

ECMAScript、Web 標準の型は どう管理されているか

fukuoka.ts #1



pixiv Inc.
森内建太 petamoriken

pixiv



petamoriken

カスタマーオペレーション
& セーフティ本部
オペレーション開発部

自己紹介

- Web エンジニア
- Deno コントリビューター
- ECMAScript や Web 標準を追うのが好き



moriken's project

2024-07 の TC39 meeting

まとめ

Agenda: <https://github.com/tc39/agendas/blob/main/2024-07.md>
速報: <https://dev.to/hemanth/updates-from-the-102nd-tc39-meeting-14f>
Note: <https://github.com/tc39/notes/pull/337>

決まったこと

Web compatibility issues / Needs Consensus PRs

Normative: Intl.DurationFormat Display negative sign on leading numeric-style zeroes (PR) (slides) | Ben Allen

特定の条件下でマイナスが表示されないバグの修正。

```
javascript
new Intl.DurationFormat('en', {hours: "numeric"}).format({minutes: -1, seconds: -2});
// currently: 0:01:02
// should be -0:01:02
```

Normative: fully define Math.sqrt (PR), part of tighter constraints on implementation-approximated values | Michael Ficarra and Dan Gohman

Math.sqrt を implementation-approximated でなくする変更。

2014年の議論で Math 関数の誤差を最大で 1 ULP にするべきという話が上がり合意されたが仕様変更はされなかった。その後アプリケーションからの報告によって何度か議題に上がり、2015年末に [collim](#) によって正しい丸めを実装できるかもしれないと期待されたが、当時フォローアップ調査の結果実用的でないとしてされた。今日ではまた状況が変わっているかもしれないとのこと。

Math.sqrt については WebAssembly 側にも命令があることもあって、implementation-approximated でなくしていいだろうということでこの話が上がった。

implementation-approximated については uhyo さんの記事が詳しい。
<https://zenn.dev/uhyo/articles/javascript-math-accuracy>

Normative: pretend primitives aren't iterable (PR) | Michael Ficarra

今後 Iterable を受け取る API はオブジェクトを期待するという合意が得られた。既に string には String.prototype[Symbol.iterator] が仕様の中に存在しており Iterable となっているが、これは失敗だったと捉える

DOM Observable は Push ストリームのデファクトスタンダードになるか？



ECMAScript Module Harmony



早く正式に来て日時処理に最終的な解決を与えてくれ
月が1始まりになるから早く使いたい



コメントを入力



いいね! 42

共有

ニコニ広告

ギフト



好きなTC39 Proposals発表ドラゴン / 重音テト

2024/4/15 ▶ 1,282 21



petamoriken / 森建

★
フォロー

© ンバヂ <https://x.com/nbaji9>

<https://www.nicovideo.jp/watch/sm42515407>

TypeScript の組み込み型定義

tsconfig.json の compilerOptions.lib で指定する


- ECMAScript
 - “ES2023” や “ESNext”
- Web 標準
 - “DOM” や “DOM.Iterable”



tsconfig.json











```
{
  "compilerOptions": {
    "target": "ES2023",
    "lib": [
      "ES2023",
      "DOM",
      "DOM.Iterable"
    ],
    "module": "ESNext",
    "moduleResolution": "Bundler",
    "noImplicitAny": true,
    "noImplicitOverride": true,
    "noUncheckedIndexedAccess": true,
    "strict": true,
    "useUnknownInCatchVariables": true
  }
}
```

組み込み型定義の実態

microsoft/TypeScript の src/lib 内にある

TypeScript / src / lib / 

 rbuckton built-in iterators should be disposable (#59633) ✓ f025a5b · 2 weeks ago  History

Name	Last commit message	Last commit date
 ..		
 README.md	Intl 2021 Updates (#45647)	3 years ago
 decorators.d.ts	dprint the codebase (#54820)	last year
 decorators.legacy.d.ts	Change ParamaterDecorator to allow an undefined propertyKey (#53365)	last year
 dom.asynciterable.generated.d.ts	Disambiguate BuiltinIterator/BuiltinIteratorReturn (#59506)	3 weeks ago
 dom.generated.d.ts	DOM update 2024-07-12 (#59259)	last month
 dom.iterable.d.ts	Disambiguate BuiltinIterator/BuiltinIteratorReturn (#59506)	3 weeks ago
 dom.iterable.generated.d.ts	Disambiguate BuiltinIterator/BuiltinIteratorReturn (#59506)	3 weeks ago
 es2015.collection.d.ts	dprint the codebase (#54820)	last year
 es2015.core.d.ts	Add missing parameters from Array.toLocaleString on ES2015 libs (#57679)	5 months ago

組み込み型定義の実態

- ECMAScript
 - 型の追加、修正は手作業
- Web 標準
 - 2つ以上のブラウザに実装された型が自動更新
 - microsoft/TypeScript-DOM-lib-generator

(余談) ES2024 の型を実装中

Introduce ES2024 target and fix some types #58573

petamoriken wants to merge 44 commits into `microsoft:main` from `petamoriken:feat/update-types`

