ASINT: Learning AS-to-Organization Mapping from Internet Metadata

Yongzhe Xu
Virginia Tech
USA

Weitong Li
Virginia Tech
USA

Eeshan Umrani
Virginia Tech
USA

Taejoong Chung
Virginia Tech
USA

ABSTRACT

Accurately mapping Autonomous Systems (ASNs) to their owning or operating organizations underpins Internet measurement research and security applications. Yet existing approaches commonly rely solely on WHOIS or PeeringDB, missing important relationships (e.g., cross-regional aliases, parent–child ownership) and failing to unify organizations scattered across different RIR identifiers. We introduce ASINT, an end-to-end pipeline that fuses bulk registry data with unstructured Web sources, then employs retrieval-augmented generation (RAG) to guide large language model (LLM) inference. Through a multi-stage procedure, ASINT merges ASNs into "organization families," capturing nuanced ties beyond the scope of simpler heuristics.

ASINT maps 111,470 ASNs to 81,233 organization families; compared to both AS2ORG+ and AS-Sibling, ASINT identifies more cross-regional groupings (e.g., operator aliases, rebrands) that other datasets overlook. Moreover, our refined mappings enhance multiple security and measurement tasks: ASINT exposes 27.5% more intra-organizational RPKI misconfigurations, cuts false-positive hijack alarms by 9.4%, and lowers erroneous IP leasing inferences by 5.9%.

Finally, ASINT supports periodic updates and cost-sensitive LLM selection, demonstrating that broader Web evidence can provide a more accurate, evolving view of the Internet's organizational structure.

1 INTRODUCTION

The Internet's global reach depends on tens of thousands of autonomous systems (ASes) owned and operated by diverse entities—from individual enterprises to large service providers spanning multiple continents. Identifying which real-world organization stands behind a given AS number (ASN) is essential for understanding Internet topology, analyzing resource ownership, and diagnosing security incidents. Unfortunately, mapping ASNs to their controlling organizations is far from trivial. Corporate ownership is often masked by fragmented WHOIS records, outdated registry data, and rebranding or acquisitions that unfold faster than administrative databases can be updated.

Accurate AS-to-Organization mappings underpin critical research and operational use cases. For example, RPKI measurements rely on knowing the legitimate origin AS of an IP prefix to detect misconfigurations [7], while BGP hijack detection systems must filter out internal re-announcements among ASes owned by the same entity [1, 11]. Understanding whether two ASNs belong to the same organization can also guide Internet topology analyses, ensuring correct "customer cone" measurements and more accurate ISP rankings [21]. In short, the integrity and reliability of many network and security studies hinge on precisely associating ASNs with their true parent organizations.

Existing approaches—such as AS2ORG [8], AS2ORG+ [2], or AS-Sibling [9]—attempt to solve this problem by consolidating WHOIS records and sometimes incorporating PeeringDB [26]. While they have proven valuable in practice, they exhibit notable shortcomings:

- *Stale or incomplete WHOIS*: Large organizations manage multiple identical organization records across RIRs. Mergers and acquisitions are not always reflected in registry updates.
- *Limited free-text usage*: Simple string matching or notes fields in PeeringDB may fail to recognize acquisitions listed on corporate websites, news, or investor reports.
- *Lack of relationship granularity*: Most mappings only mark "same organization," neglecting whether an AS is a subsidiary or a rebranded brand of a larger parent.

In this work, we push the boundaries of AS-to-Org mapping by leveraging recent advances in natural language processing (NLP)—particularly retrieval-augmented generation (RAG) [23], named entity recognition (NER) [17], and a multi-step post-processing pipeline to unify aliases, deduce parent–child links, and scale to tens of thousands of ASes. Our system collects structured data from WHOIS and PeeringDB, but also supplements these with unstructured text from company websites, Wikipedia, and news sources, which a large

language model (LLM) interprets for hidden connections. We then apply a post-filtering and clustering mechanism to correct or consolidate partial overlaps, ensuring that rebranded names, parent subsidiaries, and acquisitions are captured in our final "organization families".

Using our pipeline, we map 111,470 ASNs to 88,101 organizations, ultimately forming 81,233 organization families. Along the way, we discover thousands of additional relationships missed by existing datasets, significantly increasing coverage of cross-RIR ownership, rebrandings, and up-to-date acquisition histories. Our main contributions are:

(1) *Novel use of modern NLP*: We show how RAG, NER, and an LLM-based inference pipeline can transform messy text data into precise AS-to-Org links.
(2) *In-depth alias and hierarchy resolution*: Beyond labeling "same organization," we incorporate parent–child relationships reflecting acquisitions or subsidiaries, offering a more complete corporate view.
(3) *Extensive validation and real-world impact*: We demonstrate how our mapping helps reduce false positives in hijack detection, detect more RPKI misconfigurations, and improve organization-level ISP ranking.

Through ASINT, we aim to provide the networking community with an up-to-date and robust mapping of ASNs to their true organizational structures, bridging a key gap that underlies many measurement, security, and operational analyses. To foster community-wide progress and facilitate new lines of research, we will open-source all tools used in this work along with regularly updated AS-to-Organization mapping datasets via a public repository.

2 BACKGROUND AND RELATED WORK

2.1 Mapping ASNs to Organizations

Mapping ASNs to their parent organizations underlies numerous studies of Internet structure, routing security, and resource ownership [7, 12, 21]. However, there is no comprehensive global registry mapping each ASN to a single entity. Frequent corporate changes (e.g., mergers, acquisitions, rebranding) and ASN records fragmented across multiple Regional Internet Registries (RIRs) further complicate the task. Hence, researchers typically synthesize data from WHOIS and PeeringDB [26] to infer ASN–organization links.

Whois-Based Methods. WHOIS remains a foundational source for ASN-to-organization mappings. RIR databases contain `OrgID` fields, addresses, and contact information, which CAIDA's AS2Org [10], which we call CA2O, leverages to group ASNs under common owners (e.g., shared `OrgID` or email domains). Despite its broad coverage of public ASNs, WHOIS data has well-known limitations:

- *Data Quality and Consistency:* WHOIS records are updated manually by thousands of organizations, often lagging behind corporate changes or containing inaccuracies [8]; because of this, Chen et al. observed that WHOIS-only datasets often overmerge or split organization identities erroneously [9].
- *No Unified Identifiers:* Different RIRs (or even within the same RIR) employ distinct `OrgID` formats, complicating cross-region merges for the same operator (e.g., Orange's AS5511 in RIPE as `ORG-FT2-RIPE` vs. AS36912 in AFRINIC as `ORG-OCS1-AFRINIC`).
- *Variable Record Structures:* RIR WHOIS structures differ (e.g., ARIN has an `OrgID` field, LACNIC may not), fragmenting one organization into multiple entries.

Earlier projects like Cai et al. [8] aggregated WHOIS data with additional operator input, but inaccuracies and incomplete participation proved challenging. WHOIS-based mappings therefore remain a strong starting point but benefit from complementary data sources to capture missing or outdated ownership changes.

Leveraging PeeringDB and Heuristic Parsing. PeeringDB complements WHOIS by providing operator-maintained *network name*, *organization ID*, and an optional *website* field. Arturi et al. [2] (AS2ORG+) showed that combining these `OrgID` with CAIDA's WHOIS-based mappings can unify more ASNs under major organizations, particularly those spanning multiple RIRs.

Despite this benefit, PeeringDB exhibits partial coverage: among the 111,470 total ASNs in WHOIS, only 30,058 (27.0%) appear in PeeringDB. Heuristic parsing in fields like *notes* is also brittle—only 3,897 (3.5%) of networks provide any such notes, which can contain ambiguous references (e.g., abuse contacts or unrelated upstream peers). Consequently, text extraction via regex or domain matching can yield false positives when overlapping email substrings spuriously link unrelated organizations.

Recently, Selmo et al. [28] introduced Borges, which applies a large-language model (LLM) to PeeringDB data. The system (i) prompts the LLM to pull additional sibling ASNs from the free-text `aka`/`notes` fields, and (ii) merges networks whose PeeringDB URLs ultimately resolve to the same domain or share a favicon/brand signature.

This LLM-assisted strategy reveals roughly 7.0% more sibling links than AS2ORG+. Because every signal originates in PeeringDB, however, Borges's reach is still inherently limited to the 27.0% of global ASNs that register there—and only when they supply a usable website—leaving the rest unmapped.

2.2 Recent Advances with NLP and Multi-Source Inference

Most current systems rely on WHOIS and PeeringDB, using heuristic approaches (e.g., regular expressions on "notes" fields) [2, 9]. While these can match repeated textual patterns, the unstructured, variable nature of registry data still leads to errors. Crucially, corporate changes (mergers, rebrands) are often documented outside WHOIS or PeeringDB.

