Towards a Quantitative Comparison of the Cost-Benefit Trade-offs of Location-Independent Network Architectures

ABSTRACT:
This paper presents a quantitative methodology and results comparing different approaches for location-independent communication. Our approach is empirical, making comparisons using realistic Internet topologies, routing tables from real routers, and a measured workload of content and device mobility across network addresses in today’s Internet. To this end, we make the following contributions. First, we conduct a measurement of the extent of network mobility (or changing network addresses) exhibited by mobile users with a home-brewed Android app deployed on hundreds of smartphones, as well as the extent of network mobility exhibited by popular content served by CDNs today. Second, we develop a quantitative methodology to investigate the different cost-benefit tradeoffs struck by different location-independent architectures with respect to forwarding table size, routing update cost, and path stretch. Third, we combine this methodology with the network mobility traces to quantify these tradeoffs for a hypothetical deployment of these architectures on top of existing networks in the current Internet. We find that with purely name-based routing approaches, each event involving the mobility of a device or popular content may result in an update to up to 8% of Internet routers; however, the fraction of impacted routers is much smaller for the long tail of unpopular content. These results suggest that recent proposals for information-centric networking may be suitable for content that does not move frequently but may need to be augmented with Mobile IP or DNS like approaches to handle device mobility. We also find that more than 20% of users make more than 10 IP address transitions a day, suggesting that mobility is the norm rather than the exception, so intrinsic and efficient network support for mobility is critical.

1. INTRODUCTION
The goal of providing an abstraction of location-independent communication – enabling communication using fixed names without concern for changing network locations – has been a long-time goal of networking research. For example, proposals designed to provide seamless host mobility target an abstraction of the form $\text{connect}(\text{host_id})$; a number of proposals for information-centric networking target an abstraction of the form $\text{get}(\text{content_name})$.

Why is the current TCP/IP Internet seen as falling short of this goal? A common criticism is the so-called location-identity conflation problem. The Internet uses an IP address to identify an interface as well as its network location. As a result, connections break when an endpoint changes network addresses, requiring application-layer workarounds to provide a semblance of seamless mobility. Advocates of information-centric networking argue that the Internet, having inherited a century-old, tethered, device-to-device communication abstraction from the wired telephony world, is poorly-suited for an Internet dominated by content traffic and a communication abstraction requiring endpoints to first obtain the network location of a host serving the requested content instead of simply procuring a copy from any convenient location. In response, researchers have proposed a number of designs to refactor naming, addressing, and routing for location-independent communication.

Our work is motivated by the observation that, although several architectural proposals share location-independence as a key design goal, there has been little prior research quantitatively comparing the different cost-benefit tradeoffs struck by these architectures in achieving that goal, particularly for the cases that device or content are mobile. One reason for the paucity of cross-architectural comparisons is that network architecture is considered by some as part science and in good part art. Another is that until recently, most Internet architectural efforts rarely went beyond paper designs, so a lack of a reasonably complete design specification and protocol-level implementation made it hard to justify investing research effort comparing different architectures; however programs such as GENI, FIA, and FIRE are changing this state of affairs. We believe that a quantitative comparison of different approaches for location-independent communication is timely.
and indeed critical for gaining a deeper understanding of cross-cutting architectural principles.

In this paper, we present a quantitative methodology and results comparing different location-independent architectures based on a common set of metrics that include update cost at routers, path stretch, and forwarding table size in a context in which either devices or content may be mobile. We find that a number of existing approaches for location independence can be categorized into one of three “puristic” classes: (2): indirection routing (e.g., Mobile IP or GSM), name-to-address resolution (e.g., DNS and IP), and name-based routing (e.g., ROFL [11] or NDN [26]). Our methodology is empirical, comparing these three pure approaches using realistic Internet topologies, routing tables from real routers, and a measured workload of content and device mobility across addresses in today’s Internet. (§).

To measure device mobility across network locations, we have developed an Android app, AppName that is currently deployed on 288 Android devices and has been collecting data on device mobility among networks for over 10 months. To measure content mobility, we deployed a system across distributed Planetlab locations to estimate the rate of change of network addresses of popular content domain names (including those delegated to CDNs) as well as unpopular content domain names. We combine this measured mobility workload with our evaluation methodology to arrive at the following key findings.

• Pure name-based routing entails a prohibitively high router update cost to handle device mobility today, e.g., we find that some routers may be impacted by up to 8% of all device mobility events.

• Over 20% of users make more than 10 network address transitions a day, suggesting that mobility is the norm rather than the exception, so intrinsic support for mobility is critical. We also find that the median user spends around 25% of a day in ASes at least two AS hops away from the dominant AS, implying a commensurate path stretch for indirection routing approaches.

• Pure name-based routing imposes a much lower update cost for content mobility compared to device mobility today, e.g., we find that with best-port forwarding, routers are impacted by at most 4% of popular content mobility events, and are hardly impacted by unpopular content mobility.

• Content names have an aggregateability of 2× to 16× implying a commensurate reduction in the size of forwarding tables compared to the total number of content domains.

An important implication of our findings is that name-based routing approaches in their pure form are more well-suited for content distribution alone, and may need to be augmented with addressing-based approaches such as DNS, Mobile IP or a next-generation name-resolution service in order to serve as a general-purpose replacement for the TCP/IP Internet. Our findings also illustrate the important differences between device and content mobility, as well as the emerging importance of the strategy layer in content-oriented architectures.

The remainder of this paper is structured as follows. In the following section we provide background and discuss related work. Section 3 describes our evaluation methodology, and Section 4 describes our approach for measuring device transitions among networks. Section 5 describes an analytic path-stretch model and motivates our later empirical evaluations of this metric. Sections 6 and 7 analyze the cost-benefit tradeoffs offered by the three major approaches for location independence in the face of device and content mobility, respectively. Section 8 provides additional discussion and Section 9 concludes this paper.

