

Spirent TestCenter IQ Automation Reference

General steps to utilize Spirent TestCenter IQ in automation



Following is the typical automation workflow, with some modifications.

- Notes:**
- View Spirent TestCenter IQ results in the database on your computer or in the database on a remote server. For instructions, see *Result Service Connections*. Search for **DOC12115** on the Spirent Customer Service Center (support.spirent.com)
 - Refer to the *Spirent TestCenter Automation Overview* manual for details on the automation workflow steps.

- 1 Set up a communication link between your PC and your Spirent TestCenter chassis.
- 2 Prepare the DUT/SUT.
- 3 Connect the Spirent TestCenter chassis to the DUT/SUT.
- 4 Initialize the Spirent TestCenter API to establish the object set context.
- 5 Create a project object and set the project attributes.
- 6 Create Port objects and set the port attributes.
- 7 Create StreamBlock objects (and, if necessary, header objects for the traffic), and set the appropriate attributes.
- 8 Set up the Spirent TestCenter generator and analyzer for traffic support.
- 9 Establish the software connection from your PC to the Spirent TestCenter chassis.
- 10 Start the test.
- 11 Take snapshots if needed (TestCenter IQ)
- 12 Review/Retrieve the test results (TestCenter IQ)
- 13 Cleanup after the test has completed.

Results are stored in a Postgres database. In order to retrieve results, a query must be sent to Result Service. This service is the interface to the database.

The TestCenter IQ web UI provides example queries in the standard and the drill-down table views.

The next section describes how to acquire queries and use them in a script. After copying the queries, there are two ways to embed them into an automation script:

- Spirent TestCenter commands
- Interface with the Result Service REST API directly

When these queries are set up, the subsequent test runs will be fully automated.

Acquire Query

To acquire a query that you can use to verify results, copy the query in a relevant table View in TestCenter IQ.



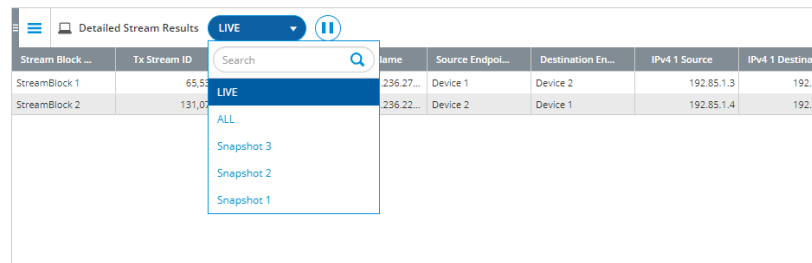
Follow these steps:

- 1 Launch the Spirent TestCenter application.
The application must be running in order to start up Result Service.
- 2 Launch TestCenter IQ and open the current test.
- 3 Open a Standard or Drill-down table View that shows the results of interest.



Note: If you would like snapshot results, take the snapshots before continuing to step 4. A snapshot consists of the result values at a point in time. This is similar to the Classic Results *Save Result* action.

- 4 Complete any of the following actions to create the correct query.
 - Select **LIVE** or **Snapshot** from the drop-down options.
This action defines the scope of results to be included in the query.



- Select **Edit** to choose counters/columns to retrieve from results.

The screenshot displays the 'Detailed Stream Results' window in Spirent TestCenter. A context menu is open over the table, and an 'Edit: Detailed Stream Results' dialog box is overlaid on the right side.

Context Menu Options:

- Save As...
- Export As...
- Edit
- Add Conditions
- View Query
- Refresh Rate
- Duplicate
- Remove
- About this View

Table Data:

Tx Stream ID	Tx Port Name	Rx Port Name	Source Endpoi...	Destination En...
65,536	Port //10.28.236.22...			
131,072	Port //10.28.236.27...			

