Control Centre** 页面**Engine Events** 表格中会出现错误提示。

5.4.4.2 在表格中移动命令

在**Pre-simulation Dataset Processing Commands**表格，为了向上或向下移动一个命令，选择想要移动的命令所在行的空格，点击  或  按钮。 命令可以是按任意顺序排列的。

注意：命令执行顺序是由Pre-simulation Dataset Processing Commands表格中的顺序决定的。

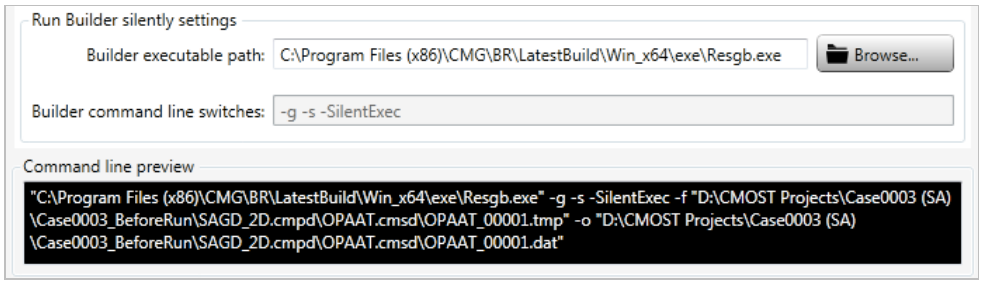
5.4.4.3 删除命令

为了删除一个命令，所有的命令所在行都必须选择。点击左边命令名字灰色空白处，该命令所在行都被选择，然后Delete按钮被激活，选择 Delete  按钮来删除该行，也可以使用DELETE键。

点击命令左边名字的空白处，利用鼠标上下拖动，可以同时删除多个命令。SHIFT 和CTRL 功能键也可以使用。Delete  按钮或DELETE键也可以用于删除命令。

5.4.4.4 配置Run Builder Silently 命令

Builder 能够执行公式计算或者更新相渗曲线数据。一旦添加该命令后，不需要再为**Run Builder Silently**添加更多的配置信息。



5.4.4.5 运用用户自定义配置

将模拟方案提交给模拟器之前，用户可根据自己的需要写程序（可执行的）修改模拟方案文件。通过输入信息，CMOST提供用户自定义的命令，例如 **Experiment Name Tagged Parameter File (.par)**，**Experiment Name Tagged Temporary Dataset File (.tmp)**，**Experiment Name Tagged Dataset File (.dat)**或**User-Specified File**。用户根据自己编写的程序来修改原始数据（**Experiment Name Tagged Dataset File**或**Experiment Name Tagged Temporary Dataset File**）来产生一个新的实验方案，然后将其提交给模拟器运算。

Experiment Name Tagged Dataset File 必须用于**Run User Defined** 命令的输出文件。

1. 在External Executable Path空白处，输入执行(.exe)文件路径，或点击 Browse... 按钮查找执行文件位置。
2. 在**Command Line Switches** 空白处，输入必要的命令行改变可执行文件。
3. 点击 Insert ▾ 然后选择文件类型，如下：
   - **Experiment-Name-Tagged Parameter File (.par)**：该文件包括制定job的参数取值，例如：

     File name: MyWork_00008.par

Porosity	*0.09*
Kv_kh_ratio	*0.25*

   - **Experiment-Name-Tagged Temporary Dataset File (.tmp)**：临时文件和JobNameTaggedDatFile 有同样的内容。例如，文件MyWork_00008.tmp 和 MyWork_00008.dat有相同的内容。
   - **Experiment-Name-Tagged Dataset File (.dat)**：对于一般命令，CMOST模拟方案可以用来作为输入或输出文件，当它用于输入文件时，它指的是CMOST产生的原始文件或者作为先前命令的修改文件。

注意：特殊指定的数据文件必须用于自定义命令的输出文件。

   - **User-Specified File**：如果用户自定义的命令需要一个输入或输出文件，那么在这里输入。
4. 根据自己的选择需要在**Command Line Arguments** 表格添加一行。
5. 5如果有必要填写**Argument Switch** 和 **Argument File** 空白。

5.4.4.6 运行GOCAD Silently Command 配置

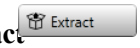
使用GOCAD脚本语言，可以将GOCAD 软件范例中的SGrid 数据导入到CMG文件。地质模型及属性参数都可以输出。

CMOST利用 **Run GOCAD Silently** 命令衔接GOCAD。 CMOST 将触发GOCAD来运行定义的脚本语言和工作流程以便产生CMGinclude文件（.inc），然后CMOST将产生的include文件用于模拟方案文件。
建立链接的步骤:

1. 为**Run GOCAD Silently** 命令准备GOCAD 文件，包括GOCAD Project (.gprj) 文件，GOCAD脚本 (.script)文件和(optional) GOCAD 工作流 (.xml) 文件。关于GOCAD更多信息，请查看Preparing GOCAD Master Script File for Run GOCAD Silently Command。

2. 编辑CMOST主文件 (.cmm)。插入GOCAD产生的include文件，例如:

<cmost>outputPoro</cmost> ***Porosity*
<cmost>outputKi</cmost> PERMJ ***Permeability I*
EQUALSI ***Permeability J*

3. 建立 **Run GOCAD Silently** 命令。
   3.1 添加**Run GOCAD Silently** 命令。
   3.2 如上设置GOCAD .gprj， .script，.xml files。
   3.3 3.3 点击**Extract** 

按钮，在.script 和 .xml文件中提取CMOST参数。新的参数**Parameters**会添加至页面。

5.4.4.6.1 为Run GOCAD Silently 命令准备GOCAD 脚本文件

1. 在GOCAD 支持不同属性的输出命令，然而， CMOST-GOCAD，仅仅 *“write_sgrid_as_CMG_ascii_file”* ，该输出命令能够用于脚本文件来输出油藏几何信息或参数属性include文件(.inc)。

2. 至少有一个 *“write_sgrid_as_CMG_ascii_file”* 输出命令用于GOCAD脚本文件。 另外，CMOST关键字必须用来作为输出文件名字，例如：

   Gocad on SGrid GridName write_sgrid_as_CMG_ascii_file File_name "<cmost>outputPoro</cmost>" origin 0 switchIJ 0 vertical_scaling 1 horizontal_scaling 1 save_geometry 0 use_deadcell 0 properties "POR+Porosity1" lgr_scenario "";

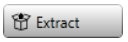
3. 在输出命令文件中仅仅有一个属性能够用来输出。如果有多个参数需要输出，就要使用多个输出命令，例如：

   Gocad on SGrid GridName write_sgrid_as_CMG_ascii_file File_name "<cmost>outputPoro</cmost>" origin 0 switchIJ 0 vertical_scaling 1 horizontal_scaling 1 save_geometry 0 use_deadcell 0 properties "POR+Porosity1" lgr_scenario "";

   gocad on SGrid GridName write_sgrid_as_CMG_ascii_file File_name "<cmost>outputKi</cmost>" origin 0 switchIJ 0 vertical_scaling 1 horizontal_scaling 1 save_geometry 0 use_deadcell 0 properties "PERMI+PermH1" lgr_scenario "";

4. 如果需要CMOST关键字可以用于GOCAD脚本文件中的其他部分 。
5. 一个工作流文件是可选的。如果需要，使用下面的命令加载GOCAD工作流文件。
   gocad load_xml_parameters name "Property_study_loaded" file “Example_GOCAD_Master_XML_File.xml"
6. 在脚本文件的末尾，使用下面的命令退出GOCAD：
   gocad quit really true

5.4.4.6.2 Extract Parameters from GOCAD Master Files

点击Extract  按钮之后，CMOST将复制GOCAD 脚本和XML 文件中的所有参数到任务文件。如果使用*this[OriginalValue]* 语法，缺省值也从文件中复制而来。

下面是个从GOCAD中提取CMOST参数的例子：

Gocad on SGrid GridName write_sgrid_as_CMG_ascii_file File_name "<cmost>This[“poro.inc”]=outputPoro</cmost>" origin 0 switchIJ 0 vertical_scaling 1 horizontal_scaling 1 save_geometry 0 use_deadcell 0 properties "POR+Porosity1" lgr_scenario "";

提取后，参数源*outputPoro* 自动设置为FORMULA，缺省值是 *“poro.inc”*，公式的数值设置为*JobName* + *"outputPoro.inc"*。我们建议不改变这个公式值。

5.5 目标函数

通过Objective Functions，可以定义表达式或参数来优化最小或最大值。例如，在历史拟合的情况下，想减小生产历史数据及模拟数据之间的误差。在方案优化时，可能想得到最大的净现值。 更多信息参考 Objective Functions。

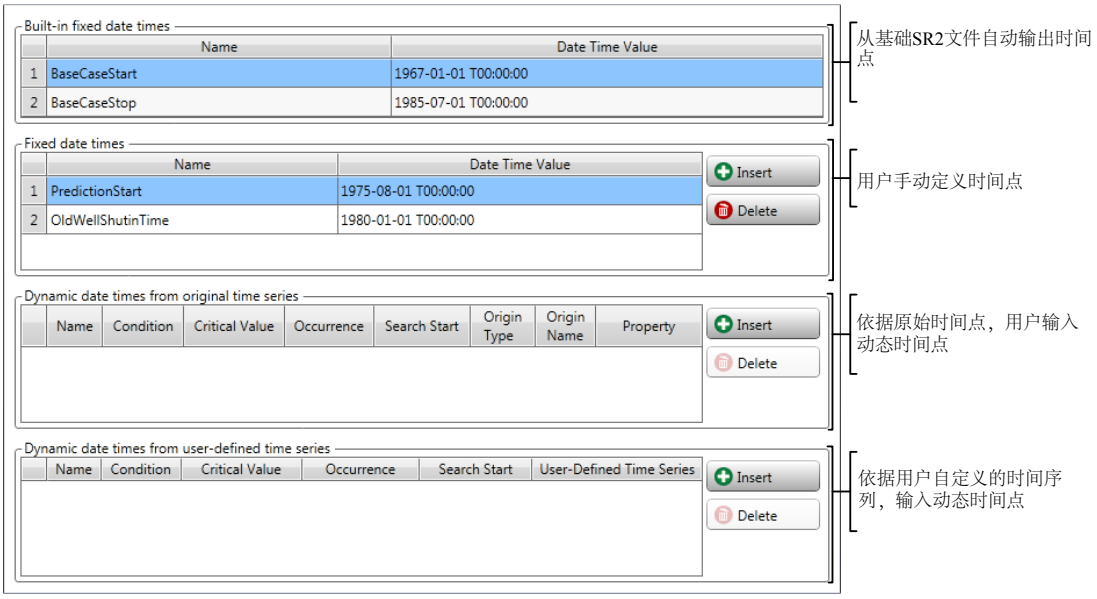
根据Study研究的目的在Objective Functions界面配置子节点。无论Study的研究目的如何，都需要从以下四项中选择一项作为目标函数：

- Basic Simulation Results
- History Match Quality
- Net Present Values
- Advance Objective Functions

例如，如果想对 NPV进行敏感性分析，就不需要配置Basic Simulation Results 界面，然而，需要配置Net Present Values 界面。同样的，如果想要查看输入的参数对累产油Cumulative Oil的影响，就需要配置Basic Simulation Result 界面，最后，如果定义的目标函数用到了Excel，用户自定义编码或可执行文件，那么可能会用到高级目标函数。

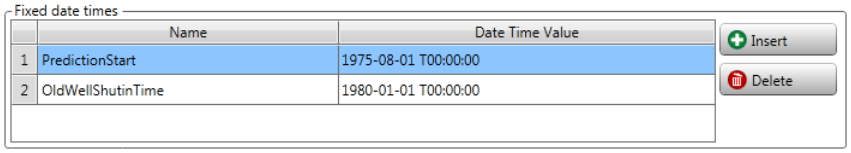
5.5.1 典型日期时间点

通过Characteristic Date Times 界面，如下所示，可以指定目标函数及目标函数项中使用的日期，也可以定义用于计算目标函数的动态时间。



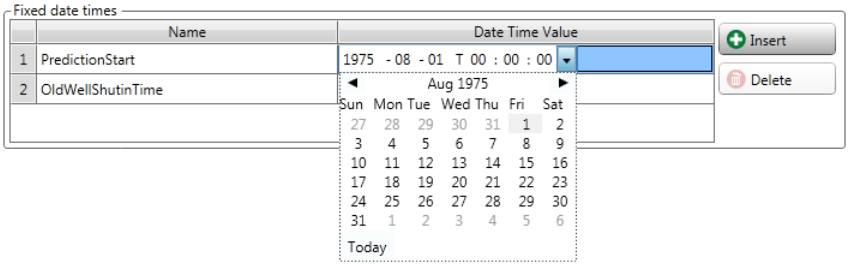
如上所示，通过**Characteristic Date Times**界面定义了四种不同类型的时间点：

- **Built-in fixed date times**：时间点从SR2文件自动输出，在上面的例子中 BaseCaseStart 和 BaseCaseStop 。
- **Fixed date times**：Specific用户自定义时间点，在表格右侧点击Insert，然后输入Name和Date Time Value，如下所示：



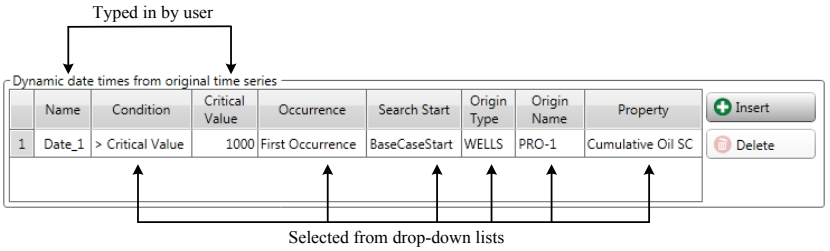
通过以下两种方式输入**Date Time Value**：

1. 点击当前Date Time Value，然后使用TAB键在输入日期来回移动，输入年、月、日。
2. 使用日历下拉选项选择日期，如下：

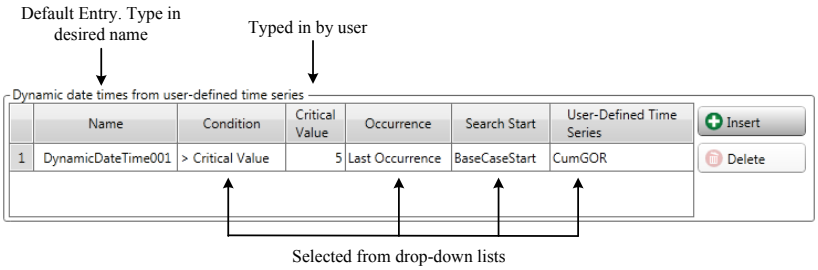


- **Dynamic date times from original time series**：依据原始时间序列定义动态时间点，例如，当累产油量超过某一数值时，定义对应的时间点。

在下面的例子中，*dynamic date Date_1* 是*BaseCaseStarton* 时间点之后的第一个时间点，该时间点是井*PRO-1* 的累产油超过*1000 m3*对应的时间点。



- **Dynamic date times from user-defined time series**：依据用户定义的时间序列定义动态时间点，例如*DynamicDateTime001* 累积油汽比*CumGOR*大于5的最后一个时间点。



5.5.2 基础模拟结果

参考Objective Functions 开头部分，讨论在Objective Functions界面配置子节点。

通过**Basic Simulation Results**界面，可以直接从原始时间序列和用户定义的时间序列定义目标函数。这些结果可用于目标函数或定义总目标函数参考值和软性约束条件。

可以定义下面目标函数类型：

- **Basic Simulation Result from Original Time Series**：通过这个表格，输入的结果都是来源于原始时间序列SR2文件。在下面的例子，定义了基础模拟结果*ProducerCumOil*，*ProducerCumWater* 和 *InjectorCumWater*：

Basic Simulation Result from Original Time Series

	Name	Characteristic Time	Origin Type	Origin Name	Property
1	ProducerCumOil	YearEnd2011	WELLS	PRODUCER	Cumulative Oil SC
2	ProducerCumWater	YearEnd2011	WELLS	PRODUCER	Cumulative Water SC
3	InjectorCumWater	YearEnd2011	WELLS	INJECTOR	Cumulative Water SC

Insert | Repeat | Delete

- **Basic Simulation Result from User-Defined Time Series**：通过这个表格，可以将用户定义的时间序列作为来源定义目标函数。在下面的例子中，定义了*CumGORSC_YearEnd2011*作为用户定义的时间序列*CumGORSC*下*characteristic date YearEnd2011*：

Basic Simulation Result from User-Defined Time Series

	Name	Characteristic Time	User-Defined Time Series
1	CumGORSC_YearEnd2011	YearEnd2011	CUMGORSC

Insert | Delete

- **Characteristic Time Durations**：运算时间介于两个*characteristic times* 之间。在下面的例子中，*Duration001* (按天计算）在

Base

Characteristic Time Durations

	Name	Duration Start Time	Duration End Time	Time Unit
1	Duration001	BaseCaseStart	YearEnd2011	Day

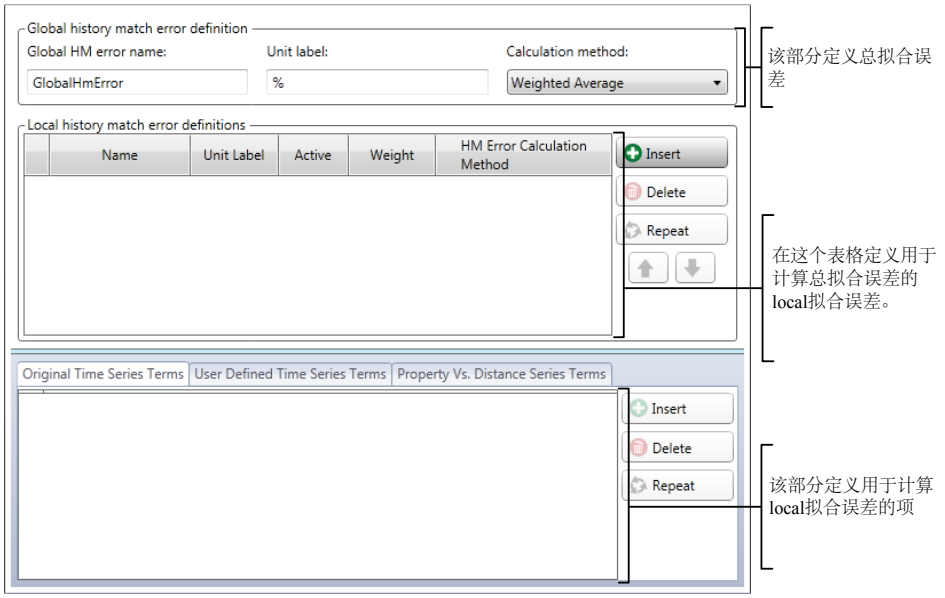
Insert | Delete

5.5.3 历史拟合精度

参考Objective Functions开头部分，讨论在Objective Functions界面配置子节点。历史拟合误差是模拟结果与生产历史数据之间的误差。更多信息，参考History Match Error。

定义历史拟合误差函数的步骤，如下：

1. 打开**Objective Functions | History Match Quality** 界面：



2. 在**Global HM Error Definitions**空白处，定义总拟合误差：

   - **Global HM Error Name**：输入名称，该名称能够清晰的描述总目标函数。
   - **Unit Label**：显示总目标函数单位。该设置，缺省单位是“%”，是可选的，不会影响总拟合误差的计算。

   • **Calculation Method**：选择*Weighted Average* 或 *Get Maximum*。如果设置*Weighted Average*，CMOST 将对所有的local拟合误差按照权重进行平均，以此用来计算总拟合误差如下：

$$\text{Global HM Error} = \frac{\sum w_i \, \text{LHME}_i}{\sum w_i}$$

   其中LHME_i 是 local 拟合误差i的值，w_i 是它的权重因子。如果选择*Get Maximum*，总拟合误差等于最大的local拟合误差。

3. 在**Local History Match Error Definitions**表格，定义 local 拟合误差，用于计算总拟合误差，如下：

   - **Name**：拟合误差的名称必须是唯一的。
   - **Unit Label**：单位应该显示local拟合误差函数。该设置缺省的单位是“%”，它是可选的，但不会影响总目标函数拟合误差计算。

- **Active**：**Active** 选项框决定了CMOST是否使用local拟合误差来计算总拟合误差。如果选上**Active**选项框，CMOST将使用拟合误差，否者local拟合误差不用于计算总拟合误差。所有未被激活的local拟合误差，如果它们是目标观察函数，也会被运算。

- **Weight**：如果在**Calculation Method** 中选择*Weighted Average* ，那么local拟合误差会根据权重因子对总拟合误差造成影响。权重因子越大，对总拟合误差的影响也就越大。

- **HM Error Calculation Method**：选择历史拟合误差计算方法（HM Error Calculation Method）加权平均（*Weighted Average*）或者or最大拟合误差（*Get Maximum*）中的一种。如果选择加权平均（*Weighted Average*），CMOST会根据项权重因子（*Term Weight*）平均所有的项用于计算local拟合误差，如下：

$$\text{Local HM Error} = \frac{\sum w_i t_i}{\sum w_i}$$

其中，ti 是项i，wi 是其权重因子。

如果选择最大拟合误差（*Get Maximum*），拟合误差等于最大的项拟合误差。

4. 在**Local history match error definitions** 表格中，选择第一个local 拟合误差，根据实际情况，选择计算local拟合误差所对应的**Original Time Series Terms**，如下例所示：

Original Time Series Terms | User Defined Time Series Terms | Property vs. Distance Series Terms

	Origin Type	Origin Name	Property	Start Time	End Time	Reset Cumulative	Absolute Measurement Error	Term Weight	Normalization
1	WELLS	Injector	Cumulative Gas SC	BaseCaseSta	BaseCaseStop	☐	0	1	Auto

为每个原始时间序列项插入一行空白，如下：

- **Origin Type**：从下拉菜单中，选择数据类型。从*WELLS*、*GROUPS*、*SPECIALS*、*SECTORS*、*LAYERS*或 *LEASES*中选择一个。所有的 **Origin Types** 都来自结果文件。

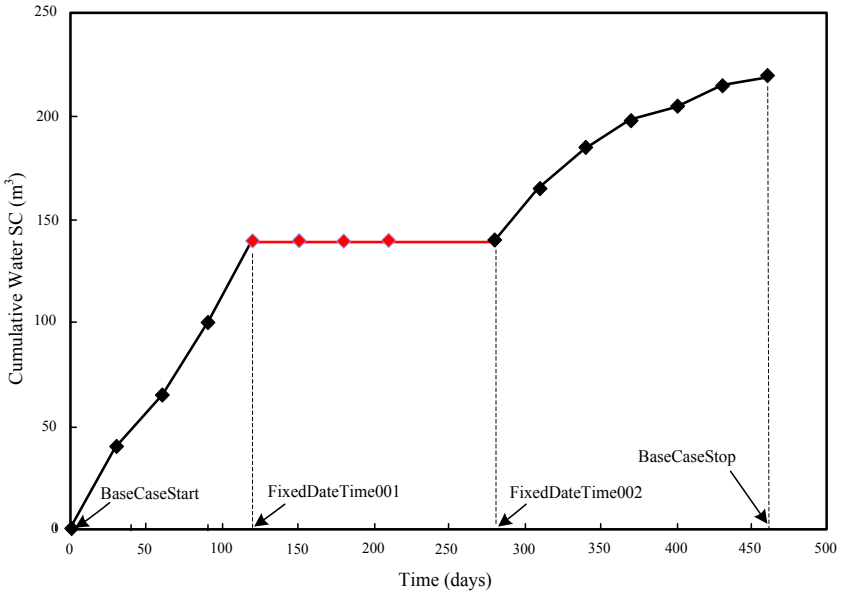
- **Origin Name**：该选项都是基于**Origin Type**在下拉菜单中的选择。如果在对应**Origin Type**下没有任何项，**Origin Name**列将是空白。 如果情况属实，那么**Origin Type** 不能使用。例如，如果**Origin Type** 选择的是*WELLS*，那么在**Origin Name** 将会包括所有基础文件中井的名称。

- **Property**：该选项都是基于**Origin Name**在下拉菜单中的选择。例如，如果**Origin Type** 选择 *WELLS*，则**Property** 方框会有一系列的选项，例如*Cumulative Oil SC* 或 *Gas Rate SC*。

• **Start Time**：从基础模型文件中，选择开始计算的特征时间点。该时间点必须在模拟起始和结束时间点之间。缺省的开始时间是基础模型运行的起始时间点。

• **End Time**：选择CMOST停止分析数据的特征时间点。该时间点必须在模拟起始和结束时间点之间。缺省的结束时间是基础模型运行的结束时间点。

• **Reset Cumulative**：在做历史拟合时，用户可能会发现部分生产历史数据是不可信的，因为有时候生产历史数据没有记录或者是估算的。可靠的数据部分与不可靠的数据部分混在一起。例如，在下图中，黑色部分表示可靠的数据，红色部分表示不可靠数据，换句话说，我们不知道红色部分真实的产水量。



为了拟合上图中的累产水曲线，我们需要为黑色部分两个时间阶段定义历史拟合误差项，忽略被认为是不可靠数据的红色部分。为了正确使用有效的累产数据，我们也有必要为每个时间段计算和使用独立的累产曲线，这就意味着每个阶段开始时间点对应的累产量是零，并结合每个时间点累产量的变化量。如果累产量（例如累产油/水/气）来源于速率（例如油/水/气速率），则需要使用该关系式。为了能够使用该关系式，用户为历史拟合项选择***Reset Cumulative***。

对于上面的累产水曲线，定义历史拟合误差Water1：

	Origin Type	Origin Name	Property	Start Time	End Time	Reset Cumulative	Absolute Measurement Error	Term Weight	Normalization
1	GROUPS	Default-Field-PRO	Cumulative Water SC	BaseCaseStart	FixedDateTime001	☐	0	1	Auto

Water 3：

	Origin Type	Origin Name	Property	Start Time	End Time	Reset Cumulative	Absolute Measurement Error	Term Weight	Normalization
1	GROUPS	Default-Field-PRC	Cumulative Water SC	FixedDateTime002	BaseCaseStop	☑	0	1	Auto

在BaseCaseStart 和 FixedDateTime001之间，定义历史拟合误差项 Water1 。可以选择为该选项设置Reset Cumulative，因为前四个月的累产水是可信的。

在接下来的五个月（从*FixedDateTime001* 到 *FixedDateTime002*），是上面提到的红色部分累产水曲线，该部分不需要历史拟合误差项，因为该时间阶段的数据是不可信的，应该忽略。

- **绝对测量误差：**用来表示生产历史数据的精度。该数值被认为是绝对误差的一半，意味着如果是模拟结果在（historical value – ME）和（historical value + ME）之间，表示拟合精度满足要求（因为拟合精度在要求的范围之内）。更多关于Measurement Error用来计算历史拟合误差的内容参考History Match Error 部分。

- **权重因子**：Term Weight（权重因子） 表示当前历史拟合误差项的重要程度。Term Weight（权重因子） 越高，该项对拟合误差影响越大。通常，权重因子越大，说明该井越重要，拟合起来也越困难。

- **Normalization**：关于这个参数的更多信息，参考History Match Error。

5. 根据实际情况，选择**User Defined Time Series Terms** 用来计算local拟合误差，如下例所示：

Original Time Series Terms | User Defined Time Series Terms | Property Vs. Distance Series Terms

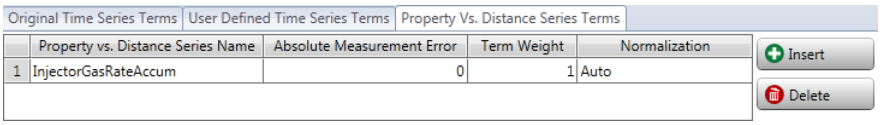
	User-Defined Time Series	Start Time	End Time	Absolute Measurement Error	Term Weight	Normalization
1	CumGORSC	BaseCaseStart	BaseCaseStop	0	1	Auto

Insert Delete

为表格中每个用户定义的时间序列项插入一行，输入空白处，如下：

- **User-Defined Time Series**：选择已经通过**Fundamental Data | User-Defined Time Series**输入的时间序列。
- **Start Time**：如上面for Original Time Series Terms所述。
- **End Time**：如上面 Original Time Series Terms所述。
- **Absolute Measurement Error：如上面Original Time Series Terms所述。**
- **Term Weight**：如上面Original Time Series Terms所述。
- **Normalization**:：如上面Original Time Series Terms所述。

6. 选择Property vs. Distance Terms键，根据实际情况选择用于计算local拟合误差的参数VS.距离项，如下例所示：



在表格中为每个参数VS.距离项插入一行，然后填充空白处，如下：

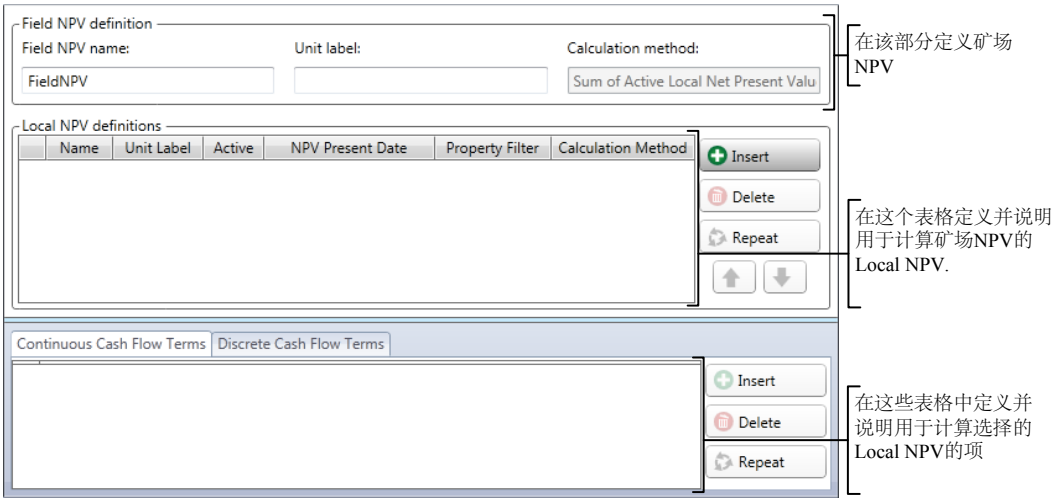
- **Property vs. Distance Name**：从**Fundamental Data | Property vs. Distance Series**中输入的时间序列中选择某一时间序列。
- **Absolute Measurement Error：**如上面**Original Time Series Terms**所述。
- **Term Weight**：如上面**Original Time Series Terms**所述。
- **Normalization**如上面**Original Time Series Terms**所述。

7. 为其他的local拟合误差重复4-6步。

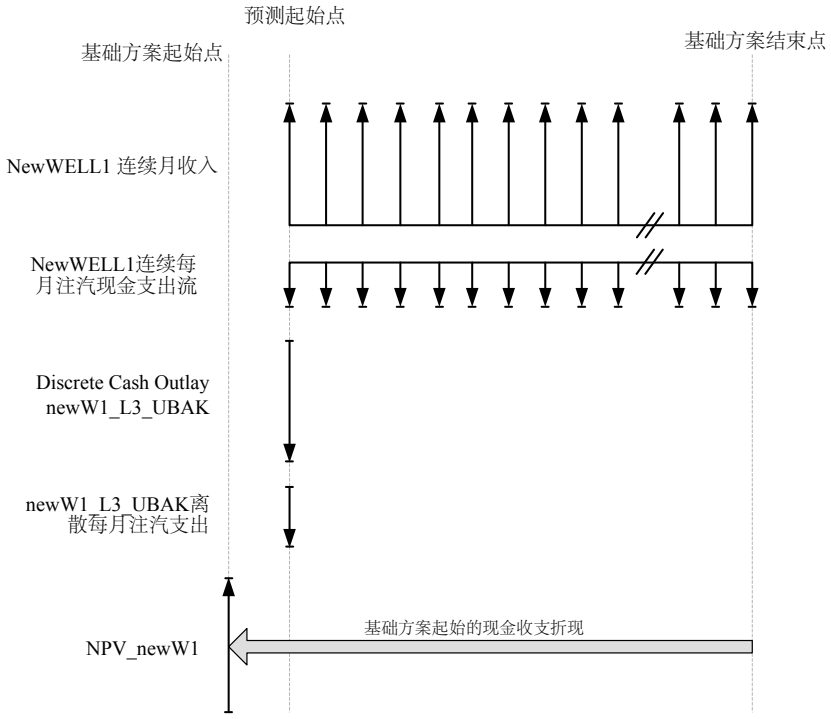
5.5.4 净现值

参考Objective Functions 开头部分，讨论在Objective Functions 界面配置子节点。

通过**Net Present Values**界面，你可以定义矿场NPV总目标函数，如下所示：



为了配置**Net Present Values**界面，参考以下算例：



1. 现金流是折现的，每月的折现率是从每年的折现率计算的（该例子中所有折现率为0.1）。
2. 所有现金流目前的值决定了NPV_newW1。
3. 矿场NPV是本地NPV之和：

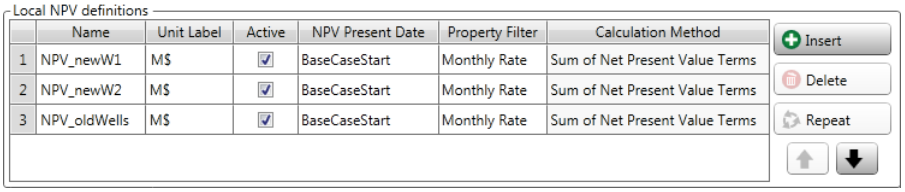
   NPV_newW1 + NPV_newW2 + NPV_oldWells

下面部分用来说明上面总NPV目标函数的定义步骤：

1. 在**Field NPV Definition** 界面，如下例所示，定义矿场NPV：

- **Field NPV Name**：输入唯一的，具有描述性的名称。
- **Unit Label**：输入想要表示矿场NPV的单位。例如，或许想用百万美元（M$ ）来表示净现值。该设置是可选的，不会影响矿场NPV的计算。
- **Calculation Method***：* 该空白处设置为*Sum of Active Net Present Values*，不能再做修改。矿场NPV是所有激活的*Local NPV*的算术求和。

2. 在**Local NPV Definitions** 表格，输入并配置用于计算矿场NPV的Local NPV，如下：

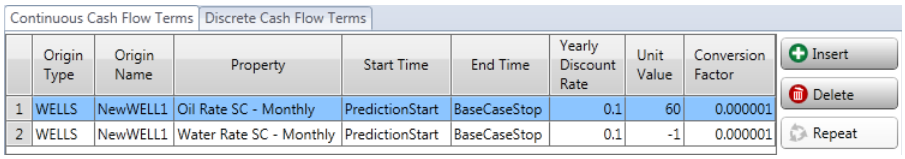


3. 点击第一个Local NPV灰色空白处，填充空白，如下：

- **Name**：Study中的Local NPV名称必须是唯一的。
- **Unit Label**：单位应该和Local NPV一起显示。这个设置不会影响矿场NPV计算。
- **Active**：Active 选项决定了CMOST是否使用Local NPV 来计算矿场NPV。如果选择Active，CMOST 将使用Local NPV，否者Local NPV也运算，但不能用于计算矿场NPV。 如果所有未被激活的Local NPV是观察目标函数的话，也会执行计算。
- **NPV Present Date**：为未来的现金流折现选择时间点，例如，金钱的时间价值。
- **Property Filter**：作为筛选器帮助用户设置属于local NPV 的现金流。例如， 如果选择*Daily Rate*，在现金流项，属性栏的下拉菜单中仅仅有日速率。

- **Calculation Method**：该空白处设置为*Sum of Net Present Value Terms*，不能修改。*Local NPV* 是连续和离散现金流的算术求和。

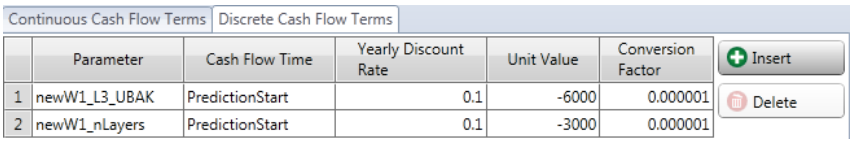
4. 选择 **Continuous Cash Flow Terms** 标签，输入所有连续现金流，用来计算 Local NPV，如下面例子*NPV_newW1*所示：



- **Origin Type**：从下拉菜单中，选择需要的类型，*WELLS*、*GROUPS*、*SPECIALS*、*SECTORS*、*LAYERS*或*LEASES*。所有的 Origin Types 都来源于模拟结果文件。

- **Origin Name**：根据选择的Origin Type，然后在下拉菜单中选择需要的选项。如果对应的Origin Type 中没有相对应的条目，那么Origin Name 列表是空白的。如果是这种情况Origin Type 则不能使用。例如，如果 Origin Type选择的是*WELLS*，则 Origin Name 将包括基础模型中所有井的名称。
- **Property**：根据选择的Origin Name，然后在下拉菜单中选择需要的选项。例如，如果Origin Type 选择的是*WELLS*，则Property 空白处包含所有井的参数，例如月产油速率*Oil Rate SC-Monthly* 或月产水速率*Water Rate SC-Monthly*。
- **Start Time**：选择计算现金流起始日期点。
- **End Time**：选择计算现金流结束时间点。
- **Yearly Discount Rate**：以小数的方式输入折现率，例如0.1，而不是10%。
- **Unit Value**：为属性值输入单位价值。按照惯例，收入为正值，支出为负值。在上面的例子中，*Oil Rate SC – Monthly* 是按桶为单位计算。每桶$60。
- **Conversion Factor**：如果需要将现金流转换成常用单位，就需要输入转换因子。在我们的例子中，我们将现金流单位转换为百万美元。*Oil Rate SC – Monthly* 单位价值是每桶$60，因子我们需要乘以*0.000001* 来转换成百万美元。

5. 选择**Discrete Cash Flow Terms** 项，然后在表格中输入需要计算Local NPV 所有现金流。如下例所示：



- **Parameter**：为离散的现金流项输入唯一的、描述性的名称。
- **Cash Flow Time**：选择发生离散现金流动的时间点。
- **Yearly Discount Rate**：以小数的方式输入年折现率，例如 0.1而不是10%。
- **Unit Value**：为离散现金流输入单位价值。按照惯例，收入为正，支出为负，在上面的例子中，newW1_L3_UBAK 单位价值为-6000 （支出6000）。
- **Conversion Factor**：如果需要将现金流转换成常用单位，就需要输入转换因子。在我们的例子中，我们将现金流和离散值转换成百万美元。我们例子中的单位是美元，因此我们需要乘以0.000001 将其转换成百万美元。

6. 为其他Local NPV重复3-5步。这样就为矿场NPV定义了目标函数。

5.5.5 高级目标函数

通过**Advanced Objective Functions**节点，定义高级目标函数：

- **Excel spreadsheet calculation**：可以配置CMOST写入参数值和模拟结果到某个指定的工作表格中，然后从该表格读取目标函数结果。
- **User-defined source code calculation**：利用JScript代码来自定义目标函数。参考Using JScript Expressions in CMOST。
- **User-defined executable calculation**：可以使用第三方软件，例如MATLAB来计算并且得到目标函数值。

注意：对所有三种类型的高级目标函数， 提供一个 

，使用基础文件 Test 和参数来计算高级目标函数。

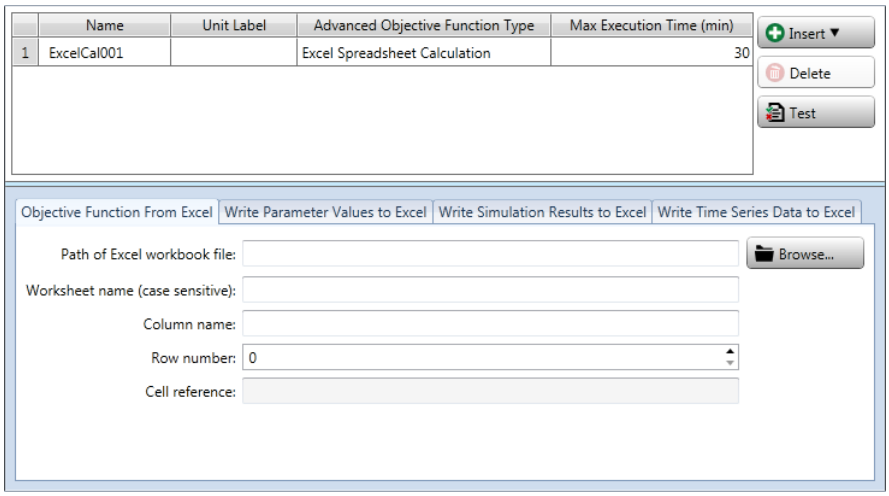
5.5.5.1 输入Excel 电子表计算输入

在开始该步骤之前，先定义Excel 电子表，CMOST将写入数据并且读取高级目标函数。为每个Study生成一个电子表，其名称是每个Study试验的名称，数量是所有试验方案的总数，例如：

Study_1_00062.xls

高级目标函数电子表将保存在cmsd 文件夹。

1. 在 **Advanced Objective Functions** 节点，点击 Insert ，然后选择**Use Excel Spreadsheet Calculation**。一个新的高级目标函数就添加至**Advanced Objective Functions** 表，该高级目标函数使用缺省名字。下面的表格定义 Excel 电子表的界面，如下所示：



2. 在Advanced Objective Functions 表格，输入以下部分：

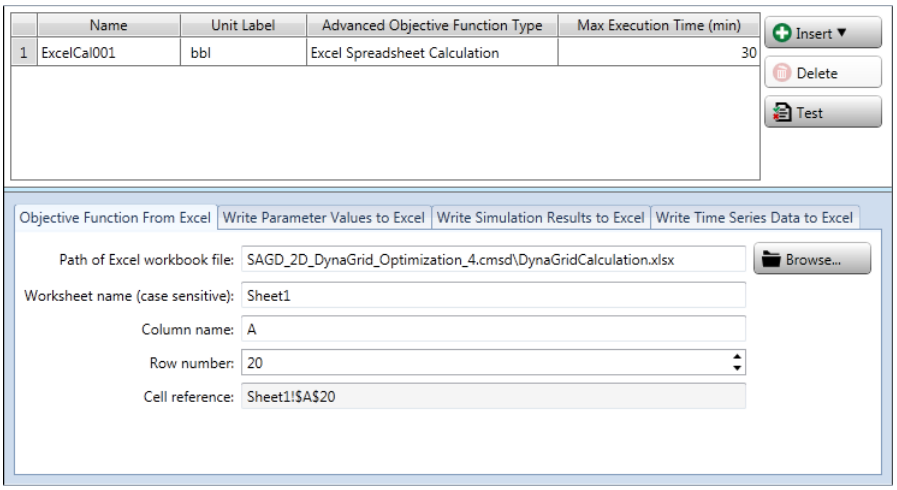
   - **Name**：输入一个描述高级目标函数的名称。 Name 会在结果图形中显示，将用于计算总目标函数。

   - **Unit Label**：输入高级目标函数的单位。这些单位也将在结果图形中显示。

   - **Advanced Objective Function Type**：该选项仅供阅读，不能修改。

   - **Max. Execution Time (min)**：输入CMOST最大执行时间（按分钟计算）。如果在最大执行时间内，还没有运算完成，CMOST引擎将会停止，在**Control Centre** 界面**Engine Events** 的表格中会出现一个错误提示。

3. 在 **Objective Function From Excel** 选项， CMOST 将从定义的Excel 电子表读取目标函数值，如下例所示：



在例子中，我们浏览了Excel电子表以及指定的工作表，CMOST将读取目标函数中的行和列的信息。为每个试验方案执行相应的计算，得到的目标函数值将会呈现在Experiments Table。同时，电子表为每个Study来计算高级目标函数，并将结果保存于Study文件夹。

4. 在**Write Parameter Values to Excel** 选项， 指定想要提交至Excel电子表的参数， worksheet、column、以及 row如下例所示：

	Parameter Name	Worksheet Name	Column Name	Row Number	Cell Reference
1	INTOI	Sheet 1	A	1	Sheet 1!$A$1

在上面的例子中，CMOST将写入*INTOI* 值，用于Excel电子表工作表 *Sheet1* 中的$A$1。

5. 在 **Write Simulation Results to Excel** 选项，指定想要提交至Excel电子表模拟结果，其中worksheet、column、以及 row如下例所示：

	Origin Type	Origin Name	Property	Simulation Date Time	Worksheet Name	Column Name	Row Number	Cell Reference
1	WELLS	PRODUCER	Cumulative Oil SC	2015-12-01 T00:00:00	Sheet1	A	2	Sheet1!$A$2

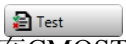
在上面的例子中，对于PRODUCER，CMOST 将在定义的空白处及相应时间点写入*Cumulative Oil SC* 数值。

6. 在**Write Time Series Data to Excel** 选项，定义想要写入到Excel电子表时间序列，另外，也可以定义数据写入的方向（水平或垂直）以及按照时间序列写入。

	Origin Type	Origin Name	Property	Calculation Frequency	Worksheet Name	Start Column Name	Start Row Number	Start Cell Reference	Output Direction	Write Time Array
1	WELLS	PRODUCER	Cumulative Oil SC	EveryMonth	Sheet 1	A	4	Sheet 1!$A$4	Vertical	☑

在上面的例子中，对于井PRODUCER，CMOST将写入累产油时间序列数值，按月计算，开始于Sheet1:$A$1，按照时间序列垂向排列。

依据写入Excel电子表中的时间序列数据和其他数据，CMOST将从Objective Function from Excel 选项计算和读取高级目标函数。

7. 点击Test  按钮来测试电子表格计算，会创建一个Test.xlsx 电子数据表。可以在CMOST引起启动之前打开这个电子数据表，来验证Excel计算的正确性。

5.5.5.2 输入用户自定义源-码计算

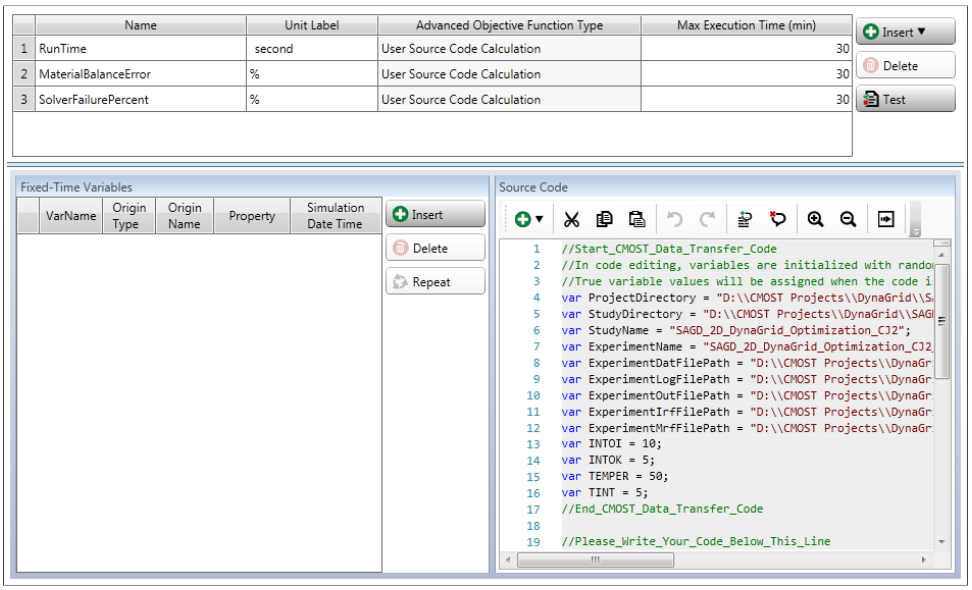
1. 在**Advanced Objective Functions** 节点，点击**Insert** ，然后选择**Use User Defined Source Code Calculation**，在**Advanced Objective Functions** 表格中，添加了一个新的高级目标函数，并且使用了唯一的缺省名称。在**Advanced Objective Functions**表格，输入：

   - **Name**：输入名称，该名称描述了高级目标函数。 Name 将呈现在结果图片中，用于计算总目标函数。

   - **Unit Label**：输入高级目标函数单位，这些单位将呈现在结果图片中。

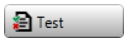
   - **Advanced Objective Function Type**：该选项只能用于阅读，不能修改。**Max. Execution Time (min)**：输入CMOST最大执行时间（按分钟）。如果在规定的最大执行时间内，该操作没有完成，CMOST引擎将停止，并且在Control Centre 界面Engine Events 表格中会出现错误信息。

可能需要JScript 计算，通过输入的Jscript编码来进行筛选，所以提供了一个表格用于输入确定的时间序列。更多信息请参考Using JScript Expressions in CMOST：



2. 如果有必要，插入并且配置在Jscript计算中需要的固定时间变量。

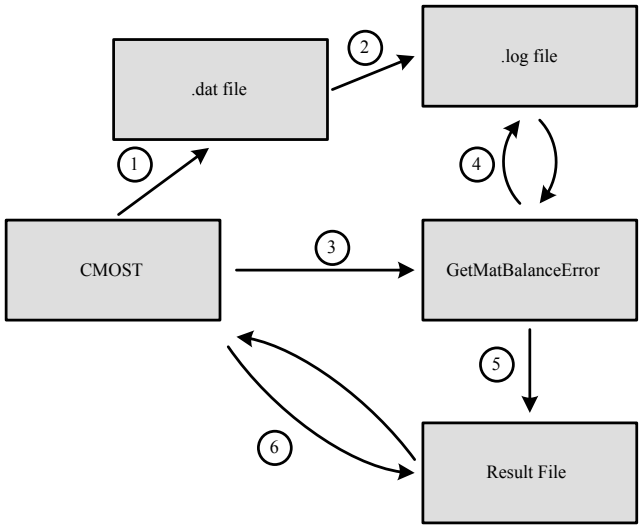
3. 在**Source Code** 部分需要注意，输入JScript 编码需要执行计算。

4. 在启动CMOST引擎之前，点击**Test** 

按钮来测试JScript 编码语言。如果在**Test Calculation Results** 对话框返回一个数值，表示运算正确。

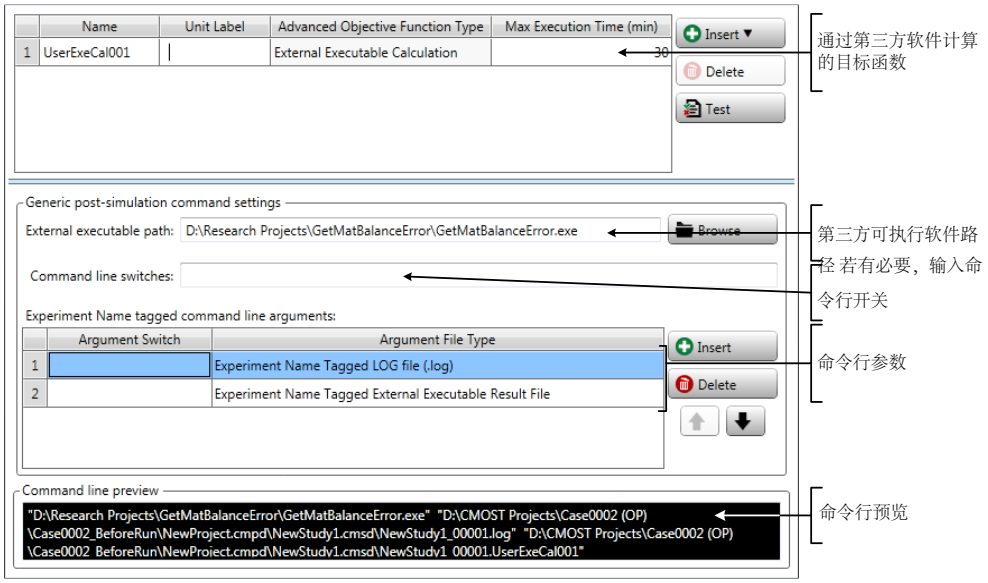
5.5.5.3 输入用户自定义执行计算

应用第三方软件来读取模拟结果信息，计算目标函数值并将该值写入文件。CMOST 可以从文件中读取该值，并且将这些数值存储在**Experiments Table**。为了说明该部分，参考下面的例子。CMOST可以使用可执行文件 *GetMatBalanceError.exe*，该软件主要用于计算高级目标函数-**UserExeCal001**：



1. CMOST创建试验数据，例如：*SAGD_2D_DynaGrid_Optimization_4_00001.dat*.
2. 模拟器运行模拟方案，生成.log，.out和SR2文件。
3. CMOST计算目标函数。计算目标函数 **UserExeCal001**，CMOST 跟踪可执行程序*GetMatBalanceError.exe* (存储于文件夹D:/ResearchProjects/GetMatBalanceError）。 CMOST通过两个可执行参数- .log 文件的路径，*SAGD_2D_DynaGrid_Optimization_4_00001.log*和结果文件的路径，*SAGD_2D_DynaGrid_Optimization_4_00001.UserExeCal001*。
4. 从.log f文件*GetMatBalanceError.exe* 中，读取物质平衡误差。
5. *GetMatBalanceError.exe*将物质平衡误差写入结果文件。
6. CMOST读取结果文件中的数值，将其设置为UserExeCal001。CMOST在算法中使用该值，根据实际情况，将其插入**Experiments Table** 中**UserExeCal001**列。

按上面所述，配置**Advanced Global Objective** 界面，如下所示：



NOTE: 也可以使用 ↑ 和 ↓ 键来选择命令行参数来排列顺序。

在界面顶部**Advanced Objective Functions** 表格，选择用户可执行计算，然后点击Test Test 按钮来验证计算引擎是否正确，如果在Test Calculation Results 对话框返回一个数值，表示运算正确。

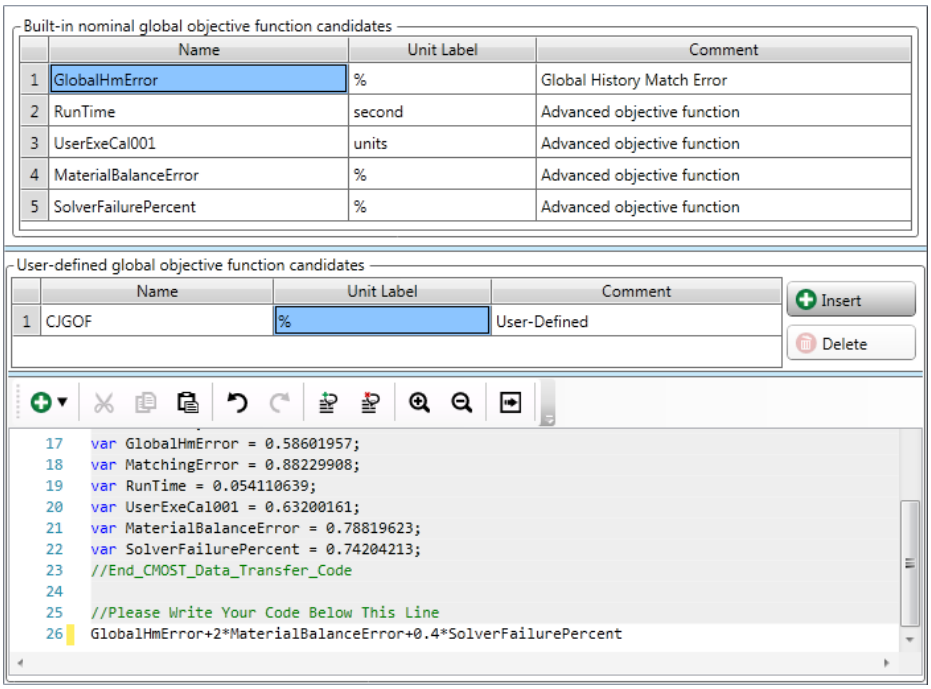
5.5.6 总目标函数候选值

通过**Global Objective Function Candidates**节点，可以查看建立的总目标函数候选值，如果已经定义了下面的目标函数作为基本模拟结果：

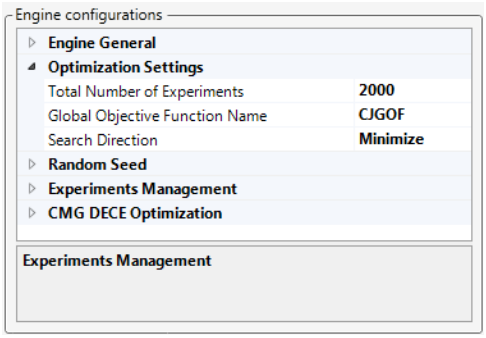
- 历史拟合误差
- 净现值
- 高级目标函数
- 基础模拟结果

当然，通过**Global Objective Function Candidates** 节点，使用建立常用的总目标函数候选值作为变量 ，也可以定义额外的总目标函数。在下面的例子中，我们定义了*CJGOF*，作为建立的几个常用总目标函数候选值的函数：

$$CJGOF = GlobalHmError + 2 \times MaterialBalanceError + 0.4 \times SolverFailurePercent$$



正如建立的常用目标函数候选值，例如，通过Engine Settings 节点，可以指定用户自定义总目标函数的方向（最值或最大值），如下所示：



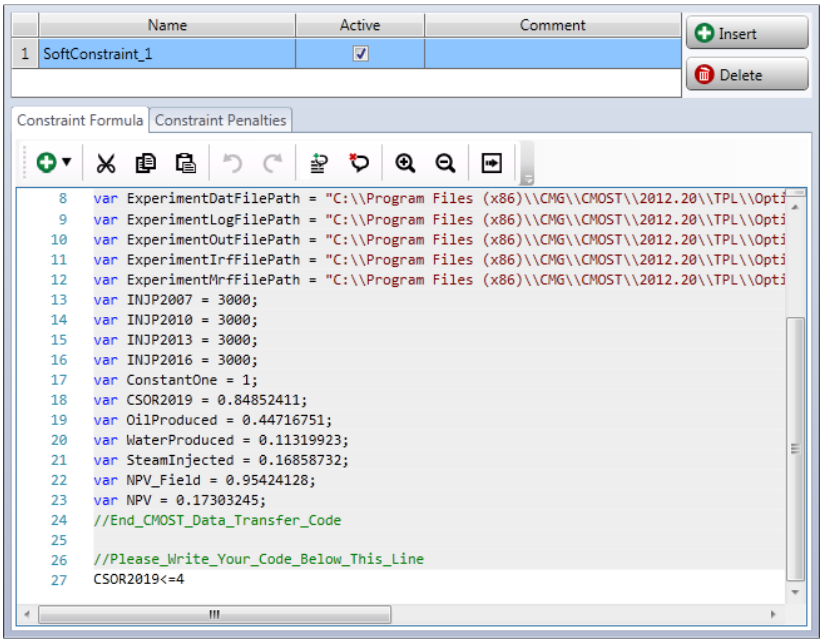
5.5.7 软性约束条件

可以为历史拟合和方案优化定义硬性和软性约束条件。当违反了软性约束条件时，允许用户改变目标函数。关于硬性约束条件的更多信息，请参考Hard Constraints 。