2. BACKGROUND AND RELATED WORK

Despite the enormous body of work on refactoring naming, addressing, and routing so as to enable location-independent communication, most known approaches fundamentally take one of just three different approaches: (1) indirection routing; (2) name resolution; (3) name-based routing.

The fact that there are not too many different approaches to enable location independence should not be surprising – in order to enable a communication abstraction of the form connect(B,A) where B and A are fixed names of endpoint principals (e.g., hosts, content, or services), each of which could be simultaneously domiciled at different locations at any point in time and the set of these locations could suddenly change, endpoint B must resort to one of three options in order to send the first packet to A—(1) send the packet to one (or a small number of) network router(s) that know(s) A’s current location; (2) acquire knowledge of A’s current location through an extra-network service and send the packet to that location; or (3) send the packet stamped with the name A to any router, trusting a coordinated routing and forwarding strategy across routers to deliver the packet to A’s current location. These three approaches are illustrated in Figure 4. Let us next consider how a number of proposed network architectures embody these approaches.

Indirection routing translates the problem of fixed-name routing to fixed-address routing. A name resolution infrastructure may be infrequently queried to translate an endpoint’s name to a home address that rarely changes by design. A home agent in the home network as in GSM, Mobile IP and other architectures (e.g., [29]) or a randomly chosen rendezvous location
in i3 [44] is responsible for maintaining up-to-date visited network locations of its subscribers and detouring packets to them. The main strength of indirection is simplicity – an endpoint remains completely oblivious to endpoint mobility. The downside is path stretch – in order to remain oblivious, a sender must detour all packets through the destination’s home agent [40], even if the two endpoints happen to be in the same local network. The update cost of indirection routing is minimal because an endpoint only needs to update its home agent upon each mobility event. The forwarding table size at routers remains unchanged and is equal to the number of disaggregated prefixes (e.g., ≈ 400K at core routers in today’s Internet).

Name resolution approaches rely on an extra-network service like DNS as the first step in all network communication in order to resolve a destination’s name to its current address. A name resolution service is an integral part of the Internet as well as a number of network architectures such as HIP, AIP, LISP, Nimrod, MobilityFirst, and XIA. As in indirection routing, the update cost of handling a mobility event is minimal because it is sufficient to make a single update at a logical centralized service like DNS. Network routing is based on structured addresses, so the forwarding table size at routers depends on the “aggregateability” of endpoint IP addresses, e.g., if all endpoints move about randomly, then routers would have to store on the order of 4B IP addresses instead of 400K prefixes today. The path stretch depends only upon the underlying network routing: a shortest-path routing network has no stretch by definition while policy-driven interdomain routing as in the Internet can incur significant path inflation [43]. Enabling location-independence through name resolution only adds a lookup latency at connection initiation time, but does not add additional data path stretch compared to underlying network routing.

Name-based routing, unlike both approaches above, routes directly over names without using addresses at all. Examples of such architectures include flat-label routing architectures such as ROFL [11] or SEATTLE [28] (for a single enterprise network) or information-centric routing architectures such as TRIAD [15] and NDN [26]. From an endpoint’s perspective, eliminating addresses is a silver bullet as the network comes with intrinsic support for location-independent communication. However, achieving this abstraction purely at the network layer without inducing significant stretch and without long outage times upon mobility events is non-trivial. Quantifying this tradeoff is an important focus of our work.

We note that all three broad approaches above have a number of additional advantages and disadvantages (e.g., incremental deployability, manageable, security, and handoff outage times). We have omitted a discussion of these in line with our goal of analyzing the cost-benefit tradeoffs with respect to quantitative metrics such as path stretch, update cost, and forwarding table size. An explicit non-goal of our work is to determine the “best” among existing location-independent architecture or to propose yet another new architecture.

2.1 Related work

The contributions of our paper lie at the intersection of network measurement (we empirically characterize the changing connectivity of both devices and content to the Internet) and network architecture (we analyzing the cost-benefit tradeoffs with respect to several metrics for different approaches for achieving location-independence). Consequently, there is related past research in several areas.

Network mobility measurement. Numerous studies have empirically characterized physical human mobility among access points or base stations and discussed the impact of physical user mobility patterns on network performance and design. Human mobility traces have been collected from diverse access networks such as WLANs [30, 24, 13], Bluetooth networks [13], and cellular networks [22, 38, 25]. [22, 38, 14] have related human mobility patterns to AP and base station resource use. [22, 38] have found that the extent of users’ physical mobility is low and concentrated among a small number base stations within a provider’s network, with infrequent visits to other base stations in that network. [53] characterizes metro-level path inflation (rather than mobility itself) experienced by mobile users accessing Google, identifying inter-domain routing, peering and carrier topology as causes of path infla-

Figure 1: A mobility event (a) and three purist approaches (b, c, d) to handle them.
tion but not directly characterizing mobility. Similarly, numerous efforts have focused on the measured performance (throughput, delay) of WiFi or cellular connections in the wild but focus on connection performance rather than on mobility itself. Acceloc and CelloScope describe applications that take measurements aimed at capturing location-specific information about cellular connections, but do not focus on mobility among locations nor among access networks as in our work.

Individual WiFi and cellular measurements include data from a single type of network, and more importantly characterize physical user/content mobility among access points or base stations, rather than changing points of attachment to the Internet (i.e., mobility among networks, as characterized by the changing IP addresses or AS numbers). Our AppName measurements characterize this latter aspect of mobility among networks, rather than physical mobility in space - a critical distinction. To illustrate this distinction, note that a physically mobile user might maintain its same IP address as it moves among base stations in a provider network; for our purposes (and for the purposes of the architectures whose performance we will study), this user is equivalent to being stationary - its IP address does not change in spite of physical mobility.

