UDC [004.774:37](043.2)

A.O. Lavreniuk
V.P. Maidaniuk

COMPARATIVE ANALYSIS OF DATA TRANSMISSION PROTOCOLS FOR EDUCATIONAL WEB PLATFORMS

Vinnytsia National Technical University

Анотація

У тезах проведено порівняльний аналіз сучасних протоколів передачі даних – HTTP/1.1, HTTP/2, HTTP/3 (QUIC), WebSocket та gRPC – з точки зору їх застосування в освітніх веб-платформах. Розглянуто ключові характеристики кожного протоколу: затримка, пропускна здатність, мультиплексування запитів, підтримка реального часу та поведінка в умовах нестабільних мереж. Проаналізовано функціональні та нефункціональні вимоги до сучасних освітніх систем і встановлено відповідність між архітектурними компонентами платформи та оптимальними протоколами. Запропоновано гібридну протокольну стратегію, яка поєднує HTTP/2 або HTTP/3 для завантаження статичного контенту, WebSocket для інтерактивної взаємодії в реальному часі та gRPC для міжсервісної комунікації. Результати дослідження можуть бути використані при проектуванні масштабованих освітніх веб-платформ.

Ключові слова: протоколи передачі даних, HTTP/2, HTTP/3, WebSocket, gRPC, QUIC, освітні веб-платформи, порівняльний аналіз, архітектура програмних систем.

Abstract

This paper presents a comparative analysis of modern data transmission protocols – HTTP/1.1, HTTP/2, HTTP/3 (QUIC), WebSocket, and gRPC – from the perspective of their application in educational web platforms. Key characteristics of each protocol are examined: latency, throughput, request multiplexing, real-time interaction support, and behavior under unreliable network conditions. The functional and non-functional requirements of modern educational systems are analyzed, and the correspondence between platform architectural components and optimal protocols is established. A hybrid protocol strategy is proposed, combining HTTP/2 or HTTP/3 for static content delivery, WebSocket for real-time interactive communication, and gRPC for inter-service communication. The research findings can be applied when designing scalable educational web platforms.

Keywords: data transmission protocols, HTTP/2, HTTP/3, WebSocket, gRPC, QUIC, educational web platforms, comparative analysis, software architecture.

Introduction

The global shift toward digital education, significantly accelerated by the COVID-19 pandemic, has positioned educational web platforms as critical infrastructure for modern universities and schools. Platforms supporting course delivery, collaborative document sharing, real-time lecture streaming, and peer content exchange must simultaneously satisfy demanding and often contradictory requirements: low latency for interactive features, high throughput for file transfers, reliability under unstable mobile connections, and scalability to thousands of concurrent users during examination periods [1].

At the network communication layer, the choice of data transmission protocol fundamentally determines whether these requirements can be met. While the web has historically relied on HTTP/1.1, the past decade has introduced a succession of protocol improvements – HTTP/2, HTTP/3 based on the QUIC transport, WebSocket for bidirectional communication, and gRPC for efficient binary RPC – each with distinct performance profiles and applicability boundaries [2]. Yet protocol selection in educational platform design is frequently ad hoc, driven by developer familiarity rather than systematic performance analysis.

This paper addresses the gap by conducting a structured comparative analysis of five major data transmission protocols relevant to educational web platforms, evaluating each against criteria derived from the typical requirements of peer learning and content exchange systems. The goal is to provide evidence-based guidance for protocol selection decisions in educational web platform architecture.

Background and Related Work

Research on protocol performance for web applications has produced a substantial body of literature. Biswal and Naskar [3] demonstrated that HTTP/2 reduces page load times by 40–60% compared to HTTP/1.1

in high-latency conditions, primarily due to header compression (HPACK) and multiplexing. Marx et al. [4] showed that HTTP/3 outperforms HTTP/2 in lossy network environments – conditions representative of mobile learners in areas with poor connectivity – by eliminating head-of-line blocking at the transport layer through the QUIC protocol.

For real-time features, Lubbers et al. [5] established WebSocket as the de facto standard for persistent bidirectional connections, with connection overhead amortized over session duration. In the microservices context relevant to scalable educational platforms, Pismenny et al. [6] demonstrated gRPC's throughput advantage over REST/HTTP/2 of approximately 7–10× for high-frequency inter-service calls, attributable to Protocol Buffers binary serialization.

However, a systematic comparison specifically targeting educational platform requirements – including the interplay between content delivery, real-time notification, file upload, and administrative analytics traffic – remains underrepresented in the literature. This paper contributes such a comparison, structured around the distinct functional components of a representative educational platform.

Requirements of Educational Web Platforms

To ground the protocol comparison in realistic operational conditions, we identify the characteristic functional components and their network communication patterns in a modern educational content exchange platform. These components represent the architectural building blocks present in virtually all major educational platform categories, from LMS systems to peer learning repositories.

Content discovery and search requires efficient request-response communication between the browser and a full-text search backend. Queries are short-lived but latency-sensitive – users expect sub-second results. Multiple concurrent filtered queries may originate from a single page load, making multiplexing capability important [7].

