KDE Planet - Latest News

- Transitous Hack Weekend January 2025 (2026/01/17 10:15)
  Last weekend I attended a rather spontaneous Transitous Hack Weekend in Berlin, again hosted at Wikimedia's WikiBär. Topics Elevator data Data about current and planned future elevator outages is becoming available in a standardized format (SIRI SX and SIRI FM), in Germany and Switzerland at least. That's crucial information especially for wheelchair routing, and MOTIS, the routing engine used by Transitous, can take this into consideration. However, it doesn't support the SIRI format yet. The main challenge here is identifying the affected elevator: The German SIRI FM data uses DIID identifiers for elevators. While those also appear in the OpenStation dataset, we currently have neither geographic coordinates for them nor are DIIDs referenced in OpenStreetMap. This leaves us with no practical way to map the elevator status information to an elevator in the OSM data for routing, at this point. The Swiss SIRI SX data uses OSM node/way ids for elevators. While there are concerns about the stability of those as identifiers, this should nevertheless work sufficiently in the majority of cases. NeTEx So far all static schedule data used by Transitous is using the GTFS format. That's a relatively simple set of CSV files in a ZIP. There's a another format for this though, NeTEx. It's a vastly more complex and rather verbose XML, but it can also model a number of things that cannot be represented in GTFS so far, such as vehicle attributes. MOTIS v2.8 added initial support for NeTEx, but due to the complexity of the format and its tendency to offer multiple different ways to model the same thing it remains a case-by-case investigation whether a specific NeTEx feed is working sufficiently well. We looked at three feeds that seem particularly promising at this point. While using NeTEx there would give us clear benefits, all of them would also introduce regressions over the status quo that need to be addressed first. DELFI NeTEx feed for Germany: Contains correct train names for non-IC/non-ICE long distance trains (EC, ECE, RJ, RJX, etc). Contains vehicle attributes (on a similar level as provided by DB's website). Less details regarding bus stops compared to GTFS, e.g. missing many platform names. SNCF NeTEx feed for France: Would finally give us proper route types and train names for TGV, IC and TER trains. Misses some trips included in the corresponding GTFS feed. Realtime data doesn't match against the NeTEx feed. We tested a workaround by importing both the NeTEx and GTFS feeds in the right order and have MOTIS merge the common trips correctly. This works as such, but identified two pre-existing merging issues that need to be fixed in MOTIS first. Swiss national NeTEx feed: Would give us train numbers for international trains and at least some vehicle attributes. Matching rates against realtime data are lower than with GTFS. Workarounds like currently in use in Germany might help, augmenting the schedule data with stop registry data. While it will still take a bit of time before any of those feeds will enter production on Transitous, we have started to prepare Transitous' import pipeline and documentation to not exclusively assume GTFS as the input format anymore. And more… There were plenty more topics discussed beyond those two: Making the Grafana dashboard more useful for feed/region maintainers. Using Wikidata as the canonical source for data augmentation, and how we could reliably match GTFS agencies or routes to Wikidata items. Requirements for a routing profile for blind users, such as routing along tactile and acoustic markers and minimizing steps and crossings. Implementation details for adding "on-trip" queries, ie. routing requests that don't start from a location but on board of a vehicle. Resolving duplicates between multiple GBFS v3 aggregated feeds. For more details, also see the meeting notes. And to end this with a screenshot, we also fixed the font rendering on the Transitous map which was missing labels in e.g. Egypt, Georgia and Thailand. Thai script labels on the map in Bangkok. Upcoming events There's several more opportunities in the upcoming weeks to meet

members of the Transitous community: FOSDEM in Brussels on January 31 - February 1. The OSM Hack Weekend in Karlsruhe on February 21-22. FOSSGIS-Konferenz in Göttingen, March 25-28. There's of course also the Transitous Matrix channel to get involved.

- This Week in Plasma: dark mode switch and global push-to-talk (2026/01/17 00:03)
  Welcome to a new issue of This Week in Plasma! This week we closed the door on features for Plasma 6.6, which opened another one for those features to land in 6.7. As a result, several features were merged for Plasma 6.7, including some particularly juicy ones! As for Plasma 6.6, this starts the one-month period where the core Plasma team focuses almost entirely on fixing bugs. As you'll see below, we already fixed quite a few this week! So there's a huge amount of stuff to go over, and let's get right to it: Notable New Features Plasma 6.6.0 System Monitor now lets you set the priority of processes graphically, just like the older KSysGuard app did. (Matthieu Carteron, plasma-systemmonitor MR #381 and libksysguard MR #455) (The lack of a darkened underlay is unintentional; we'll get that fixed up soon.) Plasma 6.7.0 Added a switch to the Brightness and Color widget that lets you instantly go from light mode to dark mode (or vice versa)! (Kai Uwe Broulik, powerdevil MR #576) (At some point we'll add a nice cross-fade transition here, too.) Added a global push-to-talk feature: if you set a push-to-talk key, all microphones will be muted until the specified key is held down. (Kai Uwe Broulik, Aleix Pol Gonzalez, and Shubham Arora, plasma-pa MR #394, kglobalaccel MR #41, and plasma-workspace MR #6126) Notable UI Improvements Plasma 6.5.6 The HDR calibration wizard now temporarily disables Night Light while calibrating, to ensure that you get an accurate result. (Xaver Hugl, kscreen MR #448) Plasma 6.6.0 Mounting a removable disk no longer performs a file system scan by default; now this is a manual action you initiate from the expanded actions list. (Akseli Lahtinen, KDE Bug #505852) The screen chooser dialog now includes a search/filter field so you can easily find a screen by name even when there are a zillion windows open. (Harald Sitter, xdg-desktop-portal-kde MR #506) Kicker's search results no longer flicker or resize while typing, keeping columns stable during searches for a smoother and snappier experience. (Christoph Wolk, plasma-desktop MR #3439) HDR Calibrator now offers a summary page with a setting to better support Windows HDR applications and games. (Xaver Hugl, kscreen MR #443) Replaced some technical gobbledygook in the titles of Bluetooth status and error notifications with more relevant and user-friendly text. (Nate Graham, bluedevil MR #238) (The body text isn't great either, but that's also being worked on!) If you happen to have a keyboard or other input device with "Seek Forwards" and "Seek Backwards" buttons, pressing them now works as expected out of the box. (Vlad Zahorodnii, KDE Bug #514680) Plasma 6.7.0 System Settings' Game Controller, Mouse, and Touchpad pages now only appear when the devices they configure are present. (Alexander Wilms, plasma-desktop MR #3436) Discover now shows sub-categories in its "Games" group for game launchers and game tools. (Jakob Dev, discover MR #1224) Improved how the global edit mode works with a touchscreen. (Shubham Arora, plasma-workspace MR #6161 and plasma-desktop MR #3442) On System Settings' Accessibility page, the Mouse Navigation tooltip now explains how to switch mouse click modes with the numeric keypad. (Jaimukund Bhan, KDE Bug #505687) Searching for "memory" now turns up the System Monitor app in search results. (Nicolas Fella, plasma-workspace MR #6194) Frameworks 6.23 Improved the touch-friendliness of open/save dialogs. (Méven Car, KDE Bug #513606) Improved the icon selection algorithm for missing icons so that it no longer returns downscaled versions of much larger icons, which may have a completely different style. (Alexander Wilms, KDE Bug #466678) By default, sidebars and left edge drawers in Kirigami-using apps now have exactly the width needed to avoid being too wide or too narrow. Some apps still override the default width, and that will need to be un-done now, so expect the weirdly-sized sidebars to get fixed over time, rather than all at once when you upgrade to Frameworks 6.23. (Marco Martin, KDE Bug #505693) KDE Gear 26.04.0 System Settings' pages related to audio CDs (if you have them installed) now only appear when the computer has any optical drives. (Nate Graham, KDE Bug #513661 and KDE Bug #513664) Notable Bug Fixes Plasma 6.5.5 Fixed a case where KWin could

crash on launch when the GPU did something weird when trying to render screencasts or window thumbnails. (Xaver Hugl, KDE Bug #513710) Fixed an issue that made the fingerprint enrollment dialog's "Add" button go missing if you canceled enrollment and then immediately re-opened the dialog. (Christoph Wolk, KDE Bug #514088) Fixed an issue that sometimes made Weather Report widget's tooltip use the wrong unit and display numbers with excessive decimal places. (Ismael Asensio, KDE Bug #514419) Fixed an issue that made System Settings' search field sometimes not show the proper language-specific placeholder text. (Albert Astals Cid, KDE Bug #512187) Plasma 6.5.6 Fixed one of the top Plasma crashes that could happen when turning off certain screens. (Vlad Zahorodnii, KDE Bug #511757) Fixed a crash in System Settings' Game Controller page when using certain devices and versions of the SDL library. (David Edmundson, KDE Bug #511859) If you have multiple Plasma panels, clicking on different ones over and over again while in edit mode no longer makes multiple panel edit dialogs appear. (Marco Martin, KDE Bug #513135) Plasma 6.6.0 Fixed a KWin crash that could happen when waking up a laptop connected to an external screen. (Xaver Hugl, KDE Bug #514229) Fixed an issue relating to focus on the lock screen with multi-monitor setups. (Oliver Beard, KDE Bug #512028) Discover's window no longer becomes un-maximized when any popup dialogs appear. (Akseli Lahtinen, KDE Bug #503801) Having an exotic network setup or a lot of Docker containers no longer breaks the layout of System Settings' Remote Desktop page while the feature is turned on. (Christoph Wolk, KDE Bug #513504) Plasma 6.7.0 If you've turned on the login sound, it now plays at the right time. (Kai Uwe Broulik, KDE Bug #510923) Frameworks 6.22.1 Fixed a regression that made KDE Connect crash due to clipboard shenanigans. (Nicolas Fella, KDE Bug #514512) Frameworks 6.23 Fixed two issues with tooltips that could sometimes cause them to be offset or rapidly appear and disappear in the Kickoff Application Launcher widget. (Alexey Rochev, KDE Bug #510860 and KDE Bug #511875) Fixed some bugs and visual glitches with certain sidebars and list items in Kirigami-based apps when using an RTL language like Arabic or Hebrew. (Marco Martin and Christoph Wolk, kirigami MR #2027 and kirigami MR #2026) Fixed a strange issue that could cause notifications from the Quod Libet music player specifically to stop appearing. (Alexander Wilms, KDE Bug #489910) Symbolic icons for KDE Connect now re-color themselves properly when using a non-default color scheme. (Ángel Navarro, breeze-icons MR #522) Qt 6.10.3 Fixed two of the most common Plasma crashes that were caused by Qt's QML Compiler doing something weird under the hood. (Ulf Hermann, KDE Bug #513527 and KDE Bug #513012) Notable in Performance & Technical Plasma 6.6.0 Implemented version 2 of the Wayland color management protocol. (Xaver Hugl, kwin MR #8033) Reduced some visual glitches in Firefox when turning on its off-by-default HDR mode. (Xaver Hugl, KDE Bug #514599) Plasma 6.7.0 Implemented support for network activity monitoring on FreeBSD in the System Monitor app and widgets. (Jesper Schmitz Mouridsen, ksystemstats MR #41) How You Can Help Since the Plasma 6.6 beta period has commenced, this is a great time to submit bug reports for all the niggling issues you've been suffering with but haven't formally reported yet. We're focusing more than usual on bug triage too, so your reports will be seen. There's a new Troubleshooting help page that can help narrow down issues, too. Check it out! And helping to triage other people's reported issues is a big help, too. In addition, "This Week in Plasma" needs your help! Publishing these posts is time-consuming and needs community assistance to be sustainable. Right now there are two ways to help: Help put together the posts using the current mostly manual process Help automate the process Work can be coordinated in the relevant Matrix room. Beyond that, you can help KDE by directly getting involved in any other projects. Donating time is actually more impactful than donating money. Each contributor makes a huge difference in KDE — you are not a number or a cog in a machine! You don't have to be a programmer, either; many other opportunities exist. You can also help out by making a donation! This helps cover operational costs, salaries, travel expenses for contributors, and in general just keep KDE bringing Free Software to the world. To get a new Plasma feature or a bugfix mentioned here Push a commit to the relevant merge request on invent.kde.org.

