A DNS-Inspired Federated Resolution Model for ISBN-Based Book Metadata Discovery

H. A. Mooduto[a,*]

[a] *Department of Information Technology, Politeknik Negeri Padang, Kampus Limau Manis, Padang, 25164, Indonesia*
Corresponding author: mooduto@pnp.ac.id

***Abstract*— The International Standard Book Number (ISBN) is allocated through a hierarchy resembling Internet address allocation, from an international agency through national agencies to publishers. Unlike the Domain Name System (DNS), however, the ISBN ecosystem has no resolution protocol, so title-level metadata remain scattered across national databases and aggregators, and Indonesian books are often missing from global lookup services. This study aims to design and evaluate a DNS-inspired federated resolution model that locates book metadata by routing an ISBN to the authority responsible for its prefix. The method parses numbers with the official range rules, selects an authoritative server by longest-prefix matching, follows referrals to publisher delegations, and applies positive and negative time-to-live caching with aggregator fallback. A Python prototype was evaluated in a controlled testbed using synthetic Indonesian datasets of 9,100 titles and an emulated 25 ms delay per server hop. Group detection agreed with a reference library on all 50,705 commonly assigned numbers in a random sample of 100,000; all 1,170,000 single-digit errors and 93.8% of adjacent transpositions were rejected; and all 300 routing cases reached the correct authority. Cache hits were answered in about 0.01 ms, compared with 27.8 ms for one hop and 54.1 ms for a referral, and under Zipf workloads the cache served 70.6–88.0% of 20,000 queries, cutting mean latency by 77.9% at α = 1.0. National agencies could therefore expose a lightweight resolution interface; future work should pilot it with real agency data, response signing, and governance.**

***Keywords*— ISBN; Domain Name System; persistent identifier; federated resolution; caching; bibliographic metadata.**

I. INTRODUCTION

Books are identified worldwide by the International Standard Book Number (ISBN), defined in ISO 2108 and operationalized by the International ISBN Agency through its users' manual [1], [2]. Since 2007 an ISBN consists of 13 digits arranged in five elements: a GS1 prefix (978 or 979), a registration group that usually denotes a country or language area, a registrant element that identifies the publisher, a publication element, and a check digit. Numbers are not issued centrally. The International ISBN Agency in London appoints national agencies and allocates blocks of numbers to them, and each national agency in turn allocates registrant prefixes to publishers in its territory. In Indonesia this role has been performed by the National Library of the Republic of Indonesia (Perpustakaan Nasional, Perpusnas) since 1985, with an online service introduced in 2012 [3]. National Library Regulation No. 5 of 2022 and its technical guidelines subsequently tightened the eligibility criteria for ISBN assignment in response to the rapid consumption of the national number blocks [4], [5]. Recent Indonesian studies on ISBN have focused on service delivery and data validity, for example information systems that streamline ISBN applications in institutional libraries [6] and the verification of check digits with modular arithmetic [7].

The allocation hierarchy of the ISBN is strikingly similar to that of Internet Protocol (IP) addresses, in which the Internet Assigned Numbers Authority delegates address blocks to regional and national registries, which delegate further to network operators and finally to individual hosts. The similarity, however, ends at allocation. Once a number has been assigned, the ISBN system offers no standardized mechanism that tells a client *where* the descriptive metadata of the corresponding book can be retrieved. The International ISBN Agency maintains registers of publisher prefixes rather than of individual titles [2], so title-level metadata are held by national agencies, national bibliographies, publishers, and commercial or community aggregators. These sources differ in coverage, update cycles, and metadata quality, a recurring concern in the metadata quality literature [8], [9]. Harvesting-based aggregation does not by itself guarantee sustained availability either: a recent study of the Indonesia One Search ecosystem found that well-documented metadata do not necessarily translate into sustained interoperability [10]. A practical consequence, familiar to Indonesian authors and librarians, is that a valid ISBN issued by Perpusnas often cannot be found in international lookup services, because the title was never ingested by the aggregator that the user happens to query.

The Internet solved a structurally similar problem decades ago. The Domain Name System (DNS) combines a hierarchical namespace, administrative delegation, authoritative servers, and caching resolvers so that any client can locate the authority for a name without a central

database of all records [11]–[13]. Caching is the cornerstone of DNS performance: trace-driven and measurement studies show that resolver caches absorb most queries and that time-to-live (TTL) choices directly shape latency and server load [14], [15]. More recent work has examined TTL compliance in deployed resolvers [16], the behavior and responsiveness of open and public resolvers [17]–[19], and the analytical modelling of TTL caches [20], [21]. Operational practice continues to refine these mechanisms, for instance by serving stale data during outages and by caching resolution failures to protect authoritative servers [22], [23], with the vocabulary consolidated in a current terminology standard [24].

