MCP, Nov 2026 to May 2027: the breaking change already shipped

Revision 2026-07-28 is still the current spec. The next date a deprecated feature can leave it is July 28, 2027, after this brief.

Last updated November 10, 2026

Executive summary

The next six months of MCP are a compatibility window. The breaking change already has a date, and that date is in the past. On November 10, 2026 the versioning page still names 2026-07-28 as the current protocol version, 105 days after the revision. A version string in this protocol is the last day a backwards-incompatible change landed. Backwards-compatible edits do not mint a new date. The docs switcher on that page also offers a Draft. This brief does not quote the draft. [1]

Revision 2026-07-28, measured against 2025-11-25, took the session out of the wire. Streamable HTTP no longer has protocol-level sessions or an Mcp-Session-Id header. The initialize handshake is gone. Every request carries its protocol version and client capabilities in _meta, and servers must implement server/discover. The long-lived GET stream is replaced by subscriptions/listen. Server-initiated requests such as elicitation/create, sampling/createMessage, and roots/list are replaced by a result whose resultType is input_required. The client retries the original request with inputResponses. structuredContent may be any JSON value, and inputSchema and outputSchema may use any JSON Schema 2020-12 keyword. OAuth clients must check a present iss against the issuer they recorded, and credentials are bound to that issuer. Dynamic Client Registration is deprecated in favor of Client ID Metadata Documents. [2]

Four features are on a removal clock that does not open inside this brief. The deprecated-features registry says Roots, Sampling, Logging, and Dynamic Client Registration become eligible for removal in the first revision released on or after July 28, 2027. Eligibility is the maintainer's earliest option. The same page says no feature has been removed under the policy. HTTP+SSE, deprecated since 2025-03-26, is listed as removable three months after SEP-2596 reaches Final. The SEP page read the same day has status Final and does not print the day it became Final, so this brief does not turn that sentence into a calendar date. The twelve-month outlook is the page that carries July 28, 2027 next to the other dated marks. [3][29]

The catalog is large and, by its own rules, uncurated. A full walk of the registry API with version=latest, from 10:45:53 to 10:49:01 GMT on November 10, counted 61,146 latest records: 60,183 active and 963 deprecated. That is 1,640 more latest records than this site's November 8 walk of the same endpoint, about 45 hours earlier. Active is a lifecycle flag. The about page says the registry is in preview, that the metadata is deliberately unopinionated, and that host applications should consume a downstream marketplace. A search for aiagentsinfo on that API at 10:49 GMT returned no rows. This site's own MCP server is a public read-only endpoint that the publish step has not put in the official list. [8][9][10]

Install counts and star counts are real, and they measure different objects. From October 11 through November 9, 2026, npm recorded 307,682,080 downloads of @modelcontextprotocol/sdk. The same hour, that package's latest tag was 1.32.2, and its version list had no 2.x. GitHub had already published typescript-sdk v2.4.0 on October 23, split across @modelcontextprotocol/core, server, client, and node. Those four packages drew 102,038,883, 79,656,643, 53,405,727, and 20,311,677 downloads in the same window. pypistats reported 235,193,557 recent last_month downloads for the Python package mcp, whose PyPI version was 2.3.0. A download is a file request. modelcontextprotocol/servers had 91,902 stars. A star on that repository is not a running server. [11][12][13][15]

Clients agree that a server can expose tools. They do not agree, on the pages read for this brief, that they speak 2026-07-28. Claude Code's docs say the v2 runtime, on MCP TypeScript SDK 2.0, adds that revision and falls back for servers that lack it. The OpenAI Agents SDK for Python, when MCP Python SDK v2 is

installed, probes server/discover and falls back to the initialize handshake. Cursor, VS Code, the Responses API guide, Gemini CLI, and Devin Desktop document transports and sign-in. Those five pages do not print a protocol date. Gemini CLI does say it validates the OAuth iss parameter, which is a July revision rule. [16][17][18][19][20][21][22]

The security record inside this source set is a 2025 attack note plus three advisories, not a new CVE opened this week. Invariant Labs, on April 1, 2025, described tool poisoning: instructions in a tool description that the model reads. The specification's own security pages, read November 10, 2026, forbid token passthrough and require per-client consent before an MCP proxy forwards a user to a third-party authorization server. CVE-2025-6514 (mcp-remote, CVSS 9.6, patched in 0.1.16), CVE-2025-49596 (MCP Inspector, CVSS v4 9.4, patched in 0.14.1), and CVE-2025-68143 (mcp-server-git, patched in 2025.9.25) are the three advisories opened here. This is not a claim that the advisory database holds only these three. [23][24][25][26][27][28]

What is dated inside November 2026 through May 2027 is a process date, not a feature list. SEP-3398, status Accepted and dated October 14, 2026, proposes moving the release that had been targeted for December 2026 to March 30, 2027, the last Tuesday of March, subject to a transition calendar. The August 22 roadmap lists five priority areas and says they are not commitments. Governance sits with named maintainers inside a Linux Foundation project series. The December 9, 2025 foundation announcement is the donation. It is not a release schedule. [4][5][6][7]

