

図 3 提案手法のアーキテクチャ

3. 提案手法

エッジコンピューティング基盤のためのオブジェクトストアの構成手法を提案する。図 3 は、提案手法のアーキテクチャである。提案手法では、各エッジノードに対してドメイン名を付与し、それに基づいて Skip Graph [11] を構築する。また、同一ドメイン配下のエッジノードは、それらでオブジェクトストアを提供する。これにより、クライアントが同一ドメイン内のオブジェクトを取得する場合、他のドメインに属するエッジノードと通信する必要がなくなる。他のドメインに属するオブジェクトを探索する場合でも、ドメインをまたぐ通信は必要最小限に抑えられる。

提案手法は、ドメインに基づくオブジェクトの操作をクライアントに提供する。クライアントは、ドメイン名とオブジェクト名を指定して (1) オブジェクトを格納する操作と (2) オブジェクトを取得する操作、(3) オブジェクトを削除する操作の 3 つの操作を行うことができる。例えば、ドメイン名を `jp.example`、オブジェクト名を `example-object` とした場合、クライアントは `jp.example/example-object` に対して (1)–(3) の操作を実行可能である。

以降では、提案手法のオブジェクトストアをどのように実現するかを詳細に述べる。

3.1 オブジェクトストアの構築手法

3.1.1 ドメイン名の付与

各エッジノードにドメイン名を付与する。ドメイン名は、ドットで区切られる以下の形式の文字列とする。

`first.second.last`

同一ドメイン配下のエッジノードでオブジェクトストアを提供するために、上記の `last` の部分はエッジノードの識別子として使用することとし、ドメイン内でユニークな文字列とする。また、同一サイト内のエッジノードは、`last` を除いた部分文字列をドメイン名として共有していることとする。これは、サイトにドメインが対応することを意味する。複数のサイトを包含するサイトをエリアとする。エリアはドメインの階層構造に相当するものである。

ドメイン名の決定方法としては、

- 地理的位置情報に基づいて決定する方法
 例 `jp.nara.ikoma`, `jp.osaka.suita`, ..., etc.
- 組織構造に基づいて決定する方法
 例 `jp.naist.is`, `jp.naist.bs`, `jp.naist.ms`,
`jp.ac.jaist`, ..., etc.

の 2 つが考えられる。

3.1.2 構造化オーバーレイネットワークの構築

全サイトのすべてのエッジノードにより、前節で付与したドメイン名に基づく Skip Graph を構築する。このとき、エッジノード間の順序関係にはドメイン名の辞書式順序を採用する。辞書式順序とは、以下で定義される順序関係である。

定義 1 (辞書式順序). 全順序集合を Σ とする。直積集合 $\Sigma \times \Sigma$ 上の辞書式順序 \leq_{dic} は次のように定義される。

$$(a, b) \leq_{\text{dic}} (a', b') \Leftrightarrow (a < a') \cup (a = a' \cap b \leq b'). \quad (1)$$

ここで、 $a, b \in \Sigma$ とする。

アルファベットを $\Sigma_{\text{mv}} = \{0, 1\}$ とする。 $w_1, w_2, \dots \in \Sigma_{\text{mv}}$ としたとき、 Σ_{mv} 上の語を $w = w_1w_2\dots$ と表記する。また、語 w の長さを $|w|$ で表す。特に、 $|w| = 0$ となる語 w を ε と書く。 $|w| < \infty$ となる w 全体からなる集合を Σ_{mv}^* とし、 $|w| = \infty$ となる w 全体からなる集合を $\Sigma_{\text{mv}}^\omega$ とする。

ノード x は、いくつかの双方向連結リスト S_w に属する。ノード x が属する双方向連結リストは、ランダムなメンバーシップベクタ $\text{mv}(x) \in \Sigma_{\text{mv}}^\omega$ により決定される。双方向連結リスト S_w は、 w が $\text{mv}(x)$ の接頭語となるようなノード x がドメイン名の辞書式順序 \leq_{dic} の昇順で並んだリストである。 S_ε はすべての要素 x が昇順で並んだリストとなる。以上の手順で双方向連結リストの族 $\{S_w\}$ 、すなわち Skip Graph を構築する。双方向連結リストのリンクは、オーバーレイネットワークにおける辺を意味する。

