

Avro Schema 格式说明

2016年9月16日 8:31

Avro简单格式列表（8种）

原生类型	说明
null	表示没有值
boolean	表示一个二进制布尔值
int	表示32位有符号整数
long	表示64位有符号整数
float	表示32位单精度浮点数
double	表示64位双精度浮点数
bytes	表示8位无符号字节序列
string	表示字符序列

Avro复杂格式列表（6种）

只需掌握：**record**

复杂类型	属性	说明
Records	type name	record
	name	a JSON string providing the name of the record (required).
	namespace	a JSON string that qualifies the name(optional).
	doc	a JSON string providing documentation to the user of this schema (optional).
	aliases	a JSON array of strings, providing alternate names for this record (optional).
	fields	a JSON array, listing fields (required).
	name	a JSON string.
	type	a schema/a string of defined record.
	default	a default value for field when lack.
	order	ordering of this field.
Enums	type name	enum
	name	a JSON string providing the name of the enum (required).
	namespace	a JSON string that qualifies the name.
	doc	a JSON string providing documentation to the user of this schema (optional).
	aliases	a JSON array of strings, providing alternate names for this enum (optional)
	symbols	a JSON array, listing symbols, as JSON strings (required). All symbols in an enum must be unique.
Arrays	type name	array
	items	the schema of the array' s items.
Maps	type name	map
	values	the schema of the map' s values.
Fixed	type name	fixed
	name	a string naming this fixed (required).
	namespace	a string that qualifies the name.
	aliases	a JSON array of strings, providing alternate names for this enum (optional).
	size	an integer, specifying the number of bytes per value (required).

Test.avsc文件 所有格式示例

说明：Test.avsc文件，利用avro的插件可生成对应的Test 类，这个类可以利用avro的API序列化反序列化)

```
{ "namespace": "avro.domain",
  "type": "record",
  "name": "Test",
  "fields": [
    { "name": "stringVar", "type": "string" },
    { "name": "bytesVar", "type": [ "bytes", "null" ] },
    { "name": "booleanVar", "type": "boolean" },
    { "name": "intVar", "type": "int", "order": "descending" },
    { "name": "longVar", "type": [ "long", "null" ], "order": "ascending" },
    { "name": "floatVar", "type": "float" },
    { "name": "doubleVar", "type": "double" },
    { "name": "enumVar", "type": { "type": "enum", "name": "Suit", "symbols": [ "SPADES", "HEARTS", "DIAMONDS", "CLUBS" ] } },
    { "name": "strArrayVar", "type": { "type": "array", "items": "string" } },
    { "name": "intArrayVar", "type": { "type": "array", "items": "int" } },
    { "name": "mapVar", "type": { "type": "map", "values": "long" } },
    { "name": "fixedVar", "type": { "type": "fixed", "size": 16, "name": "md5" } }
  ]
}
```