



## IMS OneRoster® : ベストプラクティスと実装ガイド

### IMS 最終リリース バージョン 1.1

発行日 : 2017年4月17日  
最新バージョン : <http://www.imsglobal.org/lis/>

#### 知的財産権と配布に関するお知らせ

この文書の受信者は、この文書に記載された仕様書のいずれかの実装によって侵害される可能性があることを認識したうえで、関連するすべての特許のクレームまたはその他の知的財産権の通知をコメント付きで提出するよう、またサポートを提供するように要求されています

IMSは、本書に記載されている技術の実装や使用に関係すると主張される可能性がある知的財産やその他の権利、あるいはそういう権利の元でのライセンスがどの程度まで利用できるのか利用できないのか、という有効性または範囲に関して、いかなる立場も取りません; また、そうした権利を特定するために取り組みを代表していません。IMS仕様の権利に関するIMSの手順の情報は、[http://www.imsglobal.org/ipr/imsipr\\_policyFinal.pdf](http://www.imsglobal.org/ipr/imsipr_policyFinal.pdf) のIMS知的財産権のウェブページで見つけることができます。

Copyright © 2017 IMS Global Learning Consortium. 無断転載・複製を禁止します。

製品やサービスを開発するために、この仕様書を使用するには、IMSのウェブサイト : <http://www.imsglobal.org/speclicense.html>。

提案の要求の必要に応じて、この文書からの抜粋を使用するすべての当事者に許可されます。

上記で付与された制限付きのアクセス許可は永続的であり、IMSまたはその継承者または割り当てによって取り消されることはありません。

この仕様は、いかなる保証もなく提供されています、特に、非侵害のいかなる保証をはっきりと否定しています。この仕様の利用については、実装者自身の責任で行われ、コンソーシアム、そのメンバーまたは提出者のいずれも、直接的または間接的に、この仕様の利用から生じるいかなる性質の損害についても、実装者または第三者に対して、いかなる責任を持つものではありません。

公共の貢献、コメントや質問はここに掲載することができます : <http://www.imsglobal.org/forums/ims-glc-public-forums-and-resources/learning-information-services-oneroster-public-forum>。

© 2017 IMS Global Learning Consortium, Inc.  
無断転載・複製を禁止します。

商標情報 : <http://www.imsglobal.org/copyright.html>

IMS のロゴとOneRosterは、米国および/またはその他の国において、IMS Global Learning Consortium, Inc. の商標です。

ドキュメント名 : IMS OneRoster® 1.1 ベストプラクティスおよび実装ガイド

改訂 : 2017年4月17日

[コラムの先頭へ](#) | [先頭へ](#)

## 目次

### 1. はじめに

- 1.1 スコープとコンテキスト
  - 1.2 このドキュメントのステータス
  - 1.3 このドキュメントの構造
  - 1.4 関連ドキュメント
  - 1.5 命名法
  - 2. OneRoster 相互運用性アーキテクチャー
  - 3. Common REST/CSV 導入のベストプラクティス
    - 3.1 識別子とその使用
      - 3.1.1 ソース識別子
      - 3.1.2 リソース ベンダーID
      - 3.1.3 ユーザーID
      - 3.1.4 学校と学区の識別子
    - 3.2 'Users'クラスの'enabledUser'属性
    - 3.3 ステータスフィールドの値の意味
  - 4. CSVファイル処理のベストプラクティス
    - 4.1 ファイルのエンコーディングとファイルの順序
    - 4.2 CSVファイルの交換
      - 4.3 V1.1とV1.0の間の往復
      - 4.4 セマンティックの一貫性の確保
    - 4.5 CSVファイルのデータモデルの拡張
      - 4.5.1 語彙の拡張
      - 4.5.2 データモデルの拡張
    - 4.6 推奨される拡張フィールド
      - 4.6.1 'courses.csv' データモデル
      - 4.6.2 'orgs.csv' データモデル
      - 4.6.3 'users.csv' データモデル
  - 5. RESTベースの交換のベストプラクティス
    - 5.1 バージョン1.0と1.1の間のREST APIの互換性
    - 5.2 認証
    - 5.3 コレオグラフィーのインプリケーションの読み取り、書き込み、削除
    - 5.4 エンドツーエンドのサービス機能のマッチング
    - 5.5 評価表の交換
    - 5.6 REST APIの拡張
  - 6. ユースケースの実現
    - 6.1 K-12 情報のための学区リポジトリの確立と維持
    - 6.2 名簿情報を利用可能にする
    - 6.3 リソース割り当ての識別
    - 6.4 評価表情報のSIS / LMS交換
    - 6.5 OneRoster情報のリアルタイムの一括交換
    - 6.6 OneRoster情報の一括バッチ交換
- Appendix A 主要な用語と定義
- このドキュメントについて
- 貢献者のリスト

# 1. はじめに

## 1.1 スコープとコンテキスト

このドキュメントの目的は、IMS OneRoster 1.1標準を採用するために収集および照合されたベストプラクティスの推奨事項を提供することです。この情報は、標準の開発中、およびOneRosterを採用したIMSメンバーのフィードバックから作成されました。

## 1.2 このドキュメントのステータス

このドキュメントは最終リリースです。つまり、技術的ソリューションが公開ドキュメントとして利用可能になり、複数のIMSメンバーが仕様のリリース時に適合性認定を成功裏に完了しています。

IMSは、OneRoster標準の進化と改善を継続するためにフィードバックを提供することをメンバーとコミュニティに強く推奨しています。OneRosterに焦点を合わせたIMS開発者および適合性認定コミュニティに参加するには、IMS OneRoster アライアンスにオンラインでアクセスしてください：

<http://www.imsglobal.org/lis/index.html>

パブリックコントリビューション、コメント、質問は、こちらから投稿できます：[パブリック OneRoster フォーラム](#)。

## 1.3 このドキュメントの構造

このドキュメントの残りの構造は次のとおりです：

- |                                 |   |
|---------------------------------|---|
| 2. OneRoster 相互運用性アーキテクチャー      | OneRoster ベースのデータ交換のための想定アーキテクチャー。              |
| 3. Common REST/CSV 導入のベストプラクティス | REST/JSON サービスと CSV ファイルバインディングの両方に関連する一連の推奨事項。 |
| 4. CSV ファイル処理のベストプラクティス         | CSV ベースのデータ交換を採用するための一連の推奨事項。                   |
| 5. REST ベースの交換のベストプラクティス        | REST/JSON サービスバインディングの採用に関する一連の推奨事項。            |
| 6. ユースケースの実現                    | OneRoster 標準を使用して多くのユースケースを実現する方法の説明。           |
| Appendix A - 主要な用語と定義           | OneRoster で使用されるさまざまな主要な用語と定義の簡単な説明。            |

## 1.4 関連ドキュメント

- [AfA PNP, 10] すべてのデジタルパーソナルニーズへのアクセスとデジタル配信情報モデル2.0の設定の説明、R.シュベルトフェガー、M.ロスベルグおよび C.スマイス、IMS グローバルラーニングコンソーシアム、2010年4月、[https://www.imsglobal.org/accessibility/accpnpv2p0/spec/ISO\\_ACCPNPInfoModelv2p0.html](https://www.imsglobal.org/accessibility/accpnpv2p0/spec/ISO_ACCPNPInfoModelv2p0.html)。
- [OneRoster, 17a] OneRoster 仕様および REST バインディング v1.1、P.ニコルスおよび C.スマイス、IMS グローバルラーニングコンソーシアム、2017年4月、<http://www.imsglobal.org/lis/imsonerosterv1p1/imsOneRoster-v1p1.html>。
- OneRoster, 17b] OneRoster 1.1 CSV テーブル、P.ニコルスおよび C.スマイス、IMS グローバルラーニングコンソーシアム、2017年4月、<https://www.imsglobal.org/lis/imsOneRosterv1p1/imsOneRosterCSV-v1p1.html>。
- [OneRoster, 17c] OneRoster 1.1 適合および認証、C.スマイスおよび P.ニコルス、IMS グローバルラーニングコンソーシアム、2017年4月、<https://www.imsglobal.org/lis/imsOneRosterv1p1/imsOneRosterConformance-v1p1.html>。
- [RFC1951] DEFLATE 圧縮データフォーマット仕様 1.3 (RFC 1951)、P.デュッシュ、IETF1996年5月、<https://www.ietf.org/rfc/rfc1951.txt>。

[RFC3629] UTF-8: ISO 10646 (RFC 3269) の変換フォーマット、F.イエルガー、IETF, November 2003, <https://www.ietf.org/rfc/rfc3269.txt>。

