ASSISTANT / SMART HOME

Developer Preview of Local Home SDK

JUL 09, 2019

Posted by Toni Klopfenstein

Recently at Google I/O, we gave you a sneak peek at our new Local Home SDK, a suite of local technologies to enhance your smart home integrations. Today, the SDK is live as a developer preview. We've been working hard testing the platform with our partners, including GE, LIFX, Philips Hue, TP-Link, and Wemo, and are excited to bring you these additional technologies for connecting smart devices to the Google Assistant.

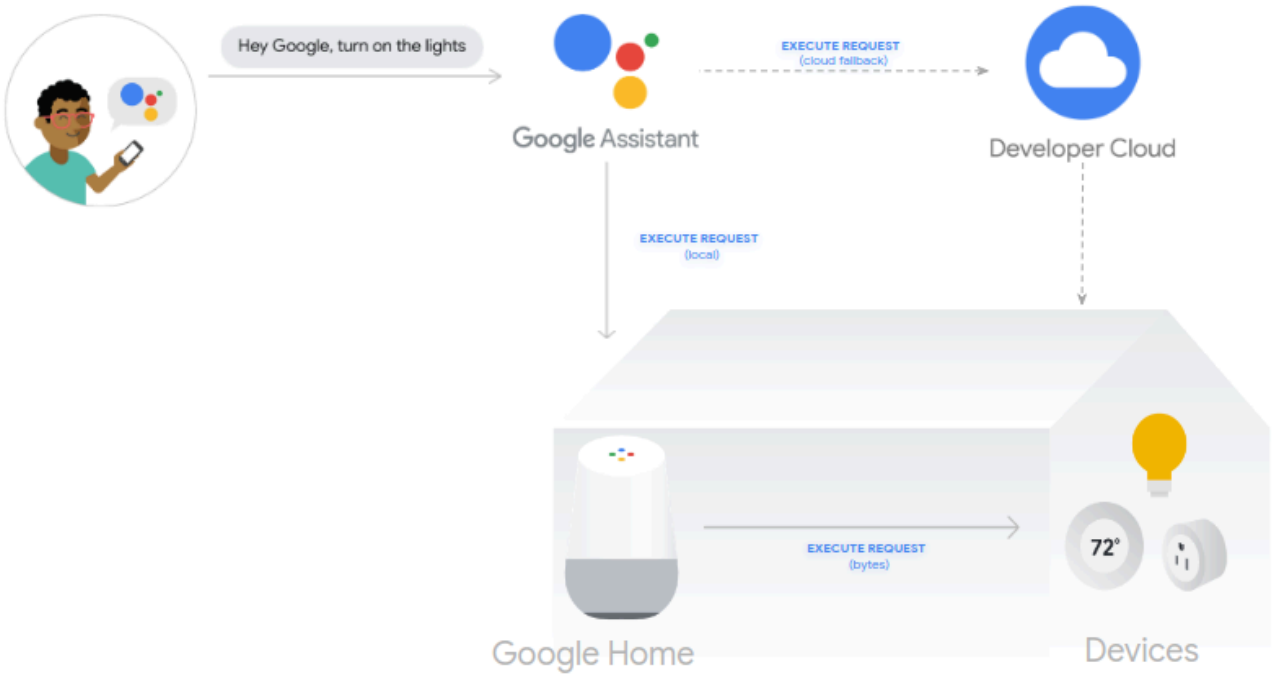


Figure 1: The local execution path

This SDK enables developers to more deeply integrate their smart devices into the Assistant by building upon the existing Smart Home platform to create a local execution path via Google

Google for Developers

The SDK introduces two new intents, `IDENTIFY` and `REACHABLE_DEVICES`. The local home platform scans the user's home network via mDNS, UDP, or UPnP to discover any smart devices connected to the Assistant, and triggers `IDENTIFY` to verify that the device IDs match those returned from the familiar Smart Home API `SYNC` intent. If the detected device is a hub or bridge, `REACHABLE_DEVICES` is triggered and treats the hub as the proxy device for communicating locally. Once the local execution path from Google Home to a device is established, the device properties are updated in Home Graph.

