

JANOG58 MATSUYAMA

2026年7月15日(水)

国内でCDN作ってる人集まれ！ ～国内のCDN事業者の実際の話～



株式会社Jストリーム 高見澤信弘

はじめに

▶ このセッションのテーマ

- インターネット上の重要なインフラであるCDNは、実際どうやって作られているの？
- 各社CDNの特徴を紹介いただき、各社の違いや得意分野からを知っていただきたい
- CDNを語ってもらい、作る楽しさを感じてもらいたい、議論に参加していただけると嬉しい

▶ 全体の流れ

- CDNにおけるキャッシュの動作を簡単に紹介
- 各社のCDNのシステムの全体像・成り立ちなどの紹介
- パネルディスカッション：会場からの質問も受けながら
 - 各社CDNの特徴や得意分野を語っていただき違いを知る
 - CDNのコア技術であるキャッシュについての本当のところ
 - CDNを作っていて楽しい！と思うところはどこか？

パネリストのご紹介



JOCDN株式会社
岡 淳一



さくらインターネット
株式会社
山本 真史



合同会社
レッドボックス
小川 勝久



株式会社Jストリーム
高見澤信弘

自己紹介

- ▶ 名前：高見澤信弘
- ▶ 出身地：山形県天童市
- ▶ 所属：株式会社 J ストリーム (AS24253)
 - 新卒で J ストリームへ入社
 - エンジニアリング推進室 & プロダクト企画部（アーキテクト）
- ▶ お仕事
 - CDN(Content Delivery Network)の企画、構築
 - ネットワーク企画
- ▶ 好きなもの
 - ロードバランサー → お家にBIG-IP
 - おうち19インチラック勢
- 活動
 - IPoE協議会 IPv6地理情報共有推進委員会 幹事
 - 海賊版対策実務者意見交換会 海賊版対策技術検証チーム(WG) メンバー



Jストリームとは

もっと素敵な伝え方を。



www.stream.co.jp

株式会社Jストリーム

Jストリームは1997年の設立以来、動画配信を主軸として事業展開を続けております。
自社で保有・運営する独自のコンテンツ配信ネットワーク（CDN）を活用した動画配信に加え、
これまで積み上げてきたノウハウを活かした動画の企画・制作・運用から、Webサイト制作、
システム開発、動画広告による収益化支援まで、総合的なサービスとソリューションを提供し、
企業のマーケティングやコンテンツビジネスなどを支援しております。

