

Introduction

The FDB format is created to provide free, universal, fast, and portable dictionary bases. The bases may contain not only dictionaries but books, encyclopedias, and all other kinds of this data. The other purpose of the project is to present a completely open format. In other words, all information published under this format cannot be copyrighted, and can be edited and redistributed by anybody for any purpose without references to the source.

The Java library which allows to work with the FDB format is called Dictan-Core, and available at <http://dictan-core.googlecode.com> under the GNU LGPL v3 (and later) license.

The format is built up on the top of the open source SQLite Database library. SQLite is a fully transactional RDBMS that is natively supported by all popular operating systems including mobile ones. Now there are free SQLite drivers for nearly every programming language what makes this database available for a wide range of developers.

FDB Format Structure

words - A table to store the list of words sorted using the embedded collator.

article_blocks - A table of article blocks which have a special format, and compressed with gzip. Each word is connected to the article by ID and the number of article in the block.

media_resource_keys - A table to store the list of words sorted using the embedded collator.

media_resource_bocks - A table of media resource blocks which have a special format, and compressed with gzip. Each media resource key is connected to the article by the ID and the number of resource in the block.

abbreviations – A table which contains abbreviations used in articles. The number of abbreviations is not supposed to be big, and access to them must be fast to provide dynamical connection from articles. Therefore, this table contains abbreviations with definitions with no compression.

language_directions - The language directions of a dictionary base. Each row of the table consists of the languages from which and to which the translation is supported.

base_properties – All string parameters and meta info of the base.

base_resources – All kind of string and binary data that is used in the base. Particularly collation data for the supported language directions is stored in this table.

Collation

The FDB sorting (collation) rules are based on ICU (International Components for Unicode) rules with the Java RuleBasedCollator. SQLite may have its own collators but they must be connected at compile-time and can be based on the 3rd party libraries. To get rid of this ambiguity, the collation rules are stored within the FDB bases to make them compatible with all platforms and SQLite libraries. FDB defines its own default collation rules and uses Java (ICU based) localized ones. With this approach the sorting rules can always be read from the FDB base, parsed, and used for search.

File Size Limitations

One of the limits of particular SQLite library implementations and file systems is the file size. E.g. Android cannot open SQLite bases larger than 2 GB, FAT-32 does not allow files larger than 4 GB, etc... To overcome this limitation, FDB supports multi-part bases making parts no greater than 2 GB.