- KDE Ni! OS – Plasma Login Manager now available (2026/01/16 21:45)
  KDE Ni! OS is a custom flavour (configuration, not a separate distribution) of NixOS that showcases KDE software. It builds on NixOS with the aim to reimplement the same features other popular immutable distributions have, while providing a first-class KDE Plasma setup. The Plasma Login Manager support has been merged into Ni! OS. If you want to use it, there are two prerequisites: you are using Wayland, not X11; and you are on unstable NixOS. It is in the “works for me” state. I don’t use auto-login, virtual keyboard, etc. Going unstable If you are on the stable channel, which would surprise me as you’re reading this post, it is easy to switch to unstable just by running these commands as root user (sudo, or su, or…). nix-channel --add https://channels.nixos.org/nixos-unstable nixos nix-channel --update nixos-rebuild switch --upgrade This will switch you to the unstable version of NixOS. Mind that, as NixOS is an immutable distribution, you can easily boot back into the stable version – the previous version of your system is still accessible in the bootloader menu. Switching from SDDM to the Plasma Login Manager There’s a new option in Ni! OS called experimental.use_plasma_login_manager. The only thing you need to do in order to switch from SDDM to the Plasma Login Manager is to set it to true, and just switch your setup to the new configuration with: nixos-rebuild switch Plasma Login Manager on Ni! OS Switching back is also trivial – just change the value back to false and do the switch again. There and back again One new thing in Ni! OS is a custom label for the versions of the system (derivations in NixOS terminology). If you enable an experimental feature such as the Plasma Login Manager, the label will clearly denote that. It makes it easy to get back to a version without the experimental features enabled. As you can see in the following screenshot, the default label is kde-ni-os and all enabled experimental features are appended to it – when enabling the Plasma Login Manager, the label becomes kde-ni-os:plasma-login-manager. These labels can be seen in the following screenshot: Boot menu with labels shown [edit] Upstream Was notified by NixOS KDE maintainer K900, that PLM will officially become a part of nixpkgs as Plasma 6.5.5 gets updated to 6.6. You can follow the progress of the patch here. Once Plasma 6.6 is released, and it becomes available in nixpkgs, Ni! OS will start using the official package instead of my local hack.
- Web Review, Week 2026-03 (2026/01/16 13:37)
  Let’s go for my web review for the week 2026-03. European Commission issues call for evidence on open source Tags: tech, foss, politics, europe Go and get your voice heard! This is important matter, especially if you’re interested in Free Software. https://lwn.net/Articles/1053107/ US tech giants allying with European far-right to strip back EU rules Tags: tech, gafam, business, politics, europe They’ll do anything to further their grip on tech. The European Union is sleep walking on this one.
  https://www.brusselstimes.com/belgium/1916422/us-tech-giants-allying-with-european-far-right-to-strip-back-eu-rules So, You’ve Hit an Age Gate. What Now? Tags: tech, politics, law, surveillance, privacy Clearly the regulators don’t really understand the level of intrusiveness they’re unleashing with mandating age gates. This is one more layer of surveillance for large parts of the population.
  https://www.eff.org/deeplinks/2026/01/so-youve-hit-age-gate-what-now The Next Thing Will Not Be Big Tags: tech, innovation, foss, business, community This is a very rich article. There’s indeed more and more a rift between Open Source projects used by hyperscalers and the ones used by smaller businesses and individuals. You likely want to aim for the latter. https://blog.glyph.im/2026/01/the-next-thing-will-not-be-big.html Four More Tech Bloggers Are Switching to Linux Tags: tech, linux, foss, desktop Looks like the trend continues. Let’s hope the Linux desktop user base will keep growing this year. https://m.slashdot.org/story/451196 How Markdown took over the world Tags: tech, markdown, history, blog, commons Wondering where Markdown is coming from and how it became such a success? The piece helps answer those questions.
  https://www.anildash.com/2026/01/09/how-markdown-took-over-the-world/ Mailing lists vs Discourse forums: open source communities or

commodities? Tags: tech, foss, community, email Interesting points. Forums are clearly not good replacements for mailing lists. They might be a good complementary to mailing lists but both have very different affordances.
https://danielpocock.com/en/mailing-lists-vs-discourse-forums-open-source-community-or-commodity/ WhatsApp is untrustable Tags: tech, messaging, foss, security, privacy, foss If you needed a reminder about why you can't trust WhatsApp, this is a good explanation.
https://toki.la/posts/whatsapp I'm The Captain Now: Hijacking a global ocean supply chain network Tags: tech, security, api, secrets Friendly reminder that securing APIs and secrets is a must. Not doing so can have really bad consequences.
https://eaton-works.com/2026/01/14/bluspark-bluvoyix-hack/ New Social Web Working Group at W3C Tags: tech, web, standard, fediverse, social-media This is a welcome development at the W3C. Let's hope this working group will bring good things and stewardship for the related standards. https://socialwebfoundation.org/2026/01/15/new-social-web-working-group-at-w3c/ HTTP RateLimit headers Tags: tech, http, failure, standard Maybe we can expect improvements in how HTTP rate limiting is handled? https://dotat.at/@/2026-01-13-http-ratelimit.html Why We Don't Use AI Tags: tech, ai, machine-learning, gpt, copilot, ethics I agree with this so much. It's another one of those I feel I could have written. I have a hard time thinking I could use the current crop of "inference as a service" while they carry so many ethical issues.
https://yarnspinner.dev/blog/why-we-dont-use-ai/ AI Coding Degrades: Silent Failures Emerge Tags: tech, ai, machine-learning, copilot, quality, ethics There is a real question about the training data used for the coding assistant models. It's been a problem from the start raising ethical concerns, now it shows up with a different symptom. https://spectrum.ieee.org/ai-coding-degrades On FLOSS and training LLMs Tags: tech, ai, machine-learning, gpt, copilot, foss, law, ethics, copyright I'm not sure the legal case is completely lost even though chances are slim. The arguments here are worth mulling over though. There's really an ethical factor to consider.
https://chronicles.mad-scientist.club/tales/on-floss-and-training-llms/ Be Wary of Digital Deskilling Tags: tech, ai, machine-learning, copilot, business, economics, work, quality Is this really to improve your work? Or make you dependent? In the end it might be the users who loose.
https://calnewport.com/be-wary-of-digital-deskilling/ The coolest feature in Python 3.14 Tags: tech, python, debugging, containers OK, this is definitely a very cool hack. It can definitely help to debug locally. https://savannah.dev/posts/the-coolest-feature-in-314/ How to parametrize exception testing in PyTest? Tags: tech, python, tests, exceptions Neat little Python trick for testing exceptions.
https://borutzki.github.io/2026/01/15/how-to-parametrize-exception-testing-in-pytest.html Handling secrets (somewhat) securely in shells Tags: tech, security, secrets, shell What's the right way to manipulate secrets in your shell to avoid leakage? The answer definitely varies, here is the paranoid version. https://linus.schreibt.jetzt/posts/shell-secrets.html How Safe is the Rust Ecosystem? A Deep Dive into crates.io Tags: tech, rust, supply-chain, security There are growing concerns regarding the Rust supply chain. It's still time to address them but it's became important to tackle this area. https://mr-leshiy-blog.web.app/blog/crates_io_analysis/ Volumetric Cloud Rendering Tags: tech, graphics, 3d, shader, physics Long and good walkthrough on how to render nice clouds in real time. https://www.jacktollenaar.top/articles/clouds.html permission to begin learning Tags: tech, programming, language, learning, craftsmanship There's a lot to this. Learning different languages to get out of your habits definitely brings compound benefits. https://ficd.sh/blog/permission-to-begin-learning/ The PERFECT Code Review: How to Reduce Cognitive Load While Improving Quality Tags: tech, codereview This is an interesting way to frame where the effort should be spent in code reviews.
https://bastrich.tech/perfect-code-review/ One bottleneck at a time Tags: tech, engineering, management, productivity, kanban This is good advice. To improve your organisation, focus only on the biggest constraint. Otherwise you'll quickly be spread thin.
https://www.theengineeringmanager.com/growth/one-bottleneck-at-a-time/ Toyota Culture 20 Years Later: Why Jeffrey Liker's Lessons Still Matter

Tags: management, agile, lean, culture, trust, leadership, problem-solving This has been documented for a long while. Of course, it's been followed by an unhealthy fascination for the "Toyota way". This kind of cargo cult of course lead you nowhere to doing things properly. And yet, now that the dust settled, there are good lessons to learn from Toyota management back then. https://www.leanblog.org/2026/01/toyota-culture-jeffrey-liker-20-years-later/ Bye for now!