A direct approach might ask: *"Which organization operates AS X?"* to a large language model (LLM). However, up-to-date coverage is not guaranteed, and domain-specific or long-tail details may be missing [19]. Moreover, simply dumping all relevant web content into an LLM risks computational blowouts and hallucinations [14, 33].

Retrieval-Augmented Generation (RAG). To address these limitations, *Retrieval-Augmented Generation (RAG)* separates knowledge retrieval from LLM inference [14, 23, 29]. Under this framework, an *information retrieval* step first narrows down relevant snippets (e.g., corporate archives), and only then does the LLM generate output based on these targeted documents. A core building block of RAG is often a *vector database* that stores dense text embeddings to support high-precision semantic searches. By filtering large corpora into smaller, context-specific sets, RAG reduces hallucinations and adapts to dynamic updates. This approach has shown effectiveness in diverse domains, including question answering [32], code analysis [15, 24], and more, making it well-suited for scenarios like ours where relevant knowledge is scattered across numerous, frequently evolving sources.

Few-Shot Prompt Engineering. LLMs also benefit from well-structured prompts. Ambiguous queries (e.g., "Any relation between Org-A and Org-B?") can yield inconsistent results [18], whereas explicit instructions (e.g., "*Is Org-A a subsidiary of Org-B?*") improve accuracy [6, 30]. Few-shot examples further guide the model to apply consistent and accurate reasoning [34].

Building on these insights, ASINT combines WHOIS, PeeringDB, and broader Web sources (e.g., press releases, industry reports) within a RAG pipeline. By filtering relevant snippets before prompting the LLM, ASINT more accurately identifies shared ownership, especially where registry data are incomplete or out-of-date.

3 MOTIVATION AND GOALS

3.1 Goal

A precise mapping of ASNs to the organizations that own or operate them is essential for accurate Internet measurement, routing security (e.g., detecting hijacks), and resource allocation (e.g., RPKI). While WHOIS and PeeringDB supply partial ownership information, they often fail to track cross-regional operators or keep up with corporate changes (e.g., mergers, acquisitions, rebranding). These gaps can significantly impact downstream applications that require a current view of organizational structures.

For example, AS6128 and AS54004 remain officially registered under different names (Cablevision Systems Corp. and Lightpath), yet they have both been operated by Optimum [3, 4] for years. Such corporate histories frequently go unrecorded in PeeringDB and lag in WHOIS, obscuring a unified picture of ownership.

Against this backdrop, our work addresses three primary weaknesses in current ASN-to-organization mapping methods:

(1) *Restricted coverage*: WHOIS and PeeringDB often lag behind real-world corporate transitions and lack comprehensive cross-regional updates.
(2) *Oversimplified relationships*: Existing datasets typically track only "same organization" links and seldom reflect acquisitions, subsidiaries, or internal brand aliases.
(3) *Static or simplistic parsing*: Rule-based heuristics (e.g., regex matching) can misinterpret unstructured fields and fail to adapt to new corporate naming conventions or rebrandings.

3.2 Beyond WHOIS and PeeringDB

While WHOIS and PeeringDB remain fundamental for baseline mappings, they rarely capture the full range of aliases or historical rebrandings. Our approach supplements these databases with additional *unstructured* online sources:

- *Company Websites*: Official portals and investor pages often clarify rebranding, acquisitions, or internal subsidiaries.
- *Wikipedia and Other Wikis*: Crowd-sourced articles frequently catalog mergers and executive transitions that appear long before registry updates.
- *News Articles and Blogs*: Media outlets, industry newsletters, and specialized blogs may report real-time rebranding or consolidation events.

For example, a corporate homepage might confirm that two separately branded networks belong to the same organization; wiki articles could trace stepwise acquisitions under one umbrella. Incorporating these sources reveals many relationships that WHOIS or PeeringDB alone overlook.

3.3 Challenges

Although these external data streams can fill critical gaps, effectively integrating them poses several challenges:

- *Locating Relevant Information at Scale*: Sifting through the entire Web to find evidence of ownership changes is

infeasible. A selective retrieval or filtering step is essential to identify only the text segments likely to clarify actual organizational ties.

- *Understanding Unstructured Content*: Most websites and news reports discuss organizations by name, not ASN. Simple string matching often fails for rebranded or transliterated names. Large language models (LLMs) can identify statements like "Company A is now part of Company B," but they must be carefully guided to avoid confusion with unrelated mentions.
- *Hallucination and Outdated Knowledge*: General-purpose LLMs can generate erroneous or fabricated relationships ("hallucinations") [16], and may not reflect recent corporate events if trained on older data.
- *Avoiding Over- or Under-Inference*: Presenting partial or contradictory data can induce spurious links or missed merges. Validation is necessary to reconcile inconsistencies and ensure that inferred parent–child connections represent real corporate affiliations.

Our Approach. To overcome these limitations and provide a comprehensive, dynamic mapping of ASNs, we propose an approach that integrates *unstructured* Web sources with LLM-based inference. The following section details ASINT, including how it aggregates disparate evidence, handles large-scale updates, and remains aligned with the actual corporate landscape of the global Internet.

4 ASINT: DESIGN

In this section, we describe the design of ASINT, our end-to-end system for creating an accurate AS-to-organization mapping. As illustrated in Figure 1, the pipeline comprises four major stages:

(1) *Data Collection* (§4.1): We assemble ASN and organization data from WHOIS, PeeringDB, and external web sources, forming the initial basis for identifying potential ownership relations.
(2) *Data Processing and Storage* (§4.2): We clean, normalize, and enrich the collected data. We create *organization records*, apply Named Entity Recognition (NER), and store the filtered text chunks in a local knowledge base.
(3) *LLM-Based Inference* (§4.3): Based on the knowledge base, we employ a Retrieval-Augmented Generation (RAG) pipeline, prompting an LLM to classify pairs of organizations as either aliases, parent–child, or unrelated.
(4) *Post-Filtering and Clustering* (§4.4): We refine the raw LLM outputs and unify them into a consistent global hierarchy. This step merges organizations with the same real-world identity and identifies parent–child ownership links, ultimately creating coherent *organization families*.

4.1 Data Collection

ASINT begins by aggregating information from multiple registries and online sources:

- *WHOIS Databases*: We gather bulk WHOIS dumps from all five RIRs, optionally enriched by other National Internet Registries (NIRs) (e.g., JPIRR). These databases form the primary listing of ASNs and their registered owners.
- *CAIDA's AS2Org (CA2O)*: We incorporate CAIDA's dataset to address ambiguous or partial WHOIS records, especially in regions lacking a clear `OrgID`.
- *PeeringDB*: We add metadata such as website URLs, alternative organization names, or peering "notes" that might indicate mergers or brand variations.
- *Web Crawls of Company Websites*: For organizations not adequately represented in WHOIS or PeeringDB, we crawl official websites and search-engine results. These often reveal rebrandings, subsidiary details, or historical acquisitions that registries miss.

Labeling ASNs to Organizations. We start by parsing the bulk WHOIS data to compile a preliminary list of ASNs. In the rare event that the same ASN appears in multiple RIRs, we rely on *as-block* records to select the authoritative source. If an ASN references an explicit organization identifier (e.g., `OrgID` in ARIN), we immediately create a corresponding *organization record* and link the two.

For ASNs lacking clear ownership (common in LACNIC or APNIC bulk files), we consult CA2O. If that remains inconclusive, we turn to PeeringDB. If all else fails, we assign the ASN's descriptive label (e.g., the WHOIS `descr` field) as a temporary organization name.

Enriching Organization Profiles. Having associated each ASN with an initial organization record, we then collect more details (e.g., rebranding history, known aliases):

- *PeeringDB*: For each record, we note any listed website URL, which we can then crawl for official statements, news releases, or brand announcements.
- *Targeted Web Searches*: We generate search queries (e.g., "acquired by," "parent company," "Wikipedia") for each organization name, storing only the top few URL results to avoid duplication.

Finally, we crawl these URLs and associate any retrieved HTML with the relevant organization record. These pages often contain the unstructured data needed in subsequent stages to confirm organizational relationships.