The persistent identifier (PID) community has adopted similar ideas. The Handle System and the Digital Object Identifier (DOI) system resolve prefix-structured identifiers through a federated infrastructure grounded in the digital object architecture [25]–[27]. Research on PIDs has since emphasized connecting identifiers through their metadata in a PID graph [28], the risks and trust issues of the PID landscape [29], users' understanding of PIDs [30], and the comparison of FAIR Digital Objects with Linked Data as distributed object systems [31]. Decentralized identifiers (DIDs) push resolution further toward distributed ledgers, at a measurable cost in latency [32], [33]. For books specifically, a Uniform Resource Name (URN) namespace for ISBN has been defined [34], but it specifies syntax rather than a resolution service. To the best of our knowledge, no prior study has proposed and quantitatively evaluated a resolution architecture that exploits the native prefix hierarchy of the ISBN itself, without requiring publishers to register a second identifier.

This paper addresses that gap. Its contributions are as follows:

- a structured mapping between Internet address and name delegation and the ISBN allocation hierarchy, identifying which DNS mechanisms transfer to ISBN and which do not;
- a federated resolver architecture that combines range-based ISBN parsing, longest-prefix zone matching, referral-based delegation to publishers, positive and negative caching, and aggregator fallback; and
- an open prototype and an empirical evaluation of parsing correctness, routing correctness, resolution latency, and cache effectiveness under skewed workloads.

The remainder of the paper is organized as follows. Section II describes the analogy, the proposed architecture, and the experimental design. Section III presents and discusses the results, including a comparison with existing approaches and implications for the Indonesian ISBN ecosystem. Section IV concludes the paper and outlines future work.

II. Materials and Method

A. Analogy Between Address Allocation and ISBN Allocation

Fig. 1 places the two allocation hierarchies side by side. In both, a global authority hands out blocks of a fixed-length numbering space, intermediate authorities subdivide those blocks, and the leaves of the hierarchy are individual units: host interfaces in the case of IP, and publications (a specific edition in a specific format) in the case of ISBN. Indonesia, for example, currently administers the registration groups 978-979, 978-602, and 978-623 [35], which were allocated successively as earlier blocks were consumed, a situation that mirrors the depletion of address blocks.



Fig. 1 Allocation hierarchies of IP addresses and ISBNs. Both delegate fixed-length number blocks through intermediate authorities down to individual units.

Table I summarizes the correspondence, together with the points where the analogy breaks down. Three differences shape the design. First, an ISBN identifies a product and carries no location; it plays the role of a name rather than of a routable address, which is why the resolution layer must be built on top of it, as DNS is built on top of names. Second, an ISBN is permanent and is never reassigned, whereas IP addresses are reallocated and domain names expire. Resolution answers for existing titles can therefore be cached for long periods without the agility concerns that constrain DNS TTLs [15]. Third, the ISBN prefix is variable-length within each group, in the same way that classless inter-domain routing uses variable-length prefixes, so a parser cannot locate the boundaries between elements without the official range rules.

TABLE I
Correspondence Between Internet Delegation and ISBN Allocation

Concept	Internet (IP/DNS)	ISBN
Root authority	IANA; DNS root zone	International ISBN Agency
Intermediate authority	RIR/NIR; TLD registry	National ISBN agency
Block holder	Network operator; domain owner	Publisher (registrant)
Leaf unit	Host address; host name	Edition in one format
Prefix length	Variable (CIDR)	Variable (range rules)
Resolution	Recursive query with referrals	None (proposed here)
Reuse	Addresses reassigned; names expire	Never reused
Exhaustion	IPv4 depletion	New groups per country

B. ISBN Structure and Prefix Parsing

Fig. 2 shows how the elements of an ISBN-13 map onto the routing concepts used in this paper. The GS1 prefix and the registration group together form the *zone key*, comparable to a network prefix, which identifies the national or language agency responsible for the number. Appending the registrant element yields the *delegation key*, comparable to a sub-prefix delegated to an organization, which identifies the publisher.

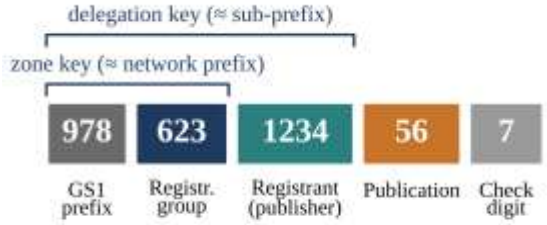


Fig. 2 Elements of an ISBN-13 and their roles in resolution. The zone key selects an agency; the delegation key selects a publisher.

