

# FAQ for sellers.json and SupplyChain Object

July 2019

Updated – September 2020

注意：

本ドキュメントは、IAB Tech Lab が公開しているドキュメントを簡易的に翻訳したものです。正確な情報は、下記オリジナルドキュメントを参照してください。

[https://iabtechlab.com/wp-content/uploads/2019/07/FAQ-for-sellers.json\\_supplychain-object.pdf](https://iabtechlab.com/wp-content/uploads/2019/07/FAQ-for-sellers.json_supplychain-object.pdf)

## 目次

この FAQ を読むと有益な人は？ .....	4
Sellers.json と SupplyChain についての 1 段落の説明は？ .....	4
VAST リクエスト(または同様のタグ)で、SupplyChain 情報はどのように広告システムに渡され、誰が DSP に渡すのですか？ .....	4
SSAI ベンダーは SupplyChain に含まれるのか？ .....	4
Sellers.json または SupplyChain オブジェクトの存在は、正しさを保証しますか？ .....	5
“seller ID”とは何ですか？ Publisher ID と同じものですか？ .....	5
DSP は、SupplyChain または sellers.json 情報が改ざんされている可能性があることをどのように特定しますか？ .....	5
パブリッシャーですが、どうすればいいですか？ .....	7
Sellers.json ファイルに seller_type を設定する方法は？ .....	7
is_passthrough の設定方法は？ .....	7
パススルーシステムやダウンストリーム広告システムに在庫を供給するパブリッシャー/仲介業者は、is_passthrough=1 を設定する必要がありますか？ .....	8
広告プラットフォームがヘッダービディングパートナーに報酬を支払っていない場合、SupplyChain オブジェクトと sellers.json の両方でヘッダービディングのインテグレーションをどのように伝えるべきですか？ .....	8
Examples .....	8
標準ヘッダーまたはビディングタグのシナリオ .....	9
パブリッシャーダイレクト .....	9
Ad Network.....	10
パススルー／エクスチェンジビディングシナリオ .....	12
シナリオ 1: パブリッシャーからエクスチェンジビディング参加者へのパススルー..	12

シナリオ 1a：リセラーからエクスチェンジビディング参加者へのパススルー.....	14
シナリオ 1b：在庫が Exchange1 からパススルーエクスチェンジ経由でバイヤーに送 られます .....	17
シナリオ 2：パブリッシャーから EB Exchange 参加者へのパススルー(エクスチェン ジ参加者がパブリッシャーに直接支払う場合) .....	20
セールスハウスのシナリオ .....	21
#1、#2、#3 のそれぞれについて、SupplyChain と sellers.json に関して何が期待さ れていますか？ .....	22
マルチインテグレーションシナリオ(Wrapper, EB etc.):.....	27
ダイレクトパブリッシャー在庫のディールプロバイダーシナリオ .....	29

この FAQ を読むと有益な人は？

Sellers.json や SupplyChain オブジェクトを実装する人は、作成者としても消費者としても、この中で共有されているコンテキストから恩恵を受けられます。この文書と支援リソースの進化が期待されます。

Sellers.json と SupplyChain についての 1 段落の説明は？

Sellers.json と SupplyChain は、購入プラットフォームからパブリッシャーに戻る金銭の流れに参加するすべての仲介者を識別するメカニズムです。金銭の流れに参加しない仲介者は含まれず、サービスに対する手数料は支払われるが、上流のセラーには支払わないシステムも含まれません。サプライチェーンが複雑な場合、これにより透明性が高まり、消費広告の事業方針に従って、不正な供給元やその他の容認できない供給元を特定し、防止することが可能になります。

VAST リクエスト(または同様のタグ)で、SupplyChain 情報はどのように広告システムに渡され、誰が DSP に渡すのですか？

リクエストがパブリッシャーに代わって販売されるインベントリである場合、広告システムは SupplyChain 情報を提供し、タグは SupplyChain 情報を含む必要はありません。リクエストが仲介者の代理で販売されるインベントリの場合、仲介者(および上流の仲介者)の SupplyChain ノードがタグに提供され、広告システムの SupplyChain ノードはリクエストを送信する前に追加されます。

OpenRTB の SupplyChain オブジェクトのドキュメントでは、サプライチェーン情報を文字列としてエンコードするための標準化された構文を規定しています。広告システムは、ドキュメントで指定されているように、SupplyChain オブジェクトからサプライチェーンの詳細を受信することをサポートすべきです。

SSAI ベンダーは SupplyChain に含まれるのか？

購入システムからパブリッシャーに至るまで、支払いの連鎖の一部であるすべての仲介者が含まれることが期待されます。SSAI ベンダーが仲介業者として機能している場合は、SSAI ベンダーを含めるべきです。SSAI ベンダーが広告配信ベンダーのような役割を果たし、パブリッシャーから広告配信料を受け取り、メディアの金銭の流れに関与していない場合は、リストされません。

Sellers.json または SupplyChain オブジェクトの存在は、正しさを保証しますか？

いいえ。これらは、情報の消費者(DSP など)が検証する必要がある追加情報を提供するツールであり、複数の方法で行えます。消費者は、偽りの可能性のある情報が出現することを予期し、防御する必要があります。

