

Project Tin Can から派生した  
次世代 SCORM の要件に関する ADL の見解

*Advanced Distributed Learning (ADL) Co-Laboratories*  
4 January 2012



<http://creativecommons.org/licenses/by-nc-sa/3.0/deed.ja>

このドキュメントは、Andy Johnson によって作成されました。  
国防副次官補（レディネス）Advanced Distributed Learning (ADL) イニシアティブの  
問題解決に関わるサポートオフィスのコンタクターです。  
ご意見、お問い合わせは、ADL テクニカルチームにお送りください。

[helpdesk@adlnet.gov](mailto:helpdesk@adlnet.gov)

原文

「An ADL Perspective on Next Generation SCORM Requirements as Derived from  
Project Tin Can」

[http://www.adlnet.gov/wp-content/uploads/2012/01/NEXTGEN-SCORM-requirements-20120130\\_v1.pdf](http://www.adlnet.gov/wp-content/uploads/2012/01/NEXTGEN-SCORM-requirements-20120130_v1.pdf)

日本語訳者 田口 邦彦

作成日 2012-06-29

## イントロダクション

共有可能コンテンツ・オブジェクト参照モデル (SCORM) は 2001 年以降、主要なオンラインラーニングの規格でした。ADL は、現在の Web 2.0、それ以降の技術を用いて SCORM をアップデートする次世代 SCORM の取り組みを行っています。この取り組みは、コミュニティからのフィードバックに直接応え、今後の SCORM の要件が決定されます。

Rustici ソフトウェアは、ADL を通じて広域アナンスを行い、以前の SCORM のバージョンと現在のバージョン 両方の欠点について、コミュニティから多くのフィードバックを集めました。このフィードバックは、さまざまな形で受け取り、最も重要視したデータは、Tin Can ユーザーボイスサイトから集められたデータです。このサイトは、オープンエンドコミュニティからのフィードバックを収集、分類して、参加者が、最も関連性の高い問題に、投票ポイントを行うものです。Ben Clark と Mike Rustici は、SCORM に関して 10 年の経験と持っており、このプロセスを推し進めました。

このホワイトペーパーは、社会ニーズを理解して、私たち自身の総体的な経験、そして、TinCan のプロセスを基点とした成果に基づいて、次世代 SCORM の要件に関して ADL の見解を示します。バージョン 1.0 から SCORM の各バージョンの貢献者として、SCORM テスター、認証者、プロトタイプ開発者、コース設計者、そして何百ものヘルプデスク・チケットのオーサー、実用的な視点とコミュニティの声から、SCORM の欠点とは何か、適切な考えを持っています。私は、自身の経験と、ADL のプログラムに関わり、対話や電子メール、そしてプレゼンテーションなど、それらを振り返り結び付けています。

次のエントリは、Tin Can User Voice サイトでスコアを持った順序です。また、これらは、いくつか類似していると思う問題は、組み替えたり、少なくとも一緒に分類できるように構成しています。ここでは、SCORM の“良く知られた”欠点のいくつかは、新しい問題として議論せず、また「蒸し返す」ようなことはしていません。言い換えれば、リストは完全ではありませんが、ADL コミュニティの関心が何であるか、いくつかの見解を示します。

## **SCORM は、分散コンテンツを扱えるべき。**

すべてのコンテンツは、オリジナルの SCORM が期待するような事前にパッケージ化や、事前に作成ができるわけではありません。パッケージまたは zip ファイルの概念は、ますます時代遅れとなってきています。コンテンツをアップデートするとき、所有権に関する問題など、さまざまな理由でコンテンツは、分散されたサーバに展開することがあります。(クロスドメインスク립ティングなど) 技術的な問題では、コンテンツパッケージにすべて含まれない、数多くのコンテンツ展開に、混乱を引き起こしています。