Before any routing takes place, the resolver validates the number. ISBN-10 inputs are converted to ISBN-13 by prefixing 978 and recomputing the check digit. For an ISBN-13 with digits $d_1, \ldots, d_{13}$, the check digit is

$$d_{13} = (10 - (\sum_{i=1}^{12} w_i d_i \ mod\ 10)) \ mod\ 10 \qquad (1)$$

where the weight w_i equals 1 for odd positions and 3 for even positions [1]. Element boundaries are then determined with the range rules that the International ISBN Agency publishes as an XML range message [35]. For a given prefix, the parser reads the next seven digits as an integer, pads it with zeros if necessary, and looks up the interval that contains it; the interval specifies the length of the next element, or zero if the interval is not yet allocated. Applying the rule set for the GS1 prefix yields the registration group, and applying the rule set for that group yields the registrant element. When the full range message is not available, the prototype falls back to a built-in subset that resolves only the registration group, which is sufficient for zone-level routing.

C. Proposed Architecture

Fig. 3 presents the architecture. A client submits an ISBN to a recursive resolver, which contains three components: a parser, a cache, and a zone table. The resolver forwards cache misses to authoritative servers operated by agencies or publishers and, only when no authoritative answer is obtained, to aggregators.

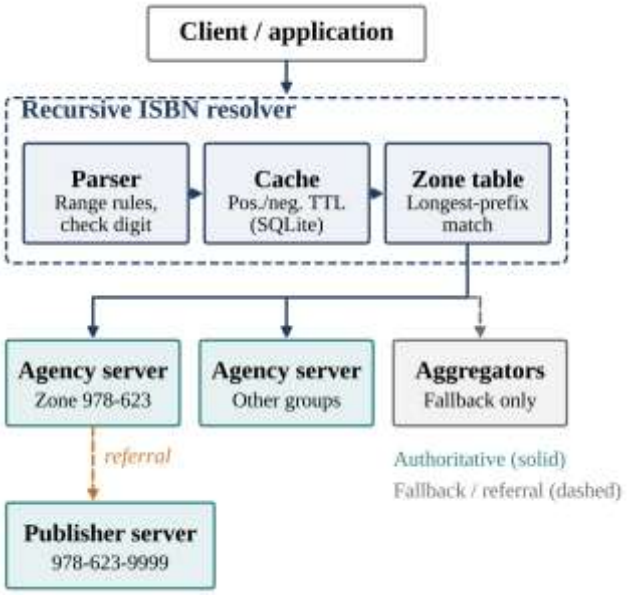


Fig. 3 Architecture of the proposed federated ISBN resolver. Solid arrows denote authoritative queries; dashed arrows denote referrals and fallback queries.

1) Zone table: The zone table is the equivalent of the root zone. Each entry is keyed by a hyphenated prefix, such as 978-623 for an Indonesian group or 978-623-9999 for a delegated publisher, and lists one or more servers, a fallback flag, and an optional address for manual verification. The resolver chooses the entry by longest-prefix matching, preferring the delegation key, then the zone key, then the GS1 prefix, and finally a default entry. This order allows a publisher to operate its own authoritative server while the national agency remains responsible for the rest of the group.

2) Authoritative servers and protocol: Authoritative servers expose a deliberately minimal HTTP interface: a GET request for /isbn/{isbn13} returns either a JSON metadata record, a JSON referral that names another server, or a not-found status. Referrals play the role of DNS delegation records and are followed iteratively up to a fixed limit of three hops, which prevents referral loops. Records are normalized into a common set of fields (title, authors, publisher, year, and source) so that answers from different authorities can be consumed uniformly.

3) Caching: Answers are stored in a local SQLite cache with two TTLs: a long positive TTL (seven days in the prototype), justified by the permanence of ISBN assignments, and a short negative TTL (one hour) for numbers that no authority recognizes, analogous to negative caching in DNS. Transport failures are deliberately not cached, so a temporary outage does not produce a persistent false negative. This choice differs from current DNS guidance, which recommends caching resolution failures briefly to shield servers from retry storms [23]; adopting that behavior, together with serving stale answers during outages [22], is left for future work.

4) Fallback: When the authoritative servers of a zone return no record, the resolver can query aggregator adapters such as the Open Library books interface [36]. Fallback answers are labelled with their source so that clients can distinguish authoritative metadata from aggregated metadata.

D. Resolution Procedure

Fig. 4 illustrates a complete resolution that involves delegation. The resolver parses the ISBN, misses the cache, and selects zone 978-623. The agency server recognizes that the registrant prefix 978-623-9999 has been delegated and returns a referral. The resolver follows it to the publisher server, receives the metadata, normalizes and caches them, and returns the answer together with a trace of every step. The trace makes the resolution path auditable in the same way that DNS diagnostic tools expose the chain of referrals.

