General QBank data formats

QBank and the OEA Tool store data in JSON documents, mimicking a MongoDB structure. MongoDB is based on the NoSQL document store model, in which data objects are stored as separate JSON documents inside a collection instead of in the traditional columns and rows of a relational database.

Each item has an _id field that matches the name of the document. For example:

57b952a8ed849b7a42085916.json

Looks like:

```
{
 "displayName": {
  "text": "Grade 9",
  "languageTypeId": "639-2%3AENG%40ISO",
  "formatTypeId": "TextFormats%3APLAIN%40okapia.net",
  "scriptTypeId": "15924%3ALATN%40ISO"
 },
 "license": {
  "text": "",
  "languageTypeId": "639-2%3AENG%40ISO",
  "formatTypeId": "TextFormats%3APLAIN%40okapia.net",
  "scriptTypeId": "15924%3ALATN%40ISO"
 },
 "recordTypeIds": [],
 "brandingIds": [],
 "providerId": "",
 "genusTypeId": "bank-genus-type%3Aclix-grade%40ODL.MIT.EDU",
 "_id": "57b952a8ed849b7a42085916",
 "description": {
  "text": "CLIx content for grade 9",
  "languageTypeId": "639-2%3AENG%40ISO",
  "formatTypeId": "TextFormats%3APLAIN%40okapia.net",
  "scriptTypeId": "15924%3ALATN%40ISO"
 }
}
```

Files are organized by "service" (i.e. what something does) and "object" (i.e. what something is). For example, all assessment items are located in "assessment", then "item":
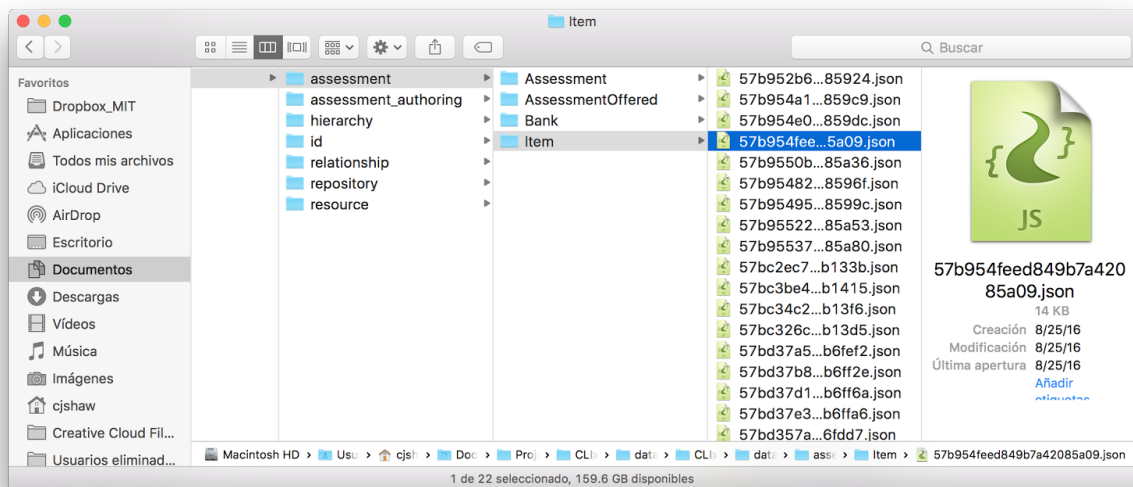
Example data bundle

You can find an example data bundle of JSON at this link:
https://www.dropbox.com/s/pi0kttbniajdk9v/02-09-2016_15-47-PM-staging.tar.gz?dl=0

Data locations

In deployment configurations

To find where an instance of qbank is storing its data on the harddrive, you can open your browser to:

