

令和 5 年度

教育関連データのデータ連携の実現に向けた実証調査研究

＜別添資料 1xAPI プロファイル仕様に係る検討結果＞

令和 6 年 3 月 29 日

エヌ・ティ・ティ・コミュニケーションズ株式会社

< 目次 >

1. 本ドキュメントについて	4
1.1 本ドキュメントの構成について.....	4
1.2 本仕様のxAPIプロファイル仕様としての宣言と仕様運用方針	4
1.3 本仕様の記載ルールについて	4
2. 共通ユースケース.....	6
2.1 個人識別子による学習者等の指定方法.....	6
2.2 LRPの指定方法	6
2.3 学習内容の指定	6
2.3.1 学習要素サブセットIDの指定方法	6
2.3.2 学習コンテンツに対する観点の指定	7
2.3.3 学習コンテンツに対する難易度の指定	7
2.3.4 学習の自主学性の指定方法.....	8
3. クイズまたはCBT解答ユースケース.....	9
3.1 学習コンテンツ間の関連表現方法	9
4. アンケートユースケース.....	10
5. statement例.....	11
5.1 navima	11

5.1.1 navimaを開こう！	11
5.1.2 ドリルをやろう！	11
5.1.3 課題をやろう！	11
5.2 WEBQU.....	12
6. LRS構築に関する要件	13
6.1 データ量に関する要件	13
6.2 性能に関する要件	14
6.3 コスト効率の良いアーキテクチャに関する要件	14
6.4 パブリッククラウドおよびオンプレミスによる構成の検討	16
6.4.1 データ受信用および提供用API	16
6.4.2 書き込み用高可用DB	16
6.4.3 参照用高機能DB	16
6.4.4 長期保存用低価格ストレージ.....	17
6.4.5 高可用cache DB.....	17
6.4.6 データ処理分析・匿名化等ドライバ.....	17
6.4.7 可視化機能等.....	17

1. 本ドキュメントについて

本ドキュメントに記載される仕様（以下、本仕様）は、「教育関連データのデータ連携の実現に向けた実証調査研究」（以下、本事業）と他事業の xAPI に関する検討を踏まえて xAPI 仕様における statement の構成および拡張方法を統制するための仕様である。xAPI 仕様で規定された概念の中における本仕様の位置づけは、xAPI プロファイルの一つと整理される。

本仕様が参照する主な国内検討については、文部科学省の教育データ標準¹（以下、教育データ標準）、「スマートスクール・プラットフォーム技術仕様²（以下、スマスク仕様）」および「学習 e ポータル標準モデル Ver.3.00³（以下、MEXCBT 仕様）」の 3 仕様を主に参照する。

1.1 本ドキュメントの構成について

本事業で取り扱う学習コンテンツのユースケースは、「ドリルのユースケース」「授業支援クラウドのユースケース」の 2 種類に分類される。このため、本仕様では、全体の共通内容を加えた上記のユースケースで statement の表現方法の規定を構成する。

1.2 本仕様の xAPI プロファイル仕様としての宣言と仕様運用方針

本仕様は、実証実験用の仕様であることから、2 次利用のための語彙等の登録および公開を実施する際は、別途ドメイン名の改定が必要になる。上記を前提に、当該語彙が既存の xAPI プロファイルで規定されている際はこれをそのまま ID として用いる。また止むを得ずドメインの指定を伴う拡張が必要になった際は、「example.com」を独自ドメインで拡張を行うこととする。

1.3 本仕様の記載ルールについて

本仕様の statement に期待される項目は、xAPI プロファイル仕様の template⁴の形式で記載されるべきものであるが、現時点では簡易的に以下の前提で記載している。

- ・ jsonpath で指定された location に対する値の条件を文章で規定し、必須要素の actor, verb, object は、指定が無い項目については、別途補完されなければならない。
- ・ verb の type は、指定が無い限り Activity であることが期待される。

¹ https://www.mext.go.jp/a_menu/other/data_00001.htm

² https://www.soumu.go.jp/main_sosiki/joho_tsusin/kyouiku_joho-ka/smart.html

³ <https://ictconnect21.jp/document/eportal/#standard>

⁴ <https://adlnet.github.io/xapi-profiles/xapi-profiles-structure.html#statement-templates>

なお、xAPI プロファイル仕様の「8 Statement Template」および「8.1 Statement Template Rules」から本来期待される表記方法を以下に記載する。