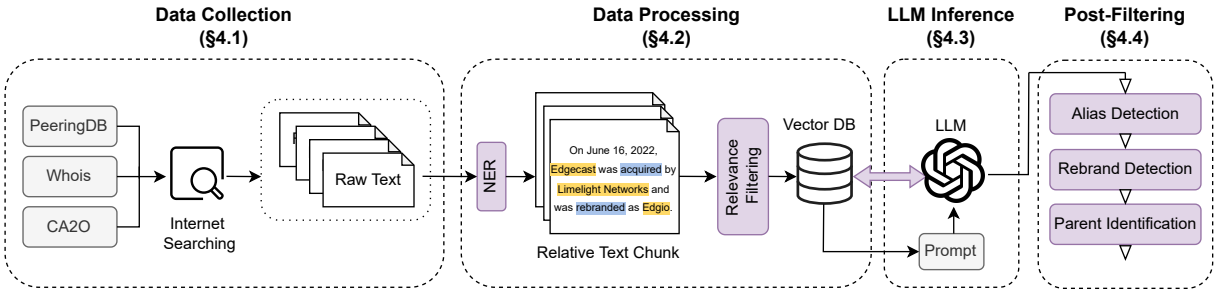

Figure 1: An overview of the ASINT framework, from data collection to final clustering of organization families.

4.2 Data Processing and Storage

The collected raw text typically includes large amounts of irrelevant or noisy material. We therefore apply a multi-step filtering procedure to isolate only text relevant to each organization record. The result is a local *knowledge base*, stored in a vector database, that indexes pertinent snippets by organization.

(1) *Global Name List.* We compile a master list of *canonical names* from all organization records. This list serves as the reference for our NER step.
(2) *Text Splitting.* For each organization record, we divide the crawled HTML into smaller, semantically coherent chunks, introducing minimal overlap so that no key phrase is lost near chunk boundaries.
(3) *Named Entity Recognition (NER).* We run NER on each chunk to extract potential *candidate organization names*. Any extracted name not matching our global list is discarded to avoid random or irrelevant entities.
(4) *Relevance Filtering.* We retain only chunks referencing both the *target organization record* (i.e., the record whose website or queries we crawled) and at least one candidate organization from NER. Chunks lacking this co-occurrence seldom indicate a genuine relationship.
(5) *Knowledge Base Storage.* Finally, we store each retained chunk (plus recognized entity names) in a local vector database keyed by the target organization's record ID. We also compile a deduplicated list of candidate organizations discovered in those chunks.

After this stage, each organization record is linked to a relatively small, focused set of text segments that specifically mention potential affiliations.

4.3 LLM-Based Relationship Inference

We now determine whether these organizations (co-mentioned in the knowledge base) are aliases of the same real-world entity or related via parent–child ownership. A brute-force approach, comparing all pairs, is infeasible; likewise, feeding *all* crawled text into an LLM at once would be computationally overwhelming. Instead, ASINT focuses solely on the *candidate organizations* associated with each *target organization record* to narrow the search space, and adopts a RAG pipeline that restricts the LLM to only those text segments relevant to each specific organization pair.

(1) *Candidate Retrieval.* For each *target organization record*, we gather the set of *candidate organizations* that appeared with it in relevant text (§4.2). We ignore organizations never co-mentioned in any chunk.
(2) *Context Retrieval.* For each candidate, we query the vector database to retrieve only chunks referencing *both* the target organization and that candidate organization. This ensures the LLM sees only highly pertinent snippets.
(3) *LLM Classification.* We send these snippets to a large language model, along with a structured prompt (Appendix C), asking it to classify the relationship between the two organizations as:
   - `alias`: Different names for the same operator.
   - `parent/child`: One entity owns or controls the other.
   - `no-relation`: No ownership or alias link.
(4) *Assignment.* If the LLM output is `alias` or `parent/child`, we update the target record's lists (e.g., `alias` or `parents`). Otherwise, we disregard the candidate.

By confining the model to small, relevant snippets, we reduce computational cost, minimize hallucinations, and maintain a clearer chain of evidence for each relationship.

4.4 Post-Filtering and Clustering

Up to this point, each organization record has been processed individually, generating localized relationships (aliases, parent–child). However, multiple records can represent the same real-world operator, or partial data can falsely suggest a link between organizations with generic or misleading names. We thus run a final *post-filtering and clustering* step, unifying duplicates, handling rebranding scenarios, and forming a cohesive hierarchy.

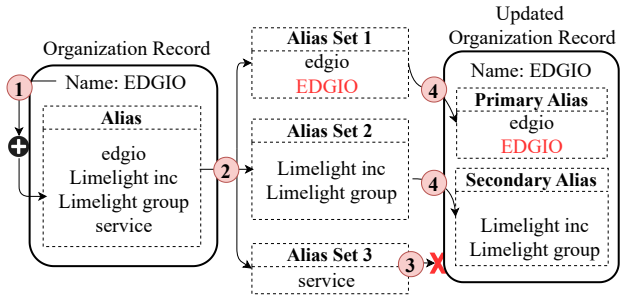

Figure 2: Stage 1: Creating primary and secondary aliases.

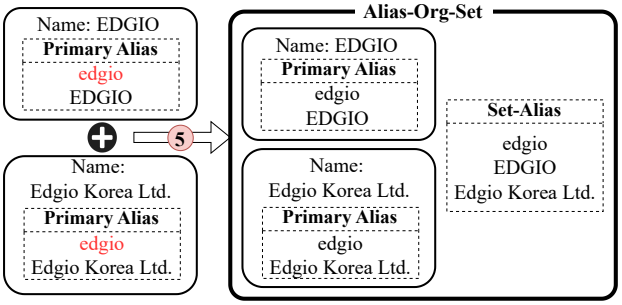

Figure 3: Stage 1: Merging into `alias-org-sets` via shared primary aliases; secondary aliases are omitted for brevity.

4.4.1 Stage 1: Alias Detection via Shared Text. We first cluster organization records that share overlapping *validated* aliases (e.g., "`Limelight inc.`" vs. "`Limelight Company`"). Figure 2 and Figure 3 illustrate this process:

(1) *Augment Alias List.* Each record's *canonical name* is added to its alias list (`alias`).
(2) *Local Clustering.* We compare aliases via Jaccard Similarity [22], forming `alias-sets`. Any single-string set (e.g., "Group") is discarded as likely noise.
(3) *Primary vs. Secondary Aliases.* Clusters containing the canonical name become `primary-alias` sets; all others are labeled `secondary-alias`.
(4) *Graph-Based Union.* We construct an undirected graph where nodes represent organization records and edges appear if two records share at least one `primary-alias`. A connected-component search merges them into `alias-org-sets`, each representing what might be a single real-world operator.

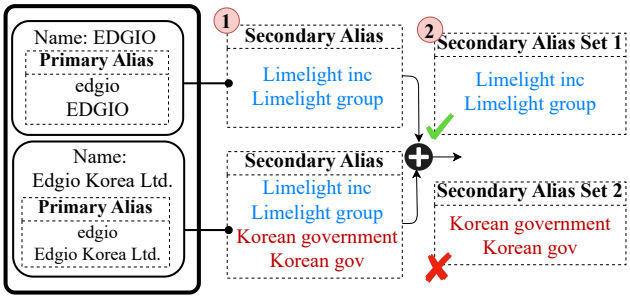

Figure 4: Stage 2: Applying majority voting on secondary aliases.

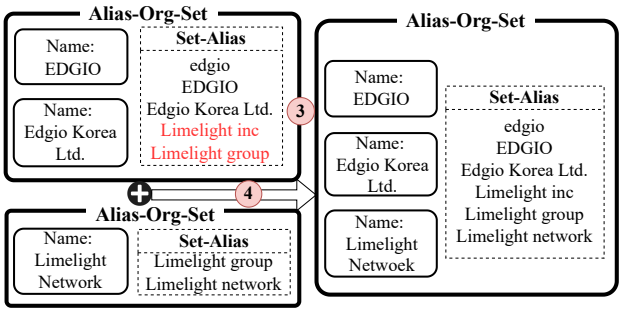

Figure 5: Stage 2: Merging `alias-org-sets` after rebrand detection.

4.4.2 Stage 2: Rebranding and Renaming. Minor string overlap can miss major rebrands (e.g., "`Limelight Networks`" rebranded to "`Edgio`"). We address these via the `secondary-alias` (Figure 4, Figure 5):

(1) *Collect Secondary Aliases.* For each `alias-org-set`, gather all `secondary-alias` from the member records and cluster them with the same Jaccard method.
(2) *Majority Voting.* We keep only those clusters used by at least 50% of the `alias-org-set`'s records, mitigating one-off errors or hallucinations.
(3) *Alias Promotion.* Surviving clusters are upgraded and added to `set-alias`, indicating recognized name changes.
(4) *Set-Level Merge.* If two `alias-org-sets` now share at least one element in newly updated `set-alias`, they merge into a single set. This unifies records that were previously split due to a complete rename.