ソーシャルメディア (及び、プリロードできない、その他のコンテンツ形態) の学習体験は、増えていく傾向にあり、SCORM でトラッキングされる手段を備えていません。多くの学習体験は .zip ファイルの中に、ひとまとまりになっておらず、学習管理システム (LMS) を通じてアップロード/アクセスを行います。このような経験は、意味を持つ方法でトラッキングされる必要があります。従来の LMS に見られる、他の形式化された学習体験とトラッキングを統合することができます。SCORM は、このような体験のトラッキングと完了レポートが可能となるようにストリーミングデータモデルを採用する必要があります。リンクとブックマーキングのモデルは、学習体験に多くのデータをもたらすことが示唆されています。

さらに、SCORM は、常にブラウザをベースとした仕様でした。リッチインターネットアプリケーション (RIA) は、ブラウザ技術を使用することはできませんが、トラッキング程度はできます。さらに、SCORM は、大きなスケールで外側に展開できるだけでなく、さまざまなテクノロジー・プラットフォームと一体化して動作するようになれば、理想的です。

## **SCORM はシンプル (手軽) である必要があります！**

SCORM は、主に 2 つの欠点を持ち、LMS において、ビルトイン依存を持っています。1 つは、導入コストです。LMS は、平均的な小さな組織では必要以上の負担が掛かります。オープンソースの可能性があります、多くは導入が厄介であり、インストールをするにしても専門の人を必要とします。2 つ目の欠点は、LMS に含まれる製品セットに対する依存です。それは、すべての製品の機能を、よく検討しなければなりません。おそらく LMS は、あなたにとって期待していない挙動を、少なくとも 1 つは行うでしょう。そのため「気に入らなくとも、まだマシなもの」を選んでしまうことです。アプリケーションプログラミングインターフェイス (API) が、よりよく公開され、カスタマイズ可能なプログラムと、サービスが分離できるようになれば、良くなっていくでしょう。そのような実装例を 2 つ挙げるとすれば、Google API と Amazon API です。

SCORM は、対象となるユースケースのほとんどを解決することができますが、コンテンツ

の作成において、修正を必要とせず動作させるには、未だに難しいものがあります。現在、コンテンツ開発者は着手する前に 200 ページの技術仕様を読み解く必要があります。Tin Can ユーザーボイスサイトでは、仕様は「標準的な実装で簡単であること、そして可能な限り複雑な物も作れること」がベストだと言われています。認識すべき重要なことは、コンテンツ開発においてプログラマが SCORM を使わずに機能を実装するようになって、恐らくまだ、1 つぐらいは、SCORM を利用して機能を実装することがあるということです。ツールやテンプレートを利用することで、このプロセスは緩和されますが、SCORM のシーケンス機能は、ツールベンダーが効果的に実装するには、未だに難しいのです。そのため前提条件は、もっと単純なものでなければなりません。imsmanifest.xml は厄介であり、XML は制約が厳し過ぎます。

### **SCORM は、オフラインや長時間にわたるコンテンツに対応すべき。**

API が LMS を見つけられずコンテンツが動作しないことや、そしてオフラインのため使用できないときに発生するエラー文字列は、PC の環境が整わない所で利用するとき、学習者やコンテンツ開発者が直面する、苦勞が耐えないひとつの実例です。ネットワークがときどき途切れる程度、または接続されていないコンテンツであっても、特にオンラインラーニングをサポートするモバイル機器が普及してきているなら、トラッキングできる必要があります。現在のところ、状態検出と解決は、SCORM の範囲を越えています、それは強く望まれている機能です。理論的に LMS は、こうしたサービスを作ることもできますが、前述のように、LMS への負担と構造から次世代 SCORM がサポートすべき方向ではありません。SCORM は、オフラインコンテンツを検出するためのツールを持ち、複数のプラットフォーム/デバイスに応じて、学習者をトラッキングできる必要があります。

ゲームやシミュレーションでは、特に SCO の境がないものがあり、実行時間の長いコンテンツになることがあります。SCORM は、流動的に多くの手段をサポートして、オフラインであってもコンテンツにアクセス、トラッキング、そして経験できる必要があります。