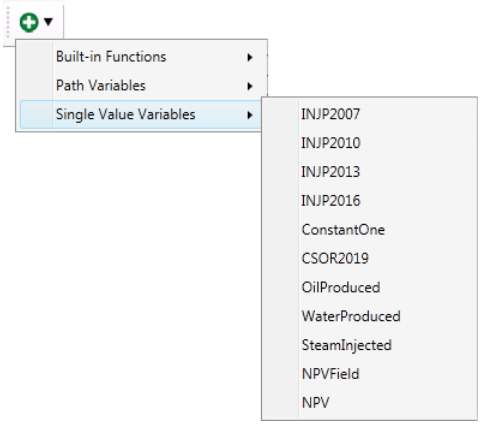
通过**Soft Constraints** 界面，可以定义软性约束条件，如果违反了软性约束条件，将会重写目标函数。当模拟正常运行的时候，将会检查是否违反了软性约束条件，如果违反了软性约束条件，立即对目标函数进行相应的“惩罚”。

定义软性约束条件：

1. 选择**Constraint Formula** 选项：
2. 点击**Soft Constraints**表格右面的 Insert 按钮。输入软性约束条件。如果必要的话，可以输入一个名称和说明。选择CMOST Active命令，检查是否违反了约束条件。
3. 当创建了一个软性约束条件时，在Constraint Formula部分打开一个 CMOST Formula Editor，为软性约束条件输入一个公式，如下所示：



通过点击 按钮，可以为公式输入任何函数和变量，例如：



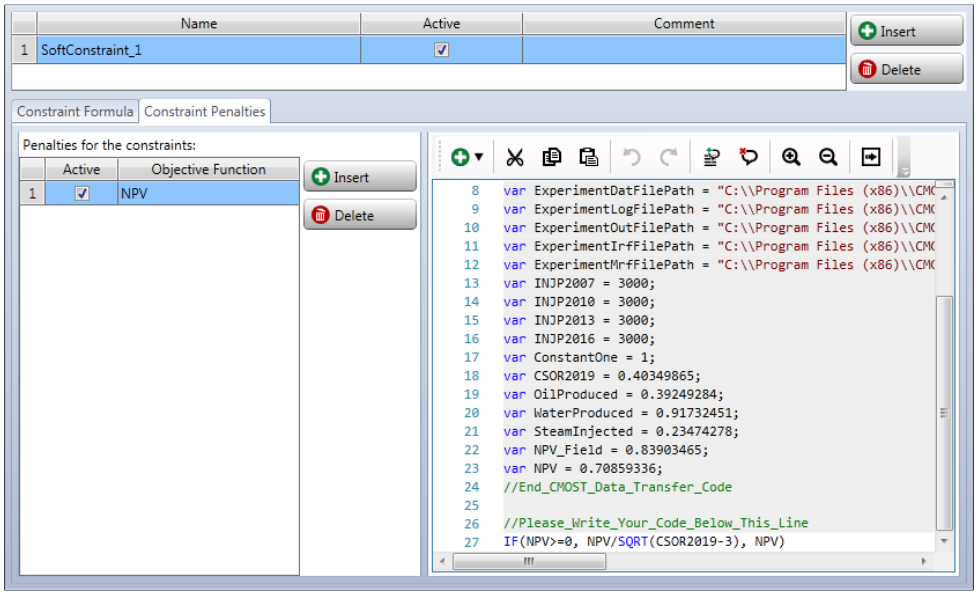
该列的变量包括先前定义的参数、目标函数以及基础模拟结果。可以采用交替的方法手动输入公式。在上面的例子中，如果变量*CSOR2019* 不小于等于4，则违反了*Soft_Constraint_1* 。

定义一个“惩罚”：

定义软性约束条件的另外一部分是定义 “惩罚”，该“惩罚”用于违反软性约束条件：

1. 在**Constraints** 表格，点击软性约束条件。对于在**Constraints** 表格中的每个软性约束条件指定一个或多个“惩罚”。

2. 选择**Constraint Penalties** 选项：



3. 点击**Penalties for the constraints** 旁边 Insert 添加一个“惩罚”。

4. 从下拉菜单中，选择**Objective Function** ，根据实际情况，选择**Active**选项框。

5. 在Formula Editor 空白处，输入“惩罚”，定义“惩罚”可能需要多行。在上面的例子中，如果违反了Soft_Constraint_1 ，按如下方式调节 NPV ：

如果目标函数*NPV* 大于等于0， NPV 被如下公式代替：

$$\frac{NPV}{\sqrt{(CSOR2019-3)}}$$

如果*NPV* 小于0，它的值不变。

如果目标函数违反了多个软性约束，那么目标函数按照最后一个软性约束条件执行。

当模拟方案运行时，在**Control Centre** 界面**Engine Events** 表格会呈现违反的多个软性约束条件。

6 CMOST 运行和控制 (Running and Controlling CMOST)

6.1 简介

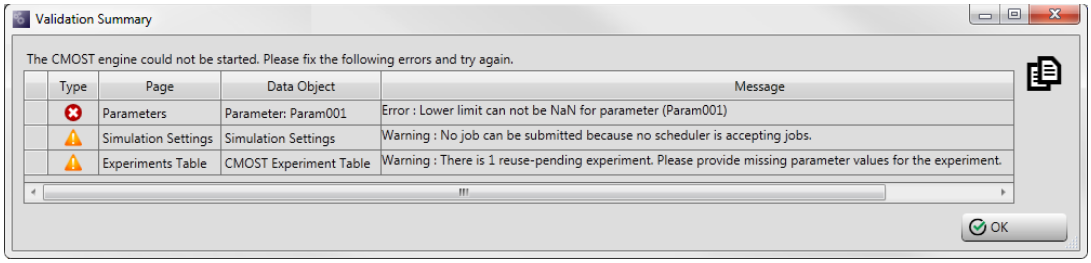
Control Centre节点及其子节点用于设置模拟器及实验方案，然后运行CMOST任务。

6.2 控制中心

通过**Control Centre**界面，可以启动引擎，检测运行过程，暂停和停止CMOST任务。在开始CMOST任务之前，重新查看以及配置以下界面：

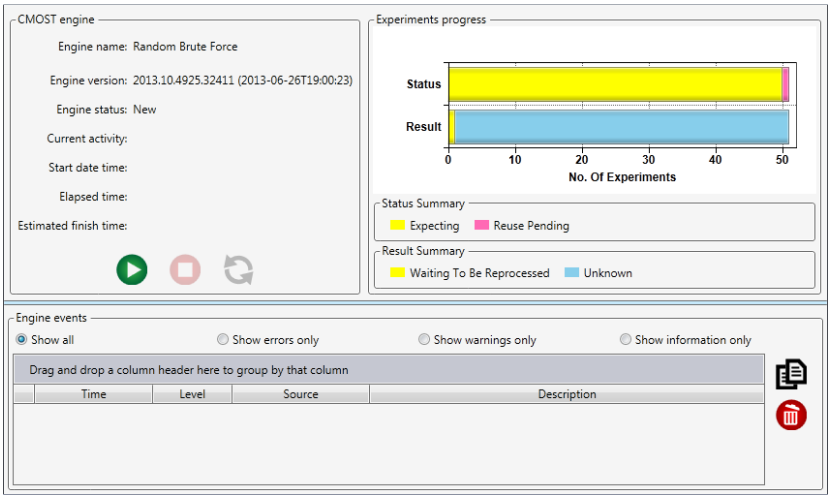
- Engine Settings
- Simulation
- Experiments Table

一旦完成上述设置，就可以通过工具栏**Control Centre** 节点点击**Start Engine** 按钮来运行CMOST任务。如果出现显著的错误，对话框 **Validation Error Summary** 会出现以下提示：



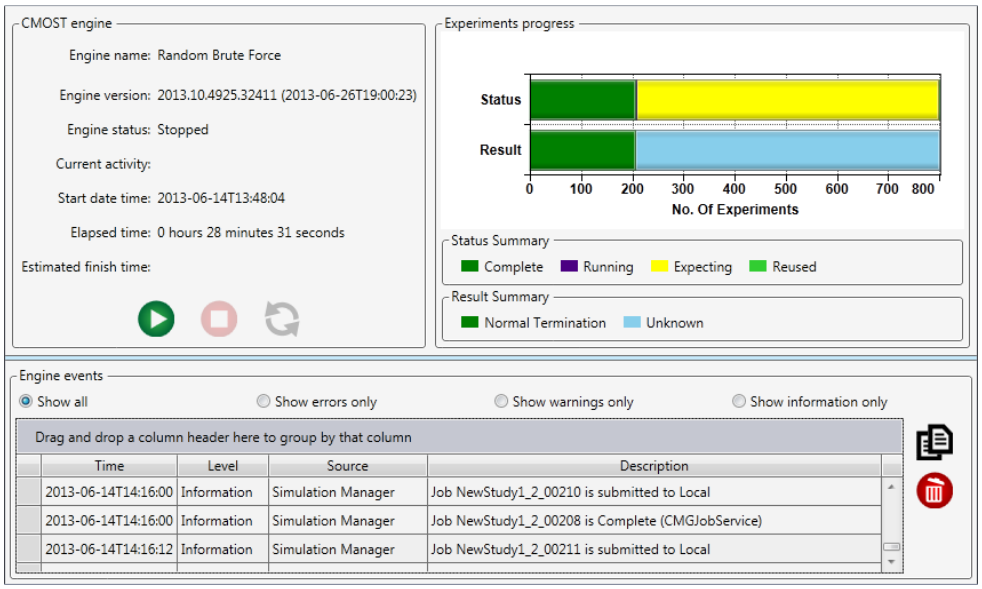
解决完这些错误后才可以启动引擎。出现警告提示是可以忽略的。可以点击 按钮来复制**Validation Error Summary** 表格中的内容到Windows 系统中的Word或Excel 文件。

在引擎启动之前，依据引擎类型， **Control Centre** 节点配置如下所示：



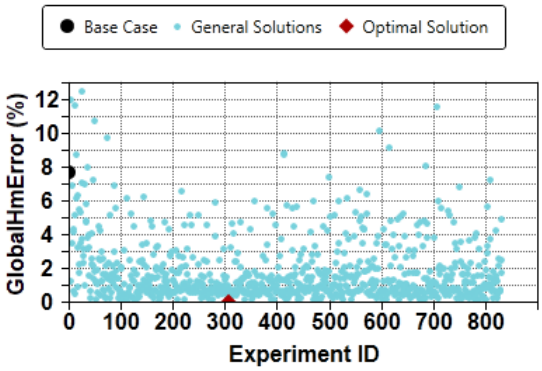
屏幕底部 **Engine events**部分，在该表格中可以根据需要显示引擎事件类型。点击Run 按钮，启动CMOST引擎。 当任务运行时：

- 根据用户的选择，显示相应的引擎事件。在下面的例子中，我们显示了所有Show all引擎事件：



注意：如果将鼠标移至运行进度条，会显示不同种类的实验数量。

注意：如果用户定义的Study类型是敏感性分析或不确定性分析，那么在**Experiments progress** 部分将包含和上图类似的CMOST任务运行进度条。如果是历史拟合或方案优化，那么在**Experiments progress**部分CMOST运行进展状况如下图所示，显示了基础方案和优化方案。如果将鼠标移至图中的数据点，则会显示试验方案ID及相应数据



- 可以打开Experiments Table 来监测试验方案及输出数据的进展情况。

- 可以打开Simulation Jobs 界面来监测CMOST任务运行状态。

- 在任何情况下，都可以通过点击**Pause** 按钮来暂停CMOST引擎。当CMOST引擎暂停时，不会将新的CMOST任务提交给Scheduler。所有未完成的CMOST任务将继续运行直到结束，CMOST会处理其结果。

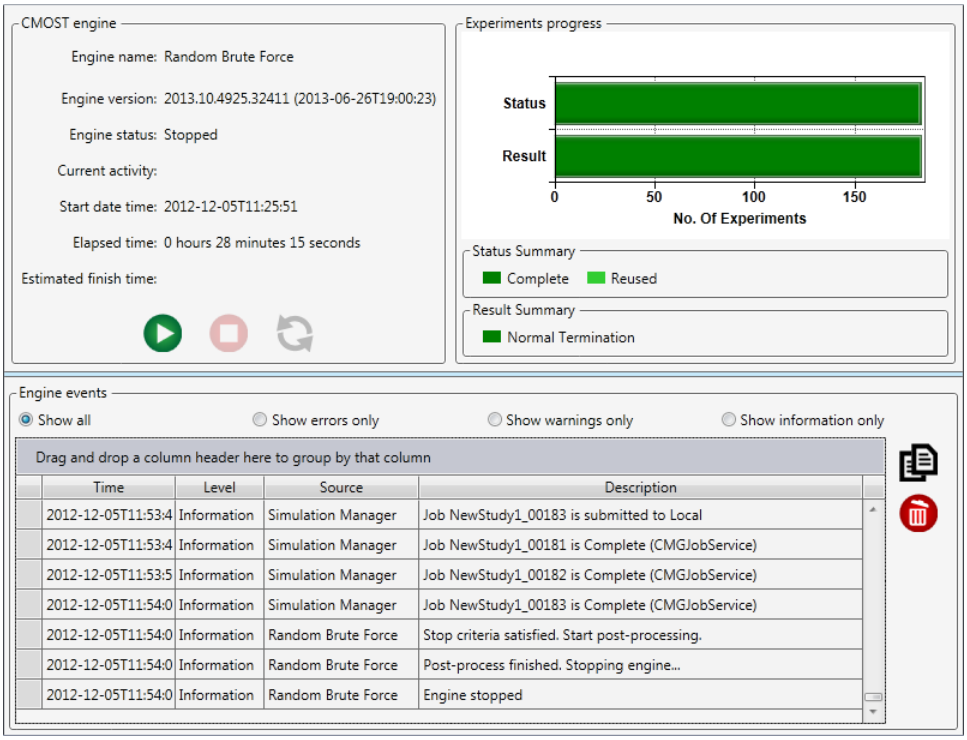
- 在任何时间点，都可以通过点击**Stop** 按钮来停止CMOST引擎。不会有新的CMOST任务提交给Schedulers。所有未完成的CMOST任务将会继续运行直到结束，然而，CMOST不再处理其结果。点击**Stop** 按钮后，可以点击**Run** 按钮来重新启动CMOST引擎。

- 在任何时间点，都可以通过点击Refresh 按钮来刷新CMOST引擎状态。如果不点击这个按钮，引擎状态会根据更新频率来自动更新。

- 可以通过点击Delete All 按钮来清除**Engine Events** 表格中的Events内容。

- 可以通过点击**Copy All Events to Clipboard** 按钮来复制**Engine Events** 表格中的内容，然后将其粘贴Excel 或 Word。

如果成功运行完成，**Control Centre** 界面显示如下：



在上面的例子中，所有的实验方案都是正常完成的。如果有些实验方案运行失败或出现异常，可以通过Simulation Jobs 界面中的Experiments Table 得到更多信息。

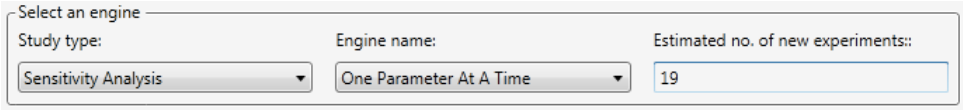
在适当的时候，可以通过Proxy Dashboard 查看可能得到的结果，通过Results & Analyses 节点查看运算和分析的结果。有些结果在CMOST运行过程中就可以查看 。

6.3 引擎设置

通过这个节点，可以定义用于Study中的引擎设置。

6.3.1 关于引擎

点击**Control Centre | Engine Settings** 子节点，设置Study类型，输入引擎名称：



注意：Estimated no. of new experiments 仅仅用于阅读，它是基于Study类型和引擎名称来评估实验方案的数量。

可以使用下面的引擎:

Engine	User Defined	SA	HM	OP	UA
Manual Engine	√				
External Engine	√				
Response Surface Methodology		√			
One Parameter At A Time		√			
CMG DECE			√	√	
Particle Swarm Optimization			√	√	
Latin Hypercube Plus Proxy Optimization			√	√	
Differential Evolution			√	√	
Random Brute Force			√	√	
Monte Carlo Simulation Using Proxy					√
Monte Carlo Simulation Using Reservoir Simulator					√

- **Manual Engine**：在用户定义Study类型的例子中，可以使用该引擎类型。使用该引擎意味着不会自动创建试验方案，所有的实验方案都是用户通过典型的实验设计、拉丁超立方或手动完成。

- **External Engine**：在使用用户定义的Study类型下，使用该引擎允许用户使用自己的优化算法。更多信息，请参考External Engine and User-defined Executable。

- **Response Surface Methodology**：对于Sensitivity Analysis 和 Uncertainty Assessment 使用典型的实验设计或Latin hypercube design，然后应用 response surface 方法。响应面（RSM）是探究输入变量（参数）与响应（目标函数）之间的关系。一组实验设计用来建立一个代理模型（近似的）。最常用的代理模型是线性或二次方程的形式。建立代理模型后， Tornado plots 显示一系列参数敏感性的强弱。更多信息参考Response Surface Methodology 。

- **One Parameter At A Time (OPAAT)**：敏感性分析时使用的传统的方法，决定参数是否敏感的因素仅仅是参数的取值范围。对于所有的参数来说，过程以此重复。更多信息参考One-Parameter-At-A-Time Sampling 。

- **CMG DECE**：CMG研发的优化算法，主要用于历史拟合和敏感性分析，可以快速的找到最优的解决方案。更多信息参考CMG DECE。

- **Particle Swarm Optimization**：用于历史拟合和方案优化，初始化时随机给一组解决方案。通过搜索空间无限接近最优的解决方案，收敛于最优的解决方案。更多信息参考Particle Swarm Optimization 。

- **Latin Hypercube Plus Proxy Optimization**：拉丁超立方设计用来创建试验方案，然后通过创建的实验方案得到训练数据，进而得到一个经验上的代理模型。利用代理模型来决定哪个是最优的实验方案。更多信息参考Latin Hypercube plus Proxy 。

- **Differential Evolution (DE)**：用于历史拟合和方案优化，初始化时随机给一组解决方案。DE 尝试着以智能的方式寻找最优解决方案。更多信息参考Differential Evolution (DE) 。

- **Random Brute Force**：用于历史拟合和方案优化，检验所有参数的组合，每次运行时，起始点和路径都是不同的。更多信息详见Random _Brute Force_Search 。

- **Monte Carlo Simulation Using Proxy**： 使用蒙特卡洛模拟，输入的参数是通过概率密度分布函数随机产生的。将这些输入的参数送到响应面模型，用于判定油藏模型的不确定性。

- **Monte Carlo Simulation Using Reservoir Simulator**：在这种情况下，从蒙特卡洛模拟选择的输入参数决定了油藏模型的不确定性。

6.3.2 常规设置

不论选择那种引擎类型，都要进行下面的常规设置：

- 引擎常规设置
  - **Auto Save Result Interval (minutes)**：当CMOST引擎运行时，定义自动保存结果的时间间隔。

Default Keep SR2 Option for New Experiments：在敏感性分析和

- **Experiments Management:**

  - **Default Keep SR2 Option for New Experiments**：在敏感性和不确定性分析下，如果设置为Yes，CMOST 将为所有的实验方案保留SR2文件。

  - **Number of Failed Jobs to Exclude an Experiment**：如果一个试验方案运行了很多次都没有完成，则该实验方案将会被排除。

  - **Number of Optimum Experiments to Keep Simulation Files**：在历史拟合和方案优化时，CMOST将保留最优方案的输入和输出文件的个数。CMOST引擎将删除其他所有方案的模拟文件。

  - **Number of Perturbation Experiments for Each Abnormal Experiment**：一个扰动试验方案是通过简单修改异常终止的实验方案产生的。对于数值控制部分非常有用，在数值控制部分做一个细微的改变，就可能是异常终止的实验方案顺利完成运算。

- **Optimization Settings**：在方案优化的算例中：

  - **Global Objective Function Name**：方案优化时总目标函数的名称。

  - **Search Direction**：如果设置为Minimize，优化的目标是找到最小的目标函数。如果设置为Maximize，优化的目标是找到最大的目标函数。

  - **Total Number of Experiments**：这是在执行方案优化时运行的最大实验方案数。一旦达到了实验总数，引擎就会停止运行。该设置在引擎运行时可以修改。

    用户定义的Study中使用额外的引擎，需要补充几项设置。更多信息参考External Engine。

- **Random Seed**：对于引擎来说，需要一个种子来产生随机的数据，用户决定是否指定该种子。如果用户不指定的话，将会使用随机种子。如果用户指定的话，就使用同样的种子，设置同样的参数得到同样的实验方案，这意味着结果是可重复的。

6.3.3 引擎特有设置

根据选择的引擎类型，设置以下部分：

6.3.3.1 CMG DECE Optimization （引擎特有设置）

- **Honour Parameter Hard Constraints**：如果设置为True，当创建新的实验方案时，引擎会遵守硬性约束条件。如果设置为False，创建的实验方案可能会违反硬性约束条件，然而它们不会被运行。

- **Continuous Parameters Sampling**：该处不是用户选择的。DECE引擎总是假定连续参数都是均匀分布的概率分布函数。
- **Discrete Parameters Sampling**：该处不是用户选择的。在历史拟合或方案优化时，DECE引擎对待离散值都有一样的取值概率。

- **Number of Initial Effect Screening Experiments**：这是DECE引擎检测参数时初始的实验方案数。它是内部计算得到的，用户不可以修改。

6.3.3.2 Latin Hypercube plus Proxy Optimization （引擎特有设置）

- **Continuous Parameters Sampling**： 在历史拟合或方案优化时，连续参数的抽样方法如下所示：
  - **Discrete Sampling Using Pre-defined Levels**：如果使用该选项，参数的取值范围被预先等分成几份，实验方案只能取离散值。
  - **Continuous Uniform Sampling within the Data Range**：参数在取值范围内均匀取值。
  - **Continuous Sampling Using Prior Distribution**：当取值时，按照事先定义好的概率分布函数进行取值抽样。
- **Discrete Parameters Sampling**：在历史拟合和方案优化时，离散参数的抽样方法如下所示：
  - **Treat Discrete Values Equally Probable**：所有参数的候选值取值概率都相等。
  - **Honour Prior Distribution of Discrete Values**：依据用户定义的概率分布函数进行抽样。
- **Proxy Model Type**:设置*RBF Neural Network* 或*Polynomial Regression*。
- **Maximum time (minutes) allowed for proxy calculations in each iteration**：允许代理模型计算和建议二次迭代的最大时间。在某些情况下，如果用户发现迭代之间需要花费较长时间，该选项可以帮助减少时间。另外，一般我们建议使用缺省值。

- **Number of Initial Proxy Training Experiments**：用来建立初始代理模型的训练方案数。它是由CMOST计算得出的，用户不能修改。

6.3.3.3 Monte Carlo Simulation Using Proxy （引擎特有设置）

当模型不确定性由一系列独立或者相关的不确定性参数表示时可以使用该选项。

- **Proxy Model Type**: *可以选择Polynomial Regression 或 RBF Neural Network。如果选择使用Polynomial Regression，会得到下面的结果：*
    - Effect Estimates (tornado plots)
    - Proxy Model Statistics (summary of fit, analysis of variance, effect screening, and polynomial equations)
    - Model Quality Check (Monte Carlo results)
    - Sobol Plot
    - Morris Plot

  如果选择使用*RBF Neural Networks*，会得到下面的结果：
    - Statistics
    - Proxy Models (weights, if the number of experiments is less than 200)
    - Monte Carlo Results
    - Sobol Plot
    - Morris Plot

  如果选择使用*Polynomial Regression*，下面的每一项可用于引擎停止运行的条件：

- **Interested Terms [not applicable for RBF NN]**：如果设置为 *Linear*，CMOST引擎将会检查是否达到了线性代理模型可接受的条件或标准，如果达到该条件，CMOST引擎会停止。如果设置为 *Linear* + *Quadratic*，CMOST引擎将会检查是否达到了线性+二次方程代理模型可接受的条件或标准，如果达到该条件，CMOST引擎会停止。如果设置为 *Linear* + *Quadratic*+ *Interaction*，CMOST引擎将会检查是否达到了线性+二次方程+中间项代理模型可接受的条件或标准，如果达到该条件，CMOST引擎会停止。
- **Acceptable R-Square [not applicable for RBF NN]**: R-square (R^2) 表示代理模型与实际观察数据之间的相关性。如果R^2 为1，表示代理模型与实际数据完全吻合（误差为0）。如果R^2 为0意味着代理模型与实际数据完全不吻合。该处定义了可接受的相关性系数数值。如果没有达到定义的该数值，并且额外实验方案的总数没有超过改善代理模型需要的额外实验方案百分比限制（Percentage Limit of Extra Experiments for Improving Proxy），CMOST 为改善代理模型精度将产生更多的实验方案。

- **Acceptable R-Square Adjusted [not applicable for RBF NN]**：R^2 校正是对R^2 的修正，主要用来调整模型中解释项的数量。不像 R^2，R^2 校正仅仅增加了新解释项来改善代理模型。R^2校正可以是负值，通常它小于等于 R^2。该处定义了可接受的R^2校正数值。如果没有达到定义的该数值，并且额外实验方案的总数没有超过改善代理模型需要的额外实验方案百分比限制（**Percentage Limit of Extra Experiments for Improving Proxy**），CMOST 为改善代理模型精度将产生更多的实验方案。

- **Acceptable R-Square Prediction [not applicable for RBF NN]**：R^2 预测表示代理模型预测响应的精度。取值范围0～1，该值越大，表示精度越高。该处定义了R^2 预测的可接受数值。如果没有达到该数值，并且额外实验方案的总数没有超过改善代理模型需要的额外实验方案百分比限制（Percentage Limit of Extra Experiments for Improving Proxy），CMOST引擎为改善代理模型精度将产生更多的实验方案。

- **Acceptable Relative Error of Proxy Verifications (%)**：该处定义了验证实验方案的最大可接受误差。如果没有达到该数值，并且额外实验方案的总数没有超过改善代理模型需要的额外实验方案百分比限制（Percentage Limit of Extra Experiments for Improving Proxy），CMOST引擎为改善代理模型精度将产生更多的实验方案。
- **Percentage Limit of Extra Experiments for Improving Proxy (%)**：该条件或标准决定了满足上述条件的最大实验方案数。例如，对某一Study，期望的实验方案数为100，如果为改善代理模型精度需要额外实验方案百分比限制为25%，那么最多还能再添加25个额外的实验方案用来改善代理模型精度，满足上述停止引擎的条件或标准。

如果选择*RBF Neural Networks* 代理模型，**Acceptable Relative Error of Proxy Verifications (%)** 和 **Percentage Limit of Extra Experiments for Improving Proxy (%)** 可以用来作为引擎运行停止的标准。

6.3.3.4 Monte Carlo Simulation Using Simulator （引擎特有设置）

当模型是由离散的不确定性认识（例如，地质模型或历史拟合模型）表示时，应该使用该不确定性方法。在这种情况下，为不确定性分析使用代理模型是不恰当的，因为离散的认识不能被连续的数值所表征。

如果选择该引擎类型，你可以在引擎自动停止之前定义执行蒙特卡洛模拟方案数量（Number of Monte Carlo Simulations）。

6.3.3.5 One-Parameter-At-A-Time

- **Reference Case Parameter Values**：如果设置为*Use Parameter Median Values*，那么参数参考值使用中间值。如果设置为*Use Parameter Default Values*，那么参数的参考值使用缺省值。
- **Continuous Parameter Testing**：如果设置*Test All Discrete Levels*，那么从每个参数不同级别抽取数值用于产生实验方案。如果设置为*Test Lower and Upper Limit Only*，那么每个参数仅仅使用最大和最小值用于生产实验方案。
- **Discrete Parameter Testing**：如果设置*Test All Candidate Values*，那么每个候选值都用于产生实验方案。如果设置为*Test Lower and Upper Bound Only*，

那么每个参数仅仅使用最大和最小值用于生产实验方案。

6.3.3.6 Particle Swarm Optimization (PSO) [引擎特有设置]

更多信息参考Particle Swarm Optimization 。

- **Inertia Weight**：该设置必须在0.4 和 0.9之间。较大的惯性权重因子用于全程搜索，同时较小的惯性权重因子适用于局部搜索。
- **Cognition Component (C1), Social Component (C2)**：Cognition and social components in the PSO. 这些设置控制算法的探测/开发能力。在历史拟合中，广泛地探测查找未知参数中所有可能的取值组合，通过深度查找先前没有探测的区域得到一个最好的拟合方案。这些参数的低值支持探测的搜寻空间，但是这样会影响收敛性。强烈推荐C1 和 C2取值在 1 和 2之间。
- **Population Size**：缺省值设置为20 。当模拟方案总数较多时，使用较大的群体规模。建议 PSO 迭代次数（模拟方案总数除以规模大小）大于20。

当你选择Particle Swarm Optimization，在窗口底部可以选择Pareto Front Particle Swarm Optimization，配置以下部分：

- **Second Global Objective Function** （必要）：从列表中选择总目标函数。
- **Second Search Direction**：选择*Minimize* 或 *Maximize*。
- **Maximum Number of Pareto Leaders**输入想要查找的帕累托前导字符的最大数量。
- **Third Global Objective Function （可选）**：从列表中选择总目标函数。
- **Third Search Direction**：选择 *Minimize* 或*Maximize*。

更多信息，参考Pareto Particle Swarm Optimization 。

6.3.3.7 Differential Evolution (DE) [引擎特有设置]

更多信息，参考Differential Evolution：

- **Scaling Factor (F)**：该参数比例因子F ∈ [0,4]，缺省值为0.5。
- **Crossover Rate (Cr)**：该参数，交叉概率Cr ∈ [0,1]，控制群体的多样性。缺省设置为 0.8 。
- **Population Size (Np)***：该参数，群体规模大小Np ∈ [4, 200]，缺省值30。*

6.3.3.8 Random Brute Force Search （引擎特有设置）

更多信息，参考Random Brute Force Search：

- **Continuous Parameters Sampling**：在历史拟合或方案优化的情况下，连续参数按照以下方法进行抽样：

  - **Discrete Sampling Using Pre-defined Levels**：如果使用该选项，参数的取值范围被平分为若干个级别，然后从若干个级别中选离散值组成实验方案。
  - **Continuous Uniform Sampling within the Data Range**：参数在取值范围内均匀抽样。
  - **Continuous Sampling Using Prior Distribution**：抽样时考虑用户定义的先验概率分布函数。

- **Discrete Parameters Sampling**：在历史拟合或方案优化时，需要考虑离散参数取值，其方法：

  - **Treat Discrete Variables Equally Probable**：所有候选值按照相同的概率取样。
  - **Honour Prior Distribution of Discrete Values**：按照用户定义的先验概率分布函数进行抽样。

6.3.3.9 Response Surface Methodology （引擎特有设置）

- **Interested Terms**：如果设置为*Linear*，CMOST引擎会核实线性代理模型可接受的条件。如果满足该条件，CMOST将会停止运行。 如果设置为 *Linear* + *Quadratic*，那么*Linear* + *Quadratic* 多项式代理模型将会被核查是否满足停止引擎运行的条件。如果设置为*Linear* + *Quadratic* + *Interaction*，那么 *Linear* + *Quadratic* + *Interaction* 多项式代理模型将会被核查是否满足停止引擎运行的条件。
- **Acceptable R-Square**：*R-square (R^2)*表示代理模型proxy model 的精度。R^2 为1表示完全拟合。R^2 为0 表示完全拟合不上。该处定义可接受的精度R^2 。如果没有达到设定的拟合精度，并且运算的实验方案总数的比例没有超过为了改善代理模型精度而产生额外实验的百分比，CMOST引擎将会产生更多的实验方案来改善代理模型精度。
- **Acceptable R-Square Adjusted**： *R-square (R^2)*校正是对*R-square (R^2)* 的修正，用来修正模型中的解释项。与R^2不同，R^2 校正仅仅增加了新解释项来改善代理模型。 R^2校正可以是负值，总是小于或等于R^2。该处定义的数值被认为是可接受的。如果没有达到这个数值，并且运算的实验方案总数的比例没有超过为了改善代理模型精度而产生额外实验的百分比，CMOST引擎将会产生更多的实验方案来改善代理模型精度。

- **Acceptable R-Square Prediction**：*R-square (R^2)*预测表示代理模型预测的精度。变化范围是从0到1，较大的值表示预测精度较高。该处定义的数值被认为是可接受的。如果没有达到这个数值，并且运算的实验方案总数的比例没有超过为了改善代理模型精度而产生额外实验的百分比，CMOST引擎将会产生更多的实验方案来改善代理模型精度。

- **Acceptable Relative Error of Proxy Verifications (%)**：该处定义了验证实验的最大可接受误差。如果没有达到这个值，并且运算的实验方案总数的比例没有超过为了改善代理模型精度而产生额外实验的百分比，CMOST引擎将会产生更多的实验方案来改善代理模型精度。

- **Percentage Limit of Extra Experiments for Improving Proxy (%)**：该停止运算的标准决定了满足上述所有条件需要产生的最大实验方案数。例如，对某一Study来说，期望的实验方案数是100。如果为改善代理模型精度而产生额外实验的百分比设置为25%，那么最多可以添加25个实验方案用于改善代理模型精度来满足停止运算的条件。如果运算了125个实验方案，仍未满足停止运算的条件，CMOST引擎也将停止运算。

6.3.3.10 External Engine and User-defined Executable (引擎特有设置)

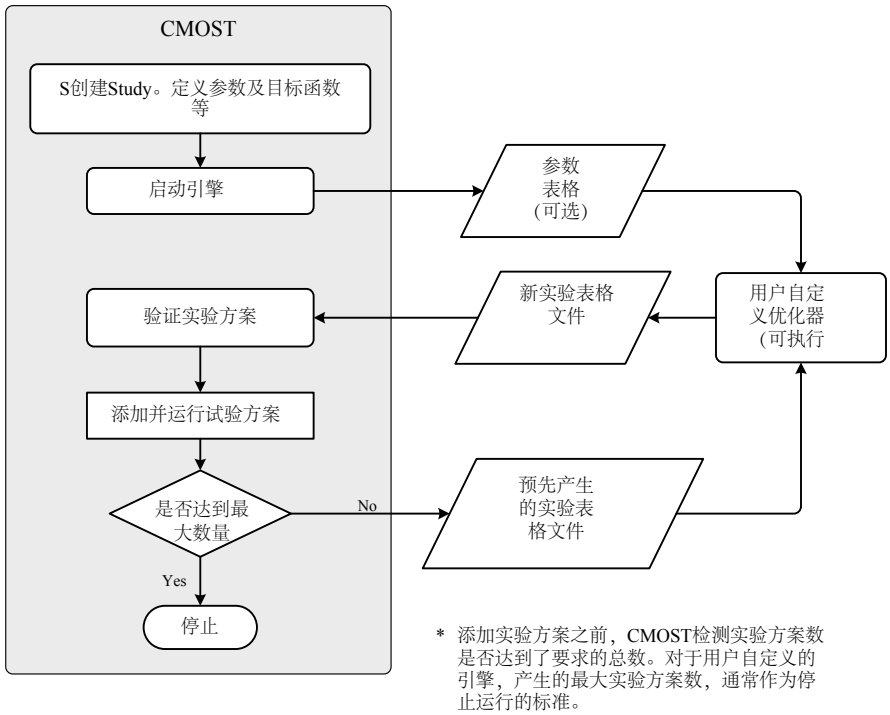
在用户自定义优化模拟器时，需要定义CMOST与优化模拟器之间的接口，尤其是**Engine Settings** 和 **Input/Output Tables**。这些设置用来决定表格文件的存放位置，这些表格文件可能用于定义外部引擎。

工作流程：

1. CMOST 调用模拟器，为所有的实验方案计算目标函数值。
2. 当所有存在的实验方案运算完成，CMOST以CSV文件格式输出实验表格，调用用户定义的优化模拟器：

   user.exe [userargument] iteration_number iteration_number 以 0开始，如果有必要，用户可以自定义userargument.
3. 用户自定义的优化模拟器读取文件并执行优化运算。
4. 用户自定义的优化模拟器推荐一系列新的实验方案。被推荐的实验方案以CSV格式写入，用户自定义执行接口。
5. CMOST读取新的实验方案表格文件， 如果产生的实验方案数没有超过最大的实验数，就会继续产生实验方案，并将其提交给模拟器。到第二步。

用户定义的优化模拟器工作流程如下：



当代实验表格文件

试验表格中的所有实验运算完成后，CMOST将所有试验方案的信息输出到该文件夹。表格是以CSV（逗号分隔值）形式输出。在CMOST Experiments Table 界面中所有的列，如下所示：

ID,Generator,Status,Result Status,Proxy Role,Keep SR2,Has SR2,Highlight,Para1, Para2, Para3,Para4,Obj1,Obj2…
18,External Engine,5,4,0,0,False,False,0.5,200,1,0.45, 3450.3,54.4, …
19,External Engine,5,4,0,0,False,False,1.9,300,1,0.21, 2980.3,125.8, …
20,External Engine,5,4,0,0,False,False,1.0,200,3,0.45, 1480.3,50.2, …

用户自定义执行文件

该执行文件包含用户的优化算法。当目前的实验方案完成运算后被调用。用户自定义可执行文件的主任务包括但不局限于以下几点：

1. 从**Previous Generation Experiments Table** 文件读取先前实验方案的结果。

2. 分析结果，如有必要推荐一系列新实验方案。
3. 将推荐的实验方案写入New Experiments Table 文件，由CMOST读取。

注意创建用户自定义的可执行文件：

1. 用户自定义的可执行文件可以使用任何编程语言进行编辑。
2. 用户自定义的可执行文件必须使用一套基础的生成算法；例如在每个生成算法中，其方案总数必须是个常数。
3. 当CMOST调用用户自定义的可执行文件时，迭代步数作为唯一的命令变量提交给可执行文件。
4. 用户定义的可执行文件可以保存信息，或简化输出该信息到调式文件中。
5. 用户定义的可执行文件的最大允许时间为60分钟。
6. 在完成运算后，用户定义的可执行文件必须能够静静的存在。

新实验表格文件

新实验表格文件是由用户自定义可执行文件输出的CSV（逗号分隔值）表格文件。推荐的实验方案必须等于总数大小。表格的第一行是由逗号分开的参数名称，其他行是推荐的实验方案。
注意，一些不合法或重复的实验方案将会被拒绝，由CMOST定义的随机实验方案所代替。

新实验表格的例子：

```
I Para1, Para2, Para3, Para4
1.9, 200, 0.56, 2 1.5, 300,
0.32, 1 0.5, 100, 0.21, 3
```

参数表格文件

外部引擎开始运行后， CMOST 将写入一个参数表格文件。用户定义的可执行文件能够从该文件读取参数信息。用户可以修改执行文件中的硬性编码码参数信息，这样的话，参数表格文件将被忽略。

每个参数的信息输出到文件中呈一行。可以使用不同的形式，主要决定于参数类型：

- 对离散整数和离散实数的参数来说，其信息格式如下：

  ParamterName, Source, candidateValue1, candidateValue2, …, candidateValueN

- 对离散文本参数，以下面的格式输出：

  ParamterName, DiscreteText, candidateNumericalValue1, candidateNumericalValue2, candidateNumericalValue3…, candidateNumericalValueN

- 对连续实数，其格式：

  ParamterName, ContinuousDouble, minValue, maxValue

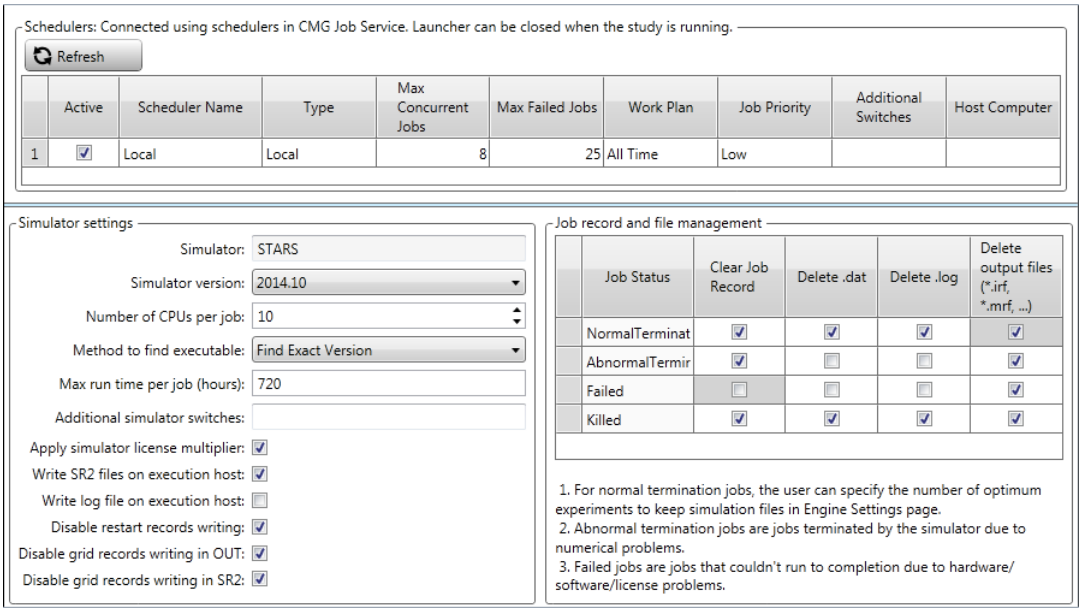
下面是参数表格文件内容的例子：

Para1, DiscreteDouble,0.5,1.0,1.5,1.9 Para2, DiscreteInteger,100,200,300 Para3, ContinuousDouble,0.1,0.6 Para4, DiscreteText,1,2,3

6.4 模拟设置

在运行CMOST模拟任务之前，需要配置**Simulation Settings**界面，如下所示：

- Schedulers
- 模拟器版本
- 运行每个任务需要的CPU
- 最大的模拟运行时间
- 任务记录和文件管理



如上图所示，**Simulation Settings**界面包含三部分：

6.4.1 Schedulers

通过Schedulers 部分，可以分别调整每个scheduler 的设置。

Schedulers配置表格内容如下：

- **Active**： Active 选项框决定了CMOST是否激活该Scheduler。如果选中了Active 选项框， CMOST将使用该Scheduler；否者不适用该Scheduler。

注意：如果使用了 Local 以外的**Schedulers** ，所有的Study必须存储在UNC（Universal Naming Convention常用命名约定）目录里。

- **Scheduler Name**：Scheduler名字在Scheduler Name 列中显示。‘Local’ 表示打开当前Project的计算机。该信息不能在CMOST里面编辑。如果需要修改该信息，需要通过Launcher来实现。

- **Type**：Scheduler 类型如下所示：
  - Local（本地计算机运行CMG任务）
  - CMG Drone Scheduler （远程计算机运行CMG任务）
  - MSCC Scheduler （微软Windows计算机群）
  - LSF Scheduler （平台计算LSF）
  - SGE Scheduler （Sun 网格引擎）

- **Max Concurrent Jobs**：不同的任务可以在不同的 Scheduler 上面运行。Max Concurrent Jobs 可以编辑，因此CMOST将一定数量的任务分配给每个scheduler。缺省值是1，当改变这个数值时，需要考虑：

  - 每个Scheduler的处理器数量：N
  - 每个任务需要的处理区数量：nj
  - 该Scheduler是否被其他用户共享？

  Max Concurrent Jobs 不应该大于N/nj。如果某个Scheduler 被其他用户共享，应该限制**Max Concurrent Jobs** 数量来阻止自己使用所有的处理器。

注意:

1. 正在运行和排队的任务都是经过考虑后挂起的。
2. 如果历史拟合或方案优化时，使用CMG DECE优化算法，建议所有Scheduler使用的Max Concurrent Jobs（最大同时运行的任务数）不超过10个。 这是因为如果有太多的任务同时运行，优化器就不能在找到最优模拟方案的同时快速减少模拟任务数量。另一方面，如果目标仅仅是为了减少运行时间，该建议可以忽略。

如果该值设置为0，就不会运行任何CMOST任务；然而如果某一特殊的Scheduler不执行CMOST任务运算，可以将其前面的激活复选框去掉，这样就会避免混淆。

- **Max Failed Jobs**：如果一定数量的CMOST任务运算失败，那么Schedulers就会停止CMOST任务的提交。由于硬件或软件的原因，失败的任务不能重新运算，例如：
  - 网线被拔出
  - 关闭计算机
  - CMG 服务器许可停止运行
  - 没有安装模拟器，或
  - 没有可用的CMG许可

  由于数值控制问题导致模拟器任务意外终止的不属于失败的任务。由于硬件或软件问题， CMOST将停止提交任务到 scheduler，缺省最大任务失败个数为25。
- **Work Plan**：CMOST可以用来设置在不同时间提交任务到不同的Schedulers。在工作计划中有三个选项：
  - All Time
  - Evenings and Weekends
  - Weekends

  如果 **Work Plan**设置为 *All Time*，把任务列入计划中没有任何限制，该选项是缺省选项。

  如果 **Work Plan** 设置为*Evenings and Weekends*，那么提交任务的时间为周一至周五的晚八点到早六点以及周末48小时。该选项适用于工作日正常工作时间计算机正常使用，而工作日晚上及周末不使用计算机的用户。

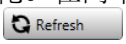
  如果 **Work Plan** 设置为*Weekends*，那么任务只有在周六及周天全天提交。该选项适用于计算机在工作日正常使用，而在周末空闲的用户。

注意：在工作计划中， 需要一段时间运行取值范围之外的任务。工作计划仅仅保证设定的时间范围之内，不向Scheduler发送模拟任务。如果有必要的话，可以在Launcher界面中Kill掉模拟任务。一旦Kill掉模拟任务，CMOST将会通知并采取恰当的行动。对敏感性分析和不确定性分析，新任务中计划使用相同参数取值。对历史拟合和方案优化，被Kill掉的任务会被忽略，依据优化的过程，会计划一个新任务。

- **Job Priority**: 不同的scheduler可以设置不同的优先级。如果有多个任务在scheduler中排队，优先级别高的先运算。缺省的优先级别是Low。

- **Additional Switches**：如果需要Scheduler开关，就需要输入额外开关所在列。更多关于Scheduler开关信息查看Launcher 用户手册。

- **Host Computer**：主机列显示运行的Scheduler。如果选择的是Local scheduler，则没有Host Computer。 该信息不能在CMOST中编辑。如果需要修改，应该在Launcher界面编辑。

- **Refresh**：通过Launcher 添加或删除一个Sscheduler，在Scheduler 表格可以通过更新显示这些变化。在两个计算机之间复制Study文件时Scheduler也需要更新。点击Refresh 

按钮来更新Scheduler表格。如果添加了新的Schedulers ，它们将会使用缺省值添加至表格。

6.4.2 模拟器设置

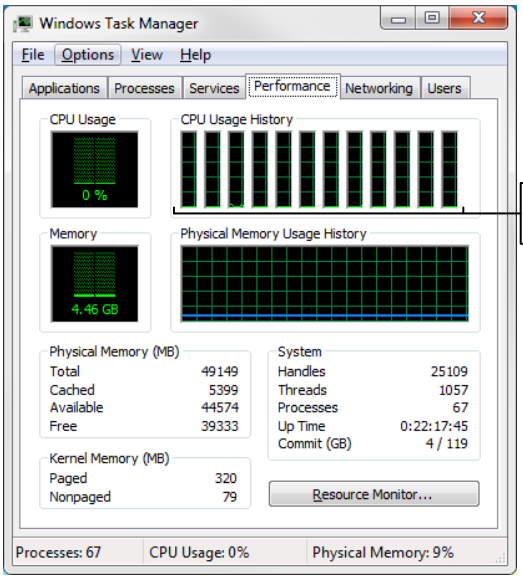
在Simulator Settings 配置以下部分：

- **Simulator**：指定CMOST使用的模拟器。只读。
- **Simulator Version**：如果有多个模拟器版本安装在本地计算机，将会显示不同版本模拟器列表。推荐在所有计算机的scheduler安装相同版本的模拟器。
- **Number of CPUs per job**：定义执行每个任务使用的处理器个数。

注意：这个数值不能大于scheduler处理器数。

如果在本地计算机上运行，右键Windows 任务栏，然后选择**Properties**。选择**Start Task Manager**。在**Performance** 选项中显示可用的CPU数。在该处输入某一数值。

在下面的例子中，可用的CPU数为12。



- **Method to Find Executable**：找到可执行文件的方法定义了使用远程计算机的模拟器。缺省的是CMOST使用的是在Simulator 部分的模拟器版本。如果在远程计算机不存在该版本，CMOST将其他方法找到其他可供选择的模拟器版本。以下三个选项可选：

  - Find Closest Version
  - Find Latest Version
  - Find Exact Version

  Find Closest Version 选项试着查找最接近基础模型模拟器的版本。*Find Latest Version* 选项试着找到远程计算机最新的版本。如果使用 *Find Exact Version* 选项，仅仅使用定义的模拟器选项，如果该版本不存在的话，则不能向Scheduler提交作业。

- **Max Run Time per Job (hours)**：每个任务允许运行的时间。如果在定义的时间内，没有完成CMOST任务的运算，该任务会被CMOST引擎Kill掉。缺省的最大运行时间是720小时（30天）。
- **Additional Simulator Switches**：如果需要模拟开关，可以通过**Additional Simulator Switches** 文本框输入。更多信息可参考模拟器和Launcher 用户手册。
- **Apply Simulator License Multiplier**：如果选择使用CMOST提交给模拟器进行模型运算，会使原有许可数量倍增。例如，对同样许可数量来说，使用CMOST提交至IMEX模拟器时，仅占用许可数量的¼，提交至GEM和STARS模拟器时，仅占用许可数量的½ 。

- **Write SR2 Files on Execution Host**：如果选择该选项，在执行运算的计算机上将SR2文件写入临时文件夹，当模拟完成后将文件夹复制到Study文件夹 。如果不选该选项，SR2文件被直接写入Study文件夹。

- **Write Log File on Execution Host**：如果选项该选项，在模拟过程中，将模拟运算记录文件（log文件）写入主机中的临时文件夹，当模拟完成后将文件夹复制到Study文件夹。如果不选该选项，模拟运算记录文件（log文件）被直接写入Study文件夹。
- **Disable restart records writing**：缺省值，选择该处表示重启动记录不会写入SR2文件。
- **Disable grid records writing in OUT**：缺省值，选择该处使这些文件尽可能的小，网格参数不被写入.OUT文件。
- **Disable grid records writing in SR2**：默认选中，使SR2文件尽可能的小。因为网格参数没有被写入SR2文件，在Results 3D 中仅仅能够看到第一个时间步的场图。

6.4.3 任务记录和文件管理

通过**Job Record and File Management** ，可以配置有多少CMOST任务记录在Launcher和.OUT文件。例如，可以命令CMOST对于非正常终止的任务是保留或删除模拟结果文件（.irf、.mrf以及.out)），如下所示：

Job Record and File Management

Job Status	Clear Job Record	Delete .dat	Delete .log	Delete output files(*.irf, *.mrf, ...)
NormalTermination	☑	☑	☑	☑
AbnormalTermination	☑	☐	☐	☑
Failed	☐	☐	☐	☑
Killed	☑	☑	☑	☑

1. For normal termination jobs, the user can specify the number of optimum experiments to keep simulation files in Engine Settings page.
2. Abnormal termination jobs are jobs terminated by the simulator due to numerical problems.
3. Failed jobs are jobs that couldn't run to completion due to hardware/software/license problems.

6.5 实验表格

依据引擎设置，CMOST自动产生一系列实验方案，手动或额外引擎除外。也可以根据自己的算法手动添加额外的实验方案。在Study运行过程中可以通过**Experiments Table**查看运算结果。该部分提供以下信息：

- Navigating the Experiments Table

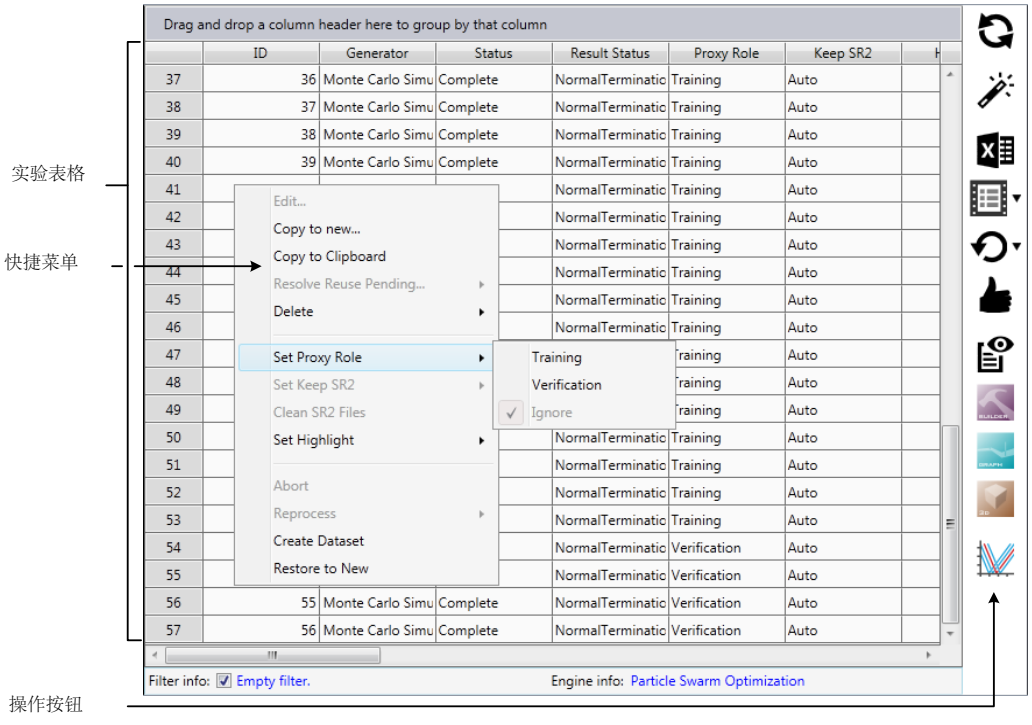
- Creating and Importing Experiments，除此之外，CMOST依据引擎设置可以自动创建试验方案。
- Configuring the Experiments Table，可以配置Experiments的外观，尤其是表格列的顺序，另外，还可以根据某一列或某几列来配置行的顺序。

- Checking Experiment Quality，可以检查试验方案的正交性。
- Exporting the Experiment Table to Excel.
- Viewing the Simulation Log.
- Clearing the SR2 Files
- Reprocessing Experiments.

6.5.1 实验表格操作

试验表格（Experiments Table）如下所示，将包含以下试验：

- 根据引擎设置自动产生CMOST实验方案。
- 通过CMOST实验设计工具（例如拉丁超立方设计）产生实验方案。
- 手动输入实验方案（用户自定义实验方案）。



6.5.1.1 实验表格列

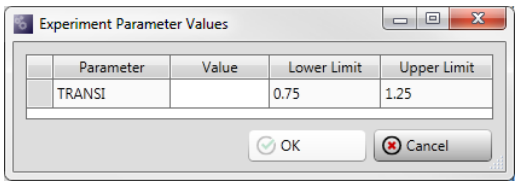
注意：实验表格中的数据是否需要编辑由其实验状态决定。对于新实验方案中，**Proxy Role**、**Keep SR2**、 **Highlight**以及**Comments** 列是可以编辑的。对于已完成的实验方案，仅仅有**Highlight**、**Proxy Role** 以及 **Comments** 是可以编辑的。**Proxy Role** 和**Keep SR2** 列可以通过右键试验方案，然后选择需要的菜单栏进行编辑。

- **ID**：唯一的实验ID号码，创建试验方案后，由CMOST分配。如果随后实验方案被删除，则ID号码将不会再使用。
- **Generator**：该发生器用来产生实验方案，例如：*LatinHyperCube*（拉丁超立方）、 *Response Surface Methodology*（响应面）、或用户自定义。
- **Status**：当试验方案运行时，其状态：

  - **Reuse Pending**：该状态表示创建试验方案后再添加新参数。试验表格如果有再使用-待定试验，需要在启动CMOST引擎之前解决掉这些实验方案。通过提供未知的参数值，为每个实验方案解决**Reuse Pending** 状态，如下：

    a. 右键点击某一个或某几个再使用-待定试验，如有必要可以使用CTRL和SHIFT键。

    b. 选择**Resolve Reuse Pending**，然后**Resolve Selected Experiment(s)**。**Experiment Parameter Values**对话框中参数未知数值如表所示：

    注意： 如果选择**Resolve Reuse Pending | Resolve All Experiments**，通过这单一操作可以解决所有的***Reuse Pending*** 状态。

    c. 在**Value** 列，在选择的实验方案中为新参数输入数值。

    d. 点击 **OK** 应用。

    例如，如果运行了100个实验方案，然后添加一个新参数，这100个实验方案的状态是Reuse Pending 。这些实验方案和新参数使用缺省值已经运算完成，应该为这100个实验方案输入新参数的数值。

  - **New**： 新实验方案已经添加，但数据文件还未创建。

- **Creating dataset**：CMOST正在为实验方案创建模型数据文件。
- **Dataset created**：CMOST 为实验方案创建模型数据文
- **Running**：CMOST正在提交模型数据文件给Launcher。
- **Complete**：CMOST从Launcher接收SR2文件。
- **Reused**：在当前的Study中，重新使用先前已经完成运算的实验方案。
- **Aborted**：在实验完成运算之前，可以在任何时间点，手动中止实验方案运算。中止运算的实验方案被CMOST忽略。例如，中止实验方案的结果不用来决定代理模型或最优开发方案。

- **Result Status**：实验方案模拟结果的状态：
  - **Unknown**：CMOST 没有检查.log、 .out以及 .irf 文件。
  - **Incomplete**：模拟未完成。
  - **Exceed max run time**：实验超出了Simulator Settings中定义的最大运算时间。
  - **Abnormal termination**：在到达指定停止运行时间之前，CMOST任务意外终止。
  - **Normal termination**：模拟运算至最后一个时间点，生成OUT文件。 CMOST 将会清除掉Launcher中的记录，从OUT文件中得到必要的信息，然后根据在**Simulation Settings** 界面**Job Record and File Management**表格中的定义来处理这些信息。
  - **Violate hard constraints**：如果违反了硬性约束条件，不允许进行模拟运算。更多信息参考Hard Constraints 。
  - **Waiting to be re-processed**：该状态显示，当运算完实验方案，修改单位制、矿场数据、基础数据或目标函数，CMOST需要重新运算目标函数。启动引擎将自动处理这些实验方案或可以迫使一个或多个实验方案重新运算，如下：
    1. 点击想要重新运算的某一个实验方案，如有必要可以使用CTRL键来选择多个实验方案。
    2. 右键选择实验方案，然后选择 **Reprocess | Reprocess Selected Experiment(s)** 或点击**Reprocess Experiment(s)**  按钮然后选择 **Reprocess Selected Experiment(s)**.

