

An introduction to using API for SIDRA INTERSECTION 8 and VOLUMES Utility

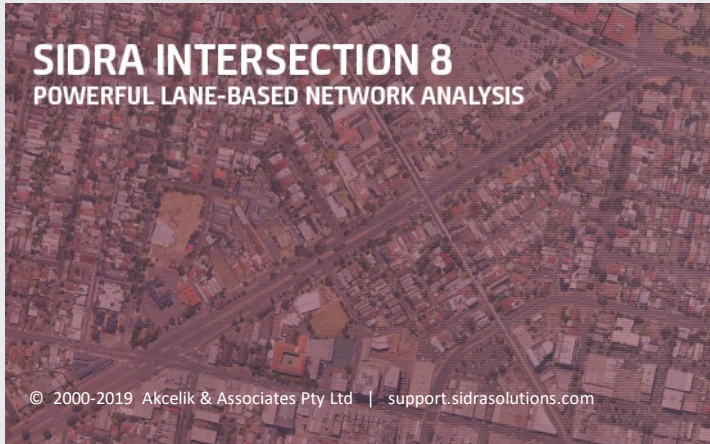
Mark Besley

**Presentation at the SIDRA USER GROUP MEETING
Melbourne, Nov 2019**

Updated: 17 Nov 2019

**SIDRA
SOLUTIONS**
CELEBRATING 20 YEARS

PRESENTATION OBJECTIVES / CONTENTS



SIDRA Application Programming Interface

- What is the SIDRA API?
- Methods of using the API.
- Resources available for users.
- Support in future versions.
- Examples and sample code.

What is the SIDRA API?

The Application Programming Interface for SIDRA INTERSECTION (**SIDRA API**) provides a programming interface that enables external applications to communicate with the software.

This happens outside the SIDRA User Interface.

We will also discuss direct access to the database in the SIDRA INTERSECTION Project File.

What Can You Do Using the API?

- Get full programmatic access to the Site and Network input and output data contained in the SIDRA INTERSECTION Project file.
- Create a Project file and add a Site / Network / Common Control Group (CCG) / Route.
- Specify geometry data (e.g. add and remove intersection legs and lanes, specify lane configuration and lane disciplines).
- Specify phasing and timing data, and other data as required.
- Process a Site or Network.

Ways of Using the API

Three Options

- Programming interface via Component Object Model (**COM**)
- Programming interface via **Microsoft .NET**
- Direct access to project file database

COM API Access

- **Component Object Model (COM)** is a binary-interface standard for software components introduced by Microsoft in 1993.
- COM provides a stable **application binary interface (ABI)** that does not change between compiler releases.
- COM would be used when programming in languages such as **C, C++, Python, Visual Basic**.
- A licensed copy of SIDRA INTERSECTION must be installed on the computer in order to use the COM interface.

.NET API Access

- Microsoft introduced the **.NET Framework** in 2002 with the idea of language interoperability and to simplify programming tasks such as memory management and version control of software modules.
- Many .NET languages but **C#** is by far the most popular.
- To overcome issues with version control when SIDRA updates are released, a **Late Binding DLL** must be used. This is available from our website.
- A licensed copy of SIDRA INTERSECTION must be installed on the computer in order to use the .NET interface.

Direct Access to SIPx File Database (1)

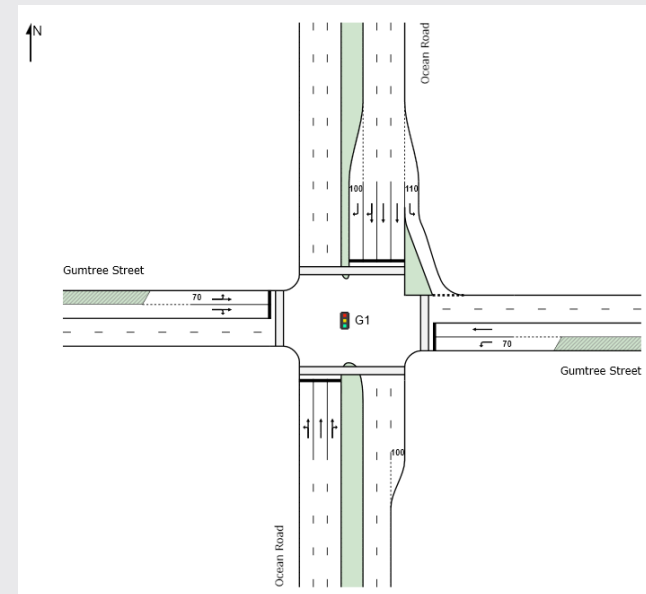
- The SIDRA Project File is a **Compact SQL** database and can be edited directly using SQL commands.
- Changing to **SQLite** in **Version 9**.
- It is not required to have SIDRA INTERSECTION installed on the computer to edit the Project File in this way.