**In-network name resolution.** A number of studies have considered the performance of in-network name resolution. compares the performance of an instantiated NDN forwarding plane with traditional IP forwarding, with an emphasis on security and congestion mitigation. compares the performance of network-based name resolution in an instantiated NDN context versus a logically centralized approach as in MobilityFirst’s GNS approach, considering forwarding table size as a function of an abstract, parameterized model of name aggregation. Neither of these works consider either content or device mobility – the key consideration in our present paper.

Several recent efforts have considered name-based content request and delivery in a mobile environment. considers information dissemination in a linear V2V network using NDN, focusing primarily on the impact of protocol timer values on performance; our present paper focuses on mobility among multiple networks with general topologies and is aimed at a broad comparison between different location-independent network architectures. Proxies and/or indirection points (such as the HLR in cellular networks and home agent in Mobile IP) have been a common feature of many architectures supporting mobility, including recent proposals for NDN-like architectures. both adopt a proxy-based approach and rely on underlying tunneling or the existence of IP addresses to deliver content. Most recently, Kim proposes the use of an indirection point where mobile content publishers and subscribers can register (content publishers) mobility-related name changes or query (content subscribers) for new names associated with a mobile publisher. presents a protocol for NDN-like architectures for real-time, single-sender-single-receiver scenarios, conjecturing that “… content providers, and their locations are relatively stable. Hence, the mobility problem for the ‘stored contents’ is limited to the scope of user side mobility.” Our measurement results in Section 7 suggest otherwise. Also, our NDN focus here is on in-network name resolution, rather than on the use of proxies or indirection points.

The strategy layer in NDN is a new, key concept that we believe will have far-reaching impact on the forwarding table size, update cost, path stretch, and data/control plane overheads. We quantify and discuss these overheads in Sections 6 and 7 for a simple strategy layer. Investigation of a more sophisticated strategy layer, e.g. a strategy for probing multiple possible forwarding paths to names content, is only now beginning and represents both “important design choices and research challenges”.

**Mobile IP and Indirection.** Much of the analysis of Mobile IP has focused on analyzing proposed enhancements that improve handoff performance (e.g.,) or minimize signaling overhead (e.g.,,). Our goal, instead, is to empirically characterize location update rates as nodes change their point of attachment to the Internet, and the average “distance” from their home network - two key performance considerations for any architecture with a component similar to a home agent. As noted above there are numerous measurement studies of mobility among individual access points and base stations, but none of these studies characterize the rate at which users change their IP address of AS-affiliation, and it is this latter aspect of mobility (not intra-network mobility that is “invisible” outside of that network) that determines home agent location-update rates.

The triangle routing problem in Mobile IP, which results in longer paths between a sender and receiver when routed through an indirection point (i.e., the home agent) is well known, and enhancements to allow direct routing have been proposed. The tradeoff between providing shortest path routes versus the overhead needed to do so (e.g., in routing table size) is characterized by compact routing results. For example, with N flat identifiers, to be within 3x stretch of shortest-path, each router needs to maintain Ω(N) forwarding entries.; for up to 5x stretch, it is Ω(√(N)). These works, that either focus on developing new protocols to minimize path stretch due to indirection, or theoretically characterize path stretch versus table size tradeoffs, address different challenges that our work, which empirically characterizes a mobile node’s distance...
3. EVALUATION METHODOLOGY

In this section, we explain our methodology to evaluate the cost-benefit tradeoffs struck by different location-independent approaches for handling mobility. The metrics of interest are mainly routing update cost, forwarding table size, and path stretch for mobility of devices as well as content.

3.1 Intradomain device mobility

Consider a simple shortest-path routing network as shown in Figure 1(a) and a singly-homed endpoint A that changes its address from say 22.33.44.55 belonging to the 22.33.44.0/24 subnet to 22.33.88.55 belonging to the 22.33.0.0/16 subnet. With name-to-address resolution (indirection routing). A simply needs to update DNS (its home agent) and data traffic can subsequently flow directly (indirectly) to A’s new location. With a purely network-layer approach, some routers such as R may need to update their forwarding behavior in order to continue routing packets correctly to A. Whether R needs to update its forwarding table depends on whether R’s shortest-path forwarding entries for 22.33.44.0/24 or 22.33.0.0/16 point to different output ports. If they do, say to ports 5 and 3 respectively, then R must introduce another entry [22.33.44.55/32, 5] in its forwarding table so that longest-prefix matching continues to route correctly to A. More generally, a router needs to update its forwarding table if an endpoint moves from the longest-matching prefix $P_1$ to a longest-matching prefix $P_2$ in its forwarding table, each pointing to different output ports. We refer to such mobility events as and endpoint A being displaced with respect to router R.

![Figure 2: Example of a device or content displaced w.r.t. a router’s forwarding table.](image)

Displaced content names induce an update at name-based routers by moving across hierarchical name spaces in a manner analogous to endpoints moving across IP address spaces. An example is shown in Figure 1(b), wherein say /20thCenturyFox/StarWars-EpisodeIV moves to /Disney/StarWinds-EpisodeIV because of a distribution rights transfer. Router Q must update its forwarding table if Q maintained different output ports for the corresponding longest-matching prefixes /20thCenturyFox/* and /Disney/*, i.e., if StarWars-EpisodeIV is displaced with respect to Q.

3.2 Interdomain device mobility

From the example above, it is clear that the update cost of mobility depends on at least three factors: (1) the nature of network mobility patterns of endpoints; (2) the physical topology of the network; and (3) route selection policy. The policy used to select routes, e.g., shortest-path routing or BGP-style policy-driven route selection, matters because that is what determines the forwarding table of a router. Unlike shortest-path routing in a single network, wherein it is straightforward to answer whether or not a mobility event causes an update at any given router, it is much harder to answer this question in an interdomain network like the Internet driven primarily by policy routing.