    https://<host>:<port, if needed>/datastore_path
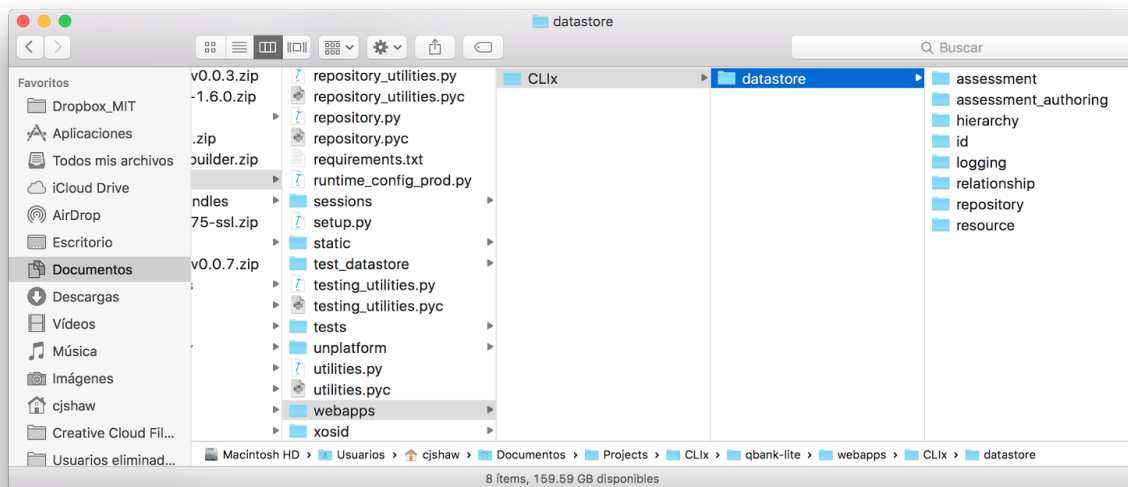
For example, on our staging server it is:

    https://qbank-clix-staging.mit.edu/datastore_path

In a classroom deployment, it might be:

    https://localhost:8080/datastore_path

You can then navigate to that location on the harddrive and find all the qbank data under the webapps / CLIx / datastore folder.

Within the qbank data bundle

For analyzing student results, there are three important folders:
- Assessment
- Logging
- Repository

Assessment Response and Timing data

Each record of an unplatform session is logged in assessment / AssessmentTaken records. Let's look at record 57c98260ed849b643f814a10.json.

```
{
  "assessmentOfferedId": "assessment.AssessmentOffered%3A57c9825fed849b643f814a0f%40ODL.MIT.EDU",
  "displayName": {
    "text": "",
    "languageTypeId": "639-2%3AENG%40ISO",
    "formatTypeId": "TextFormats%3APLAIN%40okapia.net",
    "scriptTypeId": "15924%3ALATN%40ISO"
  },
  "description": {
    "text": "",
    "languageTypeId": "639-2%3AENG%40ISO",
    "formatTypeId": "TextFormats%3APLAIN%40okapia.net",
    "scriptTypeId": "15924%3ALATN%40ISO"
  },
  "recordTypeIds": [
    "assessment-taken-record-type%3Areview-options%40MOODLE.ORG"
  ],
  "gradeId": "",
```

```
"completionTime": null,
"takingAgentId": "osid.agent.Agent%3Aexternal_identifier%40MIT-ODL",
"genusTypeId": "GenusType%3ADEFAULT%40DLKIT.MIT.EDU",
"assignedBankIds": [
  "assessment.Bank%3A57b9547bed849b7a4208596b%40ODL.MIT.EDU"
],
"score": "",
"actualStartTime": {
  "hour": 13,
  "month": 9,
  "second": 6,
  "microsecond": 66666,
  "year": 2016,
  "tzinfo": null,
  "day": 2,
  "minute": 45
},
"_id": "57c98260ed849b643f814a10",
"sections": [
  "assessment.AssessmentSection%3A57c98261ed849b643f814a11%40ODL.MIT.EDU"
],
"takerId": ""
}
```

The field called "takingAgentId" encapsulates the unplatform session ID (highlighted in red above). It is assumed that each session will be unique. Note that external_identifier is the default provided by the OEA player. The Assessment Taken record also logs the time it was started, or first requested by the students, in the actualStartTime field. If the students clicked on "Finish" at the end of the assessment, the completionTime field will also be populated. Note that all times are UTC time.

To find the actual responses and questions seen during this session, we look at the "sections" list, to find a list of Assessment Sections. These can be found in assessment / AssessmentSection, again organized by ID. Let's look for ID 57c98261ed849b643f814a11.

