

Office of Liquor and Gaming Regulation

# Data Requirements for QCOM3 Machines

Version 1.0

© The State of Queensland (Department of Justice and Attorney-General) 2016. Copyright protects this publication. The State of Queensland has no objection to this material being reproduced but asserts its right to be recognised as author of its original material and the right to have its material remain unaltered. Enquiries should be addressed to [crown.copyright@qld.gov.au](mailto:crown.copyright@qld.gov.au)

*The information contained herein is subject to change without notice. The copyright owner shall not be liable for technical or other errors or omissions contained herein. The reader/user accepts all risks and responsibility for losses, damages, costs and other consequences resulting directly or indirectly from using this information.*

*The QCOM specification is the intellectual property of The State of Queensland. In order to implement the QCOM specification or subsequent versions, the necessary licensing arrangements will be required to be entered into.*

***For further information, please contact the Office of Liquor and Gaming Regulation on 13 QGOV (13 74 68) or visit <https://www.business.qld.gov.au/industry/liquor-gaming>***

# 1 Contents

<b>1</b>	<b>Contents</b>	<b>3</b>
<b>2</b>	<b>Introduction</b>	<b>4</b>
<b>3</b>	<b>Definitions</b>	<b>4</b>
<b>4</b>	<b>Design Requirements</b>	<b>4</b>
<b>5</b>	<b>Data Schemas</b>	<b>5</b>
5.1	QCOM3 Gaming Machine Program Hash List	5
5.1.1	Example of QCOM3 Gaming Machine Program Hash List	5
5.1.2	JSON Schema for QCOM3 Gaming Machine Program Hash List	7
5.2	QCOM3 Platform List	9
5.2.1	Example of QCOM3 Platform List	9
5.2.2	JSON schema for QCOM3 Platform List	10
5.3	QCOM3 Master Component List	11
5.3.1	Example of QCOM3 Master Component List file	11
5.3.2	JSON schema for QCOM3 Platform Component List	13
<b>6</b>	<b>Revision History</b>	<b>15</b>

## 2 Introduction

This document acts as an addendum to the existing OLGR requirements document titled “Data Requirements for Monitored EGMs”. This document defines the data that is to be transferred between the Licensed Provider’s Systems and this Office and between this Office and Licensed Providers in relation to QCOM 3 gaming machines.

## 3 Definitions

CSV – Comma Separated Value is a non-standard format utilising commas as delimiters for fields of tabular data stored in plain text. The CSV files in this document refers to the files stipulated in the Data Requirements for Monitored EGMs v2.51.

JSON – Javascript Object Notation is an open standard format that uses human-readable text to transmit data objects consisting of attribute-value pairs.

QCOM 3 Content Auditing – Refer QCOM3 Interface Specification document - section 26: “Content Auditing”, QCOM 3’s version of what was termed Program Hash Request/Hash responses in QCOM v1.

## 4 Design Requirements

The data requirements provided to operators must be reflective of the QCOM3 data schemas used in QCOM3 Content Auditing.

The new data schemas must eliminate limitations of the previous designated list of approved software tied with an “Approval Number”. These previous data files were too strict with respect to a single approval number. For example: A single Approval Number corresponded to a single configuration of a game software. This brought issues where multiple software configurations (different BIOS and/or Base combination) would each require an additional approval number.

The proposed changes in this version of game data files will instead provide a list of “approved software” per manufacturer. Removing the reliant on approving multiple versions (configurations) of the same software – simply approving “game software” for a specific manufacturer.

The structure of the game details in the new data files should follow QCOM3 games with respect to content auditing. QCOM 3 promotes a categorised approach to content auditing:

- Common content,
- Game content and
- Peripheral content.

This document describes both Common and Game content. Peripheral content auditing (e.g. Coin/Note Acceptors firmware) is outside the scope of this document.

The data requirements to be provided for operators must be able to be serialised. The use of CSV while effective as a means of providing data, does not provide suitable flexibility and extensibility.

The use of JSON also allows for smaller message size – it is considered light weight in comparison to XML and it can provide structural information when compared to a CSV file. An added benefit is programmatically, JSON files are easier to parse compared to a CSV or XML file when using python or Lua programming languages.

The JSON data format should allow the describing of the Gaming Machine Program Hash List for QCOM 3 Gaming Machines as an object hierarchy, similar to the output of what a QCOM user would see output from a QCOM 3 Gaming Machine via the QCOM API.

## 5 Data Schemas

### 5.1 QCOM3 Gaming Machine Program Hash List

These details are controlled by the Technical Unit, OLGR, and contain one month (31 days) of program hashes per approved software component. For QCOM3 Gaming Machines, SHA256 will be used as its algorithm.