“seller ID”とは何ですか？ Publisher ID と同じものですか？

seller ID とは、広告システムがプラットフォーム上のインベントリセラーのそれぞれに割り当てるユニークな ID のことです。これは通常、OpenRTB ビッドリクエストの“publisher”オブジェクトの“id”フィールドに記載されます。また、ads.txt ファイルの 2 列目にあるのと同じ ID でもあります。歴史的に、この ID の OpenRTB 用語は、パブリッシャーとの直接的な関係のみを持つシンプルなネットワークエクステンジを想定しています。このフィールドは、エクステンジ上のセラーアカウントを表すことを意図していますが、このフィールドの意図する目的について混乱しているエクステンジは、実際に誰が在庫を供給しているかに関係なく、ドメインやアプリの所有者の抽象的な概念を表す ID を提供することがあります。

Seller ID は、sellers.json および SupplyChain の目的のために、広告システムがインベントリに対して直接支払う単一のエンティティを表す必要があります。1 つの広告システムで 1 つのビジネスを表すために複数のセラー ID を使用することはできますが、1 つの Seller ID で複数のビジネスを表すことはできません。

DSP は、SupplyChain または sellers.json 情報が改ざんされている可能性があることをどのように特定しますか？

特定のビッドリクエストの SupplyChain 情報が改ざんされている可能性もありますが、十分なデータポイントを相関させることで、ほとんどの場合、疑わしい情報を検出できます。

SupplyChain 情報を検証するためのサンプルメソッド:

- 指定されたノードについて、指定された広告システムの seller ID(sellers.json)と関連付けられた名前は、直前のノードの広告システムと一致しなければなりません。そうでない場合、チェーンの断絶を意味します。
- 特定のドメイン/アプリの Ads.txt レコードは、そのドメイン/アプリの SupplyChain

の上流ノードに存在すること。これは既存の ads.txt 仕様から拡張されたガイダンスですが、ベストプラクティスと考えるべきです。

- DSP はスポットチェックを行い、パブリッシャーが期待する在庫の販売方法とサプライチェーンが一致しているかどうかをパブリッシャーに問い合わせられます。また、SupplyChain の情報を使って、特定のチェーンや仲介業者経由で販売された在庫の合計を、任意の期間に亘って通知することもできます。
- “complete”フラグが設定されている場合、最初のノードのエンティティ名が、サイトやアプリの既知の所有者と一致していることを確認できます。
- “complete”フラグが設定されている場合、最初のノードの seller\_type が PUBLISHER であることを確認できます。“complete”フラグが設定されている場合、最初のノードに PUBLISHER という売り手タイプがあるかどうかを確認できます。
- SupplyChain オブジェクトの最初のノードが、パブリッシャーの ads.txt ファイルに DIRECT セラーとしてリストされているか確認してください。ほとんどの場合、そうになっているはずですが、そうでない場合は、以下の 2 つの理由のいずれかを考慮してください:
  - 実際の最初のノードがチェーンから削除されている(SupplyChain が改ざんされている)。
  - パブリッシャーが ads.txt ファイルに RESELLER としてレコードを誤ってリストしています。

#### Sellers.json 情報の検証方法のサンプル

- DSP は、sellers.json ファイルが特定の seller ID がそのパブリッシャーを直接表しているとしている場合、パブリッシャーに確認を求めることでスポットチェックできます。
- さまざまな不規則なパターンを探せます。例えば、publisher ID とドメイン/アプリの相関が軒並み 1:1 である場合、seller ID が正しく使用されていないことを示唆します。また、ある seller ID に対して、一見無関係なアプリ/ドメインが多数観察されますが、seller\_type PUBLISHER に設定されているか、セラー名が仲介者であることが判明している場合も同様です。
- パブリッシャーが ads.txt ファイルでリレーションシップタイプを誤って宣言することがあることを理解した上で、サイトの ads.txt ファイルに記載されている DIRECT セラーアカウントと、ads.txt ファイルに記載されている seller\_type および

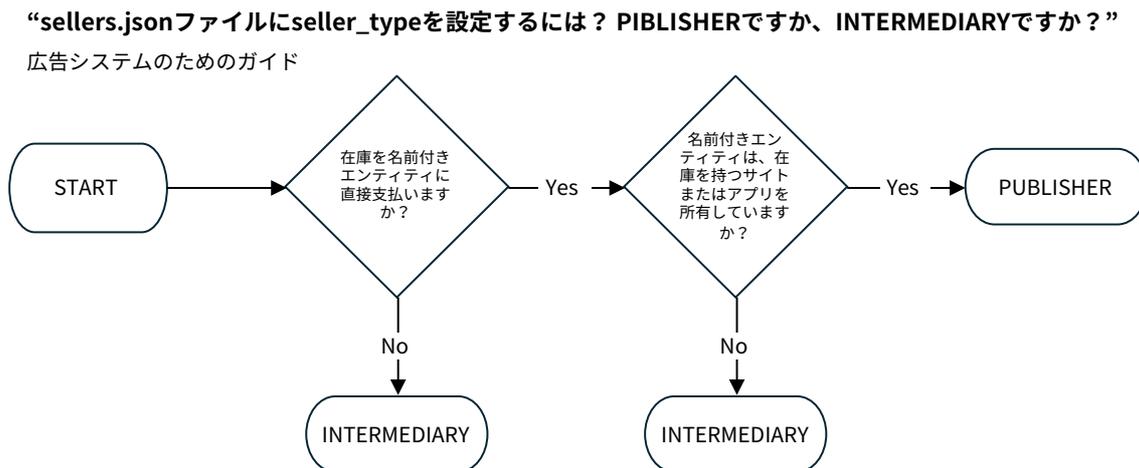
is\_passthrough 属性(およびエンティティ名)の間には、一般的な一貫性を保つ必要があります。属性(およびエンティティ名)の間の一般的な整合性を保つ必要があります。以下に例を示します。