What 2026-07-28 took out of the handshake

A protocol version here is a date, and the date only moves when a change breaks old peers. The versioning page says 2026-07-28 is current, and that backwards-compatible edits stay on that string. 105 days later, the string has not moved. The changelog measures the revision against 2025-11-25, which is the last handshake-based revision the versioning page names. [1][2]

The wire is stateless

The changes a server author has to implement start with the session. Protocol sessions and the Mcp-Session-Id header are gone, so tools/list, resources/list, and prompts/list no longer vary per connection. State that must survive a call is a handle the server mints and the client passes back as an ordinary argument. The initialize and notifications/initialized handshake is gone. Each request carries io.modelcontextprotocol/protocolVersion and client capabilities in _meta. A mismatch returns UnsupportedProtocolVersionError. Servers must implement server/discover, which advertises supported versions, capabilities, and identity. Clients may call it first, or they may send any request and handle the error. [2]

The HTTP GET stream and resources/subscribe are replaced by subscriptions/listen, one long-lived POST response for the change notifications a client opts into. Progress and log notifications still ride the response stream of the request they belong to. ping, logging/setLevel, and notifications/roots/list_changed are removed. Log level is a per-request _meta field. SSE resume via Last-Event-ID is removed: a broken stream means the client sends a new request with a new id. Experimental tasks left the core protocol for an extension, io.modelcontextprotocol/tasks, polled with tasks/get. [2]

Elicitation comes back as a result, not a server request

Multi round-trip requests replace the old pattern in which a server opened a request of its own, such as elicitation/create, sampling/createMessage, or roots/list. The server now returns resultType input_required, with the questions in inputRequests. The client retries the original call and supplies inputResponses. Ordinary results use resultType complete. Clients must treat a missing resultType from an older server as complete. The elicitation completion notification and the elicitationId field added in 2025-11-25 are removed, because the retry is the completion signal. A server that needs to correlate retries puts its own identifier in requestState. [2]

Two smaller edits matter for anyone caching tools or rendering output. Servers should return tools/list in a stable order, so a prompt cache can hit. inputSchema, outputSchema, and structuredContent are looser: any JSON Schema 2020-12 keyword, and any JSON value for structured content, with bounds on $ref. List and

read results carry ttlMs and cacheScope. Streamable HTTP requests must send Mcp-Method and Mcp-Name. [2]

OAuth got stricter, and one registration path was deprecated

Authorization servers should send iss on the authorization response, per RFC 9207, and clients must check a present iss against the issuer they stored before redeeming the code. A client must key stored credentials by issuer, must not reuse them at a different authorization server, and must re-register when the authorization server changes. Dynamic Client Registration stays available for authorization servers that do not support Client ID Metadata Documents, and new implementations are told not to adopt it. The removal clock for that deprecation is in the next section. [2][3]

"The current protocol version is 2026-07-28." Versioning page, read November 10, 2026. [1]

- Builders: Speak 2026-07-28 on new remote servers, and keep a 2025-11-25 answer until the clients you care about negotiate the new revision. Put cross-call state in a handle argument. Return a stable tools/list. Treat input_required as the elicitation path. Bind OAuth credentials to the issuer.

Who can actually change the next revision

The governance page read on November 10 says the project is Model Context Protocol, a series of LF Projects, LLC. Code and specification contributions use Apache-2.0. Membership in the technical roles is personal. The page says there are no seats reserved for companies. Lead Maintainers hold a veto. The page names David Soria Parra and Den Delimarsky. Core Maintainers named on the same page are Peter Alexander, Caitie McCaffrey, Kurtis Van Gent, Clare Liguori, Paul Carleton, and Nick Cooper. Justin Spahr-Summers is listed as lead maintainer emeritus. [6]

The Linux Foundation's December 9, 2025 release is the donation record. It announces the Agentic AI Foundation, with founding contributions of MCP from Anthropic, goose from Block, and AGENTS.md from OpenAI. That release is how the project acquired a foundation home. The governance page, not the press release, is the document that says who votes on a SEP. Specification changes go through Specification Enhancement Proposals. SEP-3398 is the process SEP that tries to put those votes on a calendar. [6][7][4]

SEP-3398's status on November 10 is Accepted, with an accepted date of October 14, 2026. It is a process SEP. It adds no protocol message. It says specification releases generally fall on the last Tuesday of March and September, on an eight-month cycle with a soft freeze 2.5 months out, an SDK beta 1.5 months out, and a hard freeze one month out. Release managers confirm the exact day in a published calendar. The transition sentence is the one that matters for this brief: move the release previously targeted for December 2026 to March 30, 2027, subject to that calendar. The page says managers publish the calendar within two weeks of adoption. This brief did not find a page that is that calendar. [4]

The roadmap is older than the SEP. It says last updated 2026-08-22, and it says the text is current thinking rather than firm commitments. Five areas are named: agentic messaging primitives, HTTP-native transport unification (including HTTP over stdio and ETags), agent identity (DPoP, workload identity, token exchange), a cleaner tools/call result and progressive discovery, and SDK generation from the spec plus conformance tests. Those are the maintainers' review priorities. They are not the table of contents of a March 30 release. [5]

March 30 waits on a calendar. SEP-3398 names March 30, 2027 and then says the day waits on a transition calendar. Until that calendar is public, March 30 is the maintainers' accepted proposal. It is not a revision you can pin in a client.