One strawman is to use publicly available intradomain [24] and interdomain [8] topologies and combine them with a simplistic model of Internet routing, namely, prefer customers over peers over providers, then use AS path length to break ties, then use early-exit routing to further break ties, and so on. Unfortunately, combined with the fact that even our knowledge of the Internet’s physical AS-level as well as intradomain topology is incomplete [37], these models serve as a poor substitute for real Internet routing that is messier, e.g., prior studies [33] have found that interdomain routes predicted by such a model had barely 30% predictive accuracy.

Consequently, in this work, we hand-pick a small set of real Internet routers whose RIBs (route information bases) including some route preference metrics are available to us. These RIBs already incorporate the global impact of the Internet’s physical topology and route import and export policy decisions made by other routers. We use these RIBs to derive the corresponding FIBs and ask, in a manner analogous to intradomain routing above, whether or not the router would update its forwarding table in response to a mobility event without having to simulate global Internet routing.

3.3 Interdomain content mobility

Next, we explain in detail our methodology to model the update cost and forwarding table size impact of content mobility events. Unlike mobile devices that today are rarely multihomed, i.e., connected simulta-
through multiple network interfaces, content is frequently multihomed. In particular, popular content served by content distribution networks is commonly served from tens or even hundreds of locations.

We begin with a content mobility trace consisting of content domain names and the set of IP addresses to which they resolve as measured [detailed in §7] from a distributed set of vantage points. Consider a domain $D$, e.g., graphics.nytimes.com, and let $A(D, t)$ denote the set of all IP addresses to which it resolves at time $t$. A content mobility event refers to a change in this set to $A(D, t_2)$ at a future time $t_2 > t_1$. Does this mobility event cause a content router $R$ to update its forwarding behavior? To answer this question, we consider the set of output ports $F(R, D, t_1)$ and $F(R, D, t_2)$ that respectively denote the set of output ports to which router $R$ forwards packets destined to $D$ at times $t_1$ and $t_2$ respectively. We distinguish between two forwarding strategies—best-port forwarding and controlled flooding—that respectively forward packets on at most a single output port (like today) and forward packets to more than one output port. The set of all eligible output ports at $R$ in order to reach $D$ at any time $t$ is the set of output ports corresponding to the set of IP addresses $A(D, t)$ at that time, which is computed using its RIB as above for interdomain device mobility.

For example, as shown in Figure 3 the forwarding entry [travel.yahoo.com, 2] is subsumed by the entry [yahoo.com, 2] as longest-prefix matching obviates storing a separate entry explicitly for the former, but a separate entry is needed for [sports.yahoo.com, 5]. We refer to the forwarding table consisting of the subset of entries in the complete forwarding table that excludes all subsumed entries as the LPM forwarding table.

We define aggregateability at a router as the ratio of the size of the complete forwarding table to the size of the corresponding LPM forwarding table.

### 4. Network Mobility Measurement

In this section, we describe the design of an Android app, AppName, an effort that we undertook motivated both by the intractable or simplistic nature of purely theoretical analyses like above as well as the lack of publicly available data on network mobility (unlike geographic mobility) of mobile devices.

![Figure 4: Screenshot and sample visualization.](image)

AppName is a lean app explicitly designed for measuring the changing IP addresses of Android devices and does little else. The value proposition for potential downloaders is that they get pretty visualizations of their network mobility statistics on a map, but realistically we expect people with a “citizen science” or an analytics bent to be more likely to install the app. The app runs in the background of the user’s Android phone or tablet and attempts to record the public-facing IP address upon each connectivity event. A connectivity event is each time either network interface wakes up and successfully connects to a cellular or WiFi network or disconnects from a network. Upon a connectivity event, the app contacts a server we maintain in order to determine its public-facing IP address, so addresses are logged only if they are usable for Internet connectivity (so automatic connects to WiFi access points blocked by a paywall or authentication page will not be logged unless successful).

The app is designed to be as inconspicuous as possible and is conservative in using battery power and data traffic as follows. First, its event-driven design obviates

<table>
<thead>
<tr>
<th>Prefix</th>
<th>Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>yahoo.com</td>
<td>2</td>
</tr>
<tr>
<td>travel.yahoo.com</td>
<td>2</td>
</tr>
<tr>
<td>sports.yahoo.com</td>
<td>5</td>
</tr>
<tr>
<td>cnn.com</td>
<td>2</td>
</tr>
<tr>
<td>mit.edu</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>

Figure 3: Example of a content forwarding entry subsumed because of longest-prefix matching.

The forwarding table size at a content router depends on two factors: (1) forwarding strategy (as defined above) and (2) aggregateability, a metric that captures the extent to which forwarding tables can be compacted by relying on longest-prefix matching and inherent network locality in the content name space. To formally define this metric, we introduce some notation. Consider a set $S$ of hierarchically organized domain names such as yahoo.com, cnn.com, mit.edu, travel.yahoo.com, etc. and a router $R$ employing some forwarding strategy to route to these domains. For each domain $d \in S$, let $F(d)$ denote the (set of) forwarding output port(s). We refer to the set of forwarding entries $\{d, F(d)\}_{d \in S}$ as the complete forwarding table. Let $d_1 \prec d_2$ mean that $d_1$ is a strict subdomain of $d_2$, e.g., travel.yahoo.com $\prec$ yahoo.com. If $d_1 \prec d_2$ and $F(d_2) = F(d_1)$, then we say that the forwarding entry for $d_1$ is subsumed by $d_2$ with longest-prefix matching.