### **SCORM は、トラッキングデータを公開する手段を提供すべき。**

LMS の実装で、ひとつ非常に残念に思うことは、ユーザーデータを参照するインターフェイスが何も提供されていないときです。SCORM は、トラッキングされる為に一定量のデータを必要としています。しかしながら、SCORM には、データを公開するために、ユーザー、もしくはプログラムインターフェースを設ける必要条件が規定されていません。それは LMS が持つべき機能であり、SCORM の範囲を超えていると考えられていました。驚くことに、多くの LMS は、コンテンツプロバイダーや管理者、教師、そして学習者が望んでいるようなレベルまでのデータを公開していません。そのため、トラッキングデータを参照するためにインストラクターが、各学生のアカウントでログインするようなことは望ましくありません。典型的なユースケースは、複数の学生を同時にモニターを行い、進捗状況を確認、そして同時に学生と SCORM のトラ

ッキングデータ、両方のやり取りを行っているインストラクターがいることです。

しかしながら、表示すべき必要事項、UI の必要要件、そのほか SCORM がどこまで応えるべきか把握することは、難しい所です。SCORM は、直接データにアクセスできるように、API を提供することができます。それはまた、LMS が、ユーザーインターフェイスを通じて見せるべきデータの要件となります。SCORM は、直接そのデータにアクセスする API を提供するのには可能だと思いますが、基本的にコンテンツに求められる要件は、共通のフォーマットであり、単一の SCO や学習を超えて、場合によってはコースをまたがり (全体のカリキュラムを処理)、レポートを生成して公開できることです。

### **SCORM は、相互運用性と移植性がある「共有可能な」コンテンツであるか確認すべき。**

SCORM は、相互運用性が有効であると考えられていますが。問題は、いくつかの段階で相互運用性が保たれていないことです。特定の LMS の環境で、動作するように設計された 1 つのコースが、別の環境で動作しないことが、未だに発生する可能性があります。良くあることは、グラフィカルな問題であり、フルスクリーンモード、(フレーム、ウィンドウなど) コンテンツ起動に関して、LMS のナビゲーションをどのようにしたら良いかです (SCORM は、コントロールの表示/非表示が可能です)。このような UI の問題は、常に開発元に一任されており、そして現在の SCORM のバージョンでは、その範囲を超えています。コンテンツパッケージから目次を作成することは、システム間でインタフェースの構造が統一されていない一つの例です。

SCORM は、ユーザーインターフェイスを検証するための API を提供するように示されています。実装では、相互運用性の解決には至らないと思われる適合性試験スイートを用いて、コンテンツが実際に LMS で動作するか、確認を行うテストセットになります。適合性試験スイートを用いた「動作確認」は、出力のみのシリーズがありません。問題は 100 を超える、それぞれのコンテンツパッケージをロードして、シリアルテストを実施することは、とても手間の掛かることなのです。

多くの利用可能なコントロールオプションに関して、貧弱なコンテンツ設計 (ナビゲーション規則を曲げるデザイナー) と貧弱な LMS インタフェース設計との間では、共有されるべき責任があります。SCORM が推奨事項を示していないところでは、混乱と矛盾を引き起こしています。例えば「コンテンツパッケージは、何であるか?」「コースは、何か?」「ロールアップはどのように使用すべきか?」「完了と合格は、どのように使用すべきか?」そしてさらにツール認証に対しては、複数の適合性、認証レベル、定義不足を抱えています。

### **SCORM は、現在のプログラムの標準を採用すべき。**

プログラムは、1990 年後半の SCORM の始まりから、変更されていきましたが、SCORM の

内部は、Web 技術の進歩に反応したアップデートは行われてきませんでした。SCORM は「時代の流れを取り込む」必要があります。(注意：このセクションの残りは、これらの技術進歩による略語を多く含みます)SCORM は、HTML を介して Web ブラウザの内部で実行される XML と JavaScript に基づいています。JSON、REST、SOAP は、現在 SCORM で使用している技術と比べると、メリットを持つ新しい Web 技術です。SCORM は、アーキテクチャを分散化する RESTful な API を使用することを推奨します。JSON は、jQuery を使用すると、データモデル全体を「get」および「set」で呼び出すことができるようになります。これらのソリューションは、クロスドメインの問題を解決し、直接的なシンクロシティの必要性を減らすことができるでしょう。「フレーム」構造を排除することも、HTML のフレーム構文をサポートしていないモバイルデバイス用に開発されたコンテンツを含む、そのほかのタイプのコンテンツにも可能性を広げるでしょう。