4.4.3 Stage 3: Parent–Child Relationship Identification. Finally, we determine ownership links *between* these merged sets (i.e., `alias-org-sets`). This step constructs a directed acyclic graph (DAG) capturing child–parent relationships:

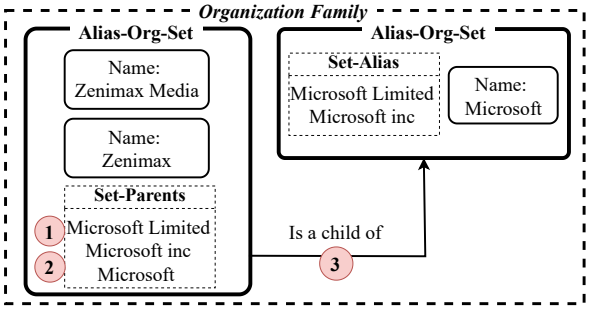

Figure 6: Stage 3: Forming a DAG of parent–child relationships.

(1) *Cluster Parent Candidates.* For each `alias-org-set`, we gather all `parents` references from the LLM inference phase and cluster them again, employing the same majority-vote filter to eliminate outliers.
(2) *Assign* `set-parents`. Surviving references become the set's consolidated parent list, `set-parents`, representing recognized ownership.
(3) *Create Directed Edges.* We add an edge from set C (child) to set P (parent) if C's `set-parents` overlaps with P's `set-alias`. Connected components in this DAG reveal hierarchical structures among organizations.

Final Output. By merging organizations with matching or rebranded names and identifying parent–child ownership, ASINT yields a reproducible, hierarchical map of *organization families*. Each node represents a real-world operator, and each directed edge denotes an ownership link.

5 RESULTS AND ANALYSIS

In this section, we present the AS-to-Org mapping results obtained from ASINT. The outcome of ASINT is a set of organization records, each of which we refer to as an *organization family*—a group of AS Numbers controlled by a single entity, including both different aliases of the same organization and any acquired child organizations.

5.1 Overview of the Data Pipeline and ASINT Coverage

In this subsection, we provide an overview of our data collection and processing pipeline, culminating in the organization familes produced by ASINT.

Data Collection. We start by gathering AS Numbers and possible organization records from a range of sources. As shown in Table 1, we first parse WHOIS data from multiple RIRs and NIRs, obtaining a total of 111,470 ASNs and 3,503,756 organization records. Not all organization records

	WHOIS	CA2O	PeeringDB
Num. of ASes	111,470	111,641	30,058
Num. of Orgs	3,503,756	95,815	31,615
Matched Orgs (with ≥ 1 AS)	65,842	95,815	28,389

Table 1: Datasets from WHOIS, CAIDA, and PeeringDB. "Matched orgs" denotes the count of organizations with at least one associated AS.

in WHOIS data are associated with an AS Number; in fact, only 65,842 of these records can be matched to at least one ASN. For example, ARIN alone maintains 3,356,545 organization records, many of which relate to IP address allocations (*NetHandle* or *V6NetHandle*) rather than AS Numbers. In addition, due to incomplete bulk access in certain NIRs regions (e.g., KRNIC), we incorporate CAIDA's AS2Org dataset to fill these gaps, acquiring additional ASes and organization names. In total, this yields 111,470 AS Numbers and 88,101 potential organization names for subsequent processing.

We then integrate PeeringDB records, which complement WHOIS data with information such as official websites, alternative names, and descriptive notes. PeeringDB covers 30,058 ASes (27.0% of our total) linked to 28,389 organizations, indicating that relying *solely* on PeeringDB would restrict global AS coverage. Nonetheless, it remains a valuable source of supplemental metadata (e.g., official websites and *aka* fields), which we leverage in subsequent web-crawling steps. For every AS in PeeringDB that offers a website URL, we automatically retrieve the corresponding web content.

Web Crawling and Text Processing. To further enrich our dataset, we perform targeted searches for all 88,101 organizations, issuing multiple query patterns (e.g., "acquired by", "parent company", etc.) and collecting the top-five results for each. We then crawl a total of 873,949 URLs, from which we extract 20,478,907 text chunks. With a pre-trained BERT based NER module [20], we filter these chunks for references to organizations of interest, reducing the corpus to 386,278 potentially relevant chunks (roughly 1.9% of the original text). We then build a local vector database using Milvus [31] from these filtered chunks, forming a knowledge base that supports subsequent retrieval-augmented generation (RAG) queries.

Inference and Final Clustering. Next, we use an LLM-based inference approach (§4), specifically ChatGPT 4o-mini, to identify organization aliases and parent–child relationships. This process involves 89,515 distinct LLM queries, after which we apply a post-filtering and clustering step to merge duplicate entities and establish hierarchical links (§5.4.3). Ultimately, we arrive at 81,233 final *organization families*, each comprising one or more AS Numbers under common

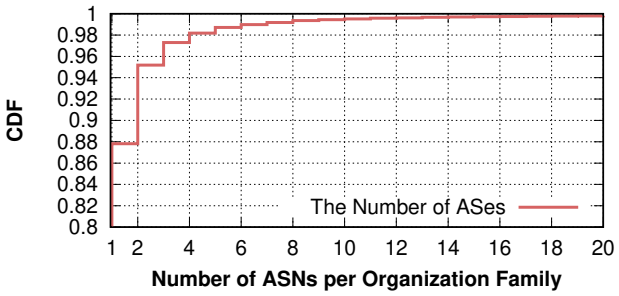

Figure 7: CDF of the number of ASes per organization family. Note that x-axis extends to 973, which is "DoD Network Information Center" that manages 973 different ASes in ARIN.

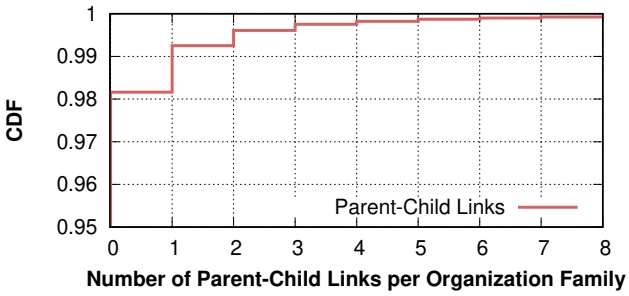

Figure 8: CDF of parent–child links within each organization family. x-axis extends to 50, which is "NTT Communications", which manages multiple organizations such as NTT Security (Switzerland) AG, NTT Cloud Communications SAS.

ownership and control. These clusters provide a comprehensive AS-to-organization mapping that captures rebranding, acquisitions, and shared operational oversight.

Table 2 summarizes the key outcomes of the pipeline. Our final dataset offers global coverage of 111,470 ASes—spanning multiple RIRs and NIRs and reflecting the constantly evolving Internet landscape—and demonstrates how NER, LLMs, and post-clustering can be combined to produce a more nuanced view of organizational relationships.

5.2 Analysis of Organization Families on the Internet

In this section, we examine the *organization families* produced by our methodology, highlighting their size, internal complexity, and cross-regional presence. Recall that an organization family unites one or more ASes under a common controlling entity, capturing both alias relationships (i.e., different names for the same organization) and parent–child relationships (i.e., acquisitions or subsidiaries).

5.2.1 Size Distribution. We begin by exploring the distribution of ASes within each organization family. Figure 7 plots the distribution of the number of ASes per family; while the majority of these families remain small, with roughly 87.8% containing only a single AS, there is a notable tail of much larger families. Specifically, 51 organization families contain equal to or more than 50 ASes, and 21 families exceed 100 ASes. This reflects a bifurcation of the Internet's organizational landscape, where most entities manage only one or a few ASes, yet a smaller number of large-scale operators manage dozens–or even hundreds–of ASes worldwide.

5.2.2 Hierarchy and Parent–Child Relationships. To quantify the complexity of corporate structures within an organization family, we measure how many distinct parent–child links appear in each. Figure 8 shows the CDF of acquisition (i.e., parent–child) relationships for all families. While most remain simple (e.g., only 1.8% of families have any parent–child links at all) a small fraction displays high internal complexity, with some families including more than 10 reported acquisitions.

Case Study: Microsoft: One illustrative example highlights how our approach captures complex corporate histories. Within our dataset, AS54947 (ZeniMax) and AS202167 (ZeniMax Germany) both appear as subsidiaries of Microsoft, alongside Microsoft-owned ASes such as AS8069 and AS40066, reflecting the 2020 acquisition of ZeniMax by Microsoft [25]. Furthermore, Microsoft's subsequent acquisition of Activision Blizzard has added AS14588 (Activision Publishing), AS10793 (id Software, a ZeniMax subsidiary), and AS60229 (Demonware, an Activision subsidiary), all converging into a single organization family. *None* of the evaluated alternative datasets (e.g., WHOIS, PeeringDB) explicitly capture these mergers under one unified entity. By contrast, ASINT leverages broader Web information and LLM-based inference to properly cluster these ASes, despite their diverse registrations and historical records.