- Kdenlive 25.12.1 released (2026/01/15 12:00)
  Happy New Year! The first maintenance release of the 25.12 series is with the usual batch of stability fixes and workflow improvements. Highlights of this release include further polishing of the new welcome screen, added AMF encoding profile for Windows, fixes to audio capture and effects alongside numerous smaller improvements throughout the interface. See changelog for more details. For the full changelog continue reading on kdenlive.org.
- REST API Development with Qt 6 (2026/01/14 11:58)
  This post describes an experiment using Qt 6.7's REST APIs to explore Stripe's payment model, and what I learned building a small desktop developer tool. Recent Qt releases have included several conveniences for developing clients of remote REST APIs. I recently tried it out with the Stripe payments REST API to get to grips with the Qt REST API in the real world. The overloading of the term API is unhelpful, I find, but hopefully not too confusing here. As with almost everything I try out, I created Qt desktop tooling as a developer aid to exploring the Stripe API and its behavior. Naming things is hard, but given that I want to put a "Q" in the name, googling "cute stripes" gives lots of hits about fashion, and the other too-obvious-to-say pun, I've pushed it to GitHub as "Qashmere": setAlternatingRowColors(true); Developers using REST APIs will generally be familiar with existing tooling such as Postman and Bruno, for synthesizing calls to collections of REST APIs. Indeed, Qashmere uses the Stripe Postman JSON definition to present the collection of APIs and parameters. Such tools have scripting interfaces and state to create workflows that a client of the REST API needs to support, like "create a payment, get the id of the payment back from the REST API and then cancel the payment with the id", or "create a payment, get the id of the payment back from the REST API and then confirm it by id with a given credit card". So why create Qashmere? In addition to REST APIs, Stripe maintains objects which change state over time. The objects remain at REST until acted on by an external force, and when such an action happens a notification is sent to clients about those state changes, giving them a chance to react. I wanted to be able to collect the REST requests/responses and the notified events and present them as they relate to the Stripe objects. Postman doesn't know about events or about Stripe objects in particular, except that it is possible to write a script in Postman to extract the object which is part of a JSON payload. Postman also doesn't know that if a Payment Intent is created, there are a subset of next steps which could be in a workflow, such as cancel, capture or confirm payment etc. Something that I discovered in the course of trying this out is that when I confirm a Payment Intent, a new Charge object is created and sent to me with the event notification system. Experimental experiences like that help build intuition. Stripe operates with real money, but it also provides for sandboxes where synthetic payments, customers etc can be created and processed with synthetic payment methods and cards. As Qashmere is only useful as a developer tool or learning aid, it only works with Stripe sandboxes. Events from Stripe are sent to pre-configured web servers owned by the client. The web servers need to have a public IP address, which is obviously not appropriate for a desktop application. A WebSocket API would be more suitable and indeed the stripe cli tool uses a WebSocket to receive events, but the WebSocket protocol is not documented or stable. Luckily the stripe cli tool can be used to relay events to another HTTP server, so Qashmere runs a QHttpServer for that purpose. Implementation with Qt REST API The QRestReply wraps a QNetworkReply pointer and provides convenience API for accessing the HTTP return code and for creating a QJsonDocument from the body of the

response. It must be created manually if using QNetworkAccessManager directly. However the new QRestAccessManager wraps a QNetworkAccessManager pointer, again to provide convenience APIs and overloads for making requests that are needed in REST APIs (though some less common verbs like OPTIONS and TRACE are not built-in). The QRestAccessManager has conveniences like overloads that provide a way to supply callbacks which already take the QRestReply wrapper object as a parameter. If using a QJsonDocument request overload, the “application/json” Content-Type is automatically set in the header. One of the inconveniences of QRestAccessManager is that in Qashmere I use an external definition of the REST API from the Postman definition which includes the HTTP method. Because the QRestAccessManager provides strongly typed API for making requests I need to do something like: if (method == "POST") { rest.post(request, requestData, this, replyHandler);} else if (method == "GET") { rest.get(request, this, replyHandler);} else if (method == "DELETE") { rest.deleteResource(request, this, replyHandler);} There is a sendCustomRequest class API which can be used with a string, but it does not have an overload for QJsonDocument, so the convenience of having the Content-Type header set is lost. This may be an oversight in the QRestAccessManager API. Another missing feature is URL parameter interpolation. Many REST APIs are described as something like /v1/object/:object_id/cancel, and it would be convenient to have a safe way to interpolate the parameters into the URL, such as: QUrl result = QRestAccessManager::interpolatePathParameters("/v1/accounts/:account_id/object/:object_id/cancel", { {"account_id", "acc_1234"}, {"object_id", "obj_5678"} }); This is needed to avoid bugs such as a user-supplied parameter containing a slash for example. Coding Con Currency In recent years I’ve been writing and reading more Typescript/Angular code which consumes REST services, and less C++. I’ve enjoyed the way Promises work in that environment, allowing sequences of REST requests, for example, to be easy to write and read. A test of a pseudo API could await on requests to complete and invoke the next one with something like: requestFactory.setBaseURL("http://some_service.com");async testWorkflow(username: string, password: string) { const loginRequest = requestFactory.makeRequest("/login"); const loginRequestData = new Map(); loginRequestData.setParam("username", username); loginRequestData.setParam("password", password); const loginResponse = await requestAPI.post(loginRequest, loginRequestData); const bearerToken = loginResponse.getData(); requestAPI.setBearerToken(bearerToken); const listingRequest = requestFactory.makeRequest("/list_items"); const listingResponse = await requestAPI.get(listingRequest); const listing = JSON.parse(listingResponse.getData()); const firstItemRequest = requestFactory.makeRequest("/retrieve_item/:item_id", { item_id: listing[0].item_id }); const firstItem = await requestAPI.get(firstItemRequest);} The availability of async functions and the Promise to await on make a test like this quite easy to write, and the in-application use of the API uses the same Promises, so there is little friction between application code and test code. I wanted to see if I can recreate something like that based on the Qt networking APIs. I briefly tried using C++20 coroutines because they would allow a style closer to async/await, but the integration friction with existing Qt types was higher than I wanted for an experiment. Using the methods in QtFuture however, we already have a way to create objects representing the response from a REST API. The result is similar to the Typescript example, but with different ergonomics, using .then instead of the async and await keywords. struct RestRequest{ QString method; QString requestUrl; QHttpHeaders headers; QHash<QString, QString> urlParams; QUrlQuery queryParams; std::variant<QUrlQuery, QJsonDocument> requestData;};struct RestResponse{ QJsonDocument jsonDoc; QHttpHeaders headers; QNetworkReply::NetworkError error; QUrl url; int statusCode;};QFuture<RestResponse> makeRequest(RestRequest restRequest){ auto url = interpolatePathParameters(restRequest.requestUrl, restRequest.urlParams); auto request = requestFactory.createRequest(url); auto requestBodyDoc = extractRequestContent(restRequest.requestData); auto requestBody = requestBodyDoc.toJson(QJsonDocument::Compact); auto reply = qRestManager.sendCustomRequest(request, restRequest.method.toUtf8(), requestBody, &qnam, [](QRestReply &) {}); return

QtFuture::connect(reply, &QNetworkReply::finished).then([reply]() { QRestReply restReply(reply); auto responseDoc = restReply.readJson(); if (!responseDoc) { throw std::runtime_error("Failed to read response"); } RestResponse response; response.jsonDoc = *responseDoc; response.statusCode = restReply.httpStatus(); response.error = restReply.error(); response.headers = reply->headers(); response.url = reply->url(); return response; });} The QRestAccessManager API requires the creation of a dummy response function when creating a custom request because it is not really designed to be used this way. The result is an API accepting a request and returning a QFuture with the QJsonDocument content. While it is possible for a REST endpoint to return something else, we can follow the Qt philosophy of making the most expected case as easy as possible, while leaving most of the rest possible another way. This utility makes writing unit tests relatively straightforward too: RemoteAPI remoteApi;remoteApi.setBaseUrl(QUrl("https://dog.ceo"));auto responseFuture = remoteApi.makeRequest({"GET", "api/breed/:breed/:sub_breed/images/random", {}, { {"breed", "wolfhound"}, {"sub_breed", "irish"} }});QFutureWatcher<RestResponse> watcher;QSignalSpy spy(&watcher, &QFutureWatcherBase::finished);watcher.setFuture(responseFuture);QVERIFY(spy.wait(10000));auto jsonObject = responseFuture.result().jsonDoc.object();QCOMPARE(jsonObject["status"], "success");QRegularExpression regex(R"(https://images\.dog\.ceo/breeds/wolfhound-irish/[^.]+.jpg)");QVERIFY(regex.match(jsonObject["message"].toString()).hasMatch()); The result is quite similar to the Typescript above, but only because we can use spy.wait. In application code, we still need to use .then with a callback, but we can additionally use .onFailed and .onCanceled instead of making multiple signal/slot connections. With the addition of QtFuture::whenAll, it is easy to make multiple REST requests at once and react when they are all finished, so perhaps something else has been gained too, compared to a signal/slot model: RemoteAPI remoteApi;remoteApi.setBaseUrl(QUrl("https://dog.ceo"));auto responseFuture = remoteApi.requestMultiple({ { "GET", "api/breeds/list/all", }, {"GET", "api/breed/:breed/:sub_breed/images/random", {}, {{"breed", "german"}, {"sub_breed", "shepherd"}}}, {"GET", "api/breed/:breed/:sub_breed/images/random/:num_results", {}, {{"breed", "wolfhound"}, {"sub_breed", "irish"}, {"num_results", "3"}}}, {"GET", "api/breed/:breed/list", {}, {{"breed", "hound"}}},});QFutureWatcher<QList<RestResponse>> watcher;QSignalSpy spy(&watcher, &QFutureWatcherBase::finished);watcher.setFuture(responseFuture);QVERIFY(spy.wait(10000));auto four_responses = responseFuture.result();QCOMPARE(four_responses.size(), 4);QCOMPARE(four_responses[0].jsonDoc.object()["status"], "success");QVERIFY(four_responses[0].jsonDoc.object()["message"]. toObject()["greyhound"].isArray());QRegularExpression germanShepherdRegex(R"(https://images.dog.ceo/breeds/german-shepherd/[^.]+.jpg)");QCOMPARE(four_responses[1].jsonDoc.object()["status"], "success");QVERIFY(germanShepherdRegex.match(four_responses[1].jsonDoc.object()["message"].toString()).hasMatch());QRegularExpression irishWolfhoundRegex(R"(https://images.dog.ceo/breeds/wolfhound-irish/[^.]+.jpg)");QCOMPARE(four_responses[2].jsonDoc.object()["status"], "success");auto irishWolfhoundList = four_responses[2].jsonDoc.object()["message"].toArray();QCOMPARE(irishWolfhoundList.size(), 3);QVERIFY(irishWolfhoundRegex.match(irishWolfhoundList[0].toString()). hasMatch());QVERIFY(irishWolfhoundRegex.match(irishWolfhoundList[1].toString()). hasMatch());QVERIFY(irishWolfhoundRegex.match(irishWolfhoundList[2].toString()). hasMatch());QCOMPARE(four_responses[3].jsonDoc.object()["status"], "success");auto houndList = four_responses[3].jsonDoc.object()["message"].toArray();QCOMPARE_GE(houndList.size(), 7);QVERIFY(houndList.contains("afghan"));QVERIFY(houndList.contains("basset"));QVERIFY(houndList.contains("blood"));QVERIFY(houndList.conta