Skip Graph を採用することによって、ドメインを探索する際のトラフィックが最適化され、必要最小限のドメイン間で探索が実現できるようになる。例えば、ドメイン

jp.naist.is からドメイン jp.naist.bs の探索を行う場合、探索は、ドメイン jp.naist の配下のエッジノードのみを経由して実現される。また、Skip Graph は、探索に必要なオーバーレイネットワーク上のホップ数をノード数に対して対数オーダーに抑えられるという特徴があるため、膨大なサイト数が広域に分散するエッジコンピューティング環境において Skip Graph の採用によるメリットは大きい。

3.1.3 オブジェクトストアの構成

同一ドメイン内のエッジノードをストレージノードとして分散ハッシュ表を構成し、サイト内のオブジェクトストアとして利用する。ここでは、実用性の観点から分散ハッシュ表として Kademia [8] を採用する。クライアントが同一ドメイン内のオブジェクトを探索する場合、リクエストを受け取ったエッジノードは分散ハッシュ表を参照して対象のオブジェクトを探索する。クライアントが他のドメイン内のオブジェクトを探索する場合、リクエストを受け取ったエッジノードは Skip Graph を探索してドメインの探索を完了後、探索したドメインの分散ハッシュ表を参照して対象のオブジェクトを探索する。

4. 考察

オブジェクトの挿入や取得時のトラフィックの流れを観察し、提案手法について考察する。

図 4 は、提案手法により構成したオブジェクトストアを用いてオブジェクトの探索を行なったときのトラフィックの流れを示すシーケンス図である。図 2 (a) は、ローカルサイトのストレージノードにオブジェクトの格納を行う場合のトラフィックの流れである。クライアントは、ローカルサイトからストレージノードを一つ選択し、そのノードを介して分散ハッシュ表にオブジェクトを挿入する。図 4 (b) は、ローカルサイトのストレージノードからオブジェクトの取得を行う場合のトラフィックの流れである。クライアントは、ローカルサイトからストレージノードを一つ選択し、そのノードを介して分散ハッシュ表にアクセスしてオブジェクトを取得する。図 4 (c) は、リモートサイトのストレージノードからオブジェクトの取得を行う場合のトラフィックの流れである。クライアントは、ローカルサイトからストレージノードを一つ選択し、そのノードを介して Skip Graph にアクセスして指定されたドメインに属するストレージノードを探索する。次に、探索したストレージノードは、指定されたドメインの分散ハッシュ表から目的のオブジェクトを探索し、探索結果をクライアントに返す。

提案手法は、オブジェクトの格納場所を分散ハッシュ表を用いずドメイン名に基づいて管理している。以上からわかるとおり、ドメイン名に基づいた Skip Graph は、ドメインをまたぐトラフィックを必要最小限に抑えることができるという特徴がある。そのため、ネットワークの分断時

やサービスの停止時であってもシステム全体への影響が局所的となる。

5. まとめ

本論文では、エッジコンピューティング基盤のためのオブジェクトストアの構成手法を提案した。提案手法により構成されるオブジェクトストアは、サイトに相当するドメインという概念に基づいて構造化オーバーレイネットワークを構築することにより、サイト間をまたぐトラフィックを最適化する。これにより、オブジェクトの格納や取得にあたって、そのオブジェクトがどのエッジノード上に存在するかというメタデータを管理する必要がなくなる。他にも、構築される構造化オーバーレイネットワークは、パスやオブジェクトに局所性を有するためサービスやネットワークの分断時におけるストレージサービスへの影響を局所的に抑えることができる。また、オブジェクトの操作を行う際のトラフィックのシーケンス図を作成し、提案手法が上述の性質を満たすことを確認した。