パブリッシャーですが、どうすればいいですか？

何もありません。sellers.json と SupplyChain は、広告システムによって実装または消費される仕様です: DSP、SSP/Exchange、Ad Server などです。同様に、パブリッシャーが SupplyChain オブジェクトを、仲介者に代わって送信されないタグを介して供給する必要はありません。

Sellers.json ファイルに seller\_type を設定する方法は？

以下のフローチャートを参照してください:



最後に、"Yes"と"No"の組み合わせが当てはまる場合は、"BOTH"を設定します。

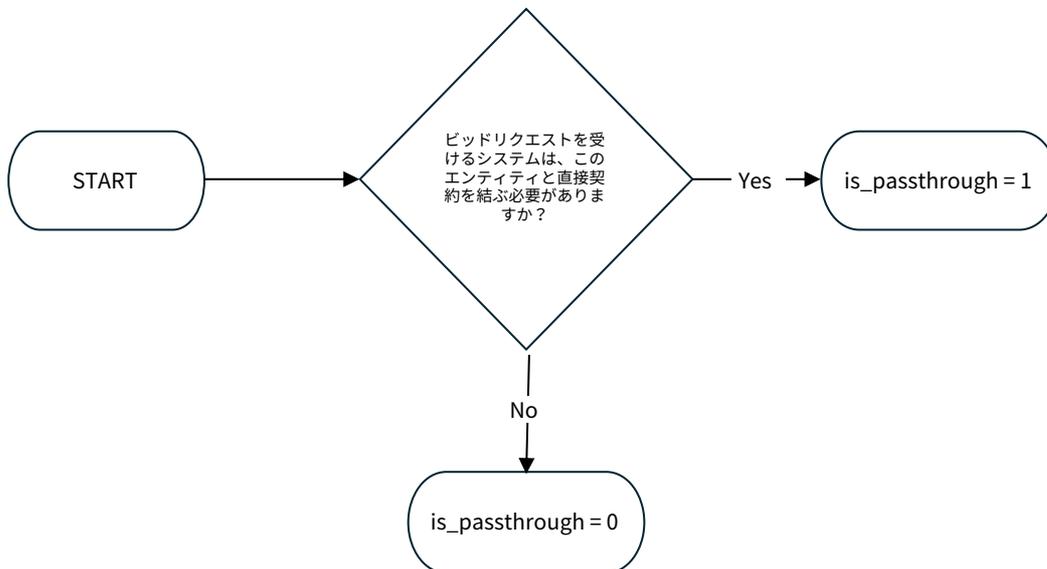
is\_passthrough の設定方法は？

このフィールドは、広告システムが消費システムに対して、指定されたエンティティと直接的な契約関係を保持することを要求するときに設定されます。消費側広告システムは、さらにビッドリクエストを他者にブロードキャストできますが、受信側にも供給元との直接的な契約関係を保持することを要求しない限り、sellers.json ファイルの is\_passthrough フィールドを 1 に設定しません。

以下のフローチャートを参照してください:

## “sellers.jsonファイルにis\_passthroughを設定するには？

広告システムのためのガイド



シナリオ例はこちら。

パススルーシステムやダウンストリーム広告システムに在庫を供給するパブリッシャー/仲介業者は、is\_passthrough=1 を設定する必要がありますか？

いいえ、パススルーとして動作している広告システムだけが is\_passthrough=1 を設定する必要があります。

広告プラットフォームがヘッダービディングパートナーに報酬を支払っていない場合、SupplyChain オブジェクトと sellers.json の両方でヘッダービディングのインテグレーションをどのように伝えるべきですか？

現時点では、バイイングシステムとパブリッシャー間の直接的な支払い連鎖に含まれないテクノロジーベンダーは、SupplyChain オブジェクトにリストされるべきではありません。また、これらのベンダーに seller ID を割り当てる必要もありません。

### Examples

OpenRTB ビッドリクエストの sellers.json と SupplyChain オブジェクトは、さまざまなシナリオで連携して動作します。以下の例では、ads.txt、sellers.json、SupplyChain ノードにプロパティを設定する方法について、スニペットと詳細を用いていくつかのシナリオの

概要を説明します。

最初に、ごく標準的なヘッダービディングまたはタグのシナリオの詳細を説明します。次に、パススルー/エクステンジビディングのシナリオを見ます。最後に、セールスハウスのシナリオとマルチインテグレーションのシナリオについて説明します。

標準的なヘッダーまたはビディングタグ

パススルー/エクステンジビディング

セールスハウス

マルチインテグレーション

標準ヘッダーまたはビディングタグのシナリオ

標準的なヘッダーまたはビディングタグでは、広告リクエストはパブリッシャーから直接来るか、Ad Network から来るかもしれません。ここでは、パブリッシャーから直接と Ad Network のシナリオをカバーします。

パブリッシャーダイレクト

パブリッシャー広告のリクエストは、Ad Exchange に送信されます。Ad Exchange はパブリッシャーに代わってオークションを設定し、実行します。このビッドリクエストの schain には、1つのノード(exchange.com)とパブリッシャーの ID のみが含まれます。このサプライチェーン情報はエクステンジから提供される必要があります。

**exchange.com/sellers.json**

```
...
"sellers": [
  {
    "seller_id": "184003",
    "name": "Meredith",
    "seller_type": "PUBLISHER",
    "domain": "meredith.com"
  }
]
```

...