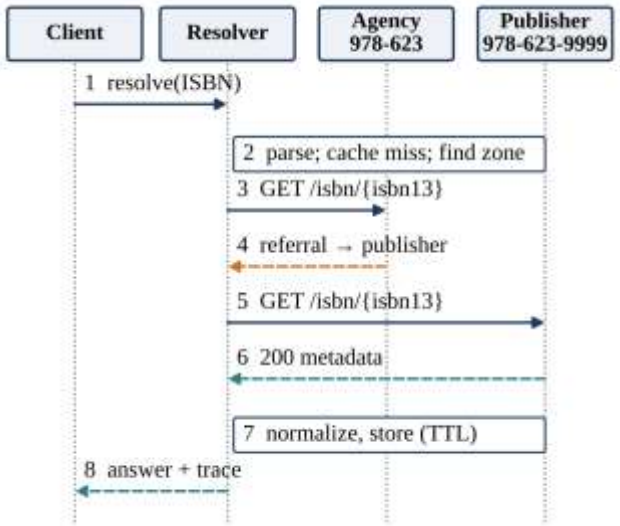


Fig. 4 Resolution sequence for an ISBN whose registrant prefix is delegated from the national agency to a publisher.

E. Prototype Implementation

The prototype consists of three small modules written in Python with no third-party dependencies, which keeps it portable to the modest server environments typical of national libraries and university presses. The first module implements validation, ISBN-10 to ISBN-13 conversion, and range-based parsing, and can load the official range message at start-up. The second module implements the resolver: the zone table is read from a JSON file, each data source is wrapped in an adapter that returns one of four outcomes (found, not found, referral, or error), and the cache is a single SQLite table keyed by the normalized ISBN-13. The resolver can be used from the command line or run as an HTTP service that returns the answer together with its resolution trace. The third module is a reference authoritative server that loads records and delegations from a JSON file and implements the protocol described above, so that an agency or publisher can expose its catalog by exporting it into that format. Because every adapter shares the same four-outcome contract, new authorities or aggregators can be added without modifying the resolution logic.

F. Experimental Design

The prototype was implemented in Python 3.13 using only the standard library and evaluated in a cloud-hosted Linux container. Agency and publisher servers ran on the loopback interface, and each server request was delayed by 25 ms to emulate a wide-area round trip, so that measured latencies separate the cost of network hops from the cost of local processing. Because the testbed had no access to external networks, aggregator fallback was disabled during the measurements. All bibliographic records were synthetic. The agency dataset contained 9,000 titles in groups 978-623 and 978-602, and a delegated publisher held 100 titles under the registrant prefix 978-623-9999. Four experiments were conducted, as summarized in Table II.

TABLE II
SUMMARY OF THE EXPERIMENTS

ID	Purpose and procedure	Metric
E1	Parse 100,000 random valid ISBN-13 values; compare group detection with the isbnlib reference library; inject every single-digit substitution and adjacent transposition into 10,000 valid ISBNs	Agreement rate; detection rate; parse throughput
E2	Resolve 200 agency-held and 100 delegated titles	Correct zone and path
E3	Resolve 300 queries per scenario: agency hit, cache hit, referral, unknown number, negative cache hit	Median and 95th-percentile latency
E4	Issue 20,000 queries drawn from a Zipf distribution over 9,100 titles for $\alpha \in \{0.8, 1.0, 1.2\}$; baseline of 2,000 uncached queries	Hit ratio; mean latency; upstream requests

Book demand is highly skewed, so the workload in E4 follows a Zipf distribution, the standard model for popularity in caching studies [20]. The probability of requesting the title of rank r among N titles is

$$P(r) = \frac{r^{-\alpha}}{\sum_{k=1}^{N} k^{-\alpha}} \tag{2}$$

where α controls the skew; larger values concentrate requests on fewer titles. The cache was empty at the start of each run.

III. RESULTS AND DISCUSSION

A. Parsing Correctness and Error Detection

Of the 100,000 random ISBN-13 values generated in E1, 50,705 fell into registration groups that both the prototype and the reference library could identify, and the two agreed on every one of them. A further 4,997 values were identified by only one of the two implementations, reflecting the difference between the built-in subset of range rules and the complete rules bundled with the reference library; the remaining values fell into unallocated ranges. This confirms that the parsing algorithm is correct and that full coverage depends only on loading the official range message.