5.2.3 Multiple WHOIS Organization Handles. Even a same organization operating multiple ASes can register them under *different* WHOIS handles, whether for historical reasons, regional differences in naming, or acquisitions. Focusing on families with more than one AS, we find that 52.1% use a single handle across all their ASes, while the remaining 47.9% rely on multiple handles, with 165 (1.7%) of families referencing equal to or more than 10 distinct WHOIS organization handles.

This pattern underscores how formal registry data can *fragment* a single real-world operator into multiple nominal entities, complicating any purely WHOIS-based attempt to unify AS-to-organization mappings.

	Organizations	Crawled URLs	Crawled Text Chunks	Filtered Text Chunks	LLM Queries	Organization Families
Count	88,101	873,949	20,478,907	386,278	89,515	81,233

Table 2: Summary of pipeline statistics for ASINT.

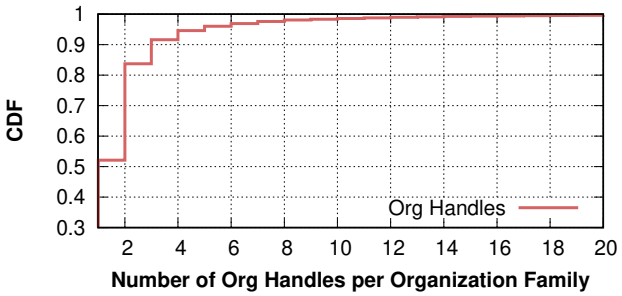

Figure 9: CDF of distinct WHOIS organization handles per organization family. The x-axis extends to 106, corresponding to `ChinaNet`, which has 106 organization handles.

Number of RIRs (or NIRs)	1	2	3	4	5	6
Organization Families	79,556	1,287	298	74	16	2

Table 3: Distribution of organization families by how many RIRs (or NIRs) they are registered in.

Case Study: Orange. Orange, a major ISP headquartered in France, exemplifies the complexity of operators using multiple WHOIS handles for globally distributed ASes. For example, AS8523 is listed under *ORG-BA89-RIPE* as "Orange Business Digital Sweden AB," AS30985 under *ORG-IS28-AFRINIC* as "Orange Mali SA", and AS327710 under *ORG-OCD2-AFRINIC* as "Orange Côte d'Ivoire" . These region-specific names reveal subtle metadata differences. PeeringDB coverage is likewise fragmented: AS8523 appears as "Basefarm AS" with their website. AS30985 is "Orange Mali SA" with no listed website, and AS327710 is absent altogether—further underscoring Orange's multi-country model and the need for consistent data reconciliation.

5.2.4 Cross-RIR Registration. RIRs separately oversee AS Number (and IP address) allocations in different parts of the world. Consequently, *multinational organizations* often hold resources from multiple RIRs, making it challenging to identify cross-regional ownership based solely on WHOIS or PeeringDB. By combining multiple data sources, ASINT reveals that 1,677 organization families appear in more than one RIR. Table 3 summarizes how many families span {1, 2, 3, 4, 5, 6} registries (including NIRs).

Case Study: Deloitte. Table 4 showcases one family, Deloitte that registers ASes in all five RIRs. For instance, AS42536 is registered with RIPE under "Deloitte LLP", while its subsidiary [5], "National TeleConsultants LLC," registers AS55228 in ARIN with different organization name. Other ASes reflect further naming variations across APNIC, LACNIC, and AFRINIC. Although PeeringDB covers some of Deloitte's ASes, they appear under various partial names, illustrating that neither standalone WHOIS nor PeeringDB captures the complete scope of Deloitte's worldwide network footprint.

ASN	Organization Name	RIR/NIR	Incl. PeeringDB
328312	Deloitte Touche South Africa	AFRINIC	Yes
42536	Deloitte LLP	RIPE	No
132384	Deloitte Consulting India Pvt. Ltd	APNIC	No
55228	National TeleConsultants LLC	ARIN	No
272103	DELOITTE TOUCHE LTDA	LACNIC	No
131077	Deloitte Tohmatsu Group LLC	JPIRR	Yes

Table 4: Deloitte's organizations appear in six different RIRs, yet only two are listed in PeeringDB.

5.2.5 Summary. In summary, our analysis shows that:

- *Most families have a single or small number of ASes*, but a select few control hundreds of ASes worldwide.
- *Parent–child relationships* are relatively sparse, yet large families often involve multiple acquisition links, reflecting complex corporate structures.
- *Multiple WHOIS handles* for a single organization are common, especially for large ISPs or multinational operators, complicating efforts to unify these disparate references without additional context.
- *Cross-RIR registration* poses a significant challenge for purely RIR-based or PeeringDB-based methods, but our broader data collection and inference steps succeed in linking multinational entities into cohesive organization families.

Collectively, these findings highlight the value of augmenting WHOIS and PeeringDB records with open-text Web sources, especially when combined with an LLM-based pipeline that merges aliases and parent–child structures. The resulting organization families illuminate the global landscape of AS ownership and management, revealing how complex and intertwined the modern Internet has become.

Metric	CA2O		AS2ORG+		AS-Sibling	
	ASINT	CA2O	ASINT	AS2ORG+	ASINT	AS Sibling
Common ASes	111,346		111,351		111,067	
Organization Families	81,160	89,069	81,162	88,866	81,152	87,927
Size > 1	9,888	8,263	9,889	8,235	9,881	8,559
Avg. family size (ASes)	4.05	3.69	4.05	3.73	4.03	3.70
Identical families	77,313		77,188		77,684	
Aggregations (Org. Families)	3,847 (11,756)	0 (0)	3,816 (11,591)	78 (158)	3,231 (10,101)	128 (236)
Aggregations (ASes Num)	20,267	-	20,151	2,424	18,076	1,787
Man. Validated Families (AS num)	100 (1,686)	-	100 (1,686)	40 (778)	100 (1,686)	50 (1,001)
True Positive Ratio	93.6%	-	93.6%	88.2%	93.6%	92.7%
Residual families	0	0	0	9	1	14

Table 5: Comparison of ASINT with three baseline datasets. Note that Avg. family size (ASes) only counts families containing more than one ASes; "Aggregations (Org. Families)" reports B families from a baseline dataset are merged into A families of ASINT in format "A (B)". When the numbers appear in the other columns, this indicates a "reverse aggregation" by the baseline.

5.3 Comparison with Prior Datasets

In this section, we compare ASINT with three previously published AS-to-Organization datasets.[1]

- *CA2O* [10]: A WHOIS-based mapping of ASes to organizations.
- *AS2ORG+* [2]: An extension of CA2O that supplements WHOIS data with additional metadata from PeeringDB and CAIDA's `as_relationship` dataset.
- *AS-Sibling* [9]: A dataset focusing on sibling relationships between ASes, derived primarily from PeeringDB plus CA2O and BGP data.

We aim to quantify how often ASINT identifies additional organization links (i.e., "aggregations") not seen in these other datasets, as well as whether ASINT sometimes misses valid links or merges ASes erroneously.

We also conduct a manual validation study on selected cases to assess the accuracy of ASINT in real-world scenarios.

5.3.1 Methodology. To ensure fair comparisons, we first identify the *common ASes* appearing in both ASINT and each baseline dataset. While each dataset may have a different total number of ASes (due to data-access constraints or collection dates), analyzing only the intersection allows us to evaluate how each system groups the *same* ASes.

All four datasets cluster ASes under higher-level entities, but they use different terminology: "Organizations" (CA2O, AS2ORG+) and "Siblings" (AS-Sibling). For consistency, we refer to these groupings as organization families throughout the comparisons.

Core Metrics. For the overlapping AS sets, we compute:

(1) *Organization Family Counts and Sizes:* The total number of organization families in each dataset and their average size (mean number of ASes per family for families containing more than one AS).
(2) *Identical Families:* The number of families whose AS membership is *exactly the same* in both datasets.
(3) *Aggregation Cases:* Situations where one family from Dataset A *fully contains* multiple smaller families from Dataset B. This usually indicates that Dataset A merges ASes that Dataset B keeps separate. We measure this for both directions.
(4) *Residual Families:* Families that do not match or get fully subsumed on either side (i.e., they remain uniquely defined in one dataset, with no counterpart in the other).

5.3.2 Overall Results and Statistics. Table 5 summarizes the main outcomes of our comparisons across the three baseline datasets. We list the number of ASes common to both datasets, the resulting number of organization families (and how many of those have more than one AS), and the average family size.

Overall, ASINT tends to identify slightly larger groups of ASes (i.e., higher average family size), suggesting that our pipeline captures additional corporate or alias relationships beyond those found purely through WHOIS or PeeringDB.

