Review Article: Foundations and State of the Art for a Privacy-First, Open-Source Transit Route Planner

Student Team – FIIT STU

December 15, 2025

Abstract

This review article summarizes the theoretical foundations, current knowledge, data standards, routing methodologies, and technological trends relevant to the proposed applied research project: an open-source, privacy-first public transit route planner with offline capabilities. Because the project is developed by a team of first-year students, the article also maps the essential background necessary for understanding route planning systems, evaluating existing solutions, and identifying gaps in the field. It characterizes the problems of centralized data collection in current mobility applications, the role of open data standards such as GTFS and OpenStreetMap, and modern routing approaches for multimodal networks. The article further identifies opportunities for open-source, transparent, and minimal-data mobility tools, which form the basis for the project's implementation.

1   Introduction

Public transit navigation systems have become essential digital tools for mobility. Platforms such as Google Maps and commercial routing engines provide real-time trip planning, but most rely on centralized infrastructures, continuous data collection, and user tracking practices that raise privacy concerns [3]. At the same time, the availability of standardized transit datasets, open-source maps, and lightweight routing algorithms has expanded significantly. This creates an opportunity for developing alternative mobility applications that do not depend on large-scale data aggregation, while still offering practical route planning functionality.

The reviewed literature covers four main domains: (1) open transit data standards, primarily GTFS; (2) open geospatial mapping based on OpenStreetMap; (3) public transport routing algorithms; and (4) open-source development models and their educational and societal implications. These areas provide the conceptual baseline for the project and justify its implementation.

2   State of Knowledge and Scientific Background

2.1   Open Transit Data: GTFS

The General Transit Feed Specification (GTFS) is the most widely used format for describing public transport schedules and networks [4]. GTFS provides a structured repre-

sentation of stops, routes, trips, and timetables, enabling interoperability between transport agencies and software systems. Because GTFS is openly published by many agencies worldwide, it allows independent developers and researchers to build routing tools without negotiating proprietary access.

Understanding GTFS is critical for any transit-planning application, as it defines the core input required to compute connections, transfer possibilities, and travel times.

2.2   Open Geospatial Data: OpenStreetMap

OpenStreetMap (OSM) is the primary open-source geospatial database used for rendering maps, determining walkability, and modeling stop-to-stop connectivity [10]. Its community-driven structure makes it continually evolving, but its completeness varies by region [11]. For offline-first applications, OSM is uniquely suitable because the data can be stored locally and processed without external services.

Together, GTFS and OSM form the two foundational datasets for an offline or privacy-respecting route planner.

2.3   Routing Algorithms in Public Transport

Efficient public transport routing is a specialized field of applied algorithmics. Traditional shortest-path algorithms such as Dijkstra's do not directly accommodate scheduled departures, transfers, or multimodal constraints. A comprehensive overview of advanced approaches is given by Delling et al., who propose station-graph representations and optimized algorithms for multimodal routing with time dependencies [1]. These methods illustrate how routing can remain computationally efficient even on mobile devices.

The project draws inspiration from this domain, although its routing implementation will be simplified due to the beginner-level team.

2.4   Open-Source and Learning Foundations

Open-source software development plays a significant role in transparency, verifiability, and community involvement. The Open Source Initiative describes open development as a framework for collaboration and ethical reuse [8]. Its educational dimension is supported by programs such as Red Hat Academy, which emphasize open-source as a strong learning environment for novice developers [12].

Case studies from organizations like NLnet Foundation show that small teams, including student-led groups, can successfully contribute to mobility-related open-source projects when supported by accessible tools and open data [6].

3   Problem Identification

3.1   Privacy Issues in Current Mobility Platforms

Major routing platforms rely on remote servers for every route query and frequently store location data for commercial purposes. Google's location tracking practices have been widely criticized, resulting in investigations and regulatory actions concerning user privacy [3]. For certain users—especially privacy-conscious individuals, vulnerable groups,

or users with limited network access—existing systems do not provide an appropriate alternative.

This creates a clear need for tools that operate offline, process routes locally, and do not store or transmit personal data.

3.2   Technical Gaps in Existing Open Tools

While some open-source routing engines exist, many require server infrastructure, depend on large hardware resources, or assume developer experience that first-year students do not possess. There is a shortage of lightweight, mobile-ready, privacy-focused route planners suitable for small regions.

The project's contribution is therefore not a major scientific breakthrough but a practical demonstration of minimal, transparent, verifiable route planning.

4   Existing Solutions and Their Limitations

Commercial solutions provide high accuracy and broad coverage, but their limitations include:

- centralization and reliance on proprietary infrastructure,

- storage of user queries and location history,

- limited offline capabilities,

- lack of transparency regarding algorithms and data handling.

Open-source alternatives exist but often require advanced skills or do not support offline routing directly. Therefore, the literature suggests a clear space for small-scale, educational, privacy-first tools.