3. CMOST将试着重新计算所有试验方案的目标函数。

- **Re-process failed**：对于该实验，CMOST找不到足够的数据重新计算目标函数。例如，添加了一个新目标函数后，所有的实验方案需要重新处理。如果SR2文件被删除，VDR文件不包含计算新目标函数的数据，这时就可能发生重新计算失败。

- **Proxy Role**：如果设置为*Training*，该实验方案结果用来训练代理模型。如果设置为*Verification*，该实验方案结果用来验证代理模型的精度。如果设置为*Ignore*，该结果既不用于训练目标函数也不用于验证代理模型精度。关于CMOST代理模型的更多信息参考Proxy Dashboard 和 Proxy Analysis。

- **Keep SR2**：如果设置为*Auto*，CMOST将坚持模拟设置。如果设置为*Yes*，CMOST不论模拟器设置如何，将保留模拟器运算实验方案的SR2文件。如果设置为*No*，CMOST不论模拟器设置如何，在试验方案运算完成后删除SR2文件。

- **Has SR2**：如果模拟结果（SR2）文件生成后，CMOST没有删除结果文件，该选项框被选上。该选项框仅仅只能阅读，不能编辑。

- **Highlight**：可以在CMOST曲线中高亮显示某一实验方案。如果选择 Highlight，实验方案模拟结果在图中显示紫色曲线 ——— 和紫色数据点◆。

- **Parameter Value**：输入或生成的参数候选值（数值、文本或其他）在**Experiments Table**中显示，参数名称在列头显示。如果参数的来源定义为*Formula*，则显示公式计算的结果。如果参数没有被激活，则该参数使用缺省值。不论参数激活与否，都在试验表格中显示。

- **Objective Function Values**：计算的每个目标函数值都呈现在试验表格中，目标函数的名称在列头显示。

- **Execution Node**：试验方案在主计算机上运行。
- **Dataset Path**：基础模型的名称及路径在此显示。在大部分情况下，正常结束运行的CMOST任务，其 .dat、.log以及OUT文件会被删除。因为这是在**Simulation Settings** 界面**Job Record and File Management**部分的缺省设置。

- **Optimal**：通常仅有一个最优方案。例如，如果目标函数设置为 NPV，优化的方向是Maximize （该处在 Engine Settings 界面设置），拥有最大NPV 的就

是最优实验方案。在运行的过程中，最优方案是在时时变化的。 Optimal 选项框仅仅只读，不能编辑。缺省的最优方案在图中显示不同的颜色，所以很容易分辨。

在某些情况下，多个实验方案可能有相同的目标函数值。当某些参数对目标函数影响较小时，就可能导致出现多个最优实验方案。

- **Comment**：该处只读。

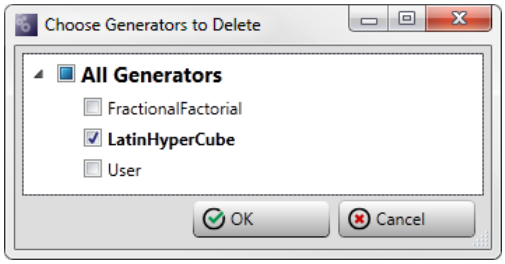
6.5.1.2 快捷菜单

一旦选中Experiments Table，可以选择一个试验（选中的实验会显示蓝色底纹），右键实验行显示快捷菜单，然后选择某一菜单项来执行下面的操作：

- **Edit**：如果选择用户自定义的实验方案可以执行该功能。如果选择生成的实验方案则不可执行该功能。

  如果选择用户自定义的实验方案，将会显示Experiment Parameter Values对话框，通过此对话框可修改实验方案参数值。点击OK后，在表格中显示新数值。
- **Copy to new**：复制选择的实验方案到用户自定义实验方案，然后在新实验方案改变参数，确保它是唯一的。
- **Copy to Clipboard**：复制实验方案表头，选择实验方案行到Windows剪贴板，然后将其粘贴到Excel等。
- **Resolve Reuse Pending**：如果实验方案状态改为*Reuse Pending*，因为，例如一个新参数添加到Study，就可以使用该状态栏。点击它，将建议用户输入未知参数值，因此可以被CMOST中重新使用。更多信息参考Status – Reuse Pending。

- **Delete**：如果选择一个生成的实验，将有个选项**Delete all Experiments in Generators**。如果继续进行会出现对话框**Choose Generators to Delete dialog**如下面例子所示：



在上面的例子中， **Generator Name** 选择**LatinHyperCube**。如果点击OK，那么所有由LatinHyperCube generator 生成的实验方案全部被删除。

如果选择用户自定义实验方案，然后右键点击Delete，仅仅能删除选择的实验方案，或者通过Choose Generators to Delete 对话框，删除所有用户自定义的实验方案。

可以通过CTRL和SHIF键删除多个用户自定义的实验方案。

- **Set Proxy Role**：改变实验方案的代理角色。例如，如果试验方案设置为*Training*，可以把它更改为*Verification* 或 *Ignore*。当设置为*Training*时，实验方案的模拟结果用于训练代理模型。如果设置为*Verification*，其结果用于验证代理模型的精度。如果设置为*Ignore*，该模拟结果既不用于训练代理模型也不用于验证代理模型的精度。关于CMOST代理模型的更多信息参考Proxy Dashboard 和 Proxy Analysis 。

- **Set Keep SR2**：如果设置为*Auto*，CMOST将根据 **Simulation Settings** 界面中**Job Record and File Management** 表格的设置来决定是否保留实验方案的模拟结果。如果设置为*Yes*，在试验方案运算完成后，无论模拟设置与否，都将保留其运算结果。如果设置为*No*，在试验方案运算完成后，无论模拟设置与否，都将不保留其运算结果。
- **Clean SR2 Files**：删除SR2文件，节省磁盘空间。
- **Set Highlightt**：如果设置为 *False* （缺省），图中不会高亮显示试验方案模拟结果。如果设置为*True*，图中会高亮显示试验方案模拟结果。在图中可以同时高亮显示多个实验方案的模拟结果。通过**Highlights**选项框，该设置可以直接输入到表格。

- **Abort**：无论实验什么方案状态，都可以更改为*Aborted*。如果实验方案还没有运算，则不会被运算。如果实验方案已经被提交给模拟器，该任务也会被Kill。
- **Reprocess**：为实验方案重新运算所有目标函数。
- **Create Dataset**：利用实验参数创建模型。如果选择定义好的实验方案路径，可以点击**Launch Builder**按钮，用Builder打开模型。这样操作的目的仅仅是为了查看模型，因为CMOST不会知道通过Builder做了哪些改变。

- **Restore to New**：该设置是将一个复杂的实验方案的状态恢复为 New。保持同样的参数值，然而所有的目标函数值都被清除掉。

6.5.1.3 操作按钮

下面提供的操作按钮位于Experiments Table右面。

按钮	名称	操作
	刷新	如果从其他Study导入实验方案，或许需要点击该按钮刷新实验方案。

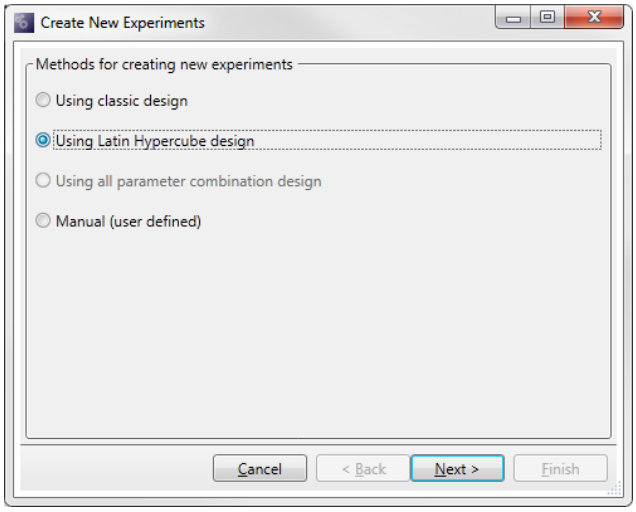
按钮	名称	操作
	创建试验方案	手动创建试验方案，CMOST引擎自动生成除外。
	输出至表格	输出表格内容到Excel文件。参考 Exporting the Experiment Table to Excel.
	配置表格	配置列或应用过滤器，概述详见 Configuring the Experiments Table
	再处理实验方案	为所有或选中的实验方案重新运算目标函数。
	检验精度	打开 Experiments Quality窗口查看实验方案的正交性。更多信息参考 Checking Experiment Quality。
	查看模拟结果Log文件	打开选中实验方案的模拟结果log文件，可以在**Simulation Settings**节点指定是否保留log文件。
	打开Builder	如果有必要，用Builder打开选中的实验方案。可以在Simulation Settings节点指定是否保留模型文件。
	打开 Results Graph	如果有必要，用Results Graph打开选中实验方案SR2文件，如有必要可以对 Has SR2 选项框打钩。可以在Simulation Settings节点指定是否保留模型SR2文件。
	打开Results 3D	如果有必要，Results 3D 打开选中实验方案SR2文件。如有必要可以对 Has SR2 选项框打钩。可以在Simulation Settings节点指定是否保留模型SR2文件。

按钮	名称	操作
	打开交互数据可视化工具	打开交互可视化工具，可以通过散点图、散射矩阵、直方图以及平行坐标更好的查看n维数据。通过该工具，可以更好的理解CMOST结果，也能更有效的表达这些结果。更多信息参考 CMOST Interactive Data Visualization Tool 。

6.5.2 创建实验方案

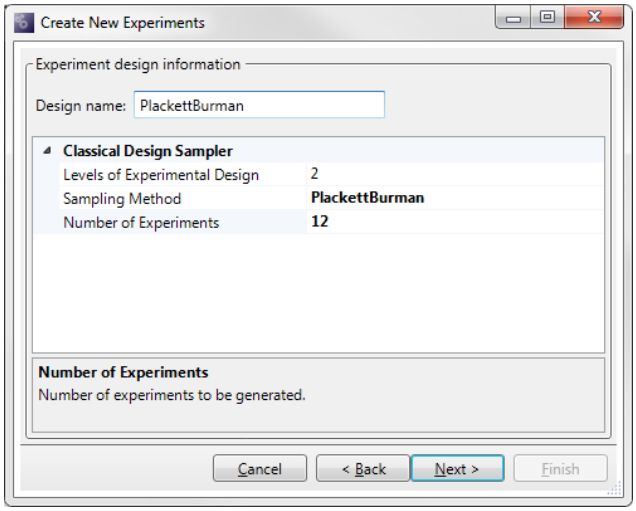
除CMOST自动生成的实验方案以外，还可以自己创建试验方案，如下：

1. 点击  **Create Experiments**图标。出现**Create New Experiments** 对话框：

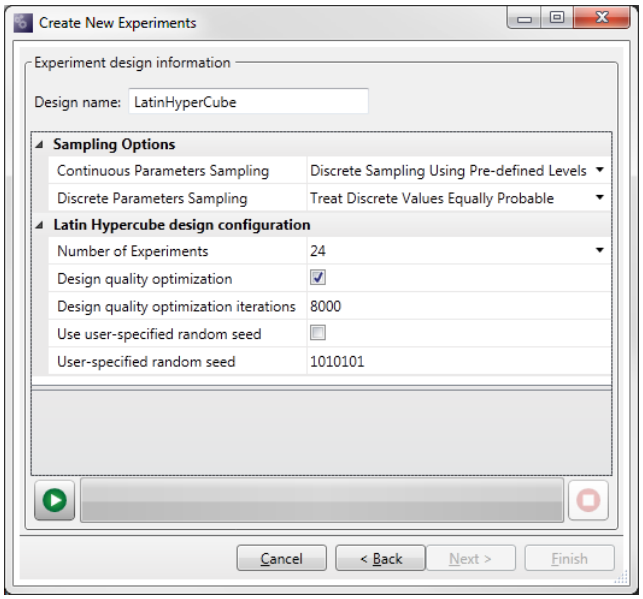


2. 选择想要创建新实验方案的方法，然后点击**Next**。

3. 如果选择Using classic design，对话框Choose classic experiment显示如下：



   a. 在**Levels of Experimental Design**选择*2*或*3*。
   b. 选择抽样方法。如果在**Levels of Experimental Design**选择2 。抽样方法可以设置为*Plackett-Burman*、*Fractional Factorial*或 *Full Factorial*中的一种。如果在**Levels of Experimental Design**选择*3*。抽样方法可以设置为*Box Behnken* 或*CCD Uniform Precision*。关于抽样方法的更多信息，请参考Classical Experimental Design。
   c. 依据以上配置来设置**Number of Experiments** 。
4. 如果选择Using Latin Hypercube design，会显示Create New Experiments 对话框（更多关于拉丁超立方实验设计的信息，请参考Latin Hypercube Design)。



a. 为**Continuous Parameters Sampling**选择抽样选项，其中：

Discrete Sampling Using Pre-defined Levels	如果选择该选项，数据范围被均匀的分成若干个事先定义的水平。实验方案仅能选取离散值。
Continuous Uniform Sampling within the Data Range	参数从取值范围内均匀抽样。
Continuous Sampling Using Prior Distribution	按照用户定义的先验概率分布函数进行抽样。

b. 为**Discrete Parameters Sampling** 选择抽样选项，其中：

Treat Discrete Values Equally Probable	所有候选值都有相同的概率。
Honour Prior Distribution for Discrete Values	在抽样过程中，使用用户定义的先验概率分布函数。

c. 选择**Number of Experiments**。这是由CMOST初始设置，主要依据参数及候选值的数量确定。

d. 如果选择 **Design quality optimization**，CMOST 将优化实验设计精度（最小的最大相关系数和最大的最小欧氏距离）。

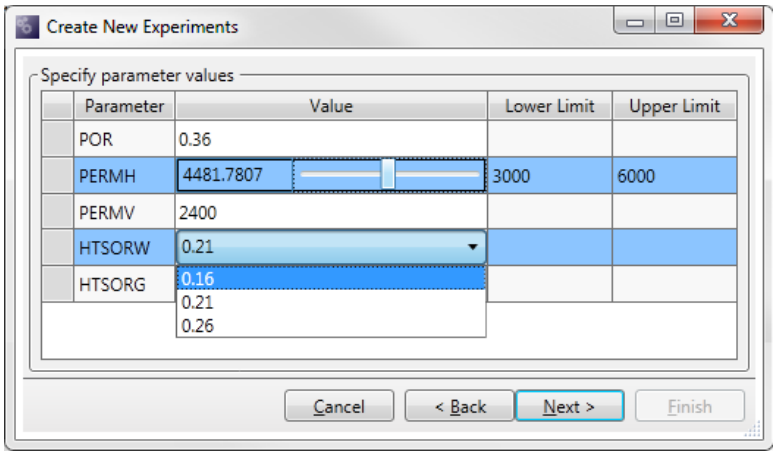
e. 设置**Design quality optimization iterations**，优化实验设计时允许的最大迭代数。仅仅当选择**Design quality optimization**时才可以使用。

f. 若需要，选择**Use user-specified random seed** 来生成实验方案，然后输入**User-specified random seed**。如果使用相同的随机种子，产生实验方案的随机顺序是可以重复的。如果没有选择**Use user-specified random seed**，会使用缺省的随机种子。

g. 点击  按钮。CMOST依据抽样选项和实验设计配置来创建试验方案。进度条将指示实验设计已经完成。

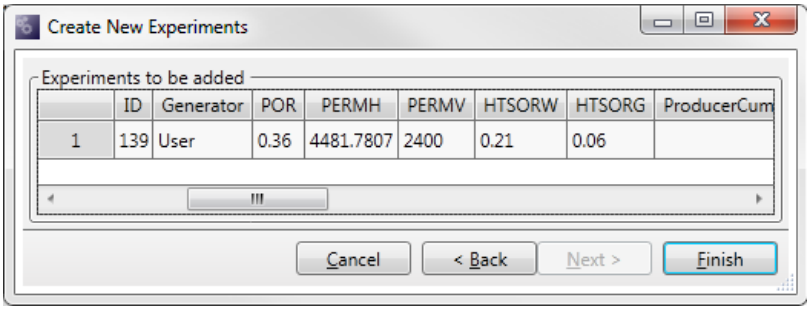
5. 如果参数是离散实数、整数或文本文件，则只能使用**Using all parameter combination design** 选项。以公式为基础的参数也是被允许的，因为这些参数是独立的。CMOST 支持65000个参数值组合，因此如果实验方案达到或超出该组合数后，则不能使用**Using all parameter combination design** 选项。

6. 如果选项 **Manual (user defined)**，可以在**Experiments Table**定义并添加自己的实验方案：



a. 如上所示，为离散变量选择候选值（例如HTSORW），使用滑轮设置连续变量（例如PERMH）。

b. 点击**Next**，将实验添加至 **Experiments to be added**。



c. 将所有的实验方案添加至Experiments Table，点击Finish 。

7. 如果所有的参数都是离散值，选择**Using all parameter combination design**选项，将生成参数全组合的一系列实验。

8. 无论使用什么样的添加实验方案方法，在点击**Finish** 后，都会将实验方案添加至**Experiments Table** 表格。例如：

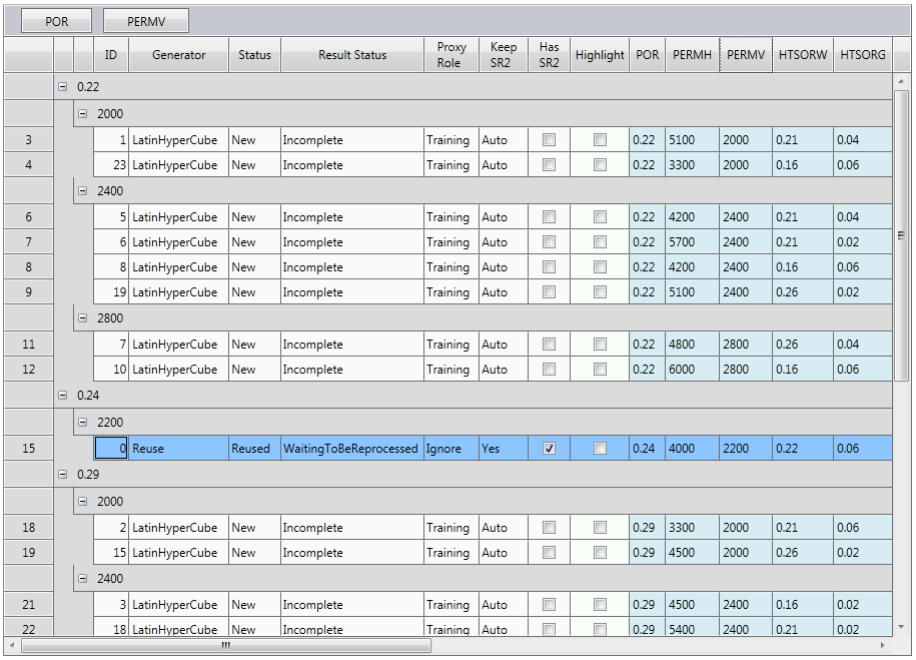
Drag and drop a column header here to group by that column

	ID	Generator	Status	Result Status	Proxy Role	Keep SR2	Has SR2	Highlight	POR	PERMH	PERMV	HTSORW	HTSORG
1	0	Reuse	Reused	WaitingToBeReprocessed	Ignore	Yes	☑	☐	0.24	4000	2200	0.22	0.06
2	1	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.22	5100	2000	0.21	0.04
3	2	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.29	3300	2000	0.21	0.06
4	3	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.29	4500	2400	0.16	0.02
5	4	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.29	3000	2800	0.21	0.04
6	5	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.22	4200	2400	0.21	0.04
7	6	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.22	5700	2400	0.21	0.02
8	7	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.22	4800	2800	0.26	0.04
9	8	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.22	4200	2400	0.16	0.06
10	9	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.36	5400	2400	0.21	0.04
11	10	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.22	6000	2800	0.16	0.06
12	11	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.36	6000	2800	0.26	0.06
13	12	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.29	3900	2800	0.16	0.02
14	13	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.36	3900	2000	0.26	0.04
15	14	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.29	3000	2800	0.26	0.06
16	15	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.29	4500	2000	0.26	0.02
17	16	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.36	3600	2800	0.16	0.02
18	17	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.36	5700	2000	0.16	0.04
19	18	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.29	5400	2400	0.21	0.02
20	19	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.22	5100	2400	0.26	0.02
21	20	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.36	4800	2000	0.16	0.04
22	21	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.36	3600	2400	0.26	0.02

6.5.3 配置实验表格

6.5.3.1 把试验方案分组

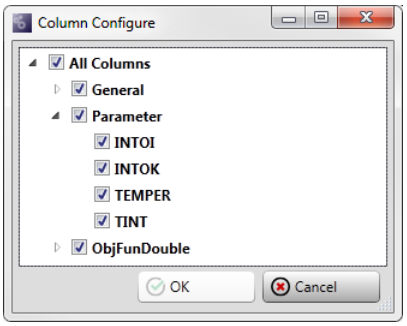
拖拽列表头到表格上面的空白区域，然后根据某列进行分组。在下面的例子中，实验方案先是按照参数POR分组，然后按照PERMV进行分组：



		ID	Generator	Status	Result Status	Proxy Role	Keep SR2	Has SR2	Highlight	POR	PERMH	PERMV	HTSORW	HTSORG
	0.22													
	2000													
3		1	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.22	5100	2000	0.21	0.04
4		23	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.22	3300	2000	0.16	0.06
	2400													
6		5	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.22	4200	2400	0.21	0.04
7		6	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.22	5700	2400	0.21	0.02
8		8	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.22	4200	2400	0.16	0.06
9		19	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.22	5100	2400	0.26	0.02
	2800													
11		7	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.22	4800	2800	0.26	0.04
12		10	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.22	6000	2800	0.16	0.06
	0.24													
	2200													
15		0	Reuse	Reused	WaitingToBeReprocessed	Ignore	Yes	☑	☐	0.24	4000	2200	0.22	0.06
	0.29													
	2000													
18		2	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.29	3300	2000	0.21	0.06
19		15	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.29	4500	2000	0.26	0.02
	2400													
21		3	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.29	4500	2400	0.16	0.02
22		18	LatinHyperCube	New	Incomplete	Training	Auto	☐	☐	0.29	5400	2400	0.21	0.02

6.5.3.2 限制列显示

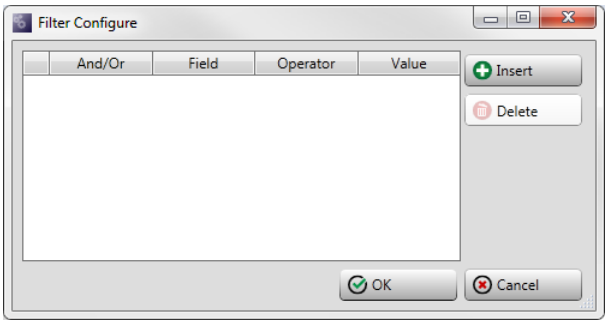
通过点击 **Configure Table**按钮，然后选择**Column Configure**，可以限制显示实验表格的内容。 **Column Configure** 对话框如下图所示。通过该对话框，可以选择显示常规、参数以及目标函数：



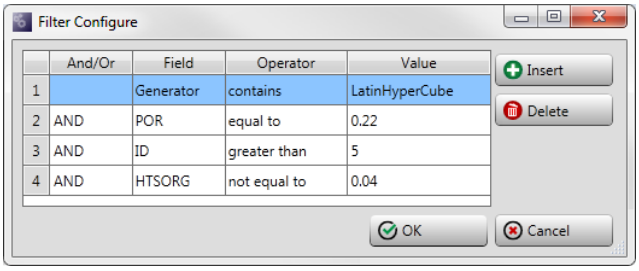
6.5.3.3 筛选实验方案

如果有大量的实验方案，可以定义一个过滤器来显示想要查看的某些实验，如下：

1. **点击Configure Table**  按钮，然后选择**Filter Configure**。对话框如下：

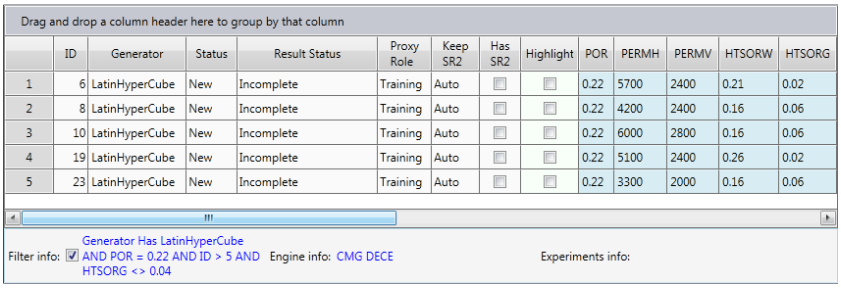


2. 根据需要配置该对话框。在下面的例子中，仅仅显示这些实验方案，其中Generator包含字符串*"LatinHyperCube"*， **POR**等于 0.22， **ID** 大于5，**HTSORG** 不等于 0.04：



3. 配置完成，点击**OK**。**Experiments Table** 底部**Filter info** 标签显示其约束条件，选中右面的复选框。

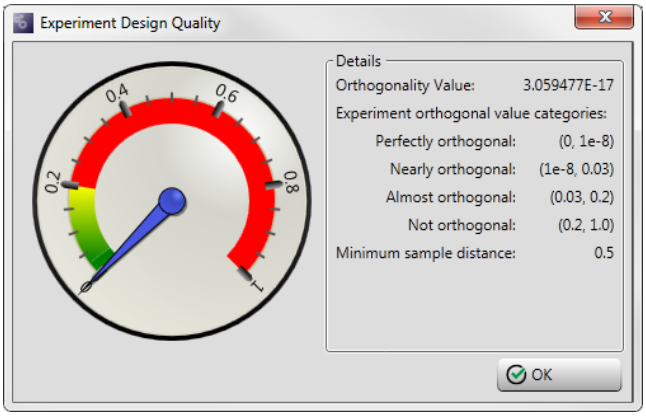
4. 如果选中**Filter info**复选框。**Experiments Table** 将仅仅显示满足上述条件的实验方案，我们的例子如下所示：



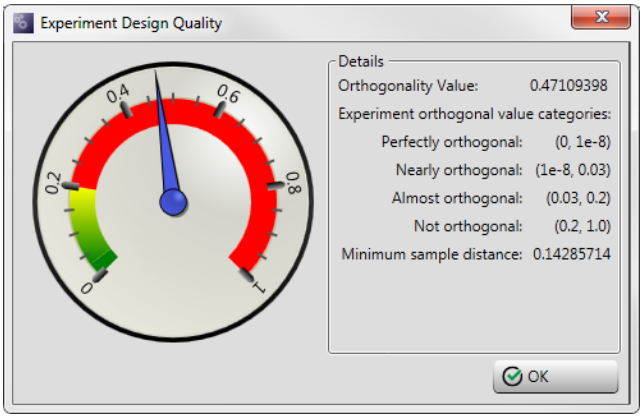
注意：为了显示所有的实验方案，清除 Filter info 复选框。

6.5.4 检验实验方案设计质量

点击 **Check Quality** 按钮打开**Experiments Quality**，该窗口提供了实验设计的精度。关于实验正交性和参数抽样点分布空间的讨论参考 Sampling Methods。在下面的例子中，实验设计正交性精度为 3.1E-17，在完全正交（*Perfectly orthogonal*）范围内。



在下面的例子中，正交精度为*0.47*，在非正交范围内。



注意：如果实验设计精度在红色区域分布，你可以尝试添加典型实验设计或拉丁超立方实验设计来改善实验设计精度。

6.5.5 输出实验表格到Excel

点击  Export to Excel 按钮，将表格内容输出到Excel文件。该命令仅仅对筛选行进行保存。

6.5.6 查看模拟结果日志文件

点击  View Simulation Log 按钮，在文本编辑器中打开模拟结果log文件。缺省的没有设置文本编辑去打开log文件，会出现一个对话框提示用户选择一个程序来查看log文件，该log文件可能对油藏工程师非常有用。通过Simulations Settings节点 中的 Job Record and File Management 部分，用户可以指定什么时候保留或删除log文件。

6.5.7 再处理实验方案

如果更改目标函数、单位制或历史数据，实验方案的目标函数需要重新计算后才可以再使用。点击  按钮来重新处理（例如，重新计算所有的目标函数）所有试验方案或选中的部分实验方案。

6.6 代理仪表盘

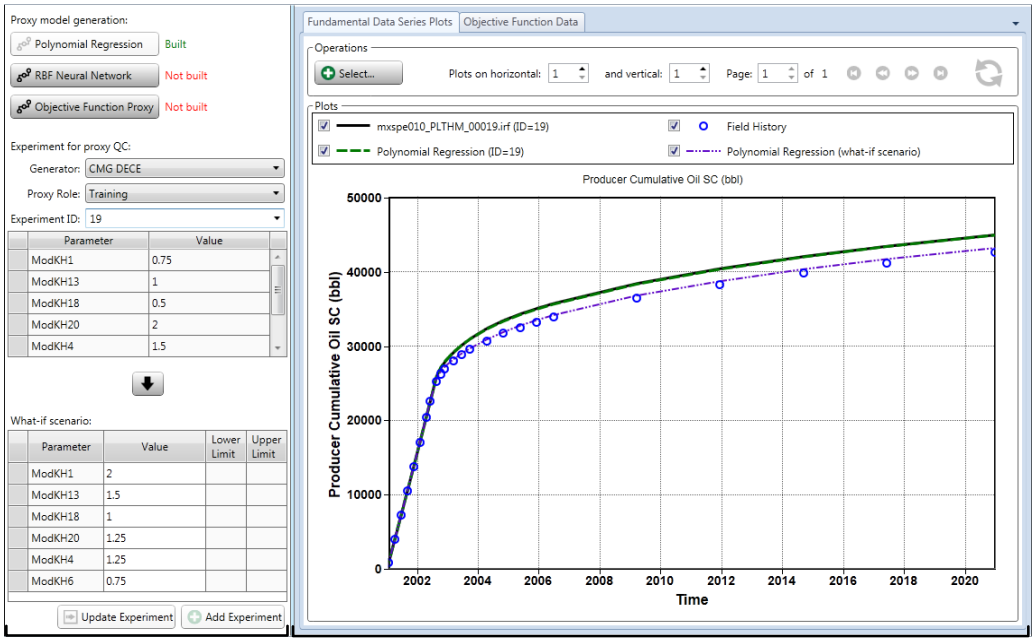
CMOST代理仪表盘提供了一个有效方法来快速评价参数值对结果的影响，即使当CMOST任务还在运行过程中，也可以查看。不用等到所有试验方案模拟结束，就可以查看结果。在运算大量模拟方案中的历史拟合和方案优化中特别有用。

在所有的实验方案运算完成之前，通代理仪表盘，可以:

- 使用初始的代理模型来进行预测。
- 研究输入参数变量的影响。
- 定义并添加training 或 verification实验方案到Study。
- 对比不同的代理模型。

6.6.1 打开代理仪表盘

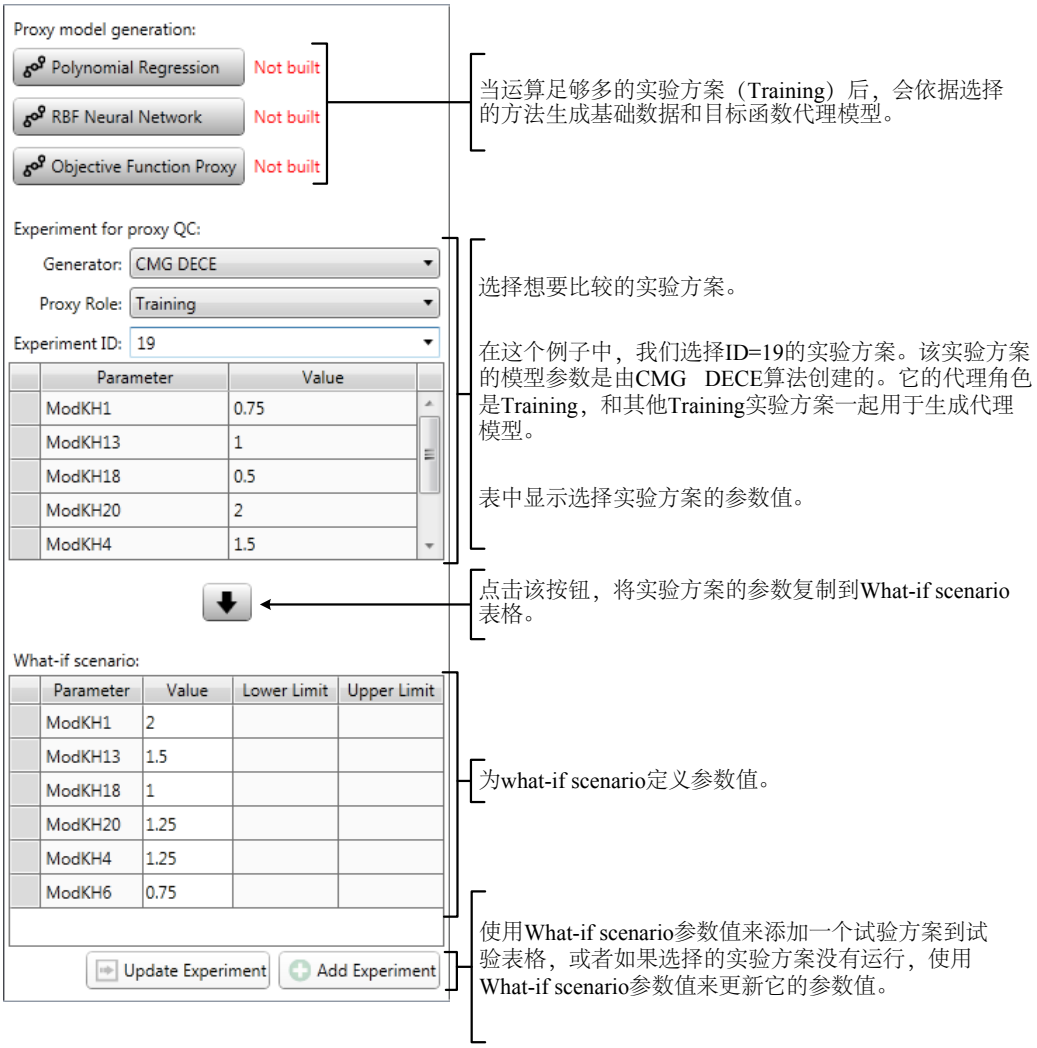
CMOST引擎启动后，就可以通过**Proxy Dashboard**，查看已完成运算的实验方案模拟结果，如下：



Configuration Area　　Display Area

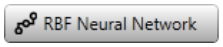
6.6.2 配置代理仪表盘

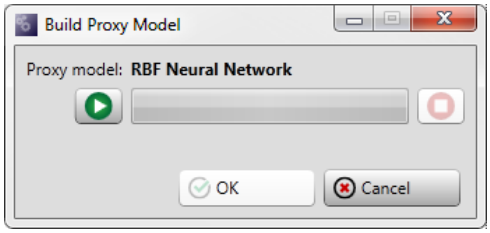
在对话框左侧，可以配置**Proxy Dashboard**，如下：



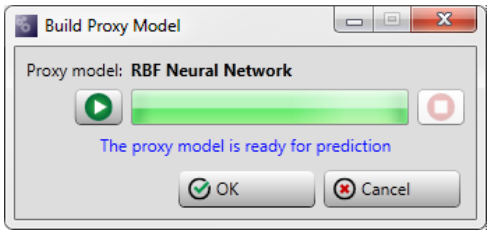
关于代理模型类型的更多信息，参考 Proxy Modeling。

6.6.3 创建代理模型为了
创建代理模型：

1. 点击想要建立的代理模型，例如：


Build Proxy Model对话框如下：


注意：如上所述 ，仅仅代理模型右面的按钮Not built 或 Out of Date 是可用的，才可以建立。如果已经建立了代理模型，只有当添加额外实验方案后才可以重新建立代理模型。

2. 为了建立模型，点击  会利用已经运算完成的Training 实验方案的模拟结果来创建选择的代理模型。创建代理模型所需的实验方案的数量主要依赖于参数个数及所选代理模型类型。当代理模型创建完成后，进度条变成绿色，同时OK键也被激活：


3. 点击**OK** 来保存代理模型。

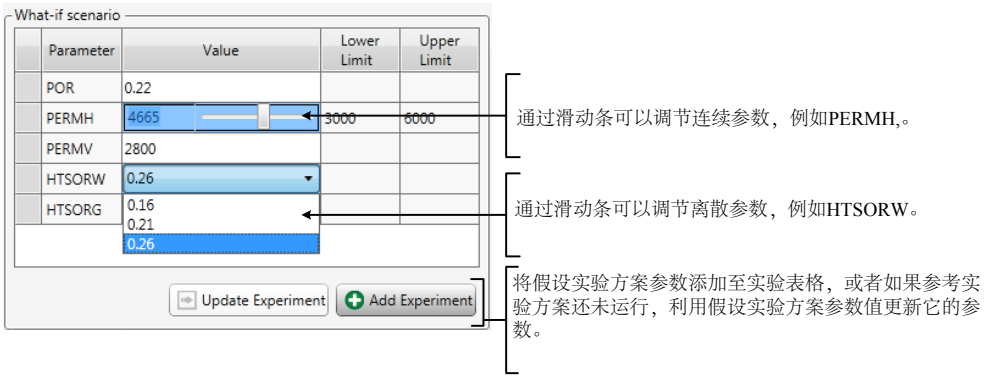
6.6.4 选择实验方案

选择某一实验方案用来比较生产历史数据、模拟器预测（SR2文件）以及代理模型预测（使用实验方案参数设置）：

1. 在上面的例子中，选择发生器*CMG DECE* 。
2. 选择**Proxy Role**，*Training* 或*Validation*。在**Experiment ID**列表中显示的实验方案必须满足一定的条件。
3. 选择**Experiment ID**，对应实验方案参数值在**Selected Experiment**表格中显示。同样，如果选择该实验方案，在代理模型可视化中也会显示该实验方案的曲线。

6.6.5 定义和应用What-if Scenarios

如果点击 按钮，选择的实验参数值将复制到What-if scenario表格，然后利用滑动条在What-if scenario表格调节这些参数值，如下图所示：



在**Plots**部分，通过代理模型预测可以查看假设实验方案的响应曲线。该技术证明了改变参数值对代理模型预测的影响。如果还没有开始运算，**What-if scenario** 值可以用来定义新实验方案和调整已经存在的实验方案。

- 如果点击**Update Experiment**，**What-if scenario** 表格中选中的实验方案将会更新**What-if scenario** 表格中的参数值，只要实验方案还未运行，它就是唯一的。
- 如果点击 **Add Experiment**，一个新实验方案将会添加**Experiments Table**（只要实验方案是唯一的），该实验方案的参数是在**What-if scenario** 部分定义的。

6.6.6 代理仪表盘交互影响

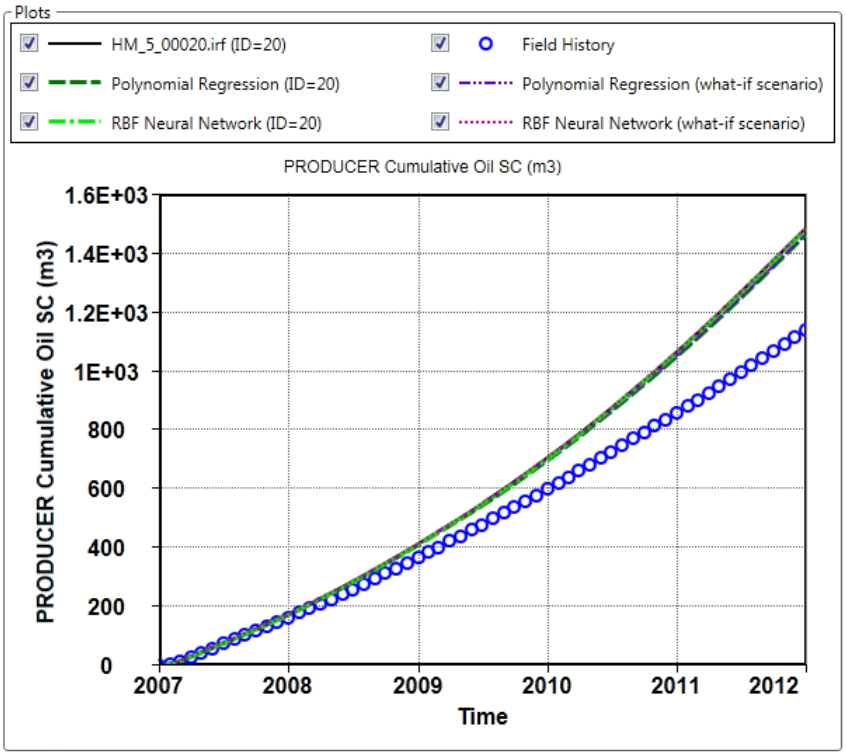
6.6.6.1 查看改变参数值对代理模型预测的影响

考虑下面的例子：

1. 在**Proxy Dashboard**左上角选择想要建立的代理模型。
2. 选择想要开始的实验方案。将该实验方案复制到 **What-if scenario**。
3. 在窗口右半部，选择**Fundamental Data Series Plots** 。
4. 点击 Select... 然后选择想要查看的时间序列。在Operations部分可以选择多条时间序列曲线。也可以在单一页面中配置曲线数量，还可以在多个页面之间切换。

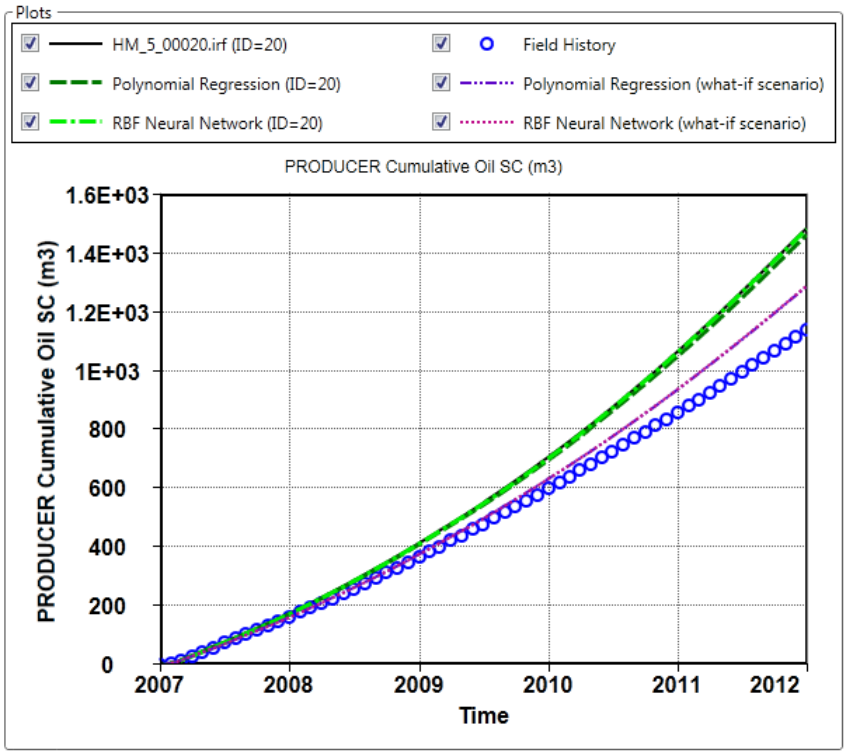
5. 在**Plots** 部分，选择想要显示的曲线。在下面的例子中，我们展示了历史拟合中实验方案（实验ID=20）生产井累产油的曲线。

   - 实验方案模拟结果
   - 选中实验方案代理模型预测结果
   - 假定实验方案代理模型预测结果
   - 目标函数生产历史数据，在这个例子中是生产井累产油。*Cumulative Oil SC*.



如上面的例子所示，模拟结果和多项式代理模型以及RBF代理模型预测结果较一致，但是两种代理模型预测结果和生产历史数据都相差较远。

6. 在**What-if scenario** 表格调整参数值来查看它是如何影响代理模型预测的。下面的例子中，我们将PERMH 从 *6000* 调整至*4500*:



如上图所示，调整后的曲线，更符合生产历史数据。

这是个简单的说明，如何使用代理仪表盘（**Proxy Dashboard** ）来查看改变参数值对结果的影响。

如上，建立了代理模型，选择任何已完成的实验方案来查看搭理模型的预测结果、模拟器运算的模拟结果与生产历史数据之间的误差。当更多的*Training* 实验方案运算完后，可以重新生成代理模型，也可以生成多个代理模型。

在上面的例子中，选择 **Objective Function Data**:

	Name	QC Simulation	QC Polynomial	QC RBF	What-if Polynomial	What-if RBF
1	ProducerCumOil	1483.066	1463.8676	1483.066	1285.9958	1287.9887
2	ProducerCumWater	4487.1601	4445.663	4487.1601	4030.0081	4016.2624
3	InjectorCumWater	4555.1595	4513.5813	4555.1595	4088.131	4074.5413
4	GlobalHmError	12.803585	10.64392	12.803585	5.227943	7.584102
5	HMError001	12.803585	10.64392	12.803585	5.227943	7.584102

该表格中的数值与图中完全一致，图中显示实验ID20 RBF神经网络代理模型预测结果与模拟器模拟结果完全对应（因为是使用这些值来训练代理模型），多项式代理模型预测稍微低一些。两个代理模型的拟合误差都比较高，调整参数后，拟合误差稍微好些。

6.6.6.2 代理仪表盘视图放大

和其他视图一样，可以定义和放大代理仪表盘局部视图，查看更多细节，然后右键选择 Un-zoom to 100% 返回原始状态。

6.6.6.3 保存代理仪表盘视图

右键代理模型视图，然后选择Save Image 来保存代理仪表盘视图。

6.6.6.4 复制代理仪表盘视图

右键代理模型视图，然后选择 Copy Image to Clipboard 拷贝视图到 Windows剪贴板，例如Excel 或 Word。如果在这一页上有多条曲线，那么该操作将复制和粘贴多条曲线。

6.6.6.5 复制目标函数数据表

在 Objective Function Data，点  按钮来复制目标函数表中的数据到Windows击 粘贴板，例如Excel 或 Word。

6.6.6.6 通过代理仪表盘添加实验

通过Add Experiment 按钮，使用What-if scenario 中修改的参数值来添加实验方案，如果选择的实验方案没有正在运行，点击Update Experiment 按钮，更新参数值。

通过 Experiments Table，可以做更多的调整，例如，可以修改参数值或实验方案代理模型类型。

通过代理仪表盘，可以运行这些添加的实验方案，然后来对比添加或修改的实验方案的模拟结果与代理模型预测的结果以及生产历史数据之前的误差。.

6.6.6.7 重新加载显示的代理仪表盘或表格

实验运算完成后，Proxy Dashboard 页面不会自动更新。除非点击 ，Refresh否者，结果文件曲线不会自动更新。

如果CMOST 引擎在后面添加了实验方案（例如，DECE引擎添加了实验方案），在点击Refresh后，添加的实验方案才可以使用。

6.6.6.8 更换代理角色

任何时间，都可以打开**Experiments Table**，更换实验方案代理角色，然后重新回到**Proxy Dashboard**。例如，如果觉得某个实验方案是异常值，不想用它作为训练代理模型，就可以更换该实验方案的代理角色。当修改某个实验方案代理模型角色时，将重新建立代理模型。

6.7 模拟任务

通过点击Control Centre | Simulation Jobs节点，可以查看模拟任务状态，实验方案运算完成后，细节被记录在Simulation Jobs 表格，如下面例子所示：

Drag and drop a column header here to group by that column

Experiment	LauncherID	Scheduler	Execution Node	Launcher Status	Submitted At	Started At	Finshed At	Dataset	Status	Result Status	Result Status Info
1	810	Local	CHARLESJ	Complete	2013-04-09T12:19:54	2013-04-09T12:19:57	2013-04-09T12:20:37	OPAAT_4.cmsd\OPAAT_4_00001.dat	Finished	NormalTermination	
2	811	Local	CHARLESJ	Complete	2013-04-09T12:20:49	2013-04-09T12:20:52	2013-04-09T12:21:32	OPAAT_4.cmsd\OPAAT_4_00002.dat	Finished	NormalTermination	
3	812	Local	CHARLESJ	Complete	2013-04-09T12:21:39	2013-04-09T12:21:42	2013-04-09T12:22:23	OPAAT_4.cmsd\OPAAT_4_00003.dat	Finished	NormalTermination	
4	813	Local	CHARLESJ	Complete	2013-04-09T12:22:30	2013-04-09T12:22:33	2013-04-09T12:23:13	OPAAT_4.cmsd\OPAAT_4_00004.dat	Finished	NormalTermination	
5	814	Local	CHARLESJ	Complete	2013-04-09T12:23:20	2013-04-09T12:23:23	2013-04-09T12:24:03	OPAAT_4.cmsd\OPAAT_4_00005.dat	Finished	NormalTermination	
6	815	Local	CHARLESJ	WaitingToStart	2013-04-09T12:24:10			OPAAT_4.cmsd\OPAAT_4_00006.dat	Pending	Unknown	

注意：在试验表格，通过拖拉实验表头对模拟结果进行分组。然而，不可以向左或向右拖拉表头来对表格进行重新整理。

表格中的列如下所示：

Experiment：唯一的实验ID号码。

- **Launcher ID**：Study中每个实验方案都有唯一的实验ID。当将实验提交给Scheduler时，就给给它们分配一个唯一的Launcher ID。Launcher ID 和实验ID不同，因为Scheduler可能处理多个Study中的实验方案。
- **Scheduler**：通过 Launcher提交作业至Scheduler 。
- **Execution Node**：计算机执行任务节点。
- **Launcher Status**： *Waiting to Start*、*Running*或 *Complete*中的某一种状态。
- **Submitted At**：Launcher提交任务到模拟器的时间。
- Started At：模拟器开始运行的时间。
- **Finished At**：模拟器运算完成的时间。
- **Dataset**：用于模拟的数据文件。如果对于正常终止的实验方案选择Delete .dat，数据文件会被删除。如果对于正常终止的实验方案没有选择Delete .dat，数据文件就会被保存在Study文件。

- **Status**：提交任务的状态， Pending 或 Finished。
- **Results Status:**
  - **Abnormal Termination**：在到达指定模拟停止时间之前，任务终止。
  - **Incomplete**：由于软件或硬件问题导致模型没有运算完成。
  - **Normal Termination**：模拟运算正常结束。
  - **Killed**：任务被用户Kill掉。
  - **Failed**：由于硬件、软件或许可问题导致任务运算失败。
  - **Unknown**:：当任务运行时，其结果状态是未知。
- **Results Status Info**：对某一特殊的**Results Status**的阐明信息。对于某一实验的结果状态是异常终止，对于方案优化来说，结果状态信息可能是不收敛。

表格右面按钮，如下所示：

按钮	名称	操作
	刷新模拟任务	点击，使CMOST得到更新后的模拟任务状态。
	查看模拟结果log文件	如果可以的话，在文本编辑器中打开选中实验方案的模拟结果log文件。在模型运算完成后，log文件是否可用，取决于模拟设置（Simulation Settings）。
	打开Builder	在Builder中打开选中的实验。在模型运算完成后，.dat文件是否可用，取决于模拟设置（Simulation Settings）。
	打开Results Graph	打开选中的SR2文件。在模型运算完成后，SR2文件是否可用，取决于模拟设置（Simulation Settings）。
	打开Results 3D	在Results 3D中打开选中的SR2文件。在模型运算完成后，SR2文件是否可用，取决于模拟设置（Simulation Settings）。

7 查看并分析结果
(Viewing and Analyzing Results)

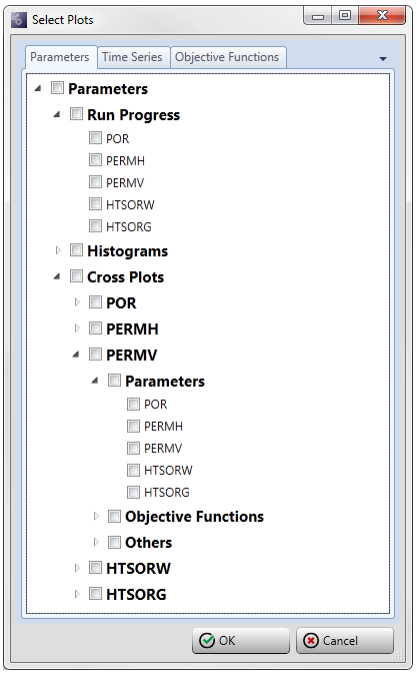
7.1 基本信息

通过**Results & Analyses**界面，查看CMOST运算结果。只要CMOST开始运行，就会显示模拟结果。

7.1.1 多图显示

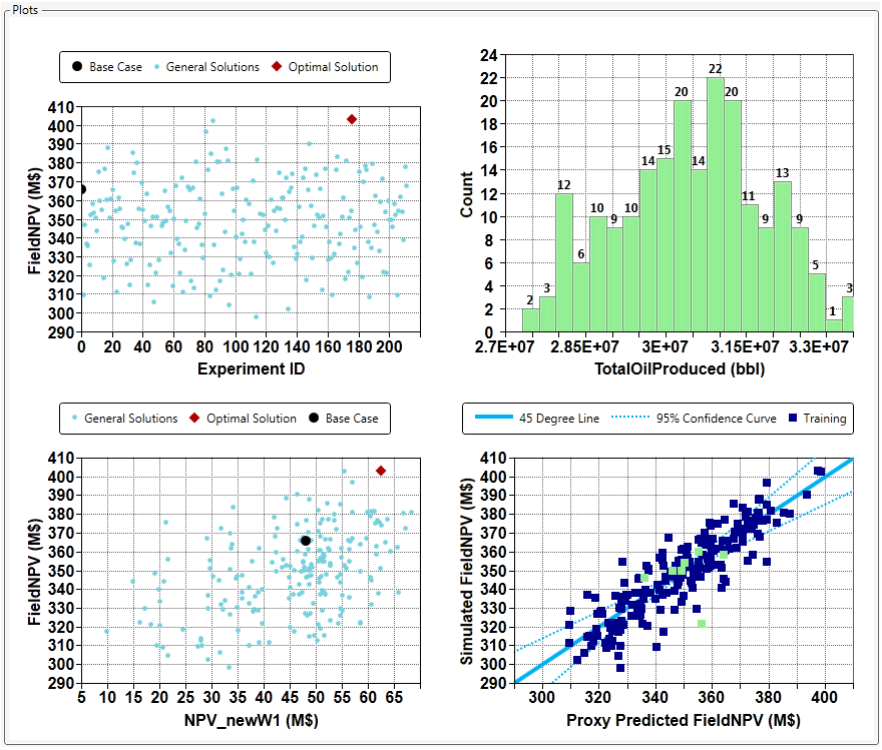
在每个Results & Analyses主页面上都有子节点，从这些子节点上可显示多张平面图，如下例所示：

1. 在Operations部分点击Select，显示 Select Plots对话框，该对话框包含可以使用的曲线类型和曲线。下面介绍一个例子：



2. 选择想要显示的曲线。可以从不同节点选择曲线。
3. 在Operations 部分，设置显示曲线的条数以及曲线位置（水平或者垂直）。在下面Results &Analyses | Objective Functions节点，显示了运行

进展图、直方图、交汇图以及模型QC图：



如果平面图数量超过了每页允许放置的最大数量，将会多页显示。另外，还可改变允许水平和垂直放置平面的数量，如上所示。

4. 如果在CMOST运行时显示了多张平面图，随着模拟的进行，这些平面图会定期更新，当然，也可以通过点击Refres 按钮进行立即更新。

7.1.2 界面操作

参考 Plots 中关于基础操作的信息，例如保存平面图文件，复制到Windows粘贴板以及放大或缩小。

7.1.3 切换到树状视图

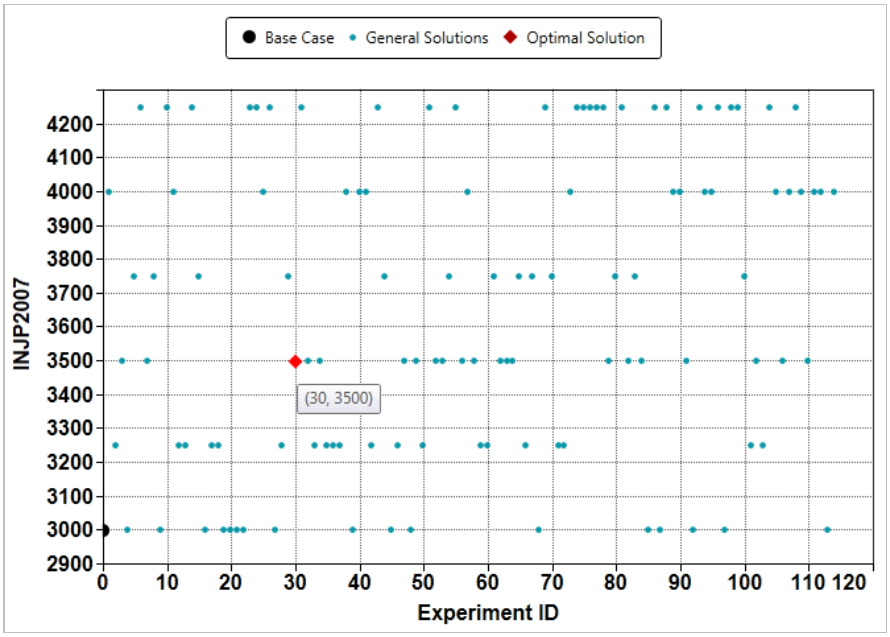
在树状图下选择某个节点，可以选择使用向上或向下箭头来切换节点，也可以使用向左或向右箭头来打开和关闭树状节点。如果没有可打开和关闭的节点，可以使用向左或向右箭头来向上或向下移动不同等级节点。