今後は、提案手法を実際に動作させ、性能の調査を行う予定である。具体的には、OpenStack 上にノードを複数台立ち上げてエッジコンピューティング環境を模した仮想的なネットワークを構築し、その環境の中でワークロードにより、オブジェクトの操作に関する I/O 性能や遅延時間を計測する予定である。

参考文献

- [1] Gartner: Gartner Says 8.4 Billion Connected "Things" Will Be in Use in 2017, Up 31 Percent From 2016, Gartner (online), available from <https://www.gartner.com/newsroom/id/3598917> (accessed 2017-02-07).
- [2] Botta, A., de Donato, W., Persico, V. and Pescap, A.: Integration of Cloud computing and Internet of Things: A survey, *Future Generation Computer Systems*, Vol. 56, pp. 684 – 700 (online), DOI: <https://doi.org/10.1016/j.future.2015.09.021> (2016).
- [3] ETSI: ETSI - Multi-access Edge Computing, ETSI (online), available from <http://www.etsi.org/technologies-clusters/technologies/multi-access-edge-computing> (accessed 2017-02-07).
- [4] Bonomi, F., Milito, R., Zhu, J. and Addepalli, S.: Fog Computing and Its Role in the Internet of Things, *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing*, MCC '12, New York, NY, USA, ACM, pp. 13–16 (online), DOI: 10.1145/2342509.2342513 (2012).
- [5] Confais, B., Lebre, A. and Parrein, B.: Performance Analysis of Object Store Systems in a Fog/Edge Computing Infrastructures, *2016 IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*, pp. 294–301 (online), DOI: 10.1109/CloudCom.2016.0055 (2016).
- [6] Confais, B., Lebre, A. and Parrein, B.: An Object Store Service for a Fog/Edge Computing In-

