SoapUI Integration

- SoapUI Integration with AIQ
- Run SoapUI as a Functional Test in AIQ
- Running SoapUI as a Performance Test
- Running Data-enabled SoapUI Test

SoapUI Integration with AIQ

Recommended guidance

SoapUI integration should be treated as legacy or customer-specific unless it is actively required. For current API testing, prefer Service Suite when it can cover the same requests, assertions, data handling, and reporting.

Integration review points

- SoapUI project location, version, plugins, dependencies, and execution host.
- Endpoint configuration, authentication, certificates, proxy, and firewall rules.
- Data sources, DPL mappings, assertions, and expected artifacts.
- Pass/fail mapping between SoapUI results and AIQ execution status.

Maintenance guidance

1. Keep SoapUI assets in version control with clear ownership.
2. Do not store secrets inside project files or shared logs.
3. Run small validation tests before scheduled or high-volume execution.
4. Retire or consolidate redundant SoapUI content when Service Suite replaces it.

Last updated: October 7, 2026

Run SoapUI as a Functional Test in AIQ

Recommended guidance

Run SoapUI as a functional test in AIQ when existing SoapUI projects still provide required validation. Keep the integration controlled and consider migration to Service Suite for new API coverage.

Before execution

- Confirm the SoapUI project file, version, dependencies, endpoint, environment, and credentials.
- Use secure placeholders or approved secret storage for sensitive values.
- Validate network, proxy, certificate, and firewall access.
- Define expected assertions, return codes, and artifacts.

Review workflow

1. Run the SoapUI test independently before AIQ execution.
2. Run a small AIQ wrapper and confirm output handling.
3. Review logs and generated reports for the first meaningful failure.
4. Document legacy dependency and replacement options if applicable.

Last updated: October 7, 2026

Running SoapUI as a Performance Test

SoapUI support was removed in release 5.2.0 of AIQ.

1. Navigate to the Appvance IQ Home screen and click Build
2. Click File >New to create a new scenario
3. Select Performance Test from the Scenario Test Type drop-down
4. Click the Test Management tab
5. Enter Iteration name, select Total Execution from the Completion Type drop-down, and enter Total Execution: 100
6. Click the Test Cases tab and click the link for Add test case execution
7. Enter the Test case name and Script name
8. Select "Service Suite-SoapUI" from the Script Type drop-down
9. Click Browse and browse the Soap UI XML file and then the Test Suite and then the Test Case from the drop-down
10. Save the test case
11. Enter 100 in the user box, Click Set (Run Time):
12. Ramp up 10 users every 10 seconds Maximum ramp-up time of 100 seconds and save the Run Time option
13. Click the Locations tab
14. Enter Name: localhost and URL: http://localhost:8080/TMServer/ws/TestNode (Should be selected by default)
15. Navigate to Resources and make sure SoapUI bundle with version 5.12 is added by default
16. Also make sure the resources are added with Name, Type, and the correct path of the XML file
17. Navigate to Options and make sure that Enable logging, Save logs on Failure are selected
18. Click check box Custom Database Credentials (Database Logging) and make the following settings:
    Database Type: MySQL
    Driver: com.mysql.jdbc.Driver
    URL: jdbc:mysql://localhost:3306/utp?useSSL=false
    Default Schema: utp
    Username: utp
    Password: utp
19. Click File >Save to save the new scenario
20. Click Play Scenario from the top navigation bar
21. The controller window will open in a new tab which displays the tests running and will display the pass / fail status of the tests upon the test completion
22. The left part shows the steps and the duration in milliseconds and the right side shows the test run status under the Output tab

23. On The transactions tab, we can find the request/response and an error tab to drill down further
24. Click Click here link to get to the list of reports that Appvance IQ supports