[RFC4180] CSV ファイルの共通フォーマットとMIMIタイプ (RFC 4180)、Y.シャフランovich、IETF、2005年10月、<https://www.ietf.org/rfc/rfc4180.txt>。

## 1.5 命名法

AfA PNP	Access for All Personal Needs and Preferences
BOM	バイトオーダーマーク、Byte Order Mark
CEDS	Common Education Data Standards
CSV	Comma Separated Values
GUID	グローバル一意識別子、Globally Unique Identifier
IETF	インターネット技術特別調査委員会、Internet Engineering Task Force
JSON	JavaScript Object Notation
LDAP	Lightweight Directory Access Protocol
LMS	学習管理システム、Learning Management System
LTI	Learning Tools Interoperability
NCES	全米教育統計センター、National Center for Educational Statistics
OR	OneRoster
ORCA	OneRoster Consumer API
REST	Representational State Transfer
RFC	リクエスト フォー コメンツ、Request For Comment
SIS	学生情報システム、Student Information System
TLS	トランスポート・レイヤー・セキュリティ、Transaction Layer Security
UTF	Unicode Transformation Format
UUID	Universally Unique Identifier

[コラムの先頭へ](#) | [先頭へ](#)

## 2. OneRoster 相互運用性アーキテクチャー

すべてのIMS仕様と同様に、OneRoster仕様は、動作中のデータ、つまり合意された相互運用性を使用して達成された情報の交換について説明しています。OneRosterの場合、交換される情報は3つのグループに分類されます。

- クラス名簿 - 現場のクラスに一定期間登録された一連の人々。
- リソース - クラスやコースに必要なリソースのセットを特定します。
- 評定表 - データは結果、lineItems（一連の結果）、およびカテゴリ（一連のlineItems）に分割されません。

OneRoster相互運用性の基礎となるモデルを図2.1に概略的に示します。

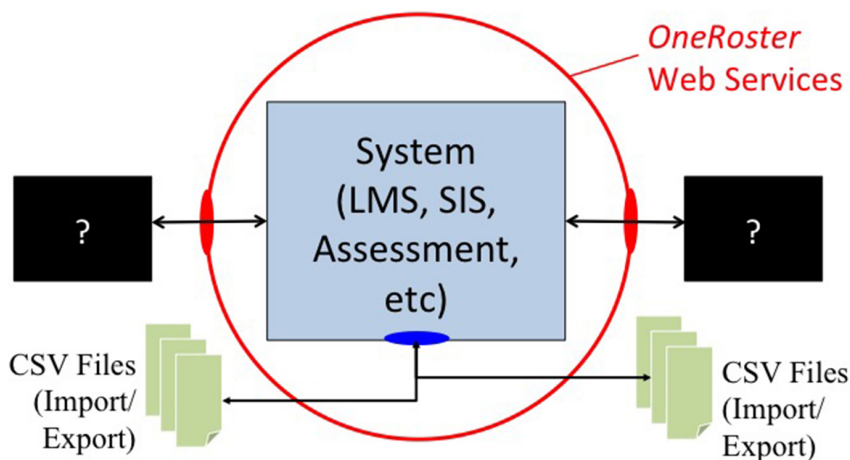


図2.1 - OneRoster相互運用性アーキテクチャー

OneRosterには、2つのデータ交換モードがあります。

- CSVファイル - データは、zipファイルとして交換される一連のCSVファイルに含まれています。システム間でデータを移動する方法は定義されていないため、実装ごとに異なる場合があります。FTP、電子メールなどを使用します。システムは、データのインポート、データのエクスポート、またはインポートとエクスポートの両方を提供します。インポート/エクスポートの表現は、システム内のデータの内部ストレージなどの可能性が低いために使用されます。LMS、SISなどはCSV形式を使用します。
- REST API - 交換はHTTP呼び出しで伝送される一連のJSONペイロードとして定義されます（これはRESTful交換モデルに基づいています）。REST APIには、そのデータをリアルタイムで交換する方法を定義する広範なサービスエンドポイントのセットがあります。ほとんどのエンドポイントはサービスプロバイダーからのデータの読み取りを許可しますが、評価表データの場合、サービスプロバイダーがサービスユーザーにデータを書き込んだり、サービスユーザーからデータを削除したりすることが可能です。REST APIには、大きなペイロードのページング、ペイロード内のデータの並べ替え、一連のフィルタリングルールによるデータの選択、完全なペイロードのサブセットのみの戻り値の指定のサポートが含まれています。

CSV交換の主な使用例は、相当量のデータをサービスプロバイダーにインポートできるようにすること、サービスプロバイダーでデータセットを定期的に再同期することです。REST APIの主な使用例は、コンシューマー/クライアントが読み取り要求/応答サービスアプローチを使用してデータを読み取れるようにすることです。評価表交換の場合、サービスプロバイダーが書き込み/確認アプローチを使用してコンシューマー/クライアントにデータをプッシュできるように拡張されました。

したがって、サービスプロバイダーは通常、REST APIがリアルタイムの交換機能を提供するCSVとRESTの両方の交換方法をサポートできます。

カラムの先頭へ | 先頭へ

## 3. 共通REST/CSV導入のベストプラクティス

### 3.1 識別子とその使用

#### 3.1.1 ソース識別子

「sourcedId」属性は、通信するエンドシステム間で関連付けられているオブジェクトを一意に識別するために使用されます。理想的には、これらはグローバルに一意の識別子である必要があります（内部形式/構造は実装に依存する問題であり、ユニバーサル一意識別子（UUID）の形式に制限されません）。識別子は内部データストアキーとして使用しないでください。つまり、識別子とキーの間のマッピングは実装で維持する必要があります。

すべてのデータオブジェクトには一意の識別子、つまりその「sourcedId」が割り当てられます。この識別子は、オブジェクトにアクセスする2つのシステムのコンテキストで一意である必要があります。つまり、識別子はグローバルに一意である必要はありません。エンドシステムは、これらの識別子のコンテキストの整

合性を維持する責任があります。「sourcedId」は、CSVファイルの処理のために関連付けられたレコードを識別できるようにすることを除いて、エンドシステムの重要性を持つ必要はありません。

「削除された」レコードの定義に使用されるのは、特定の「sourcedId」のレコードを見つけることができないことです。したがって、エンドシステムには、関連するレコードを識別するための独自の一意のキーが必要です。これにより、一貫した相互運用性の削除を提供するために、レコードをエンドシステムから削除する必要がなくなります。

オブジェクトが削除されている場合、関連付けられた「sourcedId」を別のオブジェクトに再割り当てしてはなりません。したがって、「sourcedId」が割り当てられると、関連付けられたオブジェクトに永続的に割り当てられ、「削除された」データの回復に使用できます。

「sourcedId」は、一意の相互運用性キーとして使用されます。データが複数のソースから受信される、すなわちCSVファイルの多くのセットのインポートし、集計する場合、同じ「sourcedId」が異なるオブジェクトに使用される可能性があります（各ソースにはいくつかの「sourcedId」生成/割り当てメカニズムがあるため、クラッシュの可能性はこれらのアプローチの高度化）。偶発的なデータの上書きを最小限に抑えるために、システムではアクセス制御メカニズムを使用することをお勧めします。たとえば、REST APIでは、既存のオブジェクトに対するHTTP PUTリクエストにより、そのオブジェクトが新しいデータセットに置き換えられます。したがって、書き込み要求は、HTTPコード403（禁止）を返すアクセス制御ルールに失敗した要求を含む適切なアクセス制御メカニズムによってモデルートする必要があります。

### 3.1.2 リソースVendorId

これは、ベンダーを一意に識別する識別子である必要があります。実際の「vendorId」には、「vnd.\*\*\*」の2つのコンポーネントがあります。

文字列「\*\*\*」は、OneRoster適合プログラムの一部としてIMSによって割り当てられます。

### 3.1.3 ユーザーID

ユーザークラスには、いくつかの識別子フィールドがあります。IMS仕様に準拠して、すべてのシステムで可能な限り「sourcedId」フィールドを使用することをお勧めします。つまり、SISはユーザーの「sourcedId」をLMSに提供する必要があります。LMSは、他の識別子を提供するのではなく、LTIを介して起動するすべてのツールにそのユーザーの同じ「sourcedId」を提供する必要があります。これは、すべての組織とロールに当てはまります。