- Buyers: Ask a vendor which revision their client negotiates today. A foundation membership and a roadmap area are not a feature on your contract. The people who can change the spec are the maintainers named on the governance page, through a SEP.

The registry proves a namespace. It does not review a server

The about page calls the registry the official centralized metadata repository for publicly accessible MCP

servers, backed by Anthropic, GitHub, PulseMCP, and Microsoft. It is in preview. The page says breaking changes or a data reset can still happen. A server name is reverse-DNS, such as io.github.user/server-name, and the publisher proves the GitHub account or the domain with GitHub, DNS, or an HTTP challenge. The record is a server.json: name, where the package or remote URL lives, how to run it, and a description. Private servers are out of scope. The page says to run a private registry for those. [8]

Two sentences on that page should decide how you use the list. The metadata is deliberately unopinionated, and downstream aggregators are where curation and ratings belong. Host applications are not supposed to consume the official registry directly. They are supposed to use a marketplace that implements the same OpenAPI shape. Security scanning is delegated to npm, PyPI, Docker Hub, and to those aggregators. The registry's own controls are namespace authentication, field limits, and manual takedown. [8]

"The metadata hosted by the MCP Registry is deliberately unopinionated." MCP Registry about page, read November 10, 2026. [8]

The walk behind the chart followed nextCursor with version=latest and limit=100 until the cursor ended. It started at 10:45:53 GMT and finished at 10:49:01 GMT, 612 pages, 61,146 records. 60,183 were active and 963 were deprecated. 60,183 + 963 = 61,146. Active share is 98.4 percent. This site's November 8 walk, same filter, counted 59,506. The difference is 1,640. The earlier page also showed the list moving by two records inside a few minutes, and it showed one namespace holding 2,366 records that pointed at one repository. This November 10 walk did not re-count namespaces. Use that page for the shape of the list. Use this one for the size on this date. [9][10]

A search of the same API for aiagentsinfo at 10:49:59 GMT returned zero servers. This site publishes a read-only Streamable HTTP server at /api/mcp and says the official-registry publish step is still a domain proof plus mcp-publisher. A public URL and a row in the official registry are different facts. The prompt and tool checker validates MCP tool JSON. It does not list servers. [9]

- Builders and buyers: Publish under a namespace you can prove, with a server.json a marketplace can pull. When you connect a server, connect one a vendor or your own review names. A row count of 61,146 is the size of the phone book. The about page tells hosts not to dial every number.

Latest server records from one walk of the official registry API, 10:45:53 to 10:49:01 GMT on November 10, 2026. 98.4 percent were marked active.

	Records
Active	60,183
Deprecated	963

Clients agree on tools. They do not agree on the revision

Support, on the pages below, means a documented way to reach a server and call its tools. It does not mean the client has implemented every major change from July 28. Where a page names a revision, the cell says so. Where it does not, the cell says the page does not. These pages were read on November 10, 2026. [16][17][18][19][20][21][22]

Claude Code is the page that names the new revision in public. The v2 runtime is described as the same client on MCP TypeScript SDK 2.0, which adds 2026-07-28. The doc says that runtime is selected when feature flags are fetched, on Claude Code v2.1.232 or later, and by default in listed session types on v2.1.274 or later. On v2 it asks HTTP, stdio, and claude.ai connector servers whether they support the newer revision, and it connects to every other server as v1 does. An operator can force v1 or v2 with MCP_SDK_GENERATION, and can force legacy negotiation with MCP_PROTOCOL_NEGOTIATION. SSE is marked deprecated. On v2.1.265 or later the HTTP transport is tried first and SSE is the fallback. [16]

The Agents SDK draws the same line from the other direction. It says the installed mcp package version is separate from the protocol version negotiated with a server. With Python SDK v2 it sends server/discover at the newest version the SDK supports. If the server does not implement that method, the client uses the initialize handshake and the version negotiated there. Installing SDK v2 does not force every connection onto 2026-07-28. That is the dual stack, written down by the SDK that many hosted agents import. [19]

The Windsurf docs URL is a redirect, which is worth recording because buyers still say Windsurf. On November 10, https://docs.windsurf.com/windsurf/cascade/mcp returned HTTP 308 to https://docs.devin.ai/desktop/cascade/mcp. The markdown for that path resolved to the Devin Local customizations page. That page documents MCP for the Devin Local agent in Devin Desktop, with project config at .devin/mcp_config.json and user config at ~/.config/devin/mcp_config.json. It also says legacy .windsurf rule files are imported. The page does not name a protocol revision. [22]

Ask which revision the client speaks. Seven products document MCP tools. Two of the pages describe a probe of 2026-07-28 with a fallback to the old handshake. Five do not print a protocol date. Ask for the negotiation, then read the trace.

- Builders: If your server is remote, implement Streamable HTTP and server/discover, and keep the initialize answer for clients that still send it. If your user is in Claude Code, the v2 runtime is the one that will try 2026-07-28. If your user is in Cursor, VS Code, Gemini CLI, or Devin Desktop, the page you can cite today promises a transport and a sign-in, not that revision.