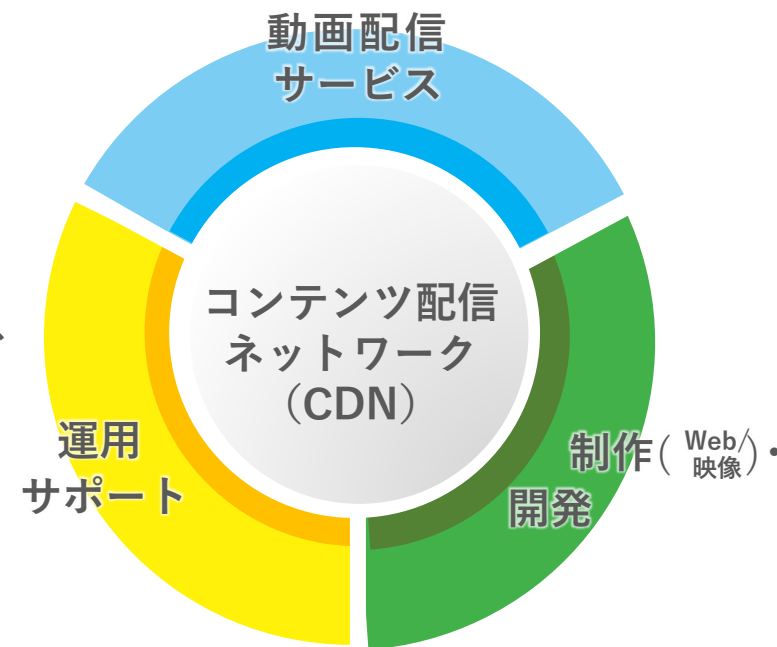
設立 1997年5月

証券コード 4308（東京証券取引所 グロース市場）

本社：105-0014 東京都港区芝二丁目5-6 芝256スクエアビル6階

西日本オフィス：530-0003 大阪府大阪市北区堂島2-4-27 JRWD堂島タワー5階

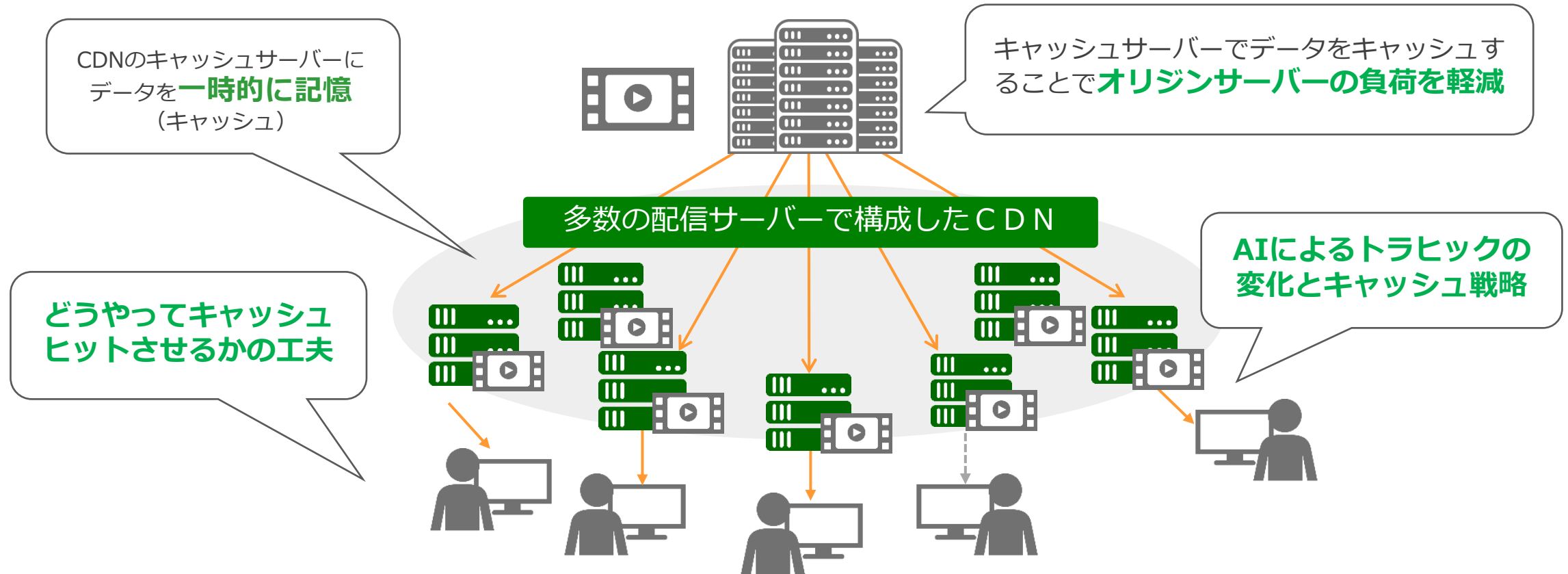
資本金 21億8,237万円（2023年12月末現在）



CDN(Content Delivery Network)とは

▶ CDNとはデータを効率よく最適に配信する仕組み

- CDNはContent Delivery Networkの略で、多数のキャッシュサーバーで構築されたネットワークのこと。コンテンツをCDNが一時的に保存(キャッシュ)し、オリジンサーバーにかわってエンドユーザーへ配信する。



キャッシュ(Cache)とは？

- ▶ 一度利用したデータなどを一時的に保存し、次回以降のアクセス時に保存したデータを利用することで処理を高速化する仕組み
 - キャッシュは、よりユーザーの近くに配置されている
- ▶ キャッシュを利用する場面はCDNだけではなく、様々な場面で利用される
 - Webブラウザでの表示高速化
 - CPUの性能向上やルーターの経路情報の検索 などなど

キャッシュヒットさせないと
CDNを使うメリットがない
(性能の向上は見込めない)