App Name
polling the network interfaces. Second, except for a single small message to infer its public facing IP address, the app stores all data logs locally until it is both connected to power and WiFi; at this point, it attempts to transfer previously untransferred log files to a postgresql database that we maintain. Each entry logged in the database is in the following format.

\[ device_id | time | ip_addr | net_type | (lat, lng) \cdots \]

\(device_id\) is the hashed device id that is used to track the user while providing limited privacy; \(time\) is the timestamp of the event; \(net_type\) indicates which kind of network is connected to, WiFi or cellular; and \((lat, lng)\) is the user’s geolocation.

The user’s geolocation is recorded only with user permission at install time and is collected only if the GPS is already on and has obtained a recent reading on behalf of some other app, i.e., AppName itself does not consume GPS resources. Users can either visualize their mobility statistics through the app or use their device ID in order to access their data from the app’s website from any device. Except for the hashed device id, we do not maintain any other information that directly identifies a user in our database.

We have acquired 288 users, mostly from the United States, Europe, and South America through word-of-mouth publicity alone. The user recruitment and data collection spanned over 10 months from Mar 2013 to Jan 2014. Because different users downloaded the app at different times and a small fraction uninstalled the app quickly (a learning experience that helped us engineer the resource and data usage optimizations), we removed users who ran the app for less than a day. Most of our analysis is based on daily statistics of network mobility, so our conclusions are unlikely to be biased significantly by the differing measurement period across users.

5. PATH STRETCH VS. UPDATE COST: AN ANALYTIC MODEL

In this section, we develop a simple analytic model to quantify the tradeoff between path stretch and update cost with two goals. The first is expository and helps us better appreciate the fundamental nature of the tradeoffs evaluated empirically in this paper. The second is to suggest that anything beyond simple toy topologies is difficult to analyze theoretically, thereby making a stronger case for empirical analysis.

Our model and results are similar in spirit to theoretical work on compact routing (see [?] for a survey) that has focused on path stretch vs. forwarding table size tradeoffs, but the difference is that we model endpoint mobility and the corresponding update costs. Informally, compact routing results say that in order to achieve small stretch over shortest-path routing in a network with arbitrary (or flat) endpoint identifiers, nearly all routers must maintain an entry for nearly all endpoint identifiers. The question we ask is: in order to achieve small stretch over shortest-path routing, how many routers need to be updated when endpoints move from one router to another?

Intuitively, if every router could be updated upon every mobility event, then the path stretch will be minimum. Also intuitively, if updates are restricted to at most one router (like a home agent) per mobility event, the path stretch could be as high as the diameter of the network as all packets to that mobile endpoint must go through the only router that knows about its whereabouts. We formally model these tradeoffs for several toy topologies and explain one of these in detail below.

5.1 Chain topology

Consider a chain network topology as shown in Figure 5 with routers numbered from 1 to \(n\), and a user \(u\) that randomly hops from one router to another. This mobility can be modeled using a discrete-time Markov process as follows. Let \(X_t\) be a random variable representing \(u\)’s location at time slot \(t\). If the transition probability \(P(X_{t+1} = j | X_t = i) = \frac{1}{n}\), then the steady-state distribution of \(X_t\) is uniform, i.e., \(P(X_t = i) = \frac{1}{n}\).

\textbf{Indirection routing.}

Let \(H(u)\) denote \(u\)’s home agent that keeps track of its current location. We define \(\text{path\_stretch}\) as the hop-count distance from the home agent to an endpoint’s location at time \(t\). If \(H(u)\) were chosen randomly instead (as would be the case in a network where different nodes were equally likely to be homed at any router and moved around randomly), the stretch is derived as follows. Below \(\text{dist}(u,v)\) refers to the hop count distance between \(u\) and \(v\).

\[
E[\text{path\_stretch}_1] = E[\text{dist}(H(u), X_t)]
\]
\[
= \sum_{i=1}^{n} P(H(u) = i) \sum_{j=1}^{n} P(X_t = j) |i - j|
\]
\[
= \frac{1}{n} \sum_{i=1}^{n} \frac{1}{n} \sum_{j=1}^{n} |i - j|
\]
\[
= \frac{1}{n} \sum_{i=1}^{n} \frac{1}{2} i (n + 1 - i)
\]
\[
= \frac{1}{n} \sum_{i=1}^{n} \frac{n^2 - i^2}{2}
\]
\[
= \frac{1}{n} \frac{n(n+1)(2n+1)}{6}
\]
\[
= \frac{n^2 + 3n + 1}{6n}
\]

Thus, with indirection, the expected path stretch is \(\frac{n}{3}\) and the expected update cost is 1 per mobility event.

\textbf{Name-based routing.}

With name-based routing, the path stretch is 0 if we assume that routers are designed to always maintain forwarding tables corresponding to shortest-path routing. What is the update cost (the fraction of routers that must be updated) to achieve this minimal stretch? We
derive it as follows. Suppose each router has three ports, a left (right) port connecting to the leftwise (rightwise) adjacent router, and a local port connected to the local subnet. Then a router \( i \) must update its forwarding table whenever an endpoint either moves from any leftward router to a rightward router or vice-versa, or moves from any router other than \( i \) to \( i \) or vice-versa. Let \( L(u) \) denote the current location of endpoint \( u \). The expected update cost at router \( k \) is

\[
E[\text{update\_cost}] = P(L(u) < k) \cdot P(L(u) \geq k) + P(L(u) = k) \cdot P(L(u) \neq k) + P(L(u) > k) \cdot P_k(L(u) \leq k)
\]

\[
= \frac{1}{n^3} + \frac{n-1}{n} \cdot \frac{n-k+1}{n} + \frac{1}{n} \cdot \frac{n-k}{n} + \frac{1}{n} \cdot \frac{n-k}{n}
\]

\[
= \frac{2}{n^3} + 3 \approx \frac{1}{3}
\]

Thus, for name-based routing, the expected update cost per mobility event is \( 1/3 \) and the path stretch is 0.