What seven official client pages name. Read November 10, 2026.

Client	Transports the page names	Sign-in the page names	Protocol revision
[Claude Code](https://code.claude.com/docs/en/mcp)	HTTP, recommended. SSE, deprecated. stdio. WebSocket.	OAuth, bearer header. On the v2 runtime, an unexpected iss fails the sign-in.	v2 runtime adds 2026-07-28 and falls back. [[16]]
[Cursor](https://cursor.com/docs/mcp)	stdio, SSE, Streamable HTTP.	OAuth with PKCE. Static client credentials when dynamic registration is unavailable.	Not named on the page. [[17]]
[VS Code](https://code.visualstudio.com/docs/agent-customization/mcp-servers)	HTTP, including Server-Sent Events. Command, stdio.	Publisher sign-in when the server requires it. Trust prompt before install.	Not named on the page. [[18]]
[Agents SDK, Python](https://openai.github.io/openai-agents-python/mcp/)	Hosted MCP, Streamable HTTP, SSE, stdio.	Authorization header. Hosted tools run on the Responses API.	SDK v2 probes server/discover, then the old handshake. [[19]]
[Responses API](https://developers.openai.com/api/docs/guides/tools-remote-mcp)	Remote server_url. Secure MCP Tunnel for a private server.	OAuth access token in authorization. Approval can be required or skipped.	The fetched page does not contain 2026-07-28. [[20]]
[Gemini CLI](https://geminicli.com/docs/tools/mcp-server/)	stdio, SSE, Streamable HTTP.	OAuth discovery on a 401. Validates iss per RFC 9207.	No protocol date on the page. [[21]]
[Devin Desktop](https://docs.devin.ai/desktop/devin-local/customizations)	stdio, Streamable HTTP, SSE.	OAuth for each transport. Approval before a tool call, by default.	Not named on the page. [[22]]

The counts you can check, and what they leave out

Three public counters moved on November 10, and none of them is a count of agents in production. They are worth printing because each one has a URL, a clock time, and a unit. [11][12][13][14]

The tall bar is the package whose latest npm tag is still 1.32.2. 307,682,080 file requests from October 11 through November 9. GitHub published typescript-sdk v2.4.0 on October 23, the same day as tag 1.32.2, and the 2.x line on GitHub is split into @modelcontextprotocol/core 102,038,883, server 79,656,643, client 53,405,727, and node 20,311,677. Adding those four bars treats overlapping installs as one population, so this page leaves them side by side. Python is a different registry. pypistats last_month for mcp was 235,193,557, last_week 79,237,289, last_day 11,679,389, retrieved 10:45 GMT. PyPI's project record said version 2.3.0, matching the GitHub release v2.3.0 published October 2. The pypistats JSON does not say it excluded mirrors, and it does not split versions. [12][13][15]

The servers repository leads at 91,902 stars. It is the reference-server repo. python-sdk is next at 24,820. The specification repository, modelcontextprotocol/modelcontextprotocol, had 9,655. Stars are a reason the repository is easy to find. They are not a concurrency number and not a security review. [11]

GitHub's search index is a third object. At 10:47:15 GMT, topic:mcp-server returned total_count 42,138, with incomplete_results false. This site's count of the same query at 03:35 UTC the same day was 42,072. The difference is 66. A topic is a label a repository owner sets. The topic note explains why

that label and the registry list are not the same census. The registry figure on November 10 is 19,008 higher than this topic were taken about three hours apart. [14][30][9]

- Buyers: Quote a download with the window and the package name. Quote a star with the repository. Quote the registry with the minute the walk finished. A vendor slide that adds these three numbers together is adding a file request, a bookmark, and a metadata row.

npm downloads for five official packages over the same 30-day window. @modelcontextprotocol/sdk's latest tag was 1.32.2. The other four are the split 2.x packages. The API does not split by version, so a bar is the package, not one release.

	Downloads
sdk (latest 1.32.2)	307,682,080
core	102,038,883
server	79,656,643
client	53,405,727
node	20,311,677

The attacks that are already written down

The specification's security pages, read November 10, are about authorization. The attack that shows up in tool descriptions was written up earlier, by a lab, and it is still the threat model for any client that pastes tools/list into the prompt. This chapter keeps those apart. [23][24][25]

Tool descriptions are part of the prompt

Invariant Labs, on April 1, 2025, named a tool poisoning attack: instructions embedded in an MCP tool description that the model sees and a simplified user interface often hides. Their write-up says a malicious server can steer the agent, including across tools the user trusts. The July 28 changelog tells servers to return tools in a stable order so prompt caches hit. That is a performance rule. It also means the description stays in the context window. The official security best-practices page read for this brief does not use the phrase tool poisoning. The attack stands as a published demonstration, not as a CVE number. A related failure mode, agents claiming success when the MCP tools cannot do the job, is measured in the MCPHallu note. That paper is a benchmark of hallucinations. It is not a vulnerability advisory. [2][25]