Conversation 43 | Commits 44 | Checks 32 | Files changed 97 | +3,574 -2,060

petamoriken commented on May 19 • edited

Fixes [#54636](#)
Past PRs [#54637](#), [#57858](#)
Refs [denoland/deno#19191](#), [denoland/deno#24238](#)

This PR includes the following changes:

Introduce ES2024 target

- Introduce ES2024 target
- Move some ESNext types to ES2024
- BREAKING:** Move `Atomics.waitAsync` types from ES2022 to ES2024
- Add Resizable/Transfer ArrayBuffer types to ES2024

Fix some types

- Fix `ArrayBuffer.prototype.slice` types in ES5
- Fix `SharedArrayBuffer.prototype.slice` types in ES2017
- Add empty arguments of `ArrayBuffer / SharedArrayBuffer` constructor types to ES2017

Reviewers: [rbuckton](#) (Requested changes must be addressed to merge this pull request.)

Assignees: [sandersn](#)

Labels: **For Backlog Bug**

Projects: **PR Backlog** (Status: Waiting on author)

Milestone: No milestone

Reactions: 😊 1 🧑‍🚀 2 ❤️ 6 🛠️ 3

TypeScript-DOM-lib-generator

Mozilla の saschanaz さんによる管理

2つ以上のブラウザで実装された型が自動更新

- TypeScript へはバージョンアップ時に取り込まれる
- 最新の型を使いたい場合は @types/web へ

TypeScript-DOM-lib-generator の仕組み

w3c/webref から最新の Web IDL を JSON で取得

Webref

Description

This repository contains **machine-readable references of CSS properties, definitions, IDL, and other useful terms that can be automatically extracted from web browser specifications** (see a [list of projects known to use the data](#)). The contents of the repository are updated automatically **every 6 hours** (although note information about published [/TR/](#) versions of specifications are updated only once per day).

Specifications covered by this repository are technical Web specifications that appear in [browser-specs](#).

The `main` branch of this repository contains **automatically-generated raw extracts** from web browser specifications. These extracts come with no guarantee on validity or consistency. For instance, if a specification defines invalid IDL snippets or uses an unknown IDL type, the corresponding IDL extract in this repository will be invalid as well.

TypeScript-DOM-lib-generator の仕組み

Browser Compat Data と突合する

- ブラウザの実装状況や廃止された API かを調べる

@mdn/browser-compat-data

<https://github.com/mdn/browser-compat-data>

The `browser-compat-data` ("BCD") project contains machine-readable browser (and JavaScript runtime) compatibility data for Web technologies, such as Web APIs, JavaScript features, CSS properties and more. Our goal is to document accurate compatibility data for Web technologies, so web developers may write cross-browser compatible websites easier. BCD is used in web apps and software such as [MDN Web Docs](#), CanIUse, Visual Studio Code, WebStorm and [more](#).

Read how this project is [governed](#).

bcd-idl-mapper

This exposes a modified `api` field value of [@mdn/browser-compat-data](#) as the default export, with the following changes:

- Mixins are in `__mixins`, e.g. `api.__mixins.WindowEventHandlers.onstorage`.
 - Each member data have no direct `__compat`
 - Each member data have subitems with `__compat`, e.g. `api.__mixins.NavigatorLanguage.language.WorkerNavigator`
- CSS properties are in `CSSStyleDeclaration` in camel case, e.g. `api.CSSStyleDeclaration.justifyContent`
- `alternative_name` and `prefix` are mapped as separate items, e.g. `api.Window.clientInformation` and `api.CSSStyleDeclaration.webkitJustifyContent`.

`.__compat.support` object has one more field named `preferred_name`, which only exists in items generated from `alternative_name` or `prefix`.

Dependencies

This package has two peer dependencies, which means you should explicitly install them together.

- [@mdn/browser-compat-data](#)
- [@webref/idl](#)











TypeScript-DOM-lib-generator の仕組み

最後にパッチを当てて型ファイルを生成

- イベント名のマッピング（文字列リテラル型対応）
- ジェネリクス対応

などなど

 **saschanaz** Update bcd to 5.5.49 (#1792) d4a482f · yesterday  History

Name	Last commit message	Last commit date
 ..		
 idl	update to @webref/idl@3.3.0 (#1288)	2 years ago
 mdn	Verbosity replacers (#659)	5 years ago
 README.md	2nd Minor PR (#1043)	3 years ago
 addedTypes.jsonc	Add midi and storage-access to PermissionName (#1765)	last month
 comments.json	docs: remove invalid documentation (#1666)	8 months ago
 deprecatedMessage.json	chore(deprecatedMessage): remove redundant items	4 years ago
 knownTypes.json	Add override for import.meta.resolve() (#1740)	2 months ago
 overridingTypes.jsonc	Allow null candidate in RTCPeerConnection.addIceCandidate (#1739)	last week
 removedTypes.jsonc	Update bcd to 5.5.49 (#1792)	yesterday

まとめ

TypeScript の組み込み型定義について

- ECMAScript の型の追加、修正は手作業
- Web 標準の型は自動更新
 - Web IDL と Browser Compat Data を使っている