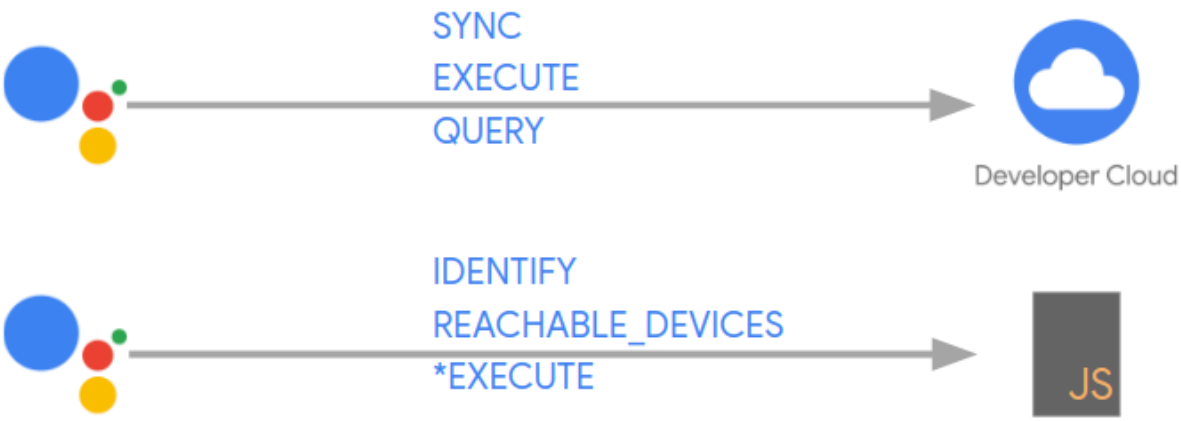


Figure 2: The intents used for each execution path

When a user triggers a smart home Action that has a local execution path, the Assistant sends the `EXECUTE` intent to the Google Nest device rather than the developer's cloud fulfillment. The developer's JavaScript app is invoked, which then triggers the Local Home SDK to send control commands to the smart device over TCP, UDP socket, or HTTP/HTTPS requests. By defaulting to local execution rather than the cloud, users experience faster fulfillment of their requests. The execution requests can still be sent to the cloud path in case local execution fails. This redundancy minimizes the possibility of a failed request, and improves the overall user experience.

Additional features of the Local Home platform include:

- Support for all Wi-Fi-enabled device types and device traits without two-factor authentication enabled.
- No user action required to deploy Local Home benefits to all devices.
- Easily configure discovery protocols and the hosted JavaScript app URL through the Actions console.

