

pyValkLib Pre-Release Documentation

Pherakki

July 30, 2022

Contents

1	Data Container Format Specifications	2
1.1	MXEC	2
1.1.1	Subcontainers	2
1.1.2	Data Structure	2
2	Metadata Container Format Specifications	4
3	File Interfaces	5

1 Data Container Format Specifications

1.1 MXEC

1.1.1 Subcontainers

The MXEC container contains no Data Containers.
It contains the following Metadata Containers:

- POF0
- ENRS
- CCRS
- EOFC

1.1.2 Data Structure

MXEC contains four “tables” of data:

- Parameters Table
- Entities Table
- Path Graphs Table
- Assets Table

The parameter table contains game data in the form of different “structures”. I will term these “structures” **ParameterSets**. Each type of **ParameterSet** consistently contains the same series of datatypes; they likely directly correspond to C structs. Many **ParameterSets** are linked to **Entities** and **PathGraphs**. **Assets** are linked to **ParameterSets**.

The entities table contains **Entities**. Each **Entity** can contain child **Entities** and attached **ParameterSets**. The child **Entities** are always the same for a given **Entity**. Those child **Entities** can have their own child **Entities**; the definition of an **Entity** is therefore a multi-layered structure. Each **Entity** type links to a number of pre-defined **ParameterSet** types. Each instance of an **Entity** contains the IDs of these **ParameterSets**.

The path graphs table contains **PathGraphs**. A **PathGraph** is a set of **Nodes** and **Edges**. Each **Node** has an associated **ParameterSet**. Each **Edge** can have multiple **ParameterSets** attached; in practice the VC1 **Edges** only link to a single **void**-type **ParameterSet**. The **Edges** connect together different pairs of **Nodes** in order to create a **Graph**. These **Graphs** are general mathematical structures; they are allowed to be cyclic (indeed, many are circular loops), paths with a definite start and end, or anything in-between. This is

likely how *e.g.* searchlight movement is implemented.

The assets table contains **Assets**. A **ParameterSet** may contain a reference to an **Asset**; if it does, that reference is stored in the **Assets** table. Each **Asset** contains a path to a certain file, and two unknown IDs. The type of the file is also stored; in the future this will be auto-calculated from the file extension once MergeTexture-type HTX files are marked on the **ParameterSets**. For now, below is a lookup table for the different filetypes:

- HMD: 1
- HTX: 2
- HMT: 3
- MCL: 6
- MLX: 8
- ABR: 9
- ABD: 10
- CVD: 12
- HST: 12
- BHV: 12
- PVS: 20
- HTX: 21 (MergeTexture)
- HTR: 22
- MMF: 24
- MMR: 25

2 Metadata Container Format Specifications

3 Public Interface Object API