- ・ Statement Template は、1 つの Verb に対して objectActivityType (object.definition.type) 等の取りうる値が規定される形で表現される。Actor の在り方には特に規定はない。
- ・ Statement Template は、出現すべき項目の可否および要否を rules 項目内の location と presence の組み合わせで表現する。
 - location は、statement 内の項目を意味し、presence は当該項目の要求水準を意味する。
 - exclude は含んではならない、include は、1 つ以上含まなければならない、recommended は推奨、と規定される。

2. 共通ユースケース

2.1 個人識別子による学習者等の指定方法

個人識別子は MEXCBT 仕様 4.2.2.2.2 actor の規定に準ずる。

2.2 LRP の指定方法

LRP は、MEXCBT 仕様 4.2.2.2.3 context の規定に準ずる。

2.3 学習内容の指定

学習内容情報は、対象となる object に対して任意に設定することができるが、設定の際は、本項目で規定された内容に従わなければならない。

2.3.1 学習要素サブセット ID の指定方法

学年、教科・科目、単元等の項目を xAPI 仕様の extensions による拡張を行わずに表現するために、これらの情報をサブセット ID で表現する。なお、本仕様に用いられるコード体系は、教科・科目に関しては教育データ標準のコード定義に準拠し、その他はスマスク仕様に準拠する。

- context.contextActivities.grouping.[*].id
 - 教材間で共通な学習要素の識別子を指定する。
 - 学習要素サブセット ID は、サブセットコード用のドメイン、教科・科目コード、学年コード、学習要素 ID で構成される IRI である。これらの構成要素のうち、ドメイン、教科コードと学年コード（小中学校の場合）または科目コード（高校の場合）の利用を必須とする。
 - 例) <http://subsetcode.example.com/1/P1#AHLTNC>

記載例

```
"context": {
  "contextActivities": {
    "grouping" : [
      {
        "id": "http://<ドメイン>/<教科・科目コード>/<学年コード>/#<単元コード>"
      }
    ]
  }
}
```

```
}
```

2.3.2 学習コンテンツに対する観点の指定

スマスク仕様 4.8.5.2 criteriaScore（観点別配点と結果）に準ずる（ただし、スマスク仕様は旧学習指導要領準拠となるため、改定は課題とする）。

記載例

```
"result": {  
  "extensions": {  
    "http://<ドメイン>/xapi/resultextension/criteriaScore": {  
      "criterion": "<学習観点コード>",  
      "raw": <得点>,  
      "min": <最低特定の配点>,  
      "max": <最高得点の配点>  
    }  
  }  
}
```

2.3.3 学習コンテンツに対する難易度の指定

スマスク仕様 4.8.4.1 difficulty（難易度）に準ずる。

記載例

```
"object": {  
  "id": "<省略>",  
  "definition": {  
    "extensions" {  
      "http://<ドメイン>/xapi/activityextension/difficulty": <難易度>  
    }  
  }  
}
```

2.3.4 学習の自主学性の指定方法

宿題などの教員の起点による学習と自主学習の識別の表現として、スマスク仕様 4.8.6.2 scene（場面）を以下に拡張する。id の Base URL は example.com に置き換えること。

プロパティ名	設定値
id	http://example.com/xapi/contentextension/scene
description	学校場面、学習場面、学習の自主性
type (inlineSchema)	object

本項目の inlineSchema に規定される子要素の項目については以下に規定する。

プロパティ名	データ型	項目の説明
lifescene	string	学校場面コードを用いる。
learningscene	string	学習場面コードを用いる。
selfstudy	boolean	自主性の有無を指定する。
taskbased	boolean	課題性の有無を指定する。

selfstudy と taskbased のプロパティは、宿題などの明示的な課題学習の場合か自主学習かを指定することし、同時に true を指定してはならない。

3. クイズまたは CBT 解答ユースケース

本ユースケースは、MEXCBT 仕様「4.2.1.2.1 MEXCBT が提供するスタディ・ログ」の規定を基準とし、以下の項目を拡張する。

3.1 学習コンテンツ間の関連表現方法

複数問題は、一つの statement に表現しないこととする。

- ・ `$.context.contextActivities.parent[*].id`
 - 問題単位で指定可能な所属項目。特定のテストや、ワーク、問題をまとめた大問題などが相当し、当該問題が所属する構成単位の `activityId` が設定される。静的な構造を表現するため、一つだけ指定できる。問題群はさらに大きな問題群に紐づくことができるため、`parent` を辿ることで階層構造化することができる。
 - 本項目は、学習 e ポータル標準モデルの必須項目となる。
- ・ `$.context.contextActivities.grouping[*].id`
 - 問題単位で指定可能な所属項目。特定のテストや、ワーク、問題をまとめた大問題などが相当する。`grouping` は、`parent` の上の `parent` といった高階層を同時に表現するために用いることができる。`parent` の内容を `grouping` で表現してもよい。