**exchange.com** からのリクエスト用の OpenRTB の SupplyChain オブジェクト:

```
"schain":
{
  "ver":"1.0",
  "complete": 1,
  "nodes": [
    {
      "asi":"exchange.com",
      "sid":"184003",
      "hp":1
    }
  ]
}
```

Ad Network

パブリッシャーが Blue Network という Ad Network と提携しています。Blue Network は、ads.txt を介してパブリッシャーの在庫を販売する許可を与えられています。

**Ads.txt entry:**

```
exchange.com, <bluenetwork ID>, reseller, {tag ID}.
```

このシナリオでは、Blue Network は、すでに SupplyChain 情報がリクエストに含まれているビッドリクエストを exchange.com に送信する責任を負います。これは、OpenRTB リクエスト経由でも、タグ内にエンコードされたものでもかまいません(リクエスト情報を送信する他の方法でもかまいません)。exchange.com はその情報をサプライチェーンに追加する責任があります。

**bluenetwork.com/sellers.json:**

...

```
"sellers": [
  {
```

```
"seller_id": "1200",
"name": "Meredith",
"seller_type": "PUBLISHER",
"domain": "meredith.com"
}
]
...
```

#### **exchange.com/sellers.json**

```
...
"sellers": [
  {
    "seller_id": "184033B",
    "name": "Blue Network",
    "seller_type": "INTERMEDIARY",
    "domain": "bluenetwork.com"
  }
]
...
```

#### **exchange.com からのリクエスト用 SupplyChain**

```
"schain":
{
  "ver": "1.0",
  "complete": 1,
  "nodes": [
    {
      "asi": "bluenetwork.com",
      "sid": "1200",
      "hp": 1
    },
    {
```

```
"asi":"exchange.com",
"sid":"184033B",
"hp":1
}
]
}
```

### パススルー／エクスチェンジビディングシナリオ

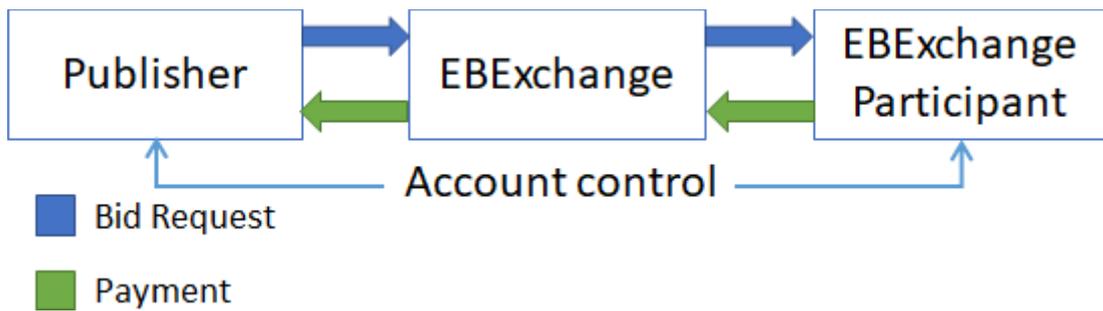
パススルーエクスチェンジは一般的なリセールや Ad Network のシナリオとは異なり、インベントリのサプライヤーからインベントリの消費者へのインベントリのファシリテーターとしてのみ機能します。サプライヤーと消費者は、サプライヤーが消費者のプラットフォーム内でアカウントを管理できるようなビジネス関係を確立しなければなりません。

このセクションでは、2つのパススルーシナリオを取り上げます。

1. パブリッシャーからエクスチェンジビディング参加者へのパススルー
  - a. リセラーからエクスチェンジビディング参加者へのパススルー
  - b. パススルーエクスチェンジを使用してエクスチェンジからバイヤーに送られる在庫
2. パブリッシャーからエクスチェンジへのパススルーで、エクスチェンジがパブリッシャーに直接支払います

#### シナリオ 1: パブリッシャーからエクスチェンジビディング参加者へのパススルー

在庫はパブリッシャーから提供されます。エクスチェンジ参加者がエクスチェンジビディングプロバイダーに支払います。エクスチェンジビディングプロバイダーがパブリッシャーに支払います。



Ads.txt: DIRECT

[ebexchange.com/sellers.json](https://ebexchange.com/sellers.json):

```

...
"sellers": [
  {
    "seller_id": "pub-0978064532142215",
    "name": "Meredith",
    "seller_type": "PUBLISHER",
    "domain": "meredith.com",
    "is_passthrough": 1,
    "comment": "Must establish account relationship with Meredith to transact"
  }
]
...
  
```

[ebparticipant.com/sellers.json](https://ebparticipant.com/sellers.json):

```

...
"sellers": [
  {
    "seller_id": "184044",
    "name": "EB Exchange",
    "seller_type": "INTERMEDIARY",
    "domain": "ebexchange.com",
    "comment": "Meredith via Exchange Bidding"
  }
]
...
  