7.2 查看并分析参数结果

通过**Parameters** 节点，可以显示Study中基础参数的结果信息。

7.2.1 运行进展图（参数）

通过**Parameters | Run Progress**节点，查看Study中参数模拟进展的情况：

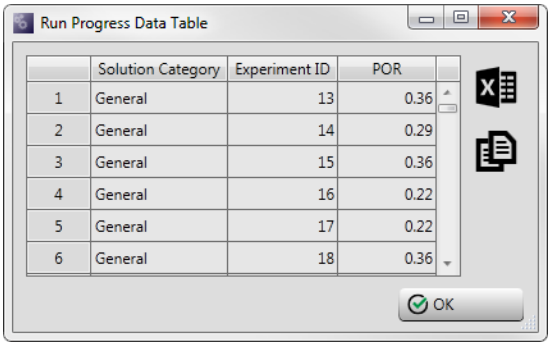


上图显示了参数INJP2007 和不同实验方案运行进展图。每个蓝色点表示一个试验方案。随着运算的进行，越来越多的点添加至图中。在历史拟合和方案优化时，当前最优方案是用红点显示。随着运算的进行，当出现一个更好的试验方案时，当前的最优方案会发生变化。

如上所示，将鼠标移至某个数据点，试验ID和对应参数（*INJP2007*）值就会显示出来。

为每个Study参数生成一个 Run Progress图，可以在树状节点查看该图。

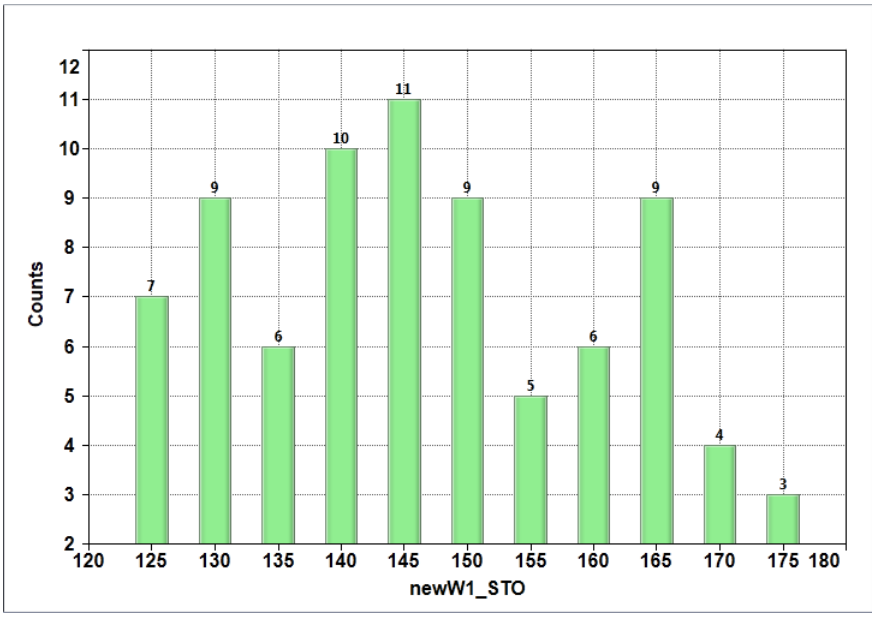
如果右键图，选择Data，可以打开 Run Progress Data Table 对话框：



使用右面的按钮，可以输出Run Progress Data Table内容到Excel或复制它到Windows粘贴板。

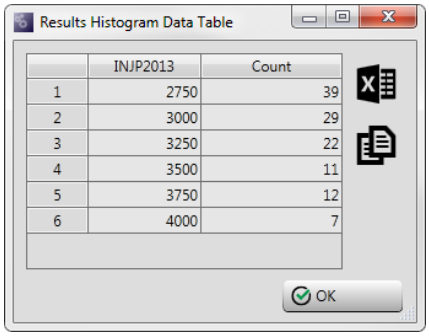
7.2.2 直方图（参数）

通过Parameters | Histograms节点，可以查看每个参数值在运算的实验方案中使用的频数，例如：



和其他平面图类似，可以将其 保存为多种图片格式或复制它到Windows剪贴板，然后将其粘贴到其他应用程序中，还可以放大或缩小。

如果右键平面图，然后选择**Data** ，可以查看并将其保存成表格形式的数据。

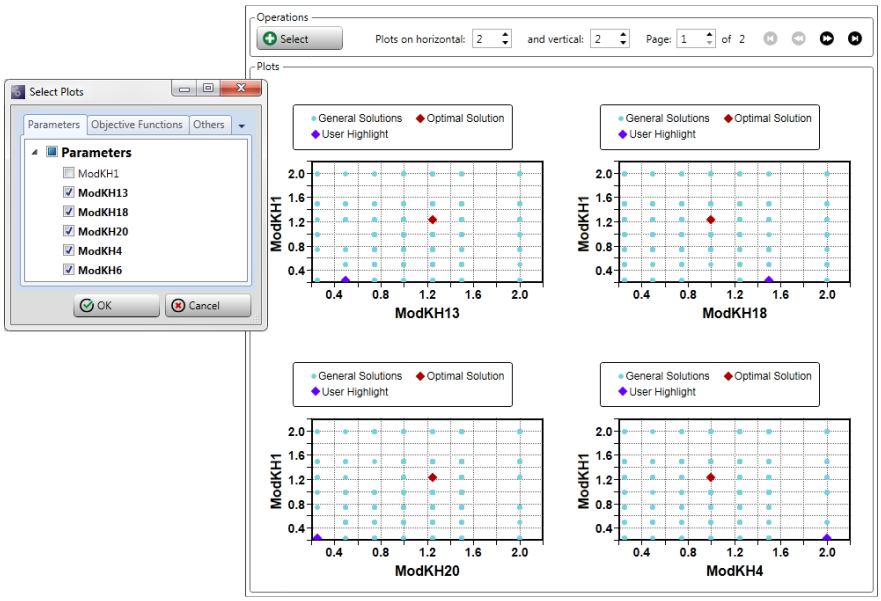


使用图形右面的按钮，可以输出**Results Histogram Data Table**内容到Excel，或复制它到Windows剪贴板，然后粘贴到其他应用程序。

7.2.3 交汇图（参数）

通过Parameters | Cross Plots节点，可以定义参数、目标函数及其他数值之间趋势和关系，如下：

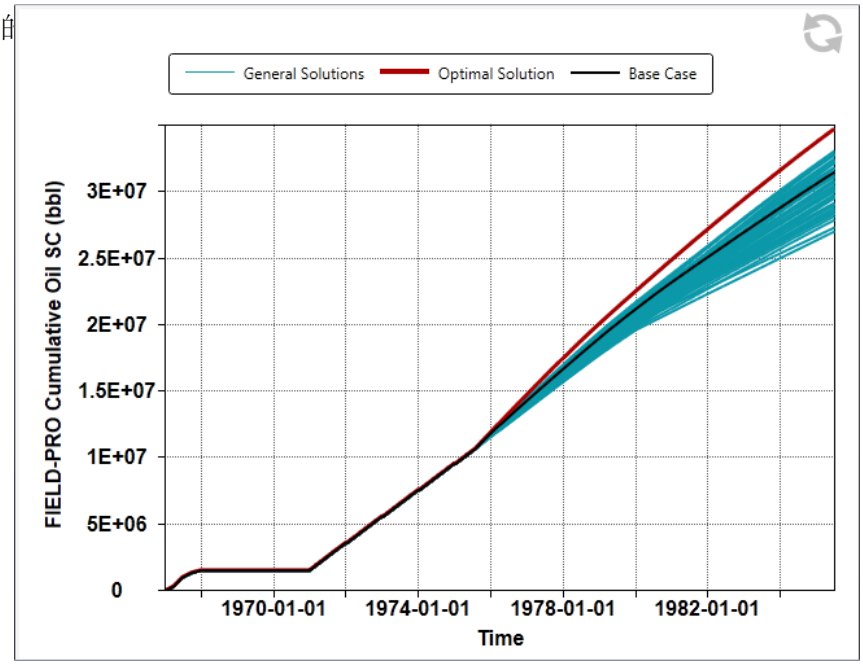
1. 在树状图，选择想要在交汇图中显示的参数。
2. 在Operations部分，点击 Select 打开Select Plots 对话框。
3. 选择想要在交汇图中显示的参数、目标函数或其他值。
4. 选择想要水平和垂直显示的平面图数。交汇图如下所示：



7.3 查看并分析时间序列结果

7.3.1 观察目标函数（时间序列）

下面显示的是结果观察曲线时间序列

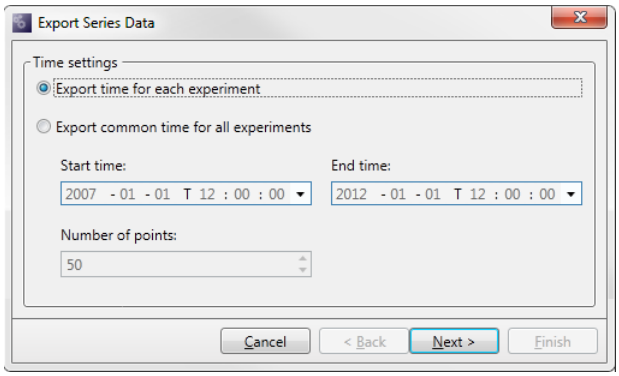


在上面的时间序列曲线中，红色曲线是最优的实验方案，其他实验方案是用蓝色曲线显示。曲线越集中说明模拟结果越相似。

如果在结果观察曲线中定义了生产历史数据，那么数据点以深蓝色圆圈显示。对于历史拟合来说，总目标函数是最小值，对于方案优化来说，总目标函数是最大值。

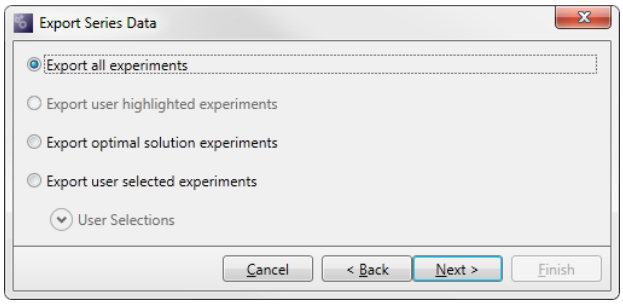
输出时间序列数据到文本（.txt）文件：

1. 在时间序列图，右键点击任意位置，然后选择**Data**，显示对话框**Export Series Data – Time Settings**：



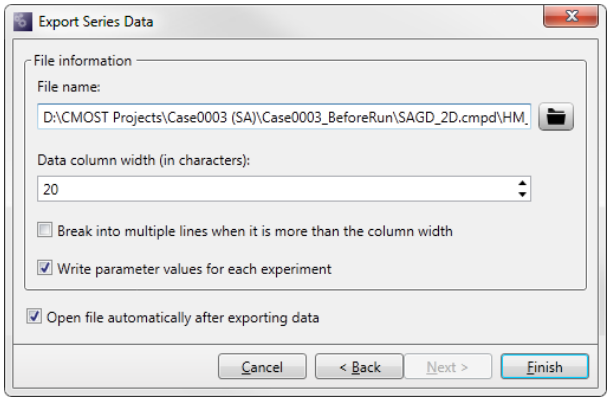
2. 设置想要输出数据的时间，下面中的一种：
   - **Export time for each experiment**：输出与每个实验方案相关的时间点数据。
   - **Export common time for all experiments**：数据输出之前，先对时间进行校正。在这个例子中，定义了开始和结束时间，输出的时间点在它们之间。

3. 点击**Next**。对话框**Export Series Data**如下图所示：



4. 选择想要输出的时间序列，下面中的一种：
   - Export all experiments
   - Export user-highlighted experiments
   - Export optimal solution experiment
   - Export user selected experiments如果选择该选项，可以点击 **User Selections**来打开**Select experiments to export** 表格，如有必要，利用CTRL和SHIFT键选择多个实验方案，然后点击Check。当然，也可以选择**Uncheck** 来取消。最后，直接点击**Export** 选项。

5. 点击**Next**。对话框**Export Series Data – File Information** 如下所示：



6. 为输出的文本文件输入说明和配置信息。
7. 点击**Finish** 输出时间序列数据。

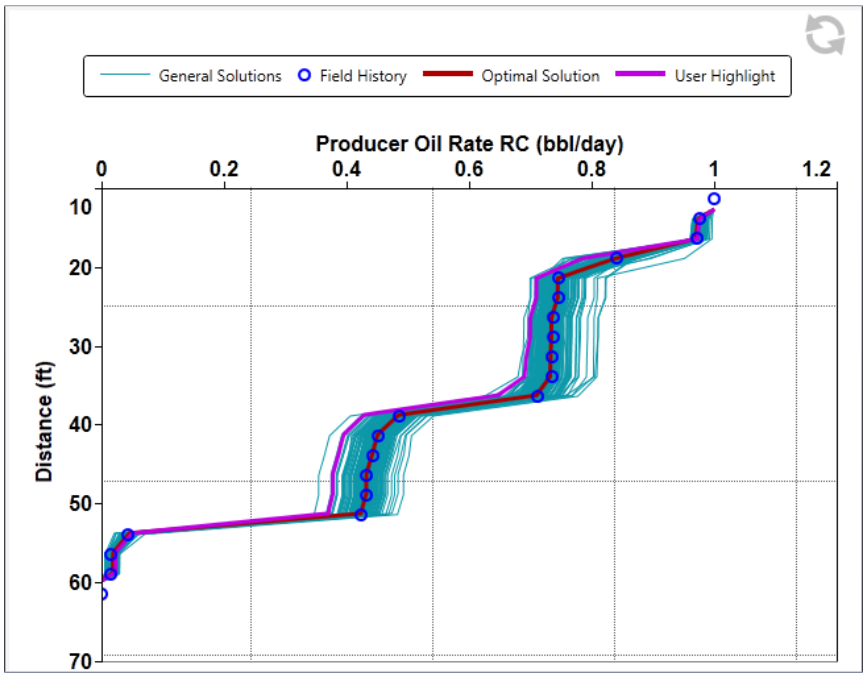
为了打开在Excel电子表格中输出的时间序列：

1. 创建输出的时间序列文本文件，如上所述。
2. 在Excel中打开文本文件。

7.4 查看并分析属性vs.距离结果曲线

7.4.1 观察目标函数（属性vs.距离）

下面是关于属性vs. 距离结果观察曲线的例子：



正如时间序列结果观察曲线一样，最优的实验方案用红色标记，用户高亮显示的曲线用紫色标记，其他普通试验方案用蓝色标记。

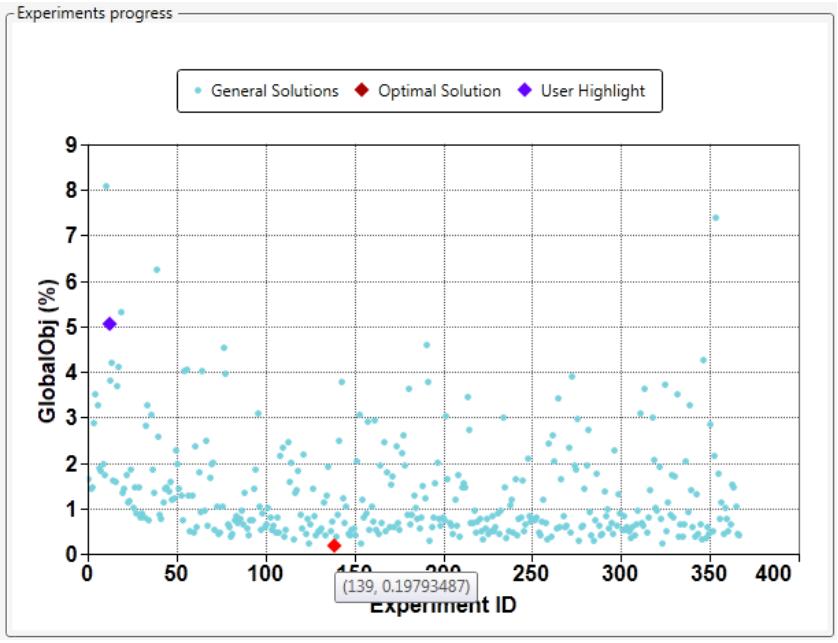
如果生产历史文件也在结果观察中定义，那么数据点是用深蓝色圆圈显示。对于历史拟合或优化最小值的任务，最优方案的总目标函数拥有最小值。对于优化最大值的任务，最优方案的总目标函数拥有最大值。

可以输出属性 vs.距离观察数据到文本文件，和输出时间序列观察数据方法类似，详见Exporting TimeSeries Data ，在属性vs.距离数据中，时间设置不是必需的。

7.5 查看并分析目标函数结果

7.5.1 运行进展（目标函数）

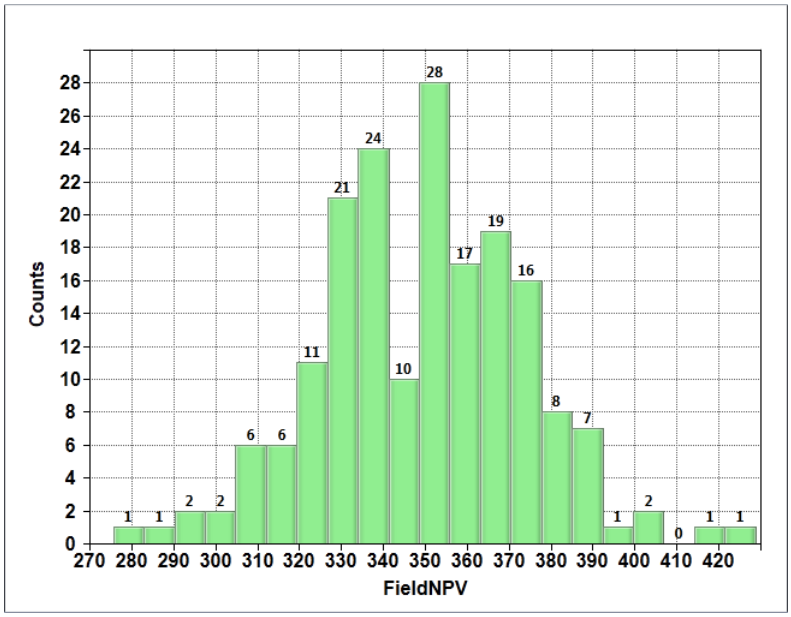
在目标函数运行进展图中，最优实验方案是以红色标记。在下面的例子中，我们将鼠标移到最优实验方案上，其显示了实验方案ID和*GlobalObj*结果：



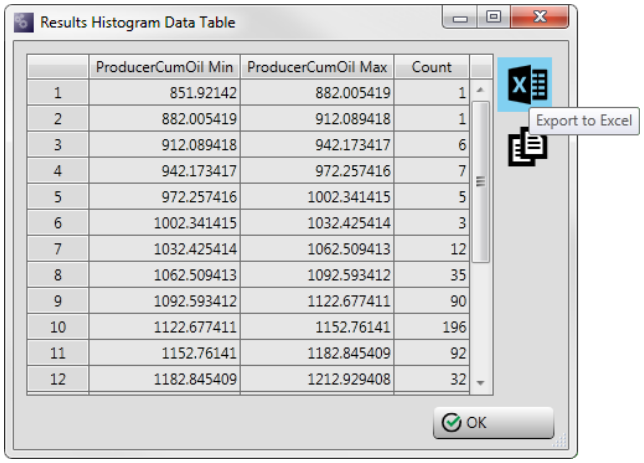
正如其他平面图，可以将其保存为多种图片格式，或者复制它到Windows剪贴板，将其粘贴到其他应用程序中。如果右键点击平面图，然后选择 **Data**，可以查看数据并将其保存为表格格式。

7.5.2 直方图（目标函数）

目标函数直方图显示从实验方案中计算目标函数的分布：



正如其他平面图，可以将其保存为多种图片格式，或者复制它到Windows剪贴板，将其粘贴到其他应用程序中。如果右键点击平面图，然后选择**Data**，可以查看数据并将其保存为表格格式：



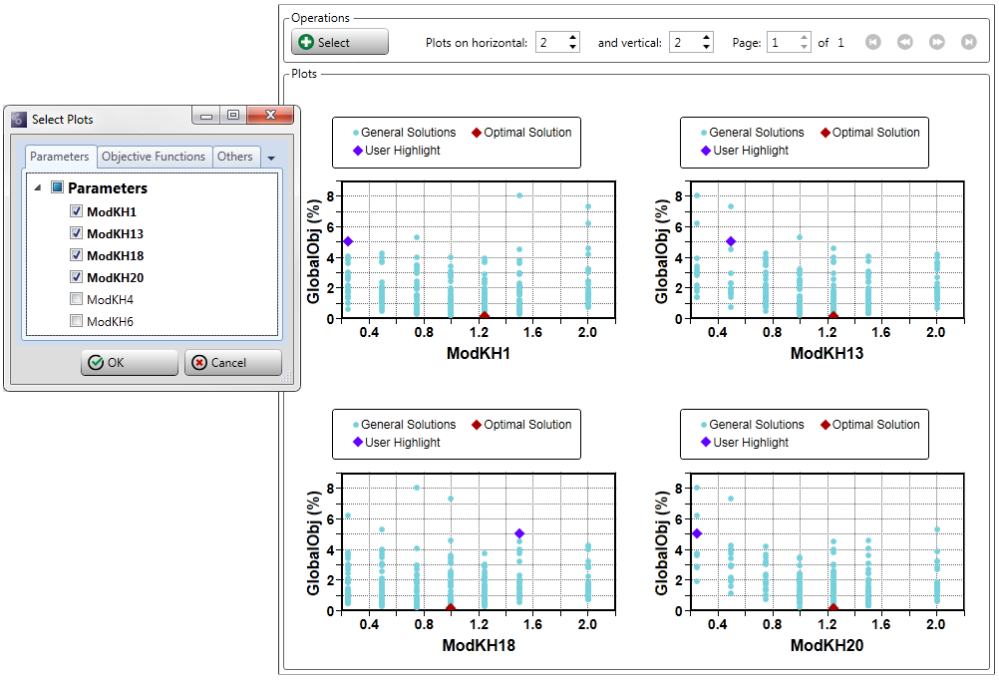
	ProducerCumOil Min	ProducerCumOil Max	Count
1	851.92142	882.005419	1
2	882.005419	912.089418	1
3	912.089418	942.173417	6
4	942.173417	972.257416	7
5	972.257416	1002.341415	5
6	1002.341415	1032.425414	3
7	1032.425414	1062.509413	12
8	1062.509413	1092.593412	35
9	1092.593412	1122.677411	90
10	1122.677411	1152.76141	196
11	1152.76141	1182.845409	92
12	1182.845409	1212.929408	32

使用上图右面的按钮，可以将**Results Histogram Data Table**内容输出到Excel，或者复制到Windows剪贴板，然后将其粘贴到其他应用程序。

7.5.3 交汇图（目标函数）

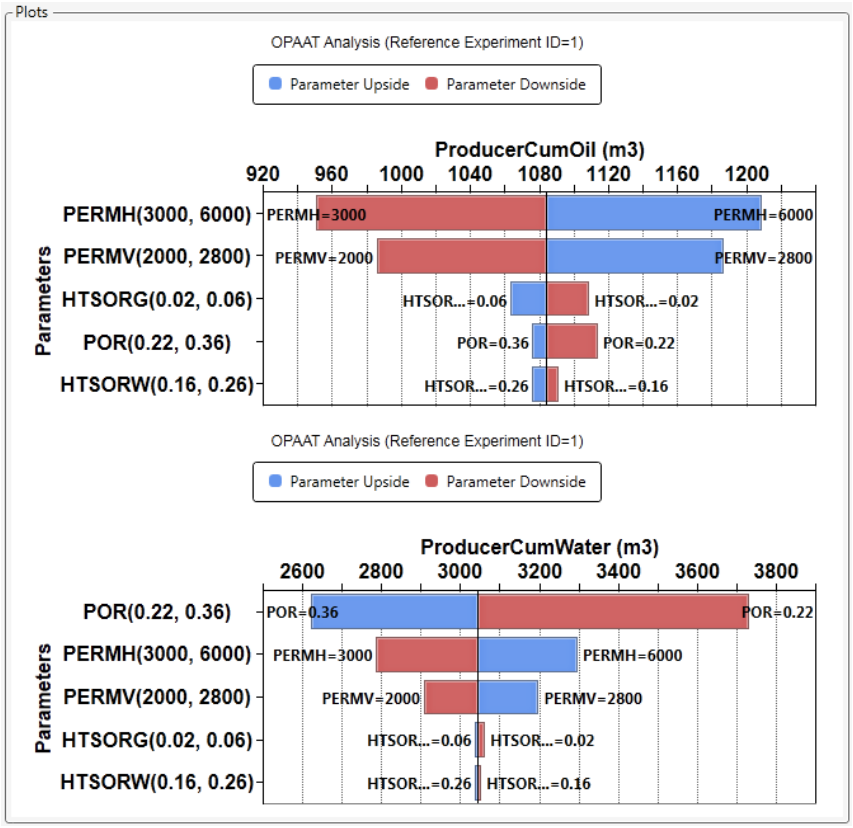
通过**Objective Function | Cross Plots**节点，可以确定目标函数、参数及其他数值之间的趋势和关系，如下：

1. 在树状图中，选择想要在交汇图坐标中显示的目标函数。
2. 在**Operations** 部分，点击 Select 来打开**Select Plots**对话框。
3. 选择需要的标签，然后选择想要在交汇图坐标中显示的参数、目标函数或其他值。
4. 设置想要水平和垂直显示图的数量。交汇图如下面例子所示：



7.5.4 OPAAT分析

如果使用OPAAT引擎来执行敏感性分析，会使用不同的实验参考为每个目标函数生成OPAAT图，如下面例子所示，我们使用了Select 按钮来选择多个 OPAAT图显示。



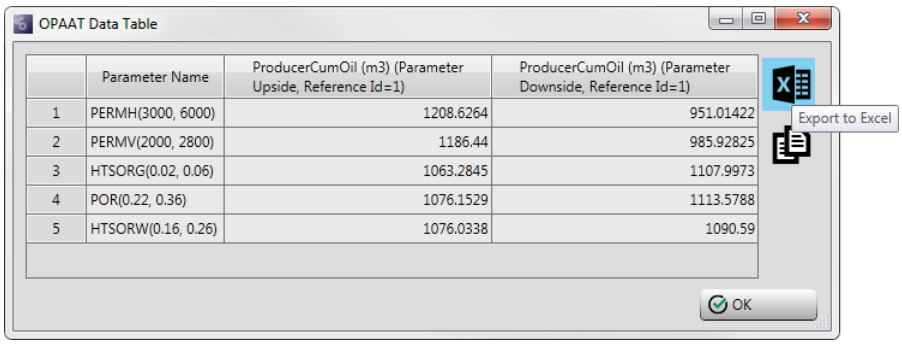
注意：如果在OPAAT图中显示多个条带，那么Y轴前沿数值将动态变化。

在上面的例子中，OPAAT图使用实验ID为1作为参考。垂向深色线的值表示实验产生的目标函数值。

每个参数对应的目标函数值在条带两侧显示。如果目标函数和参数之间正相关，参数上限 (蓝色条带)在右侧，参数下限（红色条带）在左侧，参考实验目标函数值使用垂向深线表示。另外一方面，如果目标函数和参数之间是负相关，对应的条带是相反的，蓝色条带在左侧，红色条带在右侧。条带的总长度表示改变参数值对目标函数的影响程度。

在试验ID为1的实验中， PERMH的数值为4500。如果将PERMH降低至3000，其他参数值不变，累产油减少至951。如果将PERMH增加至6000，累产油将增1209。PERMH和累产油之间正相关，红色条带在左边，蓝色条带在右边。 作为对比，HTSORG 和累产油之间呈负相关。例如，对于实验ID为1的实验，将HTSORG增加至0.06，其他参数保持不变，目标函数减少至1063。

查看OPAAT数据或输出数据时，首先在任意空白处点击右键。然后选择**Data**。**OPAAT Data Table**对话框如下所示：

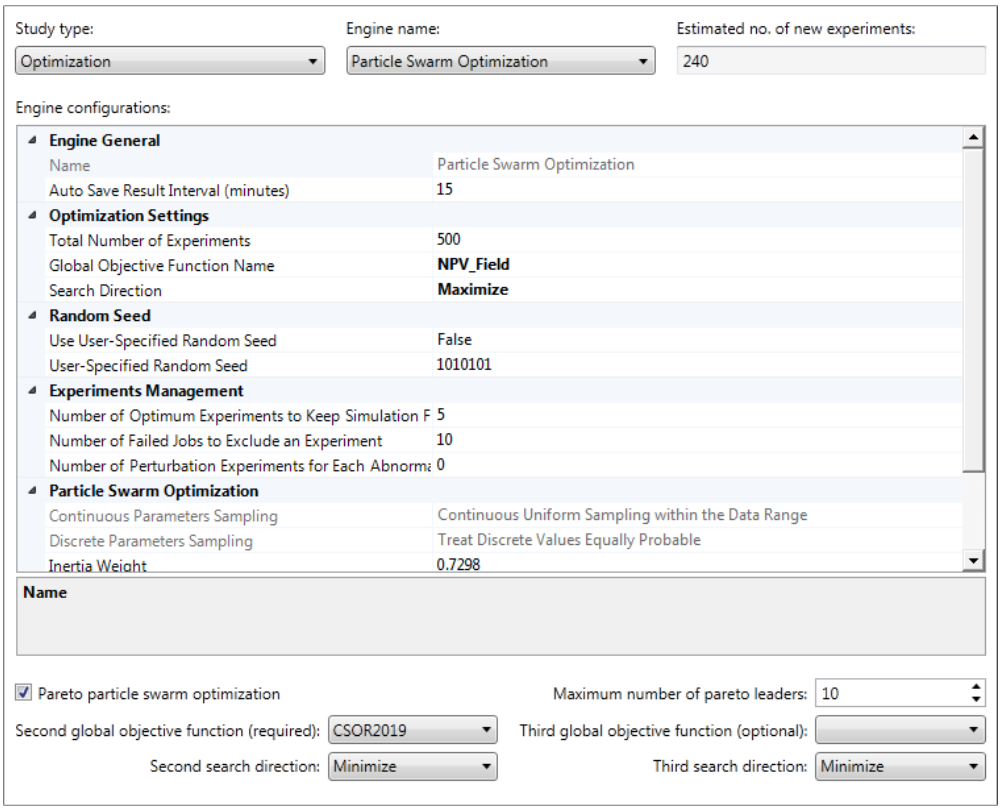


点击 输出OPAAT数据到Excel， 输出数据到Windows剪贴板或OK 来关闭对话框，不输出数据。

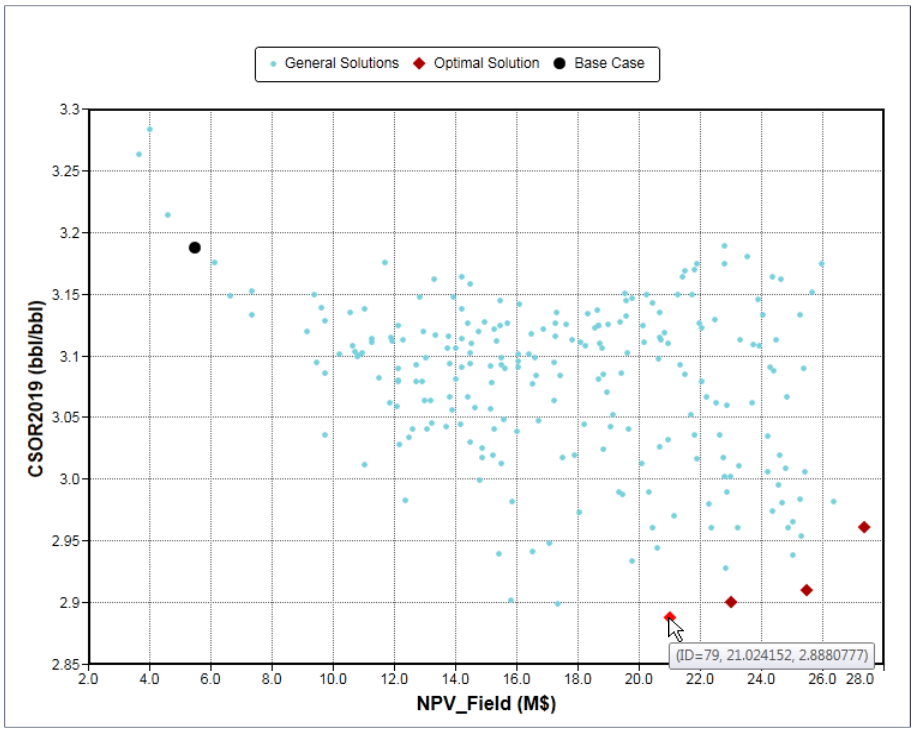
注意：当OPPAT图中的水平条带端点位于参考实验线，表示参考实验中参数的取值是该参数的极值（最大或最小值）。

7.5.5 帕累托前沿

为了说明帕累托PSO显示的结果，考察下面的方案优化，目标函数*NPV_Field*是使用**Particle Swarm Optimization** 引擎来进行优化最大值。我们已经选择**Pareto particle swarm optimization**，因此我们也可以优化目标函数*CSOR2019*最小值：


Study type:
Optimization
Engine name:
Particle Swarm Optimization
Estimated no. of new experiments:
240
Engine configurations:
Engine General
Name
Particle Swarm Optimization
Auto Save Result Interval (minutes)
15
Optimization Settings
Total Number of Experiments
500
Global Objective Function Name
NPV_Field
Search Direction
Maximize
Random Seed
Use User-Specified Random Seed
False
User-Specified Random Seed
1010101
Experiments Management
Number of Optimum Experiments to Keep Simulation F
5
Number of Failed Jobs to Exclude an Experiment
10
Number of Perturbation Experiments for Each Abnorma
0
Particle Swarm Optimization
Continuous Parameters Sampling
Continuous Uniform Sampling within the Data Range
Discrete Parameters Sampling
Treat Discrete Values Equally Probable
Inertia Weight
0.7298
Name
Pareto particle swarm optimization
Maximum number of pareto leaders:
10
Second global objective function (required):
CSOR2019
Third global objective function (optional):
Second search direction:
Minimize
Third search direction:
Minimize

上述方案优化研究帕累托前沿PSO图如下所示：



帕累托前沿包含四个最优方案，这些方案可以通过移动鼠标查看，如图所示。关于帕累托前沿PSO的理论信息，参考Pareto Front Particle Swarm Optimization，相关引擎设置信息，参考Pareto Front Engine Settings。

7.5.6 Proxy Analysis

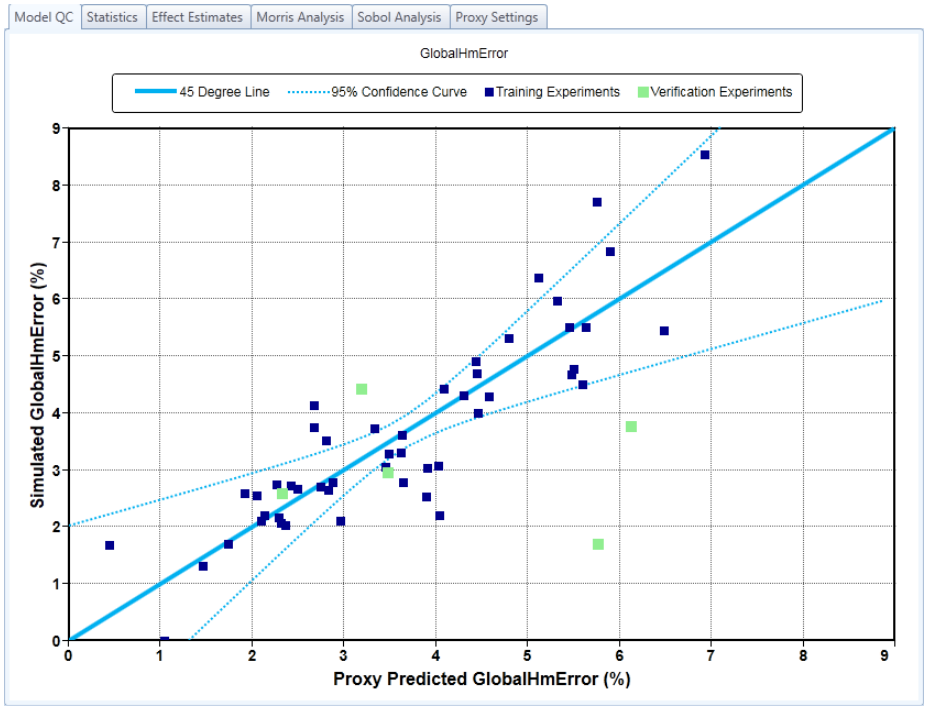
Information is displayed in this node if there are enough experiments to build the proxy model.

Once enough runs are available, proxy models (several variants of polynomial regression, and RBF neural network) will be created. For sensitivity analysis studies, the proxy models are then used to determine the main (linear) effects, interaction effects, and quadratic (nonlinear) effects. For uncertainty assessment studies, proxy models are used to perform Monte Carlo simulations using the distributions from the Prior Probability Distribution Functions defined in the Parameters page of the study file.

NOTE: It is important to check the verification plot of each response surface model for obvious outliers. If there are outliers, the cause of the outliers should be investigated. Sometimes the jobs that correspond to the outliers may need to be re-run or excluded from the analysis.

7.5.6.1 Proxy Model Verification Plot

In the **Proxy Analysis** node, if you select an objective function subnode, information about the proxy models for that objective function will be displayed, as shown in the following example, in which the **Model QC (Quality Check)** tab is selected:



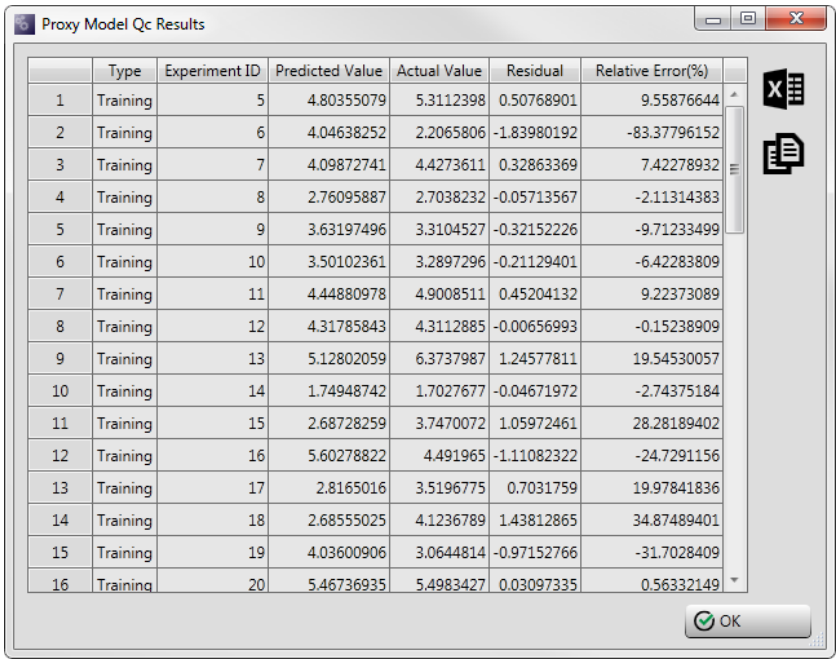
The Model QC plot shows how closely the proxy model predictions match actual values from the simulations. The 45 degree line represents a perfect match between the proxy model and actual simulation results. The closer the points are to the 45 degree line, the better the match between the predicted and actual data. The points that fall on the 45 degree line are those that are perfectly predicted. The points that are far from the 45 degree line are outliers. If there are outliers, the cause needs to be investigated before making use of the proxy model.

For polynomial regression models (Linear, Quadratic, Reduced Linear, and Reduced Quadratic), the lower and upper 95% confidence curves are superimposed on the actual by predicted plot. These confidence curves are useful for determining whether the regression model is statistically significant. The lower and upper 95% confidence curves are determined using the equations given in paper "Leverage Plots for General Linear Hypotheses", *John Sall, The American Statistician, November 1990, Vol. 44, No. 4*.

The dark blue points are the training experiments used by CMOST to create the proxy model. The green points are verification experiments used by CMOST to check if the proxy model created is a good proxy to the actual simulation results.

7.5.6.2 Proxy Model Data Table

The proxy model can be viewed in tabular form by right-clicking the verification plot then selecting **Data** to open the **Proxy Model Qc Results** table:



	Type	Experiment ID	Predicted Value	Actual Value	Residual	Relative Error(%)
1	Training	5	4.80355079	5.3112398	0.50768901	9.55876644
2	Training	6	4.04638252	2.2065806	-1.83980192	-83.37796152
3	Training	7	4.09872741	4.4273611	0.32863369	7.42278932
4	Training	8	2.76095887	2.7038232	-0.05713567	-2.11314383
5	Training	9	3.63197496	3.3104527	-0.32152226	-9.71233499
6	Training	10	3.50102361	3.2897296	-0.21129401	-6.42283809
7	Training	11	4.44880978	4.9008511	0.45204132	9.22373089
8	Training	12	4.31785843	4.3112885	-0.00656993	-0.15238909
9	Training	13	5.12802059	6.3737987	1.24577811	19.54530057
10	Training	14	1.74948742	1.7027677	-0.04671972	-2.74375184
11	Training	15	2.68728259	3.7470072	1.05972461	28.28189402
12	Training	16	5.60278822	4.491965	-1.11082322	-24.7291156
13	Training	17	2.8165016	3.5196775	0.7031759	19.97841836
14	Training	18	2.68555025	4.1236789	1.43812865	34.87489401
15	Training	19	4.03600906	3.0644814	-0.97152766	-31.7028409
16	Training	20	5.46736935	5.4983427	0.03097335	0.56332149

As with other results tables, you can save the contents of the **Proxy Model Qc Results** table to Excel or to the Windows clipboard, from which it can be pasted into another application.

7.5.6.3 Proxy Model Statistics

A detailed report of the response surface statistics is available by selecting the **Statistics** tab, shown in the following example (**Effect Screening Using Normalized Parameters** and **Coefficients in Terms of Actual Parameters** tables in the example have been cropped to fit on the page):

Objective Function Name: TotalOilProduced

Model Classification: Reduced Quadratic (alpha=0.1)

Summary of Fit

R-Square	0.815623
R-Square Adjusted	0.777004
R-Square Prediction	0.727249
Mean of Response	3.04555E+07
Standard Error	696330

Analysis of Variance

Source	Degrees of Freedom	Sum of Squares	Mean Square	F Ratio	Prob > F
Model	31	3.1745E+14	1.02403E+13	21.1195	<0.00001
Error	148	7.17615E+13	4.84875E+11		
Total	179	3.89212E+14			

Effect Screening Using Normalized Parameters (-1, +1)

Term	Coefficient	Standard Error	t Ratio	Prob > |t|	VIF
Intercept	3.00995E+07	153464	196.134	<0.00001	0.00
newW1_STO(125, 175)	179966	90291.2	1.99317	0.04808	1.11

Coefficients in Terms of Actual Parameters

Term	Coefficient
Intercept	3.27255E+08
newW1_STO	-201571
newW1_BHP	-247.432
newW1_UBAI	-3.31505E+06

Equation in Terms of Actual Parameters

```
TotalOilProduced=3.27255E+08-201571*newW1_STO-247.432*newW1_BHP-3.31505E+06*newW1_UBAI+1.87937E+06
*newW1_UBAJ+2.80745E+06*newW1_L3_UBAK-3.7744E+06*newW1_nLayers-266856*newW2_STO-1587.64*newW2_BHP+139668
*newW2_UBAI-2.66079E+07*newW2_UBAJ-4.54663E+06*newW2_L3_UBAK+2.15759E+06*newW2_nLayers+4.7654E+06
*oldWells_shutin+10987.9*newW1_STO*newW2_UBAJ+178970*newW1_UBAI*newW1_UBAJ+51.9381*newW1_UBAI
*newW2_BHP-268269*newW1_UBAJ*newW1_UBAJ-239843*newW1_UBAJ*newW1_L3_UBAK+8204.87*newW1_UBAJ
*newW2_STO+186304*newW1_UBAJ*newW2_UBAJ+343902*newW1_L3_UBAK*newW1_L3_UBAK-369398*newW1_L3_UBAK
*newW1_nLayers-357005*newW1_L3_UBAK*newW2_nLayers+239453*newW1_nLayers*newW2_UBAJ+336939
*newW1_nLayers*newW2_L3_UBAK+355.506*newW2_STO*newW2_STO+2.49569*newW2_STO*newW2_BHP+77.8491
*newW2_BHP*newW2_L3_UBAK+580694*newW2_UBAJ*newW2_UBAJ+142033*newW2_UBAJ*newW2_L3_UBAK-207316
*newW2_UBAJ*oldWells_shutin
```

For polynomial regression models (Linear, Quadratic, Reduced Linear, and Reduced Quadratic), see Proxy Modeling for an explanation of the statistical terms used in CMOST. As shown above, the **Statistics** tab contains the following five sections.

- Summary of Fit
- Analysis of Variance
- Effect Screening Using Normalized Parameters
- Coefficients in Terms of Actual Parameters
- Equation in Terms of Actual Parameters

You can copy the contents of the **Statistics** tab to Excel by selecting the desired portion of the tab with your mouse, or press CTRL+A to copy everything. Right-click in the selected area then select **Copy**. In Excel, click **Paste**.

7.5.6.4 Proxy Model Effect Estimates

For information on interpreting the information in this tab, refer to Linear Model Effect Estimates, Quadratic Model Effect Estimates, and Reduced Model Effect Estimates as appropriate.

NOTE: Depending on the number of bars displayed in the **Effect Estimates** plot, the font size of the Y-axis labels will change dynamically.

7.5.6.5 Sobol Analysis

For theoretical information about Sobol analysis, refer to Sobol Method.

Interpreting CMOST Results Using Sobol Analysis

The analysis of Sobol method results, based on the calculated main, total and interaction effects, is summarized as follows (Saltelli, Ratto, et al. 2008):

- Regardless of the strength of the model interactions, S_i (contribution to model output variance due to the variation of X_i alone) indicates the amount by which, on average, the output variance can be reduced with X_i fixed; hence, it is a measure of the main effect.
- By definition, S_{Ti} (total sensitivity index, the sum of all indices relating to X_i) is greater than S_i, or equal to S_i in the case where X_i has no interaction with other input factors. The difference $S_{Ti} - S_i$ is a measure of how much X_i interacts with other input factors.
- $S_{Ti} = 0$ implies that X_i is non-influential and can be fixed anywhere in its distribution without affecting the variance of the output.
- The sum of all S_i is equal to 1 for additive models and less than 1 for non-additive models. The difference:

  $$1 - \sum_i S_i$$

  is an indicator of the presence of interactions in the model.
- The sum of all S_{Ti} is always greater than 1. It is equal to 1 if the model is perfectly additive.

The Sobol method is powerful in quantifying the relative importance of input factors as well as their interactions. The main drawback, as with other variance-based methods, is the cost of the analysis, which in the case of computationally intensive models can become prohibitive even when using the approach described in Sobol Method for reducing the number of model executions (Saltelli, Ratto, et al. 2008). In terms of computation time, a large number of runs can be either trivial or unfeasible, depending on the model. This problem is addressed in CMOST by the use of advanced proxy models.

Another viable alternative for computationally intensive models is the Morris method, described in Morris Method. This elementary effect test is a good proxy for determining total sensitivity indices. If the model is expensive to run and has a large number of factors, the

elementary effect method can be used to reduce the number of factors and the variance-based analysis run with a reduced set of factors (Saltelli, Tarantola, et al. 2004).

Sobol Method – Example 1 – Polynomial Function

In this example, we demonstrate the Sobol method using a simple polynomial function with three parameters, with X_2 being the most important input parameter, and interactions between parameters X_1 and X_2, as follows:

$$Y = X_1 + 10X_2 + X_3 + 50X_1X_2$$

All parameters are uniformly distributed between [0,1].

The Sobol analysis for Example 1 is shown in *Figure 1* and *Table 1*:

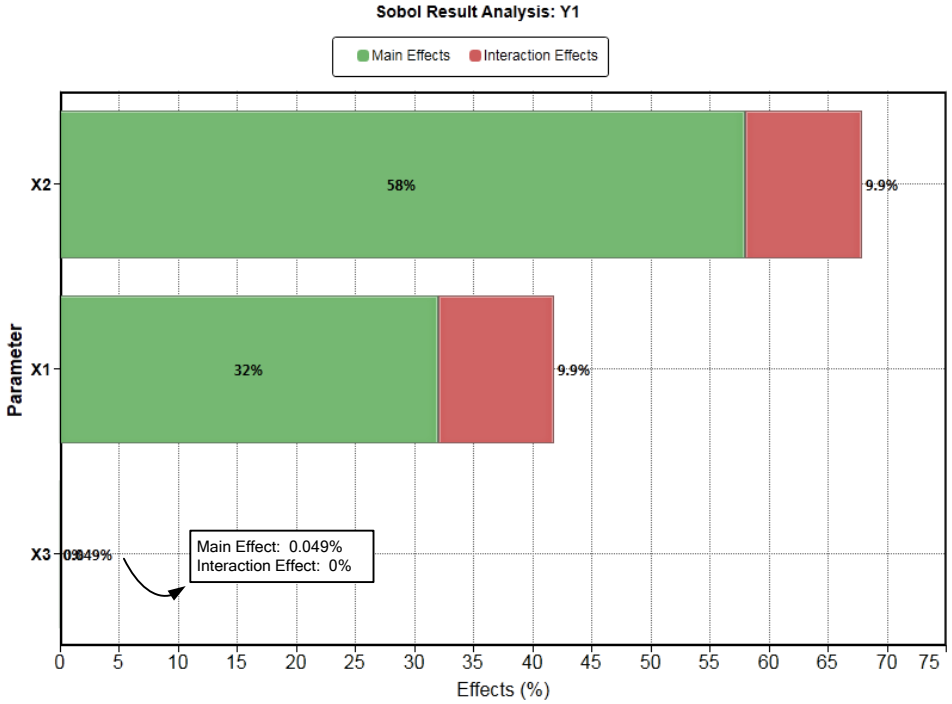


Figure 1 Sobol Chart for Polynomial Objective Function

Table 1 Sobol Values for Polynomial Objective Function

Parameter Name	Main Effects	Interaction Effects	Total Effects
X2	58%	10%	68%
X1	32%	10%	42%
X3	~0%	0%	~0%

In this example, parameter X_2 is clearly the major contributor to the variability of the objective function. Based on the results, 58% of the output variance, on average, can be

reduced if X_2 can be fixed. This is followed by parameter X_1, which contributes 32% of the objective function variability.

The difference:

$$1 - \sum_{i} S_i = 1 - (0.58 + 0.32)$$

indicates a 10% interaction effect. Since the interaction effect for parameter X_3 is zero, we can conclude that there is interaction between parameters X_1 and X_2 only. This can also be verified by inspecting the polynomial function.

Sobol Method – Example 2 – Cyclic Steam Pilot

In this example, we study a single-well cyclic-steam pilot over three cycles. The Sobol method is used to determine the sensitivity of an objective function to the model's nine different parameters and their value ranges. In particular, we want to see how much each of our history matching parameters impacts the production and injection of the well, as well as the linearity and nonlinearity of their effect on the objective functions. This will be useful for determining which parameters we should modify when we begin history matching. The most sensitive parameters will have a large impact on the results so these are the parameters that we should modify during the history matching process. Less-sensitive parameters can be held at their original value then modified at the end of the matching process to fine tune the results.

The Sobol analysis for Example 2 is shown in *Figure 2*, *Figure 3*, and *Figure 4* (as shown, we have zoomed into areas with low values to determine the main and interaction effects):

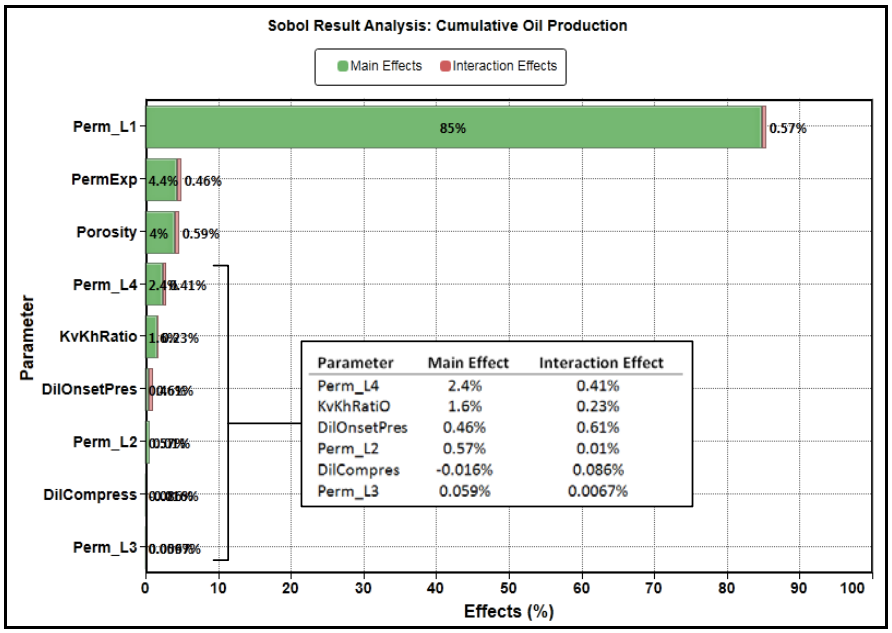


Figure 2 Sobol Chart for Cumulative Oil Production Objective Function

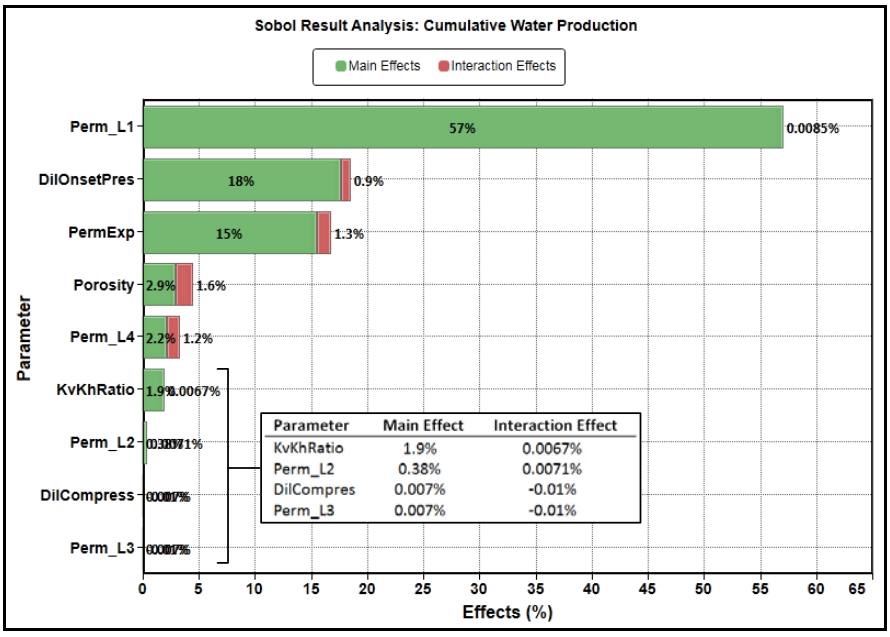


Figure 3 Sobol Chart for Cumulative Water Production Objective Function

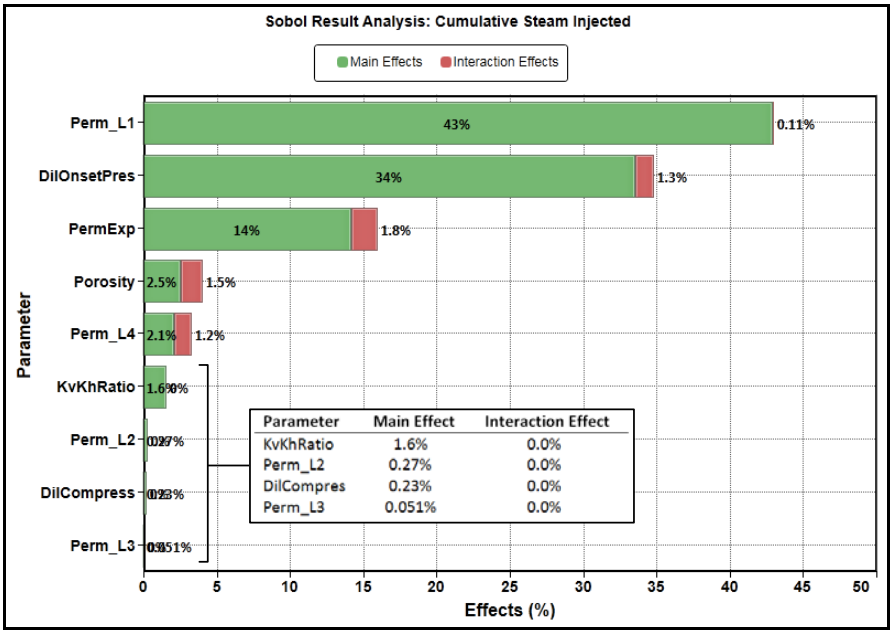


Figure 4 Sobol Chart for Cumulative Steam Injected Objective Function

Based on the results, it is obvious that parameter 5 (permeability in layer 1) has the greatest impact on all three objective functions, and is therefore the most influential factor in our model. While this is the case in general, parameter 2, dilation onset pressure, contributes much more to the variability of water production and cumulative steam injected when

compared with cumulative oil production. At the same time, all objective functions seem to have a low sensitivity to permeability in layers 2 and 3, and dilation compressibility.

Based on the values obtained for the total and main effects of all parameters, by objective function, it can also be concluded that interaction effects are not strong in the model.

7.5.6.6 Morris Analysis

For theoretical information about Morris analysis, refer to Morris Method.

Interpreting CMOST Results Using Morris Analysis

The two Morris measures, mean and standard deviation, are analyzed together (for example, on a two-dimensional graph) in order to rank input factors in order of importance and identify those inputs which do not influence output variability (Campolongo and Saltelli 2007).

Mean μ^* provides an assessment of the overall influence of the factor on the output. Standard deviation σ estimates the ensemble of the factor's effects, whether nonlinear and/or due to interactions with other factors.