File upload and download represents the highest-throughput communication pattern, involving large binary payloads (PDF, DOCX, PPTX files up to 50 MB). The protocol must support resumable transfers and provide status feedback to the user without polling, motivating the need for either server-sent events or WebSocket status channels.

Real-time notifications are required when the system needs to push asynchronous events to connected clients – antivirus scan completion, new comments, moderation decisions. A persistent connection model (WebSocket or SSE) eliminates polling overhead for thousands of concurrent users.

Authentication and session management involves frequent short token validation requests. Security requirements mandate TLS 1.2 or higher and protection against brute-force credential attacks, influencing the choice of protocols that natively integrate with modern TLS stacks.

Administrative analytics and inter-service communication between platform microservices (search engine, file storage, notification dispatcher, antivirus scanner) generate high-frequency internal API traffic. Binary protocols with compact serialization offer significant efficiency advantages at this layer [6].

These five communication patterns define the evaluation framework for the protocol comparison in the following section.

Comparative Analysis of Data Transmission Protocols

Five protocols were evaluated based on six criteria derived from the requirements identified above: latency, throughput, request multiplexing, real-time interaction support, performance under lossy/mobile networks, and browser support breadth. Each criterion is scored on a 5-point scale (1 = poor, 5 = excellent) based on published benchmark data and protocol specifications.

HTTP/1.1, despite universal browser support, is the weakest performer on nearly all criteria relevant to educational platforms. Its synchronous request-response model creates head-of-line blocking that severely limits concurrency, making it unsuitable as the primary protocol for any modern platform component [2].

HTTP/2 addresses the main limitations of HTTP/1.1 through binary framing, HPACK header compression, and full request multiplexing over a single TCP connection. Page load times for resource-rich educational content pages improve significantly, and server push can be used to pre-deliver common assets. However, HTTP/2 inherits TCP's head-of-line blocking at the transport layer, which degrades performance when packet loss occurs – a common condition for mobile users [3].

HTTP/3, built on the QUIC transport protocol over UDP, eliminates TCP head-of-line blocking entirely. Each stream is independently flow-controlled, so a lost packet in one stream does not block other streams. This makes HTTP/3 the strongest choice for file delivery and API communication in mobile learning contexts. Its

0-RTT connection establishment also reduces perceived latency for returning users. Browser support has reached approximately 95% globally as of 2024, though server-side QUIC support requires more recent infrastructure [4].

Table 1 – Comparative evaluation of data transmission protocols for educational web platforms

Criterion	HTTP/1.1	HTTP/2	HTTP/3 (QUIC)	WebSocket	gRPC	Best for EIS
Request latency	2	4	5	4	5	HTTP/3, gRPC
File throughput	2	4	5	3	5	HTTP/3, gRPC
Request multiplexing	1	5	5	4	5	HTTP/2, HTTP/3, gRPC
Real-time interaction	1	2	3	5	4	WebSocket
Mobile / lossy networks	2	3	5	3	4	HTTP/3
Browser support	5	5	4	5	3	HTTP/1.1, HTTP/2, WS
Total score	13	23	27	24	26	

WebSocket provides a persistent, full-duplex TCP connection established through an HTTP upgrade handshake. Its throughput for bidirectional messaging far exceeds polling-based alternatives, and it is the standard choice for real-time features: live notifications of file processing status, comment delivery, and collaborative annotation. WebSocket connections persist across the session lifetime, making connection overhead negligible for active users [5].

gRPC, built on HTTP/2 and Protocol Buffers, is designed for efficient inter-service communication in microservices architectures. Its binary serialization is approximately 5–7× more compact than JSON, and its streaming capabilities support server streaming, client streaming, and bidirectional streaming patterns. The primary limitation is browser support: direct gRPC from browsers requires the gRPC-Web proxy layer, which adds operational complexity [6].

Protocol Selection Strategy for Educational Platforms

The analysis supports a hybrid protocol strategy in which different platform components use the protocol most suited to their communication pattern. This approach avoids the performance compromises of a single-protocol architecture.

For client-server content delivery – search API responses, material metadata, user profiles, and authentication – HTTP/2 is recommended as the baseline, with HTTP/3 adopted where infrastructure permits. The multiplexing and header compression of HTTP/2 address the high concurrency of search and browse workflows, while HTTP/3 provides resilience for mobile users. Servers should negotiate the protocol via ALPN TLS extension, enabling transparent upgrade without client code changes [7].