**Summary of results.**

We have similarly quantified the update cost vs. path stretch tradeoff for other topologies continuing to assume a simplistic random mobility model. The proofs are deferred to an anonymized techreport [7]. Note that we have omitted the analysis for a DNS-based based approach as the data path stretch is 0 (ignoring a constant lookup overhead in the connection initiation step) and the expected update cost is simply \( O(1) \) (to the DNS), irrespective of topology.

<table>
<thead>
<tr>
<th>Topology</th>
<th>Indirection</th>
<th>Name-based routing</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>stretch</td>
<td>update cost</td>
</tr>
<tr>
<td>chain</td>
<td>( n/3 )</td>
<td>1/3</td>
</tr>
<tr>
<td>clique</td>
<td>1</td>
<td>1/3</td>
</tr>
<tr>
<td>binary tree</td>
<td>( 2 \log_2 n )</td>
<td>1/3</td>
</tr>
<tr>
<td>star</td>
<td>2</td>
<td>1/3</td>
</tr>
</tbody>
</table>

Table 1: Path stretch vs. update cost results.

6. DEVICE MOBILITY

In this section, we combine the measured AppName traces in [3] with the methodology in [8] in order to evaluate the cost-benefit tradeoffs of different approaches to handle mobility of devices across networks. We begin with an analysis of the extent of network mobility across devices using the AppName data.

6.1 Extent of device mobility across networks

Figure [6] shows the distribution across users of the average number of distinct network locations per day visited by a user [2]. The trace consists of a total of 288 users, but as different users downloaded the app at different times, the total number of days over which the average is computed is different for different users, and each user included in this trace is present for a minimum of at least one day in the trace. The figure shows

that the median user visits two ASes, two IP prefixes, and three IP addresses per day. This observation is consistent with the expectation that users typically move across a cellular, home, and work address in the course of a day.

![Figure 6: Average number of distinct network locations per day visited by users.](image)

![Figure 7: Average number of transitions across network locations per day made by users.](image)

Figure [7] also shows the distribution of the average number of transitions across network locations per day by a user. The number of AS transitions shows a lot more variation compared to the number of distinct ASes in Figure [6] which is because a user can switch many times between a small number of distinct ASes. The number of transitions depends upon the user’s physical mobility, network availability and outage patterns, as well as behavioral patterns, e.g., some users may prefer to use WiFi for some apps but use LTE for others or make these choices depending on current network qual-

---

2 We use the terms user and device interchangeably.
ity. The maximal and minimal number of average AS transitions per day across all users are 18.4 and 0.73 respectively. The median user makes one AS transition and two IP address transitions.

6.2 Update cost of device mobility

In order to provide the abstraction of location-independent communication, a device must update its changing network addresses somewhere, either in DNS, or at its home agent(s), or at routers. For the first two cases, the update cost is straightforward and directly corresponds to the rate of address transitions as shown above. For the third case, we estimate the update cost using the methodology described in §3.1 and §3.2 respectively.

6.2.1 Using Routeviews data

To this end, we use RIBs from 12 BGP-speaking Routeviews routers [6]. The set of routers includes four in Oregon (labeled Oregon-1 to Oregon-4) and one each in Virginia, California, Georgia, Mauritius, London, Tokyo, Sydney and Sao Paulo. A single entry in a router’s RIB lists several attributes of a single inter-domain route towards a given prefix. Typically, there are several routes to any given prefix and the set of all prefixes covers the entire IP address space.

<table>
<thead>
<tr>
<th>ip_prefix</th>
<th>next_hop</th>
<th>local_preference</th>
<th>metric</th>
<th>AS path</th>
<th>···</th>
</tr>
</thead>
</table>

To construct the forwarding table (or FIB) from a router’s RIB, we need to compute a rank ordering of all of the routes for a single prefix. We apply the following rules in priority order based on typical BGP policies and the priority order suggested on [6].

1. A route with a higher local_preference value is preferred. As local preference values are not available for most of the router dumps, we simply rely on the customer > peer > provider policy using standard techniques for inferring AS relationships [21].

2. A route with a shorter AS path is preferred.

3. A route with a smaller MED value is preferred.

Note that the above rules approximate typical BGP policies. The numerical value of the local_preference is uniformly 0 in these RIBs, so we use AS relationships instead in the first rule. The RIBs also do not have sufficient information to implement early- or late-exit policies (typically of higher priority than multi-exit discriminator values).

6.2.2 Update cost of name-based routing

In order to determine if the movement of a user from one address to another results in a change in a router’s forwarding behavior, we need to determine whether the mobility event induces a change in the output port corresponding to the highest ranked route for the user’s address. We use the next_hop AS path attribute as a proxy for the output port, implicitly assuming that the forwarding output port changes if and only if the next_hop attribute changes. In practice, different next_hop addresses can correspond to the same output port, and the same next_hop can correspond to different output ports at different times because of changes in intradomain routing, so our simplifying assumption may under- or over-estimate the actual update cost.

Figure 8 shows the update cost at all of the routers using the RIB data published on Nov 30, 2013. Using different dates does not qualitatively change our conclusions [7]. The update cost is shown as a fraction of all mobility events that induce a forwarding update at the router. The results show that up to 8% of mobility events can induce an update at some routers. The routers at Mauritius and Tokyo experience hardly any updates, which is unsurprising as most of our users are located in the USA, Europe, and South America, so their mobility is less likely to impact distant routers.