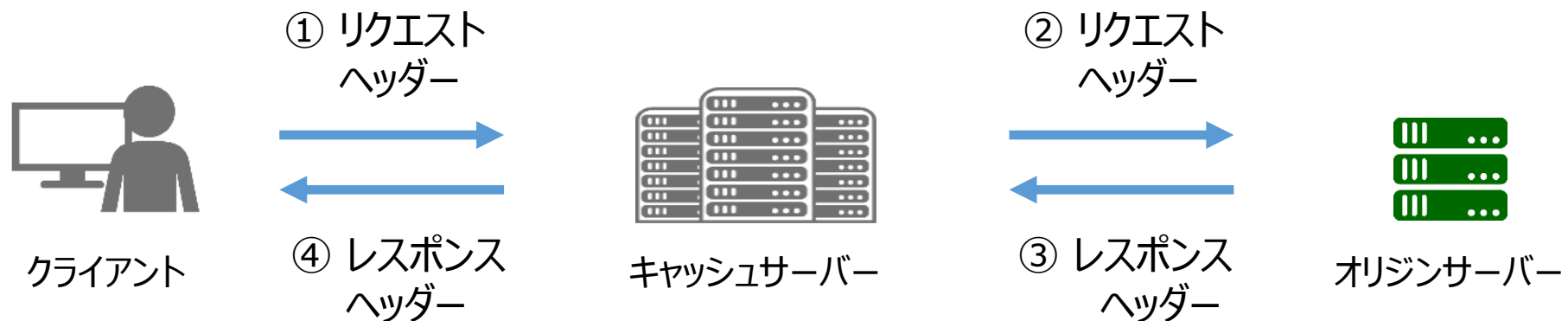
2回目の以降のアクセス時の処理の流れ



実際のCDNの動きに当てはめると

▶ キャッシュの許可/不許可やキャッシュする時間の設定

- オリジンサーバーからのレスポンスヘッダーを使ってキャッシュサーバーに指示
- キャッシュサーバーの設定として指定



直接キャッシュサーバーで設定も可能
この場合は、API/管理画面などで設定

③のオリジンサーバーからのレスポンスヘッダで設定

- Cache-Control no-cache → キャッシュ配信を不許可
- Cache-Control max-age 300 → 300秒間はキャッシュから配信

WebCDNと動画CDNの違い

▶ 動的ファイルの扱い と オブジェクトの大きさ

- ラージオブジェクト=動画・画像 / スモールオブジェクト=Web

▶ キャッシュサーバーから見た動画ファイルとWebページ(動的ファイル)の違い

- 動画ファイル : URLが一意に決まる

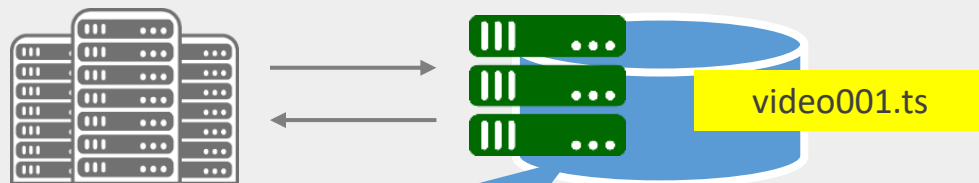
- `http://www.example.com/video/video001.ts`
- CDN側のキャッシュも一意に決まる(1 ファイルとなる)

- Webページ(動的ファイル) : URLが一意に決まらない(複数存在する)

- `http://www.example.com/index.php?p=123&user_id=5432`
- `http://www.example.com/index.php?p=123&user_id=5431`
- `http://www.example.com/index.php?p=123&user_id=5430` ※“?”以降をクエリストリングと呼ぶ

クエリストリングやクッキーなどを
キャッシュキーとしキャッシュファイル
を生成することで、動的ファイルで
もキャッシュヒットさせる

動画ファイル



キャッシュファイルが一意に確定
※m3u8にはクエリがいっぱいついて
くる場合もある。。。

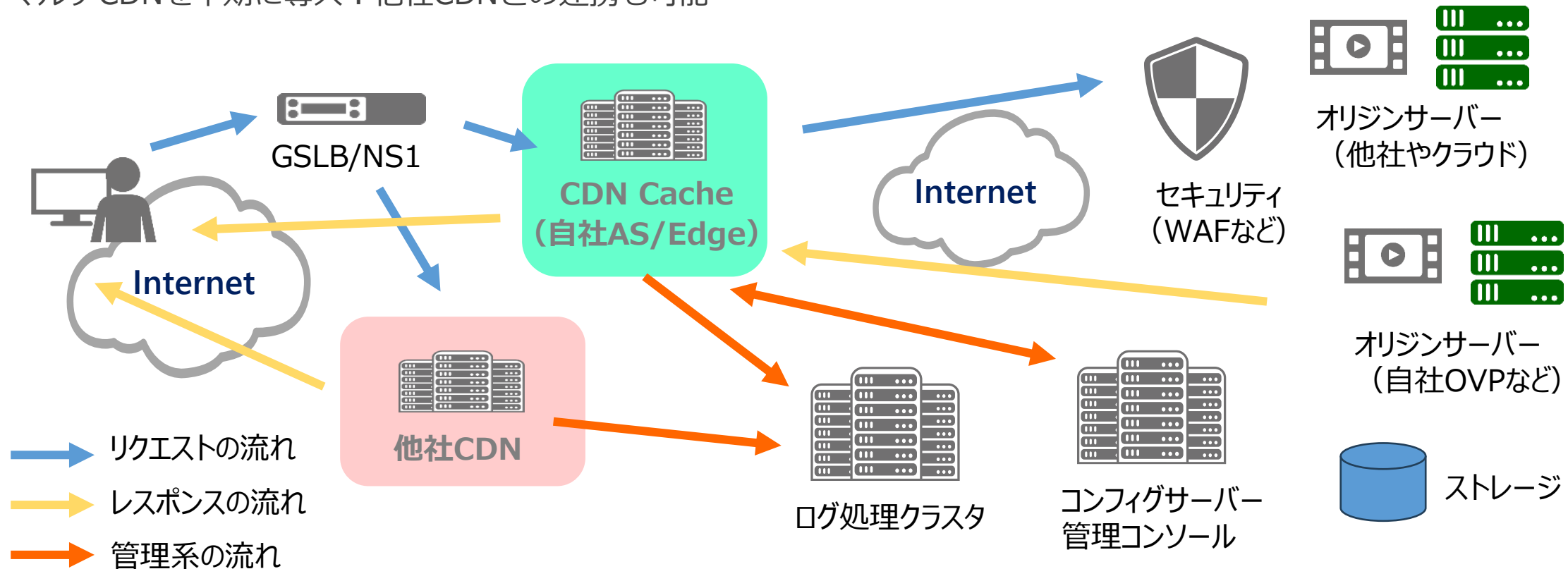
Webページ(動的ファイル)



