

# SAP2000 API 入门与 VBA 开发示例

筑信达 孙雪艳

SAP2000 是通用结构分析与设计软件，广泛应用于建筑结构、工业建筑、市政工程、桥梁工程、港口运输等不同行业、不同类型的结构分析与设计。SAP2000 适用的结构类型多，行业范围广泛，提供了较为通用的建模方法和结果输出，也提供了一些快速编辑模型的方法，如交互式数据库编辑、s2k 文本编辑、Excel 文件导入导出等方式。每个行业有各自的特点，在处理一些行业内问题时，SAP2000 提供的原生功能不能直接满足要求，在建模阶段需要一些重复的操作，在结果输出中需要对输出的结果进行二次处理。这时，利用 SAP2000 提供的 API 功能，可以通过编程的方式编制工具，操作 SAP2000，进行二次开发，将繁琐的手工操作转化为自动化的定制工具。API 功能可以使用不同的开发工具进行编译，满足不同工程师的需求。本文介绍 SAP2000 API 接口的功能、学习资源、简单的 VBA 开发示例，希望帮助更多编程基础薄弱的工程师入门二次开发。

## 1 功能简介与帮助文件

SAP2000 从 V11 版本开放 API (Application Programming Interface) 接口。API 接口是 SAP2000 提供的一系列接口函数，工程师通过编程方式使用 API 接口函数访问 SAP2000，控制 SAP2000 的使用。利用 API 接口函数可以实现参数化自动建模、荷载设置、流程控制、结果查询与统计等功能。API 接口函数包含以下功能：

- (1) 启动、关闭 SAP2000，模型文件的打开、保存等。
- (2) 属性定义，包括材料、截面、荷载模式、荷载组合、反应谱函数定义等
- (3) 建模，包括建立点、线、面单元、连接单元等
- (4) 编辑，包括带属性复制、拉伸、移动、单元分割合并等。
- (5) 指定，包括荷载、组、构件参数指定等。
- (6) 分析，包括指定分析选项、运行分析、获取应力、内力、位移等。
- (7) 设计，包括设置首选项、覆盖项，提取设计结果等。
- (8) 单元选择，包括按组选择、按属性选择、按区域选择等。
- (9) 视图刷新，可刷新视图显示。

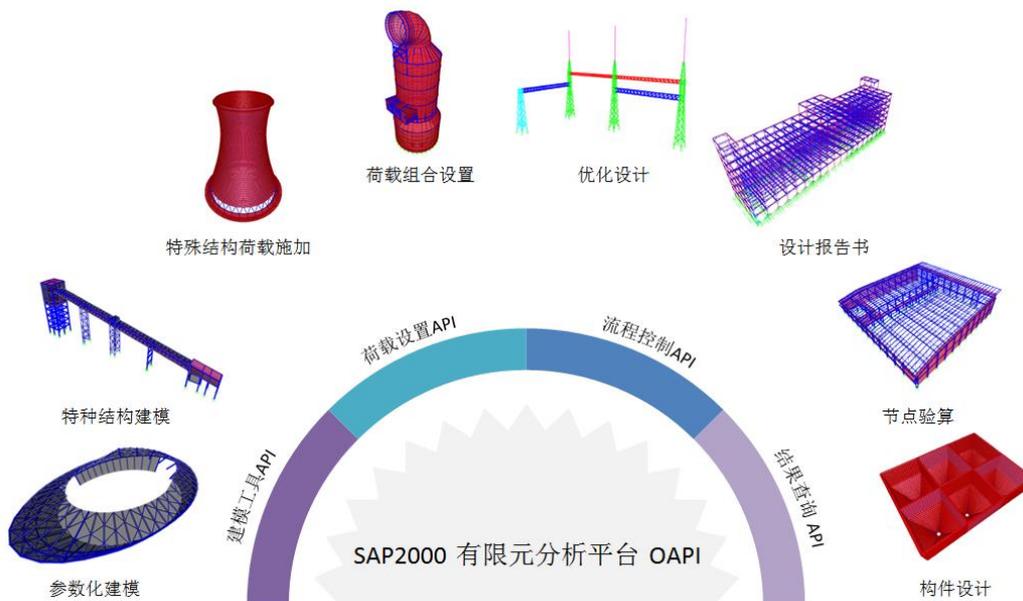


图 1 API 接口函数用途

SAP2000 安装目录下的 CSI\_OAPI\_Documentation.chm 文件是 SAP2000 提供的 API 帮助文件，如图 2 所示，包含入门指南、函数参考手册、样例代码以及版本间改变等内容，函数的具体功能查看 CSI API Functions 部分。函数按照功能类型分类，依次



为：通用函数、分析模型、编辑、文件、定义、模型、分析、设计、选择、选项、视图。通用函数用于软件的打开关闭、模型单位设置等操作；分析模型用于提取分析模型的结果；对象模型用于进行对象操作，其他功能与 SAP2000 菜单中的功能对应，可按照使用功能查找相应的函数。函数的层级见图 3，关于模型操作的函数主要是在 SapModel 中。对每一个具体的函数，给出了函数的定义、函数参数说明，函数说明以及 VBA 函数示例。样例代码部分提供了多种编译环境的示例代码，包括 VBA、VB、C#、Intel Visual Fortran、C++、MATLAB、Python 编译环境。

