

Table des matières

Contents	3
-----------------------	---

Scratch Extensions make it possible to connect Scratch projects with external hardware (such as [LEGO WeDo](#) or [PicoBoard](#)) or sources of information on the web. They add a collection of command and reporter blocks that can be used to interact with a particular device or data source. When an extension is enabled, its blocks appear in the “More Blocks” palette.



The LEGO WeDo Javascript extension blocks

To add an extension, go to the “More Blocks” category in the Scratch Editor and click “Add an Extension.”



The button to add an extension in Scratch 2.0.

There are two types of extensions in Scratch 2.0: hardware and web.

Contents

- [1 Writing Extensions](#) * [1.1 Javascript Extensions](#)
 - [1.2 HTTP Extensions](#)
- [2 Distributing Extensions](#)
- [3 See Also](#)

Writing Extensions

There are two mechanisms to extend Scratch 2.0, either with HTTP requests or Javascript.

==== Javascript Extensions ====

This extension mechanism uses Javascript to create the extensions. The official documentation for creating this kind of extension can be found on the [scratch-extensions-docs](#) github repository, along with sample extension code.

The current extensions available in the online editor, like the [LEGO WeDo](#) extension, are Javascript extensions.

==== HTTP Extensions ====

Note: This mechanism works only with [Scratch 2.0 Offline Editor](#).



[A4S](#) (Arduino for Scratch) experimental HTTP extension.

Scratch 2.0 cannot interact with hardware devices directly. Instead, hardware extensions come with a helper app, a separate application that the user must install and run on their computer. Scratch

communicates with the helper app via HTTP requests, and the helper app talks to the hardware. In the future, some extensions may package their helper apps as browser plugins.

A [document](#) (*last updated: September 2013*) has been drafted for Scratch 2.0 HTTP extension developers to describe the extension description file format, the protocol used to communicate between Scratch extension helper apps, and the extension development process. This specification is still preliminary and may change.

The Scratch 2.0 extension mechanism is still under development, and [Scratch Team](#) is still figuring out how the extensions will be shared and distributed.

Distributing Extensions

An extension distribution strategy is still being worked out. The Scratch team will probably host a small library of “supported” extensions. Users will be able to browse and select extensions from this library from within the Scratch editor. Supported extensions would be checked for quality and safety by the Scratch team. There are likely be strict criteria for including an extension in the Scratch-team supported extensions library, such as command set clarity and ease of use, size of the potential audience, widespread availability of any associated hardware, and a long-term commitment to support the extension.

Projects with experimental extensions cannot be shared on the Scratch website. Extension developers can share their extensions by distributing Scratch 2 project files (.sb2) Users can then use the “File > Upload from computer” command to import the project that uses the extension.

See Also

- [How to Connect to the Physical World](#)
- [Hardware That Can Connect to Scratch](#)
- [LEGO Education WeDo Robotics Kit](#)
- [Scratch 1.4 Modifications](#)
- [Remote Sensor Protocol in Scratch 1.4](#)
- [Scratch Modification](#)

From:

<https://wiki.libreduc.cc/> - **LibrEduC**



Permanent link:

https://wiki.libreduc.cc/fr:scratch2:extensions_scratch

Last update: **2025/01/16 20:24**