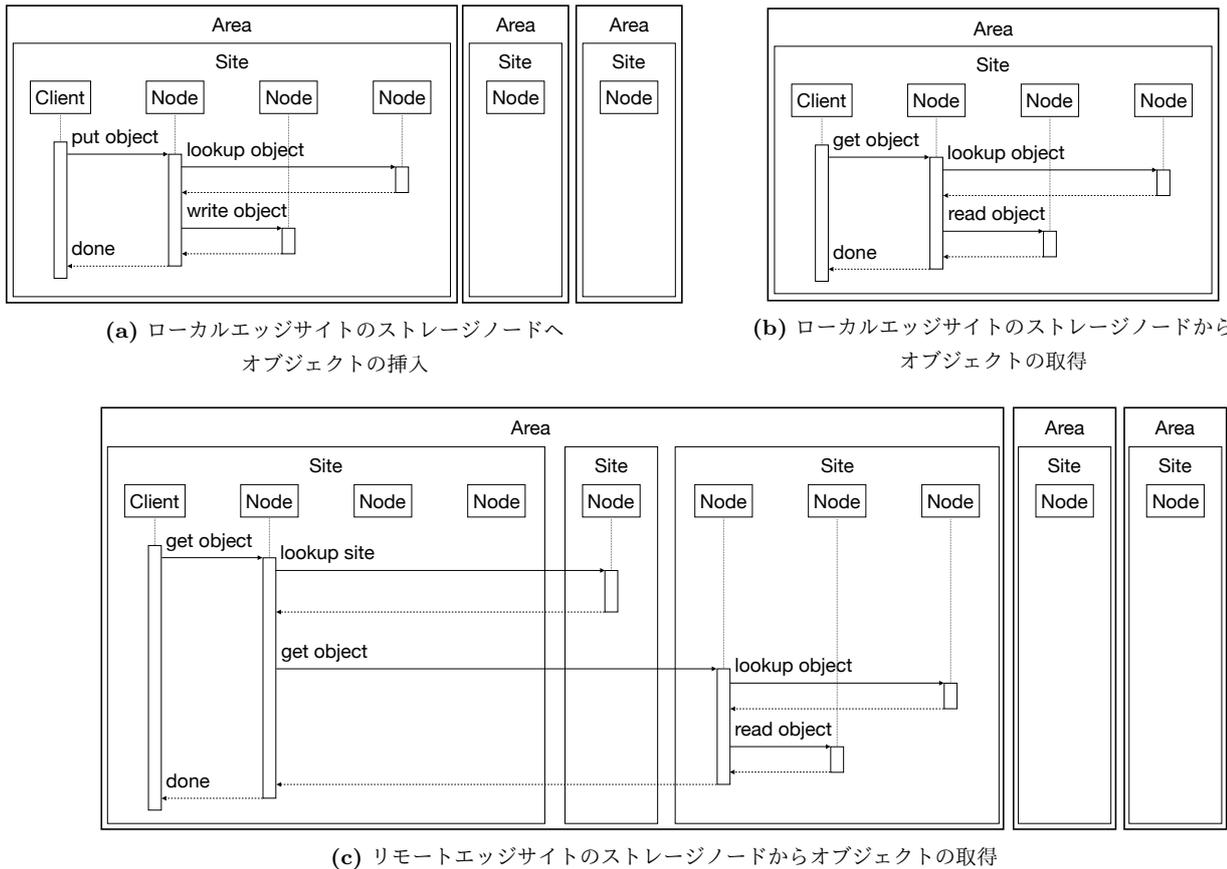


図 4 提案手法のオブジェクトストアにおけるオブジェクトの挿入・取得を行う際の
 トラフィックの流れ

rastructure Based on IPFS and a Scale-Out NAS, *2017 IEEE 1st International Conference on Fog and Edge Computing (ICFEC)*, pp. 41–50 (online), DOI: 10.1109/ICFEC.2017.13 (2017).

[7] Stoica, I., Morris, R., Liben-Nowell, D., Karger, D. R., Kaashoek, M. F., Dabek, F. and Balakrishnan, H.: Chord: A Scalable Peer-to-peer Lookup Protocol for Internet Applications, *IEEE/ACM Trans. Netw.*, Vol. 11, No. 1, pp. 17–32 (online), DOI: 10.1109/TNET.2002.808407 (2003).

[8] Maymounkov, P. and Mazières, D.: Kademia: A Peer-to-Peer Information System Based on the XOR Metric, *Peer-to-Peer Systems* (Druschel, P., Kaashoek, F. and Rowstron, A., eds.), Berlin, Heidelberg, Springer Berlin Heidelberg, pp. 53–65 (2002).

[9] Rowstron, A. and Druschel, P.: Pastry: Scalable, Decentralized Object Location, and Routing for Large-Scale Peer-to-Peer Systems, *Middleware 2001* (Guerraoui, R., ed.), Berlin, Heidelberg, Springer Berlin Heidelberg, pp. 329–350 (2001).

[10] Zhao, B. Y., Huang, L., Stribling, J., Rhea, S. C., Joseph, A. D. and Kubiatowicz, J. D.: Tapestry: a resilient global-scale overlay for service deployment, *IEEE Journal on Selected Areas in Communications*, Vol. 22, No. 1, pp. 41–53 (online), DOI: 10.1109/JSAC.2003.818784 (2004).

[11] Aspnes, J. and Shah, G.: Skip Graphs, *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '03, Philadelphia, PA, USA, Society for Industrial and Applied Mathematics, pp. 384–393 (online), available from (<http://dl.acm.org/citation.cfm?id=644108.644170>) (2003).

[12] Harvey, N. J. A., Jones, M. B., Saroiu, S., Theimer, M. and Wolman, A.: SkipNet: A Scalable Overlay Network with Practical Locality Properties, *Proceedings of the 4th Conference on USENIX Symposium on Internet Technologies and Systems - Volume 4*, USITS'03, Berkeley, CA, USA, USENIX Association, pp. 9–9 (online), available from (<http://dl.acm.org/citation.cfm?id=1251460.1251469>) (2003).