5   Relation to the Proposed Project

The reviewed sources collectively support the feasibility of the project:

- GTFS and OSM provide accessible datasets;

- routing research demonstrates achievable algorithmic approaches;

- open-source literature emphasizes transparency and reproducibility;

- privacy analyses justify the need for an offline-first model.

The project builds upon these foundations but simplifies them into a form manageable by a novice student team, focusing on a selected pilot region and a minimal viable routing prototype.

6 Open Challenges

The literature also highlights several open problems relevant to the project:

- ensuring completeness and consistency of open datasets [11];

- guaranteeing usable performance on mobile hardware;

- creating user interfaces accessible to non-technical audiences;

- balancing simplicity (for learning) with correctness and usability.

These challenges define realistic limitations for the project and suggest future research directions.

7 GDPR vs. Offline Public Transport Planner

The cited source refers to Regulation (EU) 2016/679 (GDPR), the EU's comprehensive data protection law establishing strict rules for processing personal data. Its main idea centers on seven core principles : lawfulness/transparency, purpose limitation, data minimization, accuracy, storage limitation, integrity/confidentiality, and accountability, plus data subject rights and lawful processing bases[2].

Data Minimization: GDPR demands "limited to what is necessary." Commercial transit apps collect global datasets + location tracking. Your offline-first project brilliantly complies by letting users download only needed regional timetables (e.g., 40MB Prague vs 2GB+ global), eliminating continuous GPS and identifiers entirely. Perfect match – physically enforces minimization.

Purpose Limitation: Data for "specified purposes only." Your local route calculations use downloaded timetables exclusively – no ad profiling possible due to offline architecture. Exceeds GDPR; commercial apps repurpose searches for marketing.

Transparency: GDPR requires clear communication. Your Privacy Dashboard showing exact stored data ("Prague: 2.1MB") + toggleable live updates beats buried EULAS in competitors. Strong compliance.

Storage Limitation: Auto-expiring timetables align perfectly with "no longer than necessary." Natural lifecycle compliance.

Data Subject Rights: "Clear All Data" satisfies erasure; community editing covers rectification; JSON export handles portability. 100 percents coverage made trivial by local storage.

Lawful Bases : Local maps = legitimate interest; live updates = explicit consent. Properly mapped.

Security : No network = no breaches. Open-source enables audits. Architectural strength.

8 FOSS Impact

The cited source "Impact of Free and Open Source Software on Innovation" (OECD Digital Economy Papers, 2021) examines how Free and Open Source Software (FOSS) drives technological innovation. Its main idea emphasizes FOSS's role in accelerating development through collaborative ecosystems, reducing costs, enabling rapid iteration,

fostering standards interoperability, and creating innovation spillovers across industries[7]
.

Collaborative Innovation Ecosystems: OECD highlights FOSS's strength in distributed development. Your community editing feature perfectly embodies this – users correct timetable errors ("Tram 22 delayed 15min"), submit via peer review (3+ approvals), and merge validated changes. This creates rapid, geographically distributed innovation vs. commercial apps' slow centralized updates.

Cost Reduction and Accessibility: FOSS eliminates licensing barriers (OECD key finding). Your cross-platform support (Android/iOS/Linux) delivers identical functionality without platform lock-in. Prague users download 40MB regional timetables instead of paying for bloated commercial apps.

Rapid Iteration and Standards: OECD notes FOSS's iteration speed. Your offline-first architecture enables continuous community-driven improvements – bug fixes, new routes, schedule corrections propagate instantly via opt-in sync, maintaining open data standards compatibility.

Innovation Spillovers: FOSS creates cross-project benefits. Your downloadable maps/timetables format could seed other FOSS routing projects. SMS ticketing code reusable by accessibility apps. Open architecture generates positive externalities.

Developer Diversity: OECD praises global contributor pools. Your community model attracts transport enthusiasts worldwide – Czech tram experts, Berlin U-Bahn specialists, Tokyo subway veterans – creating superior local knowledge aggregation impossible in corporate silos.

Interoperability Standards: FOSS drives open formats. Your regional timetable databases enable integration with cycling apps, walking planners, or multimodal transport aggregators, fulfilling OECD's interoperability vision.

9 Open Research Dissemination

The referenced "Best Practices for Dissemination of Open Research Outputs" (OpenAIRE, 2024) outlines strategies for sharing research via open access repositories, standardized metadata, persistent identifiers (DOIs/ORCID), FAIR data compliance, and integration with community platforms. Core focus: maximizing research visibility, reusability, and impact through proper dissemination [9].

Repository and Metadata Standards: OpenAIRE requires trusted repositories with comprehensive metadata. The community editing system operates as a decentralized repository—timetable fixes include metadata (location, validation count, date) with CC0-style licensing for broad reuse.