Filename: qgaaaaaa.JSON  
Location: Imopub / files\_from\_qg  
Frequency: On update (2 files-current month & next month)  
Record Info: Refer to QCOM3 Gaming Machine Program Hash List JSON Schema.

#### 5.1.1 Example of QCOM3 Gaming Machine Program Hash List

The following is an example of a single day (1/5/2016) of a *qgaaaaaa.JSON* data structure for a QCOM3 program hash list for two manufacturers (manufacturer = Manufacturer A & Manufacturer B) only.

```
{
  "_alg": "SHA256",
  "_date": "1/5/2016",
  "_seed": "38736B44424D316D7A3559506A614C4D71515849000000000000000000000000",
  "contents_glnk": [{
    "manufacturer": "Manufacturer A",
    "game-content-hr": [{
      "hash": "f3908e3da84ef31cead327fc670577c7335f6e379109e7d188f93ceee7dd46e",
      "uid": {
        "name": "Lots of Bucks",
        "version": "v03.01.00",
        "year": "2014"
      }
    }
  ]
}
```

```

    ]],
    "game-content-lr": [{
      "hash": "15c11f6746eebb84f6a1796e65684f1eff61207aa796e6317461146607961b7e",
      "uid": {
        "name": "Lots of Bucks",
        "version": "mv01.02.00",
        "year": "2014"
      }
    }
  ], {
    "hash": "075af7066000bf7bf36df5637845d50789594a1f31888b1d03966f6fa24b8c1c",
    "uid": {
      "name": "Lots of Bucks",
      "version": "mv01.03.00",
      "year": "2015"
    }
  }
]]
}, {
  "manufacturer": "Manufacturer B",
  "game-content-hr": [{
    "hash": "24908e3da812331cead327fc670577c7335f6e37910237d188f93ceee7dd46e",
    "uid": {
      "name": "Dashing Dollars Bears",
      "version": "v02.01.03",
      "year": "2015"
    }
  }
]],
  "game-content-lr": [{
    "hash": "075af7066000bf7bf36df5637845d50789594a1f31888b1d03966f6fa24b8c1c",
    "uid": {
      "name": "Dashing Dollars Bears",
      "version": "mv02.00.00",
      "year": "2015"
    }
  }
  ], {
    "hash": "086f7066000bf7bf36df5637845d50789594a1f31888b1d03966f6fa24b8c1c",
    "uid": {
      "name": "Dashing Dollars Bears",
      "version": "mv01.02.00",
      "year": "2015"
    }
  }
]]
}}
}

```

The following schema is available for use as a validation tool for generated JSON files – refer to <http://json-schema.org/implementations.html> on how to utilise this file.

## 5.1.2 JSON Schema for QCOM3 Gaming Machine Program Hash List

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "title": "QCOM3 Gaming Machine Program Hash List",
  "type": "array",
  "items": {
    "type": "object",
    "properties": {
      "_date": {
        "description": "Date representation in the format: DD/MM/YYYY",
        "type": "string"
      },
      "_seed": {
        "description": "Seed used for the Day",
        "type": "string"
      },
      "_alg": {
        "description": "Algorithm used for the Hash Calculation,e.g. SHA256",
        "type": "string"
      },
      "contents_glnk": {
        "type": "array",
        "items": {
          "type": "object",
          "properties": {
            "manufacturer": {
              "description": "Manufacturer Name as reported by the EGM via the
QCOM API function idMfrName()",
              "type": "string"
            },
            "game-content-hr": {
              "type": "array",
              "items": {
                "type": "object",
                "properties": {
                  "uid": {
                    "type": "object",
                    "properties": {
                      "name": {
                        "description": "UID of Approved Game
Content",
                        "type": "string"
                      }
                    }
                  },
                  "version": {
                    "description": "Version Number",
                    "type": "string"
                  },
                  "year": {
                    "description": "Year",
```

```

        "type": "string"
    }
}
},
"hash": {
    "description": "Hash result of UID",
    "type": "string"
}
},
"required": ["uid", "hash"]
}
},
"game-content-lr": {
    "type": "array",
    "items": {
        "type": "object",
        "properties": {
            "uid": {
                "type": "object",
                "properties": {
                    "name": {
                        "description": "UID of Approved Game
Content",
                        "type": "string"
                    },
                    "version": {
                        "description": "Version Number",
                        "type": "string"
                    },
                    "year": {
                        "description": "Year",
                        "type": "string"
                    }
                }
            },
            "hash": {
                "description": "Hash result of UID",
                "type": "string"
            }
        },
        "required": ["uid", "hash"]
    }
}
},
"required": ["manufacturer", "game-content-hr", "game-content-lr"]
}
},
"required": ["_day", "_seed", "_alg", "contents_glnk"]
}

```