---

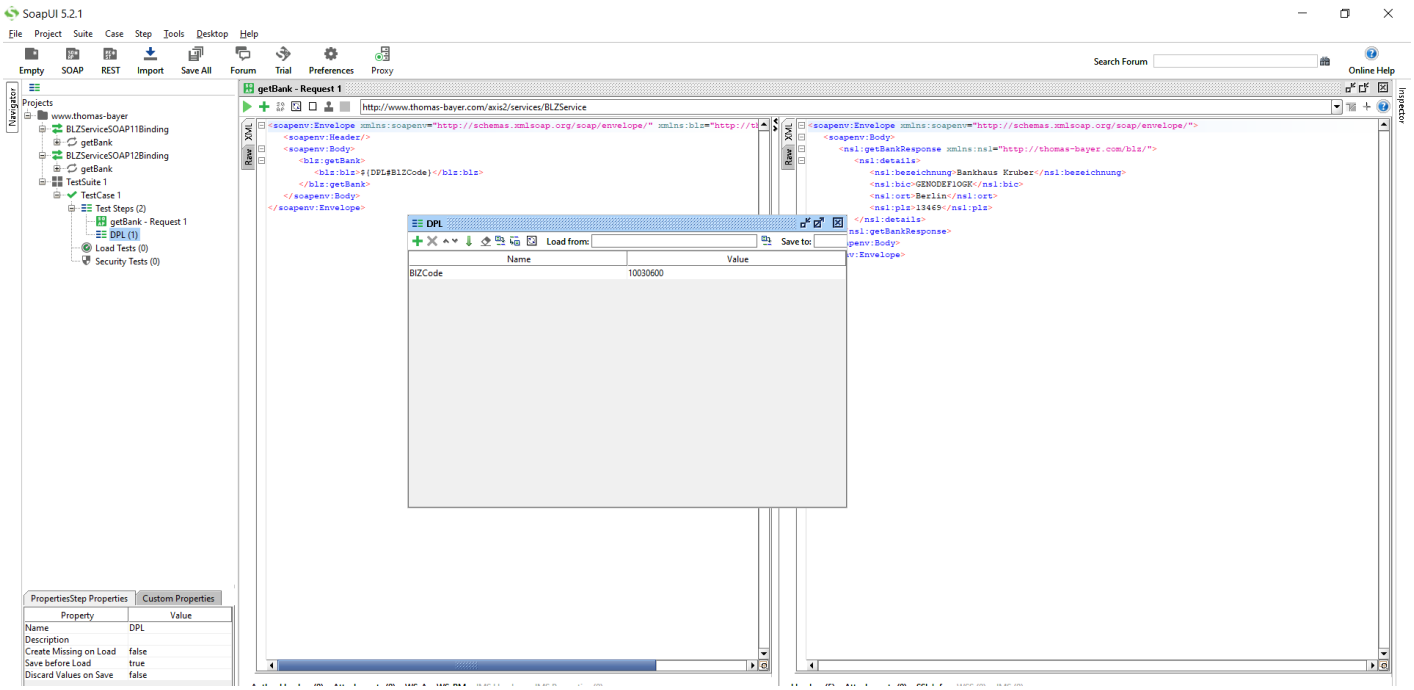
Last updated: July 25, 2024

If you have any feedback on this topic, please contact us at documentation@appvance.ai

Running Data-enabled SoapUI Test

SoapUI support was removed in release 5.2.0 of AIQ.

Make sure the SoapUI project created has the properties defined and the request is data enabled properly as shown below in the screenshot.

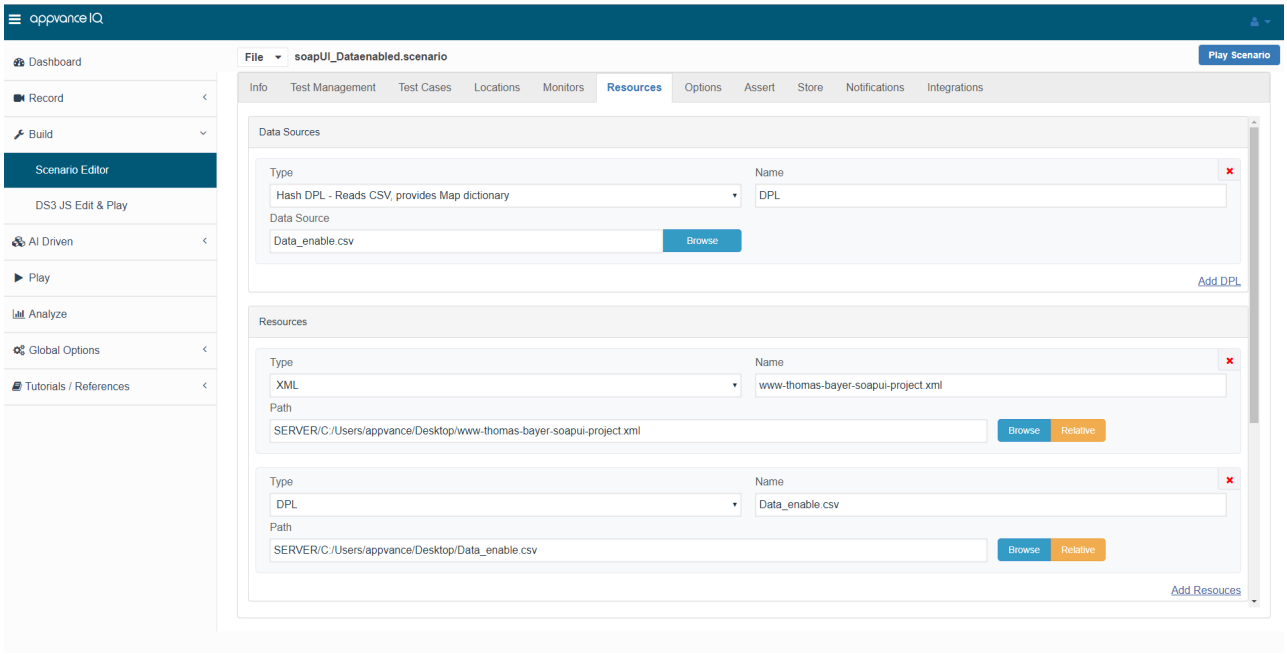


Save the project in SoapUI.

