Motor 0.38.4 (24/05/2019) - Somente Windows 64 bits

O Chromium de 69 para 74 promoveu a integração de otimizações de performance inerentes ao navegador, bem como a disponibilização de novas APIs web padronizadas. Paralelamente, a atualização do Node.js da versão 10.11.0 para as linhas 11.x e 12.x (culminando na v12.3.1) expandiu o conjunto de funcionalidades e aprimorou o desempenho do runtime JavaScript acessível às aplicações.

Essa estratégia de atualização contínua garante a compatibilidade com as tecnologias web mais recentes, fortalece a segurança através da incorporação de correções provenientes dos projetos upstream e otimiza a performance das aplicações. A adoção das novas APIs do Chromium (e.g., recursos multimídia, APIs de interface, funcionalidades de segurança) e do Node.js (e.g., melhorias no sistema de arquivos, novas APIs assíncronas) permite aos desenvolvedores explorar capacidades avançadas em seus aplicativos desktop híbridos.

Chromium 74.0.3729.169

Esta versão do navegador focou significativamente em aprimorar a segurança e a privacidade dos usuários. Protocolos de segurança mais antigos e vulneráveis, como TLS 1.0 e 1.1, foram removidos para garantir conexões mais seguras. Adicionalmente, o navegador implementou medidas mais rigorosas contra downloads inseguros iniciados em iframes e ofereceu uma opção para desativar o acesso a sensores por padrão, aumentando o controle do usuário sobre suas informações. A proteção contra o autoplay abusivo de mídia também foi reforçada para uma experiência de navegação menos intrusiva. Em contrapartida, o HTTP Public Key Pinning (HPKP), considerado uma funcionalidade complexa e com baixo índice de adoção, foi removido.

No âmbito das funcionalidades web, diversas APIs e recursos foram implementados ou aprimorados. Houve a introdução do WebAuthn Level 2 para autenticação mais segura e fácil de usar. Várias propriedades CSS modernas foram adicionadas, como scroll-snap-align, display: contents, overscroll-behavior e shape-outside, oferecendo mais flexibilidade no layout e na experiência visual. APIs JavaScript importantes como Intersection Observer v2, Intl.DisplayNames, User Activation e Element Timing foram implementadas, permitindo aos desenvolvedores criar aplicações web mais performáticas e com melhor análise de comportamento do usuário. As políticas de segurança de cookies SameSite e Feature Policy (agora Permissions Policy) receberam aprimoramentos para maior controle sobre o comportamento de cookies e recursos do navegador. O suporte para Private Fields e Methods em JavaScript (ES2019) também foi adicionado.

As ferramentas de desenvolvimento do navegador também receberam melhorias significativas para auxiliar na criação e depuração de aplicações web modernas. A aba Network agora permite a inspeção de Service Workers e da Cache API, facilitando o entendimento do comportamento de aplicações offline e o gerenciamento de cache. A depuração direta de WebAssembly foi introduzida, permitindo aos desenvolvedores inspecionar e depurar código WebAssembly diretamente nas ferramentas. A aba Performance ganhou novas visualizações e insights para uma análise mais detalhada do desempenho das aplicações. Um novo painel "Changes" permite rastrear modificações no DOM ao longo do tempo. A aba Sources foi aprimorada com funcionalidades como breakpoints condicionais e a capacidade de "step over" em código assíncrono (async/await). Por fim, um painel "Rendering" foi adicionado para ajudar a identificar e corrigir problemas de layout shifts, contribuindo para um carregamento e renderização de páginas mais suaves e eficientes.

NodeJS v12.3.1

A versão 12 LTS do Node.js trouxe consigo uma série de avanços significativos, tanto em termos de novas funcionalidades quanto de estabilidade e desempenho. No que diz respeito à linguagem JavaScript, foram incorporados os novos recursos do ECMAScript 2019 (ES2019), como Object.fromEntries para converter listas de pares chave-valor em objetos, métodos para remover espaços em branco no início e/ou fim de strings (trimStart/trimEnd), funcionalidades para achatamento de arrays (flat/flatMap) e acesso à descrição de Symbols (Symbol.description). Além disso, houve a estabilização da API fs.promises, que oferece uma interface baseada em Promises para operações de sistema de arquivos, e melhorias notáveis no desempenho geral das Promises.

Um dos destaques desta versão foi a introdução (ainda em caráter experimental) de Worker Threads, permitindo a execução de código JavaScript em threads separadas, o que pode melhorar significativamente o desempenho de aplicações que realizam tarefas computacionalmente intensas. Também foi dado suporte inicial (experimental) ao WebAssembly Threads, abrindo caminho para a execução de código WebAssembly de forma paralela. No quesito segurança, a estabilização do suporte ao protocolo TLS 1.3 elevou o nível de proteção das comunicações de rede. A atualização da engine V8 para as versões 7.0, 7.1, 7.2 e 7.4 trouxe consigo ganhos de performance e a implementação de novos recursos JavaScript, alinhando o Node.js com as mais recentes evoluções da linguagem.

Outras melhorias importantes incluem refinamentos na performance e confiabilidade do fs.watch para monitoramento de alterações no sistema de arquivos, e a introdução de novas APIs experimentais como Performance Hooks e diagnostics_channel para melhor observabilidade e diagnóstico de aplicações, além da tag <input type="module"> para carregamento de módulos ES em contextos HTML. O foco principal da versão 12 como LTS foi a estabilidade, resultando em um grande número de correções de bugs e refinamentos gerais em todo o sistema. Por fim, houve atualizações importantes nas dependências internas, como libuv e c-ares, que contribuíram para aprimorar a infraestrutura subjacente do Node.js.

Revision #5
Created 2025-04-15 13:13:05 UTC by Heron Santos
Updated 2026-03-19 14:34:04 UTC by Heron Santos