```
}
```

## 5.2 QCOM3 Platform List

These details are controlled by the Technical Unit Section of OLGR, and contain details about approved software platform. This file provides information on the required components within a platform per manufacturer.

Filename: qgaaaaaa-platform.JSON  
Location: Imopub / files\_from\_qg  
Frequency: On update  
Record Info: Refer to the QCOM3 Platform List JSON Schema

### 5.2.1 Example of QCOM3 Platform List

```
{  
  "platform": [  
    {  
      "name": "Platform 1",  
      "manufacturer": "Manufacturer A",  
      "component": [  
        "Logic BIOS",  
        "Logic CPLD",  
        "Logic WindowsXP",  
        "Logic Main APP",  
        "Logic audit APP",  
        "IO - BIOS"  
      ]  
    }, {  
      "name": "Platform A",  
      "manufacturer": "Manufacturer B",  
      "component": [  
        "Kernel D",  
        " BIOS 1.0",  
        " BIOS 2.1",  
        " BIOS 2.0"  
      ]  
    }  
  ]  
}
```

The following schema is available for use as a validation tool for generated JSON files – refer to <http://json-schema.org/implementations.html> on how to utilise this file.

## 5.2.2 JSON schema for QCOM3 Platform List

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "title": "QCOM3 Platform List - Schema",
  "description": "A platform for each manufacturer should describe all QCOM3 common-
content components",
  "type": "object",
  "properties": {
    "platform": {
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "name": {
            "description": "Platform Name",
            "type": "string"
          },
          "manufacturer": {
            "description": "Manufacturer Specified Platform Name, e.g.
'Platform 1'",
            "type": "string"
          },
          "component": {
            "type": "array",
            "items": {
              "type": "string",
              "description": "Component names must uniquely identify all
possible components i.e. no collisions"
            }
          }
        }
      },
      "required": [
        "name",
        "manufacturer",
        "component"
      ]
    }
  },
  "required": [
    "platform"
  ]
}
```

## 5.3 QCOM3 Master Component List

These details are controlled by the Technical Unit Section of OLGR, and contain details about approved QCOM3 common-content software. This is essentially a flat file that uniquely identifies all possible components. The file contains one month (31 days) of program hashes per approved software component. SHA256 will be used as the algorithm by default.

Filename: qgaaaaaa-master-component.JSON  
Location: Imopub / files\_from\_qg  
Frequency: On update  
Record Info: Refer to the QCOM3 Master Component List JSON schema