### **SCORM は、個別学習のアプローチから脱却すべき。**

SCORM は、常に他のユーザや、個人ユーザーの SCO 同士の間であっても情報が遮断されたシングルユーザートラッキングモデルでした。(SCORM2004 4th エディションで「データ」フィールドが使用可能になったことは例外とします。)そして、複数のユーザー間で同じ SCO からユーザーデータを抽出することが、重要な事項であることが分かってきました。チームベースのトレーニング、コラボレーション、結果の共有、さらには(ピアツーピアで)コンテンツの外側で質問できる機能を持つ同一コンテンツを受けている人同士が持つ知識といった、コンテンツの発想は、SCORM に含まれることはありませんでした。チームトレーニング、共同作業、ソーシャルメディア、そして、そのほか全てのコラボレーション形式の動向として、これは、明らかに広げる必要がある分野です。また、必要に応じて、インストラクターによりカスタマイズが必要になります。アクセスコントロールや、結果の可視化、そして、そのほかの判断基準は、同じコンテンツの試行であっても変わるべきです。

### **SCORM は、シーケンシングを完全に取り除く必要がある。**

シーケンシングは、その始まりから、規格のなかで理解しづらい部分でした。SCORM バージョン 1.2 は、区分化され、理解しやすく、とても簡潔なものでした。「シンプルシーケンシング」を追加したことが、1.2 以降の SCORM の各バージョンをひどく複雑にしました。SCORM2004 に使用されるシーケンシングは、想定されるほとんどのユースケースに対処することができますが、ちょっとした単純な動作であっても、とても煩雑になることがあります。シェアリングが行われない主な理由は、シーケンスが分りにくく、SCO のモジュール性のコンセプトが曖昧なためです。また、多くの開発者が、コンテンツの外でシーケンシングを実行することが非常に難しいと判断したため、コンテンツの中で実行することを選択した事情があります。理想的なものは、(1つの大きな SCO のように)コンテンツをよりシームレスに移動できて、なおかつモジュール構成(たくさんの小さな SCO)でトラッキングができるしっかりしたもの

です。

### **SORM は、認証メカニズム、特に評価データを保護する必要がある。**

配信されたコンテンツで、特に完了/習得に関連付いているコンテンツオブジェクトがあるとき、もっとも懸念されることは、配信されたコンテンツとインストラクターが同じページ上にあることを保証できる情報です。(同じものを見ていると保障されてることです) コンテンツには、SCORM のデータ値を設定して合格/不合格を決定する機能があるか？インストラクターが間に入る必要があるのか？LMS の認証が必要か？SCORM の信頼すべき API は、コンテンツのソースを検証する必要があるかもしれません。

SCORM は、常に LMS の認証メカニズムに依存してきました。この「オールインワン」的な依存から離れ、サービスベースのアプローチに移行するには、認証の中核的な仕組みが必要です。ポテンシャルがあるサービス、また統合ポイントとして、データを保護する優れたオープンプロトコルである、OAuth を使用することが提案されています。

SCORM プログラミングの基本的な性質と認証機能を持たない、もう一つの問題は、常に「不正に改変」されやすい状態であったことです。学習者はクライアントサイドのコードを取得し、多くの場合、そこから評価への正しい答えを見つけ出すことができます。次世代 SCORM は、このような不正行為ができないようにする手段を持たなければなりません。