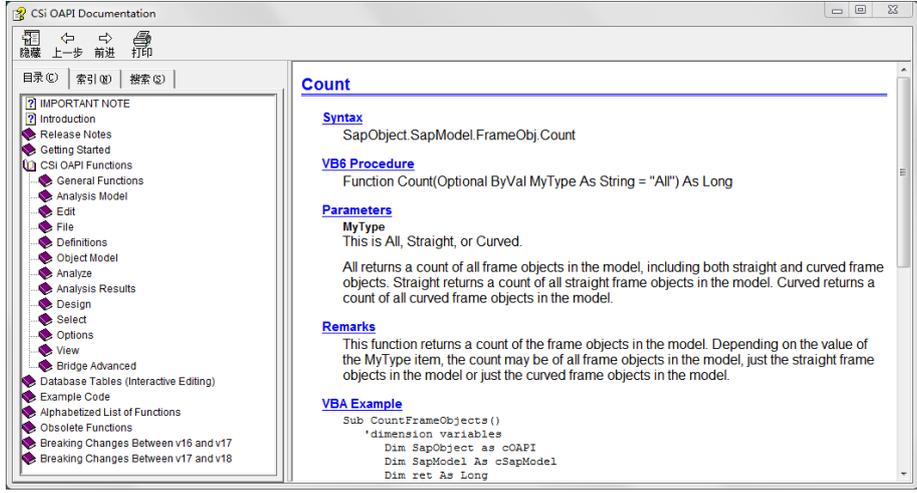


图 2 CSI\_OAPI\_Documentation 文档内容

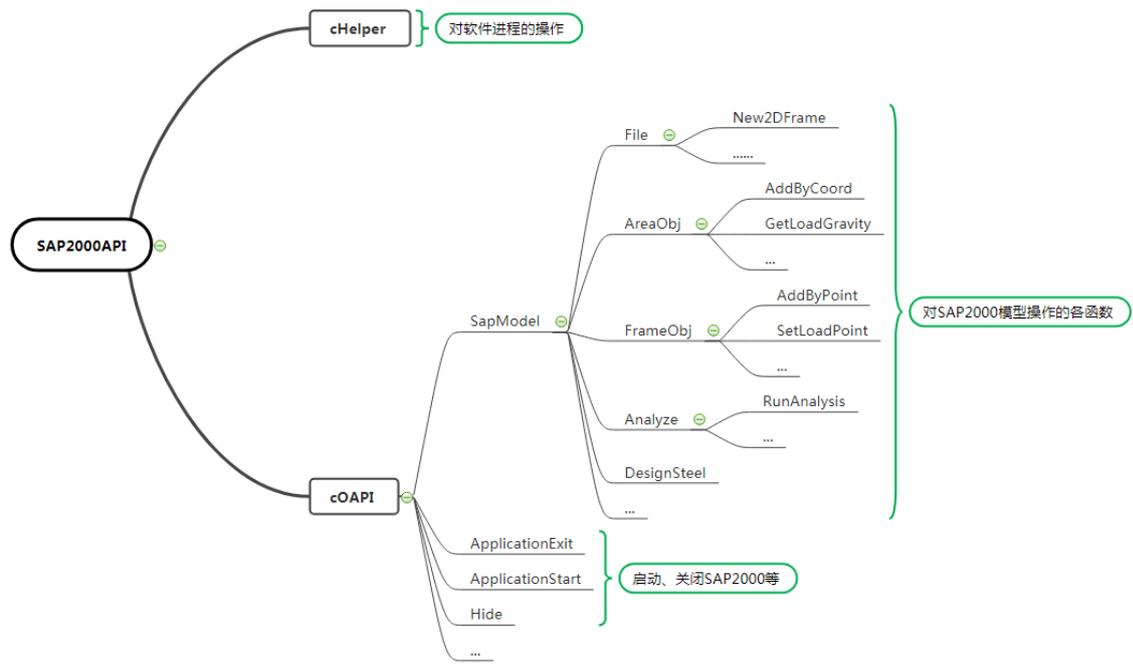


图 3 API 函数层级图

## 2 开发环境与开发方式

SAP2000 API 开发支持多种开发环境，可以使用 VBA、VB、C++、C#、Intel Visual Fortran、Python、MatLab 等进行开发。VBA 集成于 Office 的各应用程序中，如 Word、Excel 等，应用方便，不需要单独安装编译环境，其它几种开发环境均需安装单独的开发环境，并需要对应用具有一定了解，可以根据情况选择适合的开发环境。

SAP2000 API 调用方式有两种，外部调用和内部调用。外部调用是指：写一个调用 API 的程序，该程序自动启动关联 SAP2000



程序，控制软件进行建模、分析、结果提取等操作。内部调用是指：写一个功能插件，此插件是一个动态链接库（.dll 文件），将插件在 SAP2000 的工具栏加载，进行模型操作时，直接使用在工具栏中加载的插件，和 SAP2000 自有功能的使用方式相同。以下是 SAP2000 API 两种调用方式的实例。