キャッシュファイルが複数存在
・オリジンへのアクセス増
・キャッシュヒット率の低下

J-Streamのコンテンツ配信システムの全体像

- ▶ 動画配信からのスタート：自社動画配信プラットフォーム（OVP）やプレイヤーも自社開発
- ▶ キャッシュサーバー
 - PoPは自社AS(ASN:24253)に加えて、データセンター事業者のネットワーク内にも設置（Edge拠点）
 - グループ化しており、設定ファイルの管理や誘導方法を変えている（例えば海外配信やSVOD顧客など）
- ▶ マルチCDNを早期に導入：他社CDNとの連携も可能



J-StreamのCDN変遷：動画配信のインフラとしてのCDN

▶ストリーミング黎明期(1999～2005)

- 3大フォーマット(Real/WMT/QT)の時代
- シェアはWindowsMediaが中心

プロプライエタリなソフトウェアを自社拡張してキャッシュサーバーを構築
特にFlashは独自の世界(FMS)

▶ストリーミング普及期(2005～2012)

- YouTubeが普及し、FlashがストリーミングPlayerの主体に
- MicrosoftはSilverlightをリリースするも苦戦
- 各局のVoDサービスが始まり、普及期へ

アプライアンスとオープンソースを組み合わせでキャッシュサーバーを構築（自社利用ツール）

▶スマートフォンの普及(2012～2020)

- 端末がPCからスマートフォンが普及(初代iPhoneが2008年発売)
- 動画もWebのパーツへ
 - HLS登場と標準フォーマット化
 - HTML5でのvideoタグの標準化

オープンソースをベースに独自開発 & Web管理画面も作成し、正式サービス化(2016年)

▶コロナを経て大規模配信が日常化(2020～)

- 視聴デバイスはテレビに移行
- 大規模ライブ配信が話題に

パネルディスカッション

- ▶ 各社のシステムの特徴の話
- ▶ CDNのコア技術であるキャッシュについての本当のところ
 - キャッシュをどう持っているか/キャッシュヒット率を上げるための工夫
 - キャッシュしてしまっ困った話/キャッシュしなくて困った話