Direct Access to SIPx File Database (2)

- There are no limitations or error checking when accessing the Project File in this way.
- Reading of any data from the database is OK.
- Writing or changing data should only be done in limited ways for items that will not have any interaction with other data, e.g. Volumes, Lane Width.
- Operations like removing a lane or leg are likely to have side-effects that will not be taken care of correctly and can lead to a Project File that can no longer be processed or used via the User Interface.

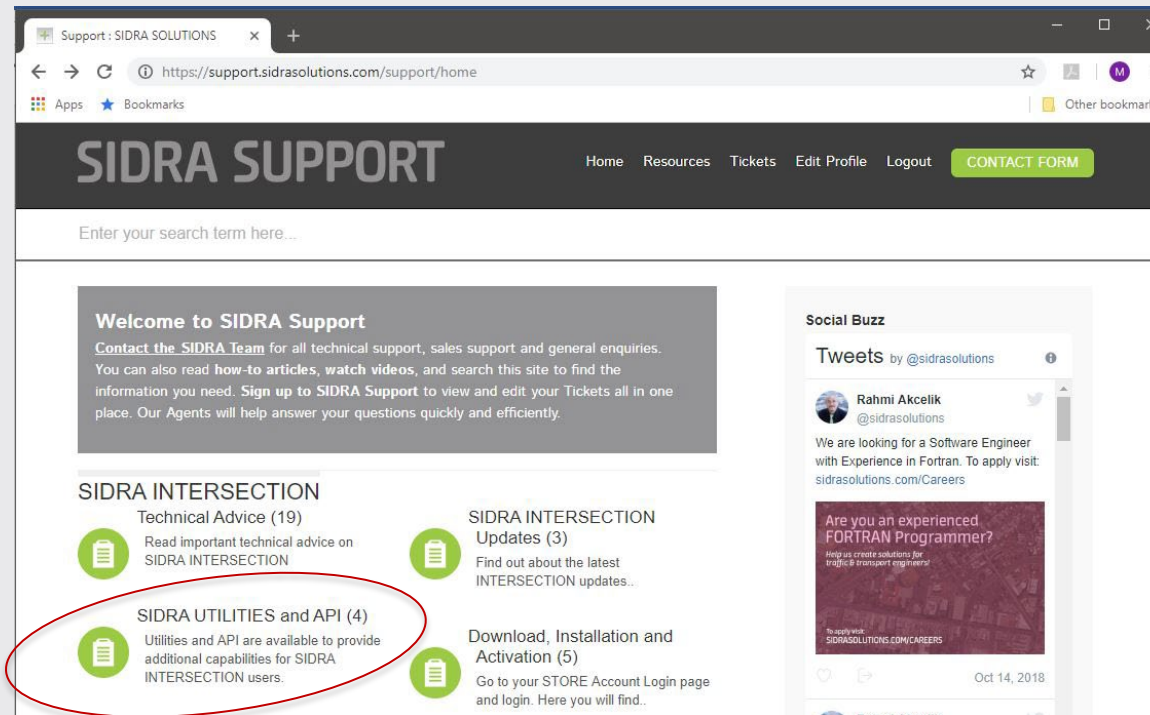
Limitations and Precautions

- Site and Network **Layout Pictures** can be created from the API but other graphical outputs are not available.
- You must not access the same Project File through multiple programs simultaneously (risk of data corruption).
- Creating unique names and IDs for Sites and Networks is the API programmer's responsibility.
- SIDRA INTERSECTION can be run and a specified project opened via a command line but there is no facility for Site selection.