For convenience, we'll look at only a subset of the document. The Assessment Taken ID is also noted at the top of the document, so we can cross-correlate back to the Takens if needed. The interesting part here is the questions array.

```
{
  "assessmentTakenId":
"assessment.AssessmentTaken%3A57c98260ed849b643f814a10%40ODL.MIT.EDU",
  "recordTypeIds": [],
  "assessmentParts": [
    {
      "assessmentPartId":
"assessment_authoring.AssessmentPart%3A57b95538ed849b7a42085aa6%40ODL.MIT.EDU",
      "requiresSequentialItems": false,
      "level": 0
```

```
      }
    ],
   "questions": [
    {
      "itemId": "assessment.Item%3A57c91c8eed849b643f81427f%40ODL.MIT.EDU",
      "questionId":
"assessment.Item%3A57c91c8eed849b643f81427f%25253F%25255B%252522id57e2f76f-d44b-4
d62-8501-e549ef57e080%252522%25252C%252520%252522id3478e191-451f-476b-a264-364eff
0a09db%252522%25252C%252520%252522idea7e9a6b-856b-4428-a68e-552d5f69848e%25252
2%25252C%252520%252522id07c9e641-c46c-4e7b-9e07-d311daa832df%252522%25255D%40
magic-randomize-choices-question-record",
      "responses": [
       {
         "itemId": "assessment.Item%3A57c98262ed849b643f814a12%40assessment-session",
         "displayName": {
          "text": "",
          "languageTypeId": "639-2%3AENG%40ISO",
          "scriptTypeId": "15924%3ALATN%40ISO",
          "formatTypeId": "TextFormats%3APLAIN%40okapia.net"
         },
         "description": {
          "text": "",
          "languageTypeId": "639-2%3AENG%40ISO",
          "scriptTypeId": "15924%3ALATN%40ISO",
          "formatTypeId": "TextFormats%3APLAIN%40okapia.net"
         },
         "recordTypeIds": [
          "answer-record-type%3Aqti%40ODL.MIT.EDU",
          "answer-record-type%3Amulti-choice-with-files-and-feedback%40ODL.MIT.EDU",
          "answer-record-type%3Afiles%40ODL.MIT.EDU",
          "answer-record-type%3Aanswer-with-feedback%40ODL.MIT.EDU"
         ],
         "choiceIds": [
          "id3478e191-451f-476b-a264-364eff0a09db"
         ],
         "fileIds": {},
         "confusedLearningObjectiveIds": [],
         "genusTypeId": "GenusType%3ADEFAULT%40DLKIT.MIT.EDU",
         "assignedBankIds": [
          "assessment.Bank%3A57b9547bed849b7a4208596b%40ODL.MIT.EDU"
         ],
         "submissionTime": {
          "hour": 13,
          "month": 9,
          "second": 9,
          "microsecond": 319643,
          "year": 2016,
          "tzinfo": null,
```

```json
      "day": 2,
      "minute": 52
    },
    "_id": "57c98409ed849b643f814a40",
    "feedback": {
      "text": "",
      "languageTypeId": "639-2%3AENG%40ISO",
      "formatTypeId": "TextFormats%3APLAIN%40okapia.net",
      "scriptTypeId": "15924%3ALATN%40ISO"
    }
  },
  {
    "itemId": "assessment.Item%3A57c98262ed849b643f814a12%40assessment-session",
    "displayName": {
      "text": "",
      "languageTypeId": "639-2%3AENG%40ISO",
      "scriptTypeId": "15924%3ALATN%40ISO",
      "formatTypeId": "TextFormats%3APLAIN%40okapia.net"
    },
    "description": {
      "text": "",
      "languageTypeId": "639-2%3AENG%40ISO",
      "scriptTypeId": "15924%3ALATN%40ISO",
      "formatTypeId": "TextFormats%3APLAIN%40okapia.net"
    },
    "recordTypeIds": [
      "answer-record-type%3Aqti%40ODL.MIT.EDU",
      "answer-record-type%3Amulti-choice-with-files-and-feedback%40ODL.MIT.EDU",
      "answer-record-type%3Afiles%40ODL.MIT.EDU",
      "answer-record-type%3Aanswer-with-feedback%40ODL.MIT.EDU"
    ],
    "choiceIds": [
      "id57e2f76f-d44b-4d62-8501-e549ef57e080"
    ],
    "fileIds": {},
    "confusedLearningObjectiveIds": [],
    "genusTypeId": "GenusType%3ADEFAULT%40DLKIT.MIT.EDU",
    "assignedBankIds": [
      "assessment.Bank%3A57b9547bed849b7a4208596b%40ODL.MIT.EDU"
    ],
    "submissionTime": {
      "hour": 13,
      "month": 9,
      "second": 1,
      "microsecond": 441488,
      "year": 2016,
      "tzinfo": null,
      "day": 2,
```