All 1,170,000 single-digit substitutions injected into 10,000 valid ISBNs were rejected by the check digit in (1). Of 113,106 adjacent transpositions of unequal digits, 106,093 (93.8%) were rejected. The 7,013 undetected transpositions are exactly those in which the two swapped digits differ by five, an inherent limitation of the alternating 1–3 weights in ISBN-13. The resolver should therefore treat a valid check digit as evidence of plausibility rather than proof of correctness, and the negative cache described below

limits the cost of mistyped numbers that pass validation. Parsing was also fast: 50,000 parse operations completed in 0.148 s, or about 3.0 µs per ISBN, which is negligible compared with any network hop.

B. Routing and Delegation Correctness

All 300 resolutions in E2 reached the intended authority. The 200 agency-held titles were answered directly by the zone server of their group, and all 100 delegated titles were answered by the publisher server after exactly one referral, as recorded in the resolution traces. Longest-prefix matching therefore behaves as intended: a delegation does not disturb resolution of the rest of the group, and adding a new publisher-level authority requires only a delegation entry at the agency and no change to clients.

C. Resolution Latency

Table III and Fig. 5 report the latency of each scenario in E3. With a 25 ms emulated delay, a single authoritative hop had a median latency of 27.8 ms, so local processing, HTTP handling, and JSON encoding added about 2.8 ms per hop. A referral required two hops and took 54.1 ms, roughly twice a direct answer, which quantifies the cost of delegation. Cache hits were answered in a median of 0.010 ms, more than three orders of magnitude faster than any upstream query. Unknown numbers cost one hop on the first query (28.0 ms) but only 0.009 ms on repeated queries thanks to the negative cache, which protects agency servers from repeated lookups of non-existent or mistyped numbers. The narrow gap between median and 95th-percentile values in every scenario indicates stable behavior.

TABLE III
RESOLUTION LATENCY PER SCENARIO (MS)

Scenario	n	Median	95th pct.
Cache hit	300	0.010	0.014
Negative cache hit	300	0.009	0.014
Agency, one hop	300	27.8	29.4
Unknown number, one hop	300	28.0	28.8
Referral, two hops	100	54.1	55.5

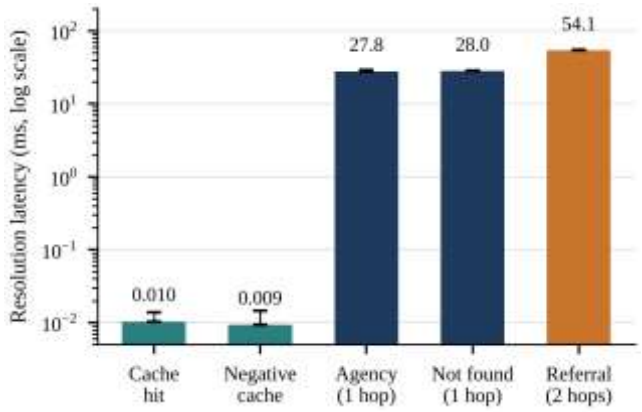


Fig. 5 Median resolution latency per scenario on a logarithmic scale; error bars extend to the 95th percentile. Each upstream hop includes a 25 ms emulated delay.

D. Cache Effectiveness Under Skewed Workloads

Table IV summarizes E4, and Fig. 6 shows how the cumulative hit ratio evolved as the cache warmed. Because the positive TTL exceeded the duration of each run, every miss was a compulsory first request for a title, and the hit ratio equals one minus the ratio of distinct titles to queries. With $\alpha = 1.0$, only 4,094 of 20,000 queries reached an authoritative server, a 79.5% reduction in upstream load; the reduction rose to 88.0% at $\alpha = 1.2$ and was still 70.6% at $\alpha = 0.8$. Mean latency fell from 26.6 ms without a cache to 5.9 ms at $\alpha = 1.0$, a reduction of 77.9%.

TABLE IV
CACHE PERFORMANCE OVER 20,000 ZIPF-DISTRIBUTED QUERIES

Workload	Distinct titles	Hit ratio (%)	Mean latency (ms)
$\alpha = 0.8$	5,890	70.6	8.45
$\alpha = 1.0$	4,094	79.5	5.88
$\alpha = 1.2$	2,393	88.0	3.43
No cache, $\alpha = 1.0$*	—	0.0	26.56

*Baseline measured over 2,000 queries.

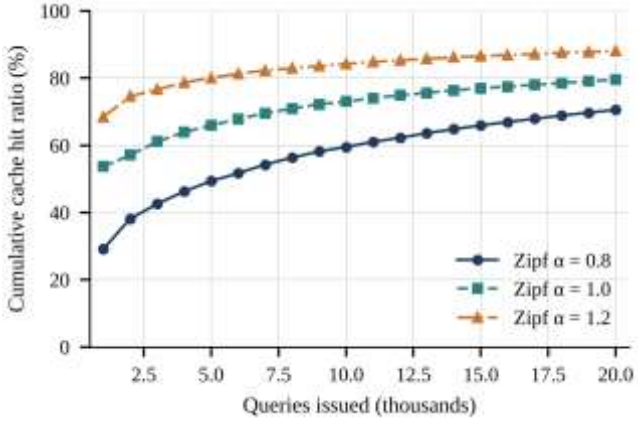


Fig. 6 Cumulative cache hit ratio as the number of queries grows, for three Zipf skew parameters.