ただし、現時点ではこれが現在の慣行ではないことがわかっているため、他の識別子がユーザークラスに追加され、ベストプラクティスへの移行が可能になっています。

「userId」フィールドは、特定のユーザーのその他の機械可読「id」を格納できるブリッジングピースとして意図されています。これは、LTI IDの場合もあれば、アクティブディレクトリまたはLDAP IDの場合もあります。LMSが外部呼び出しを行うときにユーザーの「sourcedIds」を使用できない場合は、代わりにこのフィールドを使用できます。1.1では、複数の「userId」の割り当てのサポートが追加されました。各IDには割り当てられたタイプがあります（このタイプのフィールドには事前定義された語彙がないため、LDAP、LTIなど、適切な語彙を確立するのはローカルのデプロイメントに任されています。）。

「識別子」と呼ばれるフィールドも含まれています。これは、純粋に人間が何らかの理由で特定のユーザーを参照するときに使用するためのものです。これは機械で読み取り可能なフィールドではないため、人間が読み取れる文字列以外のものとして処理しないでください。

### 3.1.4 学校と学区の識別子

これらの識別子は、特定のインスタンスを参照するときに人間が使用するためだけのものです。これらのフィールドは機械可読に設計されておらず、人間が読める文字列以外のものとして処理されるべきではありません。

NCES 識別子：米国では、全米教育統計センター（NCES）が、NCESに報告するときに使用される学校および学区の識別子を発行します。これらの識別子を、それらが属する適切な組織で使用することがベストプラクティスと見なされています。たとえば、高校の記録では、人間が読める「識別子」フィールドにその高校のNCES IDを含める必要があります。

米国以外の管轄区域は、その地域の適切な識別子について合意する必要があります。

## 3.2 ユーザークラスの「enabledUser」属性

このフィールドは、OR 1.0の「ステータス」フィールドに由来します。「users.csv」で使用されるOR 1.0の「status」フィールド値について説明する際、デルタモードの送信がバルクモードで表現できるよりも多くの情報を運ぶという問題が確認されました。デルタモードでは、「ステータス」フィールドを使用して、ユーザーを削除する必要があること、現在アクティブであること、および現在はアクティブではないことを区別できます。バルクモードでは、「行はcsvに存在します」という2つの状態があります。これは、「アク

ティブ」のデルタステータスに最もよく対応します。「行は以前の送信にありましたが、CSVから欠落しています」「tobedeleted」のデルタステータスに最もよく対応します。したがって、ステータスフィールド、非アクティブフィールドには完全な値があり、OR 1.0のバルク送信ではモデル化できません。

さらに議論を重ねた結果、「アクティブ」と「削除予定」の区別はデータ交換プロトコルの問題であり、「アクティブ」と「非アクティブ」の区別は有用なエンティティデータであることがわかりました。さらなる調査の結果、アクティブと非アクティブの区別は、ユーザーのみを対象としたものでした。

これらの観察に基づいて、すべてのファイルのステータスとして「非アクティブ」を排除し、ORクラスの「アクティブ」ステータスと「非アクティブ」ステータスの違いを表す新しいフィールド「enabledUser」をユーザークラスに追加することが決定されました。OR 1.1では、バルクファイルが3つの状態すべてを表現できないという問題が修正されています。これで、「行が存在する」と「行が欠落している」が「アクティブ」と「削除予定」を示し、「users.csv」の場合、「enabledUser」フィールドはOR 1.0の「アクティブ」および「非アクティブ」の違いを示します。

OR 1.0デルタ送信ではステータスフィールドが必須だったため、OR 1.1では「enabledUser」フィールドが必須でした。「非アクティブ」ステータスを受け入れて理解する機能はOR 1.0でも必須でした。そのため、プロセッサが「enabledUser」フィールドを受け入れることを要求するOR 1.1にそれ以上の制約を追加しません。

### 3.3 ステータスフィールドの値の意味

ステータスフィールドに「アクティブ」および「削除予定」の値が許可されました。つまり、OR 1.1の「非アクティブ」トークンは削除されました。OR 1.0との下位互換性のために、「非アクティブ」値は「tobedeleted」として扱われる必要があります。次の処理ルールが適用されます。

- バルクモードでは、データベースに格納されている、読み取り/インポートされるデータに含まれていない対応するタイプのオブジェクトのステータスは、「tobedeleted」に設定されている必要があります。
- 「totodeleted」として設定されているレコードと実際の削除アクションの間の時間の長さは実装に依存します。
- オブジェクトが削除されたら、その「sourcedId」を別のオブジェクトに再割り当てしてはなりません。つまり、同じ「sourcedId」に対するバルク/デルタアクションは、元のオブジェクトの回復（ハード削除が発生していない場合）または失敗の応答（完全削除が発生した場合）を引き起こします。

サービスプロバイダーでの一括処理/差分処理を使用したレコード管理の状態図を表2.1に示します [OneRoster, 17b]。

[コラムの先頭へ](#) | [先頭へ](#)

## 4. CSVファイル処理のベストプラクティス

### 4.1 ファイルのエンコーディングとファイルの順序

各データファイルのファイル形式は、RFC 4180 [RFC4180] で指定されているコンマ区切り値形式 (CSV) で、フィールド内で改行を許可しないという追加の制限があります。コンマと二重引用符を含むフィールドは、二重引用符で囲む必要があります。フィールドを囲むために二重引用符を使用する場合、フィールド内に表示される二重引用符は、その前に別の二重引用符を付けることでエスケープする必要があります。

CSVファイルはUTF-8エンコードされている必要があります [RFC3629]。インポートするプロセッサは、BOM (バイトオーダーマーク) プレフィックスを許容し、それらを無視する必要があります。BOMを含むUTF-8エンコードファイルでは、BOMはファイルの先頭に3バイトのシーケンス「EF BB BF」として表示されます。存在する場合、CSVヘッダーはファイルの4番目のバイトから始まります。存在しない場合、CSVヘッダーはファイルの最初のバイトから始まります。

CSVファイルとしてデータを交換する場合、1~14個のCSVファイルがあります。エクスチェンジには「manifest.csv」ファイルを含める必要があります。「manifest.csv」のみが指定されている場合は、更新が行われなかったことを示します。最大で14個のCSVファイルが交換されます。つまり、各データCSVファイルが複数存在してはなりません。ファイルのセットが一括処理用である場合、それらは意味的に一貫している必要があります。つまり、ファイル内で参照されるすべてのオブジェクトは、zipパッケージ内の同じファイルまたは他のファイルの1つでも定義されている必要があります。

CSVファイルはzipファイルとして交換されます。含まれるファイルは、ルートレベルにある必要があります。つまり、それらは、包含ディレクトリ内に含まれてはなりません。zipファイルの名前に制約はありませんが、「zip」のファイル拡張子が必要です。圧縮はRFC 1951 [RFC1951] に準拠する必要があります

す。zipファイル内では、CSVファイルの優先順位はありません。

## 4.2 CSVファイルの交換

CSVファイルは、zipファイルとして交換する必要があります。これらのファイルは暗号化しないでください。暗号化すると検証に合格しません。トランスポートメカニズムはデータの暗号化を担当し、CSVファイルの整合性を保護します。

トランスポートメカニズムは、CSVバインディングの範囲外です。

## 4.3 V1.1とV1.0の間の往復

データを失わずにOneRosterバージョン間の往復をサポートすることをお勧めします。たとえば、組織がOR1.1 CSVファイルをOneRosterゲートウェイに送信して、v1.1形式のv1.1をサポートするベンダーに送信するが、v1.0のみをサポートするベンダーにデータを送信する必要がある場合を考えます。それをv1.0形式で送信しますが、v1.0仕様の一部ではないフィールドを破棄するのではなく、これらは同じ名前のメタデータフィールドとしてエンコードされます。

たとえば、v1.1の「users.csv」にgrade列のデータが含まれている場合、それを同等のORv1.0 CSVの「metadata.ornv1p1.grade」列に変換します。1.0に準拠したシステムは、グレードのセマンティクスを理解できない場合がありますが、grade情報を必要とするエンドユーザーへのグレードを含むメタデータフィールドの表示をサポートしている可能性があります。また、システムで検索フィルターにメタデータを含めることができる場合にも使用できます。フィールドは、「metadata.ornv1p1.grade」でフィルタリングすることにより、grade情報を利用します。

v1.0のCSVをv1.1に変換する場合、「metadata.\*\*\*」を含むすべての列を列の最終セットとして配置する必要があります（順序は重要ではありません）。