Edit: Detailed Stream Results Dialog:

View Name: Detailed Stream Results

Select Data Columns for this View: [Search]

All Available Data Columns:

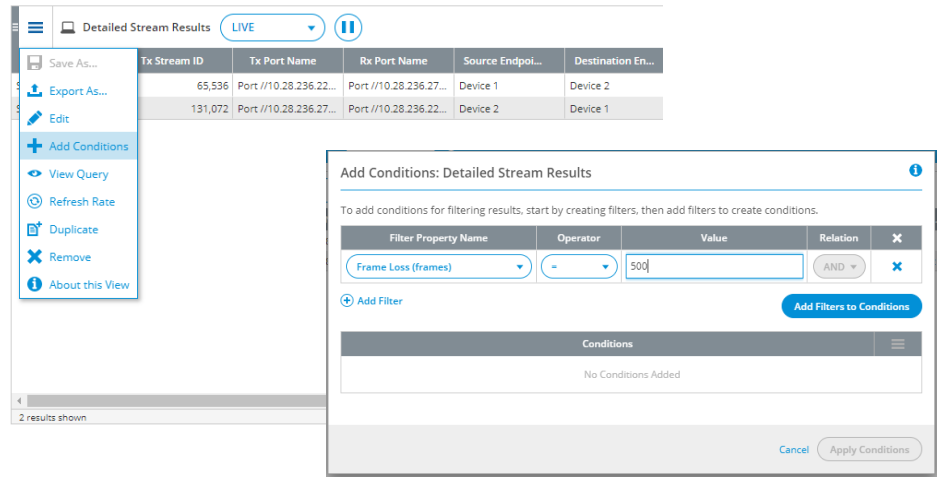
- Active In Sequencer Test
- Avg Jitter (µs)
- Dropped Frame Percent
- Dropped Frame Percent Rate
- Dropped Rate (fps)
- Duplicate Rate (fps)
- EthernetII 1 Destination MAC
- Expected RX Port Count
- Expected Sequence Number

22 Columns In View:

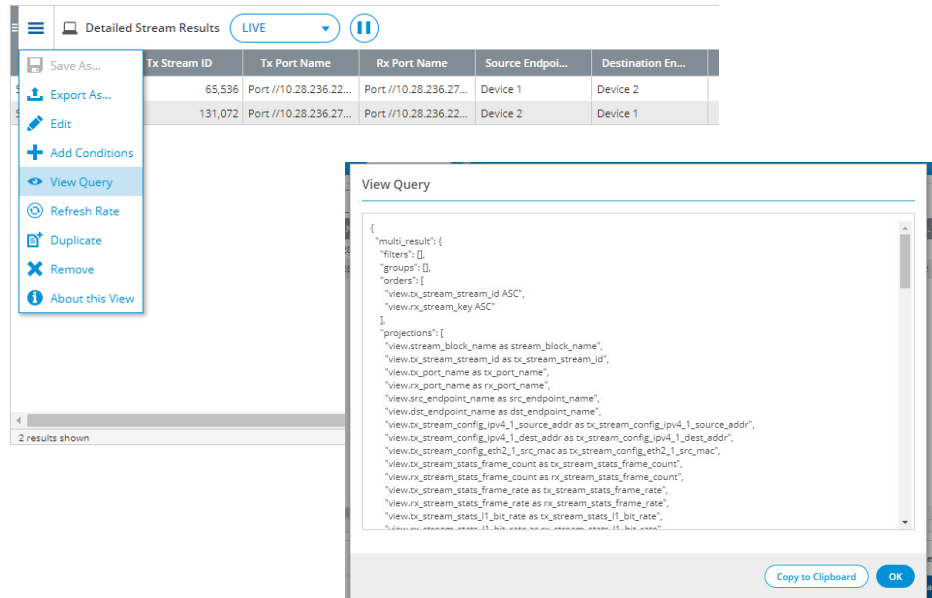
- Stream Block Name
- Tx Stream ID
- Tx Port Name
- Rx Port Name
- Source Endpoint Name
- Destination Endpoint Name
- IPv4 1 Source
- IPv4 1 Destination
- EthernetII 1 Source MAC

Buttons: Move Up, Move Down, Cancel, Apply Changes

- Select **Add Conditions** to further narrow the results query.



- Select **View Query** to display the query and then copy the entire query.