        "minute": 52
      },
      "_id": "57c98401ed849b643f814a39",
      "feedback": {
       "text": "",
       "languageTypeId": "639-2%3AENG%40ISO",
       "formatTypeId": "TextFormats%3APLAIN%40okapia.net",
       "scriptTypeId": "15924%3ALATN%40ISO"
      }
     }],
    "assessmentPartId":
    "assessment_authoring.AssessmentPart%3A57b95538ed849b7a42085aa6%40ODL.MIT.EDU",
     "_id": "57c98261ed849b643f814a11"
    }

Questions contains a list of questions with three important fields:
- itemId is a unique ID assigned to this exact instance of a question that the student saw -- it can be used to correlate against log data for media playback, changing choices, etc. See Logging and Clickstream data for more information on how you might use this ID.
- questionId is the "version" of the item that the student saw. For many questions that allow shuffling (Multiple Choice, Movable Words, Order Interaction), each student may see the choices presented in a random fashion. However, the questionId encodes the order in which the student saw the question. For example:
  - "assessment.Item%3A57c91c8eed849b643f81427f%25253F%25255B%252522id57e2f76f-d44b-4d62-8501-e549ef57e080%252522%25252C%252520%252522id3478e191-451f-476b-a264-364eff0a09db%252522%25252C%252520%252522idea7e9a6b-856b-4428-a68e-552d5f69848e%252522%25252C%252520%252522id07c9e641-c46c-4e7b-9e07-d311daa832df%252522%25255D%40magic-randomize-choices-question-record"
  - If we want to deconstruct this, we would run it through several steps.
    - First, take the content between %3A and %40
    - 57c91c8eed849b643f81427f%25253F%25255B%252522id57e2f76f-d44b-4d62-8501-e549ef57e080%252522%25252C%252520%252522id3478e191-451f-476b-a264-364eff0a09db%252522%25252C%252520%252522idea7e9a6b-856b-4428-a68e-552d5f69848e%252522%25252C%252520%252522id07c9e641-c46c-4e7b-9e07-d311daa832df%252522%25255D
    - URL decode three times (using a tool like http://meyerweb.com/eric/tools/dencoder/ or a scripting language of your choice) until you get an output like this:
    - 57c91c8eed849b643f81427f?["id57e2f76f-d44b-4d62-8501-e549ef57e080", "id3478e191-451f-476b-a264-364eff0a09db", "idea7e9a6b-856b-4428-a68e-552d5f69848e", "id07c9e641-c46c-4e7b-9e07-d311daa832df"]
    - The first part before the ? is the original itemId ⇒ this is the item as stored on the harddrive (in assessment / Item), and will be the same for all students who saw this question. You can use this to find the origin item

and its choices. You can also use this to determine the actual item type (multiple choice single option, multiple choice multi option, moveable words, etc.).

■ The list afterwards is the order in which the choices were presented to the student. You can match these to the item data, and see that the student was presented with:

id57e2f76f-d44b-4d62-8501-e549ef57e080	Kanasu
id3478e191-451f-476b-a264-364eff0a09db	Zo's grandma
idea7e9a6b-856b-4428-a68e-552d5f69848e	Zo's father
id07c9e641-c46c-4e7b-9e07-d311daa832df	The bus conductor

● Responses are each of the student attempts / responses in reverse-chronological order -- the newest response is listed first.
   ○ Responses include a timestamp of submission
   ○ For Multiple Choice, Order Interaction (story image order), Fill in the Blank, or Moveable Words Sentence, they include a list of choiceIds that the student submitted.
   ○ For Audio Record Tool or Moveable Words Sandbox, they include a fileIds reference (to be discussed later)
   ○ For Short Answer, they will include a "text" field where the response is stored
   ○ For Numeric Response, it will be in an integerValues or decimalValues field, depending on the nature of the question.