Comparison with CAIDA's AS2ORG. We collect the latest snapshot of CAIDA's AS2ORG dataset. After filtering for the 111,346 ASes shared in both datasets, ASINT groups them into 81,160 organization families, while AS2ORG produces 89,069. On average, each ASINT family contains 4.05 ASes, compared to 3.69 for AS2ORG.

Regarding the overlap, 77,313 families are identical across both datasets, containing exactly the same AS membership. However, ASINT merges 11,756 AS2ORG families into just 3,847 larger ones (involving 20,267 ASes). Because ASINT

[1]We also attempted to evaluate ASINT with Borges [28]; however, the dataset has not been released publicly at the time of submission.

partly relies on CAIDA's data as input (alongside more extensive Web-based crawling), there are no "residual" families in AS2ORG that ASINT fails to match.

Comparison with AS2ORG+. AS2ORG+ [2] extends CAIDA's baseline by incorporating PeeringDB and the `as_relationship` dataset. Recreating AS2ORG+ with our WHOIS, PeeringDB, and `as_relationship` data yields 111,351 overlapping ASes. From these, ASINT produces 81,162 organization families, whereas AS2ORG+ forms 88,866. Additionally, ASINT shows a larger average family size (4.05 versus 3.73), indicating it merges more ASes into unified entities.

While ASINT aggregates 11,591 AS2ORG+ families into 3,816 of its own, we observe that 78 AS2ORG+ families merge 158 of ASINT (covering 2,424 ASes). This "reverse aggregation" largely arises from differences in input data—particularly the `as_relationship` dataset—and highlights that ASINT, though generally more inclusive, may still miss merges driven by partnerships outside our search scope. For example, CenturyLink Communications, LLC partners with Air Force Systems Networking, leading to a reverse aggregation scenario not covered by ASINT.

Comparison with AS Sibling. We examine their snapshot (Jan 2025), finding 111,067 ASes in common. ASINT yields 81,152 families, while AS Sibling has 87,927.

As before, ASINT aggregates 10,101 AS Sibling families into 3,231 of its own (involving 18,076 ASes). Conversely, AS Sibling aggregates 236 of ASINT into 128 of its own (1,787 ASes). We also observe 15 "residual" families—1 in ASINT and 14 in AS Sibling—caused largely by differences in data-collection timestamps and PeeringDB coverage.

Although AS-Sibling adopts a narrower scope and forms fewer but broader "sibling" clusters (reducing the chance of merges and misclassifications), ASINT achieves a slightly higher true positive ratio (93.6% vs. 92.7%).

This is particularly encouraging given that ASINT tackles a wider range of ASN relationships (e.g., rebrands, parent–child) and aims for global coverage, yet still maintains superior accuracy.

5.3.3 Manual Validation of Aggregations. Not all merges or splits are necessarily correct; some may be genuine discoveries of corporate relationships, while others could be false positives. We therefore manually inspect a sample of *aggregated* families—cases where ASINT unifies multiple smaller groups that the other dataset keeps separate, or vice versa. We randomly select 100 such cases from each baseline comparison, representing 1,686 ASes.

Our analysis shows that ASINT achieves a high true-positive rate for newly aggregated families, confirming that many of its merges reflect real-world acquisitions or shared ownership. However, we also find a smaller subset of false-positive merges (6.4%) caused by:

- *Stale or incomplete acquisition histories* (e.g., a former subsidiary was later divested, but that information did not appear in our crawled data).
- *Overly permissive fuzzy matching* in search results or entity recognition, conflating two similarly named organizations.
- *LLM inference errors* that persisted despite prompt engineering and post-filtering.

Meanwhile, we also encounter a handful of *missed merges* cases where a baseline dataset merges ASes correctly, but ASINT does not. These typically stem from incomplete or inaccessible online records (e.g., paywalled sources or pages blocked by anti-bot measures).

5.3.4 Limitations and Future Improvements. Our findings demonstrate that ASINT frequently reveals additional organizational relationships beyond those found by prior approaches, yet some gaps and inaccuracies remain:

- *False-positive merges* can arise from incomplete or outdated acquisition records, ambiguous entity names, or LLM hallucinations. Further refining retrieval-augmented generation (RAG) and applying stricter cross-checking of corporate structure data may help mitigate such errors.
- *Missed merges* highlight the importance of more comprehensive data sources. Incorporating additional metadata (e.g., press releases or brand databases) could improve coverage.
- *Temporal dynamics* remain a major challenge: corporate relationships can shift quickly as organizations merge, rename, or dissolve, making our snapshot outdated.

Overall, these comparisons confirm that ASINT complements and extends existing AS-to-Organization mapping datasets by systematically leveraging unstructured Web data and LLM-based inference. While no single approach can capture the entire global Internet perfectly, our results suggest that harnessing broader data sources is a significant step toward an accurate, up-to-date view of AS ownership and management.

5.4 Performance Benchmark

In this section, we evaluate the individual components of ASINT to demonstrate how they contribute to the overall accuracy, scalability, and cost-effectiveness of our approach.

5.4.1 Effectiveness of NER-Based Filtering. While LLMs can reason about complex relationships, they are not well-suited

Model	Estimated Cost ($)			Org Families		ASes	
	Per 1M Tokens	Full Training		Identified	Overlap w/ 4o-mini	Identified	Overlap w/ 4o-mini
		Est. w/o NER	w/ NER				
4o-mini	$0.15	$1,360	$60	1,000	–	1,629	–
4o	$2.5	$23,943	$1,221	1,020	963 (96.3%)	1,629	1,553 (95.3%)
o3	$10	$148,835	$6,922	1,021	959 (95.9%)	1,629	1,538 (94.4%)

Table 6: A comparison of three model variations (4o-mini, 4o, o3) with their associated costs, organization family counts, and AS information.

to sifting through massive, noisy corpora unaided [16]. Without entity filtering or a vector-based retrieval mechanism, the model risks being overwhelmed by irrelevant text.

We tested randomly chosen 100 previously verified parent–child AS pairs in a scenario where the LLM received *unfiltered* raw HTML from our crawled data (an average of 16,473 tokens per pair vs. 310 tokens after NER filtering) The model inferred the correct relationship in only 84% of cases, and most false negatives arose because the LLM failed to locate the single relevant sentence among large, messy HTML documents. This underscores the importance of NER-based filtering; ASINT reduces cost and frees the LLM from searching vast, irrelevant text;

5.4.2 Impact of Retrieval-Augmented Generation (RAG). Another cornerstone of ASINT is the use of RAG to feed *context-specific snippets* into the LLM, rather than relying on the model's built-in or memorized knowledge. LLMs may have been trained on older or incomplete data, and organizational structures can change rapidly (e.g., new acquisitions).

We sampled 100 verified parent–child organization pairs and asked the LLM (OpenAI 4o-mini) to infer their relationship *without any reference to external context*, effectively testing whether the model "already knows" about these corporate links. The accuracy dropped to 36%, demonstrating that many valid relationships cannot be reliably retrieved from the model's training set alone.

With RAG, ASINT supplies the relevant text from web pages, press releases, which enables the LLM to consistently identify the relationship, preventing *knowledge gaps* or *training-time cutoffs* from undermining accuracy; as new acquisitions occur, simply updating our crawled metadata suffices.

5.4.3 Necessity of Post-Filtering and Clustering. Even with careful NER and RAG, the LLM can produce erroneous or overly broad inferences (e.g., merging many unrelated organizations under the same name if a few ambiguous tokens appear in their text).

To quantify the value of post-filtering and hierarchy construction (§5.4.3), we removed that step and measured the effect on final organization families. Out of 81,233 families, 12,368 changed significantly. In one extreme case, 4,459 families covering 14,310 ASes collapsed into a *single* family, caused by generic terms in the collected web data (e.g., "Telecom", "Network") that eluded NER's filtering. Post-filtering prunes these spurious merges by requiring stronger textual evidence and majority voting before final clustering.

5.4.4 Sustainability and Choice of LLM Models. Accurate, cost-effective, and up-to-date AS-to-Organization mappings demand *repeated* LLM queries to accommodate newly introduced ASes and evolving corporate structures.

Cost Sensitivity. Table 6 illustrates how three variants of OpenAI's LLM—4o-mini, 4o, and o3—differ in both pricing and outcome. Specifically, we report the cost per 1 million input tokens (output tokens cost 4 times as much), an estimate of the total expense if we processed *all* crawled text without NER filtering (*Est. w/o NER*), and the actual cost when leveraging our filtering pipeline (*w/ NER*). The contrast is striking: for 4o-mini, the projected cost drops from $1,360 (no filtering) to just $60 (with NER).

These savings underscore the value of carefully pruning irrelevant text before passing it to the LLM.