4. アンケートユースケース

アンケート用途についても、本ユースケースを利用できるものとする。スマスク仕様では「4.8.8.2 アンケート」が規定されるが、objectActivityType の指定は必須としない。

アンケートは、複数の回答に対して正誤判定をする必要がないことから複数回答の一括送信のための拡張を有し、任意に利用することができる。

- \$.verb.id
 - <http://adlnet.gov/expapi/verbs/answered>
- \$.object.definition.type
 - <http://adlnet.gov/expapi/activities/cmi.interaction> を指定
- \$.object.definition.interactionType
 - other を指定
- \$.object.definition.extensions."http://example.com/xapi/activityextension/regexResponsesPattern"
 - result.response の入力値に対する正規表現を入力
例) `"^[1-3],[1-3],[1-3]$"`
- \$.response.result
 - 複数問題の回答を指定
例) `"1,2,3"`

5. statement 例

5.1 navima

本実証におけるユースケースでの、navima の statement 例を別添資料 2 に示す。本例は、navima の児童用マニュアル⁵に対応した statement を作成したものである。なお、作成対象とした statement は本実証の分析で想定された機能に対応したもののみとする。

5.1.1 navima を開こう！

ログインは学習記録としては重要な statement になるが、学習 e ポータルの SSO などとの責任分界等について検討が必要なため本例では取り扱わない。

5.1.2 ドリルをやろう！

②の押下のタイミングでイベントを発行する。

学びたいところのトロフィーを押して、ドリルを開く。

①の押下のタイミングでイベントを発行する。

ドリルの回答を手書きで記入し、書き込み終了を示すために「反映」を押す。

④の押下のタイミングでイベントを発行する。

ドリルの回答を終え、「採点」を押す。

5.1.3 課題をやろう！

前段の statement と同様のため省略。

⁵ <https://www.keinet.com/hisakus/wp-content/uploads/sites/374/2022/07/bf59793e9c4e6b3dd6aaf765f726ae57.pdf>

5.2 WEBQU

WEBQU の statement 例を別添資料 2 に示す。

6. LRS 構築に関する要件

6.1 データ量に関する要件

今回作成した statement サンプルの平均データサイズは、650Byte だが、MEXCBT から出力される statement の平均データサイズは約 1.2KB であるため、検討対象に MEXCBT を含めるかで要件が変動する。

ここでは、自治体が整備する LRS という観点から、MEXCBT の利用を前提とすることとして、1.2kByte/statement で試算を行なう。

statement の発行は、基本的に学習行動として認められる行動に対して発行されるため、多数の statement が発行されるケースは以下が考えられる。

- ・ MEXCBT や中間期末テストなどの試験のユースケース
- ・ ツールのオペレーションによる操作

ツールによるオペレーションは、特定のユースケースで極めて高い statement の発行数の発生が考えられるが、常時発生しないため検討外とする。試験のユースケースについて、時間単位の問題量が多い TOEIC を理論的な上限値として試算すると TOEIC の試験数/試験時間は 200 問/2 時間であることから、1.6statement/分が最大値となる。公教育の現実的な利用観点では、全国学力テストが国語・数学・英語の各科目に対して 15 問が設定されているため、15statement/45 分から 0.3statement/分となる。

これらを踏まえて考えると将来的に体育等の statement が発生しない授業が全授業の 1/3 として発生 statement 数を試算すると以下となる。

$$(1 \text{ 時限 (45 分)} \times 5 \text{ 時限} \times 15 \text{ statements}) \times (2 / 3) = 50 \text{ statements}$$

さらに、年間の登校数を 200 日と仮定すると、年間の児童生徒一人当たりの 10,000 statement となる。また、児童生徒の statement の保存期間を小学校から高校卒業までの 12 年および幼児教育期間や大学連携等の引き継ぎ期間等を仮定して 15 年とすると、児童生徒一人当たりで最大 15 万 statements となる。この statement 数に対して、自治体の児童生徒在籍数を仮に 10 万人とすると、150 億 statements となる。150 億 statements に対する 1statement あたりのデータ量 1.2kByte を考慮すると、約 16TB が導出される。以上により、児童生徒 10 万人あたりで約 16TB のデータを安全に扱う要件が必要となる。この試算には、家庭学習等の授業外の考慮が無いが、授業外時間として追加 60 分の考慮を行っても、全体に対する影響は軽微である。

6.2 性能に関する要件

LRS は、データ入力時に入力システムに対してデータ保存を伴う応答のための高い性能要件が必要となり、分析等のデータ参照時には夜間バッチ処理などがある程度利用できる余地があるため高い性能要件を必要としない。