ins("english"));QVERIFY(houndList.contains("ibizan"));QVERIFY(houndList.contains("plott"));QVERIFY(houndList.contains("walker")); setAutoDeleteReplies(false); I attempted to use new API additions in recent Qt 6 versions to interact with a few real-world REST services. The additions are valuable, but it seems that there are a few places where improvements might be possible. My attempt to make the API feel closer to what developers in other environments might be accustomed to had some success, but I'm not sure QFuture is really intended to be used this way.Do readers have any feedback? Would using QCoro improve the coroutine experience? Is it very unusual to create an application with QWidgets instead of QML these days? Should I have used PyQt and the python networking APIs?

- Haruna 1.7 (2026/01/14 10:00)
  Haruna version 1.7.1 is released. playlist advanced sorting and grouping Windows version: haruna-1.7.1-windows-gcc-x86_64.exe haruna-1.7.1-windows-gcc-x86_64.7z Availability of other package formats depends on your distro and the people who package Haruna. If you like Haruna then support its development: GitHub Sponsors | Liberapay | PayPal Feature requests and bugs should be posted on bugs.kde.org, ignoring the bug report template can result in your report being ignored. Known issues The animation for the playlist can be stuttery/slow when playback is active. You can improve it by creating two custom commands that run on startup set override-display-fps 75 (replace 75 with your monitor's refresh rate) and set video-sync display-resample. These don't work for variable refresh rate monitors. Changelog 1.7.1 Bugfixes fixed searching playlist fixed tooltip background being same color as its text 1.7.0 Features Playlist added advanced sorting and grouping (Muhammet Sadık Uğursoy) added context menu to open file in Hana (thumbnail generator, only if it's installed). Get it from flathub the last active playlist will be set as visible when starting the app Other added replay gain settings (Muhammet Sadık Uğursoy) mpris thumbnail is only set for audio files, this allows the os taskbar preview to show the actual live window decreased the size of the play icon in the compact playlist Bugfixes fixed database folder not being created fixed seekbar tooltip not updating when file changes and the mouse is not moved fixed deleting custom commands fixed saving last opened url
- KDE Gear 26.04 release schedule (2026/01/13 23:43)
  This is the release schedule the release team agreed on https://community.kde.org/Schedules/KDE_Gear_26.04_ScheduleDependency freeze is in around 7 weeks (March 5) and feature freeze one after that. Get your stuff ready!
- Debian SBuild for FreeBSD People (2026/01/13 23:00)
  For other-$WORK I am doing a bit of Debian packaging (prep-work) and upstream wrangling to bring some projects into a more-modern world. So now I have a Debian 13 workstation and need to build things for Debian Unstable without breaking my host system. On FreeBSD, this is a job for Poudriere. Here's my notes on how I do something half-assedly similar on Debian (as usual, mostly documentation for "future me"). As an example of something I'm updating, libaccounts is one. KDE is moving away from it, but there are other consumers. The "Poudriere-alike" for Debian seems to be SBuild, and the instructions are quite extensive. For my rinky-dink use, though, the setup section is mostly-enough. Start with a plain Debian 13 (Trixie) installation. I have one with KDE Plasma 6 Wayland on it, which is quite nice. Creating a Tarball These commands verbatim from the SBuild instructions, as my own local user: sudo apt install sbuild mmdebstrap uidmap mkdir -p ~/.cache/sbuild mmdebstrap \ --include=ca-certificates,cmake,git,gnupg,libqt6widgets6,libqt5widgets5 \ --skip=output/dev \ --variant=buildd \ unstable \ ~/.cache/sbuild/unstable-amd64.tar.zst \ https://deb.debian.org/debian This creates a tarball. The tarball includes whatever you tell it to install additionally – so I have a couple of typical-Qt-developer things listed. This list isn't nearly complete for anything like "Qt5 and Qt6 parallel-development", because it is missing -dev packages and qmake and many other things. Debian packaging naming is an exercise in inconsistency,

IMO, so it is always an adventure to figure out what I need. The steps above are sort-of like creating a “base jail” in Poudriere. The tarball is a basic Debian Unstable that can be unpacked anywhere (creating a “jail”) and then used for whatever development work is needed. Unlike a jail, the unpacked tarball is just a chroot(1), so it doesn’t have special networking or other restrictions. The chroot, once unpacked, is persistent. That means that stuff that is installed in it remains there and is available later. I don’t need to care about disk space, so it’s fine to litter my filesystem with multiple chroots, one for each project. Extracting Tarball Create a place for the chroot of a project, and extract the tarball there: mkdir -p ~/src/chroot-libaccounts sudo tar x --zstd -f ~/.cache/sbuild/unstable-amd64.tar.zst -C ~/src/chroot-libaccounts GNU tar doesn’t have the advanced features of BSD tar which automatically understands the compression, so it needs to be specified by hand. To enter the chroot, use sudo chroot ~/src/chroot-libaccounts. Since it’s persistent, it is probably worthwhile to update and upgrade it and maybe install additional packages, or whatever else one does with a Debian system. Using a chroot with external sources Inside the chroot is where builds happen. Outside the chroot is where the source lives, so I need to mount the source directory of whatever I’m working on, in the chroot. I do so before entering the chroot. sudo mkdir ~/src/chroot-libaccounts/home/src sudo mount --bind ~/src/libaccounts-qt/ ~/src/chroot-libaccounts/home/src sudo chroot ~/src/chroot-libaccounts Why not Docker? Now that I’ve written this down in this detail, I realize that a suitable Docker image and container might have been what I was looking for. Well, almost: persistence is a thing. It’s a feature of Docker that the container re-starts in a pristine state every time. It’s an anti-feature for what I’m doing here. There’s ways of dealing with that, and I do so at $WORK. But not for this project.

- KDE Ni! OS – Plasma Login Manager teaser (2026/01/13 18:35)
  KDE Ni! OS is a custom flavour (configuration, not a separate distribution) of NixOS that showcases KDE software. It builds on NixOS with the aim to reimplement the same features other popular immutable distributions have, while providing a first-class KDE Plasma setup. Just a teaser this time. I’ve read somewhere that Fedora will be the first distribution to replace SDDM with Dave’s brand new Plasma Login Manager. Will Ni! OS be the first non-distribution to do the same? And if it does, will it become a distribution as it distributes yet another package not available in vanilla NixOS? :) Plasma Login Manager on Ni! OS
- KDE Plasma 6.6 Beta Release (2026/01/13 00:00)
  Here are the new modules available in the Plasma 6.6 beta: plasma-login-manager plasma-keyboard plasma-setup Some important features and changes included in 6.6 beta are highlighted on KDE community wiki page. View full changelog
- KDE Plasma 6.5.5, Bugfix Release for January (2026/01/13 00:00)
  Tuesday, 13 January 2026. Today KDE releases a bugfix update to KDE Plasma 6, versioned 6.5.5. Plasma 6.5 was released in October 2025 with many feature refinements and new modules to complete the desktop experience. This release adds a month’s worth of new translations and fixes from KDE’s contributors. The bugfixes are typically small but important and include: View full changelog
- 2025 Musically Wrapped (2026/01/12 23:00)
  I have a paper calendar. It hangs on the wall. I draw things on it, like Kladderadatch, to remind me where to go of an evening (or this afternoon). At the end of the year, with that calendar and my ticket history from Doornroosje (a music podium in Nijmegen) I can reconstruct my concert visits of the year. Here’s my year wrapped. Roos Rebergen & SunSun Orchestra Roos is always really peculiar, and did not fail to deliver. The classical string quintet as band worked well. It’s quite different from her pop recordings. I saw Roos in 2006 or so when she played in a local school, and it’s always stayed with me. Politie Warnsveld + Misprint POPO! It’s like Doe Maar has reincarnated. Happy ska, although now they