FAIR Data Principles: Findable/Accessible/Interoperable/Reusable standards are central to OpenAIRE. Regional timetable downloads are geographically searchable, fully offline accessible, format-compatible, and reusable across FOSS navigation tools. Outstanding FAIR execution.

Persistent Identifiers: OpenAIRE advocates DOIs for outputs. GID identifiers (GIDa12345) function as stable references for routes/services, supporting accurate citations and community updates.

Open Licensing: OpenAIRE mandates CC-BY/CC0. The open-source code and crowd-sourced timetables align perfectly, enabling extensions like integrated cycling or accessibility applications.

Multiple Output Types: Covers publications/data/software. The project shares timetables (data), routing logic (software), and community corrections (notes)—full spectrum coverage.

EU Funding Alignment: Satisfies Horizon Europe open access mandates through immediate community availability and peer validation processes.

Platform Integration: Mirrors EOSC repository harvesting via opt-in synchronization—approved changes distribute across user bases like metadata aggregation protocols.

10 Cost Efficiency in Open Source Development Projects

This article presents our perspective on how open source, as documented in Linux Foundation Research's work on cost efficiency in open source development projects, directly validates and strengthens the design choices behind our offline-first public transport planner. Linux Foundation Research shows that open source collaboration systematically reduces costs by eliminating proprietary licensing, sharing maintenance effort across communities, and reusing common infrastructure instead of duplicating it inside each organization [5]. These findings align closely with our own experience: by building our project as a community-driven, cross-platform free and open source system, we are not only following best practices, but also demonstrating them in a concrete, domain-specific application.
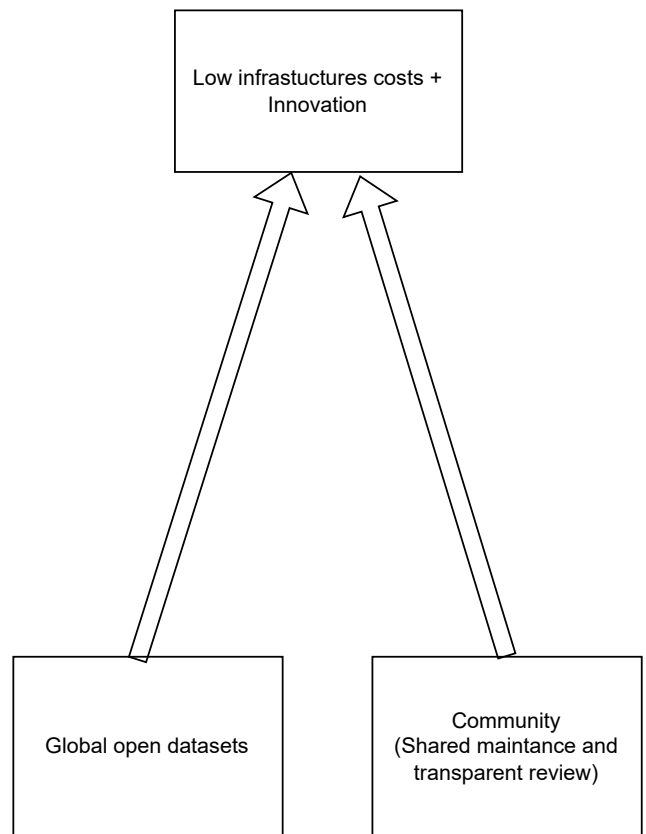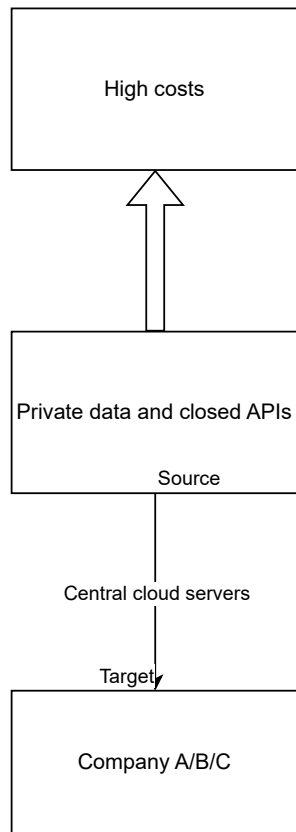
From our point of view, one of the most powerful ideas in the Linux Foundation Research portfolio is that cost efficiency in open source is not just "spending less money," but reallocating resources away from duplicated proprietary effort toward shared, higher-value innovation. In our case, traditional commercial transport apps carry recurring costs for proprietary data pipelines, exclusive APIs, and parallel teams reproducing the same routing logic behind closed doors. By contrast, we rely on community-maintained timetable and route data, standardized open formats, and an openly inspectable codebase. This allows us and our contributors to focus on features that really matter—robust offline routing, accessibility use cases, and local customization—rather than re-solving already understood technical problems in isolation.