```

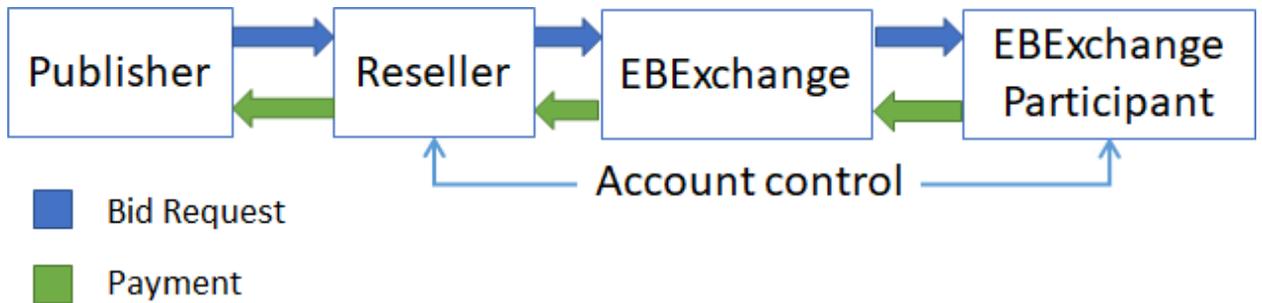
```
    }  
  ]  
  ...
```

### SupplyChain for requests from ebparticipant.com

```
"schain": {  
  "ver": "1.0",  
  "complete": 1,  
  "nodes": [  
    {  
      "asi": "ebexchange.com",  
      "sid": "pub-0978064532142215",  
      "hp": 1  
    },  
    {  
      "asi": "ebparticipant.com",  
      "sid": "184044",  
      "hp": 1  
    }  
  ]  
}
```

シナリオ 1a：リセラーからエクスチェンジビディング参加者へのパススルー

在庫はリセラーから。エクスチェンジ参加者がエクスチェンジビディングプロバイダーに支払います。エクスチェンジビディングプロバイダーがリセラーに支払い、リセラーがパブリッシャーに支払います。



Ads.txt: RESELLER

resellerexchange.com/sellers.json:

```

...
"sellers": [
  {
    "seller_id": "215",
    "name": "Sports Publisher",
    "seller_type": "PUBLISHER",
    "domain": "sportspublisher.com"
  }
]
...

```

ebexchange.com/sellers.json:

```

...
"sellers": [
  {
    "seller_id": "pub-3153065230153281",
    "name": "Reseller Exchange",
    "seller_type": "INTERMEDIARY",
    "domain": "resellerexchange.com",
    "is_passthrough": 1,
    "comment": "Must establish account relationship with Reseller Exchange to

```

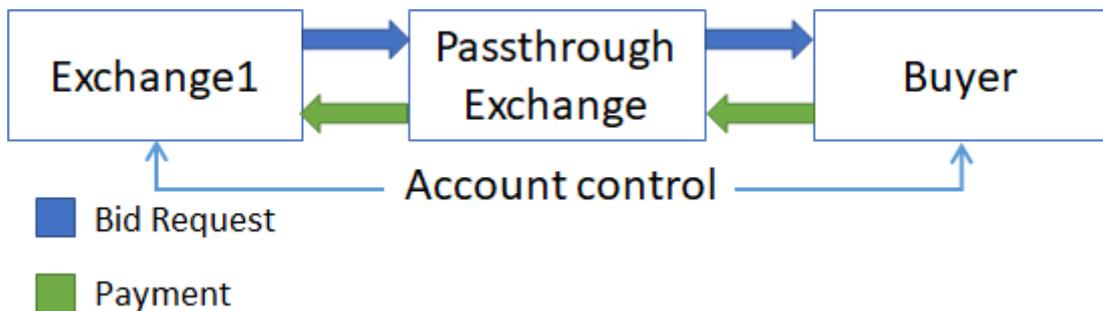
```
transact"  
  }  
]  
...
```

### SupplyChain for requests from ebparticipant.com

```
"schain": {  
  "ver": "1.0",  
  "complete": 1,  
  "nodes": [  
    {  
      "asi": "resellerexchange.com",  
      "sid": "215",  
      "hp": 1  
    },  
    {  
      "asi": "ebexchange.com",  
      "sid": "pub-3153065230153281",  
      "hp": 1  
    },  
    {  
      "asi": "ebparticipant.com",  
      "sid": "185176",  
      "hp": 1  
    }  
  ]  
}
```

シナリオ 1b：在庫が Exchange1 からパススルーエクスチェンジ経由でバイヤーに送られます

バイヤーはパススルーエクスチェンジに代金を支払います。パススルーエクスチェンジは Exchange1 に代金を支払います。Exchange1 とバイヤーはアカウント管理関係にあります。



Ads.Txt: パススルーエクスチェンジが関与していない場合と同じ状態

exchange1.com/sellers.json:

```
...  
"sellers": [  
  {  
    "seller_id": "2000",  
    "name": "Meredith",  
    "seller_type": "PUBLISHER",  
    "domain": "meredith.com"  
  },  
  {  
    "seller_id": "2001",  
    "name": "blueexchange",  
    "seller_type": "INTERMEDIARY",  
    "domain": "blueexchange.com"  
  }  
]
```

...