The measured mean latency agrees with the simple expectation

$$E[L] = hL_{hit} + (1 - h)L_{miss} \quad (3)$$

where h is the hit ratio and L_{hit} and L_{miss} are the hit and miss latencies. With $h = 0.795$, $L_{hit} \approx 0.01$ ms, and $L_{miss} \approx 28$ ms, (3) predicts 5.7 ms, close to the measured 5.9 ms; the small difference comes from the occasional two-hop referral. These results mirror the central finding of DNS research that skewed popularity makes resolver caches highly effective [14], [15]. Because ISBN assignments never change, the freshness–load trade-off that constrains DNS TTLs [15], [20] concerns only metadata corrections, not identity, which permits considerably longer TTLs than are typical for host names.

E. Security and Operational Considerations

Turning ISBN lookup into a resolution protocol also inherits the operational concerns of DNS. The most important is the integrity of answers: a resolver that accepts unsigned metadata could be fed falsified titles or authors, and the cache would then propagate the error for the whole positive TTL. Signing records at the authoritative server and verifying signatures at the resolver, in the manner of DNSSEC, would address this without changing the resolution path. The second concern is availability. Because

a single agency server is authoritative for an entire group, the design should allow several servers per zone, which the zone table already supports, and resolvers should continue to serve cached answers when an authority is unreachable, following the serve-stale practice of DNS [22]. The third concern is load. The results of E4 show that upstream requests scale with the number of distinct titles rather than with the number of queries, so shared resolvers operated by universities, library consortia, or national discovery portals would protect agency servers far more than isolated clients could. Finally, the negative cache must be bounded carefully, since a newly assigned ISBN may be queried shortly before its record is published; the one-hour negative TTL used here limits that window while still absorbing repeated queries for mistyped numbers [23].

F. Comparison With Existing Approaches

Table V positions the proposed model among existing ways of finding book or object metadata. Centralized aggregators are simple for clients, but their coverage depends on what they have ingested, which is the root of the problem described in Section I. Handle-based systems such as DOI provide mature federated resolution [25], [26], but they require each book to be registered again under a second identifier, which most Indonesian trade and academic publishers do not do. The URN namespace for ISBN [34] standardizes how an ISBN is written as a URN but leaves resolution unspecified. DIDs offer decentralized control but incur ledger-dependent latency [33]. The proposed model reuses the authority structure that already exists for every ISBN and adds only a thin resolution layer on top of it.

TABLE V
COMPARISON WITH EXISTING METADATA LOOKUP APPROACHES

Approach	Resolution model	Data authority	Main limitation
Aggregators	Central lookup	Aggregator	Coverage depends on ingestion
Handle/DOI [25], [26]	Federated, prefix-based	Registrant	Second identifier required
URN:ISBN [34]	Not specified	—	Syntax only
DID [32]	Method-specific, often ledger-based	Controller	Resolution latency
Proposed	DNS-like, over ISBN prefixes	Agency or publisher	Needs agency endpoints

G. Implications for the Indonesian ISBN Ecosystem

For Indonesia, the model suggests a concrete and incremental deployment path. Perpusnas already holds title-level records for the groups 978-602, 978-623, and 978-979 through its online ISBN service [3], [5]. Exposing those records through a read-only endpoint that follows the protocol described above would make every Indonesian ISBN resolvable by any compliant resolver, without depending on foreign aggregators. Publishers and university presses that maintain their own catalogs could be delegated at the registrant level, as demonstrated in E2, while the agency keeps authority over the remainder of the group. The same endpoint could feed national discovery services such as Indonesia One Search, where sustained interoperability has proven difficult to achieve through harvesting alone [10], and it would make metadata quality issues visible at their source rather than after aggregation [8]. Because the regulatory framework already defines who may hold and assign numbers [4], resolution authority follows naturally from allocation authority.

Adoption can also be gradual. A resolver deployed by a single university library can already combine an authoritative zone for Indonesian groups with aggregator fallback for all other groups, so users benefit immediately while other national agencies are added to the zone table one at a time. Over time, the zone table itself could be published by the International ISBN Agency alongside the range message, in the same way that the root zone anchors DNS, giving every resolver a common starting point without centralizing the bibliographic records themselves.