File submissions

If files are submitted as part of a student response, the actual files will be located in studentResponseFiles.
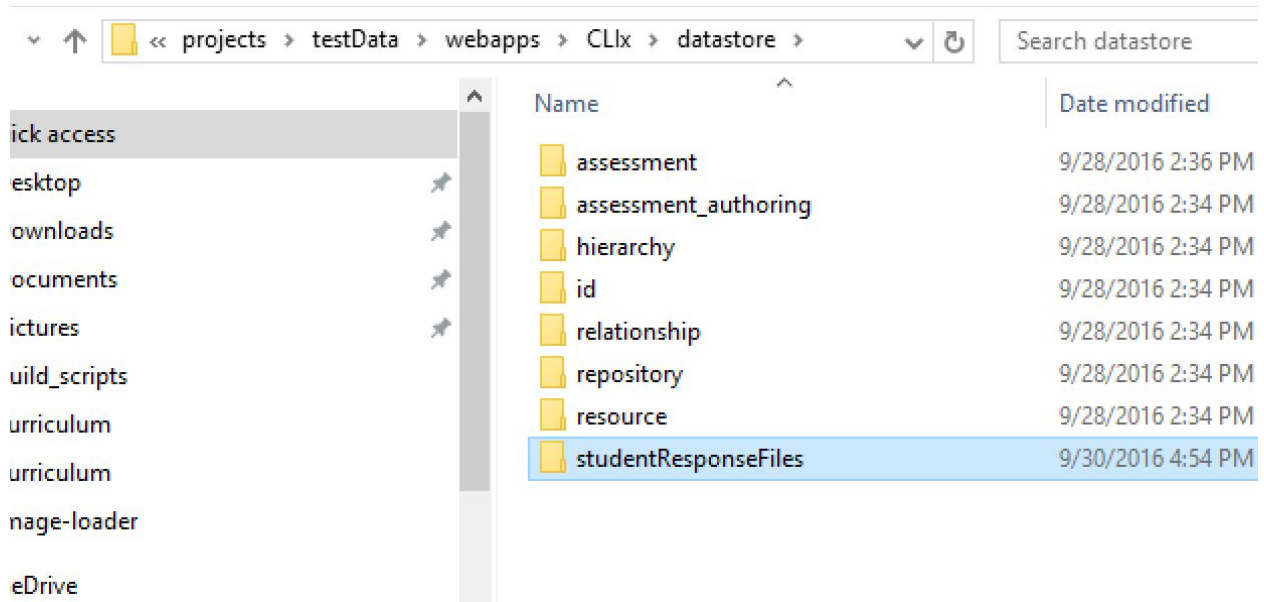
Currently, submitted audio files are stored as *.wav files. Any other extension (like StarLogTNG files) will maintain their original extensions but be assigned a unique ID and filename, as with other objects.

The response fileIds field will look like the following:

```
"fileId": {
        "assetId": "repository.Asset%3A57c5762aed849b43b4a05e50%40ODL.MIT.EDU",
        "assetContentTypeId": "asset-content-genus-type%3Awav%40ODL.MIT.EDU"
    },
```

To find the actual audio file, you first need to find the asset in repository / Asset, 57c5762aed849b43b4a05e50.json.

Within that, there is an assetContents structure:

```
"assetContents": [{
    "displayName": {
       "text": "",
       "languageTypeId": "639-2%3AENG%40ISO",
       "scriptTypeId": "15924%3ALATN%40ISO",
       "formatTypeId": "TextFormats%3APLAIN%40okapia.net"
    },
    "description": {
       "text": "",
       "languageTypeId": "639-2%3AENG%40ISO",
       "scriptTypeId": "15924%3ALATN%40ISO",
       "formatTypeId": "TextFormats%3APLAIN%40okapia.net"
    },
    "recordTypeIds": [],
```

        "assignedRepositoryIds":
    ["repository.Repository%3A57c575eded849b43b4a05e20%40ODL.MIT.EDU"],
        "url":
    "webapps/CLIx/datastore/repository/AssetContent/57c5762aed849b43b4a05e52.wav",
        "assetId": "repository.Asset%3A57c5762aed849b43b4a05e50%40ODL.MIT.EDU",
        "accessibilityTypeId": "NoneType%3ANONE%40dlkit.mit.edu",
        "genusTypeId": "asset-content-genus-type%3Awav%40ODL.MIT.EDU",
        "_id": "57c5762aed849b43b4a05e52",
        "data": ""
    }],

The _id field here tells you where to find the actual AssetContent, in studentResponseFiles: 57c5762aed849b43b4a05e52.wav.

