WoT Profiles (2)

Michael Lagally

23 June 2021

Agenda

- PRs
- Actions (continued)
- Test Plan

PRs

PRs

RFC 2119 markup: add css mark <span> into index.html

- https://github.com/w3c/wot-profile/pull/79

Discussion:

We noticed that some span markups are in editors notes – these need to be cleaned up in a further iteration.

We merged the PR, Mizushima-san will provide additional comments.

McCool will do another pass using the test tooling.

PRs

New Profile Mechanism section

- https://github.com/w3c/wot-profile/pull/79

Discussion: Incorporated, previous chapter content to be carried over to the architecture spec.

PRs

Remove WoT Core Data Model section

- https://github.com/w3c/wot-profile/pull/78

Discussion: not discussed

PRs

Device Categories

- https://github.com/w3c/wot-profile/pull/70

Discussion: Move to the architecture spec

PRs

Reference Device

- https://github.com/w3c/wot-profile/pull/69

Discussion: not discussed

PRs

Canonical representation 2

- https://github.com/w3c/wot-profile/pull/57

Discussion: close

Actions (revisited)

Actions

- The server models interactions that cause state change via actions.
- Actions can have a different **request modes**, they can be synchronous or asynchronous.
- The TD does not have a mechanism to annotate whether an action is sync or async.
  Synchronous action invocations return (*after a while)* with a response (result or timeout) when it is complete. ~~., [the actual operation (e.g. robot arm moving) may still be ongoing.]~~
- **Asynchronous** actions are triggered by the invocation, they return instantaneously with a response.
- This response may contain various data, a handle, an id, invocation status.
- The **consumer** should be able to indicate, whether he wants to wait (sync call with a timeout) or if he wants to perform an async call.

Request modes and timeout

- The request mode can be specified with query parameters "iot.sync" or "iot.async".
- By default if none of the parameters is set then the request is asynchronous.
- These parameters must not be set both.
- The asynchronous mode is a preferrable way of doing invocation as it does not consume as much server resources as compared with the synchronous mode.
- Optionally, a timeout value in milliseconds may be provided with the "iot.timeout" query parameter.

Request modes and timeout

Discussion:
we could use mode= “sync/async, potential future extensions.
There may be implementations that only can be sync or async.
Suggested approach: we force the producer to support both ways.
A server could indicate that it does not support sync calls, this could be indicated via a response code
producer should define whether it is sync or async
consumer code for sync and async is very different
hard real time is not a goal here, focus is green field
sync does not mean that the scripting API needs to be synchronous, the script is still async
we could make all actions async and meet the goals of the charter period
TD annotations for sync/async are planned per issue https://github.com/w3c/wot-thing-description/issues/890
initiate, monitor and cancel are on the charter –
let’s not defer defining basic actions to a future spec but address it in the profile
defining sync / async is easy, however there are several issues that require more thought – defer to 2.0
we should clarify actions based on actual use cases, these were done in Plugfest by various companies, Echonet binding should be considered

Test Plan

Testing

Testing Objective:

- Prove that the specification is implementable.

The profile is a subset of the TD spec, for which we have several implementations.

Deliverable:

- Implementation report

References

WoT Architecture task force

- https://www.w3.org/WoT/activities/tf-architecture/

WoT Profile repository

- https://github.com/w3c/wot-profile

Working draft:

- https://www.w3.org/TR/wot-profile/