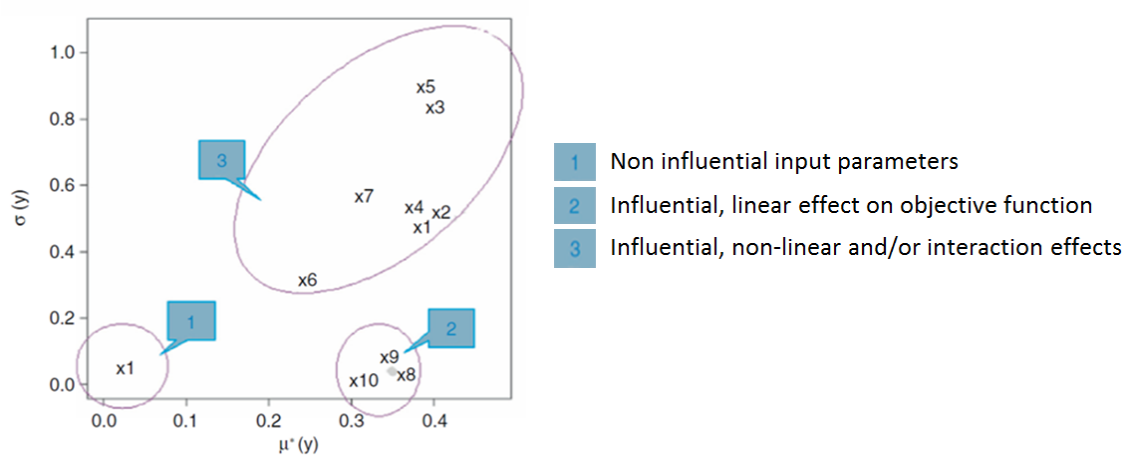
An intuitive explanation of the meaning of σ is as follows (Saltelli, Ratto, et al. 2008). Assume that for factor X_i we obtain a high value of σ. The elementary effects relative to this factor thus differ notably from one another, implying that the value of the elementary effect is strongly affected by the choice of the sample point at which it is computed, i.e. by the choice of the other factors' values. In contrast, a low value of σ indicates similar values among the elementary effects, implying that the effect of X_i is almost independent of the values of the other factors.

- If all samples of the elementary effect of the i^{th} input factor are zero, then X_i does not affect output Y, and the sample mean and standard deviation will both be zero.
- If all elementary effects have the same value, then Y is a linear function of X_i. The standard deviation of the elementary effects will then be zero.
- For more complex situations, due to interactions between factors and nonlinearity, Morris states that if the mean of the elementary effects is relatively large and the standard deviation is relatively small, the effect of X_i on Y is "mildly nonlinear". If the opposite is true, i.e., the mean is relatively small and the standard deviation is relatively large, then the effect is "strongly nonlinear".

As a rule of thumb (Ekstrom 2005):

a) a high mean indicates a factor with an important overall influence on the output, and

b) a high standard deviation indicates that either the factor is interacting with other factors or the factor has strong nonlinear effects on the output.

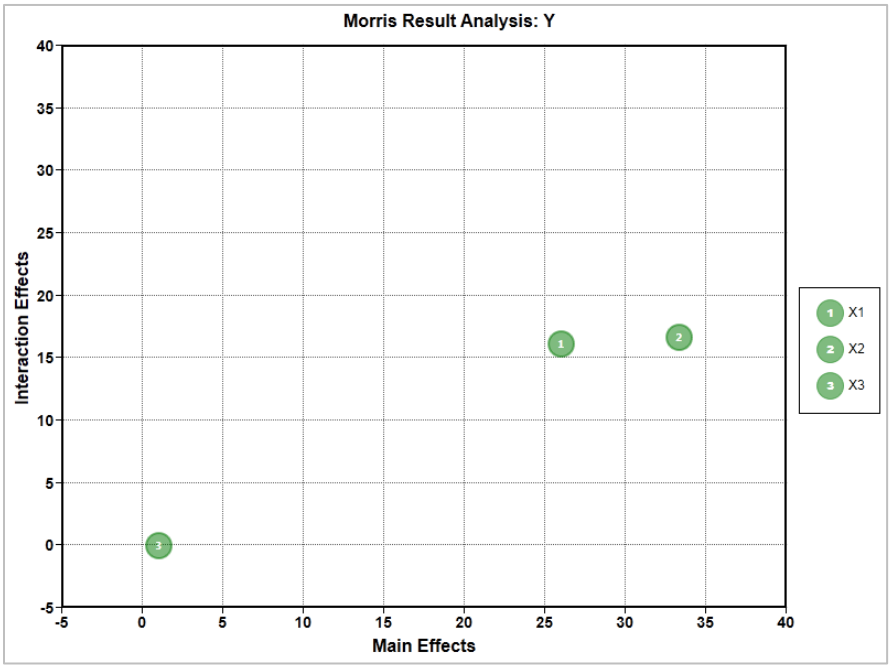


Morris Analysis - Example 1 – Polynomial Function

In this example, as we did for the Sobol method, we demonstrate the Morris method using a simple polynomial function with three parameters, with X_2 being the most important input parameter, and interactions between parameters X_1 and X_2, as follows:

$$Y = X_1 + 10X_2 + X_3 + 50X_1X_2$$

All parameters are uniformly distributed between [0,1]. The Morris analysis for Example 1 is shown below:



The Morris chart shows that parameter X_2, with the highest value of μ^*, is the most important, followed by parameters X_1 and X_3. In terms of interaction, parameter X_3 has a value of *0* on the Y axis, indicating no interaction effects for this parameter, while parameters X_1 and X_2 have similar interaction effects.

Morris Analysis - Example 2 – Cyclic Steam Pilot

In this example, we study a single-well cyclic-steam pilot over three cycles. As with the Sobol method, the Morris method is used to determine the sensitivity of an objective function to the model's nine different parameters and their value ranges. In particular, we want to see how much each of our history matching parameters impacts the production and injection of the well, as well as the linearity and nonlinearity of their effect on the objective functions. This will be useful for determining which parameters we should modify when we begin history matching. The most sensitive parameters will have a large impact on the results so these are the parameters that we should modify during the history matching process. Less-sensitive parameters can be held at their original value then modified at the end of the matching process to fine tune the results.

The Morris analysis results for Example 2 are shown in *Figure 5*, *Figure 6*, and *Figure 7*.

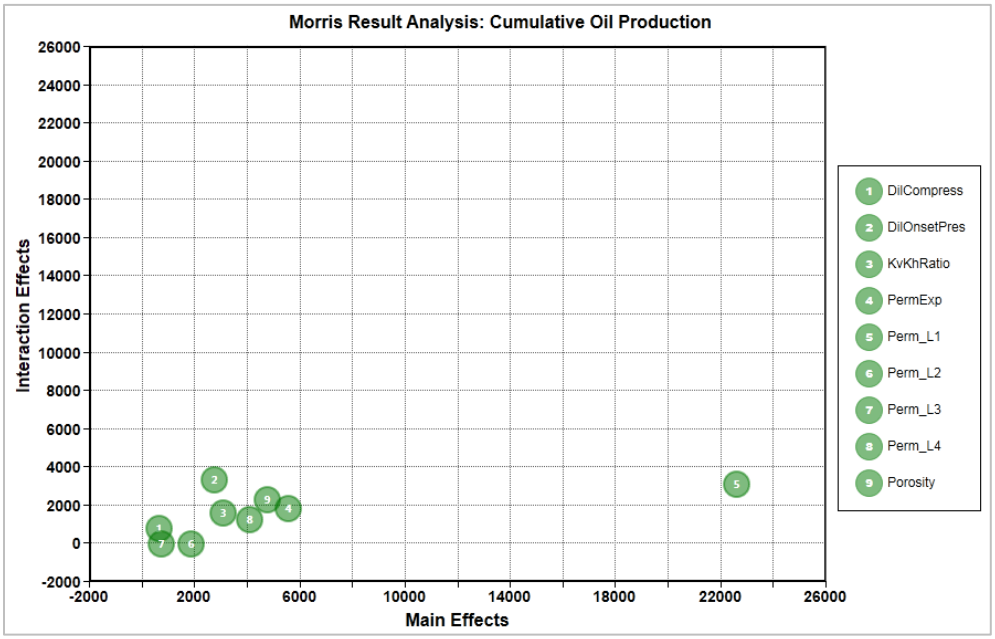


Figure 5 Morris Chart for Cumulative Oil Production Objective Function

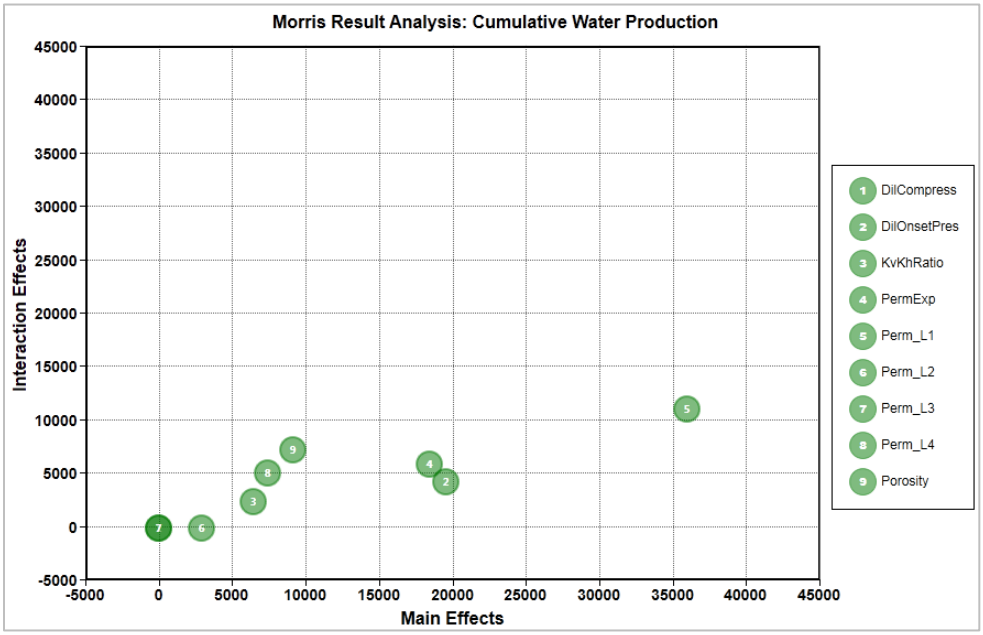


Figure 6 Morris Chart for Cumulative Water Production Objective Function

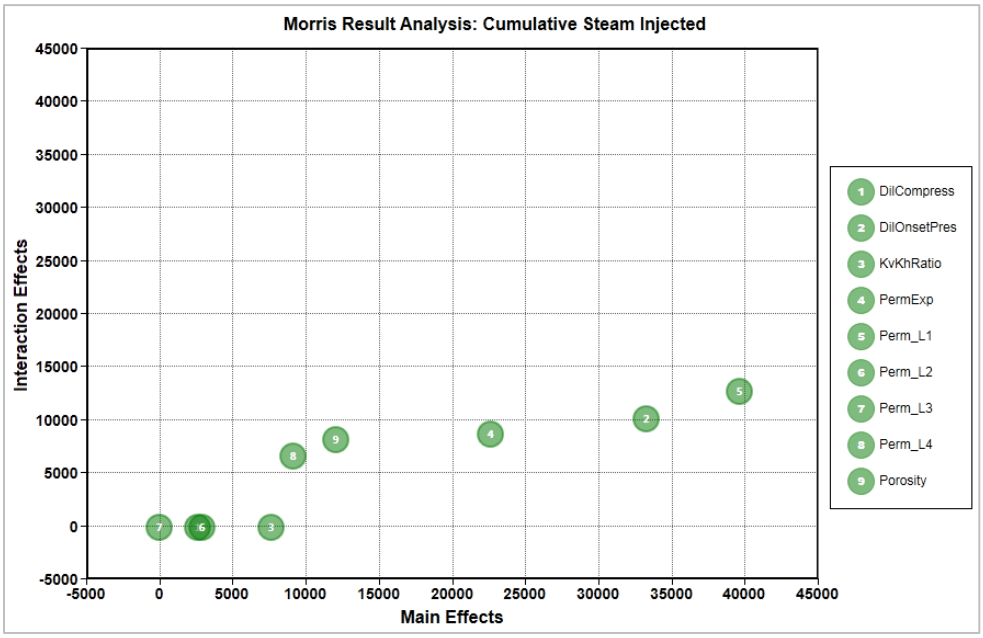


Figure 7 Morris Chart for Cumulative Steam Injected Objective Function

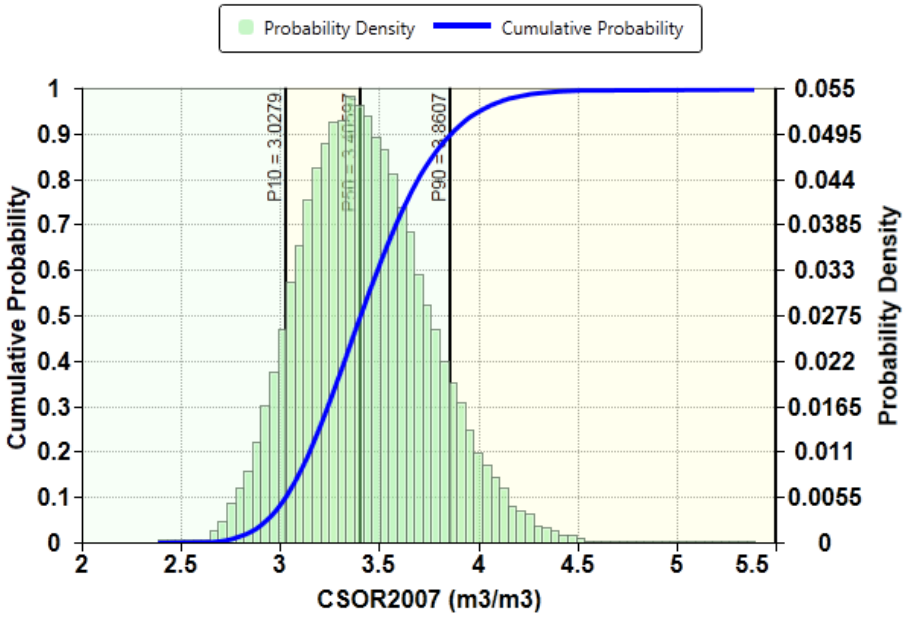
Based on the Morris analysis, consistent with the Sobol analysis, it is obvious that parameter 5 (permeability in layer 1) has the most impact on all three objective functions, and hence is the most influential factor in our model. While this may generally be the case, parameter 2

plays a more significant role in water production and injected steam than it does in oil production. At the same time, the objective functions seem to be least sensitive to parameters 7, 3 and 1 (permeability in layer 3, K_v to K_h ratio, and dilation compressibility, respectively).

The relatively low values obtained for the Y axis of the Morris chart for all objective functions, suggests that strong interactions between parameters do not exist in the model and the parameters have an almost linear effect on the outputs. This changes slightly for cumulative steam injected, particularly for parameter 2, followed by parameters 4 and 5, which indicates a higher nonlinear relationship between these parameters for this objective function, and/or stronger interaction effects between these and other parameters.

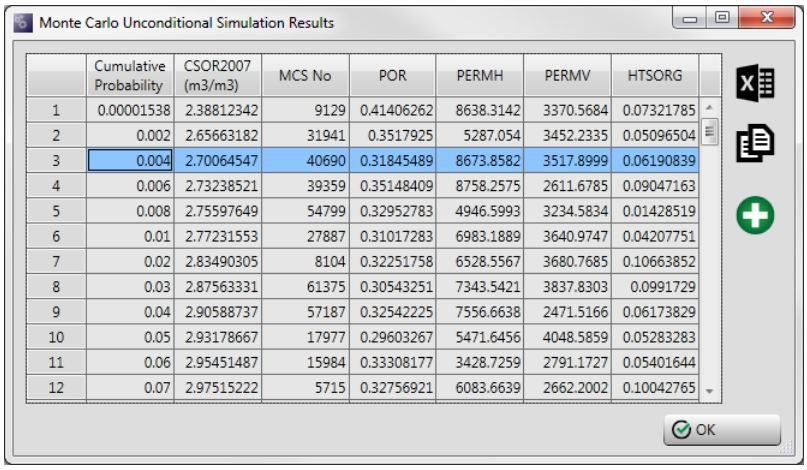
7.5.6.7 Monte Carlo Simulations

In the case of uncertainty analyses, the **Monte Carlo Simulation** tab shows the distribution of values for each objective function with all uncertain parameters sampled from the prior probability density functions, as illustrated in the following example:



The plot shows a histogram of objective function values, to illustrate the shape of the probability density function, as well as the cumulative probability. P10, P50, and P90 values are highlighted.

Data can be viewed in tabular form by right-clicking the plot and then selecting **Data**. The **Monte Carlo Unconditional Simulation Results** dialog box is displayed, as shown in the following example:



	Cumulative Probability	CSOR2007 (m3/m3)	MCS No	POR	PERMH	PERMV	HTSORG
1	0.00001538	2.38812342	9129	0.41406262	8638.3142	3370.5684	0.07321785
2	0.002	2.65663182	31941	0.3517925	5287.054	3452.2335	0.05096504
3	0.004	2.70064547	40690	0.31845489	8673.8582	3517.8999	0.06190839
4	0.006	2.73238521	39359	0.35148409	8758.2575	2611.6785	0.09047163
5	0.008	2.75597649	54799	0.32952783	4946.5993	3234.5834	0.01428519
6	0.01	2.77231553	27887	0.31017283	6983.1889	3640.9747	0.04207751
7	0.02	2.83490305	8104	0.32251758	6528.5567	3680.7685	0.10663852
8	0.03	2.87563331	61375	0.30543251	7343.5421	3837.8303	0.0991729
9	0.04	2.90588737	57187	0.32542225	7556.6638	2471.5166	0.06173829
10	0.05	2.93178667	17977	0.29603267	5471.6456	4048.5859	0.05283283
11	0.06	2.95451487	15984	0.33308177	3428.7259	2791.1727	0.05401644
12	0.07	2.97515222	5715	0.32756921	6083.6639	2662.2002	0.10042765

The table shows the distribution data for selected Monte Carlo simulations, sorted by cumulative probability values. In this data table, you can find the objective function value and the corresponding parameter values for typical cumulative probabilities, such as P10, P50, and P90.

Click to copy the data to the Windows clipboard, or to export it to an Excel spreadsheet. If you select rows of the table using CTRL and SHIFT keys, then only those rows will be copied to the clipboard or spreadsheet.

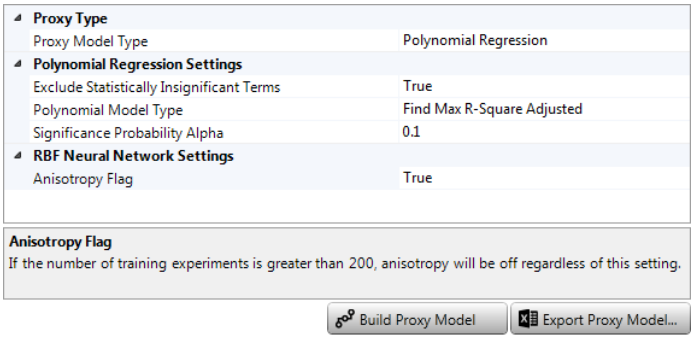
Similarly, select one or more data rows then click to create new experiments with those parameter settings.

You can also right-click the plot then select commands to save the image in one of several formats, or copy the image to the Windows clipboard.

If you right-click the plot and then select **All Generated Monte Carlo Simulation Data**, you will open a table that shows the results of all the Monte Carlo simulations. Again, you can copy the content of the table to the Windows clipboard or to a spreadsheet, as outlined above.

7.5.6.8 Proxy Settings

Through the **Proxy Settings** tab, you can build and export (to Excel) a selected proxy model type. First select the proxy type (*Polynomial Regression* in the following example) then configure the available settings:



After you enter the settings, click the 

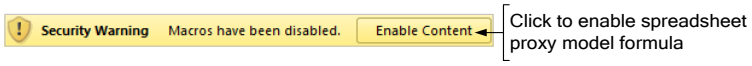
button to build the specified proxy model.

Exporting the Proxy Model to Excel

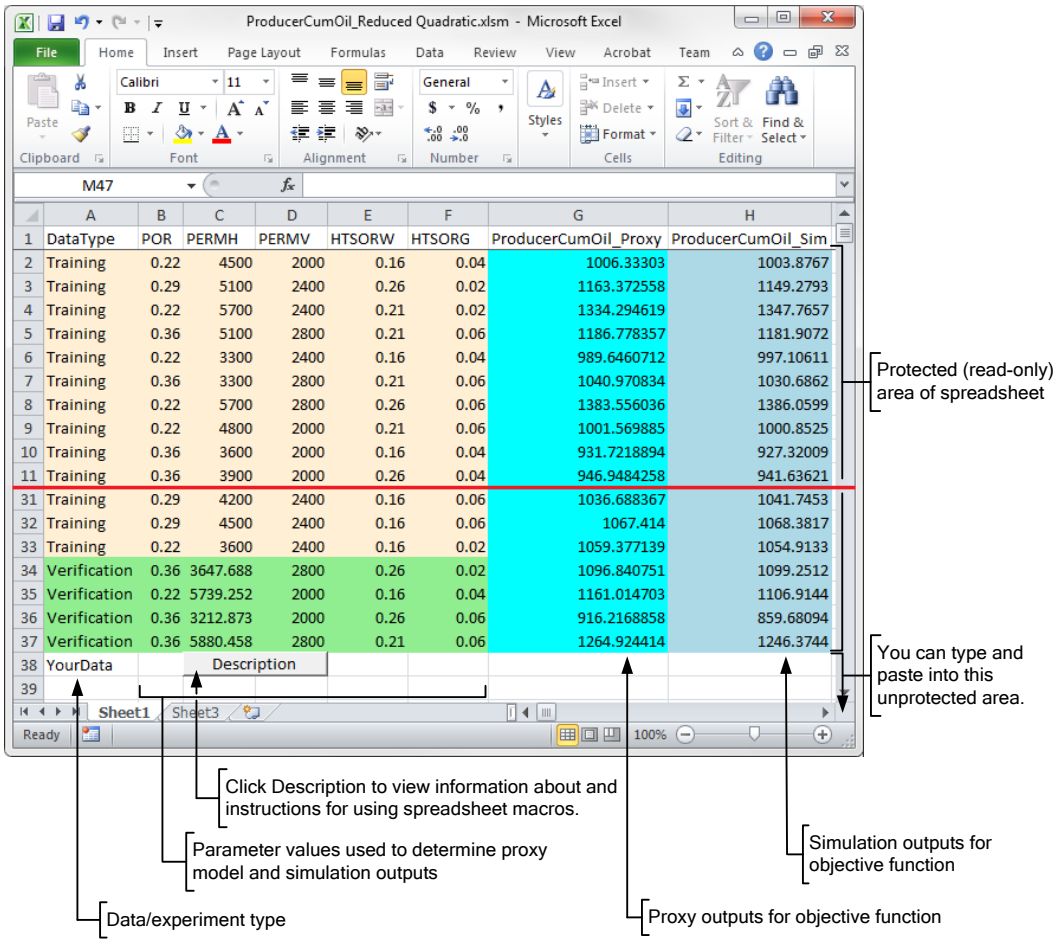
Once you have built the proxy model, the **Export Proxy Model** button will be enabled and you will be able to export the proxy model to Excel, if desired. The proxy model for the objective function is entered as an Excel formula, CMG_POL in the case of a polynomial proxy model and CMG_RBF in the case of an RBF proxy model. Through Excel, you can then investigate the effect that changes in parameter values have on the particular objective function:

1. Click the 

button. An **Explorer** session will open. The study folder is selected as the destination, and the **File name** field is populated with the default file name, which contains the objective function name and the proxy type.
2. In **Explorer**, change the destination folder and file name, if desired, then click **Save**. The first time you open the spreadsheet, you will need to click the **Enable Content** button at the top to enable the spreadsheet macro:

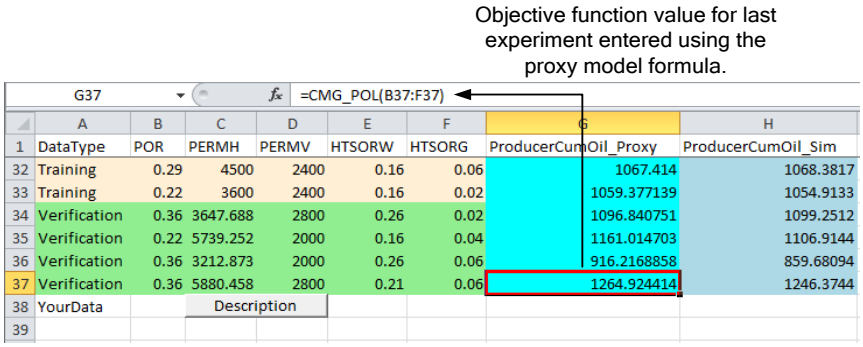


3. CMOST will save the proxy model as an Excel macro, and the simulated and proxy model objective function predictions for all completed experiments, as shown below:

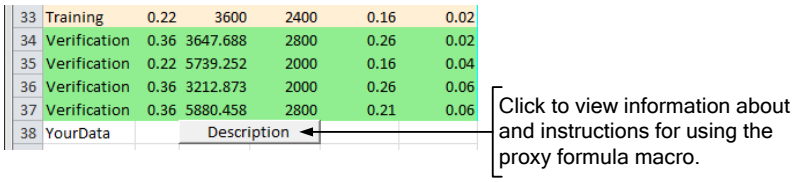


	A	B	C	D	E	F	G	H
1	DataType	POR	PERMH	PERMV	HTSORW	HTSORG	ProducerCumOil_Proxy	ProducerCumOil_Sim
2	Training	0.22	4500	2000	0.16	0.04	1006.33303	1003.8767
3	Training	0.29	5100	2400	0.26	0.02	1163.372558	1149.2793
4	Training	0.22	5700	2400	0.21	0.02	1334.294619	1347.7657
5	Training	0.36	5100	2800	0.21	0.06	1186.778357	1181.9072
6	Training	0.22	3300	2400	0.16	0.04	989.6460712	997.10611
7	Training	0.36	3300	2800	0.21	0.06	1040.970834	1030.6862
8	Training	0.22	5700	2800	0.26	0.06	1383.556036	1386.0599
9	Training	0.22	4800	2000	0.21	0.06	1001.569885	1000.8525
10	Training	0.36	3600	2000	0.16	0.04	931.7218894	927.32009
11	Training	0.36	3900	2000	0.26	0.04	946.9484258	941.63621
31	Training	0.29	4200	2400	0.16	0.06	1036.688367	1041.7453
32	Training	0.29	4500	2400	0.16	0.06	1067.414	1068.3817
33	Training	0.22	3600	2400	0.16	0.02	1059.377139	1054.9133
34	Verification	0.36	3647.688	2800	0.26	0.02	1096.840751	1099.2512
35	Verification	0.22	5739.252	2000	0.16	0.04	1161.014703	1106.9144
36	Verification	0.36	3212.873	2000	0.26	0.06	916.2168858	859.68094
37	Verification	0.36	5880.458	2800	0.21	0.06	1264.924414	1246.3744
38	YourData		Description					
39								

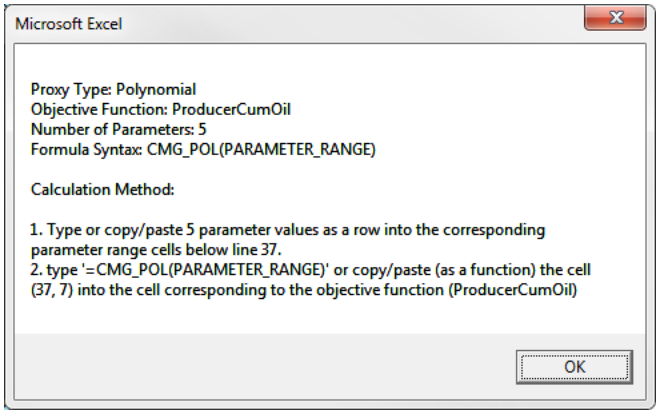
4. Once you enable content, you will observe the following:
    a. The values of the objective functions (*ProducerCumOil_Proxy* in our example) will be displayed as numbered values for all experiments, with the exception of the last experiment in the spreadsheet, in which the objective function will be displayed in terms of the proxy model formula, CMG_POL in the case of a polynomial proxy model, and CMG_RBF in the case of an RBF proxy model:



    b. The **Description** button will be available, as shown below:



    Click the **Description** button for information about, and instructions on how to use, the proxy formula. As shown in the following example, CMG_POL calculates values of the objective function using parameter values, which have been entered in a row in the **Your Data** area, in a particular order:

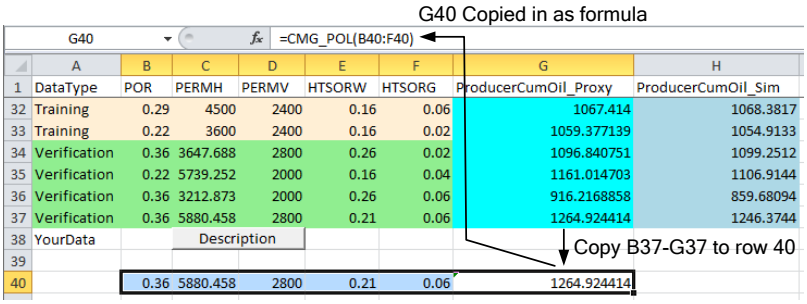


5. You cannot enter data in the rows associated with the shaded areas of the spreadsheet as these are protected; however, the area below the experiment data,

labeled **Your Data**, is available for entering data into the proxy model spreadsheet, as shown in the following example:

a. You can enter your data in one of two ways:

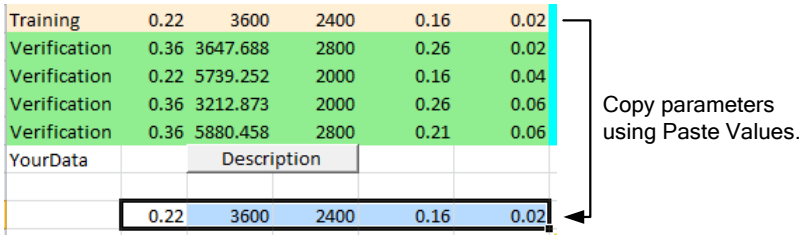
Method 1: As noted above, the objective function for the last experiment in the spreadsheet is entered as the proxy model formula (CMG_POL in the following example). Because of this, you can copy the parameters and the formula for the objective function for this experiment to the **Your Data** area, using **Paste | Formulas** (If you do not specify **Formulas**, the objective function formula will be copied in as a fixed variable).



	A	B	C	D	E	F	G	H
1	DataType	POR	PERMH	PERMV	HTSORW	HTSORG	ProducerCumOil_Proxy	ProducerCumOil_Sim
32	Training	0.29	4500	2400	0.16	0.06	1067.414	1068.3817
33	Training	0.22	3600	2400	0.16	0.02	1059.377139	1054.9133
34	Verification	0.36	3647.688	2800	0.26	0.02	1096.840751	1099.2512
35	Verification	0.22	5739.252	2000	0.16	0.04	1161.014703	1106.9144
36	Verification	0.36	3212.873	2000	0.26	0.06	916.2168858	859.68094
37	Verification	0.36	5880.458	2800	0.21	0.06	1264.924414	1246.3744
38	YourData		Description					
39								
40		0.36	5880.458	2800	0.21	0.06	1264.924414	

Method 2: You can copy the parameters from any experiment then manually enter the formula as follows.

Select the parameters for one of the experiments and copy it into the area below **Your Data**:



Training	0.22	3600	2400	0.16	0.02
Verification	0.36	3647.688	2800	0.26	0.02
Verification	0.22	5739.252	2000	0.16	0.04
Verification	0.36	3212.873	2000	0.26	0.06
Verification	0.36	5880.458	2800	0.21	0.06
YourData		Description			
	0.22	3600	2400	0.16	0.02

In an empty cell, to the right of the parameters for example, manually enter the proxy formula, specifying the range of the parameter values:

YourData		Description				
	0.22	3600	2400	0.16	0.02	=CMG_POL(B40:F40)

After you click out of the cell containing the proxy formula, the formula will be replaced with the value it calculates; however, the underlying formula will be retained and the result updated if you change one or more of the parameter values.

YourData		Description				
	0.22	3600	2400	0.16	0.02	1059.377139

b. In the **Your Data** area, you can change parameter values to see the effect of the change on the objective function, as shown below, where we have changed the value of POR from *0.22* to *0.2*.

YourData		Description				
	0.2	3600	2400	0.16	0.02	1075.005152

c. You can also copy the data to new rows in the **Your Data** area, remembering to use **Paste | Formula**, then modify the parameter values in the new rows, as shown below, where we have copied to multiple rows then changed the values of POR in each row:

YourData		Description				
	0.18	3600	2400	0.16	0.02	1093.49556
	0.19	3600	2400	0.16	0.02	1083.892557
	0.195	3600	2400	0.16	0.02	1079.359405
	0.2	3600	2400	0.16	0.02	1075.005152
	0.205	3600	2400	0.16	0.02	1070.8298
	0.21	3600	2400	0.16	0.02	1066.833346
	0.22	3600	2400	0.16	0.02	1059.377139

NOTE: If the number of parameters entered does not equal the number expected by the formula, or if one of the parameters does not have a numerical value, the proxy formula will output an error message and will set the value of the objective function to zero.

7. Save the spreadsheet changes, as necessary.

8 常用和高级选项 (General and Advanced Operations)

8.1 CMM 文本编辑器

8.1.1 关于 CMM文本编辑器

CMOST CMM文本编辑器是查看和编辑CMOST主文件（.cmm）和相关的Include 文件(.inc)。该文本编辑器可用于:

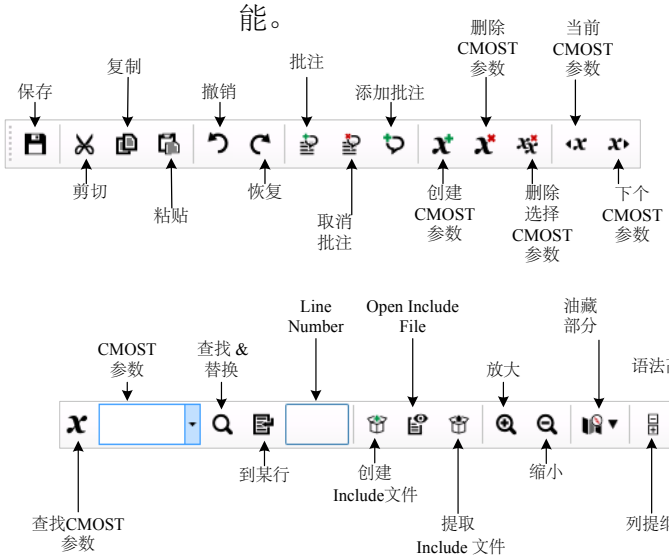
- 创建/插入/删除CMOST参数。
- 添加注释/取消注释。
- 创建/编辑/提取Include文件。
- 通过编辑文件快速查找关键字等。

8.1.1.1 启动CMOST CMM文本编辑器

为了启动 CMM文本编辑器，在 General Properties 页面，点击Edit 按钮。

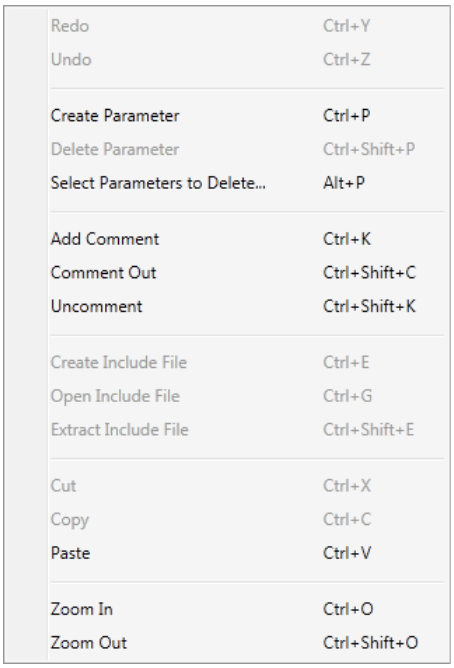
CMOST CMM 文本编辑器和状态栏

在工具栏按钮上移动光标时会显示其功能。



8.1.1.2 CMOST CMM 文本编辑器右键菜单栏

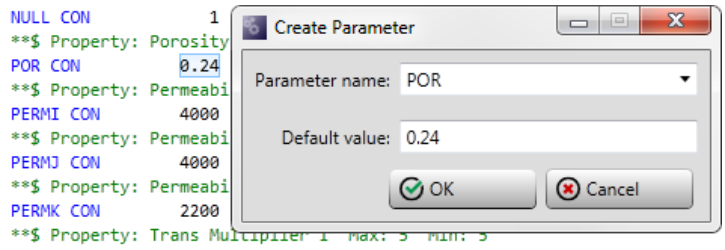
多数功能可以通过右键菜单栏实现。右键文本编辑器显示菜单栏内容：



注意：更多信息参考Keyboard Shortcuts 。

8.1.1.3 创建/插入CMOST Parameter

选择需要CMOST参数替代的部分，然后点击**Create CMOST Parameter** 按钮，输入参数名称和缺省值（可选）。插入一个已经存在的参数，在下拉菜单中选择参数名称。如果选择的参数已经被定义，可以使用先前的缺省值。



点击**OK**。选中的文本被CMOST参数替代。

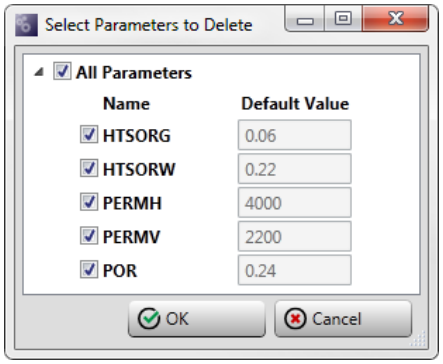
```
POR CON          <CMOST>this[0.24]=POR</CMOST>
```

8.1.1.4 删除CMOST参数

选择定义好的CMOST参数，然后点击 **Delete CMOSTParameter** 按钮。CMOST参数将被删除。如果定义为缺省的名称，参数会被缺省值所代替。

8.1.1.5 删除选中的CMOST参数

为了删除一个或多个CMOST参数，点击 **Delete Selected CMOST Parameters** 按钮。 **Select Parameters to Delete**对话框如下：



清除**All Parameters** 复选框。选择想要删除的参数，然后点击**OK**，提示再次确认，点击**OK**。所有选中的参数将会从CMM文件中删除，并用缺省值所代替。

8.1.2 添加注释

8.1.2.1 添加注释

将光标移至想要添加注释的部分，然后点击Add Comment 按钮来添加一个新注释。

8.1.2.2 选中某行添加注释

选择一行或多行，然后点击**Comment Lines** 按钮，添加注释行。

8.1.2.3 取消注释

选择一行或多行注释，点击**Uncomment Lines** **按钮**，删除 选中的注释行。

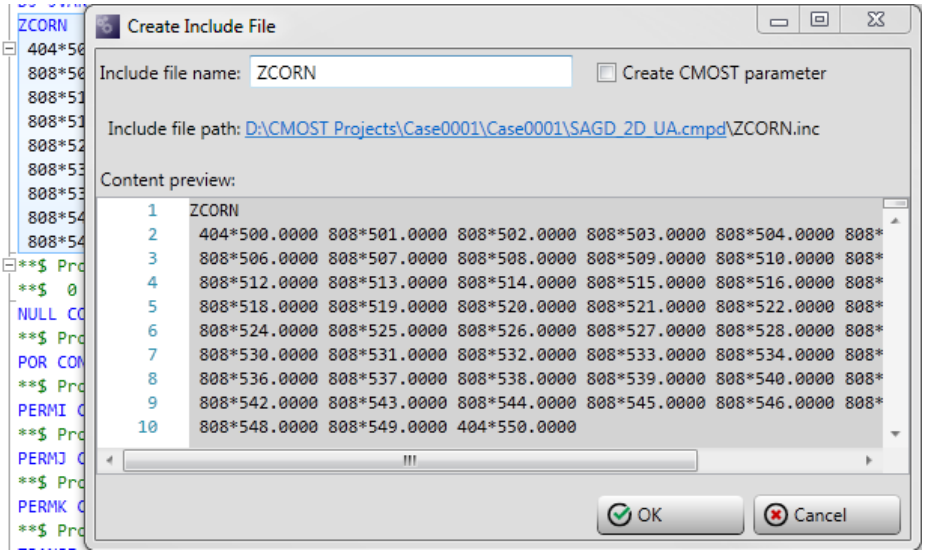
注意：仅支持 CMG缺省的注释符号 **

8.1.3 使用Include 文件

8.1.3.1 创建 Include文件

选中CMM文件中的一部分，然后点击**Create Include File** 按钮，在对话框**Create Include File**中输入Include文件名称。

注意：Include文件被保存在与主文件相同的文件夹内。



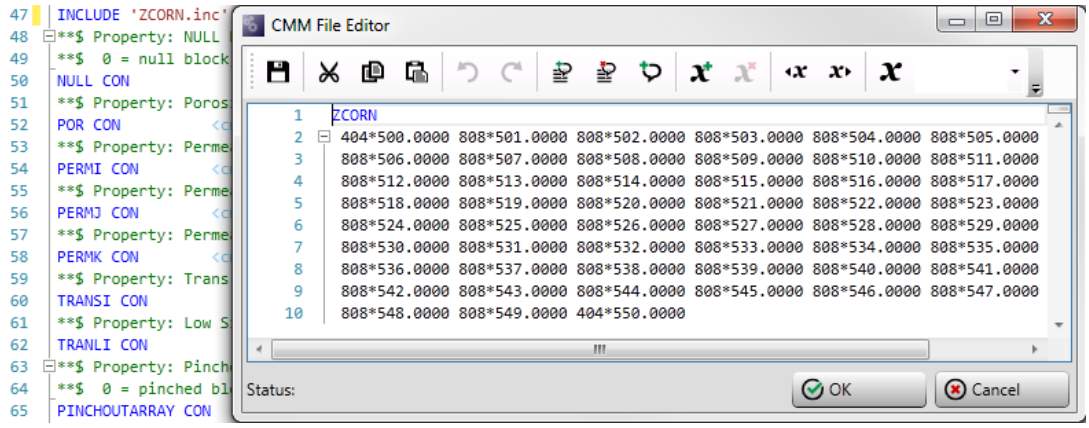
点击**OK**。Include文件创建完成，选中的文本部分被Include行所代替，如下：

```
INCLUDE 'ZCORN.inc'
```

在**Create/Extract Include File** 对话框，如果选择 **Create CMOST parameter** 复选框，Include文件会被创建为CMOST参数，并且使用缺省值。

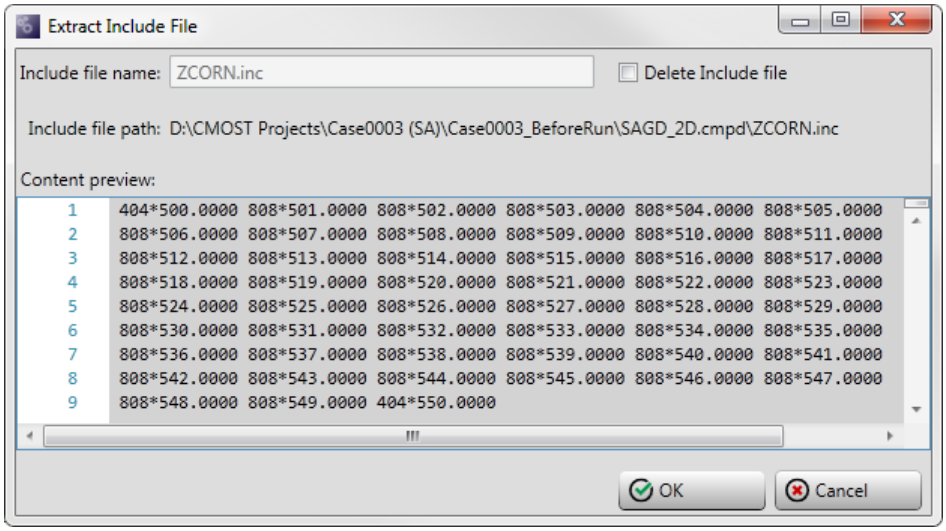
8.1.3.2 编辑Include 文件

将光标移至Include文件行，然后点击 Open Include File 按钮在新CMM文本编辑器部分打开Include文件。



8.1.3.3 提取Include文件

将光标移至Include文件行，然后点击**Extract Include File** 按钮，显示**Extract Include File** 对话框。点击OK。主文件中Include文件行由Include文件内容所代替。 如果选择**Delete include file**，提取之后，Include文件会被删除。



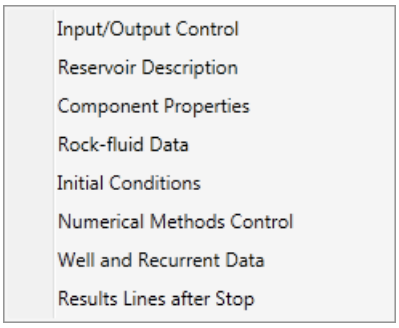
8.1.4 导航栏工具

8.1.4.1 到某一行

在工具栏Line Number中输入想要查询的行号，然后点击Enter。 在任意时间，点击 都可以找到想要的某一行。

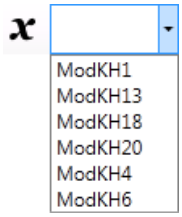
8.1.4.2 到油藏描述部分

点击**Reservoir Sections** 按钮，在下拉菜单中选择想要查看的油藏描述部分。



8.1.4.3 到CMOST参数

在屏幕右上方下拉菜单中点击CMOST Parameters，找到包含参数某一行。如果某个CMOST参数出现在文件的多个位置，点击Search CMOST Paramet 按钮，找到下个出现该参数的位置。列表中的CMOST参数按照字母数字顺序排列，如下例所示：



8.1.5 其他功能

8.1.5.1 提纲触发器

点击**Toggle Outlining** 按钮，，扩大多行注释和数据块。

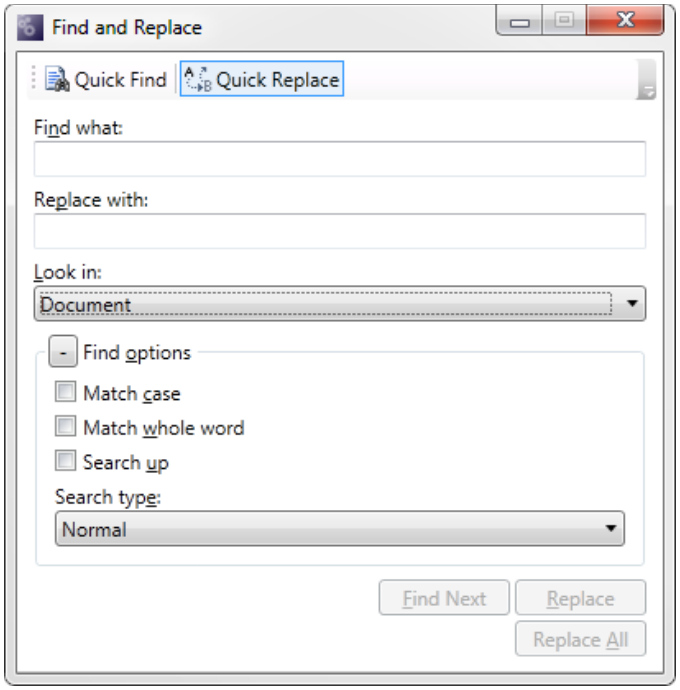
可以通过点击 ，在头节点收缩 CMM文件或者点击 来展开文件。移动鼠标至头节点收缩部分查看收缩行的内容。

8.1.5.2 语法高亮显示切换

点击 **Toggle Syntax Highlighting** 按钮来打开或关闭CMG关键字语法高亮显示。对于较大的CMM文件，将加速文件的导航。

8.1.5.3 查找/替换文本

在工具栏点击 **Find and Replace** 按钮来打开Find and Replace 对话框。支持查找常规表达式。



8.1.5.4 块状选项

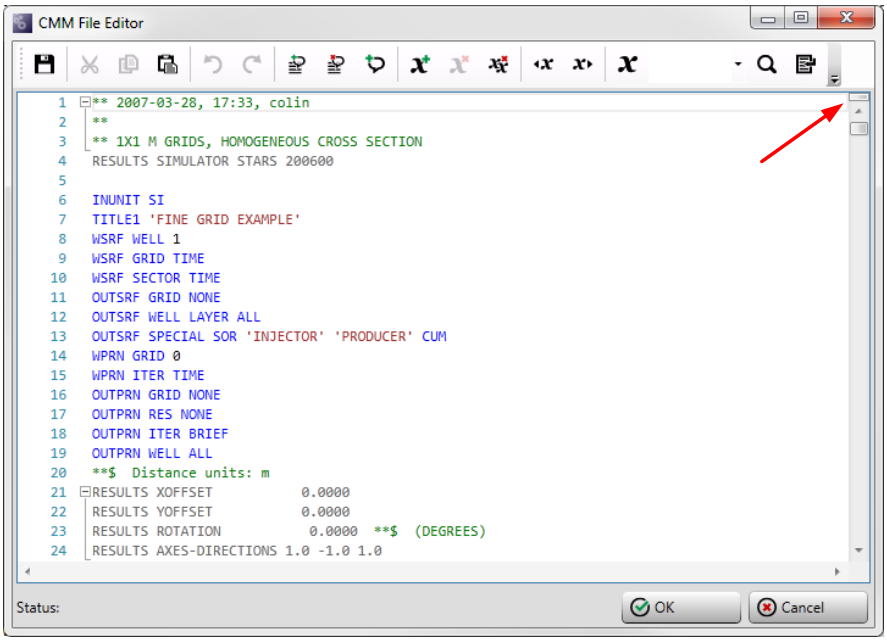
块状选项用来选择矩形文本。

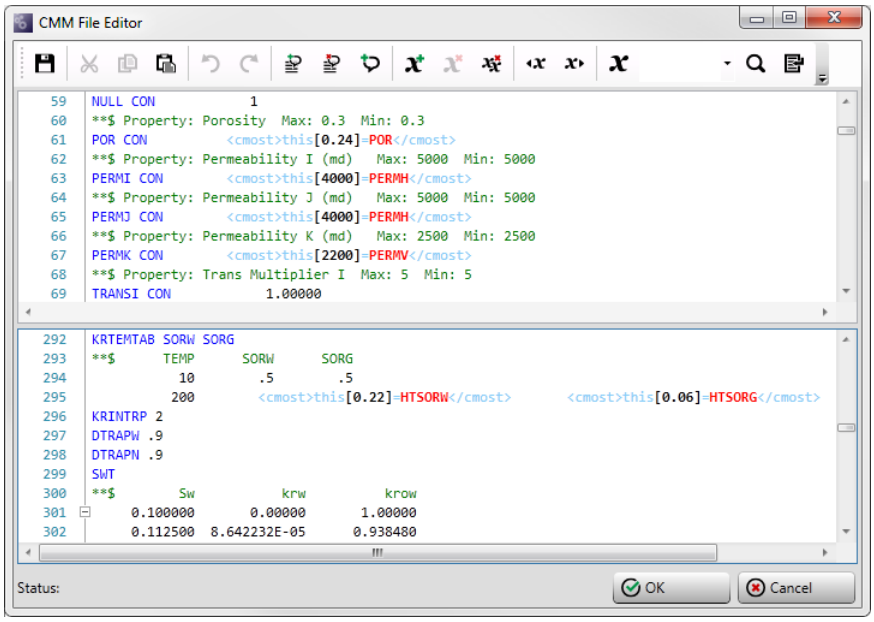
- 利用鼠标选择块状部分，按住ALT键，点击文本处，然后拖至选择部分。
- 也可以使用键盘选择块状部分文本，按住SHIFT+ALT键，然后按住向上箭头。

```
SWT
**$          Sw           krw          krow
         0.100000      0.00000       1.00000
         0.112500  8.642232E-05      0.938480
         0.125000  3.457437E-04      0.878913
         0.137500  7.779950E-04      0.821298
         0.150000  1.383193E-03      0.765637
         0.162500  2.161348E-03      0.711928
         0.175000  3.112470E-03      0.660172
         0.187500  4.236566E-03      0.610369
         0.200000  5.533642E-03      0.562518
         0.212500  7.003702E-03      0.516621
         0.225000  8.646753E-03      0.472676
         0.237500  1.046280E-02      0.430685
         0.250000  1.245184E-02      0.390646
         0.262500  1.461388E-02      0.352560
         0.275000  1.694893E-02      0.316427
         0.287500  1.945699E-02      0.282247
         0.300000  2.213805E-02      0.250020
         0.312500  2.499213E-02      0.219745
```

8.1.5.5 多视图文件

在编辑器右上角，选择然后拖住按钮到某一位置（如下图），就可以多视图查看某一文件。垂直方向上最多可支持显示两个视图，水平方向暂不支持视图劈分





8.1.6 关键字快捷键

命令	快捷键
Redo	CTRL+Y
Undo	CTRL+Z
Cut	CTRL+X
Copy	CTRL+C
Paste	CTRL+V
Find	CTRL+F
Save File	CTRL+S
Block Selection	ALT+ Mouse Selection
Create Parameter	CTRL+P
Delete Parameter	CTRL+SHIFT+P
Open Include File	CTRL+G
Create Include File	CTRL+E
Extract Include File	CTRL+SHIFT+E
Add Comment	CTRL+K
Comment Out	CTRL+SHIFT+C
Uncomment	CTRL+SHIFT+U

8.2 处理大文件

当处理较大文件时，CMM文件查看不方便，可以尝试着将网格属性（例如孔隙度和I方向渗透率等）保存为Include文件，这可以通过Builder实现。更多信息参考Builder User Guide中“Organizing Include Files” 部分 。与这些网格属性相关的关键字信息大小可能占了整个文件的90%左右，所以通过使用Include文件可以大大减少CMM主文件的大小。

8.3 公式编辑器

公式就是在CMOST运行时执行某一特定运算的方程式。公式可以出现在CMOST主文件中。在 CMOST主文件中，公式可以出现在任何地方，通常以<cmost>为起始标志，以</cmost>为结束标志。作为高级选项，CMOST也可以使用Jscript脚本语言来创建公式。（详细内容参考Using JScript Expressions in CMOST ）。

8.3.1 公式组成

通常一个公式由以下几部分组成，包含函数、变量（参数或目标函数名称）、常量以及运算符。

- 常量：将数字或文本直接输入到公式，例如0.001或 “OPEN”。
- 函数：函数POWER(30, 0.25)表示30的0.25次幂。
- 变量：为CMOST创建的模拟任务选择参数蒸汽压力值。
- 运算符：＋(加号)运算符表示加法，*（星号）运算符表示乘法。.