**ptexchange.com/sellers.json:**

...

```
"sellers": [  
  {  
    "seller_id": "exchange1_2000",  
    "name": "Exchange 1",  
    "seller_type": "INTERMEDIARY",  
    "domain": "exchange1.com",  
    "is_passthrough": 1,  
    "comment": "Must establish account with Exchange 1 to transact"  
  },  
  {  
    "seller_id": "exchange1_2001",  
    "name": "Exchange 1",  
    "seller_type": "INTERMEDIARY",  
    "domain": "exchange1.com",  
    "is_passthrough": 1,  
    "comment": "Must establish account with Exchange 1 to transact"  
  }  
]
```

...

**blueexchange.com/sellers.json:**

...

```
"sellers": [  
  {  
    "seller_id": "ABC",  
    "name": "Red Publisher",  
    "seller_type": "PUBLISHER",  
    "domain": "redpublisher.com"
```

```
}  
]  
...
```

## ptexchange.com からのリクエストのサンプル SupplyChain

### パブリッシャーからのリクエストの場合

```
"schain": {  
  "ver": "1.0"  
  "complete": 1,  
  "nodes": [  
    {  
      "asi": "exchange1.com",  
      "sid": "2000",  
      "hp": 1  
    },  
    {  
      "asi": "ptexchange.com",  
      "sid": "exchange1_2000",  
      "hp": 1  
    }  
  ]  
}
```

### 仲介業者 blueexchange.com からの依頼の場合

```
"schain": {  
  "ver": "1.0"  
  "complete": 1,  
  "nodes": [  
    {  
      "asi": "blueexchange.com",  
      "sid": "ABC",  
    }  
  ]  
}
```

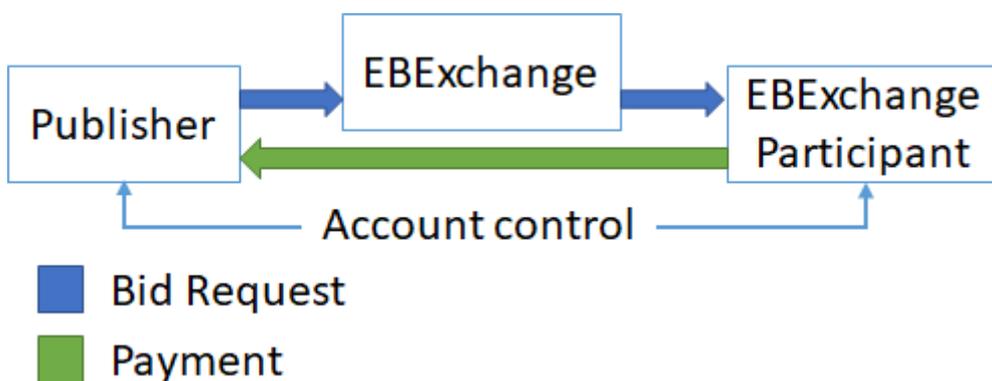
```

    "hp":1
  },
  {
    "asi":"exchange1.com",
    "sid":"2001",
    "hp":1
  },
  {
    "asi":"ptexchange.com",
    "sid":"exchange1_2001",
    "hp":1
  }
]
}

```

シナリオ 2：パブリッシャーから EB Exchange 参加者へのパススルー(エクステンジ参加者がパブリッシャーに直接支払う場合)

エクステンジ参加者がパブリッシャーに直接支払います。



Ads.txt: DIRECT

EBExchangeParticipant.com/sellers.json:

```
...
"sellers": [
  {
    "seller_id": "184044-B",
    "name": "Meredith",
    "seller_type": "PUBLISHER",
    "domain": "meredith.com",
    "comment": "via Amazon TAM"
  }
]
...
```

#### EBExchange 参加者からのリクエスト用の SupplyChain

```
"schain": {
  "ver": "1.0",
  "complete": 1,
  "nodes": [
    {
      "asi": "ebexchangeparticipant.com",
      "sid": "184044-B",
      "hp": 1
    }
  ]
}
```

#### セールスハウスのシナリオ

Concept Content はアメリカのセールスハウス/レップ会社です。テクノロジープラットフォームは所有していません。Concept Content は以下のような取り決めをしています:

1. パブリッシャー(ウェブサイトの所有者)は Concept Content に在庫を 100%管理させます。パブリッシャーが在庫を直接販売することはありません。Concept Content は、直接またはプログラマティックチャンネルを経由して、在庫の販売を完全にコントロー

ルします。

簡単のため、このケースを 1 つだけ見てみましょう。MortgageCentral はパブリッシャー(事業体)で、単一のサイト、mortgagecentral.com を運営しています。彼らは Concept Content とこのような契約を結んでいます。

2. カナダの会社("Significant Media")は、アメリカの在庫をすべて Concept Content に管理させています。パブリッシャーはカナダの在庫を直接管理しますが、アメリカの在庫は Concept Content に委託します。Significant Media は”Premium Network”モデルを運営しており、Significant Media が所有・運営するフラッグシップサイト、significant.info があります。また、所有はしていませんが独占販売代理権を持つ他の 2 つのサイトも収益化しています。

Significant Media は、アメリカのトラフィックをすべて自社のアドサーバー経由で Concept Content に誘導することで、これを促進しています。

3. Concept Content は 82mysteries.com も運営しています。

Concept Content は、Exchange1 を通じてこの在庫のすべてをプログラムで販売しています。Exchange1 の seller ID は 104 です。