Table 2 – Recommended protocol mapping by educational platform component

Platform Component	Recommended Protocol	Key Rationale	Fallback
Content search & catalog API	HTTP/2 / HTTP/3	Multiplexing; HPACK compression	HTTP/1.1 + keepalive
File upload (PDF, DOCX, PPTX)	HTTP/3 (QUIC)	Stream independence; no HOL block	HTTP/2 with chunked transfer
File download & preview	HTTP/2 / HTTP/3	Parallel streams; server push	HTTP/1.1 with CDN
Real-time notifications (scan, comments)	WebSocket	Full-duplex; low overhead per msg	Server-Sent Events (SSE)
Authentication & OAuth flow	HTTPS (HTTP/2)	TLS 1.2+; short sessions	HTTP/1.1 + TLS
Inter-service (search ↔ API ↔ antivirus)	gRPC	Binary serialization; streaming	REST / HTTP/2 JSON
Administrative analytics dashboard	HTTP/2 + WebSocket	Bulk data + live metric push	HTTP/1.1 with polling

For file upload and download of large educational materials, HTTP/3 with QUIC is the preferred choice where available, due to its stream independence under packet loss. For upload progress reporting, a WebSocket channel alongside the upload HTTP request eliminates the need for polling and enables server-push of antivirus scan completion events without increasing connection count [5].

Security requirements constrain all protocol choices: TLS 1.2 or higher is mandatory across the entire platform, and QUIC natively integrates TLS 1.3, meaning HTTP/3 deployments automatically satisfy this requirement. For WebSocket connections, the wss:// scheme over TLS must be enforced. Rate limiting (maximum five failed authentication attempts before temporary lockout) and input sanitization against injection attacks are orthogonal to protocol choice but must be implemented at the application layer regardless of the selected transport [1].

Scalability requirements influence protocol selection in a secondary way: protocols with persistent connections (WebSocket, gRPC streaming) require connection-aware load balancing (Layer 7 proxies such as NGINX or Envoy) rather than simple round-robin TCP balancing. The platform's load balancer must maintain session affinity for WebSocket connections and support QUIC UDP forwarding for HTTP/3 [8].

Conclusions

This paper has conducted a systematic comparative analysis of five data transmission protocols – HTTP/1.1, HTTP/2, HTTP/3 (QUIC), WebSocket, and gRPC – in the context of educational web platform requirements. The analysis demonstrates that no single protocol is optimal across all communication patterns present in a modern educational platform.

HTTP/3 achieves the highest aggregate score (27/30) across the evaluated criteria, driven by its elimination of head-of-line blocking and superior performance under mobile network conditions. WebSocket and gRPC score comparably well (24 and 26 respectively) and dominate their specific application domains: real-time bidirectional communication and inter-service binary RPC. HTTP/2 remains a strong general-purpose choice where HTTP/3 infrastructure is not yet available.

The primary practical recommendation is a hybrid protocol strategy mapped to platform components: HTTP/2 or HTTP/3 for content delivery APIs and file transfers, WebSocket for real-time notification channels, and gRPC for internal microservice communication. This strategy, summarized in Table 2, enables each component to achieve near-optimal network performance while maintaining full browser compatibility through appropriate fallback mechanisms.

Future work should address empirical benchmarking of the hybrid strategy under simulated educational workloads, including spiked concurrent access during examination periods, and investigate the operational complexity trade-offs of deploying HTTP/3 infrastructure in university network environments with legacy firewall configurations that may block UDP traffic.

REFERENCES

1. Fielding R. T., Reschke J. Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing. RFC 7230. IETF, 2014. 89p.
2. Belshe M., Peon R., Thomson M. Hypertext Transfer Protocol Version 2 (HTTP/2). RFC 7540. IETF, 2015. 96 p.
3. Biswal A. K., Naskar S. Performance Analysis of HTTP/1.1 vs HTTP/2 Web Protocol. International Journal of Computer Applications. 2018. Vol. 180, No. 32. P. 1–6.
4. Marx R., Wijnants M., Quax P., Lamotte W. HTTP/3 over QUIC is not always Faster than HTTP/2 over TLS. Proceedings of the Web Conference 2020. ACM, 2020. P. 2990–3001.
5. Lubbers P., Albers B., Salim F. Pro HTML5 Programming: Powerful APIs for Richer Internet Application Development. 2nd ed. Apress, 2011. 312 p.
6. Pismenny B., Liss A., Bergman E., Shahar E., Kiryati L., Kogan A. gRPC Protocol Performance Analysis. Proceedings of the ACM SIGCOMM Workshop. ACM, 2021. P. 49–55.
7. Bishop M. HTTP/3. RFC 9114. IETF, 2022. 57 p.
8. Kleppmann M. Designing Data-Intensive Applications. O'Reilly Media, 2017. 611 p.

Лавренюк Арсен Олександрович – студент групи 1ПІ-22б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, e–mail: arsenlavreniuk@gmail.com.

Майданюк Володимир Павлович – кандидат технічних наук, доцент, доцент кафедри програмного забезпечення, Вінницький національний технічний університет, м. Вінниця, e–mail: maidaniuk2000@gmail.com.

Lavreniuk Arsen O. – student of group 1PI-22b, Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: arsenlavreniuk@gmail.com.

Maidaniuk Volodymyr P. – Candidate of Technical Sciences, Associate Professor, Associate Professor of the Software Engineering Department, Vinnytsia National Technical University, Vinnytsia, e–mail: maidaniuk2000@gmail.com.