have a bigger setlist it is a little less wild. This was also a sad concert because of the death of a concert-friend – she was a big POPO fan – in an accident a few months earlier. De Kift “Ik heb rood haar, en lees wel eens gedichten” It’s a punk-jazz-improv group. Live they’re weird, but I did miss listening to the actual words. Parker Millsap Blues country, now as a solo show instead of with a band. Stippenlift Dutch-language electro-pop about his depression. ELUCID Rap, old-school. Dorpsstraat 3 Dutch indie. The Vices I’m pretty sure that during this concert I ended up thinking about the font-kerning in their logo, more than the music. Meh. Rats on Rafts I have no real recollection, it might have been boring. Ghost Funk Orchestra Jazz. If you asked me beforehand about a trombone solo, I would have said “probably boring”. Afterwards, fuck yeah! Trombone solo! They were amazing on stage. Girls to the Front A triple show with L.A. Sagne, Death Sells and C’est Qui? I’ve seen Death Sells a couple times after, they’re fun and personable. De Roos van Nijmegen is a yearly battle-of-the-bands, and I go with the kids, and we Have Opinions about things. It’s spread across 3 nights and the finals. I voted for PORTRAY. I thought Grandad was pretty good in the first round, and boring in the finals. The peeps from Pomme Grenade are the ones I run into around town most often. Liz Beekman was, as singer-songwriter, the odd-one-out in the first round, but I quite liked both her music and relaxed podium presence. The Ex + Brader Mûsikî More punk-jazz. Brader was a totally new Kurdish-language experience for me. The Ex was weird and experimental and the broad grin on the drummer’s face as she puts in more cowbell in The Apartment was magical. Tramhaus had a lot of social message that I agree with, but not amazing. Place to Bury Strangers I saw them in “old” Doornroosje, with my friend Armijn who described them as “ear-bleedingly loud”. Regulations prevent that now, but they were great and the round-through-the-crowd is a lot of fun to be part of. Elephant is a big-ish name and had a lot of radio play, so that’s why got tickets. I have no real recollection, it might have been boring. KNIVES Absolutely wild live-show, amazing energy. I told them I thought they were “amazeballs” after the show, they called me “old”. Love you too! I was here also because of Death Sells opening. Crash Fest Organized by Outahead – an indie band that don’t do it for me – but I went because Lodyne and Death Sells were playing and I’m into naked bass players. Alice Faye The first of a week of singers-songwriters. I don’t remember anything particular, but I do know I liked it for being a relaxed night out instead of sweaty and loud. Tara Nome Doyle Second one of singers-songwriters. I remember her being very chatty and open about the songwriting process and what things were about. That’s one of the nice things in a really small venue, the artists are there and all themselves. Stereolab The crowd was all fans, who could sing along with every song. I could tell they were having a ball, but it did not land at all for me. Heather Nova in the park, in the rain, with a rainbow, with ducks waddling across the stage, and a spider that dropped onto her hand during Like a Hurricane. As a consummate performer, she put the spider away and picked up the chorus again. The Hard Quartet This is a supergroup, I guess you could call it. I saw Pavement back in the day, and when Stephen is at the mike, it’s like a Pavement song. And when someone else takes over, it’s a different band. This was good to see for being a bunch of really experienced and work-well-together musicians. Bassolino Italian funk. It was funky, but the funk did not reach my hips. I’m way too much a white boy for that – put me in a mosh pit instead. Magic Bullet I have no recollections of again. It was possibly boring. Rosalie Cunningham I did not expect a ’70s hardrock revival on stage. It was amazing. She has wonderful eyes. Roscoe can sing that one song pretty good. And it was a party for all. Dick Move More punk bands should sell hot pink T-shirts. This was a blast. Preoccupations I figured “band from Calgary”. There was a Flames T-shirt on-stage. They did their thing. They left. Very little interaction with the audience. Misprint They were with POPO earlier this year, and I kept bumping into their bass player at other concerts, so it was the least I could do get tickets for their own show. Kind of middle-of-the-road, good enough. Early James Blues from Alabama. He did a nice closing number with his girlfriend, it worked pretty well as a duet. Gill Landry Blues from Louisiana. This was very personal, and you can tell Gill was a street performer before moving on to the stage. One to re-visit if he comes back. Joachim Cooder Son-of-Ry, playing an electric thumb piano. This was

very much not what I expected. It was interesting, and I told him “peculiar” after the show, but I’m afraid it did not get my feet a-tappin’. Frazey Ford Was she drunk? It took forever before the show gelled a little, but it never really moved. Leith Singer-songwriter. Blue eyes that stare right through your soul. I really enjoyed this show, and the openers, Robinson Kirby, were fun as well. DITZ Fucking well tore the house down. Her with the boots has an amazing command of the audience, the pit was wild, the communion weird and disconcerting. KNIVES opened here, but did not get nearly the same response as earlier in the year (in a different venue, must be said). zZz These guys I saw when they – and I – had no grey hairs and I remember them jumping up and down on the organ and it was loud and chaotic. They still are, although with a bad back climbing on stuff is no longer an option. Vals Alarm Punk with too much backing track. Gotta hand it to them, though, with your parents in the audience belting out “I wanna fuck some new boys / I need a new dick / new dick” takes some courage. De Niemanders This is a collective that brings singers from refugee centers in the Netherlands to the stage. With a gospel couple, and a good backing band, we can see what talent we’re squandering. I don’t know the names of the individual performers though. Ahm has amazing sustain. Habibi from Yemen, I think, is such a cute boy with an excellent delivery. Black Bottle Riot Does a end-of-year show every year in Nijmegen. Packed house, almost all fans. Random people come up to talk with you. Sabine from Zeeland, it was good to meet you. This show was almost three hours, and one big party. It’s good to have something to plan again in 360 days or so. Not in Nijmegen: Black Country New Road In Paradiso, Amsterdam. I’m clearly spoiled by easy-going Nijmegen, because I thought the venue was annoying. And every song was .. not quite it. Nothing landed, and the feels-like-American-film-music makes me unhappy. Where there was a neat idea (five recorders? sure, woodwind quintet) it was executed in a too-limited way. Bit of a disappointment, but the openers, Westside Cowboy, were fun. West Side Story in Rome. Man, the story is paper-thin, even if the singers were excellent. The only fun I had here was realizing that “having problems with the PRs” is not a GitHub thing. Opera school in Arezzo. There’s an opera school, students come from the United States to learn to sing an Italian opera, and execute it in the square. Stories still paper-thin, but such is operetta as an art form. I learned that “learn to sing” means “make the right sounds”, because the students could not actually speak Italian. Ones I missed (but did have tickets): Spinvis I was doing drywall and at 10pm remembered I had tickets to a one-off special show in Kleve. I had bought them a couple of days previous and hadn’t written it down. Felipe Baldomir Singer-songwriter week, but I was sick. Hackensaw Boys Bluegrass, but I was sick. That’s 46 concerts this year. I do try to see something every week. In principle I don’t listen to stuff in advance, I just go and find out what it is once the band starts. I have a strong preference for Merleyn, the smallest of the Doornroosje venues, because everything is close-by and personal. The beer is better there, also. Punk has the best odds of making me happy on an evening, but I’m glad I go to random other stuff to broaden my horizons. I have punk, jazz, ska and rap lined up for the next three months, and also Green Milk from the Planet Orange, whatever genre that is.

- Game Jam for Free Software Desktop Games (2026/01/11 11:46)
  The folks at GNU/Linux València are organizing a Game Jam focused on Free Software Desktop Games. You can see the details here: https://itch.io/jam/lliurejam Maybe we could take the opportunity to try to revive a bit the very very very dormant KDE Games community? Though we have the basic games covered already so someone would have to come up with an idea of what to do first :D
- Updated Tellico Handbook (2026/01/10 22:50)
  Online publication of the documentation for many KDE applications has been updated to docs.kde.org. Tellico’s current handbook can be found there.
- KJournald Update January 2026 (2026/01/10 18:19)

A surprising long time passed since my last status update about KJournald. So it’s time again to shed some light on the recent changes. KJournald is a KDE project that provides graphical browsing UI for journald log databases. For those who never heard the term “journald”, journald is the system logging service of systemd and it is found in most modern Linux systems. This means, in the journald databases one can find all the system log messages about important incidents happening on a system, which make it very important for system admins but also for all technical users who want to analyze when something is not working correctly on their systems. The kjournald-browser provides a Qt and Kirigami based UI to efficiently browse and filter those logs (note: there exist different tools for that, even systemd provides its own command line tool “journalctl”). The focus of kjournald-browser are the following use cases: ease filtering of log messages in order to efficiently reduce huge logs to sizes that a human can analyze ease analysis of interactions of different processes / systemd units by colorizing and other graphical helps provide a simple but powerfull UI/UX that focuses on log analysis support access for local, collected and remote journald logs (and do this only for journald) Since my last blog post, the kjournald-browser application became part of the regular KDE gear releases and nowadays is packages by e.g. Fedora and Suse; unfortunately, it is still not packaged on Debian or Ubuntu yet — if you want to do it and need support please reach out to me! At the moment, also the packaging as Flatpak application on Flathub is ongoing. But already since a long time though, the KDE Flatpak nightly builds provide the latest state of the app. With the last major release 25.12.0, a few new cool features were added: systemd differentiates between system service and user service and log messages are split. With the last release User Journals can be accessed in addition to System Journals. The search bar was reworked (more compact UI, additional case-insensitive search)) and now a history of recent search subjects is stored for faster access. KJournald-browser can not only used to load the log files from the local system but also to access arbitrary journald databases (e.g. collected from embedded devices for post-mortem analysis or from servers). For this loading mechanism, an improved UI database loading errors feedback is now available the validates access to loaded files. One feature was slightly too late for this release, but is already ready for the next: systemd provides the feature to use service templates, which means that every service instance has its own ID. Especially for user services that is a much used feature and this clutters the UI much. A new service grouping feature allows to group service logs by template name (this is also the default, but behavior can be selected in UI). Since 2026 is still young, there are a few features on the roadmap of this year. The two most important ones in my opinion are: adding access to the journald remote access protocol, which will be important to live-monitor servers or embedded devices that have a systemd-journal-remote service running introduce pre-filtering of processes and units for the selected boot id, which will improve usability for very long journals

- I love KDE too much to retire! Snap beta releases trickling in. Stay tuned. (2026/01/10 16:54)
  A short but sweet note to say I am coming out of my short retirement to help with snaps again. My time is extremely limited, however we are working hard on getting snaps on CI and I have some newer snaps in –beta trickling in for testing. You must install kf6-core24 from beta as well to test them (this will likely break older kde snaps in the process so beware.) This is slow going as I work on them during my hour lunch at day job and spare stolen moments. KDE is coming up on its 30th birthday!!! How cool is that! I KDE Like my work? Consider a donation. Thank you! Donate
- This Week in Plasma: car of the year edition (2026/01/10 00:01)
  Welcome to a new issue of This Week in Plasma! Let’s thank Lubos Krystynek, Rafal Krawczyk, and John Veness for stepping up to help with this week’s issue. Thanks, guys! This week, the first car running KWin won the “Car of the Year” award. Yes, really — KDE in the car! Here’s KDE’s Victoria Fischer talking about it at Qt World Summit 2023: Almost all of these posts end with “KDE has become important in the world…” and I