Logging and Clickstream data

The OEA assessment player reports back a variety of other metrics to QBank, including when students played a video / audio file, changed their choice, changed the Moveable Words, etc.

The log entries are found in logging / LogEntry, in the "text" field:

    {
      "_id": "57c436c2ed849b41296f3746",
      "displayName": {
        "text": "",
        "languageTypeId": "639-2%3AENG%40ISO",
        "scriptTypeId": "15924%3ALATN%40ISO",
        "formatTypeId": "TextFormats%3APLAIN%40okapia.net"
      },
      "description": {
        "text": "",
        "languageTypeId": "639-2%3AENG%40ISO",
        "scriptTypeId": "15924%3ALATN%40ISO",
        "formatTypeId": "TextFormats%3APLAIN%40okapia.net"
      },
      "recordTypeIds": [
        "logging.LogEntry%3Atext-blob%40ODL.MIT.EDU"
      ],
      "text": {
        "text": "{\"action\": \"pause audio\", \"questionId\":
    \"assessment.Item%3A57c43645ed849b41296f3737%40assessment-session\",
    \"assessmentOfferedId\":
    \"assessment.AssessmentOffered:57c43642ed849b41296f3734@ODL.MIT.EDU\", \"mediaId\":
    \"\", \"mediaTime\": 9.012245}",
        "languageTypeId": "639-2%3AENG%40ISO",
        "formatTypeId": "TextFormats%3APLAIN%40okapia.net",

```
                    "scriptTypeId": "15924%3ALATN%40ISO"
                },
                "priorityId": "NoneType%3ANONE%40dlkit.mit.edu",
                "genusTypeId": "GenusType%3ADEFAULT%40DLKIT.MIT.EDU",
                "timestamp": {
                  "hour": 13,
                  "month": 8,
                  "second": 6,
                  "microsecond": 229367,
                  "year": 2016,
                  "tzinfo": null,
                  "day": 29,
                  "minute": 21
                },
                "assignedLogIds": [
                  "logging.Log%3A57c40f57ed849b347a04d449%40ODL.MIT.EDU"
                ],
                "agentId": "osid.agent.Agent%3Aexternal_identifier%40MIT-ODL"
            }
```

Log Entries include three relevant fields:
- agentId: this should match the session ID from unplatform and the takingAgentId in the AssessmentTaken.
- timestamp: when this event was recorded, in UTC time
- text.text: the actual data recorded. To analyze this programmatically, you need to JSON parse it, using scripting language like Python or Javascript. This follows the schema below:

Data schema: {
  data: {
      assessmentOfferedId: "<foo>",
      questionId: "<foo>",
      action: "<verb> <media type>",
      mediaTime: "00.00.10", (depends on action)
      mediaId: "<if available, DOM ID, or src as backup>", (depends on action)
      targetWord: "", (depends on action)
      currentSentence: "" (depends on action -- whole sentence pre-move)
  }
}

Example verbs:
play, record, seek, pause, stop, connect, disconnect, click

Example media types:
video, audio, next button, previous button, word

Current sentence: Raju's bags were on the bus
targetWord: were
Action: "disconnect word"

Current sentence: Raju's bags
targetWord: were
Action: "connect word"

Example data blobs:
```
{
  data: {
        assessmentOfferedId:
"assessment.AssessmentOffered%3A1234567890abcdef12345678%40ODL.MIT.EDU",
        questionId: "assessment.Item%3A1234567890abcdef12345678%40ODL.MIT.EDU",
        action: "play video",
        mediaTime: "00.00.10",
        mediaId: "ee_u1l01a01v01"
  }
}
```

Graphical Data Model

AssessmentTaken

| Unplatform Session ID |
| Timestamp |
| AssessmentSection ID |

Items

| Question Text |
| Choices (if applicable) |

| Assessment folder |
| Logging folder |
| Repository folder |

AssessmentSection

| Questions |
| ItemId |
| Responses |
| Timestamp |
| Student Response |
| FileIds (if applicable) |
| AssetId |
| questionId (choice order, if applicable) |

LogEntry

| Unplatform Session ID |
| Timestamp |
| Action |
| Target object |
| Other data object |

Asset

| AssetContents |
| AssetContentId |

| File on disk (assetContentId.*) |