H. Limitations

This study has several limitations. The datasets were synthetic, and latency was emulated on a single host, so the absolute numbers indicate relative costs rather than performance on the public Internet, where delays vary and packet loss occurs. Aggregator fallback was implemented but could not be measured in the testbed. The prototype does not yet authenticate responses; a production deployment would need signed answers, comparable to DNSSEC, to prevent spoofed metadata. Finally, the model assumes that national agencies are willing and able to operate resolution endpoints, a governance question that technical evaluation alone cannot answer.

IV. CONCLUSION

This study showed that the allocation hierarchy of the ISBN can be turned into a resolution hierarchy by borrowing the core mechanisms of DNS: prefix-based zones, authoritative servers, referrals for delegation, and caching resolvers. The prototype parsed and validated ISBNs correctly, routed every test query to the correct agency or publisher, and answered cached queries more than three orders of magnitude faster than upstream ones. Under realistic skewed demand the cache absorbed 70.6–88.0% of queries and reduced mean latency by 77.9% at $\alpha = 1.0$. Because ISBNs are never reassigned, long cache lifetimes are safe, which makes the approach even more cache-friendly than DNS. Future research should pilot the model with real data from Perpusnas and selected publishers over wide-area networks, add signed responses, stale serving, and short-lived caching of failures, map normalized records to established bibliographic formats, examine interoperability with DOI-based identifiers for books, and develop a governance model for a global zone table maintained jointly by the International ISBN Agency and national agencies.

ACKNOWLEDGMENT

The authors thank the Department of Information Technology, Politeknik Negeri Padang, for supporting this research.

REFERENCES

[1] *Information and Documentation — International Standard Book Number (ISBN)*, ISO 2108:2017, International Organization for Standardization, Geneva, Switzerland, 2017.

[2] *ISBN Users' Manual*, 7th ed., International ISBN Agency, London, U.K., 2017.

[3] Sutiarsih, "Perkembangan layanan International Standard Book Number (ISBN) Indonesia [The development of the Indonesian ISBN service]," *Majalah Biola Pustaka*, vol. 1, no. 1, 2022 (in Indonesian).

[4] Perpustakaan Nasional Republik Indonesia, *Peraturan Perpustakaan Nasional Nomor 5 Tahun 2022 tentang Layanan Angka Standar Buku Internasional [National Library Regulation No. 5 of 2022 on International Standard Book Number Services]*, Berita Negara Republik Indonesia 2022 No. 662, Jakarta, Indonesia, 2022.

[5] Perpustakaan Nasional Republik Indonesia, *Petunjuk Teknis Layanan ISBN [Technical Guidelines for ISBN Services]*, Jakarta, Indonesia, 2022.

[6] D. H. Qudsi and M. S. Zulvi, "Pengembangan sistem informasi pengajuan ISBN di Perpustakaan Politeknik Caltex Riau [Development of an ISBN application information system at the Politeknik Caltex Riau library]," *J. Komputer Terapan*, vol. 9, no. 2, pp. 143–152, Dec. 2023, doi: 10.35143/jkt.v9i2.6199.

[7] Sujarwo and M. Fathoni, "Validasi International Standard Book Number dengan aritmetika modulo [Validation of the International Standard Book Number using modular arithmetic]," *Remik: Riset dan E-Jurnal Manajemen Informatika Komputer*, vol. 7, no. 3, Aug. 2023, doi: 10.33395/remik.v7i3.12736.

[8] V. Kumar, Chandrappa, and N. S. Harinarayana, "Exploring dimensions of metadata quality assessment: A scoping review," *J. Librariansh. Inf. Sci.*, vol. 57, no. 3, pp. 661–673, 2025, doi: 10.1177/09610006241239080.

[9] A. Massari, F. Mariani, I. Heibi, S. Peroni, and D. Shotton, "OpenCitations Meta," *Quant. Sci. Stud.*, vol. 5, no. 1, 2024, doi: 10.1162/qss_a_00292.

[10] D. F. Saputra, T. Asmiyanto, and N. Mayesti, "When metadata is not enough: Repository interoperability in a national library open-source discovery platform," in *Proc. Int. Conf. Dublin Core and Metadata Applications (DCMI)*, 2026.

[11] P. Mockapetris, "Domain names — concepts and facilities," Internet Engineering Task Force, RFC 1034, Nov. 1987.

[12] P. Mockapetris, "Domain names — implementation and specification," Internet Engineering Task Force, RFC 1035, Nov. 1987.

[13] P. Mockapetris and K. J. Dunlap, "Development of the Domain Name System," in *Proc. ACM SIGCOMM '88*, Stanford, CA, USA, 1988, pp. 123–133.