### **SCORM は、より信頼性の高いデータをトラッキングする必要がある。**

SCORM は、SCO の種類や、さらに「評価」または「非評価」として集計の区別がないため、コンテンツ開発者が理解するには、評判の悪いグレーゾーンになっていました。これは、IMS QTI や評価のため同じような統合を含まないため、多くの LMS が間違いを犯します。SCORM は、すべてのコンテンツは、各クラス、個々の学生共通として、「その場で」採点されることに、危険なほど前提としています。SCORM は、特に既存の仕様と調和がなされていない間、統合に関して、これまで ADL のプロジェクトとして取り上げられることはなく、どのような仕様も SCORM と統合しようとしたことはありませんでした。インタラクションの最小限のモデルは、コミュニティが必要とする評価をサポートできるほど十分なものではありませんが、次世代 SCORM では変わるべきです。API を通じて SCORM は、パラグラフの回答やシミュレーション、ゲームデータ、そしてインストラクターが介入する必要があるコンテンツに対して、もっと良いサポートをする必要があります。また、SCORM は、現在は存在しない、新しいタイプのインタラクションを簡単にサポートできるようにする必要があります。

SCORM は、トラッキングのデータ量に制限をかけていますが、1990 年代後半とは違い、ストレージは安くなっています。そのため、どんなに多くのデータ量であってもトラッキングして



保存することができます。そして、プログラミングはスタティックではなく、拡張可能であるべきです。「最低限保証すべき値」は、相互運用性と連動して、実質的にデータ量の上限ができてしまいました。また SCORM は、データの保存時間の長さが、部分的にあいまいなため、明確にする必要があります。どのようなデータが短期で、そして長期か？さらに、もっとも最新のものだけではなく、過去すべての SCO の試行が、記録されるべきだと言われています。「サスペンド」を効果的に使うことは、過去のデータがトラッキングされることで、コンテンツにより用いられる手段ではないのです。

## 終わりに

ADL は、次世代 SCORM を開発する取り組みでリードしています。多くの時間をかけ必要な要件をまとめ上げる取り組みは、達成されましたが、新しい要件について、決して間口を閉ざすことはありません。より効果的で有効な学習/開発環境を作り出すために、コミュニティの参加は非常に重要です。ADLnet.gov で、次世代 SCORM の進化を追い続け、あなたの専門性を発揮してください。SCORM のこれまでの成功は、ADL コミュニティへの参加と、そしてコミュニティが規格を作るプロセスへの理解があったからこそです。規格はどこからか命令されるものではなく、自主的でありながら信頼できるものである必要があります。eラーニングを使う人たちにとってより良いものとなるように、一緒に次世代 SCORM を作っていきましょう。

## 用語集

### API (アプリケーション・プログラミング・インターフェイス)

ソフトウェア・コンポーネントが互いにやりとりをするためのインターフェイスとして使われる、定められたコード体系。

### クロスドメイン・スクリプティング

クロスサイト・スクリプティングとも言います。これは自身の許可されたコンテンツがある、同じドメインやホストされた Web エリア以外に、コンテンツにリンクするプラクティスです。セキュリティ上、重大な脆弱性があるため、JavaScript をはじめとする多くの言語では、コードの同一生成元ポリシーを採用しています。(SCORM は根本的に JavaScript と切り離せません) 安全性は高まりますが、同一ドメイン上にないコンテンツは利用することができません。

## Google と Amazon API

この2つの API を例にとり業界でどのように利用されているか、サードパーティ製の同じコンテンツを利用するコード例を示すと、人気サイトと同じような外観、使い勝手、機能性をもつことができます。例えば、Amazon API は、ユーザーが検索してアマゾンの商品を見つけやすくして、開発者が広告を出すことができる機能を備えています。

## IMS QTI

IMS 質疑運用試験(QTI) 仕様書はシステム全体で共有されるべき質疑・評価ならびに結果についての要点をまとめている。

## リッチインターネットアプリケーション

RIA は、デスクトップアプリケーションに似た形式の Web アプリケーションです。一般的に、サイト固有のブラウザ、または、ブラウザのプラグインを介して配信されます。現在、Adobe Flash、JavaFX、そして Microsoft Silverlight の 3 つが最も一般的なプラットフォームです。

## 最低限保証すべき値

最低限保証すべき値は、LMS が特定のデータモデル要素に対して作成する必要があるデータフィールド数に関する SCORM 規格のルールです。例えば、インタラクションの数が 256 と定義されている場合は、LMS は、少なくともコンテンツに対して 256 のインタラクションのスペースを作成する必要があります。また、それ以上作成することもできますが、仕様に合わせる必要はありません。