Model Comparison. We evaluate these three LLMs using a randomly selected sample of 1,000 organization families including 1,629 ASes. Table 6 shows that each model uncovers a slightly different number of final organization families—1,000 for 4o-mini, 1,020 for 4o, and 1,021 for o3. Despite these small discrepancies (e.g., a 21-family difference between 4o-mini and o3), overlap remains high: 96% of the ASes are mapped identically to those in 4o-mini, indicating that ASINT achieves robust consistency across models.

6 USECASES AND IMPACT

6.1 Organization Cone Size and Ranking

Accurate estimates of an ISP's "customer cone" are central to many topology and ranking analyses [21]. CAIDA's existing AS Ranking dataset computes organization-level ranks by grouping ASes in its CA2O dataset and then measuring each group's transit cone (the set of ASNs, prefixes, or IP addresses served as customers).

Using ASINT, we recreate an analogous cone-size analysis. Among the 89,072 organizations in CAIDA's AS Rank dataset, we observe that 11,760 organizations see a larger cone size

Org Name	New Rank	New Cone Size
TATA	6 (▲ 2)	21,782 (+2,447)
Orange	12 (-)	9,958 (+1,927)
Charter	32 (▲ 32)	2,781 (+1,819)
Vodafone	14 (-)	8,479 (+1,708)
Comcast	23 (▲ 12)	3,862 (+1,513)
Liberty Global	21 (▲ 9)	4,030 (+1,361)
GlobeNet Cabos	19 (▲ 8)	4,156 (+1,340)
Telstra	13 (-)	8,930 (+1,282)
Deutsche Telekom	16 (▲ 3)	5,198 (+1,191)
Stowarzyszenie	44 (▲ 51)	1,652 (+1,014)

Table 7: Organization Rankings and Cone Metrics.

under our new mapping—often because ASINT merges previously fragmented AS sets belonging to the same real-world operator. Table 7 highlights 10 organizations that show the largest positive changes in cone size and rank. For example, Charter's cone size nearly doubles (from 962 to 1,819 ASes) after identifying 237 ASes under its control, raising its rank from 64 to 32. Such improvements cannot be captured by PeeringDB-based methods alone since Charter omits details in PeeringDB's "notes" field, and only 8 of its 237 ASes appear in PeeringDB at all.

6.2 RPKI Misconfiguration

The Resource Public Key Infrastructure (RPKI) cryptographically secures route advertisements by allowing resource holders to publish Route Origin Authorizations (ROAs). RPKI aids in preventing route hijacks, but *misconfigurations* can still arise if an organization incorrectly assigns multiple ASNs under its umbrella. Prior work [7] used CAIDA's CA2O to identify RPKI-invalid prefixes announced by the same organization's ASes.

We replicate this study using more recent data from January 2023 to July 2024, combining BGP routing tables from RouteViews [27] and ROAs from all five RIRs, leaving 42,654 RPKI-invalid prefixes for analysis. Applying CA2O finds 4,436 of these as *intra-organization* misconfigurations, whereas ASINT identifies an additional 1,219 such cases (a 27.5% increase). These newly discovered cases were previously miscategorized because CA2O did not recognize that the two ASNs belonged to the same entity.

6.3 IP Leasing Detection

As IPv4 space becomes scarce, IP leasing—where organizations temporarily rent out IP addresses—has grown into a significant market. Distinguishing legitimate intra-organization reallocation from actual cross-organization leasing is important for accurately characterizing these markets.

Building upon [12], which used CA2O to detect leased addresses, we re-examined 47,318 previously flagged "leased" IP prefixes. Under our new mapping, 2,783 (5.9%) of those prefixes turn out to be reallocated *within the same* organization, rather than leased. For example, 193.82.107.0/24 was incorrectly labeled as leased between AS1290 and AS4637, although both are actually owned by `Telstra`. Thus, using ASINT avoids overestimating lease activity.

6.4 Hijack Detection

BGP hijacking remains a serious threat to Internet security, allowing attackers to redirect or intercept traffic. Existing hijack detection platforms (e.g., Radar [11] and GRIP [1]) often rely on CA2O to filter out potential false positives: if two ASes belong to the same organization, an overlapping prefix announcement might be legitimate rather than malicious.

We collected 17,282 hijack alerts from Radar and 11,450 from GRIP spanning January 2023 to July 2024, none of which were flagged as intra-organization by CA2O. However, by applying our dataset, we find that 1,465 (8.5%) of Radar's alarms and 1,326 (11.6%) of GRIP's alarms involve AS pairs owned by the same organization—thus false positives.

In total, 1,621 (9.4%) hijack alerts across both platforms are, in fact, benign. We validated 100 randomly selected cases by emailing each allegedly "victimized" operator's publicly listed contact. Of the 32 who responded, *all* of them confirmed the event was an internal reannouncement rather than a hijack. This underscores how improved AS-to-Organization mapping can substantially reduce unnecessary alarms, saving time and effort for both operators and security teams.

7 CONCLUDING DISCUSSION

We have presented ASINT, a system that extends ASN-to-organization mapping by integrating WHOIS, PeeringDB, and a wide range of unstructured web sources through a RAG pipeline. ASINT uncovers a richer set of cross-regional aliases, major rebrandings, and complex parent–child ownership links. Our key findings include:

- *Coverage and Aggregation:* Analyzing over 111,000 ASNs, ASINT produces 81,233 organization families, merging thousands of AS2Org entries into fewer, larger families and revealing additional intra-organizational relationships.
- *Operational Impact:* By more accurately grouping ASNs, ASINT increases the detection rate of intra-organizational RPKI misconfigurations by 27.5% and reduces false alarms in hijack detection by 9.4%. It also decreases misclassified IP leasing cases by 5.9%.
- *Real-World Relevance:* Enriching registry data with external sources (e.g., news articles) allows us to track rebrandings and acquisitions well before official registry updates.

Sustainability. To ensure ASINT remains viable over time, we consider the following:

- *Continuous Updates:* ASINT is designed for periodic rechecks, enabling incremental ingestion of WHOIS data and new web crawls that reflect ongoing corporate changes.
- *Cost-Aware LLM Choices:* Since inference constitutes a recurring expense, ASINT can adapt to different LLM models based on a desired balance of cost and accuracy.
- *Additional Metadata:* Future plans include expanding data sources (e.g., brand databases, multilingual press releases) and incorporating structured signals (e.g., domain certificates, enterprise group filings) to broaden coverage and reduce remaining false positives.

Overall, ASINT demonstrates the value of fusing registry data with unstructured web evidence. By exposing cross-regional affiliations, hidden mergers, and alternate brand identities, our approach provides a more holistic view of the Internet's organizational landscape. We will publish the dataset with periodic updates, thereby supporting future research and strengthening operational practices.

REFERENCES

[1] GRIP - Global Routing Intelligence Platform. https://grip.inetintel.cc.gatech.edu.

[2] A. Arturi, E. Carisimo, and F. E. Bustamante. as2org+: Enriching as-to-Organization Mappings with PeeringDB. *PAM*, 2023.

[3] Altice USA Announces Closing of Sale of 49.99% of Lightpath Fiber Enterprise Business to Morgan Stanley Infrastructure Partners. https://www.alticeusa.com/news/articles/press-release/corporate/altice-usa-announces-closing-sale-4999-lightpath-fiber-enterprise-business-morgan-stanley.

[4] Altice acquires Cablevision and creates the #4 cable operator in the US market. https://www.alticeusa.com/news/articles/press-release/corporate/altice-acquires-cablevision-and-creates-4-cable-operator-us-market.

[5] B. BARRETT. Deloitte Consulting acquires National TeleConsultants in media enterprise push. 2022. https://erp.today/deloitte-consulting-acquires-national-teleconsultants-in-media-push/.

[6] B. Chen, Z. Zhang, N. Langrené, and S. Zhu. Unleashing the potential of prompt engineering in Large Language Models: a comprehensive review. 2024. https://arxiv.org/abs/2310.14735.

[7] T. Chung, E. Aben, T. Bruijnzeels, B. Chandrasekaran, D. Choffnes, D. Levin, B. M. Maggs, A. Mislove, R. van Rijswijk-Deij, J. P. Rula, and N. Sullivan. RPKI is Coming of Age: A Longitudinal Study of RPKI Deployment and Invalid Route Origins. *IMC*, 2019.

[8] X. Cai, J. S. Heidemann, B. Krishnamurthy, and W. Willinger. Towards an as-to-Organization Map. *IMC*, 2010.

[9] Z. Chen, Z. S. Bischof, C. Testart, and A. Dainotti. Improving the Inference of Sibling Autonomous Systems. *PAM*, 2023.

[10] CAIDA ASOrganizations Dataset. http://www.caida.org/data/as-organizations/.