1 万人が同時に学力試験などを同時時間帯に行なう場合を想定すると、45 分あたり 150 万 statements が送出されることとなり、3,000 statements/分の処理量を必要とする。ただし、MEXCBT については statement 取得タイミングが別途制御できるため、この試算は同等規模のユースケースがあった仮定のものである。

6.3 コスト効率の良いアーキテクチャに関する要件

LRS は、書き込み用途と参照用途でシステムに求められる特性が異なる。書き込み用途は、3,000 statements/分の書き込み性能が求められ、書き込み失敗時の LRP 側の再送信などの影響を抑えるための可用性が重要となる。参照用途は、高度な分析を伴うダッシュボード等へのデータ提供を行えるためのデータベースとしての機能が重要となる。可能性についても重要だが、参照失敗時に再試行が相対的に容易であるため、書き込み用途未満の性能要件が許容される。

上記を踏まえて自治体で採用する LRS の構成は、コマンドクエリ責任分離型（CQRS）の構成であることが望ましい。CQRS は書き込み（コマンド）と参照（クエリ）を異なるデータベースに分離することで、書き込み用の高可用な NoSQL 等のデータベースと参照用の高機能な RDB を利用することができる。書き込み用データベースから参照用データベースにデータ転送の仮定で、匿名化などを含む 1 次分析処理なども行える余地があるため、活用がし易い。また、CQRS の構成に、長期保存用のデータストレージと負荷削減用の cache DB をさらに追加することでよりコスト効率を向上できる。