## 4.4 セマンティックの一貫性の確保

バルク交換の場合、CSVファイルのセットは意味的に完全でなければなりません。つまり、オブジェクトの「sourcedId」参照がCSVファイルの1つで使用されている場合、そのオブジェクトはCSVファイルのセットのどこかに定義されている必要があります（参照が発生するCSVファイル）。CSVファイルのデータモデルは、以下のファイルのグループが付随する制約とともに交換される可能性があることを意味します。

1. 'academicSessions.csv' - このファイルには他のAcademySessionsへの参照が含まれている可能性があるため、そのような定義はすべて同じファイル内に含まれている必要があります。このCSVファイルは、他のファイルとは独立して交換できます。
2. 'categories.csv' - このファイルには他のファイルへの参照が含まれていないため、他のファイルとは独立して交換できます。
3. 'classes.csv' - このファイルには、関連するacademicSession、コース、および組織オブジェクトへの参照が含まれている必要があります。したがって、「academicSessions.csv」、「courses.csv」、および「orgs.csv」ファイルを伴う必要があります。
4. 'classResources.csv' - このファイルには、関連するacademicSession、クラス、組織、およびリソースオブジェクトへの参照が含まれている必要があります。したがって、「academicSessions.csv」、「classes.csv」、「orgs.csv」、および「resources.csv」ファイルを伴う必要があります。
5. 'courseResources.csv' - このファイルには、関連するacademicSession、コース、組織、およびリソースオブジェクトへの参照が含まれている必要があります。したがって、「academicSessions.csv」、「courses.csv」、「orgs.csv」、および「resources.csv」ファイルを伴う必要があります。
6. 'demographics.csv' - このファイルには、関連する組織およびユーザーオブジェクトへの参照が含まれている必要があります。したがって、「orgs.csv」および「users.csv」ファイルを添付する必要があります。
7. 'enrollments.csv' - このファイルには、academicSession、クラス、コース、組織、およびユーザーオブジェクトへの参照が含まれている必要があります。したがって、「academicSessions.csv」、「classes.csv」、「courses.csv」、「orgs.csv」、および「users.csv」ファイルを伴う必要があります。
8. 'lineItems.csv' - このファイルには、academicSession、カテゴリ、クラス、コース、組織、およびユーザーオブジェクトへの参照が含まれている必要があります。したがって、「academicSessions.csv」、「categories.csv」、「classes.csv」、「courses.csv」、「orgs.csv」、および「users.csv」ファイルを伴う必要があります。
9. 'courses.csv' - このファイルには、関連する組織への参照が含まれている必要があり、academicSessionオブジェクトへの参照が含まれている場合があります。したがって、「orgs.csv」ファイルと交換する必要があり、「academicSessions.csv」ファイルが必要になる場合があります。
10. 'orgs.csv' - このファイルには他の組織への参照が含まれている可能性があるため、そのような定義はすべて同じファイル内に含まれている必要があります。このCSVファイルは、他のファイルとは独立し



て交換できます。

11. 'resources.csv' - このファイルには他のファイルへの参照が含まれていないため、他のファイルとは独立して交換できます。
12. 'results.csv' - このファイルには、academicSession、カテゴリ、クラス、コース、lineItem、組織、およびユーザーオブジェクトへの参照を含める必要があります。したがって、「academicSessions.csv」、「categories.csv」、「classes.csv」、「courses.csv」、「lineItem.csv」、「orgs.csv」、および「users.csv」ファイルを伴う必要があります。
13. 'users.csv' - このファイルには、関連する組織への参照が含まれている必要があり、他のユーザーオブジェクトへの参照が含まれている場合があります（それらの定義は同じファイル内に含まれている必要があります）。したがって、「orgs.csv」ファイルと交換する必要があります。

このファイルのグループ化の詳細については、OneRoster CSV定義 [OneRoster, 17b] を参照してください。ただし、実装では、意味的に一貫性のあるファイルの最小セットより多くのものが必要/供給される場合があることに注意してください。たとえば、次の一連のファイルを要求/提供する実装に遭遇することは珍しいことではありません。

- 名簿交換の場合 - 'academicSessions.csv', 'classes.csv', 'courses.csv', 'enrollments.csv', 'orgs.csv' および 'users.csv' ファイル。
- リソース交換の場合 - 'academicSessions.csv', 'classes.csv', 'classResources.csv', 'courseResources.csv', 'courses.csv', 'enrollments.csv', 'orgs.csv' および 'resources.csv', 'users.csv' ファイル。
- 評定表交換の場合 - 'academicSessions.csv', 'categories.csv', 'classes.csv', 'courses.csv', 'enrollments.csv', 'lineItems.csv', 'orgs.csv', 'results.csv' および 'users.csv' ファイル。

## 4.5 CSVファイルのデータモデルの拡張

### 4.5.1 ボキャブラリーの拡張

列挙型のボキャブラリーは拡張してはいけません。

オープンボキャブラリーは拡張されるかもしれません。これらの拡張はURIの形式を取る必要があるため、用語の作成者も提供されます。

### 4.5.2 データモデルの拡張

CSV交換の場合、データモデルを拡張するには、定義された列のセットの最後に新しい列を追加する必要があります。新しいヘッダー名には、適切な命名規則を使用する必要があります。例は次のとおりです。

metadata.[orglabel].[namepart]

場所：

- 「メタデータ」は、すべての拡張機能の最初の文字セットです。
- [orglabel]は、拡張機能を作成した組織を識別するために使用されるラベルです。IMSグローバルからの拡張の場合、このフィールドは空になり、区切り文字の1つが削除されます。
- [namepart]組織によってデータモデルに割り当てられた一意のヘッダー名部分。

例：'metadata.hmh.homeemail'、'metadata.pearson.namesuffix'など

## 4.6 推奨される拡張フィールド

### 4.6.1 'courses.csv'データモデル

「courses.csv」ファイルの推奨メタデータフィールドは次のとおりです。

フィールドヘッダー	必須	フォーマット	説明
metadata.duration	No	String	コースの実行時間の説明（2週間、1学期など）。

### 4.6.2 'orgs.csv'データモデル

「orgs.csv」ファイルの推奨メタデータフィールドは次のとおりです。

フィールドヘッダー	必須	フォーマット	説明
metadata.classification	No	String	「チャーター」   「プライベート」   「公衆」
metadata.gender	No	String	「女性」   「男性」   「混合」
metadata.boarding	No	Boolean	学校が寄宿学校の場合はブール値「true」。

フィールドヘッダー	必須	フォーマット	説明
metadata.address1	No	String	組織の自由形式の住所の最初の行。例、「80アイアンポイントサークル」。
metadata.address2	No	String	組織の自由形式の住所の2行目。例、「部門2100インチ」。
metadata.city	No	String	組織が運営されている都市。例、ビスマルク。
metadata.state	No	String	組織が運営されている都道府県の郵便略称。「ND」。米国の住所のポキャブラリー： <a href="https://ceds.ed.gov/CEDSElementDetails.aspx?TermxTopicId=20837">https://ceds.ed.gov/CEDSElementDetails.aspx?TermxTopicId=20837</a> 。カナダの住所の語彙： <a href="https://www.canadapost.ca/tools/pg/manual/PGaddress-e.asp?ecid=murl10006450#1442218">https://www.canadapost.ca/tools/pg/manual/PGaddress-e.asp?ecid=murl10006450#1442218</a> 。
metadata.postCode	No	String	組織の住所の郵便番号。（米国の住所の場合）「91311」または「91311-5349」、（カナダの住所の場合）「K1A 0B1」。
metadata.country	No	String	組織が運営されている国。 <a href="https://ceds.ed.gov/CEDSElementDetails.aspx?TermxTopicId=20002">https://ceds.ed.gov/CEDSElementDetails.aspx?TermxTopicId=20002</a> にあるCEDSポキャブラリーを使用してください。

#### 4.6.3 'users.csv'データモデル

「users.csv」ファイルの推奨メタデータフィールドは次のとおりです。

フィールドヘッダー	必須	フォーマット	説明
metadata.pnpfileurl	No	String	これは、すべての個人のニーズと設定ファイルに対するユーザーのアクセスのURLです。このファイルは、ユーザーが使用するコンピューターシステムのユーザー補助機能の構成設定を定義します。このURLで識別されるファイルの内容は、関連するIMS AfA PNP 2.0仕様 [AfA PNP, 10]で定義されています。

[カラムの先頭へ](#) | [先頭へ](#)

## 5. RESTベースの交換のベストプラクティス

### 5.1 バージョン1.0と1.1の間のREST APIの互換性

OneRoster 1.0として認定されたシステムは、OneRoster 1.1への準拠について個別に認定される必要があります。OR 1.1の適合性テストのためにOR 1.0システムを更新するには、次の変更が必要です。