![Figure 8: Fraction of device mobility events inducing a router update.](image-url)

**Back-of-the-envelope calculations.**

Combining the above observation with the results in the previous section, we can arrive at a crude estimate of the absolute rate of updates induced at routers because of user mobility today. For example, if 2 billion smartphones change network addresses three (seven) times per day like our median (mean) user, and 8% of these mobility events induce an update at a router, the corresponding update rate is 5.5K/sec (11K/sec). These numbers are prohibitively high for even high-end routers today. Although it is possible to redesign router control planes to handle such high update rates using more compute power or using a logically-centralized control architecture [27], it is difficult to justify this computation cost and the bandwidth cost of propagating these
updates to a large number of Internet routers. In comparison, it is straightforward to handle this aggregate load using a small number of DNS servers or by distributing this aggregate load across a large number of home agents.

6.3 Data path stretch with device mobility

The update cost analysis above induces no data path stretch (over underlying Internet routing) for name-based routing or for a DNS-based approach. However, indirect routing inflates the data path because of triangle routing via the home agent. Next, we attempt to quantify this path stretch overhead.

6.3.1 Dominant location

We introduce the notion of a dominant location, i.e., the network location where a user spends the largest fraction of time compared to all other locations in the course of a single day. Figure 9 shows the distribution across all days and all users of the percentage of time spent in the dominant location. For example, the plot shows that over 40% of users spend around 75% of their day at the dominant IP address and around 85% of their day at the dominant AS.

The dominant location is a natural candidate for a home agent in an indirect routing architecture. In order to compute path stretch, we need to determine $C \rightarrow H \rightarrow M/C \rightarrow M$, where $C \rightarrow H \rightarrow M$ is the sum of the network latency from a correspondent C to the home agent H and that from H to the mobile M, and $C \rightarrow M$ is the network latency of routing directly from C to M. We do not have a dataset of correspondents initiating communication with mobile devices (because it is largely not possible today to initiate communication with mobile devices), so instead we seek to simply quantify the displacement of mobile users from their home agents in network distance, i.e., the latency of the path from $H \rightarrow M$.

6.3.2 Path stretch of indirect routing

In order to determine the network distance from a user’s dominant location (or home) to their current location in the AppName trace, we rely on iPlane, a system that uses daily traceroute measurements from a large number of distributed vantage points to stub networks in order to predict the route (and its latency) between an arbitrary pair of IP addresses. Although using iPlane is convenient, it comes with two sever caveats for our analysis. First, iPlane returns valid responses for only 5% of the dominant and current IP address pairs in our trace; this is because it is designed to return responses only if it has sufficient traceroutes that enable it to construct a predicted route using segments of measured routes. Second, even when iPlane returns a response, the predicted route may be inaccurate.

Figure 9: The CDF of time that each user spends in the dominant location

Figure 10 shows the distribution of network latencies across the dominant-to-current IP address pairs for which iPlane returned a predicted route. The median displacement delay from the dominant location is around 50ms and the corresponding AS hop count is 4[7]. Recognizing the limitations of the estimates obtained using iPlane, we use a different technique to estimate a lower bound on the AS hop count of the displacement from home. We compute the length of the shortest AS path from from the home to the current location using the Internet’s AS-level physical topology [8] (even if this route may not be present in the AS-level routing topology). The median AS hop count of this shortest AS path is 2, suggesting that mobile users typically wander two or more ASes away from the home AS.

Figure. 10 shows the distribution of the physical distance from the home location to the visited locations across 147 users who allowed AppName to collect their location information. The median of distribution is 1km, and about 80% physical location mobility events happened in the radius of 10km around the user’s home. The maximal distance is about 4500km.

7. CONTENT MOBILITY

In this section, we evaluate the cost-benefit tradeoffs in terms of update cost, forwarding table size, and path stretch for content mobility. We begin by describing the procedure used to measure content mobility today.

7.1 Content mobility measurement

We begin with two sets of content domain names: a popular set and an unpopular set. The former is the set of the top 500 domains ranked according to popularity by Alexa [4] and the set of all of their subdo-
mains. The latter is the least popular 500 domains and their subdomains in a list of the top 1 million domain names also ranked by popularity. We explicitly obtain a list of subdomains because Alexa ranks “websites” or top-level enterprise domains, e.g., nytimes.com or yahoo.com, but not their subdomains like graphics.nytimes.com or travel.yahoo.com. More importantly, the distribution of popular, bulky content that is ideally suited to name-based routing techniques is often outsourced to CDNs, and a common technique to achieve this is by CNAME-aliasing subdomains, e.g., the domain graphics.nytimes.com is aliased to the canonical name static.nytimes.com.edgesuite.net that is in turn aliased to a1158.g1.akamai.net that finally gets dynamically resolved to one or more IP addresses close to the querying client or its local name server.

The dynamic nature of resolution of domain names to IP addresses (either because they are resolved by a CDN delegate in a locality-aware manner or because of DNS-based load balancing employed by the origin server) means that any single vantage point will see only a subset of all IP addresses from where a domain’s content may be potentially served. Our methodology to assess content mobility (in §3.3) relies on monitoring any changes to the set of all IP addresses corresponding to a domain name. This methodology implicitly assumes that a purely name-based routing network will announce a content domain name from all of the locations (including CDN locations) where it resides today.

In order to measure a reasonably complete set of IP addresses to which each domain name maps, we conduct a measurement distributed across 74 Planetlab nodes that are chosen from as many different countries as possible and all continents (except Africa where Planetlab nodes were unavailable). We conducted the measurements for a three day period from Jan 22 to Jan 25, 2014. Each node resolves each domain name once every hour, thereby observing a subset of the domain’s IP addresses at that time. A central controller node collects measurements obtained from all of the vantage points and merges them in time so that for each domain name for each hour, the set of IP addresses is the union of all IP addresses obtained from all vantage points for that domain. As the measurement interval is just once per hour, precise time synchronization is not necessary. Note that the measurement is done just once per hour per domain because our list of subdomains corresponding to the 500 most popular Alexa domains contains 13,084 entries, so a much higher rate would overwhelm some nodes and/or trigger security alarms.