筑信达工具箱（CiSApps）是基于 SAP2000 开发的工具集，安装完成后，在 SAP2000 “工具”菜单下点击筑信达工具箱命令，打开筑信达工具箱，即对当前模型使用筑信达工具箱中提供的各项功能，如图 4 所示。筑信达工具箱根据使用性质分为四种类型：建模工具、统计查询、设计校核、模型转化。该工具箱采用内部调用的方式，集成在 SAP2000 中，工程师可随时调用。

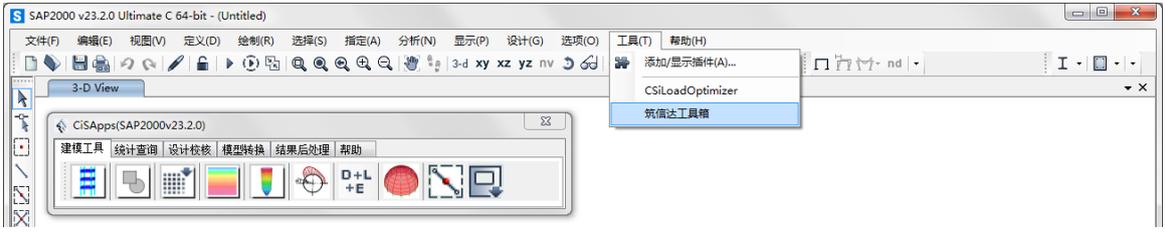


图 4 筑信达工具箱

变电站构架设计软件 CISGTCAD 是一款基于 SAP2000 API 开发的面向特定行业的一体化设计软件，采用参数化方式建立模型，通过 SAP2000 API 功能与 SAP2000 无缝对接，采用构架设计规范对构件进行校验，快速准确的进行变电站构架的分析设计与施工图设计，形成变电站构架建模、分析、设计、施工图一体化的设计系统。CISGTCAD 中的分析设计模块，是根据建立的参数化模型自动调用 SAP2000 进行分析设计，采用外部调用的方式，整个过程中不需要进行交互，自动完成。

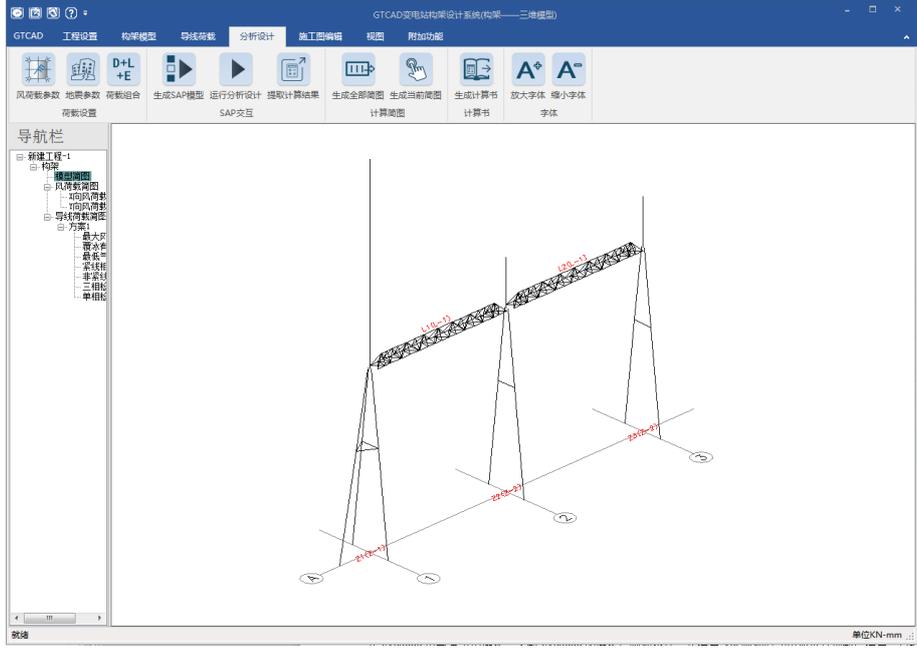


图 5 变电站构架设计软件 CISGTCAD 分析与设计模块菜单

### 3 函数使用

API 接口提供的函数同 SAP2000 中的操作相对应，如果熟悉 SAP2000 的操作，则使用 API 接口函数进行二次开发，就是将 SAP2000 的操作过程代码化，形成可应用的工具。以给杆件施加集中荷载为例来介绍 API 函数参数与实际操作的对应关系。

在 SAP2000 中对杆件指定集中荷载的操作顺序为：选择杆件，选择菜单【指定】->【框架荷载】->【集中荷载】，弹出指定集中荷载对话框，设置荷载参数，如图 6 所示，荷载模型为 DEAD，坐标系为 GLOBAL，荷载方向为 Gravity（重力方向），荷载类型为 Force（力），在相对位置中输入 0.5，下方对应的编辑框中输入 2000，选项中为“替换现有荷载”，勾选距离 1 端的相对距离，表示在杆件的中间位置指定方向向下的集中荷载，荷载值为 2kN，显示杆件的荷载信息如图 7 所示。