- ベースURLを「/ learningdata / v1」から「/ ims / oneroster / v1p1」に変更する必要があります。
- 必要なエンドポイント（名簿、リソース、評価表の3つのグループに分類）の再定義を検査する必要があります（OR 1.0では、サブグループ化はなく、エンドポイントのサブセットのみのサポートが必要です）。新しいエンドポイントのセットが追加されました。
- 詳細なデータモデルを調査する必要があります。データモデルにいくつかのオプションの追加属性が作成されました。データモデルの一部の属性は名前が変更され、一部はそのタイプが改良されています。これらにより、一部のJSONペイロードに大きな違いが生じています。
- データを書き込むための評価表エンドポイントの場合（HTTP PUT動詞を使用）、ペイロードは追加のOAuthパラメーターを使用して署名する必要があります。

### 5.2 認証

OneRoster 1.1とOneRoster1.0は、適合のために同じメッセージ署名メカニズム、つまりTLS 1.2上のHMAC-SHA1を備えたOAuth 1.0aを必要とします。OneRoster 1.1は以下も許可します

- TLS 1.2のHMAC-SHA1とは対照的に、HMAC-SHA2（256）のサポート。
- メッセージの暗号化にTLS 1.2を使用する承認のために、RFC 6750で定義されているOAuth 2ベアラートークンを使用する。

すべての実装は、適合のためにTLS 1.2上のHMAC-SHA1でOAuth 1.0aをサポートする必要がありますが、HMAC-SHA256を使用することを強くお勧めします（これに対するオプションの適合も利用可能です）。OR 1.2の場合、許可される唯一の認証メカニズムはOAuth 2ベアラートークンになります（IMS Globalはすべての認証メカニズムをOAuth 2に基づくように調整しています）。

### 5.3 コレオグラフィーの意味の読み取り、書き込み、削除

OR 1.1では、読み取り、書き込み、削除のコレオグラフィーは、評価表サービスのエンドポイントでのみ問題になります。読み取りエンドポイントのセットはデータの「プル」機能を提供しますが、書き込み/削除は「プッシュ」機能を提供します。この仕様は特定の振付を規定していません。コレオグラフィーは特定のワークフローを実現するために定義されていますが、OR仕様は幅広いワークフローを可能にするように設計されています。OR 1.1仕様 [OneRoster, 17a] は、プルエンドポイントとプッシュエンドポイントの簡単な状態図を提供します。通常、ワークフローには「プル」エンドポイントと「プッシュ」エンドポイントの混在が必要です。ワークフローを実現する際に考慮すべきいくつかの問題は次のとおりです。

- 書き込みエンドポイントと削除エンドポイントは、一度に1つのオブジェクトのみを操作できます。したがって、REST APIを大規模な削除とデータ作成に使用しないでください。
- オブジェクトが「削除」された場合、「sourcedId」は再割り当てされてはならず（MUST）、その「sourcedId」に対する読み取りリクエストは読み取りエラーを引き起こさなければなりません（HTTP 404コードを使用）。
- プロバイダーでの読み取り/書き込み要求の重複を回避するように注意する必要があります。つまり、プロバイダーは、プロバイダーによって処理されている不完全な読み取り要求を持つコンシューマーに書き込み要求を発行しないでください。
- コンシューマーでの読み取り/書き込み要求の重複を回避するように注意する必要があります。つまり、コンシューマーは、受信した書き込み/削除要求が並行して不完全な読み取り要求と競合しないようにする必要があります。競合を回避する方法は実装に依存します。

### 5.4 エンドツーエンドのサービス機能のマッチング

OR 1.1 REST実装で利用可能な認定を表5.1にまとめます。

Table 5.1 OneRoster REST準拠製品で利用可能な認定の比較

機能モード (少なくとも1つはサポートされている必要があります)	REST API			
	プロバイダー		コンシューマー	
	コア	その他	コア	その他
名簿	必須	オプション	必須	オプション
リソース	必須	オプション	必須	オプション
評価表	必須（プルまたはプッシュ）	オプション	必須（プルまたはプッシュ）	オプション

したがって、2つの認定されたOR 1.1 RESTシステムがデータを交換できるかどうかを照合する場合は、次のルールを適用する必要があります。

- プロバイダー/コンシューマーのペアは、操作の同じモードのペアリング（名簿/名簿、リソース/リソース、評価表（プル）/評価表（プル）および評価表（プッシュ）/評価表（プッシュ）ペアリング）をサポートしている場合、コアエンドポイントの相互運用性があります。
- 一般的なペアリングの場合、プロバイダー/コンシューマーの両方でサポートされる任意のエンドポイントの相互運用性もあります。
- 必須のデータ属性のみが交換されることが保証されています。オプションのデータ属性のさまざまな組み合わせをサポートするシステムは異なるため、プロバイダー/コンシューマーの機能間のマッピングを行う必要があります（この情報は、IMS認定レポートまたはIMS製品データベースからは入手できません）；
- すべてのREST実装は、OAuth 1.0a/SHA1メッセージ署名をサポートする必要があります。他の認証メカニズム（OAuth 1.0a/SHA256およびOAuth 2 Bearer Token）のサポートはオプションであり、プロバイダー/コンシューマーの両方がそのメカニズムをサポートすることが認定されている場合のみ使用できます。

### 5.5 評価表の交換

#### 5.5.1 課題評価転送のユースケース

割り当てレベルの成績の転送は、州の報告要件に SIS テクノロジーを使用する幼稚園から高校までの学区や学校にとって重要です。通常、LMS は割り当てデータの記録システムとして機能し、オンデマンドまたはスケジュールベースでの割り当て評価の転送を容易にします。SISプロバイダーは評価データコンシューマーとして機能し、エンドポイントを公開して LMS からの割り当て評価データを受け入れます。OneRoster 仕様v1.1の評価表サービスは、特定の RESTful エンドポイントのセットを提供して、評価データの転送を容易にします。

OneRoster 仕様v1.1の評価表サービスでは、割り当て評価データの転送を容易にするために、特定の RESTful エンドポイントのセットを推奨しています。

一般的な割り当て評価のデータ転送の実装では、LMS は「lineItem」および「result」オブジェクトの元のシステムとして機能します。大多数のSIS評価表では、「宿題」、「クイズ」、「エッセイ」などの「lineItem」格付けカテゴリによって「lineItem」オブジェクトを分離する必要があります。ベストプラクティスは、SISプロバイダーが「¥ categories」エンドポイントを介して SIS 評価表でエンドユーザーが使用できる「lineItem」カテゴリを公開することです。LMS プロバイダーは、オンデマンドまたはスケジュールに基づいてカテゴリを要求し、エンドユーザーが LMS で SIS LineItem カテゴリの用語と sourcedId を維持できるようにします。

一般的に、LMSは、オンデマンドまたはスケジュールベースで SIS 評価表への課題の成績の転送を要求するユーティリティをエンドユーザーに提供する必要があります。評価の転送が要求されると、LMSはLineItem および/または LineItem Result データセットを組み立て、それを SIS 評価表にプッシュします。LMS は putLineItem および putResult RESTful エンドポイントを使用して、割り当ておよび割り当ての成績オブジェクトを SIS 評価表に挿入します。

次の使用例は、業界で一般的に受け入れられています。

**Step 1:** LMSは、GET カテゴリサービスコールを介してSISから LineItem カテゴリをインポートします。

1. SISは通常以下を提供します。
  - a. sourcedId (データの共謀を避けるために UUID または GUID を使用してください)
  - b. カテゴリのタイトル
  - c. カテゴリのステータス
  - d. メタデータオプション: 対応する sourcedId を介してコース、クラス、教師ユーザー、または評価表へのリンク (LMS プロバイダーは、LMS コース評価表の成績カテゴリへのアクセスに関する適切な権限を設定できます)
2. LMS は通常、オンデマンドまたは定期的に LineItem カテゴリをインポートします。
  - a. getCategories エンドポイントを介して SIS から LineItem カテゴリをリクエストします。
  - b. LMS で SIS LineItem カテゴリを作成する。
  - c. エンドユーザーがラインアイテムを SIS からインポートされた評価カテゴリに関連付けることを許可する。

**Step 2:** LMS は通常、PUT LineItem サービス呼び出しを介して SIS に LineItem オブジェクトを作成します。

1. SISで一般的に必要なとされるデータセット
  - a. LineItem sourcedId (データの衝突を回避するためにUUIDまたはGUIDを使用)
  - b. タイトル
  - c. 期日
  - d. クラス
  - e. カテゴリ
  - f. Result の値