8.3.2 公式常量

常量是一个值，不用计算。例如，数字210和文本"OPEN"都是常量，文本必须使用双引号括起来。从某个表达式中得到的表达式或数值不是常量。

8.3.3 公式函数

函数是事先定义的公式，对指定的数值进行计算。函数可用于执行简单或复杂计算。
函数结构:

POWER(SteamPress * 0.001, 0.2388057)

- 结构：函数的构成，开始是函数名称，然后是左括号，被命令分割的参数，最后是右括号。
- 函数名称：参考List of Built-in Functions in CMOST。

- 参数：可以是常数、参数名称、公式或其他函数。定义的参数必须产生有效数值。对于LOOKUP函数来说，常数或公式数组都可以作为参数。

在某些方案中，可能需要某个函数作为其他函数的参数，在这种情况下，用做参数的函数是个嵌套函数。当嵌套函数用做参数时，通过它得到的函数值的类型必须与参数值类型一致。

8.3.4 公式变量

对于**Master Dataset and Parameters**界面中的公式，参数的名称可用于变量名。对于**Objective Functions**界面中的公式，目标函数项的名称可以用于变量名称。参数名称可识别在CMOST Study文件中定义的参数。对于CMOST创建的某一任务，每个参数选择使用某一数值，这些数值被CMOST用来参与公式计算。目标函数中的目标函数项名称可以识别目标函数项。对于已完成运算的CMOST任务，CMOST将从SR2文件中提取结果，用于公式中的运算。

例如，下面的公式包含参数的名称-蒸汽压力（SteamPress），对于CMOST创建的某一特殊模拟任务，参数蒸汽压力的数值为2500。通过下面的公式计算后，其结果为223.78。

179.7989 * POWER(SteamPress * 0.001, 0.2388057)

另外一个例子，下面的例子包含两个目标函数项：CumOil和CSOR。CumOil定义的是在2008-12-01时的累产油，CSOR定义的是在2008-12-01时的累积气油比。对某一已完成的运算方案，CumOil为6.528e5 m3 ，CSOR 为3.15。使用下面的公式计算后的结果为0.207。

1e-6 * CumOil/CSOR

8.3.5 公式运算符

公式中的操作条件定义了执行某一计算的操作类型。CMOST中包含两种类型的操作条件：四则运算和比较运算。

四则运算符：为了执行基础的数学运算，例如加、减、乘或除等使用下面的四则运算符：

算术运算符	意义（例子）
+ （加号）	加法（3+3）
– （减号）	减法(3–1) (–1)
否定运算符	
* （乘号）	乘法（3*3）
/ （除号）	除法 (3/3)

比较运算符：可以运用下面的运算符对两个数值进行比较。当两个数值应用运算符进行比较后，其结果是个逻辑值- TRUE 或 FALSE。

比较运算符	意义（例子）
== （双等于号）	等于 (A1==B1)
> （大于号）	大于(A1>B1)
< （小于号）	小于(A1<B1)
>= （大于等于号）	大于等于 (A1>=B1)
<= （小于等于号）	小于等于 (A1<=B1)
!= (不等于号)	不等于 (A1!=B1)

8.3.6 公式计算优先级

公式可以按照特定的顺序进行计算。CMOST需要根据公式运算符等级从左到右进行公式运算 。

运算符优先：如果某个公式包含若干个运算符，CMOST 按照下表中的运算符等级进行运算。如果某个公式包含同样等级的运算符，例如，如果一个公式包含一个乘号和一个除号，CMOST进行公式计算时，从左至右依次运算。

运算符	描述
–	取反数（像-1）
* 和 /	乘和除
+ 和 –	加和减
== < > <= >= !=	比较运算符

使用圆括号：为了修改运算的顺序，在公式中可以添加括号，对于括号内的部分先进行运算。例如，下面例子中公式计算的结果为11，因为CMOST在运算加法之前，先进行了乘法运算。公式中2*3，然后再加5：

=5+2*3

相反，如果使用括号改变运算顺序，CMOST先执行5+2，然后再乘以3，结果为21：

=(5+2)*3

8.3.7 CMOST内置函数列表

注意：这里罗列的仅仅是CMOST内置的函数。作为高级功能，CMOST支持脚本语言编写公式，因此所有脚本语言支持的函数在CMOST中都支持。更多信息参考 Using JScript Expressions in CMOST。

8.3.7.1 IF

如果判断某一条件是TRUE，则返回一个值，如果是FALSE，则返回另外一个值。使用IF函数来进行条件检测，即可应用于数值，也可应用于公式。

语法：

```
IF(logical_test, value_if_true, value_if_false)
```

其中logical_test可以是任意数值或表达式，用于判定TRUE或FALSE。例如，WellLen>=800是个逻辑表达式；如果WellLen大于或等于800，表达式则为TRUE，否则为FALSE。 该论点可用于任何比较运算符。

如果logical_test是TRUE，则得到的值就是value_if_true。Value_if_true可以是个常数，也可以是个公式。

如果logical_test是FALSE，则得到的值就是value_if_false。Value_if_false可以是个常数，也可以是个公式。

例 1

根据CMOST主文件中水平井的水平段长度，使用下面的IF函数来打开/关闭井的射孔。

```
PERF  GEO  'Horizontal'
**$ UBA    ff   Status Connection
9 3 12 1. OPEN FLOW-TO  'SURFACE'
9 4 12 1. OPEN FLOW-TO  1
9 5 12 1. OPEN FLOW-TO  2
9 6 12 1. OPEN FLOW-TO  3
9 7 12 1. <cmost>IF(WL>=500,"OPEN","CLOSED")</cmost> FLOW-TO 4 9 8 12 1.
<cmost>IF(WL>=600,"OPEN","CLOSED")</cmost> FLOW-TO 5 9 9 12 1.
<cmost>IF(WL>=700,"OPEN","CLOSED")</cmost> FLOW-TO 6
```

对于CMOST创建的某个任务，其WL=600，数据文件中水平井'Horizontal' 射孔如下所示：

```
PERF  GEO  'Horizontal'
**$ UBA    ff   Status Connection  9 3 12 1.
OPEN FLOW-TO  'SURFACE' 9 4 12 1. OPEN
FLOW-TO  1
9 5 12 1. OPEN FLOW-TO  2
9 6 12 1. OPEN FLOW-TO  3
9 7 12 1. OPEN FLOW-TO 4
9 8 12 1. OPEN FLOW-TO 5
9 9 12 1. CLOSED FLOW-TO 6
```

8.3.7.2 LOOKUP

对于指定的数值，LOOKUP函数搜索数组常数，在数组公式相同的位置返回结果数值。使用LOOKUP函数定义参数之间的关系。

语法

LOOKUP(lookup_value, array_constants, array_formulas)

lookup_value是LOOKUP函数查找的某一数值。它必须是一个参数名称。array_constants是想与lookup_value作比较的一组常数。array_formulas是一组公式，从LOOKUP函数中返回值。公式数目应该等于array_constants 常数个数。

附注:

array_constants 格式

- 常数用花括号{ }括起来，常数之间用（，）隔开;
- 常数数组包括数字和文本。文本区分大小写。
- 常数数组中的数字可以是整数、小数或科学计数形式。
- 文本必须用双引号括起来，例如"CLOSED"。
- 常数数组不能包含公式和参数名称。

array_formulas 格式

- 除了array_formulas 可以包含常数和变量名称以外，其格式和array_constants一致。

重要提示

- LOOKUP函数找到与lookup_value同样完全匹配的数值。如果LOOKUP没有找到lookup_value，其结果是未定义。

例 2

使用LOOKUP函数，根据CMOST主文件中气油比来设置溶解气的摩尔分数。

注意: <cmost>和</cmost> 必须在同行。

```
**$ Property: Oil Mole Fraction(Soln_Gas)
MFRAC_OIL 'Soln_Gas' CON
<cmost>LOOKUP(ogor, {7,5,4,3}, {0.10,0.08,0.06,0.04})</cmost>
```

对CMOST创建的模拟任务ogor=4，则对应溶解气的摩尔分数为0.06。

```
**$ Property: Oil Mole Fraction(Soln_Gas) MFRAC_OIL
'Soln_Gas'  CON   0.06
```

8.3.7.3 ABS

返回绝对值。绝对值是数值前面没有负号。语法：

ABS(number)

number想要取绝对值的实数。

例如：

ABS(-12.5) returns 12.5

8.3.7.4 COS

返回给定角度的余弦值。

语法：

COS(number)

number 想要取余弦值的弧度角度。

注意：

如果角度是度数，需要将其乘3.14159/180来转换为弧度。

例如：

COS(1.047) returns 0.500171

8.3.7.5 EXP

返回e的指数函数数值。常数e等于2.71828182845904，这是自然对数的指数。语法：

EXP(number)

Number是e的指数。

注意：

为了计算e的指数，使用POWER函数。

例如：

EXP(2) returns 7.389056

8.3.7.6 LN

返回一个数的自然对数。自然对数是以e（2.71828182845904）为底。

语法:

LN(number)

number是正实数。

注意:

LN 与EXP是相反

的。例如:

LN(2.7182818) returns 1

8.3.7.7 LOG10

返回以10为底的对数 。

语法:

LOG10(number)

number 是正实数。例

如:

LOG10(100) returns 2

8.3.7.8 MAX

返回一系列数中最大的数。

语法:

MAX(number1,number2,...)

number1, number2, ... 是想要从这些数中找到最大的数。

注意:

- 指定的参数可以是数值、变量名以及公式。
- 参数必须包含两个数值。

例如:

MAX(var1, 2.54) returns 2.54 if var1=1.06

8.3.7.9 MIN

返回一系列数中最小的值。

语法:

MIN(number1,number2,...)

number1, number2, ... 是想要从这些数中找到最小的数。

注意：

- 指定的参数可以是数值、变量名以及公式。
- 参数必须包括两个数值。

例如：

对于一对 SAGD井，想要根据注汽井的注汽速度来修改生产井的产液量，可以在主文件中使用MIN函数来实现：

```
INJECTOR MOBWEIGHT EXPLICIT 'WEST-I'
INCOMP  WATER  1. 0. 0.
TINJW  253.2
QUAL  0.9
OPERATE MAX BHP 4200. CONT REPEAT
OPERATE MAX STW <cmost>Steam</cmost> CONT REPEAT
PRODUCER 'WEST-P'
OPERATE MIN BHP 2000 CONT REPEAT
OPERATE MAX STL <cmost>MIN(Steam*1.6,1000)</cmost> CONT REPEAT
```

对于一个任务Steam=500，生产井'WEST-P'的操作条件将是：

```
PRODUCER 'WEST-P'
OPERATE MIN BHP 2000 CONT REPEAT
OPERATE MAX STL 800 CONT REPEAT
```

对于一个任务Seam=700，生产井'WEST-P'的操作条件将是：

```
PRODUCER 'WEST-P'
OPERATE MIN BHP 2000 CONT REPEAT
OPERATE MAX STL 1000 CONT REPEAT
```

8.3.7.10 POWER

返回一个数值的幂

语法：

POWER(number, power) number 是

基础数字，可以是任意数。

power 是幂指数

例如：

POWER(5,2) returns 25

8.3.7.11 SIN

返回给定角度的正弦值。

语法：

SIN(number)

number是转换为正弦值的弧度。

注意：

如果角度是度数，需要将其乘3.14159/180来转换为弧度。

例如：

SIN(30*3.14159/180) return 0.5 ***8.3.7.12***

SQRT

返回正平方根。

语法：

SQRT(number)

number 是想要开平方根的数。

注意：

数字必须为非负。

例如：

SQRT(2.0) returns 1.4142 ***8.3.7.13***

TAN

返回给定角度的正切值。

语法：

TAN(number)

number是转换为正切值的弧度。

注意：

如果角度是度数，需要将其乘3.14159/180来转换为弧度。

例如：

TAN(0.785) return 0.99920

8.4 在CMOST使用Jscript脚本语言

Jscript代码的执行是个强大的功能，允许在CMOST中扩展代码，但是不能编译到应用程序中。通过自定义的Jscript代码，用户可以定制和扩展CMOST。例如，用户可自己编写的代码来计算目标函数。然后CMOST可以读取编码，快速的执行相应编码应用。

执行动态编码，CMOST允许用户对CMOST流程有更先进的控制，包括产生模拟任务，后处理结果文件，目标函数计算和方案参数优化。在CMOST，Jscript脚本语言代码可以出现在**Parameters**、**User-Defined Time Series**、**Objective Functions**、**User-Defined Global Objective Function Candidates**和**Constraints**页面：

- 定义参数之间的复杂关系
- 自定义目标函数
- 定义复杂的约束条件和罚函数

JScript是ECMAScript脚本语言规范微软方言，JavaScript是另一种方言。Jscript是作为一个脚本引擎实现的。它允许嵌入ActiveX对象重用许多微软Windows应用程序的功能。例如，用户可以很容易的通过Jscript脚本编码链接CMOST到Excel电子表格计算目标函数。

完整的JScript 语言参考文献，请参考开发者网络网站：
http://msdn.microsoft.com/en-us/library/x85xxsf4.aspx

下面网站也提供了一些关于Jscript主题的参考信息：
http://ns7.webmasters.com/caspdoc/html/jscript_language_reference.htm

下面的例子是在CMOST安装目录里面，用于帮助用户理解CMOST中使用的JScript脚本语言：

- *RelPermMatch_UseInclude.cmp*: Use JScript code to create a relative permeability table.
- *SAGD_2D_DynaGrid_Optimization.cmp*: Use JScript code to define custom objective functions.
- *SAGD_NT.cmp*: Use JScript code to read simulation output files.
- *Punq_infill.cmp*: Use JScript code to define input parameters.

以下部分提供了更多关于CMOST中使用JScript脚本语言的详细内容。

8.4.1 从CMOST转换数据到用户JScript 编码.

由于用户编写的Jscript脚本编码由CMOST动态执行的，数据是以常用变量的形式在CMOST和用户编码之间传送，而这种变量在CMOST和编码之间是“可见的”。下面的表格总结了常用变量：

JScript编码位置	常用变量
参数	• 除了当前用JScript 编码定义外的其他所有参数 • 所有模拟方案的输入及输出文件
高级目标函数	• 除了当前用Jscript编码定义外的其他所有参数 • 所有模拟方案的输入及输出文件
用户定义的总目标函数候选值	• 所有目标函数项定义当前的目标函数 • 所有模拟方案的输入及输出文件 • 所有参数
硬性约束条件	• 所有参数 • 所有模拟方案的输入输出文件

JScript编码位置	常用变量
软性约束条件	• 所有参数 • 所有目标函数 • 所有模拟方案的输入及输出文件
软性约束条件惩罚	• 所有参数 • 所有目标函数 • A所有模拟方案的输入及输出文件

为了在JScript 脚本编码中使用常用的变量，需直接使用相应的名称。对于定义的新变量，不能使用相同变量名。例如，如果定义了某个参数，其名称为InjectionPress，就可以在Jscript编码中直接使用该参数名称，然而，再定义参数名称时，不能再使用InjectionPress参数名称。

另外，不能随意修改CMOST已经声明的变量名称，因为CMOST在JScript脚本编码执行之前将用实际值替代变量名。公式中的变量名是事先预订的，更多信息参考Formula Editor 。

8.4.2 即时存取模拟方案输入及输出文件

为了即时存取输入和输出模拟方案文件，使用CMOST定义的常用变量:

变量名称	值
Project目录	Project目录全路径
Study目录	Study 目录全路径
实验名称	模拟任务名称 模拟任务数据文件完整路径
实验方案数据文件路径	模拟任务Log文件完整路径
实验方案Log文件路径	模拟任务Out文件完整路径
实验方案Out文件路径	模拟任务Irf文件完整路径
实验方案Irf文件路径	模拟任务Mrf文件完整路径
实验方案Mrf文件路径	

例如，下面的Jscript脚本编码用来打开实验方案的log文件并读取最后时间步的物质平衡误差。

```
        var fso=new ActiveXObject("Scripting.FileSystemObject"); var
        ts=fso.OpenTextFile(ExperimentLogFilePathh);
        var line: String;
        var matErrorStr: String
        var matError: double
        while(!ts.AtEndOfStream)
        {
             line=ts.ReadLine();
    if(line.length>=90)    {

            matErrorStr=line.substring(84,89)
try
    {
                    matError=double.Parse(matErrorStr);
    }
    catch(e)
    {
                    //Do nothing
    }
      }
        }
        matError;
```

8.4.3 将数据从JScript 脚本编码传递到 CMOST

当用户的Jscript脚本编码在CMOST内部执行时，最后的结果将从脚本编码传递到CMOST，编码的最后一行用于计算结果。例如，下面编码中最后的结果为10：

```
        var a, b, c, d;
        a=4;
        b=3;
        c=2;
        d=1;
        a+b+c+d;
```

8.4.4 在数据文件中开启新的一行

输入模拟器系统中的关键字必须重启新的一行。为了在脚本编码中开启新的一行，需要使用 “\r\n” 字符串。例如：

```
        var kwLines: String;
        kwLines ="\r\n"+"** Group 1 Heaters ";
        kwLines+="\r\n"+"HEATR IJK "+UBAI+" 1 "+UBAK+” ”+QH; kwLines+="\r
        \n"+"UHTR IJK "+UBAI+" 1 "+UBAK+" 16351"; kwLines+="\r\n"+"TMPSET
        IJK "+UBAI+" 1 "+UBAK+" 1129"; kwLines+="\r\n"+"AUTOHEATER ON
        "+UBAI+" 1 "+UBAK;
```

8.5 在CMOST中使用Python语言

CMOST支持第三方工具IronPython-Python脚本语言 (http://ironpython.net/)。CMOST支持Python 2.x语言，但必须先安装Python。可以从 https://www.python.org/downloads/免费下载。

注意：为了包含用户常用模块，在公式开始前添加以下语句。

```
import sys
    sys.path.append(r"c:\userModulePath")  import
usermodule
```

9 CMOST 交互数据可视化工具 (CMOST Interactive Data Visualization Tool)

9.1 综述

通常，很难从原始数据中发现规律。Interactive Data Visualization Tool可以帮助你从可视化实验方案结果中发现规律。该工具不仅提供了数据的概述，也提供了更多关于数据之间的相互关系。尤其该工具可以满足不同用户的需求，尤其在：

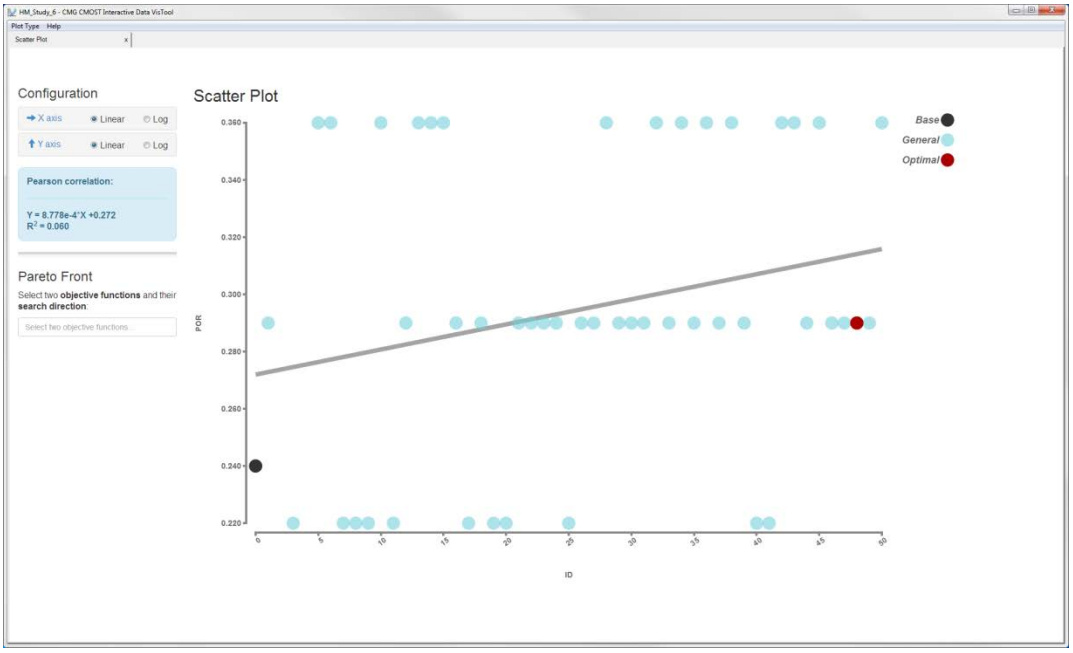
- 评价实验方案设计质量
- 确定需要添加额外实验方案的区域
- 确定数据关系式
- 依据参数和目标函数值筛选实验方案

9.2 使用交互数据可视化工具

9.2.1 打开交互数据可视化工具

为了打开Interactive Data Visualization Tool，在Experiments Table右侧，点击 **Launch Interactive** Data Visualization Tool  按钮。**Interactive Data Visualization Tool** 将打开一张散点图，x-轴是实验ID，y-轴是表格中的第一个参数或目标函数。

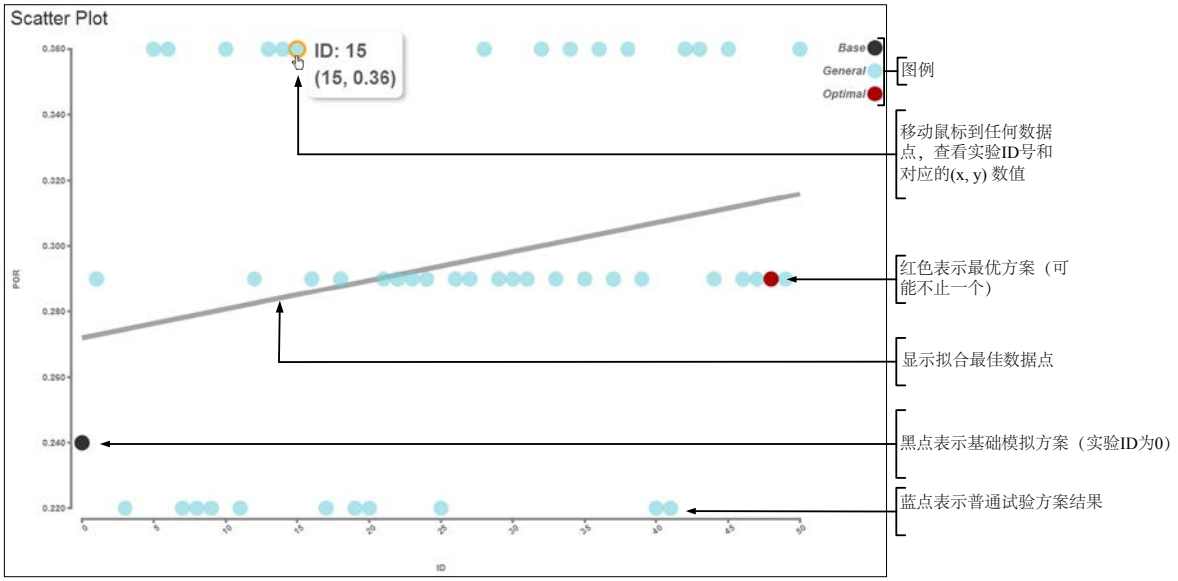
在下例图中，y-轴表示的是参数POR，因为它是**Experiments Table**中的第一个参数：



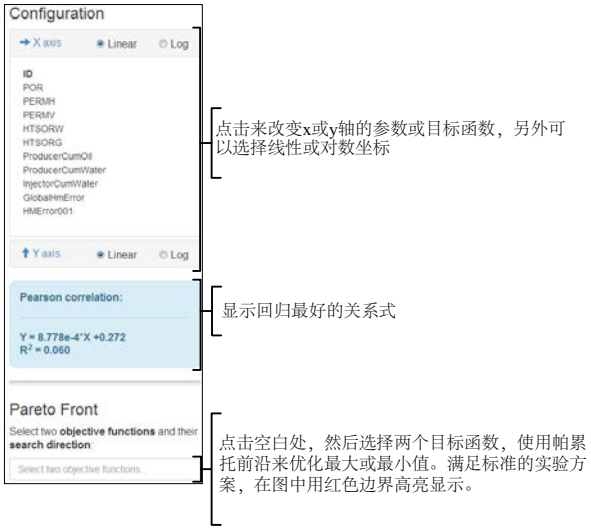
9.2.2 散点图

9.2.2.1 关于散点图

散点图组成，如下所示：

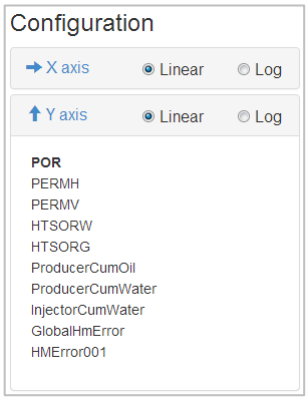


散点图左边**Configuration**和**Pareto Front**部分，如下图所示：

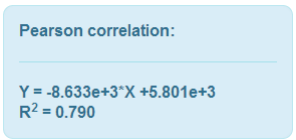


9.2.2.2 散点图配置

在散点图左边配置部分，可以改变x或y轴的参数或目标函数，另外，还可以选择使用线性或对数坐标，例如：



一旦选择参数后，在配置部分中显示最佳匹配的皮尔逊相关系数，如下所示：



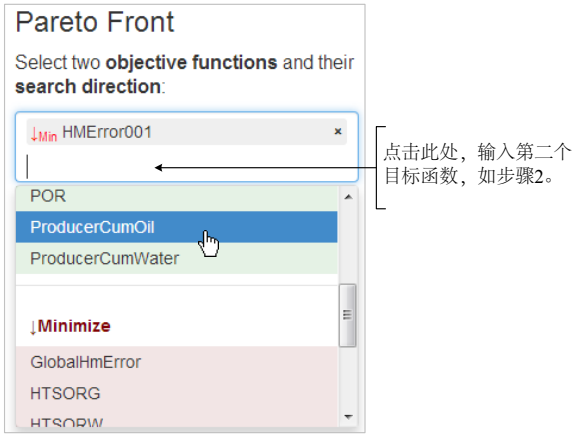
最佳匹配的线图也在散点图上用灰色的线叠加。用鼠标在某个点上移动时，可以查看相应实验的ID和(x, y)数值，如下图所示：

9.2.2.3 帕累托分析

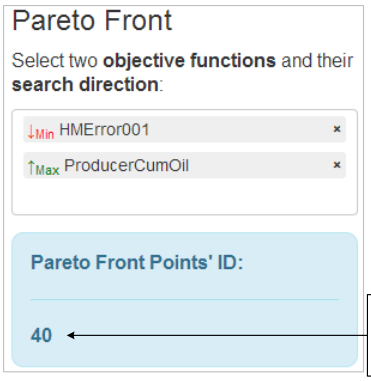
通过Scatter Plot图，基于实验方案的后处理分析结果，CMOST提供了执行帕累托分析的一种手段。无论是否选择并配置帕累托优化引擎，如下：

1. 点击**Pareto Front**部分空白处

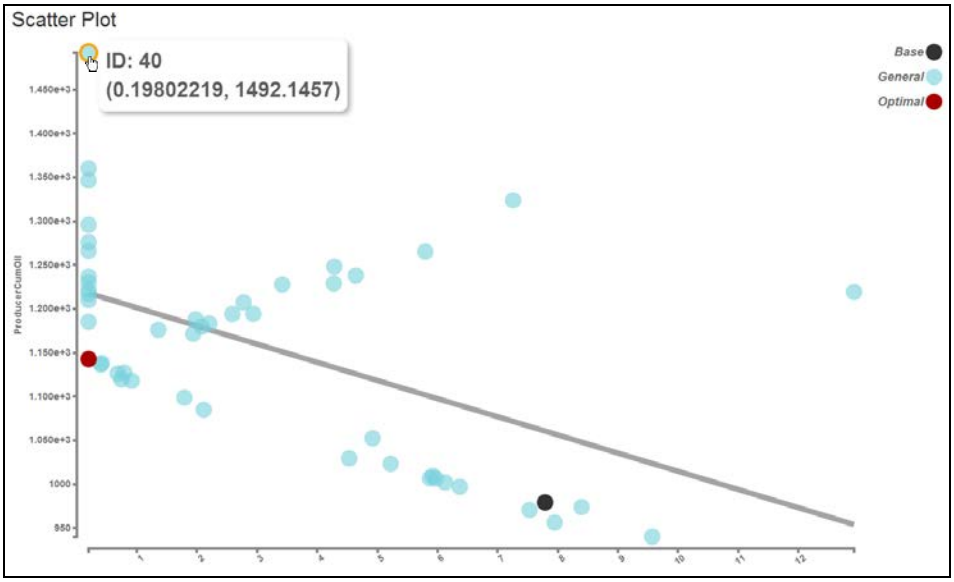
2. 选择第一个目标函数
   - 在Maximize和Minimize列表中选择目标函数，其中，阴影中的青绿色为最大值，粉红色为最小值。
   - 在空白处直接输入目标函数名称。随着输入，CMOST会自动给定建议选项，可以根据需要选择。
3. 为了输入第二个目标函数，在第一个目标函数下面点击空白处，如图所示，如步骤2：



4. 在该例子中，我们配置了**Interactive Data Visualization Tool**来使用帕累托前沿方法来查找最小的目标函数*HMError1*和最大的目标函数*ProducerCumOil*。在**Pareto Front**部分显示满足该标准的实验ID，上面例子最优的实验ID为40：



5. 在图的左上方该实验用红色加粗字体突出显示（将鼠标移动到某个实验时，呈现橘黄色）：

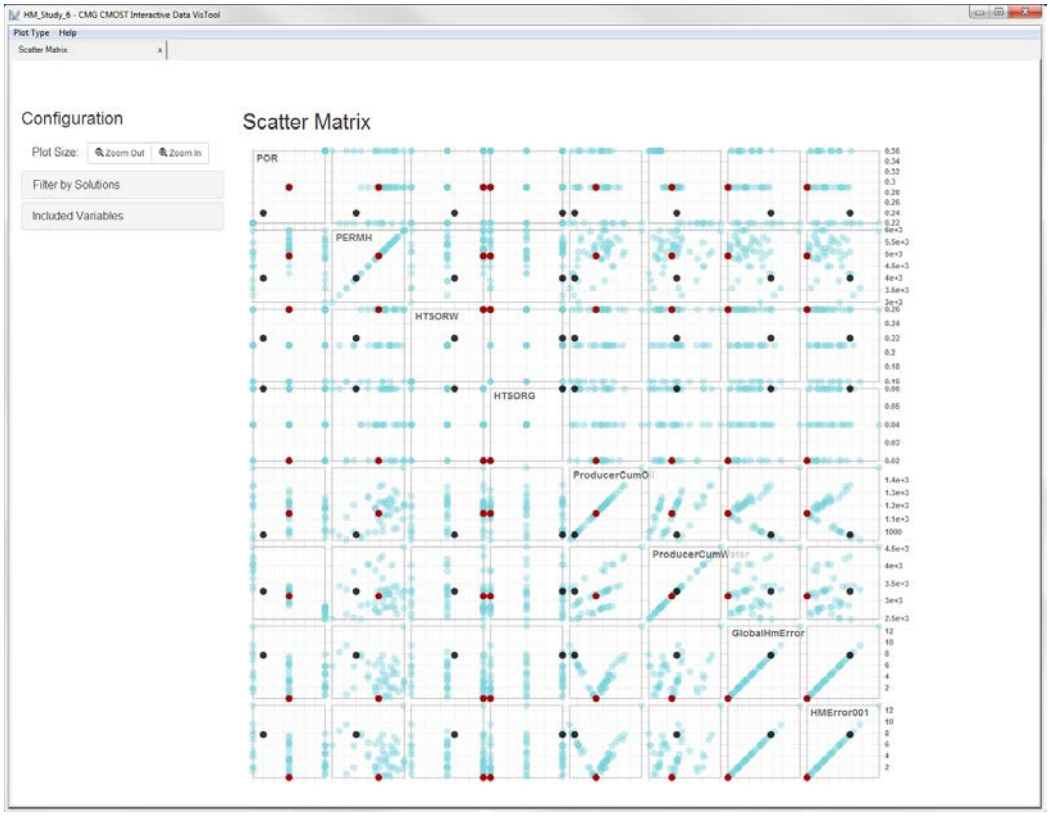


9.2.3 散射矩阵图

9.2.3.1 关于散射矩阵图

散射矩阵图用于研究两参数变量之间的成对关系，分类归并数据到组并确定异常数据值。

可以通过点击**Plot Type | Scatter Matrix**查看散射矩阵：



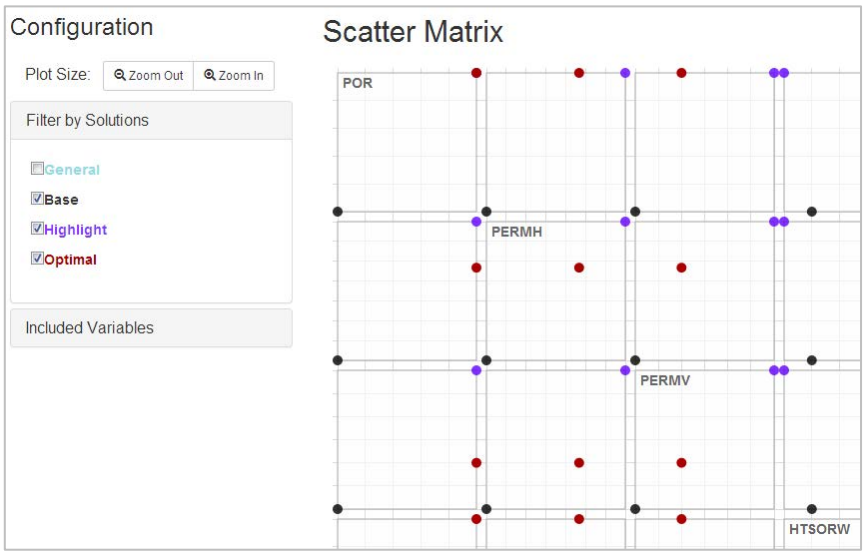
如上面例子所示，所有初始的散点图都在散射矩阵中显示；然而，如下面，可以依据方案类型、参数及目标函数来筛选散点图。

9.2.3.2 利用散射矩阵图

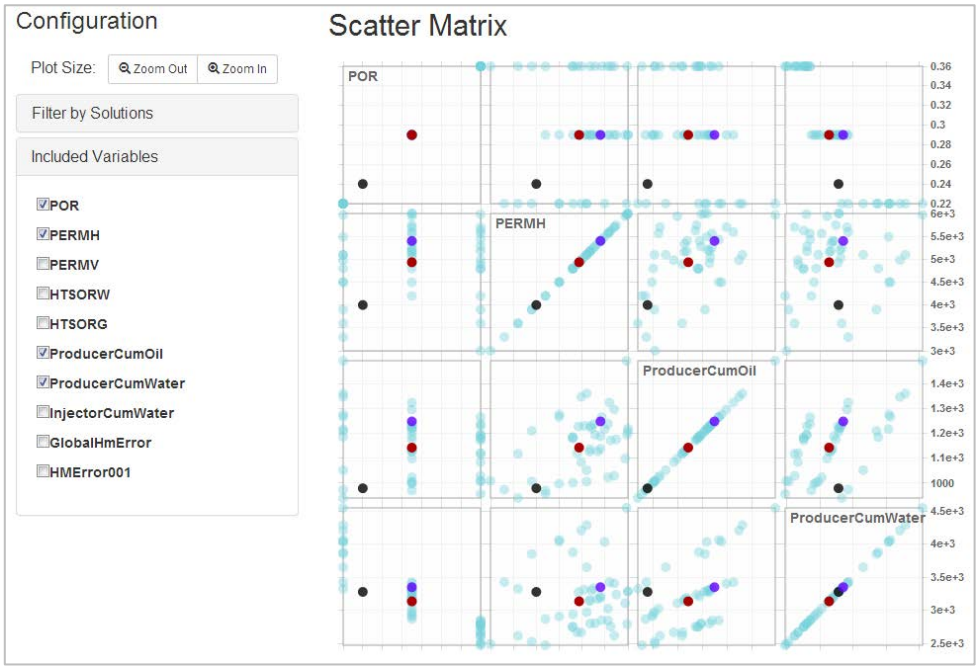
在**Configuration**部分，可以：

- 点击Zoom In 来放大查看细节，或者Zoom Out 缩小查看整体轮廓。

- 点击**Filter by Solutions** 然后在散射矩阵图中选择想要显示的解决方案类型。在下面的例子中，我们选择*Base, Highlight and Optimal*：



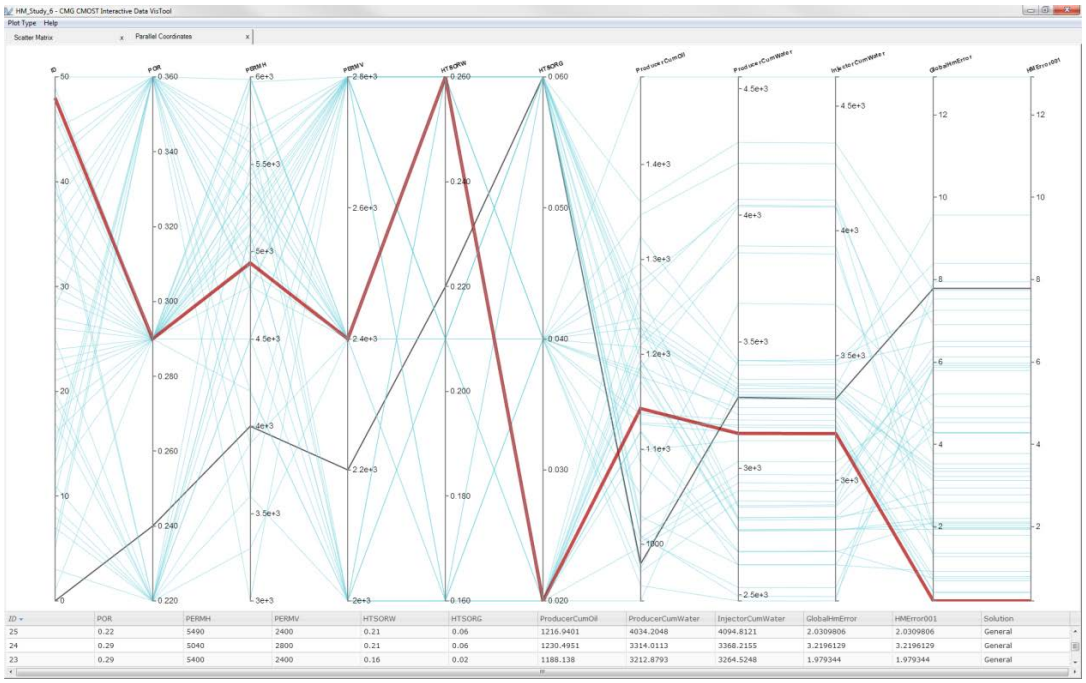
- 点击**Included Variables** 然后选择想要在散射矩阵图中显示的变量。在下面的例子中，我们选择了*POR*、*PERMH*、*ProducerCumOil*和*ProducerCumWater*：



9.2.4 平行坐标图

9.2.4.1 关于平行坐标图

平行坐标图提供了一种通用手段，直观的分析多元数据；尤其在筛选、比较、分组数据，识别模型以及评价相关性强度等方面。在CMOST 交互数据可视化工具（**Interactive Data Visualization Tool**）下，通过平行坐标图（**Parallel Coordinates**），可以查看这些图，如下例所示：



ID	POR	PERMH	PERMV	HTSORW	HTSORG	ProducerCumOil	ProducerCumWater	InjectorCumWater	GlobalHmError	HMError001	Solution
25	0.22	5490	2400	0.21	0.06	1216.9401	4034.2048	4094.8121	2.0309806	2.0309806	General
24	0.29	5040	2800	0.21	0.06	1230.4951	3314.0113	3368.2155	3.2196129	3.2196129	General
23	0.29	5400	2400	0.16	0.02	1188.138	3212.8793	3264.5248	1.979344	1.979344	General

平行坐标图（ **Parallel Coordinates**）是由窗口底部的一些实验表格、参数值、目标函数以及窗口顶部的平行坐标组成。平行坐标图是由等间距坐标轴构造而成，它表示归一化参数和目标函数值范围。每个实验方案在整个坐标轴中跟踪一个轨迹，然后在实验的参数或目标函数之间交叉。该轴在各自的参数或目标函数的最小值和最大值之间进行归一化。

9.2.4.2 使用平行坐标图

为了基础方案和最优方案的优化结果：

缺省的，一般方案用浅蓝色显示，基础方案用黑色显示，特别强调的方案用紫色显示，最优方案用红色显示。

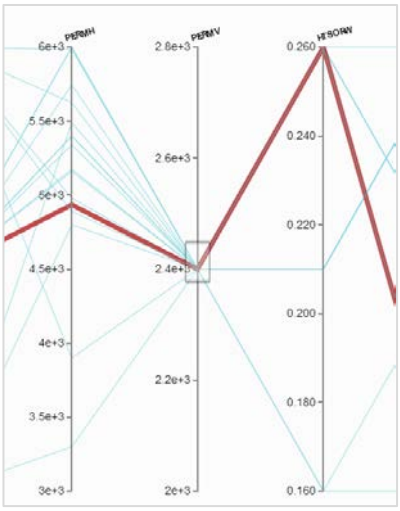
为了强调一个试验方案：

如果将鼠标移至试验表格中的某一实验方案，对应的该行由黑色变为深蓝色，平行坐标图中的轨迹以深蓝色高亮突出显示。其他实验方案轨迹将褪色。

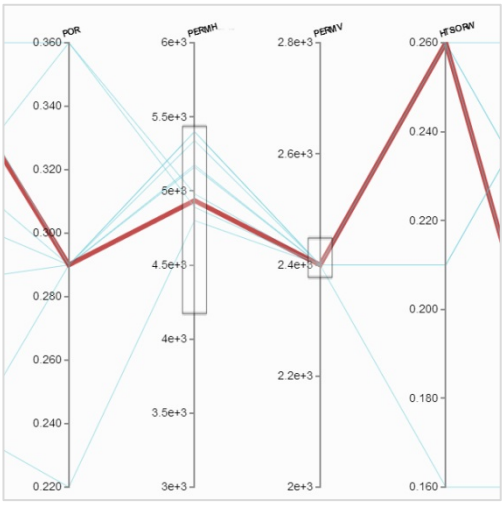
为了筛选实验方案：

可以在一个或多个坐标轴进行筛选，如下：

1. 在坐标轴移动鼠标直到出现Precision Select。
2. 点击并拖动可定义的筛选器范围。一旦确定筛选器，可以在顶部或底部任何时间通过拖拽进行调整。只有通过筛选器的轨迹才会显示，如下所示：



如果定义了第二个筛选器，仅显示通过两个筛选器的轨迹，诸如此类：



同样，只有通过筛选器的轨迹，才能显示在表格中。

为了删除筛选器，用鼠标拖住筛选器选中范围的一端至另一端，然后释放鼠标就可以完成。当筛选范围的顶端和末端重合时，影响就会被删除。

坐标轴重新排序：

点击坐标轴顶部或沿着坐标轴长度的任一值，直到出现 **Move**鼠标，然后将坐标轴向左或向右调至新位置。**Interactive Data Visualization Tool**均匀间距调整坐标轴位置。

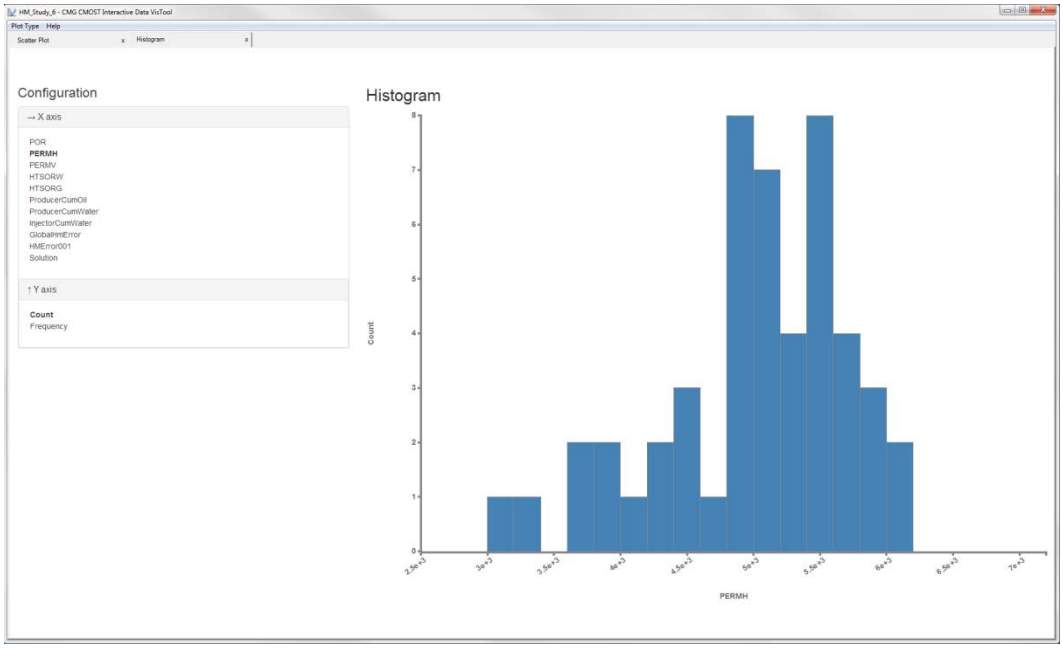
实验表格中的数字排序：

点击实验表格表头，按升序对相关参数进行排序，再次点击表头，按降序排序。

9.2.5 直方图

9.2.5.1 关于直方图

点击**Plot Type | Histogram**来打开直方图（**Histogram**），通过直方图可以查看某个参数或目标函数使用的频率，如下所示：



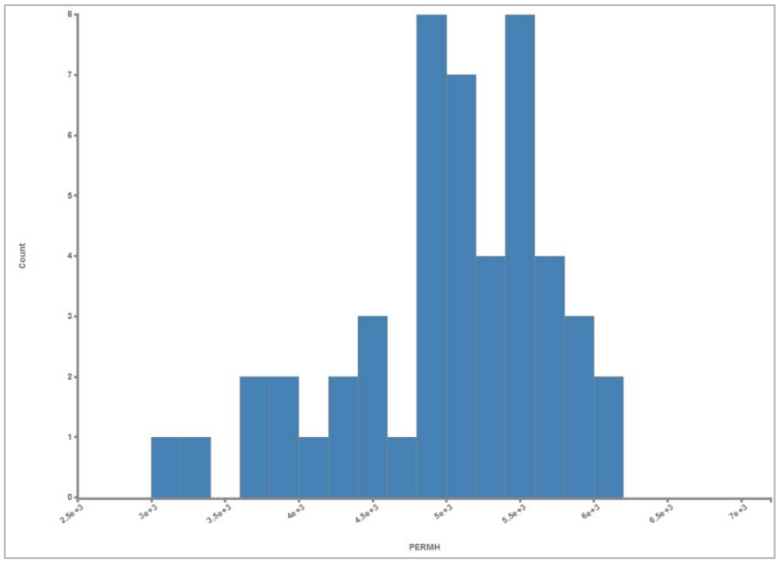
上面的例子中显示了参数PERMI在试验表格中使用的频率，它是一个连续的实数。

注意：如果直方图显示连续数值，每个“条带”代表一个间隔。如果直方图是离散值，每个“条”表示某个指定值。对于这两种情况，“条带”的高度表示个数或百分数。

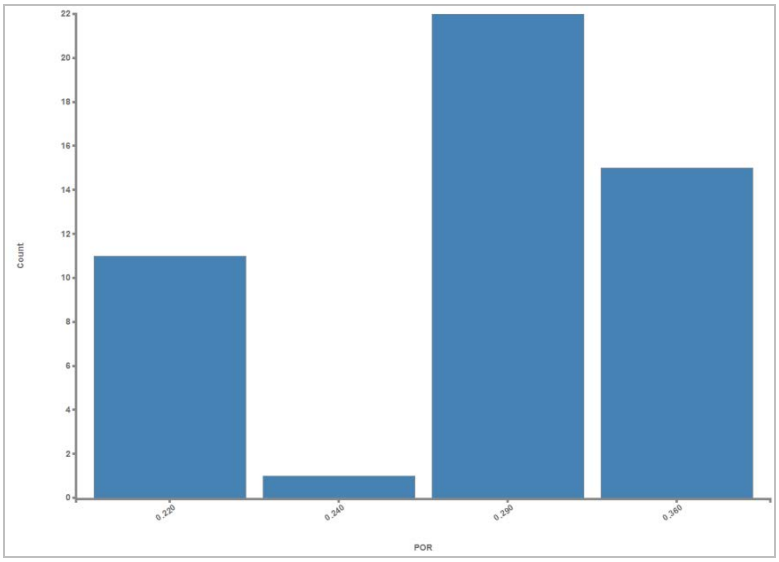
9.2.5.2 使用直方图

1. 在**Histogram** 左边的**Configuration**部分，选择想要在x轴显示的参数或目标函数以及在y轴想要显示的个数或频率。直方图根据自己的选项及时更新。

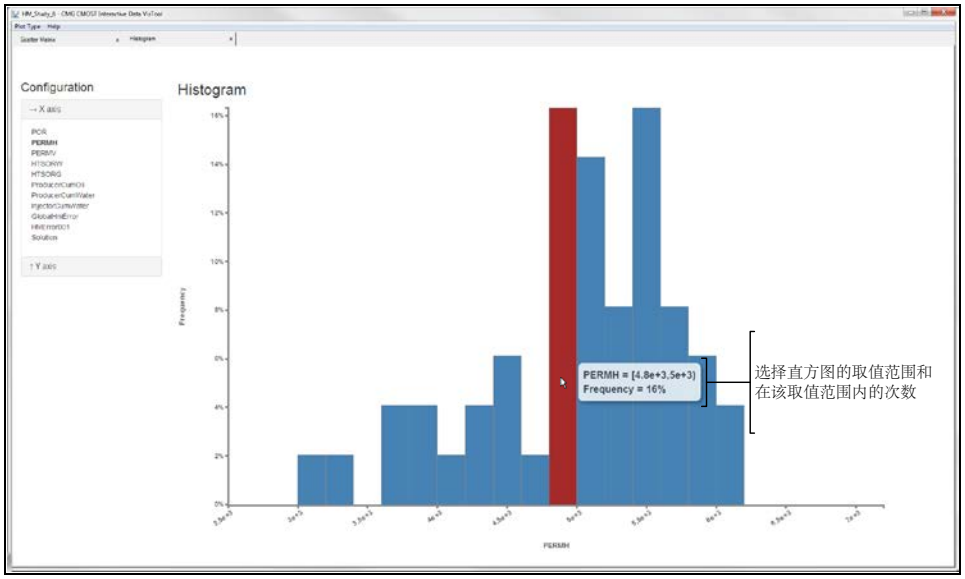
2. 如果参数或目标函数是连续变量，每个“条带”将表示某个取值范围，如下：



如果是连续参数，则“条带”之间没有间隔。如果参数或目标函数是离散值，则离散值在直方图中间，如下：



3. 如果将鼠标光点移至直方图上，则参数或目标函数取值范围（连续）或值（离散）以及个数或频率将显示在坐标，如下例所示：



9.2.6 关闭交互数据可视化工具

当关闭交互可视化工具（**Interactive Data Visualization Tool**）时，所有修改的部分都将不会保存。

10 配置Launcher 和 CMOST共同工作
（Configuring Launcher and CMOST to Work Together ）

10.1 简介

CMOST 依赖Launcher或CMG任务服务器（CMG Job Service）来运行任务。因此，在使用CMOST之前，需要恰当的配置Launcher和 CMG任务服务器（CMG Job Service）。CMOST 也必须配置连接到Launcher 或CMG任务服务器（CMG Job Service），以便排队运行。

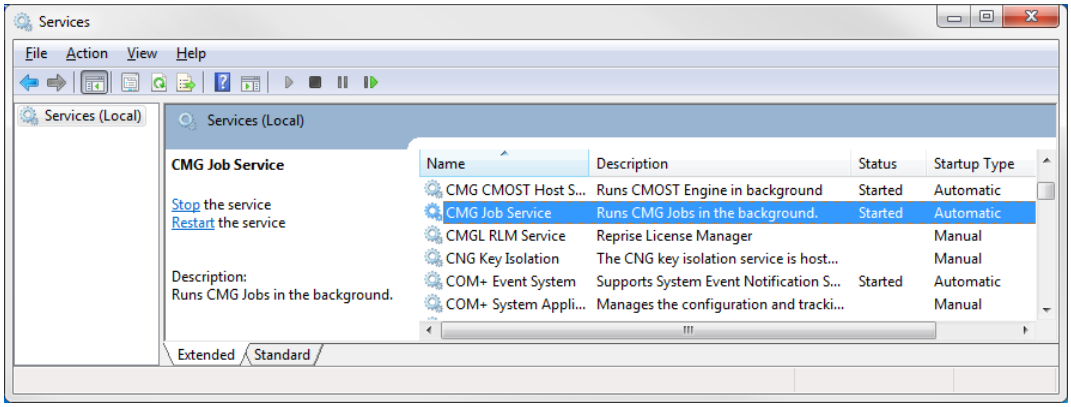
10.2 配置Launcher

10.2.1 Launcher介绍

Launcher 是 Windows图形用户界面应用（GUI）。可以在任意时间利用用户桌面的快捷键来启动安装目录(CMG_HOME\Launcher\xxxx.xx\Win32\EXE\)下面的可执行文件(CMG.exe)。

10.2.2 CMG任务服务器（CMG Job Service）

CMG任务服务器（CMG Job Service）是一个 Windows服务应用。可执行文件（CMG.JobService.exe）安装在目录CMG_HOME\CMGJobService下。服务器（CMG Job Service）是自动创建的，当CMG软件安装时，在Windows中登记。可以通过**Control Panel | Administrative Tools | Services**进入CMG任务服务器（CMG Job Service）。



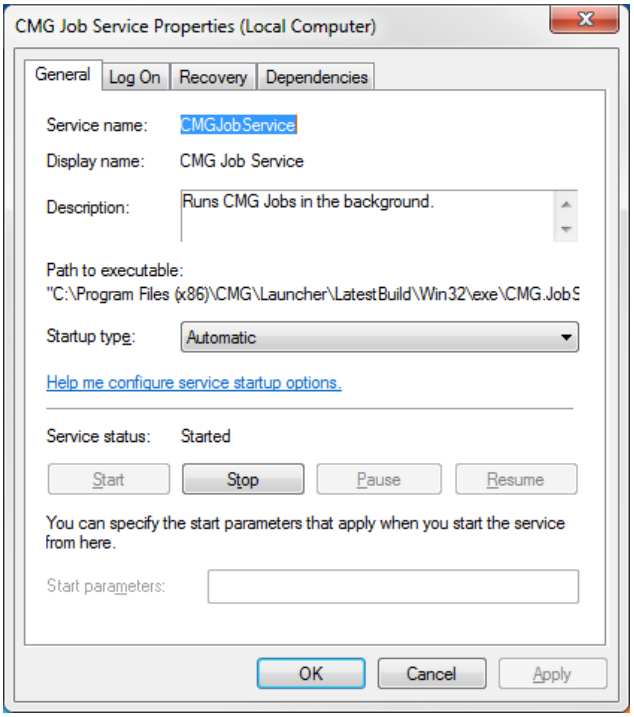
注意：需要管理员权限才能注册、开始或停止Windows服务。

如果基于某些原因导致CMG任务服务器（CMG Job Service）在Windows Services列表中不能正常使用，用户管理员可以通过使用以下的命令：

sc create CMGJobService binPath= "<path to CMG.JobService.exe>"

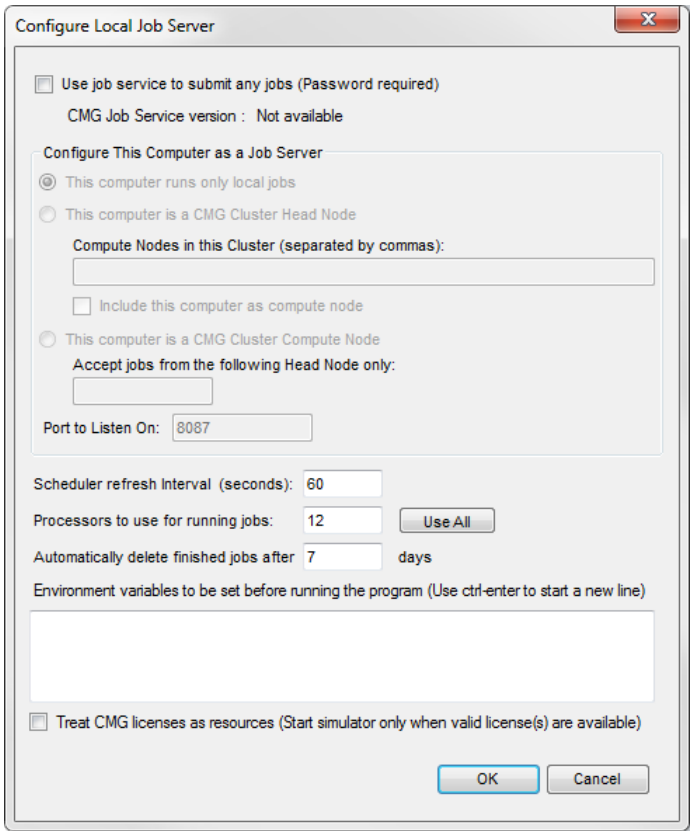
注意在等号和CMG.JobService.exe 路径之间需要空格。如果路径CMG.JobService.exe 包含空格，必须用双引号括起来。

因为CMG任务服务器（CMG Job Service）是Windows Services应用程序，启动和终止服务必须在**Control Panel | Administrative Tools | Services**中执行。如果startup type 设置为Automatic，当计算机重新启动时，CMG任务服务器（CMG Job Service）将自动启动。如果startup type设置为Manual，用户通过点击Start 按钮来手动启动服务器（管理员权力需要手动启动服务器）。



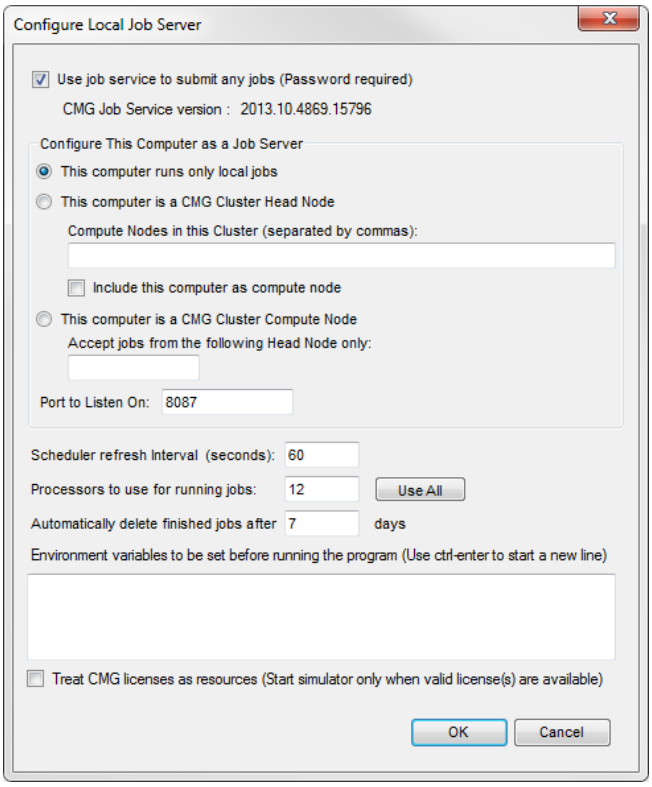
10.2.3 使用Launcher嵌入式工作模式提交作业

如果你的公司或团队使用智能卡技术作为用户凭证，可能会使用Launche嵌入式工作模式来提交工作。在嵌入式模式中，不使用CMG任务服务器（CMG Job Service），该服务器可能会被终止。在嵌入式模式中，Launcher必须总是保持开启，在 **Launcher | Configuration | Configure Local Job Server** 中清除掉“Use job service to submit any jobs”，如下面**Configure Local Job Server**对话框 所示：



10.2.4 使用 CMG Job Service提交作业

在Windows登陆时，如果使用用户名称和密码，可能会使用CMG Job Service来提交作业。在这种模式下，CMG Job Service必须运行。参考CMG Job Service，查看如何启动和停止CMG Job Service。如果作业通过CMOST提交给CMG Job Service，Launcher可以是打开，也可以是关闭的。然而，如果想在Launcher中查看排队的任务，Launcher必须是打开状态，需要在下面的对话框**Launcher | Configuration | Configure Local Job Server** 选择**Use job service to submit any jobs**。



10.2.5 将作业提交至远程计算机

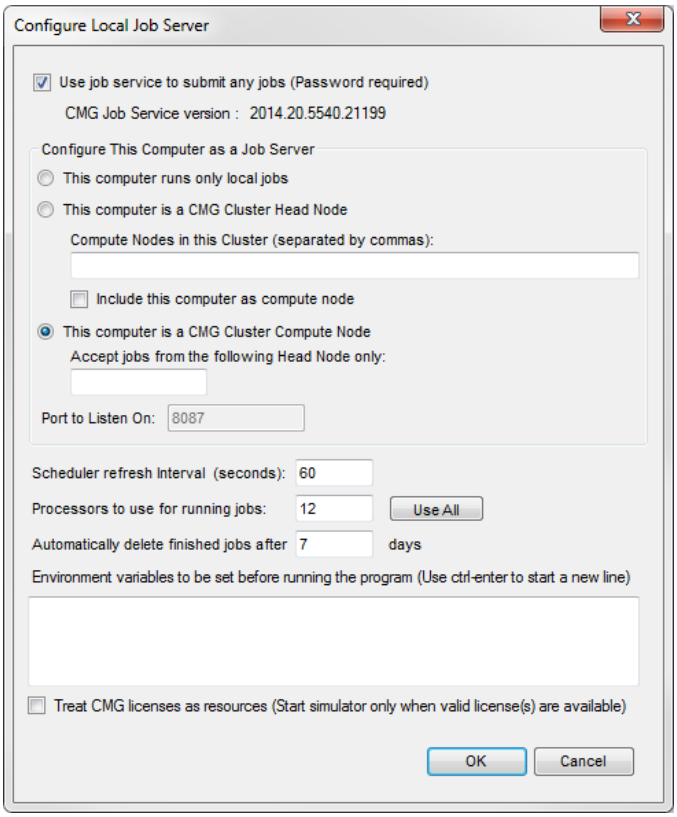
通过远程Scheduler 可以将任务提交至远程计算机运算。Launcher支持以下几种类型的远程Schedulers：

- CMG Scheduler
- IBM Platform LSF
- Microsoft HPC
- Oracle Grid Engine
- Portable Batch System (PBS/TORQUE)

这里我们展示了使用CMG Scheduler 将作业提交至远程计算机需要的配置。对于其他类型的Schedulers，参考*Launcher Users Guide*。

- 在远程计算机上安装 CMG软件。在安装过程中，选择**Use CMG Job Service to submit jobs** 。
- 在远程计算机，切换到**Control Panel | Administrative Tools | Services**打开CMG Job Service。确定startup type设置为Automatic，当计算机重新启动时，服务器自动开启。如果服务器没有启动，点击**Start**按钮来启动服务器（管理员权力需要手动启动服务器）。

- 在远程计算机，打开Launcher，切换到**Configuration | Configure Local Job Server**。确保选择**Use job service to submit any jobs**和**This computer is a CMG Cluster Compute Node**。当设置完成后，可以关闭 Launcher。



- 在用户本地计算机，打开Launcher，切换到**Configuration | Configure Remote Schedulers**。通过提供的远程计算机名称，依据向导添加一个远程CMG Scheduler。
- 在用户本地计算机，通过**Configuration | Password | Add/Modify**添加密码到Launcher（如果Launcher已经设置了密码，可以忽略该步）。

现在，用户可以在本地计算机的Launcher中提交作业至远程计算机进行运算。

注意：对于提交至远程计算机的作业，远程计算机需要使用UNC路径访问存储输入和输出文件的工作目录。如果工作目录是非UNC路径，Launcher将其转化为一个UNC路径。如果没有转换成功，不能将作业提交至远程计算机。

---

11 故障排除（Troubleshooting）

11.1 简介

该部分内容为用户提供了CMOST使用过程中可能出现问题的解决方法。

11.2 失败和意外终止的CMOST任务

注意：为了创建摄动试验方案来得到正常终止的任务，请参考 Number of Perturbation Experiments for Each Abnormal Experiment 。

1. 失败/未完成的任务是由于硬件或软件原因导致运算没有完成。如果某个任务失败/未完成，当硬件或软件问题解决后，会重新进行运算。为了找到任务失败的原因：

   - 在Launcher界面，找到该任务，在信息栏查看是否有错误提示信息。如果没有显示信息栏，点击Add/Remove任务栏添加信息栏。

   - 在记事本打开.log或.out文件，向下滚动至文件结束来检查模拟器是否写入了一些结束信息。

   - 查找.dat文件，在Launcher界面直接将其提交给模拟器运算，查看任务是否能运算。

   一些可能的原因导致失败/未完成任务：

   - Launcher中密码缺失/错误。这种情况可以通过在Launcher中直接提交任务进行验证。如果真是这种情况，使用正确的密码即可。
   Launcher | Configuration | Password | Add/Modify.
   - 磁盘空间不足。 在模拟器的工作目录检查可用磁盘空间的大小。如果某个运行的计算机设置了远程Schedulers进行辅助运算，远程计算机运算的模拟结果会被写入到一个临时文件夹，当任务运算完成后，将所有文件复制到本地计算机模拟器所在的工作目录，该目录(通常是C:驱动器)空间被临时文件占用，如果出现空间被占满时，需要从REMOTE EXECUTION COMPUTERS删除某些不想保留的临时文件。

在Windows Server 2008，临时文件夹存储在 C:\ProgramData\CMG \CopyLocalJobs。在Windows Server 2003，临时文件夹存储在C: \Documents and Settings\All Users\Application Data\cmg \CopyLocalJobs。在Windows 7临时文件夹存储在C:\ProgramData\cmg \CopyLocalJobs.