### 5.3.1 Example of QCOM3 Master Component List file for one day.

```
[{
  "_date": "1/5/2016",
  "_seed": "38736B44424D316D7A3559506A614C4D71515849000000000000000000000000",
  "_alg": "SHA256",
  "common-content": [{
    "uid": {
      "name": "Logic BIOS",
      "version": "v7.12",
      "year": "2011"
    },
    "manufacturer": "Manufacturer A",
    "platform": ["Platform 1"],
    "hash": "6c47fb643bf3537be9fef88690109d3dc19b90132364765b8688346c16a4a41e"
  }, {
    "uid": {
      "name": "Logic CPLD",
      "version": "v6.00",
      "year": "2009"
    },
    "manufacturer": "Manufacturer A",
    "platform": ["Platform 1"],
    "hash": "e97f23390299b4fa531a58d23174e15d4ea5c356f712872939f18c383957175d"
  }, {
    "uid": {
      "name": "Logic OS WindowsXP",
      "version": "v12.01.01",
      "year": "2012"
    },
    "manufacturer": "Manufacturer A",
    "platform": ["Platform 1"],
    "hash": "075af7066000bf7bf36df5637845d50789594a1f31888b1d03966f6fa24b8c1c"
  }, {
    "uid": {
      "name": "Logic Main APP",
      "version": "v3.01.03",
      "year": "2011"
    },
    "manufacturer": "Manufacturer A",
    "platform": ["Platform 1"],
    "hash": "albca0109941cc5a2416d6076f1cc25184ded52a096fcc0ac1511b612085078e"
  }, {
    "uid": {
      "name": "Logic audit APP",
      "version": "v4.1.23",
      "year": "2011"
    },
    "manufacturer": "Manufacturer A",
    "platform": ["Platform 1"],
    "hash": "98473b0bc4260e726f7b48c893373715880af41ac3b460e017ecaaf36e0e9a71"
  }
  ]
}
```

```

}, {
  "uid": {
    "name": "IO - BIOS",
    "version": "v3.2.1",
    "year": "2011"
  },
  "manufacturer": "Manufacturer A",
  "platform": ["Platform 1", "Gen6"],
  "hash": "23ca4b584baea29c0b03ddb17455436e8965968e1be417c3db11d40be5522e69"
}, {
  "uid": {
    "name": "Kernel S3QLQ03E",
    "version": "S3QLQ03E",
    "year": "2016"
  },
  "platform": ["PLATFORM 2"],
  "hash": "075af7066000bf7bf36df5637845d50789594a1f31888b1d03966f6fa24b8c1c",
  "manufacturer": "Manufacturer B"
}, {
  "uid": {
    "name": "Kernel S3QLQ03A",
    "version": "S3QLQ03A",
    "year": "2015"
  },
  "platform": ["PLATFORM 2"],
  "hash": "12af7066000bf7bf36df5637845d50789594a1f31888b1d03966f6fa24b8c1c",
  "manufacturer": "Manufacturer B"
}, {
  "uid": {
    "name": "SLV002 BIOS",
    "version": "SBSLV002",
    "year": "2007"
  },
  "platform": ["PLATFORM 2"],
  "hash": "45684baea29c0b03ddb17455436e8965968e1be417c3db11d40be5522e69",
  "manufacturer": "Manufacturer B"
}, {
  "uid": {
    "name": "SLV003 BIOS",
    "version": "SBSLV003",
    "year": "2009"
  },
  "platform": ["PLATFORM 2", "EQUINOX"],
  "hash": "23ca4b584baea29c0b03ddb17455436e8965968e1be417c3db11d40be5522e69",
  "manufacturer": "Manufacturer B"
}, {
  "uid": {
    "name": "BIOS 2.1",
    "version": "SBAL2105",
    "year": "2010"
  },
  "platform": ["PLATFORM 2", "EQUINOX"],
  "hash": "98473b0bc4260e726f7b48c893373715880af41ac3b460e017ecaaf36e0e9a71",
  "manufacturer": "Manufacturer B"
}, {
  "uid": {
    "name": "BIOS 2.0",
    "version": "SBAL2004",
    "year": "2009"
  },
  "platform": ["PLATFORM 2", "EQUINOX"],
  "hash": "9123450bc4260e726f7b48c893373715880af41ac3b460e017ecaaf36e0e9a71",
  "manufacturer": "Manufacturer B"
}]
}]

```

The following schema is available for use as a validation tool for generated JSON files – refer to <http://json-schema.org/implementations.html> on how to utilise this file.

### 5.3.2 JSON schema for QCOM3 Master Component List

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "title": "QCOM3 Master Component List Schema",
  "description": "Component names in a flat file list must uniquely identify all possible
components i.e. no collisions",
  "type": "array",
  "items": {
    "type": "object",
    "properties": {
      "date": {
        "description": "Date representation in the format: DD/MM/YYYY",
        "type": "string"
      },
      "seed": {
        "description": "Seed used for the Day",
        "type": "string"
      },
      "alg": {
        "description": "Algorithm used for the Hash Calculation, e.g. SHA256",
        "type": "string"
      },
      "common-content": {
        "type": "array",
        "items": {
          "type": "object",
          "properties": {
            "uid": {
              "type": "object",
              "properties": {
                "name": {
                  "description": "Component Name needs to quantify brand/plaform and must
be unique",
                  "type": "string"
                },
                "version": {
                  "type": "string"
                },
                "year": {
                  "type": "string"
                }
              }
            },
            "required": [
              "name",
              "version",
              "year"
            ]
          }
        },
        "hash": {
          "description": "Hash result of uid",
          "type": "string"
        },
        "platform": {
          "type": "array",
          "items": {
            "type": "string",
            "description": "Manufacturer Specified Platform Name, e.g. 'Platform 1'"
          }
        },
        "manufacturer": {
          "description": "Manufacturer Name as reported by the EGM via the QCOM API
function idMfrName()",
          "type": "string"
        }
      },
      "required": [
```

```
        "uid",
        "hash",
        "manufacturer",
        "platform"
    ]
}
},
"required": [
    "_seed",
    "_alg",
    "common-content"
]
}
```

## 6 Revision History

Version	Changes	Who	Release Date	Incept Date
1.0 draft	Initial Industry Comment Release	JA	16/12/16	* see below
1.0	Initial Release	JA	24/1/17	* see below

*\* "Where not stated otherwise the incept date for new or changed minimum requirements in this version of the document is 12 months from the release date of the document in*