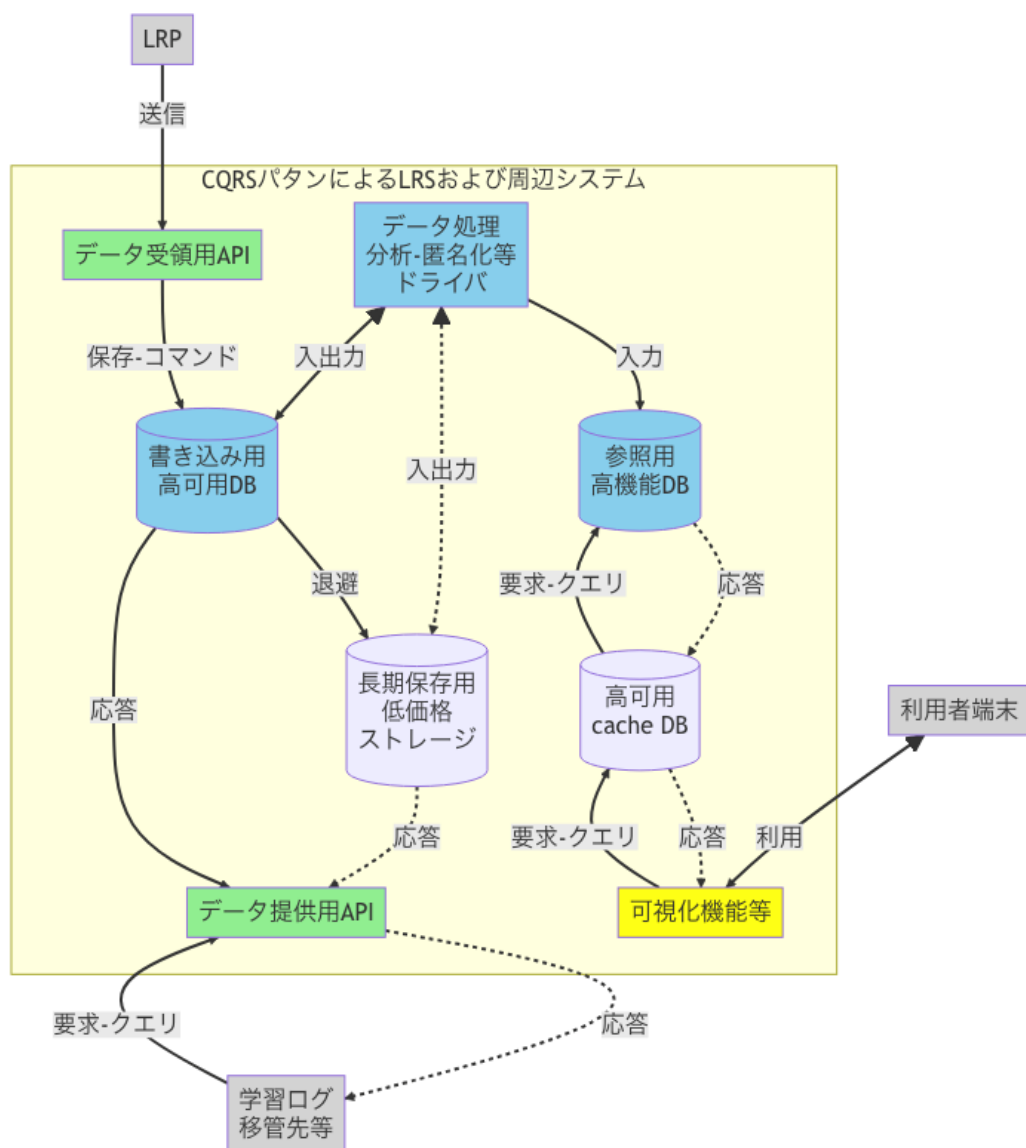


図 a. CQRS パターンによる LRS および周辺システムの構成

6.4 パブリッククラウドおよびオンプレミスによる構成の検討

CQRS パターンによる LRS および周辺システム構成のパブリッククラウドおよびオンプレミス用ミドルウェアによる構築パターンを検討する。本構成は、既存資産の制約によるマルチクラウド構成や UI のみをスクラッチ開発（オンプレミス相当）で構築を行うことを前提に検討を行った。

6.4.1 データ受信用および提供用 API

本機能は、Amazon Web Services（以下、AWS）および Microsoft Azure（以下、Azure）、オンプレミスで以下のようなサービスが該当する。データ受信用の API は、xAPI statement の実装が該当するため、利用する LRS 実装の制約によっては Lambda や Functions 等のマネジメントソリューションを利用できない可能性がある。

- ・ AWS
 - Amazon API Gateway および AWS Lambda
- ・ Azure
 - Azure API Management および Azure Functions
- ・ オンプレミス
 - Kong Gateway 等 および 任意のウェブアプリケーションフレームワーク

6.4.2 書き込み用高可用 DB

本機能は、AWS および Azure で以下のようなサービスが該当する。本機能は、既存 OSS の LRS ソリューションのサポート等を考慮して、基本的には NoSQL 型データベースを用いることが望ましい。

- ・ AWS
 - Amazon DocumentDB
- ・ Azure
 - Azure Cosmos DB
- ・ オンプレミス
 - MongoDB

6.4.3 参照用高機能 DB

本機能は、AWS および Azure で以下のようなサービスが該当する。本機能は狭義には LRS の範囲外であるが、LRS の取得データの活用についてはあらかじめ設計しておく事項であるため検討した。狭義の LRS が MongoDB 等を前提に実装されていることに対して、一般的な BI ツールが前提としている RDBMS を用いることが望ましい。

- ・ AWS
 - Amazon Aurora PostgreSQL / MySQL
- ・ Azure
 - Azure Database for PostgreSQL / MySQL

- ・ オンプレミス
 - PostgreSQL / MySQL

6.4.4 長期保存用低価格ストレージ

本機能は、AWS および Azure で以下のようなサービスが該当する。本機能は、コスト最適化のための措置であるため、相対的な優先度は低いがあらかじめ考慮されていることが望ましい。

- ・ AWS
 - Amazon S3
- ・ Azure
 - Blob Storage
- ・ オンプレミス
 - 任意のストレージソリューション

6.4.5 高可用 cache DB

本機能は、AWS および Azure で以下のようなサービスが該当する。本機能は、コスト最適化のための措置であるため、相対的な優先度は低いがあらかじめ考慮されていることが望ましい。

- ・ AWS
 - Amazon ElastiCache
- ・ Azure
 - Azure Cache for Redis
- ・ オンプレミス
 - Redis

6.4.6 データ処理分析・匿名化等ドライバ

本機能は、AWS および Azure で以下のようなサービスが該当する。本機能の実現方法は多岐にわたるため、必ずしも記載の方法で行なう必要はない。データの加工は、データベース自身が有するプロシージャの利用なども含めて要件確定時に検討されることが望ましい。

- ・ AWS
 - AWS Lambda, AWS Glue および Databricks on AWS
- ・ Azure
 - Azure Functions, Azure Data Factory および Azure Databricks
- ・ オンプレミス
 - 任意の Python 等のランタイム環境

6.4.7 可視化機能等

本機能は、AWS および Azure で以下のようなサービスが該当する。本機能はデータの利活用要件に基づいて決定されるため記載以外の方法として、要件確定時に一般的なウェブアプリケーションとして実装されることも考慮すべき事項である。

- ・ AWS
 - Amazon QuickSight
- ・ Azure
 - Microsoft Power BI
- ・ オンプレミス
 - Tableau, Redash, Metabase 等