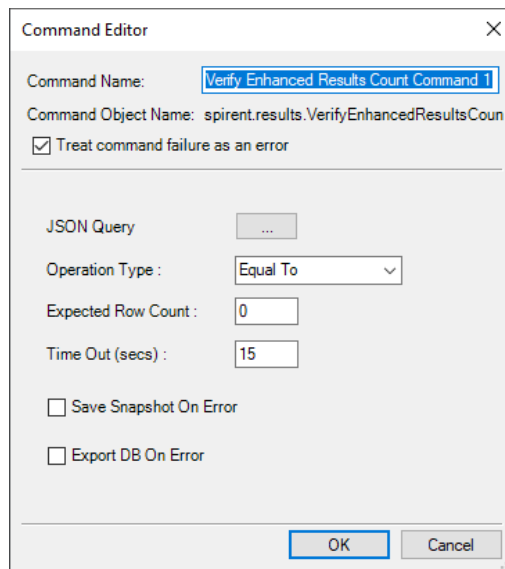
The acquired query is a JSON string. You can use it as a template for similar queries. For example, if the acquired query selects **Snapshot 1**, you can change the value to **Snapshot 2** and run the query again.

Use Spirent TestCenter Commands

TestCenter IQ-specific commands are available in the Spirent TestCenter Sequencer and for use in automation. The follow commands are available:

- StartEnhancedResultsTestCommand - this command creates a new db. If the test is in session, this command stops the current test, then starts a new one. The test name that you set is used in the TestCenter IQ Results Library.
- StopEnhancedResultsTestCommand
- StartEnhancedResultsIterationCommand - this command starts one test session; only RFC tests have multiple iterations.
- StopEnhancedResultsIterationCommand
- SaveEnhancedResultsSnapshotCommand
- DeleteEnhancedResultsTestCommand - deletes the most recent test if the Test ID is empty. If the Test ID is set that test is deleted.
- VerifyEnhancedResultsCountCommand
- VerifyEnhancedResultsValueCommand

All Spirent TestCenter commands listed are communicating with Result Service in the backend through the REST API. For example, the query acquired in last step can be used in VerifyEnhancedResultsCountCommand and VerifyEnhancedResultsValueCommand JSON Queries to retrieve results from db.



The screenshot shows a 'Command Editor' dialog box with the following fields and options:

- Command Name: Verify Enhanced Results Count Command 1
- Command Object Name: spirent.results.VerifyEnhancedResultsCount
- Treat command failure as an error
- JSON Query: ...
- Operation Type: Equal To
- Expected Row Count: 0
- Time Out (secs): 15
- Save Snapshot On Error
- Export DB On Error
- Buttons: OK, Cancel

When the Spirent TestCenter command runs, Spirent TestCenter sends the query to Result Service, which returns the results in a JSON string.

VerifyEnhancedResultsCountCommand and VerifyEnhancedResultsValueCommand have options to save the test and/or export the test on failure. This will allow the tests to be saved at the time of the failure.

Utilize Result Service RESTful API Directly

In essence, Spirent TestCenter commands use the RESTful API to get the results from Result Service. The same can be done by using the scripting languages and REST packages directly. The following scripts are provided as examples:

- Python REST API support
- Tcl REST API support

These example scripts are located in the **SampleScripts** folder under the Spirent TestCenter installation directory.

The next section describes the steps to set up the Tcl environment to run the provided sample scripts.

Set up Tcl to run a Spirent TestCenter IQ automation sample

Tcl must already be installed on your machine, and a library path must be set up, so that it can run Spirent TestCenter scripts. The sample code provided was tested with ActiveTcl 8.5. At the time of this writing, the available version is 8.5.18. Several packages must be installed for the sample code to work. This section describes the steps to set up the Tcl environment.