Google for Developers

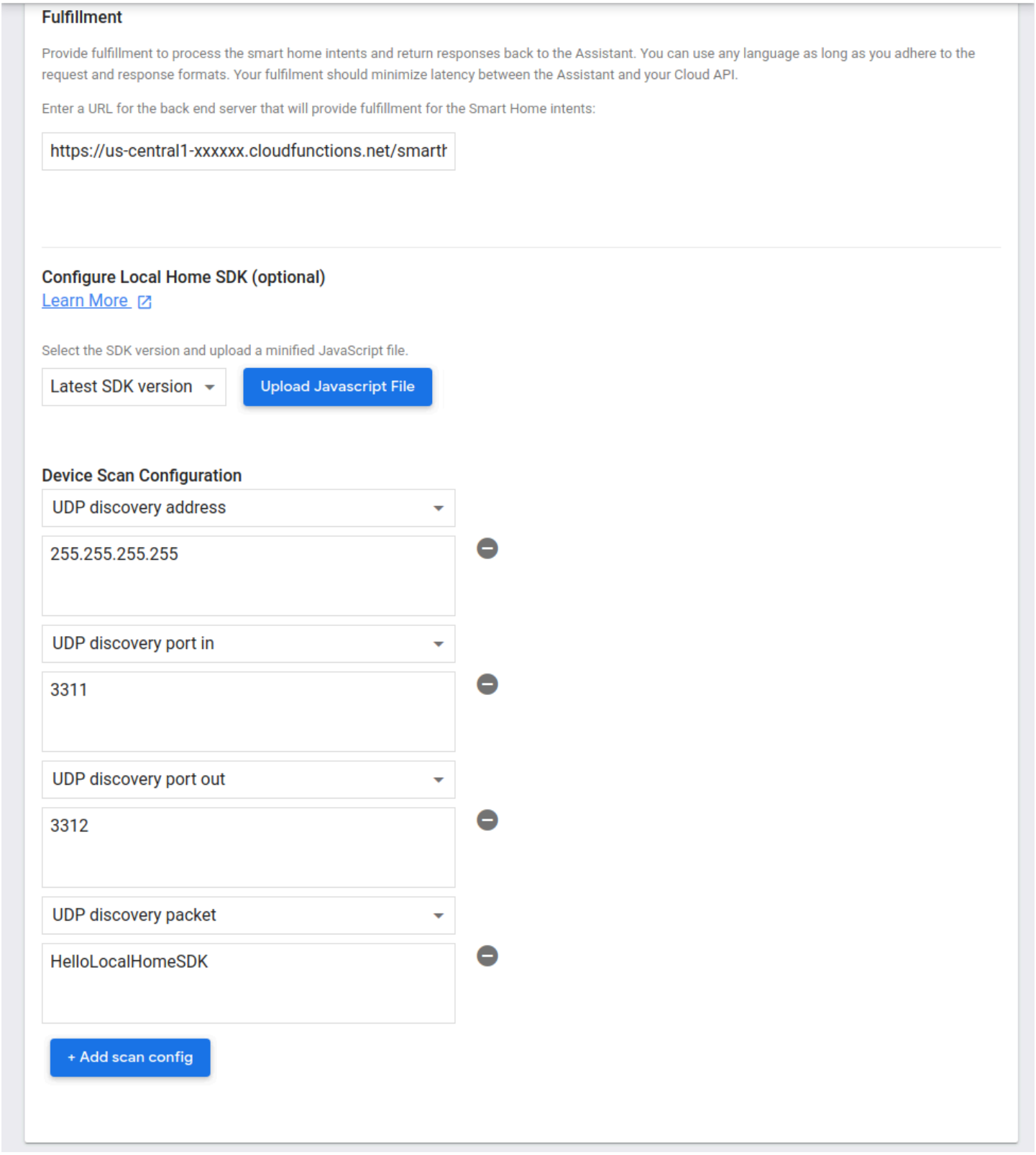


Figure 3: Local Home configuration tool in the Actions console

JavaScript apps can be tested on-device, allowing developers to employ familiar tools like Chrome Developer Console for debugging. Because the Local Home SDK works with the existing smart home framework, you can self-certify new apps through the Test suite for smart home as well.

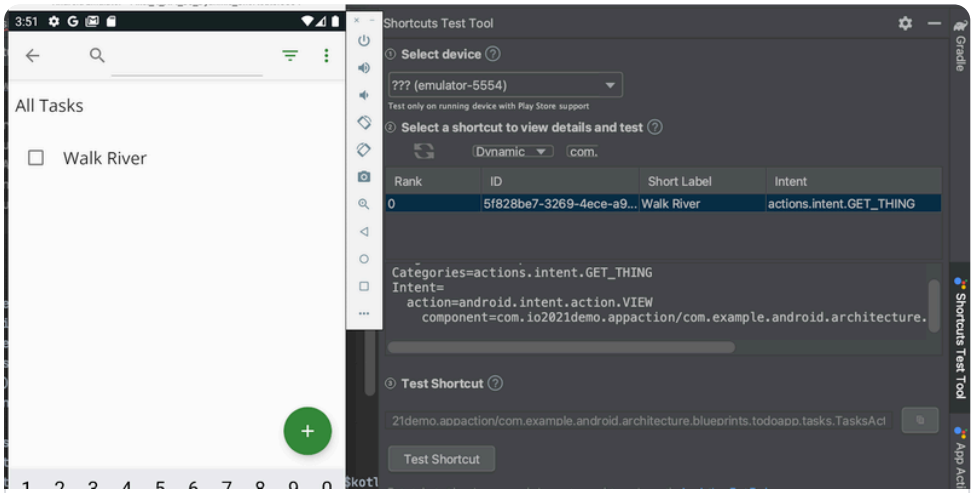
documentation, or check out the Local Technologies for the Smart Home talk from Google I/O this year.

You can send us any feedback you have through the bug tracker, or engage with the community at /r/GoogleAssistantDev. You can tag your posts with the flair local-home-sdk to help organize discussion.

POSTED IN: Assistant Smart Home Local Home

PREVIOUS NEXT

Related Posts



ASSISTANT

Personalize user journeys by Pushing Dynamic Shortcuts to Assistant

Nov. 30, 2021

SMART HOME MOBILE A

Home APIs: Enab
the home

May 15, 2024