think this is a good reminder that it’s true, not just some empty platitude. KDE is important. And all of you building or using KDE’s software are important, too. But KDE is not only important to cars; we’re incredibly important to computers! And on that subject, some really nice features and user interface improvements landed for the upcoming Plasma 6.6 release. The hard feature freeze is coming up soon, at which point we’ll move into full bug-fixing and polishing mode. But until then, enjoy some juicy new goodies! Check it out: Notable New Features Plasma 6.6.0 You can now save your current visual settings as a new global theme! (Vlad Zahorodnii, plasma-desktop MR #6097) Added a “Forget device” action to the Bluetooth system tray widget, allowing users to remove paired devices without opening System Settings. (Andrew Gigena, KDE Bug #434691) You can now search for processes in System Monitor based on their full command-line invocation when the “Command” column is visible. (Alexey Rochev, KDE Bug #448331) On supported systems, the logout screen now mentions when the system will restart into a different operating system or boot option after it reboots. (Nikolay Kochulin, plasma-workspace MR #5469) Notable UI Improvements Plasma 6.6.0 The Power and Battery widget now tells you what specific power management actions apps are blocking, instead of assuming that they’re all blocking both sleep and screen locking. (Jakob Petsovits, KDE Bug #418433) System Settings’ Thunderbolt page now hides itself when the device doesn’t support Thunderbolt. (Alexander Wilms, plasma-thunderbolt MR #47) When there are many windows open, the Task Manager widget will now scroll to the active one when you open its window thumbnail list. (Christoph Wolk, KDE Bug #499716) Notifications no longer waste space showing the same icon in two places. (Kai Uwe Broulik, plasma-workspace MR #6151) Spectacle now remembers the size (and on X11, also the position) of its main window across launches. (Aviral Singh, KDE Bug #499652) Made multiple UI improvements to the “Configure Columns” dialog in System Monitor. (Arjen Hiemstra, plasma-systemmonitor MR #405) In the Weather Report widget, when a weather station isn’t reporting the current wind speed, the widget now says it doesn’t know the wind speed, rather than claiming it’s “calm”. (Tobias Fella, kdeplasma-addons MR #969) The Kickoff Application Menu widget now does a better job of handling a huge number of favorite apps. Now the favorites column eventually becomes scrollable, instead of letting icons overlap. (Christoph Wolk, KDE Bug #424067) You can now find System Settings’ Wallpaper page by searching for “desktop background” and some other related terms. (Shubham Arora, plasma-workspace MR #6152) Frameworks 6.23 Made it possible to see more items at once in the “Get New [thing]” dialogs. (Nate Graham, frameworks-knewstuff MR #380) Open/Save dialogs now use relative-style date formatting for recent dates and times, which matches how Dolphin shows them. (Méven Car, frameworks-kio MR #2103) Folders that show thumbnails of their contents now refresh the thumbnail immediately when any of those files are removed. (Akseli Lahtinen, KDE Bug #497259) Notable Bug Fixes Plasma 6.5.5 Fixed a strange issue that broke key repeat only in the Brave web browser. (Nicolas Fella, KDE Bug #513637) Fixed an issue that could make the panel configuration dialog appear on the wrong screen with certain panel and screen arrangements. (Aleksey Rochev, plasma-workspace MR #6140) Fixed two issues with the “Show Alternatives” popup: one that made it get cut off outside of the screen area for widgets positioned on certain areas of the desktop, and another that made it not disappear when it lost focus. (Aleksey Rochev, KDE Bug #511188 and KDE Bug #511187) Plasma 6.6.0 Fixed an issue that made Plasma quit when you disconnected the last screen. (Xaver Hugl, KDE Bug #513003) Fixed an issue with the Applications table on System Monitor’s Overview page being blurry with certain scale factors. We had already previously fixed this, but it turned out there were more remaining cases where it still happened, so this should take care of the rest! (Arjen Hiemstra, KDE Bug #445759) Notable in Performance & Technical Plasma 6.6.0 Implemented support in Plasma for the up-and-coming oo7 Secret Service provider. (Marco Martin and Harald Sitter, plasma-workspace MR #6109) Fixed a hilarious issue that caused the wallpaper to bounce a tiny bit with certain fractional scale factors on secondary screens using direct scan-out while on a very recent kernel version. (Xaver Hugl, KDE Bug #513277) How You Can Help KDE has become important in the world, and your time and contributions have

helped us get there. As we grow, we need your support to keep KDE sustainable. You can help KDE by directly getting involved. Donating time is actually more impactful than donating money. Each contributor makes a huge difference in KDE — you are not a number or a cog in a machine! You don’t have to be a programmer, either; many other opportunities exist. For example, helping out to write these posts is warmly appreciated. Anyone interested in getting involved should check out the evolving documentation on the topic. You can also help out by making a donation! This helps cover operational costs, salaries, travel expenses for contributors, and in general just keep KDE bringing Free Software to the world. To get a new Plasma feature or a bugfix mentioned here, feel free to push a commit to the relevant merge request on invent.kde.org.

- Web Review, Week 2026-02 (2026/01/09 18:32)
  Let’s go for my web review for the week 2026-02. How Github monopoly is destroying the open source ecosystem Tags: tech, foss, community, ecosystem, github Github is definitely entrenched by now. Students and beginners hardly look for projects outside of it. Sad state of affair. https://ploum.net/2026-01-05-unteaching_github.html Rust At Scale: Scaleway’s Big Bet To Become THE European Hyperscaler Tags: tech, cloud, business, hardware, rust Wondering what’s on the mind of people working on an hyperscaler? This podcast and its transcript gives good insights. https://filtra.io/rust/interviews/scaleway-jan-26 ‘Bizarro World’ Tags: tech, gaming, culture An odd but interesting article. When a journalist randomly discovers that his wife is the best Tetris player in the world.
  https://archive.boston.com/news/globe/magazine/articles/2007/08/19/bizarro_world/ Everything You Need to Know About Email Encryption in 2026 Tags: tech, email, security, privacy, politics, gafam Email encryption is indeed still an open issue. There’s no fix in sight for it. It’s mostly a lack of political will though, so none of the big players are going to change anything.
  https://soatok.blog/2026/01/04/everything-you-need-to-know-about-email-encryption-in-2026/ Improving the Flatpak Graphics Drivers Situation Tags: tech, system, linux, flatpak, graphics Interesting point… What to do when there’s no good option in the application runtime for the needed graphics drivers and kernel combination? https://blog.sebastianwick.net/posts/flatpak-graphics-drivers/ Functors, Applicatives, and Monads: The Scary Words You Already Understand Tags: tech, functional, programming, type-systems, learning Functional programming is made scary due to its jargon. But it doesn’t have to be this way. https://cekrem.github.io/posts/functors-applicatives-monads-elm/ Python Numbers Every Programmer Should Know Tags: tech, python, performance A very comprehensive view of Python memory consumption and the speed of common operations. Some of the numbers are higher than I expected.
  https://mkennedy.codes/posts/python-numbers-every-programmer-should-know/ Stop Forwarding Errors, Start Designing Them Tags: tech, rust, failure, debugging, monitoring, logging Error handling is still not a properly solved problem in my opinion. At least the Rust community discusses the topic quite a bit. This is good inspiration for other ecosystems as well I think.
  https://fast.github.io/blog/stop-forwarding-errors-start-designing-them/ Embassy Tags: tech, embedded, rust Looks like an interesting framework for embedded projects. https://embassy.dev/ [uv] OnceMap: Rust Pattern for Running Concurrent Work Exactly Once Tags: tech, rust, performance, design, multithreading More interesting design ideas in uv. Didn’t know about the dashmap crate they’re using here it looks like a nice one too. https://codepointer.substack.com/p/uv-oncemap-rust-pattern-for-running Getting Real With LLMs Tags: tech, ai, machine-learning, copilot, architecture, complexity This looks like an interesting way to frame problems. It can give an idea of how likely they can be tackled with LLMs. It also shows that the architecture and the complexity greatly matter. https://www.giladpeleg.com/blog/getting-real-with-llms Coupling from a big-O perspective Tags: tech, programming, complexity, design OK maybe a longer piece than it should be. Still the idea is interesting. Clearly you want to mind the O(n) coupling in this context. https://blog.ploeh.dk/2026/01/05/coupling-from-a-big-o-perspective/ We Need to Stop

Calling Everything a Mock Tags: tech, learning, tests, tdd Indeed, the terminology has been greatly confused. I think I'll die on this particular hill though. I think it's important to name things properly. That said the trick of going through a verb might just work? https://coding-is-like-cooking.info/2026/01/we-need-to-stop-calling-everything-a-mock/ Stop Guessing, Start Improving: Using DORA Metrics and Process Behavior Charts Tags: tech, processes, metrics, data Interesting short article. Shows the use of DORA metrics and process behavior charts. This is a good way to test hypothesis and see the impact of processes changes or introduction of new practices. It needs to be done over time and be patient of course. https://www.infoq.com/articles/DORA-metrics-PBCs/ Improve Your Work System Tags: management, organisation, team Good questions to consider to gauge how you work. Can improve the organisation if you really get to the bottom of it. https://www.congruentchange.com/improve-your-work-system/ worstofbreed.net - We make bad software Tags: tech, complexity, satire, funny This is a very nice satire website about the problems in our industry. Want to work in a resume driven context? Here is how! https://worstofbreed.net/ Why Are There No Holes Around Trees? Tags: science, biology The biology of trees is just fascinating. And there's so much we still don't know about it. https://www.youtube.com/watch?v=pHJIhxZEoxg Bye for now!

