The Titan WebAPI

Starting with Titan v10.0 the Titan WebAPI was introduced:

> It is now possible to remotely control console functions via HTML
> providing a way to create custom interfaces with specific layouts and capabilities.
> [Titan v10 Release Notes]

Very essentially it is implemented so that you can trigger some actions, set some values or get some data by calling commands from your browser, e.g. `http://[ip]:4430/titan/script/Playbacks/FirePlaybackAtLevel?userNumber=1&level=1&bool=false`: the WebAPI is a special webserver running on port 4430. It is available on Titan consoles (not on Titan One and T1) since Titan v10 (T2: from v12.1)

- there is a very coarse documentation of the available commands at http://api.avolites.com
- there is a section with some demo files on the Avolites website
- there is an example implementation at https://www.avosupport.de/wiki/downloads/web-api-demo#english
- another example implementation by Olie Waits: https://github.com/owaits/avolites-webcomponents

This wiki is an attempt to give an overview, list and explain working examples, and show the most common use cases.

Limitations

While the concept is very interesting for some use cases - you can implement your own user interface; you can access functions which cannot be made available in other ways, etc. - there are some strong limitations:

- the web API runs headless, with its own programmer and UI. Unfortunately this UI cannot be accessed at all. That's why any functions which work with the programmer cannot be used in a meaningful way (hopefully this will change in a future version). As a result there are only few commands/functions which are useful at all.
- there are browser-related difficulties: modern browsers prevent techniquies which load a website from one server and then request data from another site. Hence, many examples work only if at first the page is stored locally and then opened locally. Alternatively you can tweak your browser's security settings (XSS prevention, same origin policy, cors) - but this may invoke other problems and may change with future versions
- when attempting to call Titan on the same computer (e.g. when using the PC suite) you may need an active network in order to let the computer talk with itself
- while in theory you can simply request commands via your browser's address bar it is way better to use JavaScript functions in particular as these do not reload your page each time you hit Enter. However, using JavaScript almost directly leads to the use of some JS frameworks which adds another layer of complexity: if you are familiar with a particular framework then it is absolutely fine - but if not then it is hard do find the actual API command and adapt it to your very own implementation

- some request sequences need to be executed in a given order. This may require using synchronous instead of asynchronous requests - a technique deemed as deprecated by browser manufacturers.

Tricks/Macros

Maybe it is possible to add some functions by combining the WebAPI with macros, see Store cues with current page names as legends

From:
https://avosupport.de/wiki/ - **AVOSUPPORT**

Permanent link:
https://avosupport.de/wiki/webapi/start

Last update: **2025/04/10 09:59**