- 间歇性I/O文件问题。通常很难诊断出这个问题。为了减少这个问题，强烈推荐模拟器结果文件（.irf、.mrf、.rst、.out）尽可能的小。更多信息，参考手册关键字 WRST、OUTSRF、WSRF、OUTPRN和 WPRN。如果同时运算多于5个任务，建议使用Windows 2003 或2008 文件服务器来存储CMOST输入/输出文件。

- 远程Scheduler不接受远程任务。如果将一个任务提交给远程计算机运算，而远程计算机仅仅接受本地计算机提交的任务，在这种情况下，运算将会失败。为了验证这种情况，在远程计算机打开 Launcher，检查 Launcher | Configuration | Configure Local Job Server。为了运算远程任务，必须在远程计算机选择 “Use job service to submit any jobs”和“This computer is a CMG Cluster Compute Node”。另外，CMG Job Service必须在远程计算机运行。

- 远程计算机没有模拟器需要的可执行文件。

- 模拟器许可问题。

2. 意外终止的任务是模拟器运算某个任务时，没有达到停止运行时间就终止运算。如果某个任务意外终止，即使再重新运算该数据体，它通常也会停止在同一个点。通常，修改某个数据体的数值控制部分，可能会完成数据体的运算。为了查找引起该问题的原因，检查任务的log 和/或 .out文件。
任务意外终止运算的可能原因:

   - 数据体语法错误。
   - 任务运算不收敛或时间步截断太多等。
   - 由于达到最大运算时间， CMOST会Killed任务。

11.3 异常报告

当CMOST遭遇自身不能解决的问题时，会出现CMOST未处理异常情况 （Unhandled Exception ）对话框，该对话框中包括帮助CMG用户查看问题的相关信息，如下例所示：

如果你遇到了未处理异常情况 （Unhandled Exception ）信息：

1. 点击**Copy Exception**.
2. 打开邮件，将其粘贴。
3. 添加一些你认为可以帮助CMG解决该问题的信息。
4. 在邮件主题，输入“CMOST Unhandled Exception”。
5. 将邮件发送至support@cmgl.ca.
6. 如上面异常报告所示，如果点击Continue，CMOST将忽略该信息，然后试着继续。如果点击Quit，CMOST将立即关闭。

11.4 CMG 诊断工具

典型的CMOST文件运行后生成许多不同类型的文件。对于故障诊断来说，一个具有挑战性和耗时性的工作就是收集所有相关的数据文件。为了简化该过程，CMG研发了诊断数据的工具来自动收集需要的数据文件。该工具分为两步。第一步，当一个Project被保存时，CMOST自动生成一个XML文件（PDF文件），它包含了一系列Study和数据文件的信息。该步骤在私下完成。第二步，当用户联系CMG需求故障诊断帮助时，它们可以开启诊断数据搜集工具来搜集相关的数据。在该步，也可以选择需要诊断的一些研究。该工具将浏览Project/Study目录结构来查找诊断需要的文件。最后，该工具将所需文件压缩成一个zip文件，然后通过邮件或FTP发给CMG技术支持团队。更多信息，请参考Using the CMG Diagnostic Tool。

12 Theoretical Background

12.1 Probability Distribution Functions

12.1.1 Uniform Distribution

Uniform distribution assumes that all values in the defined range are equally probable. Its probability density function is:

$$f(x) = \begin{cases} 0 & x < a \\ \dfrac{1}{b-a} & a \le x \le b \\ 0 & x > b \end{cases}$$

where *a* and *b* are the lower and upper limit of the variable.

12.1.2 Triangle Distribution

The probability density function for the triangle distribution is:

$$f(x) = \begin{cases} 0 & x < a \\ \dfrac{2(x-a)}{(b-a)(c-a)} & a \le x \le c \\ \dfrac{2(b-x)}{(b-a)(b-c)} & c < x \le b \\ 0 & x > b \end{cases}$$

where *a* and *b* are the lower and upper limit of the variable, and *c* is the peak (mode).

12.1.3 Truncated Normal Distribution

The probability density function for the Gaussian normal distribution is:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-(x-\mu)^2}{2\sigma^2}\right)$$

where μ and σ are the mean and standard deviation of the variable.

In CMOST, the normal distribution is truncated by user-defined minimum and maximum values:

$$\text{Min} \le x \le \text{Max}$$

The default min and max values are -1E+308 and 1E+308 respectively.

12.1.4 Truncated Log Normal Distribution

The probability density function for the log normal distribution is:

$$f(x) = \frac{1}{x\sigma\sqrt{2\pi}} \exp\left(\frac{-(\ln x - \mu)^2}{2\sigma^2}\right)$$

where μ and σ are the mean and standard deviation of the variable's natural logarithm. By definition, in a log normal distribution, the variable's logarithm is normally distributed.

In contrast, the mean and standard deviation of the non-logarithm values are denoted *m* and *s*.

To calculate *m* and *s* from μ and σ:

$$m = e^{\mu+0.5\sigma^2}, s = \left(e^{\sigma^2} - 1\right)e^{2\mu+\sigma^2}$$

To calculate μ and σ from *m* and *s*:

$$\mu = \ln\left(\frac{m^2}{\sqrt{s^2+m^2}}\right), \sigma = \sqrt{\ln(1+\frac{s}{m^2})}$$

In CMOST, the log normal distribution is truncated by user-defined minimum and maximum values:

$$\text{Min} \le x \le \text{Max}$$

The default *Min* and *Max* values are 1E-308 and 1E+308 respectively.

12.1.5 Deterministic Distributions

Unlike probability distributions, where the uncertainty of an input parameter is described by a distribution, deterministic distributions treat the input parameter as a constant. A fixed value is defined for the parameter, since there is no uncertainty about its value.

12.1.6 Custom Distribution

CMOST provides predefined discrete and continuous distributions for input parameters; however, if none of these distributions is appropriate for the uncertainty of an input parameter, users can create a custom distribution.

The custom distribution is given as a table of intervals and the corresponding probability values. For example, the following table provides the points that define a custom distribution which indicates the probability is 60% for values between 0.2 and 0.3:

Left Bound	Right Bound	Probability
0.0	0.1	0.05
0.1	0.2	0.15
0.2	0.3	0.60
0.3	0.4	0.15
0.4	0.5	0.05

NOTE: If the probability for all defined intervals does not sum to 1, CMOST will normalize the probability values to ensure that the total cumulative probability is equal to 1.

12.1.7 Discrete Probability Distribution

The discrete distribution is given as a table of x-values and the corresponding probability values. For example, for the discrete distribution defined by the following table, only three values (100, 200, and 300) will be used in Monte Carlo simulation. The probability of using 100, 200, and 300 is 25%, 50%, and 25% respectively. If the sum of all probability values is not equal to 1, CMOST will normalize the probability values so that it does equal 1.

X	Probability
100	0.25
200	0.50
300	0.25

12.2 Objective Function Types

Two types of objective functions are described in this section:

- History Match Error
- Net Present Value

12.2.1 History Match Error

The History Match Error measures the relative difference between the simulation results and measured production data for each objective function. If a field has multiple wells and each well has multiple types of production data to match, CMG recommends you define an objective function for each well. The objective function of each well contains multiple objective function terms, each of which corresponds to a production data type. In practice, it is also common for the quality and importance of measured data to be different for different production data types. In a manual history matching task, these variations are usually taken into account by the reservoir engineer intuitively and qualitatively. In computer-assisted history matching, a quantitative approach should be used to account for the data quality and importance. Therefore, different absolute measurement errors and weights need to be

assigned to different production data types of different wells in calculating objective functions.

In CMOST, the following equation is used to calculate the history match error for well *i*:

$$Q_i = \frac{1}{\sum_{j=1}^{N(i)} tw_{i,j}} \times \sum_{j=1}^{N(i)} \frac{\sqrt{\frac{\sum_{t=1}^{T(i,j)} \left(Y_{i,j,t}^{s} - Y_{i,j,t}^{m}\right)^2}{NT(i,j)}}}{Scale_{i,j}} \times 100\% \times tw_{i,j}$$

where:

i,j,t	Subscripts representing well, production data type, and time respectively
N(i)	Total number of production data types for well *i*
NT(i,j)	Total number of measured data points
$Y_{i,j,t}^{s}$	Simulated results
$Y_{i,j,t}^{m}$	Measured results
$tw_{i,j}$	Term weight
$Scale_{i,j}$	Normalization scale

The normalization scale is calculated using one of the following four methods.

Method #1 applies when the number of measured data points is greater than 5 and the normalization method is set to AUTO. In this method, the normalization scale is the maximum of the following three quantities:

$$\Delta Y_{i,j}^{m} + 4 \times Merr_{i,j}$$

$$0.5 \times \min\left(\left|\max(Y_{i,j,t}^{m})\right|, \left|\min(Y_{i,j,t}^{m})\right|\right) + 4 \times Merr_{i,j}$$

$$0.25 \times \min\left(\left|\max(Y_{i,j,t}^{m})\right|, \left|\min(Y_{i,j,t}^{m})\right|\right) + 4 \times Merr_{i,j}$$

where:

$\Delta Y_{i,j}^{m}$	Measured maximum change for well *i* and production data type *j*
$Merr_{i,j}$	Measurement error

The value of measurement error (ME) means that if the simulated result is between (historical value – ME) and (historical value + ME), the match is considered to be satisfactory (or perfect because it is within the range of measurement accuracy). So ME is the ½ absolute error range.

Method #2 applies when the number of measured data points is small (≤5), and the normalization method is set to AUTO, in which case, the normalization scale is obtained by:

$$\text{Scale}_{i,j} = \max(\left| \max(Y^{m}_{i,j,t}) \right|, \left| \min(Y^{m}_{i,j,t}) \right|) + 4 \times \text{Merr}_{i,j}$$

Method #3 applies when the normalization method is set to OFF, in which case:

$$\text{Scale}_{i,j} = 1$$

Method #4 applies when the normalization method is set to be MeasurementErrorOnly, in which case:

$$\text{Scale}_{i,j} = \text{Merr}_{i,j}$$

As can be seen from the above equations, the calculated history match error is a dimensionless percentage relative error for methods #1, #2, and #4. If the simulation results are exactly the same as measured data, the calculated history match error is 0% which indicates a perfect match. Our experience indicates that if the history match error is less than 5%, the match is usually acceptable.

The global history match error is calculated using the weighted average method:

$$Q_{global} = \frac{1}{\sum_{i=1}^{NW} w_i} \sum_{i=1}^{NW} w_i Q_i$$

where:

Q_{global}	Global objective function
Q_i	Objective function for well i
NW	Total number of wells
w_i	Weight of Q_i in the calculation of Q_{global}

12.2.2 Net Present Value

In finance and economics, discounting is the process of finding the present value of an amount of cash at some future date. The net present value of a cash flow is determined by reducing its value by the appropriate discount rate for each unit of time between the time when the cash flow is to be valued and the time of the cash flow. The time when the cash flow is to be valued is called the NPV Present Date in CMOST.

To calculate the present value of a single cash flow, it is divided by one plus the interest rate for each period of time that will pass:

$$PV = \frac{R_t}{(1+I)^t}$$

where:

t	Time of the cash flow
I	Discount rate (interest rate)
R_t	Net cash flow (positive for inflow, and negative for outflow) at time t

Net present value (NPV) is defined as the total present value (PV) of a time series of cash flows. It is a standard method for using the time value of money to appraise long-term projects. Each cash inflow/outflow is discounted back to its present value (PV). Then they are summed. Therefore NPV is the sum of all cash inflows/outflows:

$$NPV = \sum_{t=1}^{T} \frac{R_t}{(1+I)^t}$$

The method for calculating cash flow depends on the property. If the selected property is a daily property, such as Oil Rate SC-Daily, then cash flow is calculated daily. If the selected property is a monthly property, such as Oil Rate SC-Monthly, cash flow is calculated monthly. If a property does not specify the frequency, such as Oil Rate SC, then the cash flow is calculated daily.

Do not select cumulative properties unless the cash flow is to be calculated for one day only:

$$NPV = \sum_{j=1}^{NJ} \sum_{t=N1}^{N2} \frac{Quantity \times UnitValue \times ConversionFactor}{(1 + DailyInterestRate)^t}$$

where:

t	Time of the cash flow in days (the number of days elapsed from the NPV Present Date to the date when the Property Value is read).
N1	Number of days from the NPV Present Date to the Start Date Time.
N2	Number of days from the NPV Present Date to the End Date Time.
Quantity	Value read from the SR2 files using the user-specified origin name and property name
Unit Value	User-specified cash flow value per Quantity (positive for inflow, and negative for outflow)
j	Represents each objective function term
NJ	Number of objective function terms for the Net Present Value objective function.

The yearly interest rate is input by the user. Monthly, quarterly, daily interest rates are converted from the yearly rate; for example, the yearly interest (discount) rate is converted to the daily interest rate using the following formula:

$$\text{DailyInterestRate} = e^{\frac{\ln(1+\text{YearlyInterestRate})}{365}} - 1$$

CMOST uses the CMOST unit system defined on the General Properties page to read SR2 files and a proper Unit Value for each objective function term must be entered according to the chosen unit system. For example, if the CMOST unit system is *Field*, the unit for oil rate will be bbl/day and the unit value should be dollar per barrel. If the unit system is *SI*, the unit for oil rate will be m^3/day and the unit value should be dollar per m^3.

12.3 Sampling Methods

Parameter space sampling is the most important step in sensitivity analysis and uncertainty assessment. The outcome of parameter space sampling is a design for laying out a detailed simulation plan in advance of performing simulations. A well-chosen design maximizes the amount of "information" that can be obtained for a given amount of simulation effort. Below is an introduction to some basic terminology used in CMOST.

- **Parameters** (variables, factors): Simulation inputs which a researcher manipulates to cause changes in simulation outputs. A parameter can have two or more sample values.
- **Sample values** (levels): The different values of a parameter.
- **Objective functions** (responses): The outputs of a simulation.
- **Experiment**: An experiment represents the combination of one particular sample value for each parameter in the simulation model.
- **Parameter space** (search space): The number of all possible experiments for a given set of parameters and sample values.
- **Sampling**: Process of selecting a set of experiments from all possible experiments.
- **Design**: A set of experiments generated by the sampling process. A good design with desirable characteristics allows you to fit an accurate proxy model and draw reliable conclusions regarding parameter effects.
- **Effect**: How changing the value of a parameter changes the objective function. The effect of a single parameter, as opposed to the effect of an interaction, is also called a main effect.
- **Interaction**: Occurs when the effect of one parameter on an objective function depends on the level of another parameter.

For a given set of parameters and sample values, the parameter space is usually extremely large. For example, the number of all possible experiments for 15 parameters with three sample values for each parameter is 3^{15}, or 14,348,907. If we want to select a set of 600 experiments

from the total of 14,348,907 experiments, there is an exceptionally large number of ways to do the selection. According to statistical experimental design theory, to efficiently explore the parameter space, the design (the set of experiments) selected should possess two desirable characteristics:

- Approximate orthogonality of the input parameters.
- Space-filling, that is, the sampling points (experiments) should be evenly distributed in the parameter space. In other words, the collection of experiments should be a representative subset of all possible experiments. This is indicated by the minimum sampling distance, the larger the better.

The orthogonality of two columns in a design matrix is measured by the correlation between two column vectors $\vec{v} = (v_1, v_2, ..., v_n)$ and $\vec{w} = (w_1, w_2, ..., w_n)$:

$$\frac{\sum_{i=1}^{n}\left[(v_i - \overline{v})(w_i - \overline{w})\right]}{\sqrt{\sum_{i=1}^{n}(v_i - \overline{v})^2 \sum_{i=1}^{n}(w_i - \overline{w})^2}}$$

If two columns have zero correlation, they are orthogonal. If all of the columns in the design matrix are orthogonal, the design is an orthogonal design. An orthogonal design is desirable since it ensures independence among the coefficient estimates in a regression model.

In CMOST, the orthogonality of a design is measured by the maximum pair-wise correlation of the columns of a design matrix. The maximum pair-wise correlation is found by calculating the absolute value of the correlation coefficient for all pairs of column vectors in the design matrix, and then selecting the maximum of these values. A value of *0* is best (indicating orthogonality), and a value of *1* is worst (indicating that at least one column in the design matrix is a linear combination of the remaining columns). Generally, to ensure the accuracy of sensitivity analysis and uncertainty assessment results, the maximum pair-wise correlation of the design should be less than 0.2.

Another desirable feature for a design is its ability to evenly spread points in the parameter space. For many interpolation methods used to generate proxies for the outputs of simulations, the errors get larger as the interpolated point moves away from an observation point (in many cases the errors are zero at the interpolation points). Having the observation points evenly distributed would then guarantee a uniform accuracy for the approximation throughout the parameter space. Designs with these characteristics are called “space-filling designs”. Another benefit of space-filling designs is the avoidance of undesirable artificial correlations between the parameters.

In CMOST, the space-filling of a design is assessed by the Euclidean minimum distance which is the minimum Euclidean distance of all design points (experiments). A large value of Euclidean minimum distance means that no two points are close to each other. Between two designs, the one with the greater minimum distance between any two points (experiments) is considered to be the better design.

In CMOST, the following sampling methods are available:

- One parameter at a time (OPAAT)
- Two-level classical experimental designs: Fractional factorial and Plackett-Burman designs
- Three-level classical experimental designs: Box-Behnken and Central Composite designs
- Latin hypercube design

12.3.1 One-Parameter-at-a-Time Sampling

One-parameter-at-a-time sampling is a traditional method for sensitivity analysis. In this method, the researcher seeks to gain information about the effect of a parameter by varying only one parameter at a time. This procedure is repeated in turn for all parameters to be studied. For example, let us assume we want to perform a sensitivity analysis for the following five parameters.

	Name	Comment	Active	Default Value	Source
1	POR		☑	0.24	Discrete Real
2	PERMH		☑	4000	Discrete Real
3	PERMV		☑	2200	Discrete Real
4	HTSORW		☑	0.22	Discrete Real
5	HTSORG		☑	0.06	Discrete Real

The candidate values for these parameters are:

Parameter	Candidate Values
POR	0.22, 0.29, 0.36
PERMH	3000, 4500, 6000
PERMV	2000, 2400, 2800
HTSORW	0.16, 0.21, 0.26
HTSORG	0.02, 0.04, 0.06

Using One-Parameter-at-a-Time sampling, CMOST will generate the following 11 experiments:

ID	Generator	POR	PERMH	PERMV	HTSORW	HTSORG
0	Reuse	0.24	4000	2200	0.22	0.06
1	One Parameter At A Time	0.29	4500	2400	0.21	0.04
2	One Parameter At A Time	0.22	4500	2400	0.21	0.04
3	One Parameter At A Time	0.36	4500	2400	0.21	0.04
4	One Parameter At A Time	0.29	3000	2400	0.21	0.04
5	One Parameter At A Time	0.29	6000	2400	0.21	0.04
6	One Parameter At A Time	0.29	4500	2000	0.21	0.04
7	One Parameter At A Time	0.29	4500	2800	0.21	0.04
8	One Parameter At A Time	0.29	4500	2400	0.16	0.04
9	One Parameter At A Time	0.29	4500	2400	0.26	0.04
10	One Parameter At A Time	0.29	4500	2400	0.21	0.02
11	One Parameter At A Time	0.29	4500	2400	0.21	0.06

In the above example, the parameter default values are shown in Experiment ID 0 (base case). It can be seen from the table that for experiments 1, 2, and 3, all parameters except for POR use their middle candidate values. Therefore, we can determine the conditional main effect of POR by comparing the simulation results for experiments 1, 2, and 3. Similarly, we can determine the effect of PERMH by comparing the simulation results for experiments 3, 4, and 5.

The use of one-parameter-at-a-time sampling is generally discouraged by researchers, for the following reasons:

- More runs are required for the same precision in effect estimation
- Interactions between parameters cannot be captured
- Conclusions from the analysis are not general (i.e., only conditional main effects are revealed)

12.3.2 Latin Hypercube Design

12.3.2.1 Evolution of Latin Hypercube

To explain the fundamentals of Latin hypercube design, this section traces the line of literature from random designs to Latin hypercube sampling to Latin hypercube to orthogonal Latin hypercube (Cioppa, 2002).

Random design was proposed by Satterthwaite (1959). In a random design, a random sampling process with replacement is used to choose all or some of the elements of each variable in the design matrix. The principal criticisms of random designs are that the interpretation of the results cannot be justified due to random confounding and the estimators of the coefficients could be biased.

To improve random design, Mckay et al. (1979) proposed Latin hypercube sampling. In Latin hypercube sampling, the input variables are considered to be random variables with known distribution functions. For each input variable, all portions of its distribution are represented by input values which divide its range into *n* strata of equal probability and sampling once from each stratum. For each input variable, the *n* sampled input values are assigned at random to *n* cases.

As an example, let us assume there are four input variables, each having a uniform [0, 1] distribution and 10 simulation runs are to be made. For all four variables, one design value is independently chosen at random from within each of the 10 equal probable intervals [0.0, 0.1), [0.1, 0.2) , [0.2, 0.3) , [0.3, 0.4) , [0.4, 0.5) , [0.5, 0.6) , [0.6, 0.7) , [0.7, 0.8) , [0.8, 0.9) , and [0.9, 1.0]. For every input variable, the order in which the 10 sampled values appear in the design matrix is randomly determined. The following table shows a design matrix obtained by this procedure. It is noted that, as shown in this example, design matrices generated in this way will likely have correlations between columns.

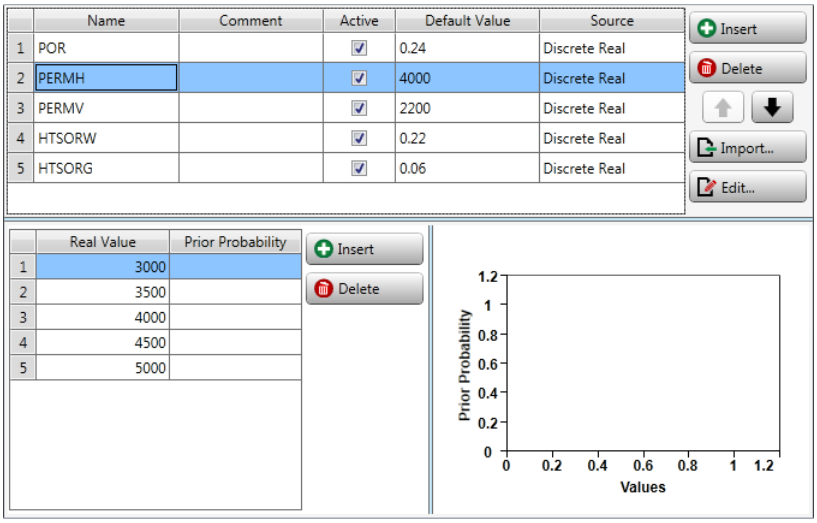
Run	X1	X2	X3	X4
1	0.32	0.17	0.91	0.71
2	0.53	0.58	0.30	0.93
3	0.92	0.84	0.48	0.12
4	0.17	0.90	0.05	0.22
5	0.29	0.02	0.16	0.30
6	0.45	0.41	0.83	0.87
7	0.63	0.68	0.74	0.04
8	0.75	0.24	0.66	0.61
9	0.87	0.79	0.52	0.48
10	0.01	0.36	0.22	0.53

A common variant of the design generated by Latin hypercube sampling is called Latin hypercube (Tang, 1993). In Latin hypercube, the input values for every variable are predetermined and there is no sampling within strata. An $n \times k$ Latin hypercube consists of *k* permutations of the vector $\{1, 2, \ldots, n\}^T$. Each element of the vector represents a sample value (level). Each of the *k* columns of the design matrix contains the levels 1, 2, …, *n*, randomly permuted, with each possible permutation being equally likely to appear in the design matrix.

To enhance the capability of Latin hypercube designs for regression analysis, Ye (1998) constructs orthogonal Latin hypercubes. An orthogonal Latin hypercube is defined as a Latin hypercube for which every pair of columns has zero correlation. Furthermore, in Ye's orthogonal Latin hypercube construction, the element-wise square of each column has zero correlation with all other columns, and the element-wise product of every two columns has zero correlation with all other columns. These properties ensure the independence of estimates of linear effects of each variable and the estimates of the quadratic effects and interaction effects are uncorrelated with the estimates of the linear effects.

12.3.2.2 Latin Hypercube Design in CMOST

The Latin hypercube designs generated by CMOST use a more general variant of the above. Specifically, each of the parameters can have any number of sample values. The sample values can be evenly distributed (uniform distribution) or not-evenly distributed as they are entered by the user. To combine the sample values to create design points (job patterns) in the design, draws without replacement are done; i.e., for the first point a value for each parameter is selected randomly from the set of possible values, for the second point the random selection is done excluding the points already selected and so on. As an example, the following algorithm describes the steps to generate a basic Latin hypercube design for five parameters.

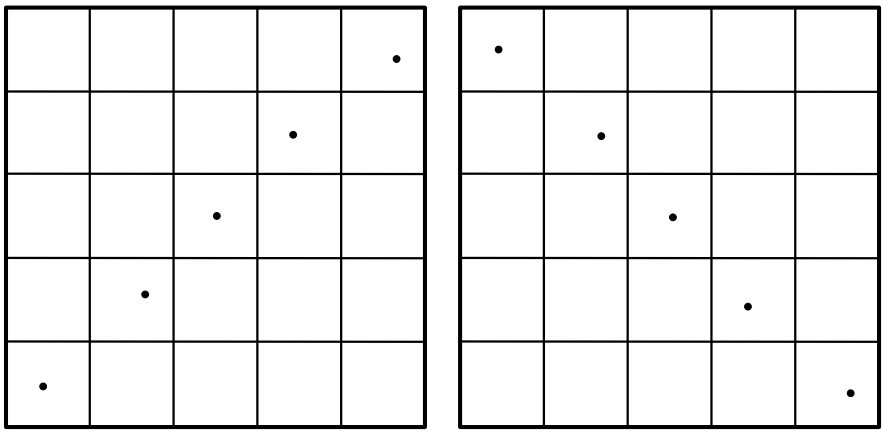


The sample values for these parameters are:

Parameter	Sample Values
POR	0.22, 0.24, 0.26 (3 values)
PERMH	3000, 3500, 4000, 4500, 5000 (5 values)
PERMV	2000, 2500 (2 values)
HTSORG	0.16, 0.18, 0.20, 0.22 (4 values)
HTSORW	0.02, 0.04, 0.06 (3 values)

1. The number of points (job patterns) should be a common multiple for all the numbers of sample values. So the available numbers of jobs patterns are 60, 120, 180, and 240. Let's assume we want to generate a design with 120 points (job patterns).
2. For each parameter, generate a vector with 120 sample values. For example, for parameter POR, the vector should have 40 values of 0.22, 0.24, and 0.26 respectively. The 120 sample values for each parameter are ordered randomly.
3. Assemble all the vectors of all parameters to form a basic Latin hypercube design with 120 design points (job patterns).

The basic Latin hypercube design generated in the above does not guarantee that the points will be evenly distributed and uncorrelated. The figure below shows two examples of valid Latin hypercube design where the points are totally correlated. It is obvious that these designs are not suitable for proxy generation and sensitivity analysis.



To avoid such undesirable artifacts, an iteration (optimization) process is adopted in CMOST to generate Latin hypercube designs with two desirable characteristics.

- Approximate orthogonality of the input parameters.
- Space-filling, that is, the sampling points (experiments) should be evenly distributed in the parameter space.

The iteration (optimization) process is described as follows:

1. Start with an initial basic Latin hypercube design (this is the initial best design).
2. Generate a new basic Latin hypercube design.
3. Calculate the maximum pair-wise correlation of the new design.
4. Calculate Euclidean minimum distance of the new design.
5. Compare the new design with the best design. If the new design outperforms the best design in terms of maximum pair-wise correlation and Euclidean minimum distance, replace the best design with the new design.
6. Repeat steps 2~5 until the number of iterations is reached or an orthogonal design is found.

It is noted that the above iteration (optimization) process does not aim at getting the optimum Latin hypercube design, but just an improvement over the initial Latin hypercube design, while constraining the time to generate the designs in the reasonable range.

12.3.2.3 References

Cioppa, T.M., "Efficient Nearly Orthogonal and Space-Filling Experimental Designs for High-Dimensional Complex Models", Naval Postgraduate School PhD Dissertation, September 2002.

McKay, M.D., Beckman, R.J., and Conover, W.J., "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code", Technometrics, Vol. 21, No. 2, May 1979.

Satterthwaite, F.E., "Random Balance Experimentation", Technometrics, Vol. 1, No. 2, May 1959.

Tang, B., "Orthogonal Array-Based Latin Hypercubes", Journal of the American Statistical Association: Theory and Methods, Vol. 88, No. 424, December 1993.

Ye, K.Q., "Orthogonal column Latin hypercubes and their application in computer experiments", Journal of the American Statistical Association: Theory and Methods, Vol. 93, No. 444, December 1998.

12.3.3 Classical Experimental Design

12.3.3.1 Two-Level Classical Experimental Designs

Two-level designs are typically used in sensitivity analysis to identify main (linear) effects. They are ideal for a quick screening study. They are simple and economical. They also give most of the information required to go to a next-step multilevel response surface experimental design if one is needed.

The standard layout for a two-level design uses - and + notation to denote the "low level" and the "high level" respectively, for each parameter. For example, the matrix below describes an experiment in which 4 runs were conducted with each parameter set to high or low during a run according to whether the matrix had a + or - set for the parameter during that run. If the experiment had more than 2 parameters, there would be an additional column in the matrix for each additional parameter.

Run	Parameter (X1)	Parameter (X2)
1	-	-
2	+	-
3	-	+
4	+	+

The following types of two-level experimental designs are available in CMOST:

- Fractional factorial designs
- Plackett-Burman designs

12.3.3.2 Three-Level Classical Experimental Designs

In uncertainty assessment, three-level experimental designs can be used. In a three-level design, each parameter effect on the response is evaluated at three levels (low, median, and high). Three-level designs are also called response surface designs because in addition to main effects (linear term), both two-term interactions and quadratic terms can be examined. The standard layout for a three-level design uses -, 0, and + notation to denote the "low level", "median

level", and the "high level" respectively, for each parameter. CMOST provides the following types of response surface designs:

- Box-Behnken designs
- Central Composite designs (Uniform Precision)

12.3.4 Parameter Correlation

The Parameter Correlation table is used to incorporate relationships that may exist between green uncertain parameters when performing uncertainty assessment studies.

The technique used by CMOST to account for parameter correlation was introduced by Iman and Conover (1982) (refer to the reference below for more information).

CMOST calculates algorithmically (Iman and Conover, 1982) the realized Spearman's rank correlation matrix if the parameter correlation (desired Spearman's rank correlation matrix) is positive definite.

NOTE: If the desired Spearman's rank correlation matrix is not positive definite, CMOST will try to find the nearest matrix which meets this condition.

12.3.4.1 References

Iman, R. and Conover, W., "A Distribution-Free Approach to Inducing Rank Correlation Among Input Variables", Communications in Statistics - Simulation and Computation, 1982.

12.4 Proxy Modeling

12.4.1 Response Surface Methodology

Response surface methodology (RSM) explores the relationships between input variables (parameters) and responses (objective functions). The main idea of RSM is to use a set of designed experiments to build a proxy (approximation) model to represent the original complicated reservoir simulation model. The most common proxy models take either a linear form or quadratic form of a polynomial function. After a proxy model is built, tornado plots displaying a sequence of parameter estimates can be used to assess the sensitivity of parameters.

12.4.2 Types of Response Surface Models

12.4.2.1 Linear Model

The linear proxy model is:

$$y = a_0 + a_1 x_1 + a_2 x_2 + \cdots + a_k x_k$$

where:

y response (objective function)

$a_1, a_2, \ldots a_k$ coefficients of the proxy model. In some statistics references, referred to as the parameter estimates or unknown parameters.

$x_1, x_2, \ldots x_k$ input variables (parameters)

12.4.2.2 Simple Quadratic Model

The simple second-degree (quadratic) polynomial model is:

$$y = a_0 + \sum_{j=1}^{k} a_j x_j + \sum_{j=1}^{k} a_{jj} x_j^2$$

where:

a_0 intercept

$a_1, a_2, \ldots a_k$ coefficients of linear terms

a_{jj} coefficients of quadratic terms

12.4.2.3 Quadratic Model

The second-degree (quadratic) polynomial model is:

$$y = a_0 + \sum_{j=1}^{k} a_j x_j + \sum_{j=1}^{k} a_{jj} x_j^2 + \sum_{i<j} \sum_{j=2}^{k} a_{ij} x_i x_j$$

where:

a_0 Intercept

$a_1, a_2, \cdots, a_k$ coefficients of linear terms

a_{jj} coefficients of quadratic terms

a_{ij} coefficients of cross (interaction) terms.

12.4.2.4 Reduced Linear Model

For a polynomial proxy model, the statistical significance of each term is characterized by its corresponding *Prob* > | *t* | value. If a term has a large *Prob* > | *t* | value, the term is statistically not significant and it can be removed from the proxy model to simplify and improve the model (i.e., to maximize $R^2_{adjusted}$ and $R^2_{prediction}$).

For further information, refer to Summary of Fit Table. The significance probability (alpha) determines whether a response surface term should be included in the reduced response surface model. If the *Prob* > | *t* | of a term is less than or equal to alpha, the term will be included. The default alpha value is 0.1.

In CMOST, the reduced linear model is built using the following three-step process:

1. Build the linear model.
2. Remove statistically insignificant terms.
3. Build the reduced linear model using the remaining (statistically significant) terms.

12.4.2.5 Reduced Quadratic Model

Similar to the Reduced Linear Model, the reduced quadratic model is built using the following three-step process:

1. Build the quadratic model.
2. Remove statistically insignificant terms.
3. Build the reduced quadratic model using the remaining (statistically significant) terms.

12.4.3 Normalized Parameters (Variables)

The coefficients of a proxy model are highly dependent on the scale of the input variables. For example, if an input variable is converted from millimeter to meter, the coefficient changes by a factor of a thousand. If the same change is applied to a squared (quadratic) term, the coefficient changes by a factor of a million. Since we are interested in the effect size indicated by the coefficients, we need to examine the coefficients in a more scale-invariant fashion. This means converting from an arbitrary scale to a meaningful one so that the magnitudes of the coefficients can be related to the size of the effects on the response. In CMOST, all input variables (parameters) are scaled to have a mean of zero and a range from -1 to 1. This corresponds to the scaling used in traditional experimental design. For a linear term, the coefficient is half the predicted response change as the input variable travels over its full range, from -1 to 1.

12.4.4 Response Surface Model Verification Plot

The model verification plot shows how the data points fit the model by plotting the actual response versus the predicted response for each training and verification job. The distance from each point to the 45 degree line is the error, or residual, for that point. The points that fall on the 45 degree line are those that are perfectly predicted.

To visually show whether the model is statistically significant, the lower and upper 95% confidence curves are superimposed on the actual (simulated) vs. proxy predicted plot. The lower and upper 95% confidence curves are determined using equations given in the paper "Leverage Plots for General Linear Hypotheses" by John Sall, published in *The American Statistician*, November 1990, Vol. 44, No. 4. If the 95% confidence curves cross the horizontal reference line defined by the Mean of Response, then the model is significant. If the curves do not cross, then the model is not significant (at the 5% level).

12.4.5 Summary of Fit Table

The **Summary of Fit** table shows the following numeric summaries of the response surface model:

12.4.5.1 R-Squared (R^2)

The coefficient of multiple determination R^2 is defined as:

$$R^2 = \frac{Sum\ of\ Squares\ (Model)}{Sum\ of\ Squares\ (Total)}$$

R^2 is a measure of the amount of reduction in the variability of the response obtained by using the regressor variables in the model. An R^2 of 1 occurs when there is a perfect fit (the errors are all zero). An R^2 of 0 means that the model predicts the response no better than the overall response mean. It should be noted that a large value of R^2 does not necessarily imply that the regression model is a good one. Adding a variable to the model will always increase R^2, regardless of whether the additional variable is statistically significant or not. Thus, it is possible for models that have large values of R^2 to yield poor prediction of new observations.

12.4.5.2 R-Square Adjusted

R^2 can be adjusted to make it comparable over models with different numbers of regressors by using the degrees of freedom in its computation, in which case:

$$R^2_{adjusted} = 1 - \frac{(n-1)}{(n-p)}(1-R^2)$$

Here n is the number of observations (training experiments) and p is the number of terms in the response model (including the intercept).

In general, $R^2_{adjusted}$ will not always increase as variables are added to the model. In fact, if unnecessary terms are added, the value of $R^2_{adjusted}$ will often decrease. When R^2 and $R^2_{adjusted}$ differ dramatically, there is good chance that non-significant terms have been included in the model.

12.4.5.3 R-Square Prediction

Defined as:

$$R^2_{prediction} = 1 - \frac{PRESS}{Sum\ of\ Squares\ (Total)}$$

where PRESS is the prediction error sum of squares. To calculate PRESS, select an observation i. Fit the regression model to the remaining $n-1$ observations and use this equation to predict the withheld observation y_i. Denoting this predicted value by $\hat{y}_{(i)}$, we can find the prediction error for point i as $e_{(i)} = y_i - \hat{y}_{(i)}$. The prediction error is often called the i[th] PRESS residual. This procedure is repeated for each observation $i = 1, 2, \cdots, n$,

producing a set of n PRESS residuals $e_{(1)}, e_{(2)}, \cdots, e_{(n)}$. The PRESS statistic is then defined as the sum of squares of the n PRESS residuals.

$$PRESS = \sum_{i=1}^{n} e_{(i)}^2 = \sum_{i=1}^{n} [y_i - \hat{y}_{(i)}]^2$$

R-square prediction provides an indication of the predictive capability of the regression model. For example, we could expect a model with $R^2_{prediction} = 0.95$ to "explain" about 95% of the variability in predicting new observations.

12.4.5.4 Mean of Response

Mean of Response is the overall mean of the response values. It is important as a base model for prediction because all other models are compared to it.

12.4.5.5 Standard Error (Summary of Fit table)

Standard Error estimates the standard deviation of the random error. It is the square root of the **Error Mean Square** in the corresponding **Analysis of Variance** table. **Standard Error** is commonly denoted as σ.

12.4.6 Analysis of Variance Table

12.4.6.1 Source

Source lists the three sources of variation: Model, Error, and Total.

12.4.6.2 Degrees of Freedom (DF)

Total DF is used for the simple mean model. Only one degree of freedom is used (the estimate of the mean parameter) in the calculation of variation, so the degrees of freedom for Total is always one less than the number of observations.

Model DF is the number of terms (except for the intercept) used to fit the model.

Error DF is the difference between **Total DF** and **Model DF**.

12.4.6.3 Sum of Squares

The **Sum of Squares** (SS) column accounts for the variability measured in the response. It is the sum of squares of the differences between the fitted response and the actual response.

Total SS is the sum of squared distances of each response from the sample mean which is the base model (or simple mean model) used for comparison with all other models.

Error SS is the sum of squared differences between the fitted values and the actual values. This sum of squares corresponds to the unexplained Error (residual) after fitting the regression model.

Total SS less **Error SS** gives the sum of squares attributed to the Model.

One common set of notations for these is SSR, SSE, and SST for sum of squares due to Regression (model), Error, and Total, respectively.

12.4.6.4 Mean Square

Mean Square is a sum of squares divided by its associated degrees of freedom. This computation converts the sum of squares to an average (mean square).

Error Mean Square estimates the variance of the error term. It is often denoted as s^2.

12.4.6.5 F Ratio

F Ratio is the **Model Mean Square** divided by the **Error Mean Square**. It tests the hypothesis that all of the regression parameters (except the intercept) are zero. Under this whole-model hypothesis, the two mean squares have the same expectation. If the random errors are normal, then under this hypothesis, the values reported in the **Sum of Squares** column are two independent chi-squares. The ratio of these two chi-squares divided by their respective degrees of freedom (reported in the **Degrees of Freedom** column) has an F-distribution. If there is a significant effect in the model, the **F Ratio** is higher than expected by chance alone.

12.4.6.6 Prob > F

Prob > F is the probability of obtaining a greater F-value by chance alone if the specified model fits no better than the overall response mean. Significance probabilities of 0.05 or less are often considered evidence that there is at least one significant regression factor in the model. This significance is also shown graphically in Simulated vs. Proxy Predicted plots, as described in Response Model Verification Plot.

12.4.7 Effect Screening Using Normalized Parameters

12.4.7.1 Term

This column names the estimated terms. The first term is always the intercept. All parameters are normalized from (Low, High) to (-1, +1).

12.4.7.2 Coefficient

This column lists the parameter estimates for each term. These are the coefficients of the response surface model found by least squares.

12.4.7.3 Standard Error (Effect Screening Using Normalized Parameters)

Standard Error is the estimate of the standard deviation of the distribution of the parameter estimate (coefficient). It is used to construct *t*-tests.

12.4.7.4 t Ratio

***t* Ratio** is a statistic that tests whether the true parameter (coefficient) is zero. It is the ratio of the coefficient to its standard error and has a **Student's *t*-distribution** under the hypothesis, given the normal assumptions about the model.

12.4.7.5 Prob > |t|

Prob > |t| is the probability of getting an even greater *t*-statistic (in absolute value), given the hypothesis that the parameter (coefficient) is zero. This is the two-tailed test against the

alternatives in each direction. Probabilities less than 0.05 are often considered as significant evidence that the parameter (coefficient) is not zero.

12.4.7.6 VIF

This column shows the variance inflation factor, which is a useful measure of the multi-collinearity problem. Multi-collinearity refers to one or more near-linear dependencies among the regressor variables due to poor sampling of the design space. Multi-collinearity can have serious effects on the estimates of the model coefficients and on the general applicability of the final model.

The larger the variance inflation factor, the more severe the multi-collinearity. Variance inflation factors should not exceed 4 or 5. If the design matrix is perfectly orthogonal, the variance inflation factor for all terms will be equal to 1.

12.4.8 Linear Model Effect Estimates

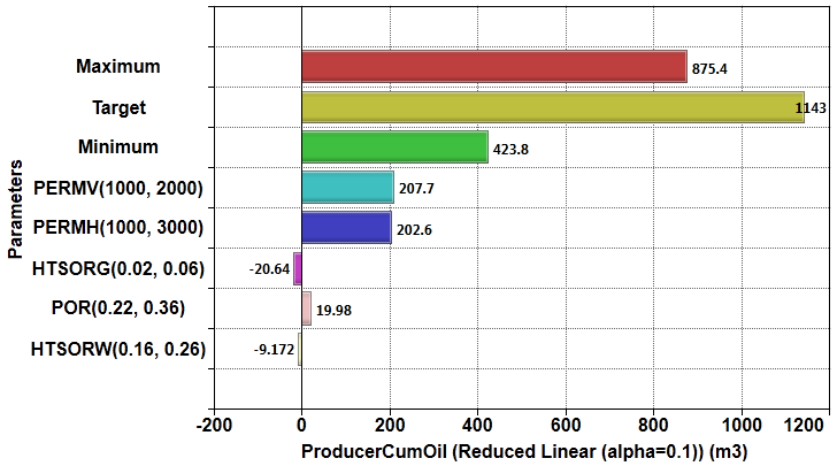
The effect estimate indicates how changing the setting of a parameter changes the response (objective function). The effect estimate of a single parameter is also called a main effect or linear effect. To determine the linear (main) effect estimates, the simulation results are fit using a linear proxy model:

$$y = a_0 + a_1 x_1 + a_2 x_2 + \cdots + a_k x_k$$

In effect screening, the coefficients $a_1, a_2, \cdots, a_k$ are called parameter estimates or effect estimates. In the above equation, a large coefficient suggests that the parameter is important because it means that increasing or decreasing the parameter value leads to a significant change in the objective function (response). On the other hand, a small coefficient would imply that the parameter is not important.

Note that parameter estimates are highly dependent on the scale of the parameter. For example, if you convert a parameter from grams to kilograms, the parameter estimates change by a multiple of a thousand. Therefore, the effect estimates should be determined in a scale-invariant fashion. There are many approaches to doing this. In CMOST, all parameters are scaled to have a mean of zero and a range of two; i.e., all parameters are scaled to have a range from -1 to 1. For a simple linear proxy model, the scaled estimate is half the predicted response change as the parameter travels its full range (i.e., from -1 to 1).

To avoid ambiguous interpretation of tornado plots, CMOST reports the actual predicted response change as the parameter travels from the smallest sample value to the largest sample value. As an example, consider the following tornado plot of linear effect estimates:



The above tornado plot shows that the linear effect estimate for PERMV(1000, 2000) is 207.7. This means that if you increase PERMV from 1000 to 2000, the expected increase of cumulative oil production is 207.7. Here the word "expected" is used because a linear proxy model is an approximation of the real reservoir simulation model. The actual increase of the objective function due to the change of PERMV from 1000 to 2000 varies for different combinations of sample values of the other parameters.

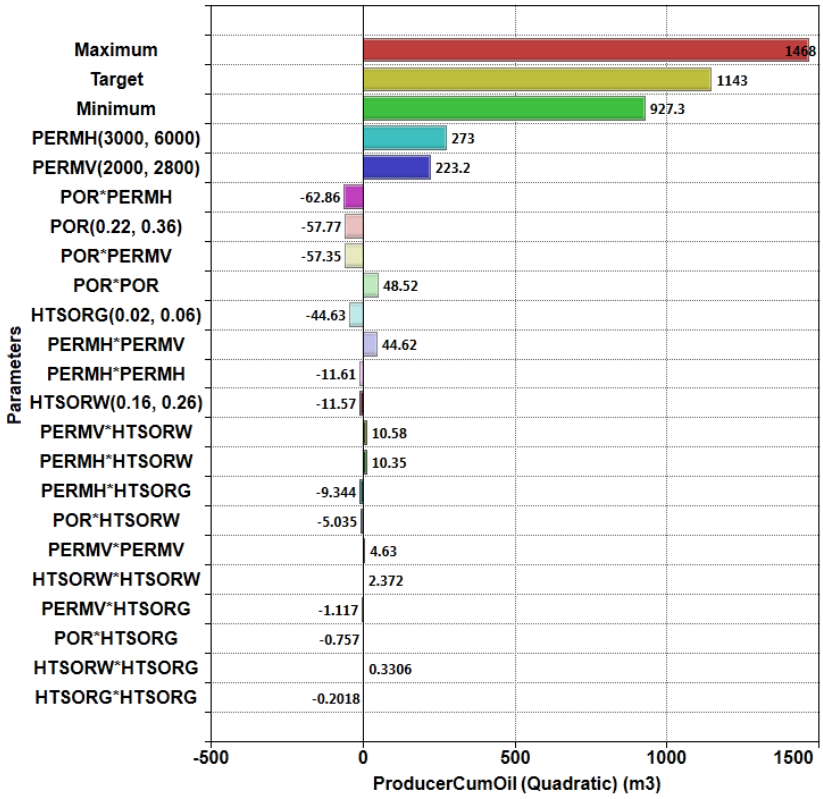
To demonstrate the relative importance of different parameters, all of the effect estimates are plotted on the same scale together with "Maximum", "Minimum", and "Target" values. The **Maximum** is the maximum objective function value of all simulation runs in the design and the **Minimum** is the minimum objective function value of all simulation runs in the design. The **Target** is the value in the target field history file, if one is specified. For example, the above plot shows that the **Maximum** is less than the **Target**. This indicates that it is not possible to match the historical value using the given set of parameters and the defined ranges. Based on the effect estimates of PERMV and PERMH, we may need to adjust the ranges to PERMV(2000, 3000) and PERMH(3000, 5000) to match the historical value (target) of cumulative oil.

12.4.9 Quadratic Model Effect Estimates

For second-degree (quadratic) polynomial models, parameter interaction effects (cross terms $x_i x_j$) and quadratic effects (x_j^2) can be extracted in addition to linear effects (x_j):

$$y = a_0 + \sum_{j=1}^{k} a_j x_j + \sum_{j=1}^{k} a_{jj} x_j^2 + \sum_{i<j} \sum_{j=2}^{k} a_{ij} x_i x_j$$

Similar to linear model effect estimates, quadratic model effect estimates are determined in a scale-invariant fashion. More specifically, all parameters are scaled to have a mean of zero and a range from -1 to 1. For this reason, for the linear and cross terms in the quadratic model, the scaled estimate is half the predicted response change as the parameter travels through its range (from -1 to 1). To avoid ambiguous interpretation of tornado plots, CMOST reports the actual predicted response change as the parameter (or the cross and quadratic terms) travels from the smallest sample value to the largest sample value. A sample tornado of polynomial effect estimates is shown below:



The following table explains the interpretation of the above tornado plot:

Term	Scaled Term	Effect Estimate (see note)
PERMH(3000, 6000)	$\overline{\text{PERMH}} = \frac{2(\text{PERMH} - 3000)}{6000 - 3000} - 1$	273
PERMV(2000, 2800)	$\overline{\text{PERMV}} = \frac{2(\text{PERMV} - 2000)}{2800 - 2000} - 1$	223.2
POR*PERMH	$\overline{\text{POR}} \times \overline{\text{PERMH}}$	-62.86
POR(0.22, 0.36)	$\overline{\text{POR}} = \frac{2(\text{POR} - 0.22)}{0.36 - 0.22} - 1$	-57.77
POP*PERMV	$\overline{\text{POR}} \times \overline{\text{PERMV}}$	-57.35
POR*POR	$\overline{\text{POR}} \times \overline{\text{POR}}$	48.52
HTSORG(0.02, 0.06)	$\overline{\text{HTSORG}} = \frac{2(\text{HTSORG} - 0.02)}{0.06 - 0.02} - 1$	-44.63
PERMH* PERMV	$\overline{\text{PERMH}} \times \overline{\text{PERMV}}$	44.62

NOTE: **Effect Estimate** is the expected change of the objective function when the scaled term travels from -1 to +1.

Analysis of this particular tornado plot suggests the following conclusions regarding the sensitivities of the parameters on cumulative oil production:

1. The two most important effects are the main (linear) effects of PERMH and PERMV
2. The linear effects of POR and HTSORG are relatively important.
3. There are interaction effects between POR and PERMH, POR and PERMV, and PERMH and PERMV.
4. The non-linear (quadratic) effect of POR*POR is also relatively important.

12.4.10 Reduced Model Effect Estimates

It is common that some model terms of a linear or quadratic model are not statistically significant. Consider the quadratic model with the following **Effect Screening** table, where all terms, including those that are statistically insignificant, are included:

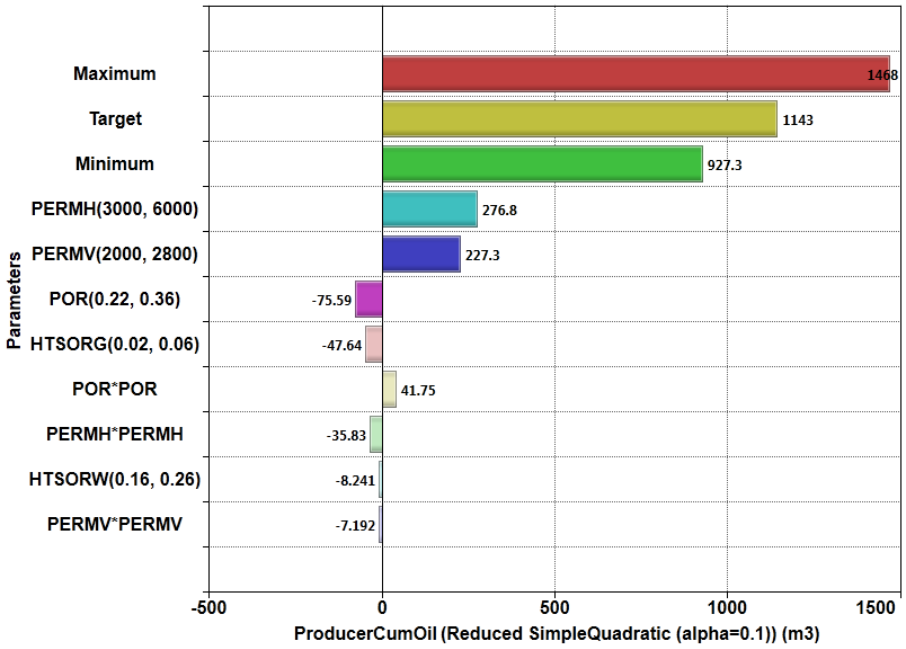
Effect Screening Using Normalized Parameters (-1, +1)

Term	Coefficient	Standard Error	t Ratio	Prob > |t|	VIF
Intercept	1080.63	1.03297	1046.14	<0.00001	0.00
POR(0.22, 0.36)	-28.8831	0.814213	-35.4736	<0.00001	2.11
PERMH(3000, 6000)	136.502	1.25812	108.497	<0.00001	1.99
PERMV(2000, 2800)	111.592	1.08473	102.875	<0.00001	1.95
HTSORW(0.16, 0.26)	-5.78541	0.814184	-7.10578	<0.00001	2.23
HTSORG(0.02, 0.06)	-22.3161	0.725486	-30.7602	<0.00001	2.58
POR*POR	24.2588	0.855241	28.3648	<0.00001	1.34
POR*PERMH	-31.4312	1.45535	-21.5969	<0.00001	2.11
POR*PERMV	-28.6733	1.25344	-22.8757	<0.00001	1.41
POR*HTSORW	-2.51768	0.858944	-2.93113	0.00365	1.52
POR*HTSORG	-0.37848	0.752128	-0.503212	0.61520	1.37
PERMH*PERMH	-5.80305	1.73751	-3.33986	0.00095	1.67
PERMH*PERMV	22.3116	1.92457	11.593	<0.00001	1.99
PERMH*HTSORW	5.17292	1.39466	3.7091	0.00025	2.17
PERMH*HTSORG	-4.67192	1.25538	-3.72151	0.00024	2.12
PERMV*PERMV	2.31511	1.10101	2.10271	0.03635	1.58
PERMV*HTSORW	5.28869	1.18111	4.47773	0.00001	1.28
PERMV*HTSORG	-0.55871	1.15002	-0.485826	0.62746	1.49
HTSORW*HTSORW	1.1861	0.881061	1.34621	0.17928	1.38
HTSORW*HTSORG	0.165283	0.76383	0.216388	0.82884	1.66
HTSORG*HTSORG	-0.100886	0.821057	-0.122873	0.90229	1.09

Through the **Proxy Settings** tab, if you set **Exclude Statistically Insignificant Terms** to *True*, then the proxy model will be built using only those terms that are significant; i.e., which have Prob > |t| values greater than the value you set for **Significant Probability Alpha**. A simple quadratic model can then be built using only significant terms. The **Effect Screening** table and its corresponding tornado plot for the simple quadratic model are shown below:

Effect Screening Using Normalized Parameters (-1, +1)

Term	Coefficient	Standard Error	t Ratio	Prob > |t|	VIF
Intercept	1082.85	1.57379	688.054	<0.00001	0.00
POR(0.22, 0.36)	-37.7964	1.29896	-29.0975	<0.00001	1.26
PERMH(3000, 6000)	138.419	2.25512	61.3796	<0.00001	1.50
PERMV(2000, 2800)	113.641	1.83268	62.0083	<0.00001	1.31
HTSORW(0.16, 0.26)	-4.1204	1.16071	-3.54991	0.00045	1.07
HTSORG(0.02, 0.06)	-23.8189	1.02712	-23.19	<0.00001	1.22
POR*POR	20.8747	1.69917	12.2853	<0.00001	1.25
PERMH*PERMH	-17.9138	3.26005	-5.49496	<0.00001	1.38
PERMV*PERMV	-3.59582	1.98926	-1.80762	0.07166	1.22



12.4.11 Radial Basis Function (RBF) Neural Network

A single-layer radial basis function (RBF) network consists of an input layer of source nodes, a single hidden layer of nonlinear processing units, and an output layer of linear weights. The input-output mapping performed by the RBF network can be described as:

$$\tilde{y}(x_p) = w_0 + \sum_{i=1}^{M} w_i \varphi(x_i, x_p) \quad (1)$$

Where $\varphi(x_i,x_p)$ is the radial basis function, which depends on the distance between the input parameter vector x_p and the center x_i and, in the general case, on the mutual orientation of these vectors in N-dimensional parameter space. The distance and scale orientation, as well as the precise shape of the radial function, are fixed parameters of the model. $\tilde{y}(x_p)$ is the value of an objective function (for example, NPV, Cumulative Oil, or SOR) at the point x_p in N-dimensional parameter space, and M is the size of the training data. Centers x_i correspond to the training parameters in the initial Latin Hypercube design.