图 6 指定集中荷载对话框

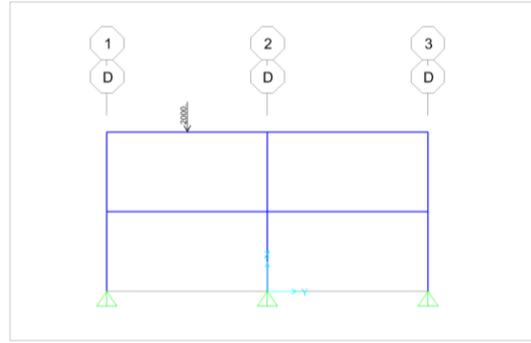


图 7 显示杆件的荷载

上述杆件指定集中荷载操作使用 API 来完成，使用函数为 `mySapModel.FrameObj.SetLoadPoint("15", "DEAD", 1, 10, 0.5, 2)`。`SetLoadPoint` 函数的具体定义见图 8 中的函数定义。每个参数的含义与 SAP2000 操作的对应可以参见图 8。函数中有多个参数，以逗号隔开，每个参数声明方式为 `ByValue Name As String`，`ByVal` 是参数的传递类型，`Name` 是参数名，`As` 后的 `String` 表示变量类型。参数的传递类型有 2 种，`ByVal` 表示此变量以值传递，`ByRef` 表示此变量以地址传递，对于需要返回数据的参数使用此种类型，`Optional ByVal` 的含义是指当前参数可选，不传递则用默认值代替。`SetLoadPoint` 函数有 10 个参数，而在上面给出的函数使用了 6 个参数，后面四个参数是 `Optional ByVal` 类型，未输入，使用默认值。不同的参数类型不同，如 `Name` 为 `String`（字符串）类型，`Val` 为 `double`（双精度浮点型）类型，`Replace` 为 `Boolean`（布尔）类型。函数定义、参数说明以及函数示例均使用 VBA 语法编写，如使用 VBA 进行 API 的开发，可以按照帮助文件中的说明进行变量的定义和函数的使用。在不同的编程软件中，变量类型、函数使用方法等有区别，如使用其他类型的开发环境，要注意变量的定义与函数的使用。

**SetLoadPoint**

**Syntax 函数语法**  
`SapObject.SapModel.FrameObj.SetLoadPoint`

**VBS Procedure 函数定义**  
 Function SetLoadPoint(ByVal Name As String, ByVal LoadPat As String, ByVal MyType As Long, ByVal Dir As Long, ByVal Dist As Double, ByVal Val As Double, Optional ByVal CSys As String = "Global", Optional ByVal RelDist As Boolean = True, Optional ByVal Replace As Boolean = True, Optional ByVal RemType As eItem = Object) As Long

**参数说明**

**Name** 为杆件或组名称，对应SAP2000中选择的构件  
 The name of an existing frame object or group, depending on the value of the ItemType item.

**LoadPat** 为荷载模式，对应窗口中的荷载模式项  
 The name of a defined load pattern.

**MyType** 为荷载类型，对应窗口中的荷载类型项  
 This is 1 or 2, indicating the type of point load.  
 1 = Force  
 2 = Moment

**Dir** 为荷载方向，对应窗口中的荷载方向  
 This is an integer between 1 and 11, indicating the direction of the load.  
 1 = Local 1 axis (only applies when CSys is Local)  
 2 = Local 2 axis (only applies when CSys is Local)  
 3 = Local 3 axis (only applies when CSys is Local)  
 4 = X direction (does not apply when CSys is Local)  
 5 = Y direction (does not apply when CSys is Local)  
 6 = Z direction (does not apply when CSys is Local)  
 7 = Projected X direction (does not apply when CSys is Local)  
 8 = Projected Y direction (does not apply when CSys is Local)  
 9 = Projected Z direction (does not apply when CSys is Local)  
 10 = Gravity direction (only applies when CSys is Global)  
 11 = Projected Gravity direction (only applies when CSys is Global)

The positive gravity direction (see Dir = 10 and 11) is in the negative Global Z direction.

**Dist**  
 This is the distance from the I-End of the frame object to the load location. This may be a relative distance (0 <= Dist <= 1) or an actual distance, depending on the value of the RelDist item. [L] when RelDist is False

**Val** 为荷载位置距离边缘的长度，可以为相对距离或绝对距离，对应窗口中的相对距离一行  
 This is the value of the point load. [F] when MyType is 1 and [FL] when MyType is 2 Val为荷载值，对应窗口中的荷载值

**CSys** 为坐标系，对应为窗口中的坐标系选项  
 This is Local or the name of a defined coordinate system. It is the coordinate system in which the loads are specified.