**Step 3:** LMS は通常、各結果レコードに GUID または UUID を提供することにより、PUT Result サービスコールを介して SIS に Result オブジェクトを作成します。

**Step 4:** LMS は通常、PUT LineItem サービスコールを介して SIS の LineItem オブジェクトを置き換えます。

1. LMSを許可するSIS
  - a. 既存のラインアイテムレコードのすべてのデータ要素を新しい値に置き換えます。
  - b. ineItemデータ要素の値を複数回上書きする。
  - c. SIS からソフト削除された後で、lineItem レコードを復元します。例、エンドユーザーが誤って SIS の lineItem を削除したが、lineItem が LMS でまだアクティブな場合。

**Step 5:** LMS は通常、PUT Result サービスコールを介して SIS の Result オブジェクトを置き換えます。

1. LMS を許可するための SIS
  - a. Line Item Result データ要素の値を複数回上書きする。

- b. SIS からソフト削除された後、PUT Result サービスコールを使用して Result オブジェクトを復元します。例、エンドユーザーが誤って SIS のラインアイテムの結果を削除したが、ラインアイテムは LMS のアクティブな Result のままです。

**Step 6:** LMSは通常、DELETE LineItem サービスコールを介して SIS の LineItemオブジェクトを削除します。

1. ラインアイテムのステータスを「削除予定」に更新するためのSIS（ソフト削除の場合）。
2. SISはSIS評価表からラインアイテムを削除します（完全削除の場合）。

**Step 7:** LMS は通常、DELETE Result サービスコールを介して SIS の Result オブジェクトを削除します。

1. SIS は、提供された Result sourcedId に一致する Result レコードの Result ステータスを「tobedeleted」ステータスに更新します（ソフト削除の場合）。
2. SIS は、提供された Result sourcedId に一致する Result レコードを SIS 評価表から完全に削除します（完全削除の場合）。OneRoster は、以前に削除された Result レコードの sourcedId を再利用することを許可していないことに注意してください。

### 5.5.2 複数の評価表データプロバイダーと SourcedId データの共謀のユースケース

評価表データの利用者が複数の評価書データプロバイダーからの lineItem および結果データを受け入れる場合、評価表データの lineItem および Result レコードが誤って上書きされないように注意する必要があります。PUT/DELETEサービスコールは、すべてのレコードの「sourcedId」値の UUID または GUID を使用して実行する必要があります。評価表サービスデータの利用者は、評価表データの整合性を維持するために、さまざまなデータプロバイダーからの要求を一意に識別したい場合があります。

### 5.5.3 LMS および SIS ユースケースにおける評価分類の解釈

学習テクノロジーによっては、評価カテゴリオブジェクトに複数のデータモデルを含めることができます。評価カテゴリは通常、特定のトピックまたは値に従って学習オブジェクトをグループ化するために使用されます。評価カテゴリは、教員が学習活動のグループを比較検討し、各トピックがコースの総合成績にどの程度貢献しているかを計算するのにも役立ちます。SISでは、評価カテゴリは、システム管理者が設定した評価カテゴリライブラリ、教員が設定したカスタム評価カテゴリのコレクション、システムのデフォルトとして使用できる評価カテゴリのコレクション、またはすべてのオプションの組み合わせとして表すことができます。

評価カテゴリは、ユーザー、クラス、コース、学校、または学区に固有である可能性があります。SIS または LMS は、カテゴリを評価するための元のシステムである可能性があります。通常の設定では、SIS プロバイダーが評価カテゴリを所有し、LMS プロバイダーが定期的に評価カテゴリをインポートします。評価カテゴリが LMS にインポートされた後、教員はそれぞれのコースで評価カテゴリの使用を開始できます。理想的には、特定のクラスまたは特定のユーザーによって作成された評価カテゴリには、クラスとユーザー sourcedId が関連付けられているため、LMS はコースまたはクラス内の評価カテゴリを適切にマッピングし、エンドユーザーに手動のデータ構成の負担をかけることを回避できます。

## 5.6 REST APIの拡張

定義により、拡張機能の作成と使用により、相互運用できない機能が作成されます。拡張機能を使用して、サポートされている機能を置き換えることはできません。拡張機能は最後の手段として作成する必要があります。REST API を拡張する方法は2つあります。

- エンドポイントの追加 - 新しいエンドポイントを定義してAPIに追加できます。このようなエンドポイントは、確立されたエンドポイントのパターンに従い、関連付けられた JSON ペイロードを適切に定義し、必要に応じて、他のエンドポイントで使用されているのと同じデータモデル拡張メソッドを使用する必要があります。
- データモデルへの追加 - データモデルには、新しい属性をJSONペイロードに追加するときに使用する必要がある「メタデータ」クラスが含まれます（このクラスの使用については、OneRoster 1.1仕様 [OneRoster, 17a] で説明されています）。他の手法を使用して JSON ペイロードに属性を追加することは禁止されています。

拡張機能を作成したい組織は、IMS Global ([Lisa Mattson, IMS COO](#)) に連絡することをお勧めします。

[コラムの先頭へ](#) | [先頭へ](#)

## 6. ユースケースの実現

OR 1.1を使用してサポートできるいくつかのユースケースは次のとおりです。

- 幼稚園から高校までの情報のための学区リポジトリを確立および維持する。
- 名簿情報を利用できるようにする。
- リソース割り当ての識別。
- 評価表情報のSIS / LMS交換。
- OneRoster情報のリアルタイムのバルク交換。
- OneRoster情報の一括バルク交換。

## 6.1 幼稚園から高校までの情報のための学区リポジトリの確立と維持

大規模なデータ集約は、CSVインポートを使用して実現する必要があります。zipファイルデータをインポートできるようにする方法は実装に依存します。つまり、OR仕様の一部として定義されていません。CSVファイルのサプライヤーは、「sourcedId」で各レコードを割り当て/識別する責任があります。したがって、zipファイルが複数のサプライヤー/ソースから提供されている場合、「sourcedId」の競合が発生する可能性があります。バルクモードを使用したCSVファイルのセットのインポートは、データセットの確立/再定義に使用されますが、デルタモードは、データセットの更新に使用されます。複数のソースからのデータの集約は、データの分離/整合性が適切に管理されるように注意深く管理する必要があります。

## 6.2 名簿情報を利用可能にする

名簿データは、REST APIを介して、またはエクスポートされたCSVファイルのセットとして利用できます。エクスポートされたCSVファイルは、リアルタイムの可用性が必要ない場合や、データセットのサイズが大きすぎる場合（ギガバイトなど）に使用する必要があります。CSVファイルを含むzipファイルが作成されると、ファイルを交換する方法は実装に依存します。REST APIは、エンドポイントに応じてデータセットのコレクションを利用できるようにします。つまり、1つのエンドポイントからすべての「ユーザー」オブジェクトを利用でき、すべての「コース」オブジェクトを取得するために別のエンドポイントが使用されます。したがって、混合セットを取得します。オブジェクトの場合、複数のエンドポイントを使用する必要があります（対応するコレオグラフィーは実装に依存します）。REST APIは、名簿データの読み取りのみを許可します。つまり、コンシューマーはプロバイダーに読み取り要求を発行する必要があります。

## 6.3 リソース割り当ての識別

OR 1.1の新機能は、クラスやコースに必要なリソースのセットを識別する機能です。この識別は、リソースのGUIDを使用します。つまり、実際のリソースは含まれません（リソースを取得する1つの方法は、識別子を介してリソースへのアクセスを提供するIMS共通カートリッジ/シン共通カートリッジを使用することで）。リソースIDは、CSVベースまたはREST APIベースの交換を使用して提供できます。

## 6.4 評価表情報のSIS/LMS交換

OR 1.1の主要な新機能。評価表情報の交換に対する広範なサポートです。CSVベースとREST APIベースの両方のアプローチを利用できます。REST APIの場合、「プッシュ」エンドポイントと「プル」エンドポイントの両方を使用できます。多くの実装では、プッシュアプローチとプルアプローチの両方を使用したコレオグラフィーがあります（これを実現する方法の詳細については、[セクション5.5](#)を参照してください）。

## 6.5 OneRoster情報のリアルタイムのバルク交換

これは、プロバイダーからデータセットを読み取るコンシューマーシステムでREST APIを使用して実現されます。コンシューマーは、バッチ交換に必要な各コレクションに対して一連の読み取り要求を発行する必要があります。REST APIは、コンシューマーが各応答ペイロードで受信するデータの量を確実に制御できるようにするページ付けメカニズムを定義しています。コンシューマーは、すべてのコレクションが受信されるまで一連のリクエストを発行します。次に、次のコレクションを要求できます。コレクションの読み取り順序は、コンシューマーによって決定されます。