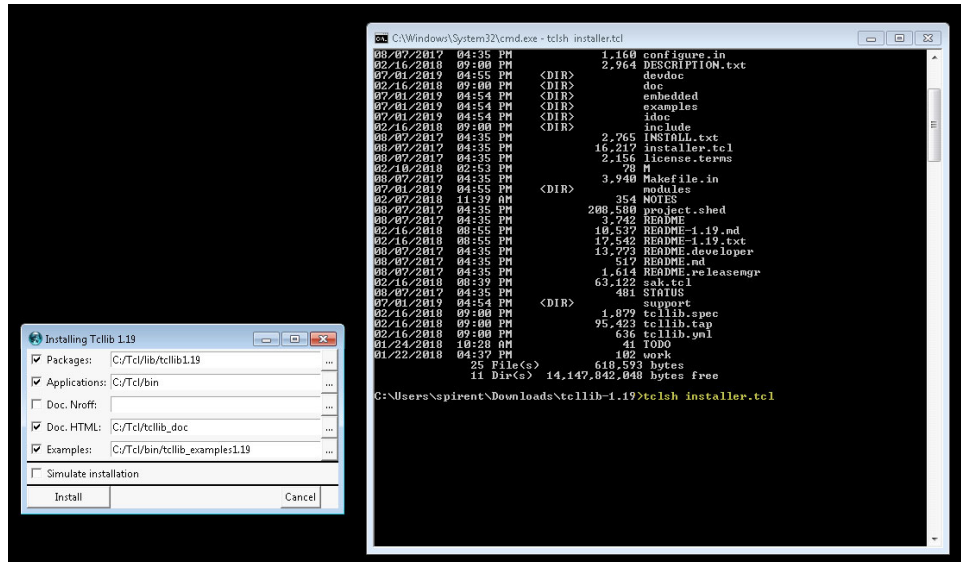
- 1 Download and install tcllib 1.19
- 2 Download and install tdom
- 3 Validate setup

Download and install tcllib 1.19

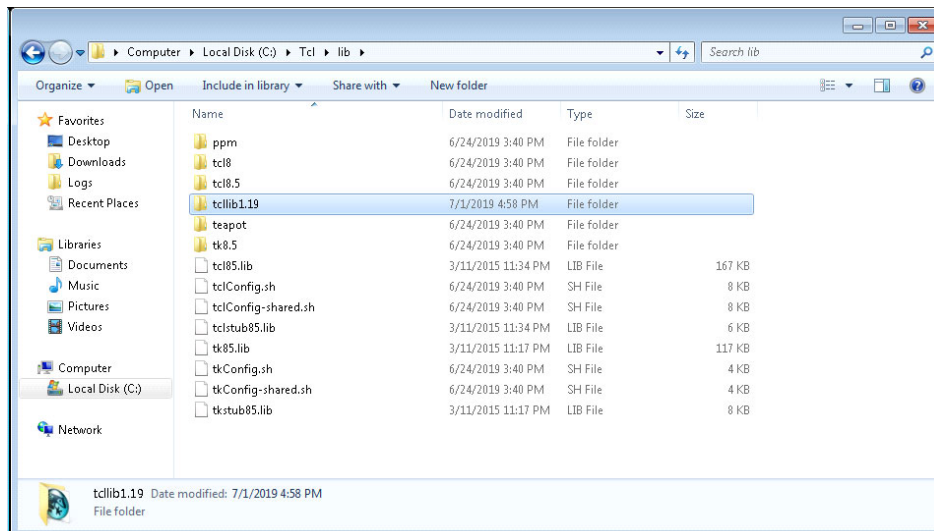
tcllib contains the rest package that is required by the sample script. Locate the download at <https://sourceforge.net/projects/tcllib/files/tcllib/>.

Download the latest version 1.19 file based on your system. For example, for Windows users, download tcllib-1.19.zip. Unzip to a folder and type:

```
tclsh installer.tcl
```



A pop-up window opens so you can confirm the location to install Tcl. Once installed, confirm that the tcllib1.19 folder is located under the corresponding location as shown in the example screen capture.



Download and install tdom

The rest package itself requires another third-party package tdom, which can be downloaded from <http://tdom.org/downloads/>.