**RelDist** 为是否为相对距离，对应窗口中的相对距离绝对距离勾选项  
 If this item is True, the specified Dist item is a relative distance, otherwise it is an actual distance.

**Replace** 为是否替换现有荷载，对应窗口中的选项，叠加现有荷载和替换现有荷载  
 If this item is True, all previous loads, if any, assigned to the specified frame object(s), in the specified load pattern, are deleted before making the new assignment.

**ItemType** 为枚举类型，可以对杆件、组或者选择构件进行操作，对应Sap2000中的选择构件操作  
 This is one of the following items in the eItem enumeration:

图 8 函数定义

从以上例子可以看出，API 函数的使用和在 SAP2000 操作使用相似，熟悉 SAP2000 的操作能更好的理解函数并应用。在使用函数时也要注意，不是所有的函数都与 SAP2000 中的操作相同，如杆件上的集中荷载删除和荷载指定使用同一个窗口，而在 API 接口中，杆件上的集中荷载删除使用函数 `mySapModel.FrameObj.DeleteLoadPoint`。

## 4 VBA 开发示例

本文以一个建模和读取结果工具为例介绍 VBA 的开发。在 Excel 中的开发工具中打开 VBA，添加对 SAP2000 Application Programming Interface (API) 的引用，如图 9 所示，如在引用窗口中找不到 SAP2000 Application Programming Interface (API)，使用浏览功能，找到 SAP2000 安装目录下的 SAP2000v1.tlb 文件，手动添加 AP2000 Application Programming Interface (API)。点击 ThisWorkBook，输入如图 10 所示代码，运行后，自动打开 SAP2000，建立如图 11 所示的二维框架，自动运行分析，分析完成后提取 DEAD 工况的节点位移结果，显示在 Excel 的表格中，如图 12 所示。