Confused deputy, and the token you must not forward

The security best-practices page describes a confused deputy that appears when an MCP proxy uses one static client id at a third-party authorization server, lets MCP clients register dynamically, and relies on a consent cookie. A later request can skip the consent screen and deliver an authorization code to an attacker's redirect. The mitigation on that page is per-client consent at the MCP server before the third-party redirect. The normative security considerations repeat it: a proxy using a static client id must obtain user consent for each dynamically registered client before forwarding. The same considerations say a server must accept only tokens intended for itself, and must not pass the client's token through to an upstream API. The best-practices page says token passthrough is explicitly forbidden. Dynamic Client Registration is the mechanism in that attack write-up, and it is also the mechanism the July revision deprecated in favor of Client ID Metadata Documents. Deprecation is not removal. The earliest removal is still July 28, 2027. [3][23][24]

The packages in the first two rows are still downloaded. In the October 11 to November 9 window, mcp-remote had 4,805,171 npm downloads and @modelcontextprotocol/inspector had 2,599,288. The downloads API does not split versions, so those figures are not a count of machines still on 0.1.15 or 0.14.0. They are a reason to pin the patched release if you still shell out to either package. [12][26][27]

- Builders and buyers: Show the full tool description and the full tool call before approval. Allowlist servers. Require an audience-bound token and refuse token passthrough. Pin mcp-remote at 0.1.16 or newer and MCP Inspector at 0.14.1 or newer if they are still in the install path. Do not treat a July 2025

patch as something the download counter has enforced.

Three advisories opened in the GitHub Advisory Database on November 10, 2026. This is not an exhaustive CVE list.

Advisory	Package	What the advisory says	Patch
[CVE-2025-6514](https://github.com/advisories/GHSA-6xpm-ggf7-wc3p)	mcp-remote (npm)	Critical, CVSS 9.6. Command injection from a crafted authorization_endpoint when connecting to an untrusted server. Published July 9, 2025.	0.1.16. Affected: >= 0.0.5 and < 0.1.16. [[26]]
[CVE-2025-49596](https://github.com/advisories/GHSA-7f8r-222p-6f5g)	@modelcontextprotocol/inspector	Critical, CVSS v4 9.4. No authentication between the Inspector client and proxy, so a request can launch stdio commands. Published June 13, 2025.	0.14.1. Affected: below 0.14.1. [[27]]
[CVE-2025-68143](https://github.com/advisories/GHSA-5cgr-j3jf-jw3v)	mcp-server-git (pip)	Medium on the advisory. git_init accepted any filesystem path. Published December 17, 2025. The tool was removed.	2025.9.25. Affected: below that version. [[28]]

What is dated from November 2026 through May 2027

Announced means a linked spec page, SEP, or maintainer document says it. Likely means that document aims the work at this stretch and does not lock a day. Watch means a later check should confirm the item or drop it. A blank cell means no source read for this brief named that month. The general six-month outlook covers product filings and regulation. This table is the protocol only. [4][5]

Announced

The versioning page names one current revision, 2026-07-28, and the deprecated-features registry names one future eligibility day, July 28, 2027, which falls after May 2027. Inside the window, the dated proposal is March 30, 2027, in an Accepted process SEP, subject to a calendar. The SEP also says planning for September 2027 begins in late January 2027. The roadmap's five areas are written down and explicitly not committed. The registry about page says the registry is in preview and names no general-availability day. [1][3][4][5][8]

Likely

Likely, in the sense this site uses, is the March release sitting in this window because the accepted SEP aims it there, and the late-January planning phrase for the following cycle. It is not a probability from this desk. The feature content of that release is the part the roadmap refuses to promise. HTTP-over-stdio, ETags, DPoP, a redesigned tools/call result, and progressive discovery are the areas the August 22 page says maintainers will review first. A server author who builds all five before the SEP is Final will be building ahead of the spec. [4][5]

Watch for

Watch the transition calendar. SEP-3398 says it gets published within two weeks of adoption, and the accepted date on the page is October 14, 2026. This brief did not find it, so March 30 can still move. Watch whether any client page that was silent on the revision starts naming 2026-07-28. Watch the removed section of the deprecated-features registry, which was empty on November 10. Watch an expedited-removal SEP: the policy allows a floor of 90 days for an active security risk, and no such shortening was on the registry. Watch the registry leaving preview. The about page does not date that. The rendered board at plan.modelcontextprotocol.io was not used. Its HTML shell contained no dates. [3][4][8][29]

- Builders and buyers: Plan the next six months as dual-stack support on 2026-07-28 plus 2025-11-25. Put March 30, 2027 on the watch list, not in the release train, until the transition calendar confirms the day. Put July 28, 2027 on the removal watch list, and read it on the twelve-month page.

Announced, likely, and watch-for notes by month, November 2026 through May 2027.