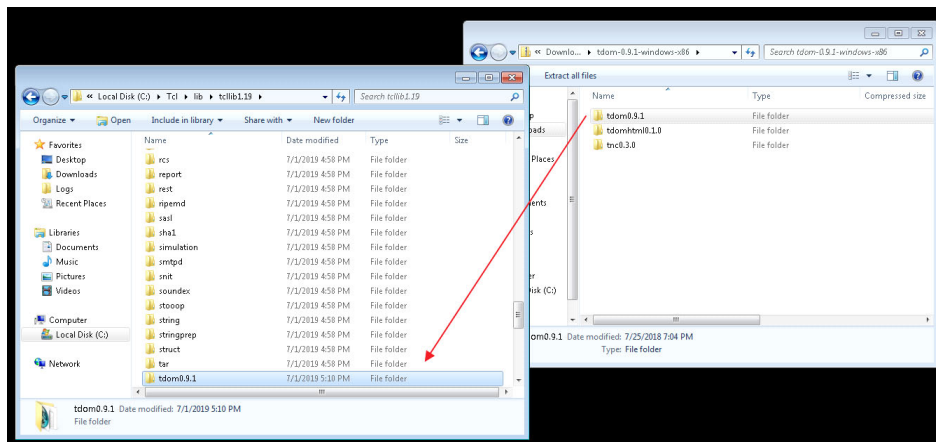
Download the appropriate files based on your system. For example, for a 32-bit Windows system, download tdom-0.9.1-windows-x86.zip.

Index of /downloads

Name	Last modified	Size	Description
Parent Directory			-
latest-src.tgz	25-Jul-2018 00:04	1.0M	
latest-src.zip	25-Jul-2018 00:04	1.1M	
tdom-0.9-windows-x64.zip	23-Aug-2017 13:21	726K	
tdom-0.9-windows-x86.zip	23-Aug-2017 13:21	689K	
tdom-0.9.0-src.tgz	24-Aug-2017 16:00	1.0M	
tdom-0.9.0-src.zip	24-Aug-2017 16:00	1.1M	
tdom-0.9.1-src.tgz	25-Jul-2018 00:04	1.0M	
tdom-0.9.1-src.zip	25-Jul-2018 00:04	1.1M	
tdom-0.9.1-windows-x64.zip	25-Jul-2018 20:56	1.3M	
tdom-0.9.1-windows-x86.zip	25-Jul-2018 20:56	1.3M	

Apache/2.2.22 (Debian) Server at tdom.org Port 80

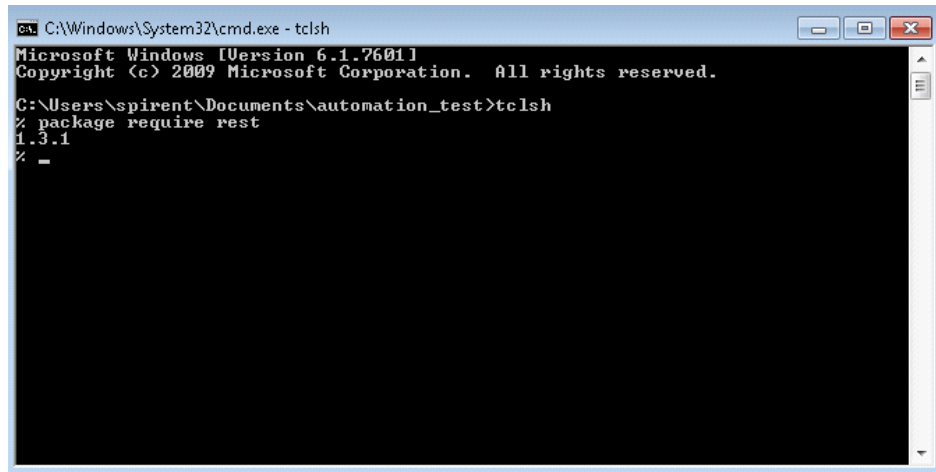
Open the downloaded file and copy tdom0.9.1 to the tcllib1.19 folder installed in the previous step.



Validate setup

Confirm that the rest package is installed correctly before running the sample script. Open a tcl command prompt and type: `package require rest`

If everything is set up correctly, it will return the version of the rest package which is 1.3.1.



```
C:\Windows\System32\cmd.exe - tclsh
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\spirent\Documents\automation_test>tclsh
% package require rest
1.3.1
% -
```

If it fails and reports an error about loading tdom.dll, make sure the tdom you used matches the tcl. For example, 64-bit Tcl requires a 64-bit tdom091.dll (Windows) or libtdomstub091.a (Linux).