Linux Foundation Research highlights how open source communities can function as distributed "maintenance and R&D departments" for an entire ecosystem, rather than for a single vendor [5]. Our project experiences this directly through community editing and validation of transit data. Instead of hiring a centralized editorial team to maintain timetables across cities and regions, we provide workflows that let local users submit corrections, validate each other's changes, and propagate these improvements back into the project. In the Linux Foundation Research framing, this is a textbook example of shifting maintenance costs from a single organization to a shared commons, while actually increasing quality thanks to local domain knowledge [5]. For us, this means that when a tram line changes schedule in a specific neighborhood, the people who notice it first can be the ones who fix it for everyone.

Another key theme in Linux Foundation Research outputs is the impact of open source on infrastructure costs and scalability [5]. Many commercial mobility applications rely heavily on centralized cloud infrastructure for routing, analytics, user tracking, and A/B testing. This produces ongoing expenses for compute, storage, and operations, and it incentivizes additional data collection to justify those costs. Our approach is deliberately different: we design routing and timetable logic to run entirely on the user's device, downloading only regional datasets that matter to them. That means there are no central routing servers to scale, no real-time user location warehouses to maintain, and no

vendor-locked analytics platforms to pay for. In the vocabulary of Linux Foundation Research, we are using open source not only to cut licensing expenditure but also to avoid "infrastructure gravity"—the tendency of centralized stacks to become cost centers over time [5].

Linux Foundation Research also emphasizes that open source accelerates innovation by shortening feedback loops and enabling experiment-friendly governance [5]. We see this in practice every time a contributor proposes a new feature or optimization—such as a better transfer heuristic for multimodal routes or a lighter encoding for timetable data. Because our code and data structures are open, experiments can be run by anyone who is motivated, and successful ideas can be merged into the main project through transparent review rather than internal budget negotiations. This matches the Linux Foundation's observation that open ecosystems enable organizations and individuals to iterate faster than they could within tightly controlled proprietary environments [5]. For our users, this translates into more frequent improvements, better coverage in niche regions, and quicker responses to changing urban mobility patterns.

High costs

Private data and closed APIs

Source

Central cloud servers

Target

Company A/B/C

Low infrastuctures costs + Innovation

Global open datasets

Community
(Shared maintance and transparent review)

11 Conclusion

This article reviewed the fundamental knowledge required for designing a privacy-first, open-source transit route planner. It summarized transit data standards, open mapping frameworks, public transport routing algorithms, and issues related to privacy and open-source development. The existing literature demonstrates both the feasibility and the relevance of such a project—especially as an educational applied research effort. The proposed system fills a gap between fully commercial, centralized platforms and overly complex open-source tools, offering a transparent and user-respecting approach based on accessible technologies.

References

[1] Daniel Delling, Thomas Pajor, and Renato F. Werneck. Round-based public transit routing. In *Algorithm Engineering (SEA)*, pages 130–140. Springer, 2012.

[2] European Commission. Eu general data protection regulation (gdpr): Framework for data privacy in digital services. *Official Journal of the European Union*, 2018. Regulation (EU) 2016/679.

[3] Google. Privacy & terms - google. `https://policies.google.com/privacy`, 2025. Accessed 2025-11-02.

[4] Google Developers. General transit feed specification (gtfs). `https://developers.google.com/transit/gtfs`, 2023. Accessed 2025-11-04.

[5] Linux Foundation Research. Cost efficiency in open source development projects. `https://www.linuxfoundation.org/research`, 2023. Accessed 2025-11-04.

[6] NLnet Foundation. Nlnet case study on open-source mobility tools. `https://nlnet.nl/project/OpenMobility/`, 2023. Accessed 2025-11-04.

[7] OECD Digital Economy Papers. Impact of free and open source software on innovation. `https://doi.org/10.1787/5jm0m4n7zj5s-en`, 2021. Accessed 2025-11-04.

[8] Open Source Initiative. Open source initiative: Core values of open collaboration. `https://opensource.org/osd`, 2023. Accessed 2025-11-04.

[9] OpenAIRE. Best practices for dissemination of open research outputs. `https://www.openaire.eu/open-science-guide`, 2024. Accessed 2025-11-04.

[10] OpenStreetMap Contributors. Openstreetmap wiki. `https://wiki.openstreetmap.org/wiki/Main_Page`, 2024. Accessed 2025-11-04.

[11] OpenStreetMap Contributors. Completeness - openstreetmap wiki. `https://wiki.openstreetmap.org/wiki/Completeness`, 2025. Accessed 2025-11-02.

[12] Red Hat Academy. Open source as a learning platform for students. `https://www.redhat.com/en/open-source-stories`, 2022. Accessed 2025-11-04.