## 6.6 OneRoster情報のバルクバッチ交換

これは、CSVエクスポート/インポートを使用して実現されます。ソースシステムは、OR 1.1 CSVエクスポートを使用してCSVファイルのセットを作成し、新しいzipファイルを作成します。このファイルは受信システムに転送されます（転送方法は実装に依存します（例：セキュアFTPの使用など））。受信側のシステムは、OR 1.1 CSVインポート機能を使用してCSVファイルセットをインポートする必要があります。インポート中にエラーが検出された場合は、手動の回復メカニズムを使用する必要があります。

## 付録A - 主要な用語と定義

バルクベースの交換	これは、現在格納されているすべてのデータが新しく提供されたデータセットで置き換えられるデータの交換です。「status」および「dateLastModified」フィールドのペイロード/csv値は空である必要があります。そのようなレコードを受け取ると、含まれているすべてのレコードが「status = active」および「dateLastModified = "...インポートの日付/時間..."」で設定されます。データストアに存在するが、新しいデータセットに含まれていないレコードは、ステータスが「tobedeleted」に設定されている必要があります。
適合性と認証	適合性とは、IMSが実装が仕様に準拠しているかどうかを判断するプロセスです。IMSは、仕様開発プロセスの一部として公開する各仕様の適合プロセスを定義します。特定の仕様に対するIMS適合性を通過するすべての実装は、準拠として認定されます。つまり、IMSはその製品の認定番号を付与します。多くのベンダーは自社の製品が仕様に準拠していると主張していますが、認定されていること、つまり認定番号があり、製品が <a href="#">IMS認定製品ディレクトリ</a> にリストされていることを確認することで確認する必要があります。
CSVベースの交換	データは、zipファイルとして交換される一連のCSVファイルに含まれています。システム間でデータを移動する方法は定義されていないため、実装ごとに異なる場合があります。FTP、電子メールなどを使用します。システムは、データのインポート、データのエクスポート、またはインポートとエクスポートの両方を提供します。インポート/エクスポートの表現は、システム内のデータの内部ストレージなどの可能性が低いために使用されます。LMS、SISなどはCSV形式を使用します。
CSVインポート	CSVファイルのセット(zipファイル)がシステムにロードされるプロセス。インポートシステムは、ファイルのセットを構文および意味的に検証することが期待されており、エラーを検出できる必要があります。エラーを含むことが判明したファイルのその後の処理は、仕様では定義されていません。
CSVエクスポート	システムが他の外部システムで使用されるOneRoster CSVファイルのセットを作成するプロセス。エクスポートシステムは、ファイルの性質によって決定される意味的に一貫したファイルのセットを作成する必要があります。例、「バルク」または「デルタ」。
CSVバリデーター	これは、CSVファイルのセットがOneRoster 1.0または1.1仕様に準拠しているかどうかを判断するために使用されるオンラインのIMS適合性テストシステムです。このバリデーターは、IMSメンバーのみが利用でき、 <a href="http://onerostervalidator.imsglobal.org:8080/oneroster-server-cts-webapp/instructions">http://onerostervalidator.imsglobal.org:8080/oneroster-server-cts-webapp/instructions</a> からアクセスできます。
CSVリファレンステストセット	これは、CSVインポート機能のサポートを要求するシステムの適合性テストをサポートするためにIMSによって生成されるテストCSVファイルのセットです。OR 1.1の場合、OneRoster Conformance Webページから入手できるこのテストセットは、既知のエラーを含む多数のzipファイルで構成されています。CSVインポートの適合性テストの一環として、システムは、圧縮された各CSVファイルセットを正しく処理し、インポート処理の結果を示すreprt/logを生成できることを示す必要があります。システムは、CSVファイル内のデータの構文および/またはセマンティクスのエラーを検出する必要がありますが、その後のそのデータの処理は実装に依存します。つまり、エラーのあるデータを拒否する必要はありません(これにより、市場における差別化が可能になります。システムはエラー回復を処理します)。
データアグリゲーター	データアグリゲーターは、1つ以上のシステムからデータを収集し、この情報を処理して、他のシステムで利用できるようにするシステムです。OneRosterデータアグリゲーターは、データの収集と転送にOneRosterを使用する必要があります。入力または出力に独自のインターフェースを使用できますが、これにより、データ集約に対する独自のゲートウェイアプローチが作成されます。典型的なデータアグリゲーターは、OR REST APIを使用して他のシステムで利用できるようにされたデータと共に、OR CSVベースのデータインポートを利用します。
デルタベースの交換	これは、現在格納されているすべてのデータが新しく提供されたデータセットによって更新されるデータの交換です。「status」および「dateLastModified」フィールドのペイロード/csv値を設定する必要があります。このようなレコードを受信すると、保存されているすべてのレコードは、データ自体と同様に、新しい「status」と「dateLastModified」の値で更新されます。データストアには存在するが、新しいデータセットには存在しないレコードは変更されません。
評定表の「ブル」エンドポイント	これは、サービスコンシューマーがサービスプロバイダーからデータを読み取る(プルする)ことを可能にするエンドポイントのセットです。これは、エンドポイントの「readAllCategories」、「readCategory」、「readAllLineItems」、「readLineItem」、「readAllResults」、「readResult」、「getLineItemsForClass」、「getResultsForClass」、「getResultsForLineItemForClass」、および「getResultsForStudentForClass」のセットです。

<b>評定表の「プッシュ」エンドポイント</b>	これは、サービスプロバイダーがサービスコンシューマーとの間でデータを管理（プッシュ）できるようにするエンドポイントのセットです。これは、「putCategory」、「deleteCategory」、「putLineItem」、「deleteLineItem」、「putResult」、および「deleteResult」のエンドポイントセットです。「プッシュ」エンドポイントは、単一のレコードのみで動作します。
<b>OneRoster コンシューマーAPI (ORCA)</b>	これは、ORコンシューマーAPIのリファレンス実装です。つまり、ORコンシューマー機能を実現するためのコードベースです。これは、OR 1.1 OpenAPI記述ファイルを使用し、Swaggerツールキットを使用してコードライブラリを作成することで作成できます（Swaggerはさまざまなコードベースをサポートしています）。
<b>OneRoster 評定表</b>	評定表情報を管理するORの交換機能のコレクション。したがって、これは「categories.csv」、「lineItems.csv」、「results.csv」、および/または「categories」、「results」、および「lineItems」のエンドポイントのコレクションです。
<b>OneRoster リソース</b>	リソース情報を担当するORの交換機能のコレクション。したがって、これは「classResources.csv」、「courseResources.csv」、および「resources.csv」ファイルのコレクション、および/または「リソース」のエンドポイントです。
<b>OneRoster 名簿作成</b>	名簿作成を担当するORの交換機能のコレクション。したがって、これは「academicSessions.csv」、「classes.csv」、「courses.csv」、「demographics.csv」、「enrollments.csv」、「orgs.csv」、「users.csv」ファイルのコレクションであり、/または「academicSessions」、「classes」、「courses」、「demographics」、「enrollments」、「gradingPeriods」、「orgs」、schools、「students」、「teachers」、「terms」、および「users」のエンドポイント。
<b>OpenAPI / Swagger 定義</b>	OpenAPI仕様は、Swagger仕様に基づくベンダー中立のAPI記述形式です。この仕様はOpenAPI Initiative (OAI) によって開発およびサポートされています。これは、Linux Foundationの下でのオープンガバナンス構造です。IMSは、JSON形式とYAML形式の両方で利用可能なOpenAPI形式を使用して、コンピューターで読み取り可能な形式のOneRoster 1.1 REST APIを提供します。
<b>REST API ベースの交換</b>	交換は、HTTPコールで伝送されるJSONペイロードのセットとして定義されます（これはRESTful交換モデルに基づいています）。REST APIには、そのデータをリアルタイムで交換する方法を定義する広範なサービスエンドポイントのセットがあります。ほとんどのエンドポイントはサービスプロバイダーからのデータの読み取りを許可しますが、評定表データの場合、サービスプロバイダーがサービスユーザーにデータを書き込んだり、サービスユーザーからデータを削除したりすることが可能です。REST APIには、大きなペイロードのページング、ペイロード内のデータの並べ替え、一連のフィルタリングルールによるデータの選択、完全なペイロードのサブセットのみの戻り値の指定のサポートが含まれています。
<b>サービスコンシューマー</b>	これは、OneRosterクライアントとして機能する実装です。つまり、適切な読み取り要求を発行し、サービスプロバイダーから対応する応答を受信したり、サービスプロバイダーが発行した書き込み/削除要求に応答したりできます。
<b>サービスコンシューマー/クライアント適合性テストシステム</b>	これらは、サービスコンシューマーがOneRoster 1.1仕様に準拠しているかどうかを判断するために使用されるテストシステムです。サービスコンシューマーの適合性テストシステムには、次の2つがあります。(a) サービスコンシューマーがサービスプロバイダーにリクエスト（プル/リード）を正しく発行したことを確認するシステム。(b) サービスコンシューマーが書き込み/削除（プッシュ）リクエストに正しく応答することを確認するためのシステム。後者は、「プッシュ」評定表サービスをサポートするサービスユーザーにのみ必要です。
<b>サービスプロバイダー</b>	サービスプロバイダーは、データをデータコンシューマーが利用できるようにするOneRoster準拠のシステムです。このデータのほとんどは、データコンシューマーによってサービスプロバイダーに発行される読み取り要求を介して提供されます。つまり、コンシューマーはサービスプロバイダーからデータを「プル」します。評定表の場合、サービスプロバイダーはHTTP PUTリクエストを使用してデータコンシューマーにデータを「プッシュ」することもできます。データは常にJSONペイロードとしてコンシューマーに提供されます。
<b>サービスプロバイダーの適合性テストシステム</b>	これらは、サービスプロバイダーがOneRoster 1.1仕様に準拠しているかどうかを判断するために使用されるテストシステムです。2つのサービスプロバイダー適合性テストシステムがあります。(a) サービスプロバイダーがサービスコンシューマーから受信したリクエスト（プル/リード）に正しく応答することを確認するシステム。(b) サービスプロバイダーが正しく形成された書き込み/削除（プッシュ）リクエストを発行することを確認するシステム。後者は、「プッシュ」評定表サービスをサポートするサービスプロバイダーにのみ必要です。