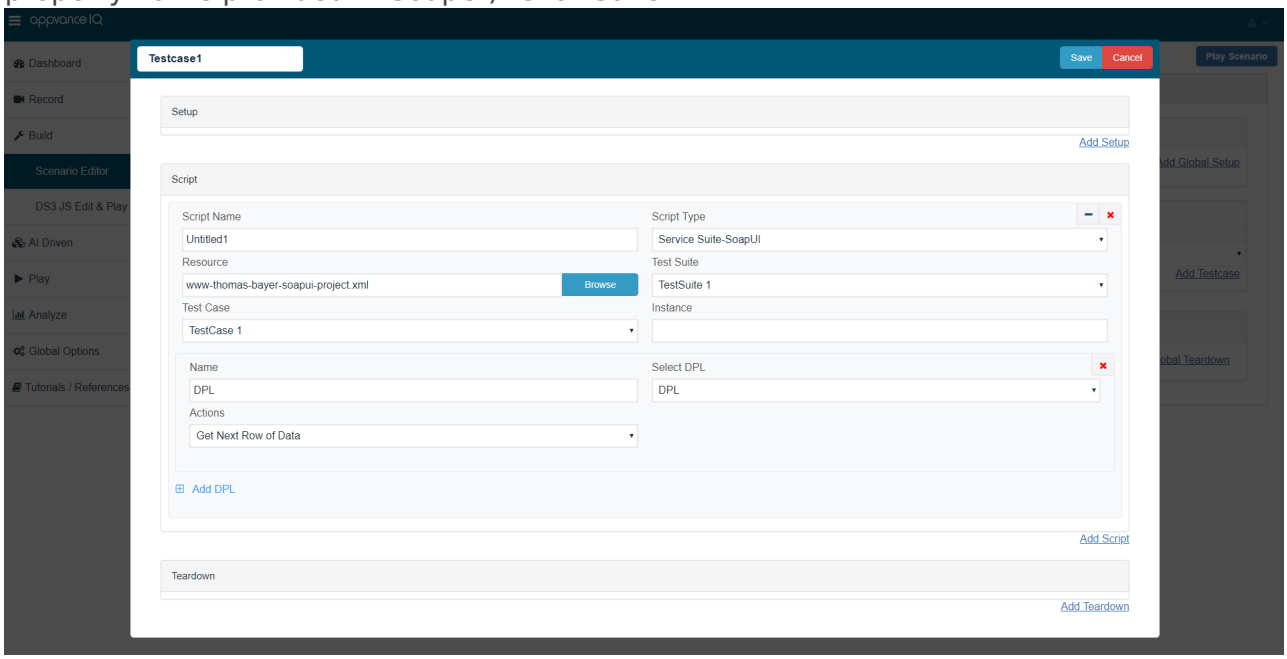
From Appvance IQ:

1. Navigate to the Appvance IQ Home screen and click Build
2. Click File ->New to create a new scenario
3. Select Functional Test from the Scenario Test Type drop-down
4. Click the Test Management tab and check: "Repeat scenario's test cases 2 times"
5. Click the Test Cases tab and click the link for Add test case link
6. Enter the Test case name and Script name
7. Select "Service Suite-SoapUI" from the Script Type drop-down
8. Click Browse and browse the Soap UI file and then the Test Suite and then the Test Case from the drop-down. Save the test case
9. Click the Locations tab
10. 10. Enter Name: localhost and URL: http://localhost:8080/TMServer/ws/TestNode (Should be added by default)

11. Navigate to Resources and make sure SoapUI bundle with version 5.2.1 is added by default
12. Also make sure the resources are added with Name, Type, and the correct path of the XML file
13. 13. Go to the Resources tab, Click Add DPL, Select 'Hash DPL - Reads CSV, provides Map dictionary' from the Type drop-down, Provide Name (Important: Should be same as the property name provided in SoapUI), select the Data source (the CSV file with all the entries created)



14. Come back to the Test cases panel, click edit and click Add DPL, under Select DPL, select the DPL that we just added, provide the name (Important: Should be the same as the property name provided in SoapUI). Click Save



15. Navigate to Options and make sure that Enable logging, Save logs on Failure are selected
16. Click check box Custom Database Credentials (Database Logging) and make the following settings:
    Database Type: MySQL
    Driver: com.mysql.jdbc.Driver
    URL: jdbc:mysql://localhost:3306/utp?useSSL=false

Default Schema: utp
Username: utp
Password: utp

17. Click File ->Save to save the new scenario
18. Click Play Scenario from the top navigation bar
19. The controller window will open in a new tab which displays the tests running and will display the pass / fail status of the tests upon the test completion
20. The left part shows the steps and the duration in milliseconds and the right side shows the test run status under the Output tab
21. On The transactions tab, we can find the request/response and an error tab to drill down further
22. Click the Click here link to get to the list of Reports that Appvance IQ supports

---

Last updated: July 25, 2024

If you have any feedback on this topic, please contact us at documentation@appvance.ai