Resources for Users

SIDRA SUPPORT Website



<https://support.sidrasolutions.com/support/solutions>

Resources for Users

- **Help File** with complete listing of **Classes, Properties and Methods**.
- Also gives insight into the project file **database structure**. Most of the names and data structure in API are the same as the SQL database.

The screenshot displays the SIDRASolutions.SI.API 8.0 Reference Documentation window. The left sidebar shows a tree view of the API structure, with 'SIAPILaneApproach' expanded under 'Properties'. The main content area shows the 'SIAPILaneApproach Class Properties' page. It includes a 'Public Properties' table with columns 'Name' and 'Description'.

Name	Description
Basic_saturation_flow	
Buses_stopping	
Buses_stopping_user	
Capacity_adjustment	
Configuration	Lane Configuration
Control_type	Lane Control Type
Departure_headway_awsc	
Grade	
Is_capacity_adj_for_network	
Is_departure_headway_awsc_appl	
Is_dominant_lane	
Is_sliplane_excluded_from_signal_analysis	
Is_sliplane_included_in_entry_lane_count	
LaneApproachMovements	Approach Lane Movements, defines the OD Movements (per Movement Class) running on this Approach Lane. It always returns eight Movements to the eight destinations (0: South, 1: South East, ... 7: South West).

Future Support

- We will try to maintain a similar structure in the API for future SIDRA Versions to minimise changes that would be necessary to existing API programs.
- Current SQL Compact will change to SQLite in SIDRA Version 9. This would only affect programs that access the SIP file data directly.

Example Programs

On the SIDRA SOLUTIONS Support Website:

- **CSharpSample** using C# (.NET)
- **CppSample** using C++ (COM)
- **SIAPythonSample** using Python (COM)
- **VOLUMES Excel** using VBA (COM)

We will demonstrate some of these today.

I will start by showing the **VOLUMES Excel**.

VOLUMES Excel Utility

The **VOLUMES** utility for SIDRA INTERSECTION is a macro-enabled Excel application:

- ❖ allows specification of **volumes and related data** within the Excel file,
- ❖ processes a selected Site in a SIDRA INTERSECTION Project using data given in the Project file or in the Excel file,
- ❖ provides Volume Output, Intersection Output, Movement Output, Lane Output and Pedestrian Output data in Excel sheets.

For detailed information about **VOLUMES Excel Utility**, refer to **SIDRA INTERSECTION 8 User Guide, Section 12**.

VOLUMES | SIDRA INTERSECTION 8
© 2009-2018 Alcatel & Associates Pty Ltd

CONTROL
For detailed information, refer to the Introduction sheet.

Click to browse for the SIDRA INTERSECTION Project file

Open SIDRA INTERSECTION Project File Save SIDRA INTERSECTION Project File

SIDRA INTERSECTION Project:
Click the **Open SIDRA INTERSECTION Project File** button to browse for the Project file (extension .sip8).
Click the **Save SIDRA INTERSECTION Project File** button if you wish to save the changes made to the as a result of the use of this Excel application.

C:\Users\Raahmi\Documents\SIDRA SOLUTIONS\SI PROJECTS\SI0 - SI8 Tests\NetworkTest 2.sip8

Selected SIDRA INTERSECTION Project file

Site:
Select the Site using the drop-down box below.

Signals T
6-leg Signals
PedSignal Two-Way
Signals T Intersection US
Rou with Bypass Lanes US
Two-Lane Roundabout US
Nepean - Metering Signals
Signals T Actuated

Drop-down list to select a Site

Process (No Volume Update)

Processes selected Site using volumes contained within the Project. Populates "Output" sheets with Site output.
Volumes and related data in the **Volume Input** sheet will be ignored and data in the SIDRA INTERSECTION Project file will

Introduction Licence **Control** Volume Input Volume Output Intersection Output Mover

END OF PRESENTATION

Thank you!



SIDRA SOLUTIONS