Radial basis functions can be used to represent any function, which can then be used as network building blocks. Examples include Gaussian kernels, which are based on the assumption that the network's response monotonically decreases with distance from central points. We have found that the network's response increases as a power function of the distance between two N-dimensional points in the parameter space, in which case, the RBF can best be approximated by the following power function:

$$\varphi_i(x_i, x_p) = 0.01\, L^{0.75}(x_i, x_p) \quad (2)$$

where function L defines the square of the distance between points x_i and x_p in the general case of anisotropic metrics. Normally, the anisotropy factor is neglected for high-dimensional cases due to the infeasibility of full-scale multi-dimensional analysis, in which case, L is selected on the basis of the squared Euclidean distance between parameters $\vec{x} + \vec{z}$ and $\vec{x}$. While this may be satisfactory, it can be improved by recognizing the inherently anisotropic nature of equation (2). With this approach, which lies somewhere between a full-scale multi-dimensional fitting (which is still infeasible as a result of limited training data), and the case where anisotropy is completely neglected, the non-Euclidean squared distance in the M-dimensional parameter space is introduced using the following non-Euclidean diagonal metric relationship:

$$L = \sum_{\alpha=1}^{M} g_\alpha \Delta x_\alpha^{\,2} \qquad (3)$$

where Δx_α is the difference between α -components of parameters at the considered points, and g_α are parameters found by fitting the experimental variogram into the analytical neuron functions of non-Euclidean distance defined by (3) under the constraint $g_\alpha \geq 0, \forall \alpha$. The latter constraint guarantees the positive definition of the metrics, as shown in (4).

It has been found that, for the base case considered below, the RBF neural networks algorithm based on the metrics (4) provides better interpolation than the corresponding algorithm based on Euclidean distance (the latter corresponds to the case $g_\alpha = 1, \forall \alpha$). As a result, training data can be reduced, resulting in less time needed to construct the proxy engine.

Parameters w_i are estimated by exactly fitting the model (1) to the training data, with the additional constraint of normalized nodal weights:

$$\sum_{i=1}^{M} w_i(x_p) = 1 \qquad (4)$$

The values of parameters w_i represent the "importance" of each parameter in the parameter space.

In the current CMOST implementation, in the proxy analysis section, the user can choose between isotropic and anisotropic versions of the RBF proxy (i.e. between Euclidean and non-Euclidean metrics); however, when the size of the Latin Hypercube exceeds 200, only the isotropic version is used since the construction of the anisotropic RBF would take too long.

It is important to understand that the quality of RBF proxy prediction (as well as the quality of polynomial proxy) improves with the increased size of the training Latin Hypercube design; however, a threshold is eventually reached, after which further improvement will be marginal, as shown in the following curve of typical prediction error dynamics:

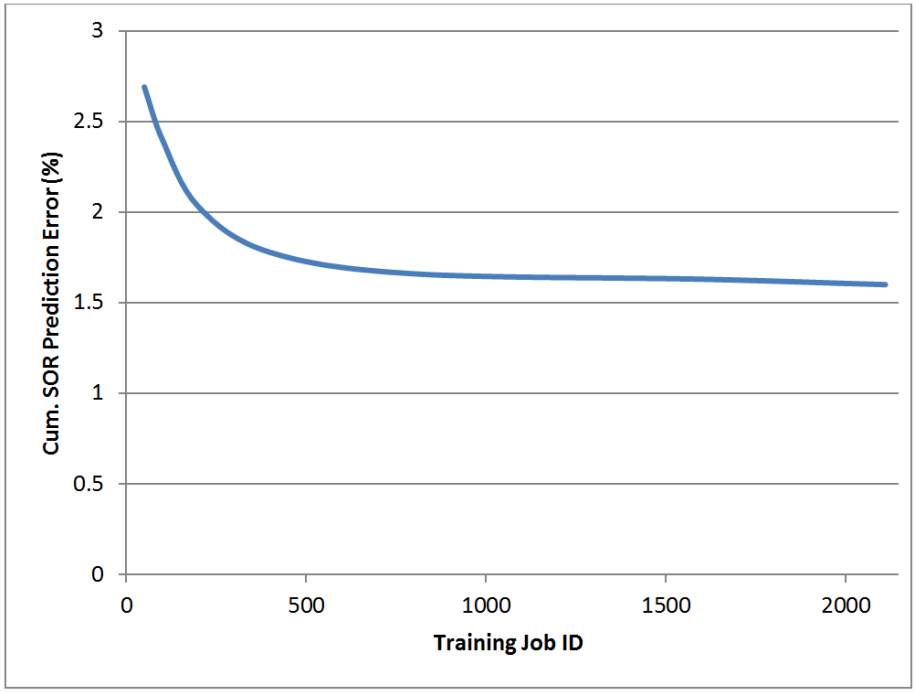


Figure 8 Typical Prediction Error Dynamics for RBF Proxy

The value of the Job ID defining the beginning of this leveling out depends on the type of proxy and the number of parameters. It is important to know this point *a priori*, before constructing the Latin Hypercube design.

12.4.12 Sobol Method

12.4.12.1 Sobol Method Overview

The Sobol method (Sobol 1993) is a type of variance-based sensitivity analysis. For information about interpreting Sobol analysis results in CMOST, refer to Sobol Analysis. The main idea of variance-based methods is to quantify the amount of variance that each input factor X_i contributes to the unconditional variance of output V(Y). Working within a probabilistic framework, the Sobol method decomposes the variance of the model output into fractions that can be attributed to inputs (Saltelli, Annoni, et al. 2010). For example, given a model with two inputs and one output, 60% of the output variance may be caused by variance of the first input, 30% by the variance of the second, and 10% due to interactions between the two. These percentages are directly interpreted as measures of sensitivity.

Sobol-method data analysis is based on the following variance-based measures of sensitivity:

- **First-order sensitivity index, or main effect**: This is the contribution to model output variance due to the variation of X_i alone. It is normalized by the total variance, to provide a fractional contribution (S_i) (Sobol 1993). The first-order index represents the main effect contribution of each input factor to the variance of the output, and is used as an indicator of the importance of X_i on Y, i.e. the

sensitivity of Y to X_i. Various other names for this ratio can be found in the literature, including importance measure and correlation ratio (Saltelli, Ratto, et al. 2008).

- **Higher-order interaction effects**: Input factors interact when their effect on Y cannot be expressed as the sum of their individual effects. Interactions may imply, for instance, that values of output Y are uniquely associated with particular combinations of model inputs, in a way that is not described by the first-order effects S_i mentioned previously (Saltelli, Ratto, et al. 2008). Interactions represent important features of models, and are more difficult to detect than first-order effects (Saltelli, Tarantola, et al. 2004).
- **Total order effect**: The sum of all first- and higher-order effects that an input factor accounts for is called the total effect. For an input X_i, the total sensitivity index S_{Ti} is defined as the sum of all indices relating to X_i (Saltelli, Ratto, et al. 2008). In the case of a model with three input factors (k = 3), for example, the total sensitivity index for input factor X_1 would be: $S_{T1} = S_1 + S_{12} + S_{13} + S_{123}$.

Brute-force computation of all effects, to obtain the total effect, is not practical when the number of input factors (k) gets large, since the number of terms that need to be evaluated is equal to $2k - 1$ (Ekstrom 2005). Instead, we use techniques for estimating total indices for a computing cost similar to that for calculating first-order indices, thereby circumventing the so-called curse of dimensionality. We generally compute the set of all S_i plus the set of all S_{Ti} to obtain a cost-effective determination of model sensitivities (Saltelli, Tarantola, et al. 2004). The method for computing these indices is outlined in the following sections.

Variance-based measures of sensitivity are attractive because they measure sensitivity across the entire input space (i.e. global sensitivity analysis), they can deal with nonlinear responses, and they can measure the effect of interactions in non-additive systems (Saltelli, Annoni, et al. 2010).

12.4.12.2 Calculation Steps

Sobol introduced the first-order sensitivity index by decomposing the model function into summands of increasing dimensionality (Sobol 1993):

$$f(X) = f_0 + \sum_{i=1}^{d} f_i(X_i) + \sum_{i<j}^{d} f_{ij}(X_i X_j) + \cdots + f_{12\ldots d}$$

This representation of model function f(X) holds if f_0 is a constant, and the integrals of every summand over any of the variables are zero, i.e.:

$$\int_0^1 f_{i_1 i_2 \ldots i_s}(X_{i_1}, X_{i_2}, \ldots, X_{i_s}) dX_{i_k} = 0, for\ k = i_1, \ldots, i_s$$

As a consequence of this, all of the summands are mutually orthogonal. The total variance V(Y) is defined as:

$$\int_0^1 f_{i_1 i_2 \dots i_s}\left(X_{i_1}, X_{i_2}, \dots, X_{i_s}\right) dX_{i_k} = 0, for\ k = i_1, \dots, i_s$$

and the partial variances can be computed from each of the terms in the decomposed model function:

$$V_{i_1 i_2 \dots i_s} = \int_0^1 \dots \int_0^1 f^2_{i_1 i_2 \dots i_s}\left(X_{i_1}, X_{i_2}, \dots, X_{i_s}\right) dX_{i_i} \dots dX_{i_s}$$

where:

$$1 \leq i_1 < \dots < i_s \leq k \text{ and } s = 1, \dots, k$$

For analytically tractable functions, the above indices may be calculated analytically by evaluating the integrals in the decomposition; however, in the vast majority of cases, they are estimated using the Monte Carlo method.

12.4.12.3 Sampling Sequence

The Monte Carlo approach involves generating a sequence of randomly distributed points inside the unit hypercube. In practice, it is common to substitute random sequences with low-discrepancy sequences to improve the efficiency of the estimators (Saltelli, Ratto, et al. 2008). This is known as the quasi-Monte Carlo method. These sequences are specifically designed to generate samples of points as uniformly as possible over the unit hypercube. Unlike random numbers, successive quasi-random points are determined with prior knowledge of the position of previously sampled points, filling in the gaps between them. Sobol sequences are examples of quasi-random low-discrepancy sequences. They outperform crude Monte Carlo sampling in estimating multi-dimensional integrals (Saltelli, Tarantola, et al. 2004).

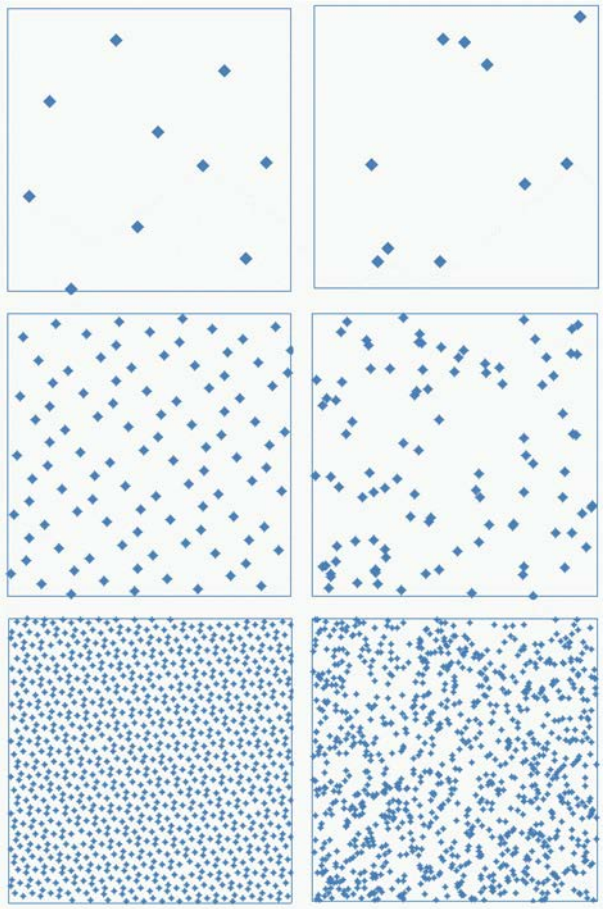


Figure 9: Sampling Points from Low-Discrepancy Sequence (left) Compared with Points from Pseudorandom Number Source (right) [low-discrepancy sequence covers space more evenly]

12.4.12.4 Estimators

Indices are calculated using a quasi-Monte Carlo method (Saltelli, Ratto, et al. 2008), as follows:

1. Generate a *(N, 2k)* matrix of random numbers (*k* is the number of inputs) using the Sobol sequence described previously, and define two matrices of data (A and B), each containing half of the sample. *N* is referred to as the base sample. The order of magnitude of *N* is from a few hundred to a few thousand.

$$A = \begin{bmatrix} x_1^{(1)} & x_2^{(1)} & \cdots & x_i^{(1)} & \cdots & x_k^{(1)} \\ x_1^{(2)} & x_2^{(2)} & \cdots & x_i^{(2)} & \cdots & x_k^{(2)} \\ \vdots & & \ddots & & \vdots & \\ x_1^{(N-1)} & x_2^{(N-1)} & \cdots & x_i^{(N-1)} & \cdots & x_k^{(N-1)} \\ x_1^{(N)} & x_2^{(N)} & \cdots & x_i^{(N)} & \cdots & x_k^{(N)} \end{bmatrix}$$

$$B = \begin{bmatrix} x_{k+1}^{(1)} & x_{k+2}^{(1)} & \cdots & x_{k+i}^{(1)} & \cdots & x_{2k}^{(1)} \\ x_{k+1}^{(2)} & x_{k+2}^{(2)} & \cdots & x_{k+i}^{(2)} & \cdots & x_{2k}^{(2)} \\ \vdots & & \ddots & & \vdots & \\ x_{k+1}^{(N-1)} & x_{k+2}^{(N-1)} & \cdots & x_{k+i}^{(N-1)} & \cdots & x_{2k}^{(N-1)} \\ x_{k+1}^{(N)} & x_{k+2}^{(N)} & \cdots & x_{k+i}^{(N)} & \cdots & x_{2k}^{(N)} \end{bmatrix}$$

2. Define a matrix C_i formed by all columns of B except for the i^{th} column, which is taken from A:

$$C_i = \begin{bmatrix} x_{k+1}^{(1)} & x_{k+2}^{(1)} & \cdots & x_i^{(1)} & \cdots & x_{2k}^{(1)} \\ x_{k+1}^{(2)} & x_{k+2}^{(2)} & \cdots & x_i^{(2)} & \cdots & x_{2k}^{(2)} \\ \vdots & & \ddots & & \vdots & \\ x_{k+1}^{(N-1)} & x_{k+2}^{(N-1)} & \cdots & x_i^{(N-1)} & \cdots & x_{2k}^{(N-1)} \\ x_{k+1}^{(N)} & x_{k+2}^{(N)} & \cdots & x_i^{(N)} & \cdots & x_{2k}^{(N)} \end{bmatrix}$$

3. Compute the model output for all input values in sample matrices A, B, and all C_i, obtaining vectors of model outputs of dimension N × 1:

$$y_A = f(A),\ y_B = f(B),\ y_{C_i} = f(C_i)$$

These vectors are all that is needed to compute the Monte Carlo estimates of total and partial variances, and from these, to calculate first and total effect indices:

$$S_i = \frac{V\left[E\left(Y \mid X_i\right)\right]}{V\left(Y\right)} = \frac{\frac{1}{N}\sum y_A^{(j)} y_{C_i}^{(j)} - \frac{1}{N^2}\sum y_A^{(j)} \sum y_B^{(j)}}{\frac{1}{N}\sum \left(y_A^{(j)}\right)^2 - f_0^2}$$

where the mean is:

$$f_0^2 = \left(\frac{1}{N}\sum_{j=1}^{N} y_A^{(j)}\right)^2$$

Similarly, the method estimates total effect indices as follows (Saltelli, Annoni, et al. 2010):

$$S_{Ti} = 1 - \frac{V\left[E\left(Y \mid X_{\sim i}\right)\right]}{V\left(Y\right)} = 1 - \frac{\frac{1}{N}\sum y_B^{(j)} y_{C_i}^{(j)} - f_0^2}{\frac{1}{N}\sum \left(y_A^{(j)}\right)^2 - f_0^2}$$

The difference $S_{Ti} - S_i$ is a measure of the degree with which X_i interacts with other input factors.

12.4.13 Morris Method

12.4.13.1 Morris Method Overview

The Morris method of global sensitivity analysis (also called the elementary effects [EE] method) is a screening method used to identify model inputs that have the greatest influence on model outputs. For information about interpreting the results of the Morris method, refer to Morris Analysis. The Morris method is a simple but effective way to screen the important input factors from all of those used by the model (Saltelli, Ratto, et al. 2008). It is based on a replicated, randomized "one-at-a-time" (OAT) experiment design where, in each run, only one input parameter is assigned a new value. This facilitates global sensitivity analysis by making r local changes at different points x_i, selected from the range of possible input values.

Data is then analyzed based on elementary effects to determine the change in output due to the change of a particular input factor in the OAT design. The method is global in the sense that it varies over the range of input factor uncertainty (Ekstrom 2005).

As with other screening methods, the EE method provides a measure of qualitative sensitivity, which leads to the identification of non-influential inputs, and ranks input factors in order of importance, without quantifying their relative importance (Morris 1991).

The Morris method provides two sensitivity measures for each input factor (Saltelli, Ratto, et al. 2008):

- μ, which provides an assessment of the overall importance of an input factor on the model output, and

- σ, which describes non-linear effects and interactions.

With these two measures, the Morris method can be used to determine if the effect of input factor X_i on output Y is:

- negligible,
- linear and additive, or
- nonlinear but involved in interactions with other input factors $X_{\sim i}$.

12.4.13.2 Calculation Steps

Sensitivity measures μ and σ are determined by constructing a set of trajectories in the input space, and randomly moving model inputs one-at-a-time (OAT) (Saltelli, Tarantola, et al. 2004).

For this purpose, the input factor space Ω is first discretized into *p* "levels" with possible input factor values constrained inside a regular *k*-dimensional *p*-level grid, where *k* is number of the model's input factors and *p* is the number of design levels.

The method starts by sampling a set of randomly selected start values (random seed) within the defined ranges of possible values, as described above, then calculating the subsequent model outcome Y.

The second step changes the value for one variable by Δ (with all other inputs maintained at their start values) and calculates the resulting change in model outcome compared to the first run. Next, the values for another variable are changed by Δ (the previous variable is kept at its changed value and all other ones are kept at their start values) and the resulting change in model outcome compared to the second run is calculated. This is repeated until all input variables are changed, and one trajectory is created.

Then, the elementary effect (EE) of a given value of input factor X_i is defined as a finite difference derivative approximation (Saltelli, Ratto, et al. 2008):

$$EE_i(X) = \frac{[Y(X_1, X_2, \ldots, X_{i-1}, X_i + \Delta, X_{i+1}, \ldots, X_k) - Y(X)]}{\Delta}$$

This procedure is repeated *r* times, each time with a different set of start values, which leads to a total number of *r* trajectories, equivalent to $r(k + 1)$ runs. The required number of runs is much less that that required by the more demanding sensitivity analysis methods (Saltelli, Tarantola, et al. 2004).

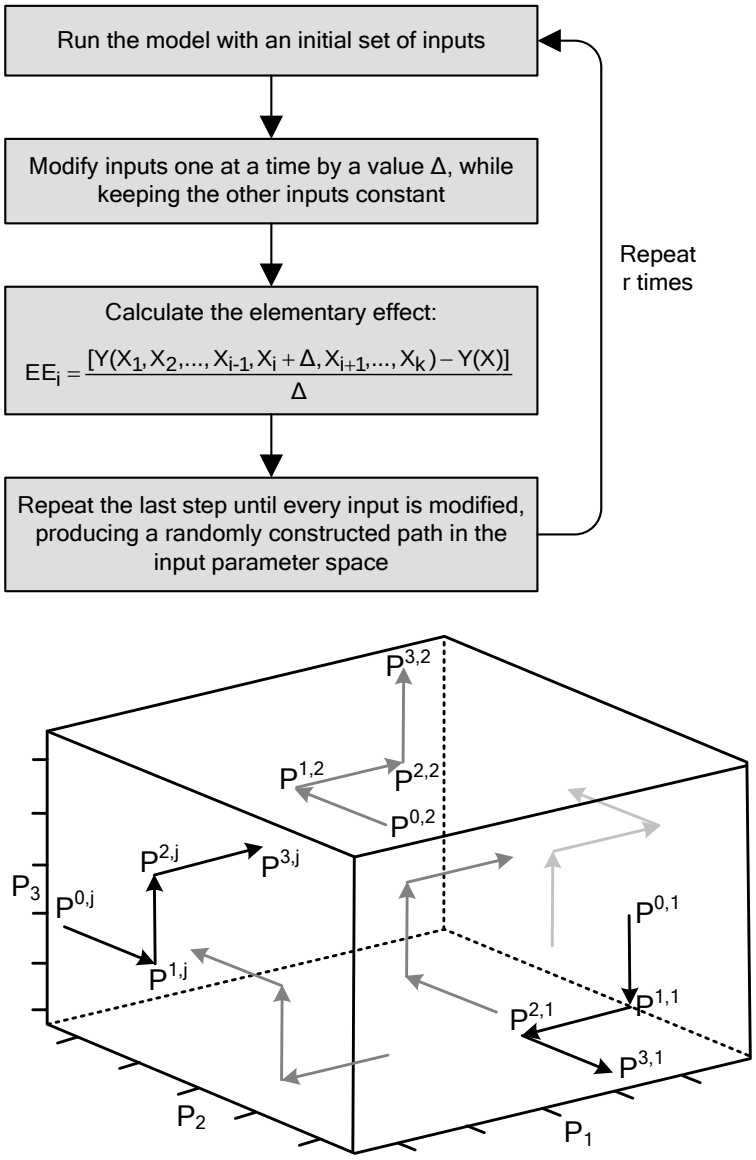


Figure 10 Example of Morris Trajectories for the Case of Three Parameters

The two measures μ and σ are then defined as the mean and standard deviation of the distribution of the elementary effects of each input (Saltelli, Ratto, et al. 2008):

$$\mu_i = \frac{1}{r} \sum_{j=1}^{r} EE(X_i^j)$$

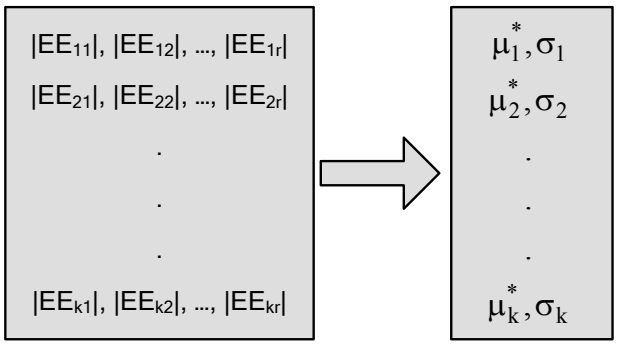
$$\sigma_i = \sqrt{\frac{1}{r-1}\sum_{j=1}^{r}(EE(X_i^j) - \mu_i)^2}$$

An important point about the mean, μ, is the problem of type II errors (failing to identify a factor with considerable influence on the model) (Saltelli, Ratto, et al. 2008). In this case, if the finite distribution of the elementary effects associated with the i^{th} input factor contains negative elements, which occurs when the model is nonmonotonic, some effects may cancel out when the mean is computed. Thus, for a factor with elementary effects that have different signs which cancel each other out), we would have a low value of μ but a large value of σ, which leads to underestimating the importance of the factor or overestimating its involvement in interaction with other factors, and the non-linearity of its effect.

To avoid this problem, (Campolongo and Saltelli 2007) proposed a revised measure, μ^*, which is the mean of the distribution of the absolute values of the elementary effects of the input factors:

$$\mu_i^* = \frac{1}{r}\sum_{j=1}^{r}\left|EE(X_i^j)\right|$$

At the end of the method, after r trajectories are created, each input will have a distribution of r elementary effects. Thus, the mean μ^* and the standard deviation σ of these distributions can be calculated as follows:



12.4.13.3 Choosing Morris Method Parameters

A critical choice related to the implementation of the Morris elementary effects method is the choice of parameters p and Δ (Ekstrom 2005). The choice of p is strictly linked to the choice of r. If a high value of p is considered, producing a high number of possible levels to be explored, the accuracy of the sampling will only seem to be enhanced. If this is not coupled with a high value of r, the effort will be wasted, since many possible levels will remain unexplored (Saltelli, Ratto, et al. 2008).

The literature shows that a convenient choice for the parameters p and Δ is to have p even and Δ equal to:

$$\Delta = \frac{p}{2(p-1)}$$

This choice has the advantage that the design's sampling strategy guarantees equal-probability sampling from the finite distribution of elementary effects associated with the input factors [for details, see (Morris 1991)].

12.5 Optimizers

12.5.1 CMG DECE

The CMOST DECE (Designed Exploration and Controlled Evolution) optimizer implements CMG's proprietary optimization method. The DECE optimization method is based on the process which reservoir engineers commonly use to solve history matching or optimization problems. For simplicity, DECE optimization can be described as an iterative optimization process that first applies a designed exploration stage and then a controlled evolution stage. In the designed exploration stage, the goal is to explore the search space in a designed random manner such that maximum information about the solution space can be obtained. In this stage, experimental design and Tabu search techniques are applied to select parameter values and create representative simulation datasets. In the controlled evolution stage, statistical analyses are performed for the simulation results obtained in the designed exploration stage. Based on the analyses, the DECE algorithm scrutinizes every candidate value of each parameter to determine if there is a better chance to improve solution quality if certain candidate values are rejected (banned) from being picked again. These rejected candidate values are remembered by the algorithm and they will not be used in the next controlled exploration stage. To minimize the possibility of being trapped in local minima, the DECE algorithm checks rejected candidate values from time to time to make sure previous rejection decisions are still valid. If the algorithm determines that certain rejection decisions are not valid, the rejection decisions are recalled and corresponding candidate values are used again.

The DECE optimization method has been successfully applied in a number of real-world reservoir simulation studies, including:

- History matching for a highly heterogeneous black oil model
- History matching of cold heavy oil production with aquifer
- History matching of cyclic steam stimulation process
- NPV optimization for a post-primary SAGD model with aquifer
- NPV optimization for a 6 well pair SAGD model

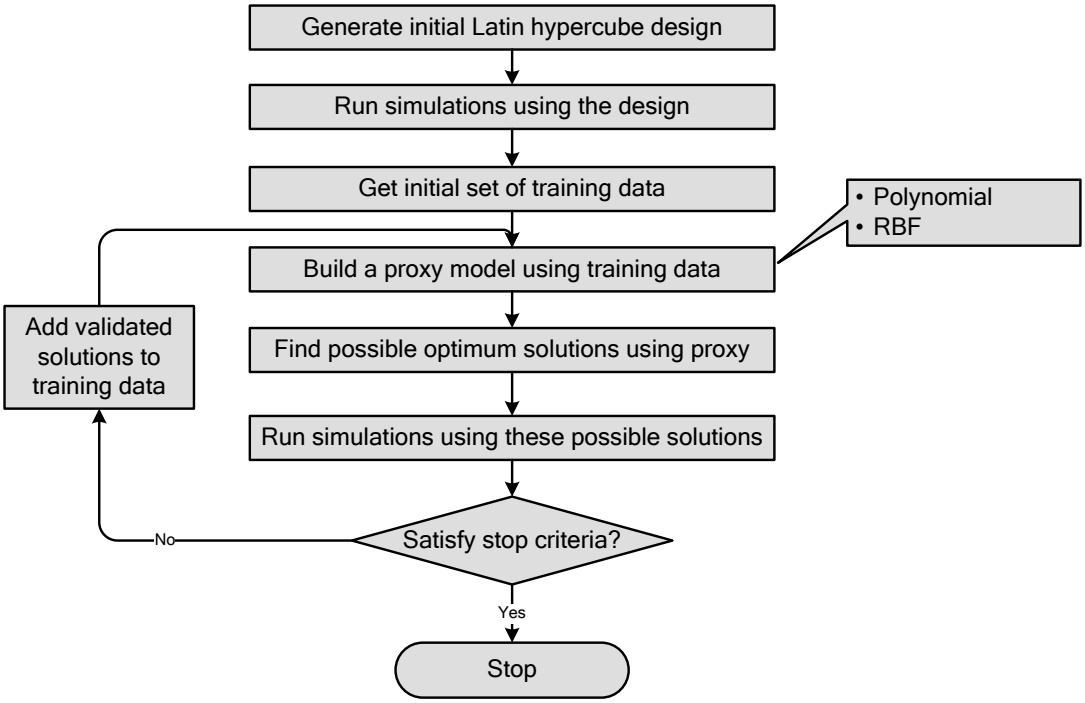
The results demonstrate that DECE optimization method is reliable and efficient. Therefore, it is one of the recommended optimization methods in CMOST.

12.5.2 Latin Hypercube plus Proxy Optimization

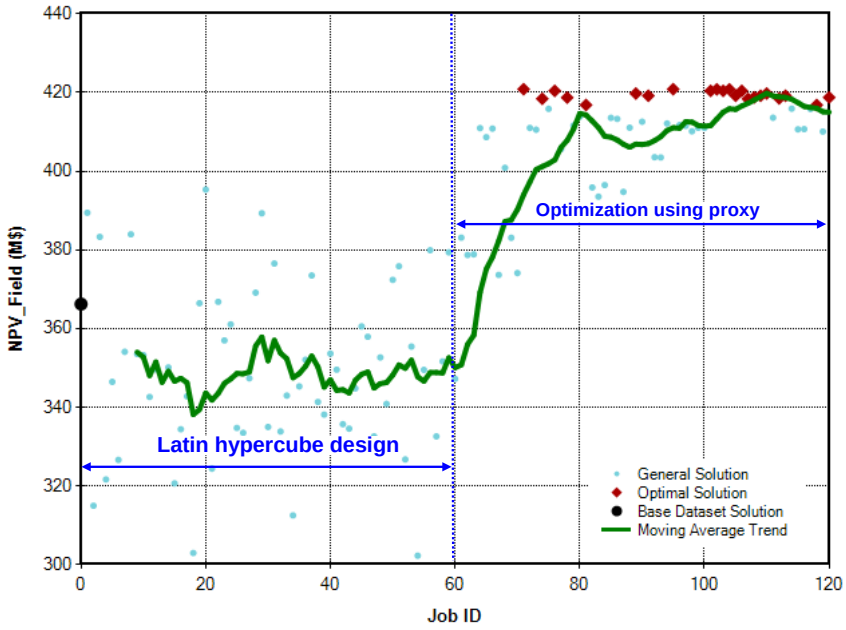
Use of this optimization algorithm involves the following four steps:

1. Latin Hypercube Design: The purpose of Latin hypercube design is to construct combinations of the input parameter values so that the maximum information can be obtained from the minimum number of simulation runs. Latin hypercube design is chosen here because it can handle any number of input parameters with mixed levels. See Latin Hypercube Design for further information.
2. Proxy Modeling: In this step, an empirical proxy model is built using training data obtained from Latin hypercube design runs. The proxy model options available are polynomial regression model and RBF (radial basis function) neural network. Polynomial regression models have been widely used for the analysis of physical and computer experiments due to their ease of understanding, flexibility, and computational efficiency. The cost of the RBF is normally significantly higher than the cost of the polynomial regression estimate; however, it is still orders of magnitude faster than actual simulation and it may provide more accurate prediction than polynomial models. Refer to Proxy Modeling for further information.
3. Proxy-based Optimization: Due to the intrinsic limitations of a proxy model, it is generally recognized that they usually cannot give accurate predictions for highly nonlinear multidimensional problems. Therefore, the optimal solution obtained based on the proxy model may not be the true optimal for the actual reservoir model. This means that certain suboptimal solutions of the proxy model may become the true optimal solution for the actual reservoir model. To counteract false optimum predictions, a pre-defined number of possible optimum solutions (i.e., suboptimal solutions of the proxy model) are generated to increase the chance of finding the global optimum solution.
4. Validation and Iteration: For each possible optimum solution found through proxy optimization, a reservoir simulation needs to be conducted to obtain the true objective function value. To further improve the prediction accuracy of the proxy model, the validated solutions can be added to the initial training data set. The updated training data set can then be used to build a new proxy model. With the new proxy model, a new set of possible optimum solutions can be obtained. This iterative procedure can be continued for a given number of iterations or until a satisfactory optimal solution is found.

The following figure illustrates the workflow of the Latin hypercube plus proxy optimization algorithm:



One unique characteristic of Latin Hypercube plus Proxy optimization is a jump in the solution quality after the Latin hypercube design is finished and proxy optimization starts. For example, as shown in the following figure, after the initial 60 Latin Hypercube design runs, the global optimum solution is quickly found within two iterations of proxy optimization (there are 10 experiments in each iteration).



12.5.3 Particle Swarm Optimization

Particle swarm optimization (PSO) is a population-based stochastic optimization technique developed by James Kennedy and Russell C. Eberhart in 1995, inspired by social behavior of bird flocking and fish schooling.

Social influence and social learning enable a person to maintain cognitive consistency. People solve problems by talking with other people about them and, as they interact, their beliefs, attitudes, and behaviors change. The changes can be depicted as the individuals moving toward one another in a sociocognitive space.

Particle swarm simulates this kind of social optimization. The system is initialized with a population of random solutions and searches for optima by updating generations. The individuals iteratively evaluate their candidate solutions and remember the location of their best success so far, making this information available to their neighbors. They are also able to see where their neighbors have had success. Movements through the search space are guided by these successes, with the population usually converging towards good solutions.

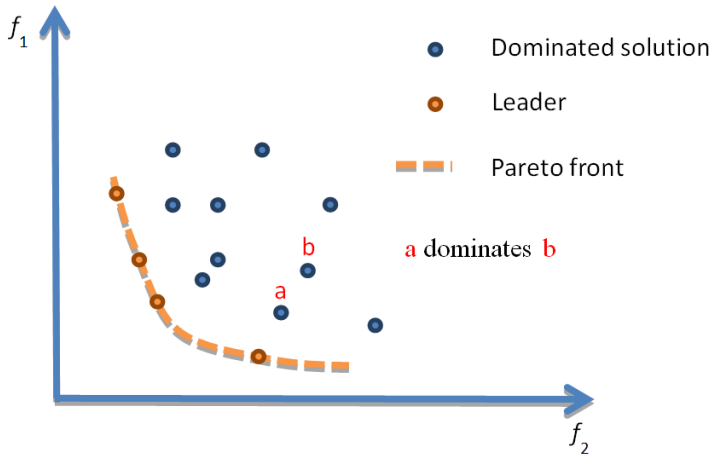
For information about configuring a CMOST particle swarm optimization, refer to Particle Swarm Optimization (PSO).

12.5.4 Pareto Front Particle Swarm Optimization

Pareto optimization aims to find the Pareto front, which consists of multiple optimal "trade-offs" of local objective functions. Consider a bi-objective minimization problem, for example. As shown in the following figure, there are two objective function values for each solution. The target is to minimize both f1 and f2 simultaneously.

To explain the concept of Pareto optimization, the following terms are defined.

- **Domination**: When two solutions, for example *a* and *b*, are compared, if all the objective values of *a* are "better" (smaller) than those of *b*, then solution *a* dominates solution *b*.
- **Leader/Non-dominated solution**: If a solution is not dominated by any other solution, it is called a non-dominated solution or leader.
- **Pareto front**: The ensemble of leaders' objectives is termed the Pareto front.



The Multiple Objective Particle Swarm Optimizer (MO-PSO) proposed by (Coello, Pulido and Lechuga 2004) is used in CMOST for Pareto front optimization. An external archive is used to store the best solutions (leaders) that the swarm has discovered. The particle updating process is very similar to that of Particle Swarm Optimization (PSO). The only difference is that a chosen leader is used to update particle velocity instead of the swarm's global best position. When selecting the leader, the one with the highest crowding distance is given the highest selection priority. Crowding distance is calculated based on the relative location of the leaders in the search space (Raquel and Naval Jr. 2005).

Once all the particles' locations are updated, the objective function values are evaluated by conducting reservoir simulation. Once all the particles are evaluated, the history of each particle is updated to get the local best position. The external archive is then updated by inserting new non-dominated leaders to the archive and deleting the dominated old leaders. The particle positions are updated, and the iteration is stopped once the preset stop criteria are met.

For information about configuring a CMOST Pareto Front particle swarm optimization, refer to Pareto Front PSO Engine Settings.

12.5.5 Differential Evolution

Differential Evolution (DE) was introduced by (Storn and Price 1995) to address optimization problems. There are four steps in the DE mechanism, as described below:

The system is initialized with a population of random solutions. It then searches for optima by updating this population. The mutation process involves adding a scaled difference of two solutions, using factor F, to the best solution in each population, to generate a new population. The crossover operation uses factor Cr to increase the newly-generated-population diversity. Finally, a selection operator is applied to preserve the optimal solutions for the next generation.

For information about configuring a CMOST differential evolution, refer to Differential Evolution (DE).

12.5.6 Random Brute Force Search

The brute force search method is a straightforward optimization method that evaluates all possible solutions and decides afterwards which one is the best. It is feasible only for small problems in terms of the dimensionality of the search space, since CMOST requires that the search space (the number of all possible parameter value combinations) be less than 65536.

To address the limitations of the brute force search, CMOST has implemented random search methods for optimization, based on exploring the domain in a random manner to find optimum solutions. These are the simplest methods of stochastic optimization and can be quite effective in some problems (small search space and fast-running simulation jobs). There are many different algorithms for random search such as blind random search, localized random search, and enhanced localized random search. The algorithm implemented in CMOST is blind random search. This is the simplest random search method, where the current sampling does not take into account the previous samples. That is, this blind search approach does not adapt the current sampling strategy to information that has been garnered in the search process. One advantage of blind random search is that it is guaranteed to converge to the optimum solution as the number of function evaluations (simulations) gets large. Realistically, however, this convergence feature may have limited use in practice since the algorithm may take a prohibitively large number of function evaluations (simulations) to reach the optimum solution.

For information about random brute force search configuration settings, refer to Random Brute Force Search.

13 Glossary

The following terms are:

- CMOST terms, or terms that have specific meaning within the CMOST context.
- Terms needed to describe the application and use of CMOST.

Term	Definition
Base 3tp File	File created by Results 3D using the base IRF. The base 3tp file is used by CMOST as the basis for displaying plots in Results 3D.
Base Dataset	A valid dataset for any CMG simulator that is used as the basis for a CMOST study. The master dataset is derived from the base dataset.
Base IRF	Simulation results file which uses default parameter values.
Base Session File	File created by Results Graph using the base IRF. The base session file is used by CMOST as the basis for displaying plots in Results Graph.
Box-Behnken Design	A set of experiments designed to have more runs at the middle values of the input parameters.
Brute Force Search	History Matching and Optimization method in which all combinations of parameter values are tested.
Candidate Values List	List of values that will be substituted for a discrete type parameter in a Master Dataset.
Central Composite Design	A set of experiments with runs which are evenly distributed at low, middle, and high values of the input parameters.
Characteristic Date Time	Dates used in the calculation of an objective function. Characteristic date times include: • Built-in fixed date times, the simulation start and end date times derived from the SR2 files. • Fixed date times, entered by users. • Dynamic date times from original time series, such as the date the value of an original time series exceeds a certain quantity. • Dynamic date times from user-defined time series.

Term	Definition
CMG DECE	See DECE.
CMM Editor	Tool for viewing, navigating, and editing the CMOST Master Dataset (.cmm) and related include files (.inc) files. For further information refer to CMM File Editor.
CMM File	See Master Dataset.
CMOST	CMG's sensitivity assessment (SA), history matching (HM), optimization (OP), and uncertainty assessment (UA) tool.
CMR File	Results file from earlier versions of CMOST. Refer to Converting old CMOST Files to new CMOST Files for information about converting CMR files to new CMOST project and study files.
CMT File	Task file from earlier versions of CMOST. Refer to Converting old CMOST Files to new CMOST Files for information about converting CMT files to new CMOST project and study files.
Coalescence	CMOST technique for combining, or coalescing lines, in time-series plots to improve display performance. Refer to Coalescence for further information.
Constraint	In History Matching and Optimization, used to prevent unnecessary simulation runs and to allow users to change Objective Function values when constraints are violated. For further information, refer to Hard Constraint and Soft Constraint.
Cross Plot	XY plots which are used to identify trends and relationships. The axes can, for example, be parameters or objective functions. For further information, see Parameter Cross Plots and Objective Function Cross Plots.
Date Time	Refer to Characteristic Date Time for information about date times used in CMOST.
DE (Differential Evolution)	History matching and optimization method in which the run is initialized with a population of random solutions or pre-defined known ones. DE attempts to find parameter values in an intelligent manner to get optimal solutions. Refer to Differential Evolution (DE) for more information.
DECE (Designed Exploration Controlled Evolution)	CMG-proprietary History Matching and Optimization method. For further information, see CMG DECE.
Dictionary File	Text file that contains a repository for descriptions (name, dimensions, data range, and so on) of simulation data items in the SR2 files.
Dynamic Date Time	A date time from an original or a user-defined time series, on which

Term	Definition
	a condition is met for the first or the last time; for example, the first date time at which a property reaches a critical value.
Experiment	A CMOST experiment is defined by a unique set of input parameters and objective functions.
Experimental Design	Definition of a set of experiments, optimally selected to obtain information about a response.
FHF (Field History File)	A text file containing reservoir production or injection data for one or more wells. Field History Files are required only for History Matching.
Fixed Date Simulation Results Observer	A Results Observer that collects data at one point in time for each simulation.
Fixed Date Times	User-defined fixed date, for example, *YearEnd2011*, used in the determination of an objective function, such as the cumulative oil produced by the end of 2011. Refer to Characteristic Date Times for information about specifying fixed data times.
Fluid Contact Depth Series	If the SR2 files contain fluid saturation data, CMOST can calculate gas-oil, water-oil, and water-gas contact depths at well locations. These depths are calculated for each time step that fluid saturation data is available. These depths can then be used as time series data for history matching.
Formula	Equation entered in a Master Dataset to perform calculations on values during a CMOST run.
Fractional Factorial Design	In classic experimental design, a sampling method in which a subset of the samples determined from a full factorial design is chosen to determine information about the important aspects of a study.
Full Factorial Design	Study with experiments that take on all possible combinations of parameter values at two or three selected levels.
Fundamental Data	Data (time, time series, distance vs. depth, and fluid contacts) that is obtained or calculated directly from SR2 files.
Hard Constraint	If a hard constraint is violated, the simulation run will not take place as these constraints are checked by the CMOST engine prior to the start of the run. See Hard Constraints for information about specifying hard constraints.
Histogram	A graph with values or ranges of values on the x-axis and bars, the height of which represents occurrence, in the y direction.
HM (History Matching)	CMOST analysis to match simulation results to production history.
History Matched	See Matched Model.

Term	Definition
Model	
History Matching Error	Percentage relative error between simulation results and production history obtained from, for example, a Field History File.
Include File	A data file (an array of porosity data, for example) that is included in a Master Dataset by reference.
Intermediate Parameter	A parameter that is entered in the Parameters table to help defined the relationships between two other parameters. For an example, refer to To add an intermediate parameter.
IRF (Indexed Results File)	Text file in the SR2 file system describing the data in the MRF (Main Results File) and how to obtain this data.
LHD (Latin Hypercube Design)	Technique for constructing combinations of input parameter values so that the maximum information can be obtained from the minimum number of simulation runs.
Latin Hypercube Plus Proxy Optimization	Latin hypercube design is used to construct experiments then an empirical proxy model is built using the training data obtained from the Latin hypercube design runs. The proxy model is then used to determine the optimal solution. See Latin Hypercube plus Proxy for further information.
Local Objective Function	A function that the user wants to minimize (history matching error, for example) or maximize (net present value, for example). Refer to Objective Function for additional information.
Master Dataset (CMM)	Version of the Base Dataset that has been modified to tell CMOST where to enter different parameter values, thereby creating a new dataset for each experiment.
Match Quality	In history matching, a measure of the match between the results of a CMOST study and a field history file. Refer to History Match Quality for further information.
Matched Model	Model produced by minimizing the history matching error. Also referred to as a history-matched model.
Monte Carlo Simulation	Simulations that involve repeated generation of outputs using randomly generated inputs which follow defined probability distributions.
Monte Carlo Simulation Using Proxy	Uncertainty Assessment method. Using Monte Carlo simulation, inputs are randomly generated from probability distributions to simulate the process of sampling from an actual population. These inputs are then fed into the response surface (proxy) model, which is used to generate outputs and determine the uncertainty in the reservoir model. See Monte Carlo Simulation Using Proxy for further information.

Term	Definition
Monte Carlo Simulation Using Reservoir Simulation	Uncertainty Assessment method. In this case, the inputs selected from the Monte Carlo simulation are run through the simulator to generate outputs and determine the uncertainty in the reservoir model. See Monte Carlo Simulation Using Simulator for further information.
Morris Method	Qualitative screening method used in sensitivity analysis to identify inputs that do not significantly affect the global objective function and to rank the inputs in order of their importance, thereby reducing computation costs. Refer to Morris Method for theoretical information, and to Sobol/Morris Analysis for information about the display and interpretation of Morris results.
MRF (Main Results File)	Main Results File, a binary file in the SR2 file system containing simulation data.
NPV (Net Present Value)	Stream of future cash flows discounted to a given date (present date or base date) to reflect the time value of money and other factors, such as investment risk.
Objective Function	An expression or quantity that the user wants to minimize or maximize. In the case of History Matching, for example, the user wants to minimize the error between field data and simulation results. In the case of Optimization, the user may want to maximize net present value.
Observer	See Results Observer.
OPAAT (One Parameter At A Time Sampling)	Traditional method for performing Sensitivity Analysis studies, in which information about the effect of a parameter is determined by varying only that parameter. The procedure is repeated, in turn, for all parameters to be studied. Refer to One-Parameter-at-a-Time Sampling for more information.
OP (Optimization)	Identification of an optimal field development plan, and operating conditions that will produce either a maximum or minimum value for objective functions that the user has specified, in particular the global objective function (GOF, for example, the net present value, or NPV) and subsidiary GOF's, which reflect the influence of selected operating parameters.
Optimal Model	Model determined from an optimization/history matching study.
Optimizer	Algorithm used to find the optimal solution for history matching or optimization studies. In the case of CMOST, these algorithms include CMG DECE, Latin Hypercube Plus Proxy Optimization, Differential Evolution, Particle Swarm Optimization, and Random Brute Force Search.

Term	Definition
Origin	Source of simulation data, for example, a well or a field.
Original Time Series	Time series data obtained directly from a simulator SR2 files.
Orthogonality	In CMOST, the orthogonality of an experiment design is measured by the maximum pair-wise correlation of the columns of a design matrix. Refer to the information about Orthogonality.
.out File	A text file that echoes the contents of the .dat file, and also includes simulation results. Users are able to read this file.
Parameter	Depending on the experimental design, values are substituted for parameters in the Master Dataset, either from a Candidate Values List or a formula.
Pareto Front Particle Swarm Optimization	A particle swarm optimization algorithm using Pareto analysis techniques to handle multiple objective function optimizations. Refer to Pareto Front PSO and Pareto Front PSO Engine Settings for further information.
PSO (Particle Swarm Optimization)	History Matching and Optimization method in which the run is initialized with a population of random solutions. Navigation through the search space is guided by the best success so far, which usually results in a convergence towards the best solution. Refer to Particle Swarm Optimization for more information.
Plackett-Burman Design	A screening method in which the resulting number of experiments is a multiple of four. This method can be used when you have a large number of potential factors and you want to quickly determine those that will most affect the objective function.
Pre-simulation Commands	Commands that are used to modify the experiment dataset before it is submitted to a simulator; for example, users may want to adjust variogram parameters in History Matching.
Prior Probability Distribution Function	The probability distribution of the input parameter values. The information is used to formulate the parameter values, so that their distribution reflects reality.
Project	A collection of studies defined for the purpose of characterizing the performance of, for example, a field, sector, group, or even a single well. CMOST projects consist of one or more studies, each of which consists of one or more experiments.
Property vs. Distance Series	Type of data series, such as saturation vs. distance along the well, which is retrieved from the SR2 files for one instant in time. Used for history matching, property vs. distance series data can be compared with data obtained from one or more well log files.
Proxy Dashboard	Through the Proxy Dashboard, you can immediately start to inspect and assess the effects of varying parameter values on time series

Term	Definition
	while the study is running. Refer to Proxy Dashboard for further information.
Proxy Model	An empirical model built using data obtained from simulation runs. The proxy model will typically run several orders of magnitude faster than actual simulations. Refer to Proxy Modeling for more information.
Proxy-based Optimization	Optimization method, in which a predefined number of possible optimal solutions, obtained from the proxy model, is run through the simulator to obtain the true optimal solution. See Proxy-based Optimization.
Radial Basis Function (RBF) Neural Network	Feed-forward neural network with a single hidden layer, which can be used by CMOST to produce time series and objective function proxy models. See Radial Basis Function (RBF) Neural Network for a summary of RBF neural network theory.
Random Brute Force Search	History Matching and Optimization method in which all combinations of parameter values are tested, with the starting point and path through the parameter values different for each run. See Random Brute Force Search for further information.
RSM (Response Surface Methodology)	For Sensitivity Analysis and Uncertainty Assessment using classical experimental design or Latin hypercube design, a response surface methodology is applied. Response surface methodology (RSM) explores the relationships between input variables (parameters) and responses (objective functions). A set of designed experiments is used to build a proxy model (approximation) of the reservoir objective function. The most common proxy models take either a linear or quadratic form. After a proxy model is built, Tornado plots displaying a sequence of parameter estimates are used to assess parameter sensitivity. Refer to Response Surface Methodology for further information.
Restart (.rst) File	This file contains the information that allows a simulation to continue from a previously halted run.
Results File (CMR)	With CMOST 2012 and earlier, results are saved to a CMR file.
Results Observers	Simulation outputs that CMOST caches in its results file. During CMOST runs, results observers display results specified by the user. As the run progresses, more and more curves or plots will appear on the plots, with the optimal runs highlighted. The user can also highlight the results of specific experiments.
R-Square (R^2)	Indicates how well a proxy model fits observed data. An R^2 of 1 occurs when there is a perfect fit (the errors are all zero). An R^2 of 0 means that the proxy model predicts the response no better than the

Term	Definition
	overall response mean.
R-Square Adjusted	Modification of R^2 that adjusts for the number of explanatory terms in a model. Unlike R^2, the adjusted R^2 increases only if the new term improves the proxy model more than would be expected by chance. The adjusted R^2 can be negative, and it will always be less than or equal to R^2.
R-Square Predicted	Indicates how well a proxy model predicts responses for new observations. Ranging between 0 and 1, larger values suggest models of greater predictive ability.
Run Configuration	Specification of the machines to which CMOST will submit jobs; for example, to the user's local machine or to a cluster of machines accessible through the network.
Sampling Method	Method by which the parameter space is sampled when performing a Sensitivity Analysis or Uncertainty Assessment. For further information, refer to Sampling Methods.
SA (Sensitivity Analysis)	Analysis carried out to determine which parameters have the greatest effect on simulation results. This information is then useful in suggesting parameters that can be eliminated from consideration in subsequent studies.
Sobol Method	Uses Monte Carlo analysis to determine a set of indices that specify the proportion of output variance that is due to individual input variables. Refer to Sobol Method for theoretical information and to Sobol/Morris Analysis for information about the display and interpretation of Sobol results.
Soft Constraint	Allows the user to override objective function values if they violate the constraint. Checking for this violation takes place while the simulation is being run. A penalty for constraint violation can also be defined. See Soft Constraints for information about specifying soft constraints.
Special Dictionary File	Dictionary file required to process SR2 files produced by a special simulator, such as the STARS-ME simulator.
SR2 Files	Group of files containing the results of a simulation run—an IRF (Indexed Results File) and an MRF (Main Results File). Results Graph and 3D use the SR2 files for post-processing of simulation output.
SR2 Processing Stack Size	Stack size (MB) used by the SR2 reader to read SR2 files. The default stack size is 40 MB.
Study	A set of parameters is varied in a defined way to assess the sensitivity of parameters on objective functions (SA, Sensitivity

Term	Definition
	Analysis), to match simulator outputs with a history file (HM, History Match), to optimize the value of objective functions by varying operating conditions (OP, Optimization), or to assess the variation of an objective function due to uncertainty in the value of a reservoir parameter (UA, Uncertainty Assessment).
Study File (.cms)	A file that contains all of the configuration data needed to run a CMOST study.
Study Folder (.cmsd)	Folder that contains all of the study .dat, SR2, .log, and .vdr files. The retention of .dat, .log and SR2 files is as specified by the user in the Job Record and File Management area of the Simulation Settings page.
Time Series Simulation Results Observer	Results observer which collects and plots data that changes with time, such as rate and pressure for all times during the simulation runs.
Tornado Plots	A tornado plot is produced for each objective function. Parameters are ordered vertically, from those that have the greatest effect on the objective function (longest bar) to those that have the least effect (shortest bar). The effect is a graph that looks like a tornado.
Training Data	Data used to build a proxy model by analyzing the relationship between input parameters and output objective functions.
Three-level Classical Experimental Design	Experiments take on all possible combinations of three values or "levels" for each input parameter; i.e., a low, median, and high level.
Two-level Classical Experimental Design	Experiments take on all possible combinations of two values or "levels" for each input parameter; i.e., a low and high level.
UA (Uncertainty Assessment)	Analysis carried out to determine the likely variation in simulation results due to uncertainty, in particular, of reservoir variables.
User-defined Time Series	Time series that is not directly available from the SR2 files, but which can be derived from available SR2 data. Refer to User-Defined Time Series for further information.
Variogram	The variogram describes the variance of the difference between the field values at two locations (x and y) across realizations of the field (Cressie, N., 1993, *Statistics for Spatial Data*, Wiley Interscience).
VDR (Vector Data Repository) Files	Files containing compressed simulation data from CMOST runs, which are used to calculate objective functions. The files are compressed to reduce disk space and runtime.

14 Bibliography

Campolongo, F., Cariboni, J., and A. Saltelli. "An Effective Screening Design for Sensitivity Analysis of Large Models." *Environmental Modelling and Software*, 2007.

Cioppa, T.M. "Efficient Nearly Orthogonal and Space-Filling Experimental Designs for High-Dimensional Complex Models." September 2002.

Coello, C.A.C., G.T. Pulido, and M.S. Lechuga. "Handling Multiple Objectives with Particle Swarm Optimization." *IEEE Transactions on Evolutionary Computation* 8, no. 3 (2004): 256-279.

Ekstrom, Per-Anders. "Eikos: A Simulation Toolbox for Sensitivity Analysis." *Uppsala Universitet*, 2005.

Iman, R., and W. Conover. "A Distribution-Free Approach to Inducing Rank Correlation Among Input Variables." 11, no. 3 (1982): 311-334.

McKay, M.D., R.J. Beckman, and W.J. Conover. "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code." *Technometrics* 21, no. 2 (May 1979).

Morris, M.D. "Factorial Sampling Plans for Preliminary Computational Experiments." *Technometrics* (American Statistical Association) 33, no. 2 (May 1991): 161-174.

Raquel, C. R., and P. C. Naval Jr. "An Effective Use of Crowding Distance in Multiobjective Particle Swarm Optimization." *Conference on Genetic and Evolutionary Computation.* ACM, 2005. 256-264.

Saltelli, A., et al. *Global Sensitivity Analysis: The Primer.* John Wiley & Sons, 2008.

Saltelli, A., P. Annoni, I. Azzini, F. Campolongo, M. Ratto, and S. Tarantola. "Variance-Based Sensitivity Analysis of Model Output - Design and Estimator for the Total Sensistivity Index." *Computer Physics Communications*, 2010.

Saltelli, A., S. Tarantola, F. Campolongo, and M. Ratto. *Sensitivity Analysis in Practice: A Guide to Assessing Scientific Models.* John Wiley & Sons, 2004.

Satterthwaite, F.E. "Random Balance Experimentation." *Technometrics* 1, no. 2 (May 1959).

Sobol, I. "Sensitivity Estimates for Nonlinear Mathematical Models." *Matematicheskoe Modelirovanie 2*, 1993: 112-118.

Storn, R., and K. Price. "Differential Evolution - A Simple Efficient Adaptive Scheme for Global Optimization over Continuous Spaces." International Computer Science Institute, Berkeley, CA, 1995.

Tang, B. "Orthogonal Array-Based Latin Hypercubes." *Journal of the American Statistical Association: Theory and Methods* 88, no. 424 (December 1993).

Ye, K.Q. "Orthogonal Column Latin Hypercubes and Their Application in Computer Experiments." *Journal of the American Statistical Association: Theory and Methods* 93, no. 444 (December 1998).

15 Index

D

E

F

G

H

I

J

L

M

N

O

P

R

S

T

U

V

Z