Month	Announced	Likely	Watch for
November 2026	The versioning page still names 2026-07-28 as the current protocol version. SEP-3398 is Accepted, with an accepted date of October 14, 2026. No specification revision is dated inside November.	No source aims a protocol change at a day in the rest of November.	SEP-3398 says release managers publish a transition calendar within two weeks of adoption. This brief did not find that calendar.
December 2026	SEP-3398 says it proposes moving the release previously targeted for December 2026. December is the month being left, not a ship month, in that sentence.	The accepted SEP does not aim the next revision at December.	A calendar that puts a revision back in December would supersede the March 30 proposal. None was in the SEP text read November 10.
January 2027	SEP-3398 says planning for the September 2027 release begins in late January 2027. It does not name the day.	Late January is the phrase the accepted SEP uses. It is still subject to the transition calendar.	The published calendar, if it appears, either confirms late January or replaces it.
February 2027	No source read for this brief names a February protocol date.	February sits between the late-January planning phrase and the proposed March 30 release. The SEP does not assign February a milestone by name.	Soft-freeze and SDK-beta offsets in SEP-3398 are measured back from a confirmed release day. Without the calendar, those offsets stay arithmetic, not dates.
March 2027	SEP-3398 proposes March 30, 2027, the last Tuesday of March, for the release previously targeted for December 2026. The date is subject to confirmation in the transition calendar. The SEP status is Accepted, not Final.	The roadmap of August 22, 2026 lists five priority areas for the next revision and says they are not commitments. It does not assign them to March 30.	Whether release managers confirm March 30, move it, or publish the frozen scope. A confirmed day is a later check, not a fact this page can add.
April 2027	No source read for this brief names an April protocol release.	April is after the proposed March release and is not given a milestone in SEP-3398.	The next removal eligibility on the deprecated-features registry is July 28, 2027, which the twelve-month outlook carries. April is not that day.
May 2027	No source read for this brief names a May protocol release. May is the end of this brief.	May is inside the compatibility window and is not a named deadline.	July 28, 2027 remains the first day Roots, Sampling, Logging, and Dynamic Client Registration can become eligible for removal. Eligibility is not deletion.

What to pin, and what to refuse

The tool layer under an agent harness is the set of servers the harness is willing to call, the transport those calls use, and the revision both sides negotiate. MCP is that layer's common port. The port's incompatible change shipped on July 28, 2026. From November 10, 2026 through May 2027 the public record adds a proposed release date, a preview catalog that grew by 1,640 latest rows in about 45 hours, and client docs that mostly still describe transports rather than the new revision. [1][4][9]

Builders should ship the new revision and the old handshake together. Implement server/discover, stateless Streamable HTTP, stable tool lists, structured content, and input_required for anything that used to be an elicitation callback. Keep initialize working for the clients whose docs do not yet name 2026-07-28. Skip Roots, Sampling, and Logging in new code. The migration on the registry is ordinary tool arguments, a direct provider call, and stderr or OpenTelemetry. Prefer Client ID Metadata Documents, and keep Dynamic Client Registration only for authorization servers that have not moved. Publish a server.json under a namespace you control. Point buyers at a marketplace or an allowlist, not at the official registry walk. [2][3][8]

Buyers should write three questions into the order. Which protocol revision does the client negotiate, and what does the trace show when the server only speaks 2025-11-25? Which servers are allowlisted, and who approves a new one? Are tool calls shown in full before they run? A connector

count, a star count, and a registry total do not answer those questions. Claude Code and the OpenAI Agents SDK are the describe the fallback in public. Cursor, VS Code, Gemini CLI, Devin Desktop, and the Responses API document how to at the revision separately. [16][17][18][19][20][21][22]

Refuse three shortcuts. Refuse a prompt that says the registry has 61,146 safe servers. The about page says the opposite. Refuse an install of mcp-remote or MCP Inspector that is not pinned at or above the patched versions in CVE-2025-6514 and CVE-2025-49596. Refuse a design that forwards the MCP client's access token to an upstream API. The specification forbids it. The general agent outlook and the 2027 forecast cover products and task cost. They are the wrong pages for the revision pin. [8][24][26][27]

- Builders and buyers: Pin revision 2026-07-28, keep the previous handshake through May 2027, allowlist servers, and treat March 30 as a date to re-check. The breaking change you needed to absorb is the one that already shipped.

Methodology

This page did not field a survey, recruit a panel, or interview vendors. It copies public pages and APIs and labels every arithmetic step. There is no hidden sample of practitioners.

What was counted

The registry walk requested https://registry.modelcontextprotocol.io/v0.1/servers with version=latest and limit=100, followed metadata.nextCursor, and counted _meta["io.modelcontextprotocol.registry/official"].status. It ran once, from 10:45:53 to 10:49:01 GMT on November 10, 2026, 188.4 seconds, 612 pages. 60,183 + 963 = 61,146. The active share is 60,183 / 61,146, shown to one decimal as 98.4 percent. The difference from the November 8 walk is 61,146 59,506 = 1,640. The November 8 figure is that page's Pass A, not a recount. [9][10]

Days from the revision to this brief are the UTC date difference from 2026-07-28 to 2026-11-10, which is 105. npm figures are the downloads field from the point API for 2026-10-11:2026-11-09. pypistats figures are the three fields in the recent endpoint. GitHub stars are stargazers_count. The topic total is total_count for q=topic:mcp-server. The 66 gap versus 03:35 UTC uses the figure published on this site's topic note. [11][12][13][14][30]

What this does not measure

It does not measure how many of the 61,146 records answer a handshake, how many namespaces are spam, or how many distinct codebases they point at. Those questions are on the November 8 page, and this walk did not repeat them. It does not split npm or PyPI downloads by version, so it does not say how many of the 307.7 million requests installed 1.32.2. It does not claim the pypistats recent count excludes mirrors, because the JSON does not say. It did not install Claude Code, Cursor, VS Code, Gemini CLI, or Devin and capture a negotiation trace. A page that omits a protocol date may still negotiate one.