### 7.2 Update cost of content mobility

![Figure 11: The average number of transitions for popular content mobility events](image)

Figure 11 shows the extent of daily mobility of popular content (i.e., the 13084 subdomains obtained from the most popular 500 domains). The median number of changes in the set of IP addresses per day is 2 (a number that happens to be coincidentally similar to that for device mobility) and the maximum is bounded at 24 because of our hourly measurement procedure.

Figure 12 shows the fraction of mobility events involving popular content inducing an update to the forwarding table to each of the twelve routers. The plot shows that up to 12% of content mobility events can induce an update at some routers when controlled flooding (or forwarding on all ports matching any of the domain’s IP addresses) is used. However, at most 6% of the mobility events induce an update at any of the routers in our dataset when best-port forwarding is used. The reason is that although there may be some flux in the set of addresses corresponding to a domain name, the address that is the closest to any given router rarely changes because most of the addresses in the set remain unchanged, i.e., unlike devices that seemingly jump from one address to a completely unrelated address seemingly randomly, content locations do not jump around arbitrarily.

Figure 13 shows the corresponding result for unpopular content or the least popular 500 domains and their
subdomains with a popularity rank of near about one million. The update cost for unpopular content is dramatically lower than that for popular content; at most 1.5% of updates induce an update at any of the routers even with controlled flooding. With best-port forwarding, none of the routers experience any update during the course of our measurement period at all! This result is not surprising as unpopular content is unlikely to be delegated to CDNs and is probably served only from a small number of network locations that rarely change; these multiple locations if at all are chosen mainly for fault-tolerance or load balancing purposes rather than for proximity to clients, so they rarely change.

Back-of-the-envelope calculations.
Performing a calculation similar to that at the end of §6, if we assume 1B content domain names (noting that DNS has ≈ 150M domains), an update rate of 2/day, and a 0.5% likelihood of inducing an update at a router, the router would receive at most 100 updates/sec. Furthermore, for the vast majority of long-tail domains ranked below 1M, the update cost is likely to be even lower even if controlled flooding is used as the forwarding strategy. Finally, with best-port forwarding, the router update cost of content mobility for most content is near-zero.

8. DISCUSSION
In this section, we discuss the limitations of our work, caveats associated with our findings, open questions, and future work.

The number of users (288) contributing to our device mobility dataset is relatively small and drawn primarily from the US, Europe and Brazil; similarly our content mobility measurements were taken from a relatively small number (74) of Planetlab nodes, with named content from the 500 most/least popular Alexa domains. Thus our data set is not (nor is it meant to be) representative of the global Internet. However, given the lack of existing datasets characterizing device and content mobility, we believe our measurements are an important and necessary first step in developing meaningful mobility datasets with which architectural alternatives may be compared. An alternative might have been to develop an abstract model (e.g., the equivalent of a ran-
dom waypoint model [52] of device or content mobility among networks), but then this abstract model would itself need measurements such as those presented here as part of the validation process.

Our characterization of content mobility by measuring the change in resolved IP addresses associated with domain names implicitly assumes a purely name-based routing network that announces content domain name from all of the locations (including CDN locations) where it resides today. That certainly need not be (and likely would not be) the case in the future. But in the absence of a measurable real-world deployment at even modest scale or a proposal for how future content locations might be chosen and changed, we believe our characterization is appropriate.

Our evaluation methodology itself suffers from some limitations. First, we have evaluated three pure strategies for location-independence but not the infinite number of possible combinations of these strategies in a network architecture. Second, network architecture itself is indeed part science and part art. Not everything may be easily quantifiable. What is easily quantifiable may not be the most relevant concern. Nevertheless, our position is that pushing the envelope of what is quantifiable is valuable for scientific discourse, and our work is a first step towards that goal.

Lastly, although it might appear “obvious” in retrospect that mobility of a device or popular content may cause significant overhead in a pure name-based routing we believe it is important to quantify that impact. Our research has been the first to do so. Some of the assumptions underlying our evaluation (e.g., that interest forwarding is done via controlled flooding) and the consequent results also suggest that this impact might be lessened by more sophisticated strategy-layer approaches, making this an important area for future research [51].

9. CONCLUSIONS

The intellectual pursuit of a location-independent communication abstraction has long intrigued networking researchers, and has in no small part influenced the design of several clean-slate Internet architectures. Our work is motivated by the observation that, despite many architectures sharing this common goal, there has been little prior work on quantitatively comparing the different cost-benefit tradeoffs struck by different architectures in accomplishing this goal.

As a first step towards addressing this gap, we have developed a quantitative methodology to empirically evaluate three different puristic approaches that drive the designs of a number of location-independent network architectures. We combine this methodology with measured traces of device mobility and content mobility on the Internet using realistic physical and routing topologies. Based on measured network mobility patterns of hundreds of devices of AppName, an Android app we developed explicitly for this goal, and hundreds of content domains including those delegated to content distribution networks, we find that pure name-based routing induces a prohibitively high update cost at routers because of device mobility, but induces a far lower update cost in conjunction with simple forwarding strategies for most of today’s content that happens to exhibit high locality. Taken together, our results suggest that recent proposals for information-centric networking in their puristic form are better suited for content distribution alone, and may need to be augmented with addressing-based approaches like DNS or Mobile IP on order to serve as a general-purpose replacement for the TCP/IP Internet.

10. REFERENCES