以降のディスカッションは会場のみ

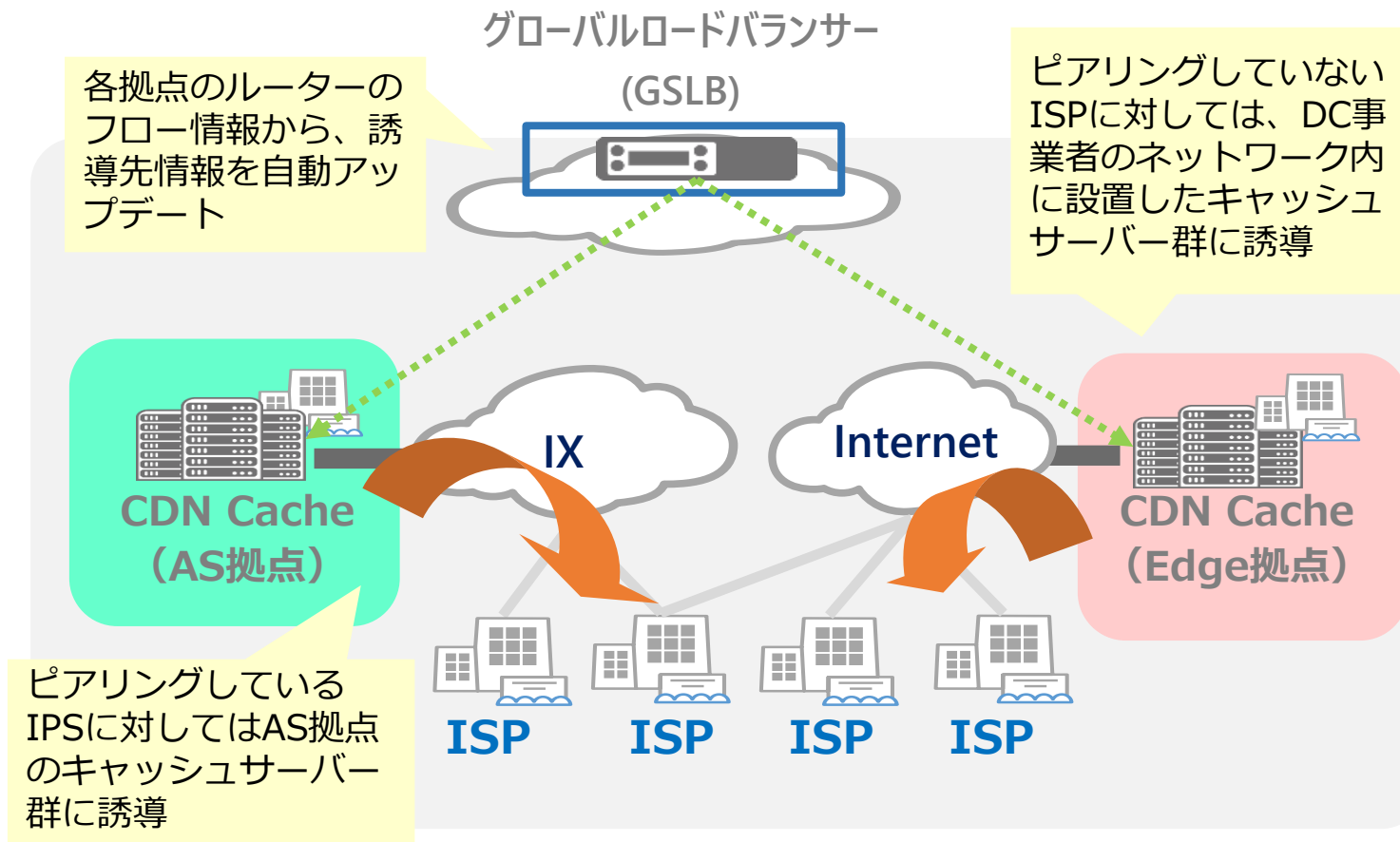
特徴：トラフィックコントロールの仕組み

▶ キャッシュサーバーへの誘導

- 複数設置したキャッシュサーバーに対して、ユーザーアクセスをどのように割り振るか
- ECS(EDNS0 Client Subnet)も積極的に導入

▶ トラフィックコントロールの仕組み

- 各拠点ルーターのフロー情報からピアリングしているISPのリスト作成
- DNSサーバーへのアクセスからISPのDNSサーバーのIPリスト作成
↓
- ISPのDNSサーバーのIPアドレスとAS番号による制御
→IXで接続しているユーザーはIX優先
→トラフィック割合もあるので調整も



キャッシュサーバー構成

▶ クラスターの構成

- フロントとストアは物理的には1台のサーバーにプロセスを分けるなどして実装している = 全サーバーが同一構成になるようにしている
→ キャッシュサーバーの増設が容易、用途転用（オンデマンド→ライブ）をロードバランスだけで実現可能
- 複数台のキャッシュサーバーノードで1つのクラスタを構成している。増設はクラスター単位で実施

▶ フロント

- ユーザーアクセスを受け、SSL処理やURL変換、トークン認証などなどの処理を担当
- キャッシュ容量は少、メモリでヒットさせる（フロントでキャッシュヒットしないアクセスがストア側に流れる）
- フロントサーバーがストアサーバーへアクセスする際にURL毎にアクセスするサーバーをバランシングしている
→ これにより、ストアサーバーのロードバランスとクラスタ化を実現している

▶ ストア

- SSD/NVMeを大量に搭載し、キャッシュヒットさせる
- 1台のストア用サーバー（プロセス）は全URLではなく、フロントサーバーでバランシングされた特定URLをキャッシュする
- ストアサーバーがオリジンサーバーへのアクセスを行う