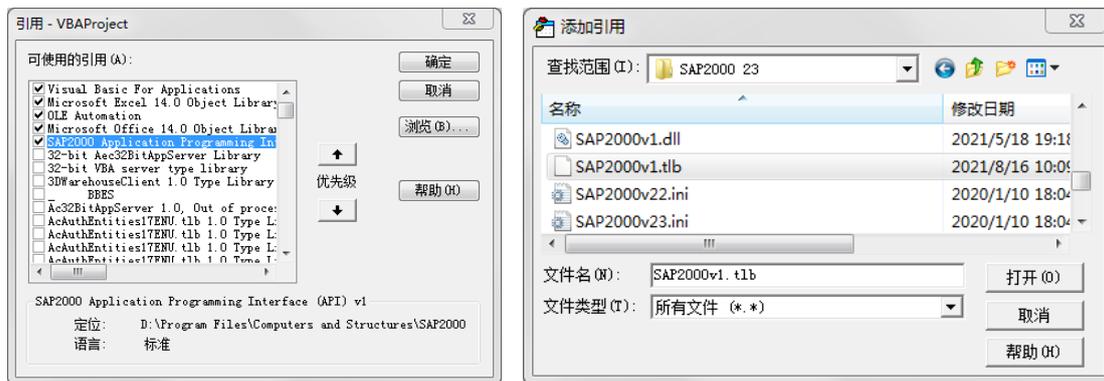


图 9 添加引用窗口

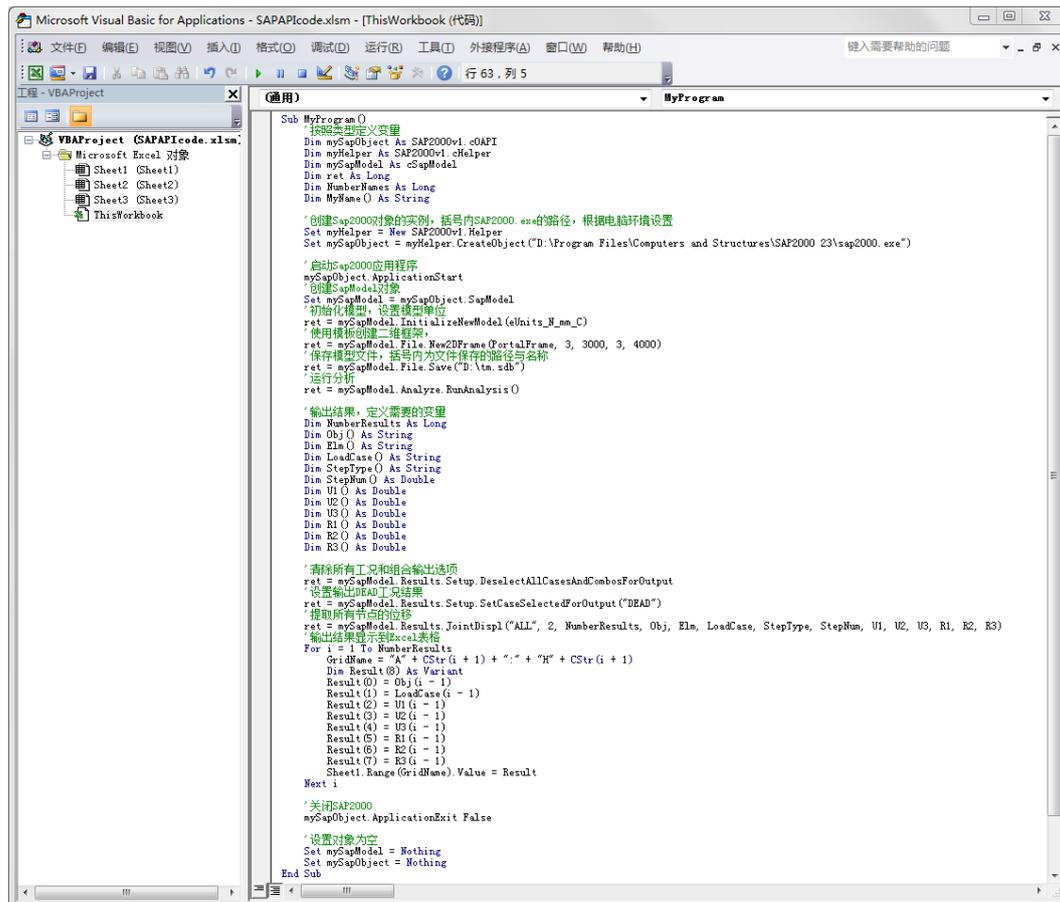


图 10 示例代码

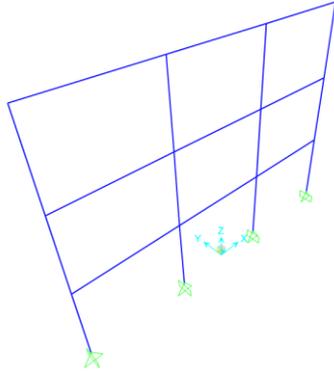


图 11 二维框架图

编号	荷载工况	U1	U2	U3	R1	R2	R3
1	DEAD	0	0	0	0	-4.8E-06	0
2	DEAD	-0.00061	0	-0.0149	0	9.96E-06	0
3	DEAD	-0.00028	0	-0.02428	0	6.68E-06	0
4	DEAD	0.001188	0	-0.02807	0	1.6E-05	0
5	DEAD	0	0	0	0	1.91E-07	0
6	DEAD	-0.0002	0	-0.02211	0	-6.3E-07	0
7	DEAD	-8.9E-05	0	-0.03628	0	6.05E-07	0
8	DEAD	0.000389	0	-0.04257	0	-8.3E-07	0
9	DEAD	0	0	0	0	-1.9E-07	0
10	DEAD	0.000204	0	-0.02211	0	6.34E-07	0
11	DEAD	8.92E-05	0	-0.03628	0	-6.1E-07	0
12	DEAD	-0.00039	0	-0.04257	0	8.27E-07	0
13	DEAD	0	0	0	0	4.85E-06	0
14	DEAD	0.000605	0	-0.0149	0	-1E-05	0
15	DEAD	0.000284	0	-0.02428	0	-6.7E-06	0
16	DEAD	-0.00119	0	-0.02807	0	-1.6E-05	0

图 12 节点位移结果

## 5 新功能介绍

从 SAP2000V23.0.0 版本开始，可以使用 DatabaseTables 类通过 API 访问数据库表。在 SAP2000 使用【显示】->【表格】，可以查看模型的各项信息，包括模型定义信息、分析结果、设计结果，如图 13 所示。模型定义中各个表数据显示模型信息，用于信息查询、建模和模型编辑，分析结果和设计结果各表显示结构的各项分析结果和设计结果，用于结果的读取与处理。有关数据库表操作的详细说明，请参阅 CSI\_OAPI\_Documentation.chm 中的 DatabaseTables 部分内容，如图 14 所示。DatabaseTable 数据表可以读取表格结果，也可以给表格赋值，进行模型的编辑。