これらはすべて、Concept Content のアドサーバーアカウントでトラフィックされる広告タグでレガシーウォーターフォールスタイルに設定されています。エクステンジのセラーアカウント内の各ウェブサイトには、別々の”Sites”が設定されています。

Concept Content は Exchange1 から手動でレポートを取得し、課金を処理します。Concept Content はサイトごとの支払いを在庫の対応するソースに割り当て、そのソースは Concept Content から支払いを受けます。

#1、#2、#3 のそれぞれについて、SupplyChain と sellers.json に関して何が期待されますか？

SupplyChain は金銭の流れを表しますので、"complete"が設定されている場合、"first"ノードは実際のサイトオーナーから下流の最初のエンティティであることが期待されます。Concept Content は、供給を受ける各ビジネスエンティティを表す sellers.json ファイルを

持っていることが期待されます。Concept Content には、供給元の各ビジネスエンティティを表す sellers.json ファイルが配置されていることが期待されます。O&O 在庫は他の誰からも取得しないため(Concept Content が所有します)、O&O 在庫にエント리는必要ありません。この場合、Concept Content は広告システムを運営していないため、各上流サプライヤーに割り当てられた ID は、特定の外部的な意味を持つ必要はなく、単純にインクリメントされた番号とすることができます。

### Concept Content の sellers.json file

Concept Content のサイトは <https://conceptcontent.info> です。

ファイルは <https://conceptcontent.info/sellers.json> にあります:

```
{
  "version": "1.0",
  "sellers": [
    {
      "seller_id": "1",
      "name": "MortgageCentral",
      "seller_type": "PUBLISHER",
      "domain": "mortgagecentral.com"
    },
    {
      "seller_id": "2",
      "name": "Significant Media",
      "seller_type": "BOTH",
      "domain": "significant.info"
    }
  ]
}
```

#1(MortgageCentral) : Concept Content が Exchange1 に送信した SupplyChain オブジェクト

SupplyChain のネイティブ JSON 表現では、SupplyChain オブジェクトは以下のような

ります:

```
"schain": {
  "ver": "1.0",
  "complete": 1,
  "nodes": [
    {
      "asi": "conceptcontent.info",
      "sid": "1",
      "hp": 1
    }
  ]
}
```

しかし、Concept Content はタグベースのインテグレーションを使用しているため、Exchange1 に以下のようなサプライチェーンの文字列を供給する必要があります:

```
1.0,1!conceptcontent.info,1,1
```

これは、例えばサプライチェーンの文字列を消費するために Exchange1 が指定した場所に挿入されます:

```
<script
src="https://ads.exchange1.com/srv?pid=194&sz=300x250&plid=2842181&schain=
1.0,1!conceptcontent.info,1,1"></script>
```

#1: Exchange1 からバイディングシステムに送信された SupplyChain

```
"schain": {
  "ver": "1.0",
  "complete": 1,
  "nodes": [
```

```

{
  "asi":"conceptcontent.info",
  "sid":"1",
  "hp":1
},
{
  "asi":"exchange1.com",
  "sid":"194",
  "hp":1
}
]
}

```

Exchange1 からバイイングシステムへのさらなるリクエストの例は省略します。上記のように、Exchange1 に行く SupplyChain と Exchange1 を表すノード、そして Exchange1 の Concept Content の seller ID となります。

#2(Significant Media inventory): SupplyChain object sent by Concept Content to Exchange1

この例は、Significant Media の O&O サイト significant.info のものです。

```

"schain": {
  "ver":"1.0",
  "complete": 1,
  "nodes": [
    {
      "asi":"conceptcontent.info",
      "sid":"2",
      "hp":1
    }
  ]
}
}

```

または、このインベントリのタグに含まれる以下のシリアル化されたデータ:

```
1.0,1!conceptcontent.info,2,1
```

もし、Concept Content が O&O 以外のインベントリをソースにきれいに戻す方法を見つけれない場合、simple.info 以外のリクエストに対して以下の SupplyChain を送信することになります:

```
"schain": {
  "ver": "1.0",
  "complete": 0,
  "nodes": [
    {
      "asi": "conceptcontent.info",
      "sid": "2",
      "hp": 1
    }
  ]
}
```

または、このインベントリのタグに含まれる以下のシリアル化されたデータ:

```
1.0,1!conceptcontent.info,2,1
```

#3(Concept Content の O&O サイト)

#3 では、82mysteries.com が Concept Content の O&O サイトであるため、SupplyChain は Concept Content から Exchange1 に送信されません。(Exchange1 は、どの内部サイト ID がセラーアカウント保有者によって所有・運営されているかの記録を保持し、それに応じて“complete”フラグを設定する必要があります。なぜなら、アップストリームの SupplyChain 情報が与えられておらず、O&O サイトであると考えられる場合、それは完全なパスでなければならないからです)。

この場合、わかりやすくするために Exchange1 から DSP へのビッドリクエストが表示されます:

```
"schain": {
  "ver": "1.0",
  "complete": 1,
  "nodes": [
    {
      "asi": "exchange1.com",
      "sid": "194",
      "hp": 1
    }
  ]
}
```

マルチインテグレーションシナリオ(Wrapper, EB etc.):