[11] Cloudflare Radar. https://radar.cloudflare.com/routing.

[12] B. Du, R. Fontugne, C. Testart, A. C. Snoeren, and k. claffy. Sublet Your Subnet: Inferring IP Leasing in the Wild. *IMC*, 2024.

[13] D. Dittrich and E. Kenneally. The Menlo Report: Ethical Principles Guiding Information and Communication Technology Research. 2012. https://www.dhs.gov/sites/default/files/publications/CSD-MenloPrinciplesCORE-20120803_1.pdf.

[14] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, M. Wang, and H. Wang. Retrieval-Augmented Generation for Large Language Models: A Survey. 2023. https://arxiv.org/abs/2312.10997.

[15] H. Husain, H.-H. Wu, T. Gazit, M. Allamanis, and M. Brockschmidt. CodeSearchNet Challenge: Evaluating the State of Semantic Code Search. 2019. https://arxiv.org/abs/1909.09436.

[16] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin, and T. Liu. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. *ACM TOIS*, 43, 2025.

[17] L. Jing, S. Aixin, H. Jianglei, and L. Chenliang. A Survey on Deep Learning for Named Entity Recognition. *IEEE Transactions on Knowledge and Data Engineering*, 34(1), 2022.

[18] J. Kaddour, J. Harris, M. Mozes, H. Bradley, R. Raileanu, and R. McHardy. Challenges and Applications of Large Language Models. 2023. https://arxiv.org/abs/2307.10169.

[19] N. Kandpal, H. Deng, A. Roberts, E. Wallace, and C. Raffel. Large Language Models Struggle to Learn Long-Tail Knowledge. *ICML*, 2023.

[20] D. Lim. Bert-base-NER. https://huggingface.co/dslim/bert-base-NER.

[21] M. Luckie, B. Huffaker, K. Claffy, A. Dhamdhere, and V. Giotsas. AS Relationships, Customer Cones, and Validation. *IMC*, 2013.

[22] M. Levandowsky and D. Winter. Distance between Sets. *Nature*, 234, 1971.

[23] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, and D. Kiela. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. *NIPS*, 2020.

[24] X. Li, Z. Liu, C. Xiong, S. Yu, Y. Gu, Z. Liu, and G. Yu. Structure-Aware Language Model Pretraining Improves Dense Retrieval on Structured Data. *ACL*, 2023.

[25] Microsoft. Microsoft finalizes acquisition of ZeniMax Media. 2020. https://news.microsoft.com/features/microsoft-finalizes-acquisition-of-zenimax-media/.

[26] PeeringDB: The Interconnection Database. https://www.peeringdb.com/.

[27] University of Oregon RouteViews project. http://www.routeviews.org/.

[28] C. Selmo, E. Carisimo, F. E. Bustamante, and J. I. Alvarez-Hamelin. Learning as-to-Organization Mappings with Borges. *IMC*, 2025.

[29] K. Shuster, S. Poff, M. Chen, D. Kiela, and J. Weston. Retrieval Augmentation Reduces Hallucination in Conversation. 2021. https://arxiv.org/pdf/2104.07567.

[30] P. Sahoo, A. K. Singh, S. Saha, V. Jain, S. Mondal, and A. Chadha. A Systematic Survey of Prompt Engineering in Large Language Models: Techniques and Applications. 2024. https://rotmandigital.ca/wp-content/uploads/2024/09/A-Systematic-Survey-of-Prompt-Engineering-in-Large-Language-Models.pdf.

[31] The High-Performance Vector Database Built for Scale. https://milvus.io/.

[32] P. Xu, W. Ping, X. Wu, L. McAfee, C. Zhu, Z. Liu, S. Subramanian, E. Bakhturina, M. Shoeybi, and B. Catanzaro. Retrieval meets Long Context Large Language Models. *iclr*, 2024.

[33] Y. Zhang, Y. Li, L. Cui, D. Cai, L. Liu, T. Fu, X. Huang, E. Zhao, Y. Zhang, Y. Chen, L. Wang, A. T. Luu, W. Bi, F. Shi, and S. Shi. Siren's Song in the AI Ocean: A Survey on Hallucination in Large Language Models. https://arxiv.org/abs/2309.01219.

[34] Z. Zhao, E. Wallace, S. Feng, D. Klein, and S. Singh. Calibrate Before Use: Improving Few-shot Performance of Language Models. *ICML*, 2021.

A ETHICAL CONSIDERATIONS

ASINT collects raw data from Internet, involves the use of web searching and crawling as the underlying data collection technique. We acknowledge that web crawling can raise ethical concerns, particularly regarding the potential for overloading servers or violating terms of service. To mitigate these risks, we follow the Menlo Report principles [13], and implement several best practices in our data collection process:

- Respecting Robots.txt: We respect the `robots.txt` files of the websites we crawl, which specify the rules for web crawlers regarding which parts of the site can be accessed.
- Rate Limiting: We have implemented rate limiting in our crawling process to avoid overwhelming any single server with requests. This helps to ensure that our crawling activities do not disrupt the normal operation of the websites we access.
- User-Agent Identification: We identify our crawler with a specific user-agent string, allowing website administrators to recognize our crawler and take action if necessary.
- Data Usage: The data collected is used solely for research purposes and is not shared with third parties without proper anonymization or aggregation.

B REPLICABILITY

We public ASINT code and analysis scripts at

https://anonymized.for.reviews

for network operators, administrators, and researchers to reproduce our work.

We also provide a public available dataset for ASINT AS-to-Organization mapping dataset, continuously updated with new information.

C PROMPT

Figure 10 and Figure 11 shows the prompt we used in the pipeline.

Few-Shot Prompt

```
You are an expert at determining how organizations that own or control Autonomous System (AS) numbers in computer
networks are related, using the provided context.

You will receive:
1) A **base_organization**.
2) A list of **candidate_organizations**.
3) **context** providing relevant organizational details.

### Definitions
For each (base_organization, candidate_organization) pair, decide which of the following relationships
best applies:

- Alias
  Both names refer to exactly the same legal entity or one is a historical name of the other.

- Parent/Subsidiary
 One organization has acquired or holds more than 50% of the stock of the other. Identify which one is the parent:
  Choose between "base_organization" or "candidate_organization".

- No_relation
  There is insufficient evidence of alias, ownership or acquisition linking them.

**Mandatory JSON Output Format**:
Your output must be in valid JSON format only, with no extra text or commentary. Use the structure:

{
   "base_org_name": <Name of base organization>,
   "candidate_org_name": <Name of candidate organization>,
   "reasoning for Alias": <Explanation for why this pair is or is not Alias>,
   "reasoning for Parent/Subsidiary": <Explanation for why this pair is or is not Parent/Subsidiary>,
   "relationship": <One of "Alias", "Parent/Subsidiary", or "No_relation">,
   "parent": <If Parent/Subsidiary, indicate "base" or "candidate"; otherwise leave empty>,
   "parent name": <If Parent/Subsidiary, exactly match the relevant org name from "base_organization"
      or "candidate_organization"; otherwise leave empty>
}

### Example:
Provide this JSON object for each (base_organization, candidate_organization) pair as an array for output.
Examples:
[
   {
      "base_org_name": "Zayo Bandwidth",
      "candidate_org_name": "company",
      "reasoning for Alias": "The candidate name is generic and lacks direct evidence connecting it
         to Zayo Bandwidth.",
      "reasoning for Parent/Subsidiary": "No indication of ownership or acquisition.",
      "relationship": "No_relation",
      "parent": "",
      "parent name": ""
   },
```

Figure 10: Full prompt used for few-shot inference.

Few-Shot Prompt (Continued)

```
   {
      "base_org_name": "Google inc.",
      "candidate_org_name": "YouTube",
    "reasoning for Alias": "Google inc. acquired YouTube in 2006, so they are not aliases but parent and child.",
      "reasoning for Parent/Subsidiary": "YouTube is a subsidiary of Google inc.",
      "relationship": "Parent/Subsidiary",
      "parent": "base",
      "parent name": "Google inc."
   },
   {
      "base_org_name": "Google inc.",
      "candidate_org_name": "google",
      "reasoning for Alias": "The name 'google' consistently refers to the same entity 'Google Inc.'",
    "reasoning for Parent/Subsidiary": "No separate ownership details suggest a parent/subsidiary relationship.",
      "relationship": "Alias",
      "parent": "",
      "parent name": ""
   }
]

### Input
"base_organization": {base_org}

"candidate_organizations": {target_org}

"Context": {context}

Now, respond by considering each candidate_organization in the list, applying reasoning, and returning your
final JSON array with one object per candidate.
```

Figure 11: Full prompt used for few-shot inference. (Continued)