**SourcedId**

各オブジェクトには、通信しているOneRosterエンドシステムがオブジェクトを正確に識別するために使用する一意の識別子が必要です。OneRosterでは、この識別子はオブジェクトの「sourcedId」と呼ばれます。この識別子は、データのローカルストレージの主キーとして使用しないでください。つまり、エンドシステムは「sourcedId」と対応するデータベースキーの間のローカルマッピングを維持する必要があります。代わりに、この識別子は相互運用性交換にのみ使用されます。これらの識別子はグローバルに一意であることが強く推奨されますが（UUIDとして認識される必要はありません）、最小の制約は、通信するOneRosterシステムのセット間で一意であることです。

[コラムの先頭へ](#) | [先頭へ](#)

## このドキュメントについて

タイトル:	IMS OneRoster® v1.1 ベストプラクティスと実装ガイド
編集者:	コリン・スマイス、IMS グローバル（イギリス） フィル・ニコルス、オラクル（米国）
バージョン:	1.1
バージョン日付:	2017年4月17日
ステータス:	IMS最終リリース
概要:	これは、OneRoster 1.1仕様を採用する際のベストプラクティスと実装に関する推奨事項のセットです。これらの推奨事項は、CSVとRESTベースのバインディングの両方をカバーしていません。
リビジョン情報:	別のドキュメントとしての「IMS OneRosterベストプラクティスおよび実装ガイド」の最初のリリース。
目的:	このドキュメントは、OneRoster仕様の採用のために一般に公開されます。
ドキュメントの場所:	IMSパブリックWebサイト（標準のダウンロード）： <a href="http://www.imsglobal.org/activity/onerosterlis/">http://www.imsglobal.org/activity/onerosterlis/</a>

[コラムの先頭へ](#) | [先頭へ](#)

## 貢献者のリスト

次の個人がこのドキュメントの開発に貢献しました：

ウィリアム・ベイカー	ピアソン（米国）
アーサー・バーストウ	マグローヒル（米国）
サリ・コンナード	パフォーマンスマターズ（米国）
ハンク・デビッドソン	ピアソン（米国）
ビジェ・ダナライ	クラスリンク（米国）
デビッド・ガッパ	サファリモンタージュ（米国）
リンダ・フェン	ユニコーン（米国）
トム・イングラム	エスキャンピア郡学区（米国）
オクサナ・ジュロセビッチ	インストラクチャー（米国）
マイク・カアストラ	デザイナー2ラーン（カナダ）
ジョン・キム	ピアソン（米国）
アンドリュー・クリツキー	HMH（米国）
リサ・マットソン	IMSグローバル（米国）

デビッド・メイズ	グウィネット郡学校 (米国)
アンディー・ミラー	ラーニングドットコム (米国)
フィル・ニコルス	オラクル (イギリス)
パドレーグ・オイセアダ	HMH (イギリス)
アペンドラ・ペネガラパティ	ピアソン (米国)
ジョージ・ペロー	オレンジカウンティ公立学校 (米国)
ジェームズ・ペロー	FLVS (米国)
パトリック・ポーター	ヒューストンISD (米国)
ウェンディ・リーディ	サンガードK12 (米国)
カート・ロンボット	ピアソン (米国)
マーク・シェフテル	ピアソン (米国)
コリン・スマイス	IMSグローバル (イギリス)
コンラッド・スティメリング	K12 (米国)
アディトヤ・スブラマニウム	スクーロジー (米国)
マット・ベツラ	スクーロジー (米国)
TJ ベリング	マイクロソフト (米国)
マーク・ウォールズ	グウィネット郡学校 (米国)
スタンリー・ワッツ	クラスリンク (米国)
マイク・ザッカーソン	インストラクチャー (米国)

[カラムの先頭へ](#) | [先頭へ](#)

## 改訂履歴

バージョン番号	リリース日	コメント
最終リリース 1.0	2015年6月3日	このドキュメントの最終リリースバージョンの最初の正式リリース。
最終リリース 1.1	2017年4月17日	<p>このドキュメントの2回目の正式リリース。主な変更点は次のとおりです。</p> <ul style="list-style-type: none"> <li>データモデルへの適度な追加。</li> <li>CSVファイル交換に追加された重要な新機能。</li> <li>REST APIにエンドポイントが追加されました。</li> <li>CSVとRESTの両方の適合性要件が、かなり要求が厳しく、具体的にになりました。</li> </ul>

[カラムの先頭へ](#) | [先頭へ](#)

IMS グローバルラーニングコンソーシアム (「IMS グローバル」) は、科学的、実験的、および学術的なコラボレーションのみを目的として、このドキュメント (「仕様」) に含まれる情報を公開しています。

IMS グローバルは、仕様の正確性または完全性に関していかなる保証も表明も行いません。

この資料は、「現状有姿」および「提供可能な限度」で提供されます。

仕様は常に予告なしに変更および修正される場合があります。

仕様に関する有用性、正確性、完全性を評価することは、お客様に関わるお客様の責任です。

IMS グローバルは、ご意見やご提案をお待ちしています。

<http://www.imsglobal.org> のWebサイトからIMS グローバルに連絡してください。

ドキュメント名: IMSOneRoster@1.1ベストプラクティスおよび実装ガイド  
Candidate Final Specification v1.0 を参照してください

日付: 2017年4月17日

このページには、IMSロゴ、Learning ToolsInteroperability® (LTI®)、Accessible Portable ItemProtocol® (APIP®)、Question and TestInteroperability® (QTI®)、CommonCartridge® (CC®などのIMS Global Learning Consortiumの商標が含まれています)、AccessForAll™、OneRoster®、Caliper Analytics™、SensorAPI™。IMS商標使用ポリシーの詳細については、[商標ポリシーページ](#)を参照してください。

© 2001-2017 IMS Global Learning Consortium Inc. All Rights Reserved. [個人情報保護方針](#)

---

This page contains trademarks of the IMS Global Learning Consortium including the IMS Logos, Learning Tools Interoperability® (LTI®), Accessible Portable Item Protocol® (APIP®), Question and Test Interoperability® (QTI®), Common Cartridge® (CC®), AccessForAll™, OneRoster®, Caliper Analytics®, SensorAPI™ and Competencies and Academic Standards Exchange® (CASE®). For more information on the IMS trademark usage policy see [trademark policy page](#)

 [Follow Us on Twitter](#)

© 2001-2020 IMS Global Learning Consortium Inc. All Rights Reserved. [Privacy Policy](#)