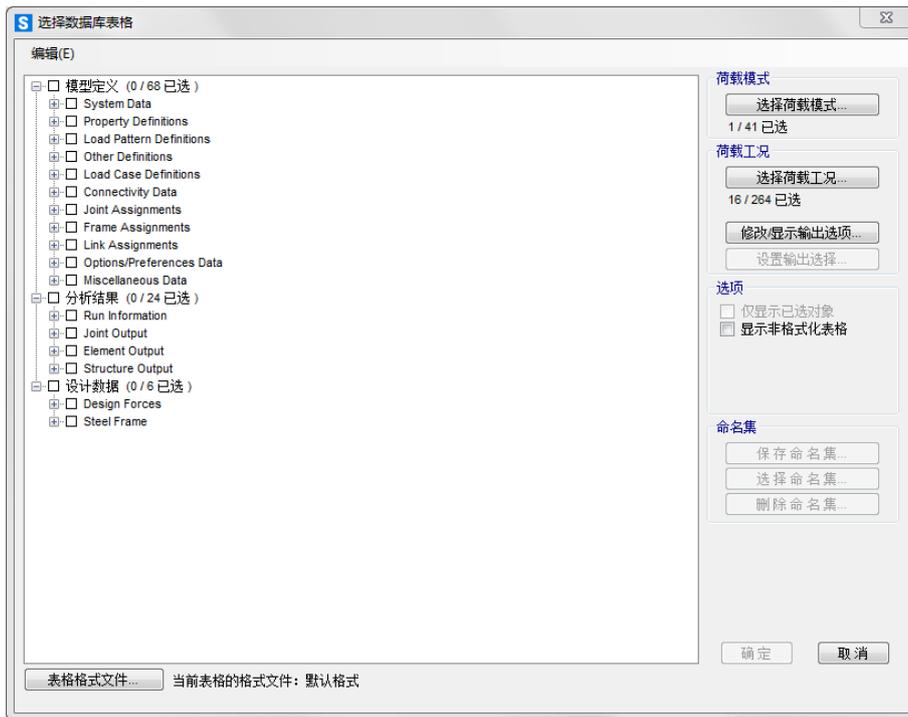


图 13 显示表

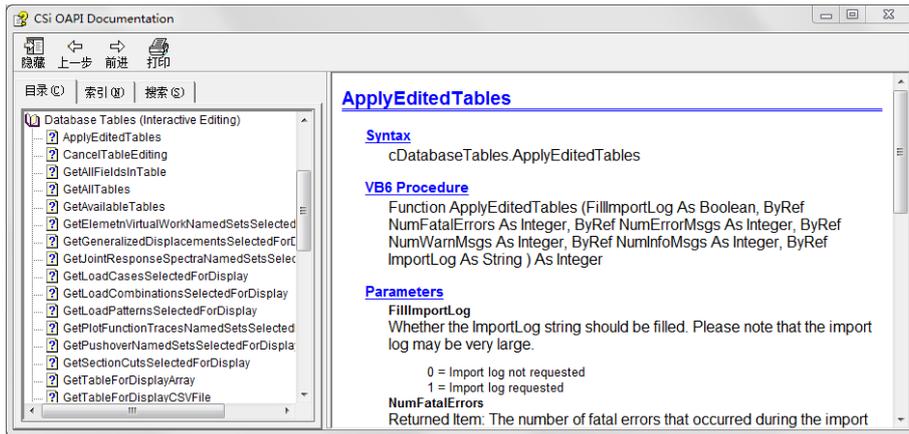


图 14 数据表函数

开发示例中的提取 DEAD 工况的结果，如使用 DatabaseTables 数据表的方式获得，可以使用图 15 中“采用数据表方式输出结果”部分的代码，选择要输出的表格为 Joint Displacements，设置输出 DEAD 工况，使用 GetAllFieldsInTable 函数获得节点位移表的表头，使用 GetTableForDisplayArray 函数获得结果，得到的结果存储在 TableData 数组中。TableData 数据是一维数组，是将所有的数据逐行放置在数组中，提取数据时要对 TableData 进行处理。