シナリオは、Exchange(flexexchange.com)が同じパブリッシャーを2つの異なる供給経路で販売している場合です。例えば、1つはヘッダービディングで、エクステンジはヘッダーから直接呼び出され、パブリッシャーに直接在庫を支払います。もう1つはエクステンジビディングで、エクステンジが別のエクステンジ(ebexchange.com)に在庫を支払い、ebexchange がパブリッシャーに支払います。このシナリオでは、flexexchange は複数の seller ID を保持する必要があります。1つの seller ID で両方のパスを表すべきではありません。

flexexchange.com\sellers.json

```
...
"sellers": [
  {
    "seller_id": "180000",
    "name": "PublisherA",
```

```
    "seller_type": "PUBLISHER",
    "domain": "publishera.com"
  },
  {
    "seller_id": "180000B",
    "name": "EB Exchange",
    "seller_type": "INTERMEDIARY",
    "domain": "ebexchange.com",
    "comment": "PublisherA via Exchange Bidding"
  }
]
...
```

ebexchange.com\sellers.json

```
...
"sellers": [
  {
    "seller_id": "pub-1234",
    "name": "PublisherA",
    "seller_type": "PUBLISHER",
    "is_passthrough": 1,
    "domain": "publishera.com"
  }
]
...
```

ebexchange.com からの在庫のための SupplyChain

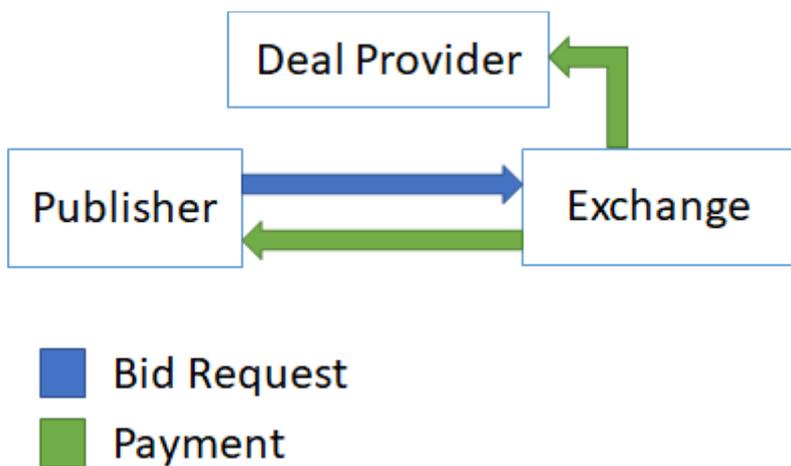
```
"schain": {
  "ver": "1.0",
  "complete": 1,
  "nodes": [
    {
```

```
    "asi":"ebexchange.com",
    "sid":"pub-1234",
    "hp":1
  },
  {
    "asi":"flexexchange.com",
    "sid":"180000B",
    "hp":1
  }
]
}
```

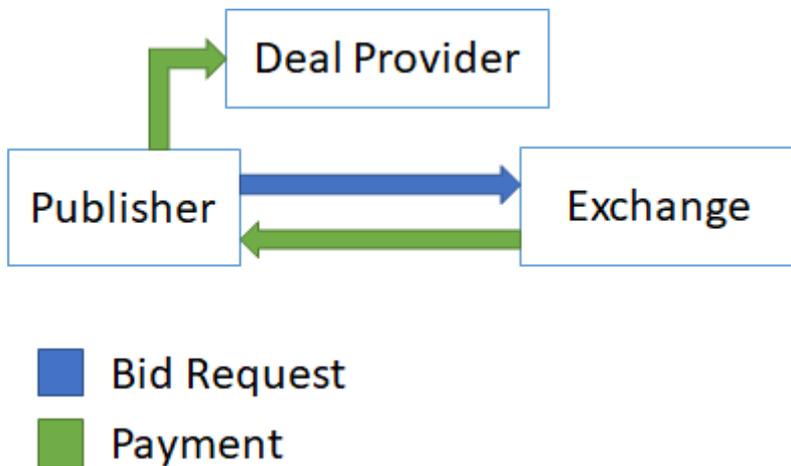
PublisherA からの在庫のための SupplyChain

```
"schain": {
  "ver":"1.0",
  "complete": 1,
  "nodes": [
    {
      "asi":"publishera.com",
      "sid":"180000",
      "hp":1
    }
  ]
}
```

ダイレクトパブリッシャー在庫のディールプロバイダーシナリオ



または



このシナリオでは、次のような取引があります:

- Publisher Direct
- Ad Network

からの取引は、トラフィックの重複を避け、レイテンシーを改善するために、1つのビッドリクエスト内で結合されます(2つのオークションをデージーチェーン接続する場合: Publisher->Ad Network, Ad Network->Buyer)。

在庫はパブリッシャーの管理下に置かれたまま(seller ID/name、サイトまたはアプリ ID/name /domain/app-bundle、すべての広告品質設定)ですが、支払いの流れは取引によって異なります。

落札者が Ad Network によって定義されたディールを使用する場合、支払いの流れは次のようになります:

バイヤーからエクスチェンジ、エクスチェンジからパブリッシャーとディールプロバイダーの両方に支払われます。

または

バイヤーからエクスチェンジ、エクスチェンジからパブリッシャー、パブリッシャーからディールプロバイダー。

バイヤーからパブリッシャーへの支払いはディールプロバイダーを経由しないため、ディールプロバイダーをサプライチェーンに含める必要はありません。