It did not read the Draft spec. It did not use the rendered release board at plan.modelcontextprotocol.io, because the HTML shell had no dates. It did not search the entire CVE corpus. The three advisories are the three this brief opened and checked against the GitHub Advisory Database. It did not use Ahrefs.

Forward-looking cells are labeled announced, likely, or watch-for, and a month stays empty when no source names it.

- Builders and buyers: If you cite this page, cite the source number and the clock time. The walk, the download window, and the pages read are the evidence. The sorting of March 30 as a proposal, rather than a ship date, is the judgment.

Sources

[1] Model Context Protocol versioning, revision 2026-07-28. Read November 10, 2026. The page says the current protocol version is 2026-07-28. The docs version switcher on that fetch lists 2026-07-28 (latest), 2025-11-25, 2025-06-18, 2025-03-26, 2024-11-05, and Draft. JSON-LD dateModified on the page was 2026-11-09. https://modelcontextprotocol.io/docs/2026-07-28/learn/versioning

[2] Key changes, specification 2026-07-28. Read November 10, 2026. Changes since revision 2025-11-25, including the stateless transport, server/discover, multi round-trip requests, structured output, and the OAuth changes. https://modelcontextprotocol.io/specification/2026-07-28/changelog
[3] Deprecated features registry, specification 2026-07-28. Read November 10, 2026. Roots, Sampling, Logging, and Dynamic Client Registration are eligible for removal in the first revision released on or after 2027-07-28. The page says no feature has been removed under the policy yet. https://modelcontextprotocol.io/specification/2026-07-28/deprecated
[4] SEP-3398: Release Cycle Updates. Read November 10, 2026. Status on the page: Accepted. Accepted date printed: 2026-10-14. The transition section proposes March 30, 2027, subject to confirmation in the transition calendar. https://modelcontextprotocol.io/seps/3398-release-cycle-updates
[5] MCP roadmap. Read November 10, 2026. The page says it was last updated 2026-08-22 and that the roadmap is current thinking rather than firm commitments. https://modelcontextprotocol.io/development/roadmap
[6] Governance and stewardship. Read November 10, 2026. Names the project as a series of LF Projects, LLC, and lists the lead and core maintainers. https://modelcontextprotocol.io/community/governance
[7] Linux Foundation, formation of the Agentic AI Foundation. Press release dated December 9, 2025. Founding contributions include Anthropic's Model Context Protocol, Block's goose, and OpenAI's AGENTS.md. https://www.linuxfoundation.org/press/linux-foundation-announces-the-formation-of-the-agentic-ai-foundation
[8] The MCP Registry. Read November 10, 2026. Preview status, namespace authentication, and the instruction that host applications should not consume the official registry directly. https://modelcontextprotocol.io/registry/about
[9] Official MCP Registry API, latest servers. One full cursor walk, HTTP Date 10:45:53 GMT through 10:49:01 GMT on November 10, 2026. 612 pages, 61,146 latest records: 60,183 active and 963 deprecated. A search for aiagentsinfo at 10:49:59 GMT returned no servers. https://registry.modelcontextprotocol.io/v0.1/servers?limit=100&version=latest
[10] AI Agents Info, how many official MCP servers are there?. Earlier walk of the same API, 13:43:33 to 13:45:25 UTC on November 8, 2026: 59,506 latest records, 58,565 active, 941 deprecated. https://www.aiagentsinfo.com/blog/how-many-official-mcp-servers-are-there
[11] GitHub REST API, stargazers_count for official MCP repositories. Thirteen repository objects, HTTP Date 10:45 GMT on November 10, 2026. The table on this page is that pull. pushed_at is copied from the same objects. https://api.github.com/repos/modelcontextprotocol/servers
[12] npm downloads API, point range October 11 through November 9, 2026. Retrieved 10:45 and 10:47 GMT on November 10, 2026. @modelcontextprotocol/sdk 307,682,080; core 102,038,883; server 79,656,643; client 53,405,727; node 20,311,677; mcp-remote 4,805,171; @modelcontextprotocol/inspector 2,599,288. A download is a file request. The API does not split versions. https://api.npmjs.org/downloads/point/2026-10-11:2026-11-09/@modelcontextprotocol/sdk
[13] pypistats recent downloads for the mcp package, and the PyPI project record. pypistats retrieved 10:45:10 GMT on November 10, 2026: last_day 11,679,389; last_week 79,237,289; last_month 235,193,557. The JSON does not say whether mirrors are excluded. PyPI project JSON the same hour: version 2.3.0, repository https://github.com/modelcontextprotocol/python-sdk. https://pypistats.org/api/packages/mcp/recent
[14] GitHub search API, topic:mcp-server. HTTP Date 10:47:15 GMT on November 10, 2026. total_count 42,138. incomplete_results false. per_page was 1. This is a search total, not a census of running servers. https://api.github.com/search/repositories?q=topic:mcp-server
[15] GitHub releases, TypeScript SDK and Python SDK. Read via the GitHub releases API at 10:47 GMT on November 10, 2026. typescript-sdk tag v2.4.0 published 2026-10-23, and tag 1.32.2 published the same day. python-sdk tag v2.3.0 published 2026-10-02. npm registry metadata for @modelcontextprotocol/sdk, same hour: dist-tag latest is 1.32.2, and the version list contained no 2.x. https://github.com/modelcontextprotocol/typescript-sdk/releases
[16] Claude Code, connect to tools via MCP. Read November 10, 2026. Documents HTTP, deprecated SSE,