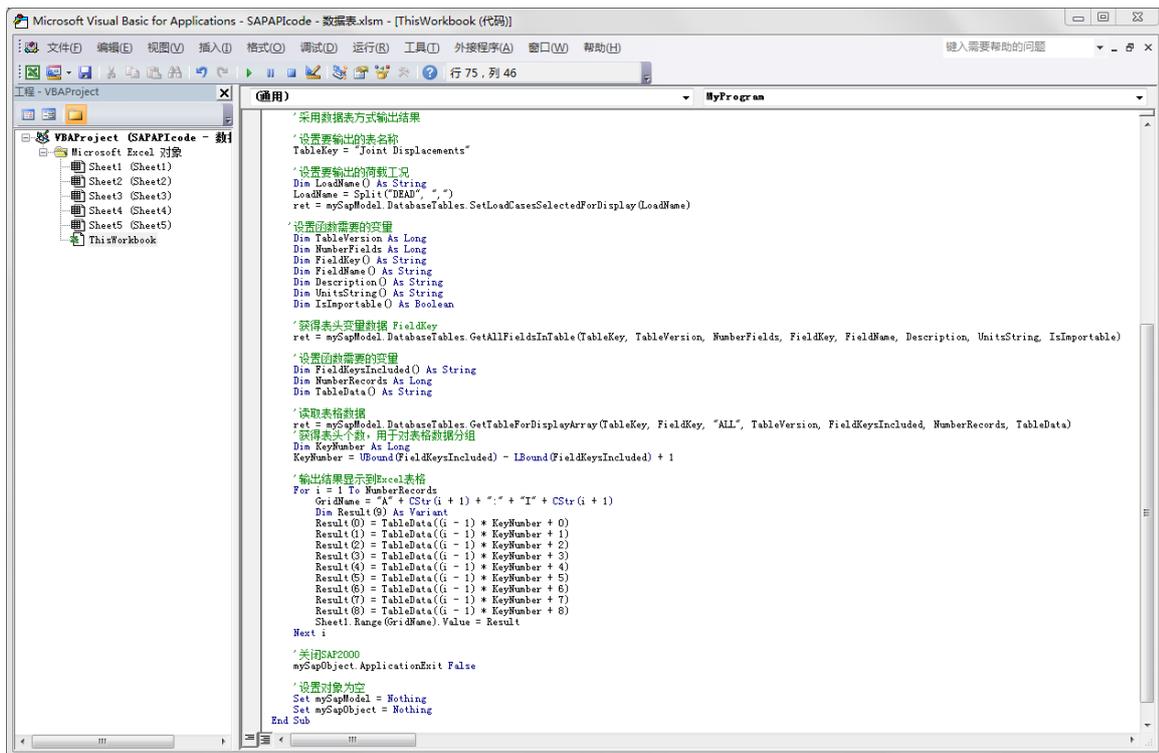


图 15 DatabaseTable 示例代码

示例中输出表格结果时，将所有的数据均输出到 Excel 表格中，如图 16 所示，输出的结果与 SAP2000 显示的节点位移表格数据完全一致。

编号	荷载工况	CaseType	U1	U2	U3	R1	R2	R3
1	DEAD	LinStatic	0	0	0.00E+00	0.00E+00	-4.85E-06	0
2	DEAD	LinStatic	-0.000605	0	-1.49E-02	0.00E+00	9.96E-06	0
3	DEAD	LinStatic	-0.000284	0	-2.43E-02	0.00E+00	6.68E-06	0
4	DEAD	LinStatic	0.001188	0	-2.81E-02	0.00E+00	1.60E-05	0
5	DEAD	LinStatic	0	0	0.00E+00	0.00E+00	1.91E-07	0
6	DEAD	LinStatic	-0.000204	0	-2.21E-02	0.00E+00	-6.34E-07	0
7	DEAD	LinStatic	-8.90E-05	0	-3.63E-02	0.00E+00	6.05E-07	0
8	DEAD	LinStatic	0.000389	0	-4.26E-02	0.00E+00	-8.27E-07	0
9	DEAD	LinStatic	0	0	0.00E+00	0.00E+00	-1.91E-07	0
10	DEAD	LinStatic	0.000204	0	-2.21E-02	0.00E+00	6.34E-07	0
11	DEAD	LinStatic	8.90E-05	0	-3.63E-02	0.00E+00	-6.05E-07	0
12	DEAD	LinStatic	-0.000389	0	-4.26E-02	0.00E+00	8.27E-07	0
13	DEAD	LinStatic	0	0	0.00E+00	0.00E+00	4.85E-06	0
14	DEAD	LinStatic	6.05E-04	0	-1.49E-02	0.00E+00	-9.96E-06	0
15	DEAD	LinStatic	2.84E-04	0	-2.43E-02	0.00E+00	-6.68E-06	0
16	DEAD	LinStatic	-0.001188	0.00E+00	-0.028073	0.00E+00	-1.60E-05	0

### 1) 输出结果

Joint Text	OutputCase	CaseType Text	U1 mm	U2 mm	U3 mm	R1 Radians	R2 Radians	R3 Radians
1	DEAD	LinStatic	0	0	0	0	-4.848E-06	0
2	DEAD	LinStatic	-0.000605	0	-0.014895	0	9.962E-06	0
3	DEAD	LinStatic	-0.000284	0	-0.024277	0	6.678E-06	0
4	DEAD	LinStatic	0.001188	0	-0.028073	0	1.6E-05	0
5	DEAD	LinStatic	0	0	0	0	1.908E-07	0
6	DEAD	LinStatic	-0.000204	0	-0.02211	0	-6.343E-07	0
7	DEAD	LinStatic	-8.9E-05	0	-0.036277	0	6.053E-07	0
8	DEAD	LinStatic	0.000389	0	-0.042573	0	-8.268E-07	0
9	DEAD	LinStatic	0	0	0	0	-1.908E-07	0
10	DEAD	LinStatic	0.000204	0	-0.02211	0	6.343E-07	0
11	DEAD	LinStatic	8.9E-05	0	-0.036277	0	-6.053E-07	0
12	DEAD	LinStatic	-0.000389	0	-0.042573	0	8.268E-07	0
13	DEAD	LinStatic	0	0	0	0	4.848E-06	0
14	DEAD	LinStatic	0.000605	0	-0.014895	0	-9.962E-06	0
15	DEAD	LinStatic	0.000284	0	-0.024277	0	-6.678E-06	0
16	DEAD	LinStatic	-0.001188	0	-0.028073	0	-1.6E-05	0

### 2) SAP2000 显示结果

图 16 输出结果对比

## 6 结语

SAP2000 API 通过编程来控制 SAP2000 运行，通过 SAP2000 的二次开发，可以定制专属工具，将繁琐的手工操作自动化，扩展 SAP2000 的应用。SAP2000 安装目录下的文件 CSI\_OAPI\_Documentation.chm 是 API 帮助文档，有详细的说明和示例。API 功能可以使用不同的开发工具进行编译，建议初学者使用 VBA。VBA 集成于常用的办公软件中（如 Word、Excel 等），不需要单独的安装，相对于专业的开发软件，简单易学，了解基本的语法后，就可以编写代码，如参数化建模，结果处理等。