- QtNat – Open your port with Qt (2026/01/09 17:38)
  QtNat is a lightweight C++ library built with Qt 6 that simplifies NAT port mapping using UPnP (Universal Plug and Play). It is designed to help developers easily expose local services to external networks without requiring manual router configuration for users. By leveraging UPnP, QtNat automatically communicates with compatible routers to create port forwarding rules at runtime. This makes it particularly useful for peer-to-peer applications, multiplayer games, remote access tools, and any software that needs reliable inbound connectivity behind a NAT. QtNat provides a simplified API to do all steps automatically: discovery and mapping. This has been tested on my local device. Feel free to test it and improve it. Use it UpnpNat nat; QObject::connect(&nat, &UpnpNat::statusChanged, [&nat, &app]() { switch(nat.status()) { case UpnpNat::NAT_STAT::NAT_IDLE: case UpnpNat::NAT_STAT::NAT_DISCOVERY: case UpnpNat::NAT_STAT::NAT_GETDESCRIPTION: case UpnpNat::NAT_STAT::NAT_DESCRIPTION_FOUND: break; case UpnpNat::NAT_STAT::NAT_FOUND: nat.requestDescription(); break; case UpnpNat::NAT_STAT::NAT_READY: nat.addPortMapping("UpnpTest", nat.localIp(), 6664, 6664, "TCP"); break; case UpnpNat::NAT_STAT::NAT_ADD: qDebug() << "It worked!"; app.quit(); break; case UpnpNat::NAT_STAT::NAT_ERROR: qDebug() <<"Error:" <<nat.error(); app.exit(1); break; } }); nat.discovery(); We create the object (l:0) We connect to statusChanged signal to get notified (l:2) When status is NAT_FOUND, we request the description (l:11) When status is NAT_READY, we request the port mapping (l:14) When status is NAT_ADD, It means the port mapping request has been added, It worked! The application quits.(l:17) When status is NAT_ERROR, Error occured and display the error text. The application exits on error. (l:21) We connect to error changed in order to detect errors. (l:14) We start the discovery. (l:28) Technical explainations The discovery Basically, we need to know if there is a upnp server around. To do so, we send an M-SEARCH request on the multicast address. Here is the code: #define HTTPMU_HOST_ADDRESS "239.255.255.250" #define HTTPMU_HOST_PORT 1900 #define SEARCH_REQUEST_STRING "M-SEARCH * HTTP/1.1\n" \ "ST:UPnP:rootdevice\n" \ "MX: 3\n" \ "Man:\"ssdp:discover\"\n" \ "HOST: 239.255.255.250:1900\n" \ "\n" void UpnpNat::discovery() { setStatus(NAT_STAT::NAT_DISCOVERY); m_udpSocketV4.reset(new QUdpSocket(this)); QHostAddress broadcastIpV4(HTTPMU_HOST_ADDRESS); m_udpSocketV4->bind(QHostAddress(QHostAddress::AnyIPv4), 0); QByteArray datagram(SEARCH_REQUEST_STRING); connect(m_udpSocketV4.get(), &QTcpSocket::readyRead, this, [this]() { QByteArray datagram; while(m_udpSocketV4->hasPendingDatagrams()) { datagram.resize(int(m_udpSocketV4->pendingDatagramSize())); m_udpSocketV4->readDatagram(datagram.data(), datagram.size()); } QString result(datagram); auto start= result.indexOf("http://"); if(start < 0) { setError(tr("Unable to read the beginning of server answer"));

setStatus(NAT_STAT::NAT_ERROR); return; } auto end= result.indexOf("\r", start); if(end < 0) { setError(tr("Unable to read the end of server answer")); setStatus(NAT_STAT::NAT_ERROR); return; } m_describeUrl= result.sliced(start, end - start); setStatus(NAT_STAT::NAT_FOUND); m_udpSocketV4->close(); }); connect(m_udpSocketV4.get(), &QUdpSocket::errorOccurred, this, [this](QAbstractSocket::SocketError) { setError(m_udpSocketV4->errorString()); setStatus(NAT_STAT::NAT_ERROR); }); m_udpSocketV4->writeDatagram(datagram, broadcastIpV4, HTTPMU_HOST_PORT); } The whole goal of the discovery is to get the description file from the server with all available devices and services. The result is stored in m_describeUrl. Request Description file Simple request using QNetworkAccessManager. void UpnpNat::requestDescription() { setStatus(NAT_STAT::NAT_GETDESCRIPTION); QNetworkRequest request; request.setUrl(QUrl(m_describeUrl)); m_manager.get(request); } Parsing Description file Your physical network device may act as several Upnp devices. You are looking for one of these device type: urn:schemas-upnp-org:device:InternetGatewayDevice urn:schemas-upnp-org:device:WANDevice urn:schemas-upnp-org:device:WANConnectionDevice Those type are followed with a number (1 or 2), It is the Upnp protocol version supported by the device. void UpnpNat::processXML(QNetworkReply* reply) { auto data= reply->readAll(); if(data.isEmpty()) { setError(tr("Description file is empty")); setStatus(NAT_STAT::NAT_ERROR); return; } setStatus(NAT_STAT::NAT_DESCRIPTION_FOUND); /* Boring XML parsing in order to find devices and services. Devices: constexpr auto deviceType1{"urn:schemas-upnp-org:device:InternetGatewayDevice"}; constexpr auto deviceType2{"urn:schemas-upnp-org:device:WANDevice"}; constexpr auto deviceType3{"urn:schemas-upnp-org:device:WANConnectionDevice"}; Services: constexpr auto serviceTypeWanIP{"urn:schemas-upnp-org:service:WANIPConnection"}; constexpr auto serviceTypeWANPPP{"urn:schemas-upnp-org:service:WANPPPConnection"}; */ m_controlUrl = /* Most important thing to find the controlUrl of the proper service.*/ setStatus(NAT_STAT::NAT_READY); } Send mapping Request Sending a request is just sending HTTP request with the proper data. I use inja to generate the http data properly. This is the inja template. <?xml version="1.0" encoding="utf-8"?> <s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/" s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"> <s:Body> <u:AddPortMapping xmlns:u="{{ service }}"> <NewRemoteHost></NewRemoteHost> <NewExternalPort>{{ port }}</NewExternalPort> <NewProtocol>{{ protocol }}</NewProtocol> <NewInternalPort>{{ port }}</NewInternalPort> <NewInternalClient>{{ ip }}</NewInternalClient> <NewEnabled>1</NewEnabled> <NewPortMappingDescription>{{ description }}</NewPortMappingDescription> <NewLeaseDuration>0</NewLeaseDuration> </u:AddPortMapping> </s:Body> </s:Envelope> Then, let’s create a json object with all data. As final step, we need to create a request, set its data, and then post it. void UpnpNat::addPortMapping(const QString& description, const QString& destination_ip, unsigned short int port_ex, unsigned short int port_in, const QString& protocol) { inja::json subdata; subdata["description"]= description.toStdString(); subdata["protocol"]= protocol.toStdString(); subdata["service"]= m_serviceType.toStdString(); subdata["port"]= port_in; subdata["ip"]= destination_ip.toStdString(); auto text= QByteArray::fromStdString(inja::render(loadFile(key::envelop).toStdString(), subdata)); QNetworkRequest request; request.setUrl(QUrl(m_controlUrl)); QHttpHeaders headers; headers.append(QHttpHeaders::WellKnownHeader::ContentType, "text/xml; charset=\"utf-8\""); headers.append("SOAPAction", QString("\"%1#AddPortMapping\"").arg(m_serviceType)); request.setHeaders(headers); m_manager.post(request, text); } Finally, just check the answer The reply has no error, it worked, the status changes to NAT_ADD. Otherwise, the status changes to error. void UpnpNat::processAnswer(QNetworkReply* reply) { if(reply->error() != QNetworkReply::NoError) { setError(tr("Something went wrong: %1").arg(reply->errorString())); setStatus(NAT_STAT::NAT_ERROR); return; } setStatus(NAT_STAT::NAT_ADD); } Don’t hesitate to test it on your own device. Just to validate, it works everywhere. Any comment or change request, please use Github for that. Source code

- KDE Ships Frameworks 6.22.0 (2026/01/09 00:00)
  Friday, 9 January 2026 KDE today announces the release of KDE Frameworks 6.22.0. This release is part of a series of planned monthly releases making improvements available to developers in a quick and predictable manner. New in this version Baloo [FileWatchTest][KInotifyTest] Minor cleanup. Commit. [FileWatchTest][KInotifyTest] Cover atomic file replacing. Commit. [FileWatchTest] Cleanup and extend comments, correct wait condition. Commit. Bluez Qt Export header file for battery. Commit. Remove duplicate headers in same file. Commit. Breeze Icons Add new icon set: view-visible-off. Commit. Add kjournaldbrowser icon symlinks for flatpak app. Commit. Optionally disable icons library. Commit. Handle cross compiling for tools. Commit. Extra CMake Modules FindInotify: fix comparison. Commit. ECMAddTests: add functions and variables to preset test name prefix. Commit. KArchive 7z: Fix infinite loop in broken files. Commit. 7z: Fix infinite loop in malformed file. Commit. 7z: Fix infinite loop in malformed file. Commit. KCalendarCore Make build qml plugin optional (default on). Commit. KCMUtils Handle buttons changing at runtime. Commit. KCodecs [KEncodingProber] Add UTF16 surrogate pair support to state table. Commit. [KEncodingProber] Stop rejecting valid MSBs in UTF-16 state machine. Commit. [KEncodingProber] Reduce model switching in HandleData(). Commit. [KEncodingProber] Avoid passing mangled data to the UnicodeGroupProber. Commit. [KEncodingProber] Untangle HebrewProber and SBCharSetProber. Commit. [KEncodingProber] Minor cleanups for SBCharSetProber. Commit. [KEncodingProber] Various minor cleanups for HebrewProber. Commit. [KEncodingProber] Add test for Windows-1253/iso8859-7 CP (Greek). Commit. [KEncodingProber] Add some more testing for windows-1252 codepage. Commit. [KEncodingProber] Add tests for Windows-1255 CP and UTF-8 Hebrew text. Commit. [KEncodingProber] Simplify and improve unit test check condition. Commit. [KEncodingProber] Verify short inputs cause no crash. Commit. [KEncodingProber] Fix UTF-16 BOM detection. Commit. [KEncodingProber] Verify confidence in unit tests. Commit. [KEncodingProber] Remove no longer used dedicated Japanese/Chinese probers. Commit. [KEncodingProber][MBCS] Allow to use only a subset of probers. Commit. [KEncodingProber][MBCS] Remove unused includes from header file. Commit. KColorScheme Use Qt API instead of KColorSchemeWatcher. Commit. KConfig Add std::chrono convenience helpers. Commit. Add long and ulong as supported types. Commit. KConfigIniBackend::parseConfig: reuse allocated buffer to read group name. Commit. Use ECM_TEST_NAME_PREFIX. Commit. Fix FindNext shortcut on macos. Commit. KConfigWidgets Fix scanning for kf6_entry.desktop files in "locale" root dirs & parents. Commit. KContacts Use ECM_TEST_NAME_PREFIX. Commit. KCoreAddons Add OpenBSD available memory and refactor. Commit. Remove duplicate headers in same file. Commit. KDav Don't hide when the principals home sets fetch failed. Commit. Use ECM_TEST_NAME_PREFIX. Commit. KDeclarative DeclarativeDragArea: Filter mouse events according to acceptedButtons(). Commit. Fixes bug #384009 KDocTools Update Turkish entities. Commit. Update Turkish entities. Commit. KGuiAddons Ksystemclipboard: Use WaylandClipboard also on wayland-egl. Commit. Ksystemclipboard: Dispatch read events in another thread. Commit. Fixes bug #480448. Fixes bug #496029. Fixes bug #502831. Fixes bug #505281. Fixes bug #506467. Fixes bug #507792. Fixes bug #509065. Fixes bug #509689. Fixes bug #511736 Remove focus hacks as preparation for clipoard thread. Commit. Add manual test for ksystemclipboard. Commit. Deprecate KColorSchemeWatcher. Commit. KHolidays Update Japanese holidays for 2024-2026. Commit. Update holiday_cn_zh-cn for 2026 CN holidays. Commit. Add comprehensive list of Nepalese holidays for 2026-2030 when possible. Commit. KI18n Docs: expand org.kde.ki18n coverage. Commit. Remove duplicate headers in same file. Commit. KIconThemes KIconLoader: When processing SVG, skip whitespace and comments. Commit. KImageformats Add YCgCo-Re AVIF test. Commit. Avif: YCgCo-Re decoding fix. Commit. HEIF tests skipped using kde-ci.yml. Commit. Add allocation limit test (0/256 MiB). Commit. AllocationLimit = 0 means no limit. Commit. KIO Drop no longer needed moc include. Commit. Add manual test for FavIconRequestJob. Commit. KFilePlacesModel: Set desktop file name for partition manager. Commit. Remove duplicate headers