stdio, and WebSocket. The v2 runtime section says MCP TypeScript SDK 2.0 adds protocol revision 2026-07-28. https://code.claude.com/docs/en/mcp
[17] Cursor, Model Context Protocol. Read November 10, 2026. Documents stdio, SSE, and Streamable HTTP, plus OAuth with PKCE and static client credentials. The page does not name a protocol revision. https://cursor.com/docs/mcp
[18] VS Code, add MCP servers. Read November 10, 2026. The add-server flow offers HTTP (HTTP or Server-Sent Events) or a stdio command, with a trust warning. The page does not name protocol revision 2026-07-28. https://code.visualstudio.com/docs/agent-customization/mcp-servers
[19] OpenAI Agents SDK for Python, Model context protocol. Read November 10, 2026. Documents hosted MCP, Streamable HTTP, SSE, and stdio. With MCP Python SDK v2 the client probes server/discover and falls back to the initialize handshake. https://openai.github.io/openai-agents-python/mcp/
[20] OpenAI, remote MCP servers on the Responses API. Read November 10, 2026. Remote MCP servers are public servers. Secure MCP Tunnel covers private servers. The fetched page does not contain the string 2026-07-28. https://developers.openai.com/api/docs/guides/tools-remote-mcp
[21] Gemini CLI, MCP servers. Read November 10, 2026. Documents stdio, SSE, and Streamable HTTP, OAuth discovery, and iss validation per RFC 9207 and the MCP specification. The page does not name a protocol date. https://geminicli.com/docs/tools/mcp-server/
[22] Devin Desktop, Devin Local customizations. Read November 10, 2026. Documents stdio, Streamable HTTP, and SSE, with OAuth, and approval before a tool call. https://docs.windsurf.com/windsurf/cascade/mcp returned HTTP 308 to https://docs.devin.ai/desktop/cascade/mcp. The markdown URL for that path resolved to this customizations page. The page still imports legacy .windsurf rule files. https://docs.devin.ai/desktop/devin-local/customizations
[23] MCP security best practices. Read November 10, 2026. Confused-deputy conditions for an MCP proxy, and the statement that token passthrough is explicitly forbidden. https://modelcontextprotocol.io/docs/2026-07-28/tutorials/security/security_best_practices
[24] Authorization security considerations, specification 2026-07-28. Read November 10, 2026. Per-client consent before forwarding to a third-party authorization server, audience checks, and the rule that an MCP server must not pass a client token through to an upstream API. https://modelcontextprotocol.io/specification/2026-07-28/basic/authorization/security-considerations
[25] Invariant Labs, MCP security notification: tool poisoning attacks. Dated 2025-04-01 on the page. Defines a tool poisoning attack as instructions embedded in tool descriptions that the model reads and the user often does not. https://invariantlabs.ai/blog/mcp-security-notification-tool-poisoning-attacks
[26] GitHub Advisory GHSA-6xpm-ggf7-wc3p, CVE-2025-6514. Read November 10, 2026. mcp-remote, critical, CVSS 9.6, affected >= 0.0.5 and < 0.1.16, patched in 0.1.16. Published 2025-07-09. OS command injection via a crafted authorization_endpoint URL. https://github.com/advisories/GHSA-6xpm-ggf7-wc3p
[27] GitHub Advisory GHSA-7f8r-222p-6f5g, CVE-2025-49596. Read November 10, 2026. @modelcontextprotocol/inspector below 0.14.1, critical, CVSS v4 score 9.4. Patched in 0.14.1. Published 2025-06-13. Missing authentication between the Inspector client and proxy. https://github.com/advisories/GHSA-7f8r-222p-6f5g
[28] GitHub Advisory GHSA-5cgr-j3jf-jw3v, CVE-2025-68143. Read November 10, 2026. mcp-server-git below 2025.9.25, severity medium on the advisory. Patched in 2025.9.25 by removing git_init. Published 2025-12-17. https://github.com/advisories/GHSA-5cgr-j3jf-jw3v
[29] SEP-2596: specification feature lifecycle and deprecation policy. Read November 10, 2026. Status on the page: Final. The header table prints Created 2026-04-17 and does not print the date the SEP reached Final. HTTP+SSE earliest removal in the policy text is three months after the SEP is Final. https://modelcontextprotocol.io/seps/2596-spec-feature-lifecycle-and-deprecation
[30] AI Agents Info, MCP server meaning and the GitHub topic count. topic:mcp-server total_count 42,072 at 03:35:36 UTC on November 10, 2026, incomplete_results false. https://www.aiagentsinfo.com/blog/mcp-server-meaning-42072-github-repos