[14] J. Jung, E. Sit, H. Balakrishnan, and R. Morris, "DNS performance and the effectiveness of caching," *IEEE/ACM Trans. Netw.*, vol. 10, no. 5, pp. 589–603, Oct. 2002.

[15] G. C. M. Moura, J. Heidemann, R. de O. Schmidt, and W. Hardaker, "Cache me if you can: Effects of DNS time-to-live," in *Proc. Internet Meas. Conf. (IMC '19)*, Amsterdam, Netherlands, 2019, pp. 101–115.

[16] P. Bhowmick, "Measuring and understanding TTL violations in DNS resolvers," M.S. thesis, Dept. Comput. Sci., Virginia Tech, Blacksburg, VA, USA, 2023.

[17] J. Park, R. Jang, M. Mohaisen, and D. Mohaisen, "A large-scale behavioral analysis of the open DNS resolvers on the Internet," *IEEE/ACM Trans. Netw.*, vol. 30, no. 1, Feb. 2022.

[18] R. Sharma and N. Feamster, "Measuring the availability and response times of public encrypted DNS resolvers," arXiv:2208.04999, 2022.

[19] N. Kernan, J. Li, R. Al-Dalky, and M. Rabinovich, "Public DNS resolvers meet content delivery networks: A performance assessment of the interplay," arXiv:2502.05763, 2025.

[20] G. Hasslinger, M. Okhovatzadeh, K. Ntougias, F. Hasslinger, and O. Hohlfeld, "An overview of analysis methods and evaluation results for caching strategies," *Comput. Netw.*, vol. 228, 2023, Art. no. 109583, doi: 10.1016/j.comnet.2023.109583.

[21] K. Elsayed and A. Rizk, "On the impact of network delays on time-to-live caching," arXiv:2201.11577, 2022.

[22] D. Lawrence, W. Kumari, and P. Sood, "Serving stale data to improve DNS resiliency," Internet Engineering Task Force, RFC 8767, Mar. 2020.

[23] D. Wessels, W. Carroll, and M. Thomas, "Negative caching of DNS resolution failures," Internet Engineering Task Force, RFC 9520, Dec. 2023.

[24] P. Hoffman and K. Fujiwara, "DNS terminology," Internet Engineering Task Force, RFC 9499, Mar. 2024.

[25] S. Sun, L. Lannom, and B. Boesch, "Handle System overview," Internet Engineering Task Force, RFC 3650, Nov. 2003.

[26] *Information and Documentation — Digital Object Identifier System*, ISO 26324:2022, International Organization for Standardization, Geneva, Switzerland, 2022.

[27] R. Kahn and R. Wilensky, "A framework for distributed digital object services," *Int. J. Digit. Libr.*, vol. 6, no. 2, pp. 115–123, 2006.

[28] H. Cousijn *et al.*, "Connected research: The potential of the PID graph," *Patterns*, vol. 2, no. 1, Jan. 2021, Art. no. 100180, doi: 10.1016/j.patter.2020.100180.

[29] P. de Castro, U. Herb, L. Rothfritz, and J. Schöpfel, "Some reflections on the current PID landscape – with an emphasis on risks and trust issues," *Procedia Comput. Sci.*, vol. 211, pp. 28–35, 2022, doi: 10.1016/j.procs.2022.10.173.

[30] G. Macgregor, B. S. Lancho-Barrantes, and D. R. Pennington, "Measuring the concept of PID literacy: User perceptions and understanding of persistent identifiers in support of open scholarly infrastructure," arXiv:2211.07367, 2022.

[31] S. Soiland-Reyes, C. Goble, and P. Groth, "Evaluating FAIR Digital Object and Linked Data as distributed object systems," *PeerJ Comput. Sci.*, vol. 10, 2024, Art. no. e1781, doi: 10.7717/peerj-cs.1781.

[32] M. Sporny, D. Longley, M. Sabadello, D. Reed, O. Steele, and C. Allen, "Decentralized Identifiers (DIDs) v1.0," W3C Recommendation, Jul. 2022.

[33] M. Alizadeh, K. Andersson, and O. Schelén, "Comparative analysis of decentralized identity approaches," *IEEE Access*, vol. 10, pp. 92273–92283, 2022, doi: 10.1109/ACCESS.2022.3202553.

[34] J. Hakala and H. Walravens, "Using International Standard Book Numbers as Uniform Resource Names," Internet Engineering Task Force, RFC 3187, Oct. 2001.

[35] International ISBN Agency, "ISBN ranges (RangeMessage.xml)." [Online]. Available: https://www.isbn-international.org/range_file_generation, Accessed on: Nov. 24, 2026.

[36] Internet Archive, "Open Library Books API." [Online]. Available: https://openlibrary.org/dev/docs/api/books, Accessed on: Nov. 24, 2026.