in same file. Commit. Deprecate PreviewJob::removeItem(QUrl). Commit. PreviewJob: fix empty enabledPlugins attribute set for thumbnail job. Commit. PreviewItem struct: drop unused cacheSize member. Commit. UDSEntryPrivate::load: reuse allocated buffer for string data. Commit. UDSEntryPrivate::load: avoid repeated look-up in cachedStrings list. Commit. Kfileplacesview: remove DropAction::MoveAction from the list of supportedActions. Commit. Fixes bug #509231 RenameDialog: Add "Compare Files" button. Commit. Use ECM_TEST_NAME_PREFIX. Commit. Kirigami Controls/SwipeListItem: remove dead validate function. Commit. Downgrade the message about not finding a plaform plugin to debug. Commit. Controls/NavigationTabButton: remove Qt 6.8 check. Commit. Layouts/FormLayout: avoid binding loop during initial load. Commit. Fixes bug #513185 Controls/GlobalDrawer: explicitly pass backItem. Commit. Controls/private: pass refreshing to PullDownIndicator. Commit. Autotests: linting changes. Commit. Examples: linting. Commit. Drop non-existing handler parameters. Commit. Primitives/Separator: import primitives. Commit. Tests: various linting fixes for the manual tests. Commit. SwipeListItem: Use implicitContentHeight/Width instead of contentItem.implicitHeight/Width. Commit. Avoid custom-parsed PropertyChanges. Commit. Remove unused imports. Commit. Port to ComponentBehavior Bound and required properties. Commit. Qualify access to parent properties. Commit. Controls/NavigationTabButton: fix pointSize binding. Commit. Templates/AbstractApplicationHeader: always set preferredHeight. Commit. Controls/Action: fix alternate shortcut. Commit. Port to Application.styleHints. Commit. Controls/AboutItem: qualify Separator. Commit. SwipeListItem: Add back checking for parent width and implicitWidth. Commit. Add extra margin only for default title. Commit. Documentation fixes. Commit. Document requirements for new components. Commit. Add TitleSubtitleWithActions. Commit. KItemModels KDescendantsProxyModel: fix invalid call to index(-1,-1). Commit. KJobWidgets Remove duplicate headers in same file. Commit. KPackage Fix typo in apidocs. Commit. KService Remove FormFactor handling. Commit. Drop Library handling. Commit. Drop allowAsDefault handling. Commit. KStatusNotifieritem Tweak KStatusNotifierItem::setAssociatedWindow. Commit. KSVG Remove noisy debug on ImageSetPrivate::findInCache. Commit. Remove duplicate headers in same file. Commit. KTextEditor Correction: fixes typo introduced in previous commit. Commit. Fix various typos. Commit. KUserFeedback CI - Flatpak - Update Runtime to 6.10. Commit. KWidgetsAddons KPageView: Fix top aligned widget stretch. Commit. Remove duplicate headers in same file. Commit. KDateTable: Don't paint days in dark red if high-contrast is active. Commit. KPageView: Use correct icon mode if high-contrast color scheme is in use. Commit. Add helper for checking if high-contrast color scheme is in use. Commit. KWindowSystem Add KWaylandExtras::{setXdgToplevelTag,setXdgToplevelDescription}. Commit. See bug #512447 Use ECM_TEST_NAME_PREFIX. Commit. KXMLGUI Remove dead gestures code. Commit. Solid /etc/mtab not exists on OpenBSD. Commit. Fix some missing overrides in win backend. Commit. Sonnet Use ECM_TEST_NAME_PREFIX. Commit. Syntax Highlighting Add support for Toit. Commit. Systemd unit: update to systemd v259. Commit. Add Mermaid syntax highlighting. Commit. Implements feature #494286 New feature: stacking several contexts at the same time. Commit. Dot: add Graphviz as alternative name. Commit. Add new QML keywords. Commit.

- Events in December 2025 (2026/01/08 22:00)
  December was quite an eventful month for me, with over 4,000 km travelled by train. This was in part caused by the holidays and visiting family, but also by the KDE PIM sprint in Paris and the 39th Chaos Communication Congress. KDE PIM sprint in Paris From the 12th to the 14th of December, I was in Paris. It was actually my first time there for more than a day trip, so I arrived a day earlier to explore the city a bit. I went on a walk across the city with Tobias and Nicolas, and I took some photos. The weekend was also very productive. We advanced our goal of making KMime a proper KDE Framework; made Message-IDs in emails more privacy-conscious; and discussed various important topics such as the retirement of the Kolab resource and the switch to SQLite as the default backend for Akonadi. Huge thanks to enioka Haute Couture for having us

in their office in Paris. The sprint being in Paris also allowed me to afterward go visit my grandma, 350 km further south of Paris, so this was particularly convenient. 39th Chaos Communication Congress (39c3) Another event I went to was 39c3, which is the third year in a row that I attended, and this year again we had an assembly as part of the Bits und Bäume umbrella, thanks to Joseph. I love the vibe of this event. It’s not very dry or only tech-focused, but also has a big artistic and political aspect to it. And while the number of attendees is very large, at the same time it’s very chill and I don’t feel overwhelmed, unlike at FOSDEM. At the KDE assembly, we met a lot of interested users, some GNOME friends, and since a bunch of KDE devs were there, we managed to work on a few productive things, like switching the map backend from Itinerary to MapLibre. And this year, I even managed to go on national TV for a few seconds to speak about Nextcloud. My German grandma called me the day afterward, very happy to have seen me.

- KDE Gear 25.12.1 (2026/01/08 00:00)
  Over 180 individual programs plus dozens of programmer libraries and feature plugins are released simultaneously as part of KDE Gear. Today they all get new bugfix source releases with updated translations, including: dolphin: Allow migration for users with the old session file format (Commit, fixes bug #513466) kate: Fix a tree view crash in the Project plugin (Commit, fixes bug #513753) skladnik: Ignore the mouse release event at the end of a drag action (Commit) Distro and app store packagers should update their application packages. 25.12 release notes for information on tarballs and known issues. Package download wiki page 25.12.1 source info page 25.12.1 full changelog
- Kraft 2.0 Announcement (2026/01/06 10:14)
  With the start of the new year, I am very happy to announce the release of version Kraft 2.0.0. Kraft provides effective invoicing and document management for small businesses on Linux. Check the feature list. This new version is a big step ahead for the project. It does not only deliver the outstanding ports to Qt6 and KDE Frameworks 6 and tons of modernizations and cleanups, but for the first time, it also does some significant changes in the underlying architecture and drops outdated technology. Kraft now stores documents not longer in a relational database, but as XML documents in the filesystem. While separate files are more natural for documents anyway, this is paving the way to let Kraft integrate with private cloud infrastructures like OpenCloud or Nextcloud via sync. That is not only for backup- and web-app-purposes, but also for synced data that enables to run Kraft as distributed system. An example is if office staff works from different home offices. Expect this and related usecases to be supported in the near future of Kraft. But there are more features: For example, the document lifecycle was changed to be more compliant: Documents remain in a draft status now until they get finalized, when they get their final document number. From that point on, they can not longer be altered. There is too much on the long Changes-List to mention here. However, what is important is that after more than 20 years of developing and maintaining this app, I continue to be motivated to work on this bit. It is not a big project, but I think it is important that we have this kind of “productivity”-applications available for Linux to make it attractive for people to switch to Linux. Around Kraft, a small but beautiful community has built up. I like to thank everybody who contributed in any way to Kraft over the years. It is big fun to work with you all! If you are interested, please get in touch.
- What does it mean to write in 2026? (2026/01/04 21:58)
  I've been writing for something like 50 years now. I started by scribbling letters on paper as a child because I was fascinated that these expressed meaning. I wrote a lot for school, for university, for work, and privately. I wrote letters, emails, posts on social media, articles, papers, documentation, diaries, opinion pieces, and presentations. I've been writing my blog for more than 20 years.Writing always has been a way for me to connect to the people, to the community, around me, communicating with my tribe. It also has always been a way to express, refine and

archive my thoughts, a bit like building a memory of insights. It also has been a way to record some of my personal history and the history of the projects I'm involved with.My writing has changed over the last couple of years. I'm writing less publicly and more focused on specific projects. It feels like it has become less personal and more utilitarian.Part of this is that the Internet has lost a good part of its strength as a neutral platform to reach the world. For a long time I knew where to reach the people I wanted to address and had control about my content and how it was distributed. Nowadays social media platforms act as distributors, but we are prey to their algorithms. So while publishing content is still simple, it's much harder to get it to your audience without compromising to the mechanisms which make the algorithms tick.Another part is the disrupting advance of AI writing capabilities. While I have relied on humans to give me feedback in the past, to get into a conversation on the topics of my posts to refine the thoughts in them, now there is this all-powerful-seeming assistant in my editor who is eager to take over those roles. And it would even write for me in my own style. So what's the value of writing in 2026? Is it even worth bothering with trying to express your thoughts in writing, when a machine can produce content which looks the same, much faster and in much larger quantity? What does this do to readers, do they still care about what I would write?My feeling is that it's still worth to put in effort to create genuine, trustworthy, truthful writing. The format, the tools, the channels might change, but the values don't. The challenge will be to figure out how to create a signal which transports these values.I have always liked the format and style of a blog, as a stream of thoughts, coming from a personal perspective, but focused on topics of relevance to others. I enjoy reading this from others and I enjoy writing in this style. And I don't have to rely on a platform I don't control, but can use my own.So it looks like this blog won't go away, but will channel my thoughts in 2026 as well.