

Data Table

Table of contents

1 Overview.....	2
2 Data Table Factory.....	2
3 Data Table in action.....	2

1. Overview

JPLDataTable is a kind of naive relational database table. It handles Records (a set of data elements) using a model of schema (field names and types).

It is used as typed properties to constraint and control fields and content.

In the JPL, it has been used for the definition of annotations.

Note:

It is based on a typed field manager that associates a field with a typed value. This mechanism is based on JPLITeleporter converters that convert String to any typed value object.

2. Data Table Factory

3. Data Table in action

```
// a factory to create specific tables
JPLDataTableFactory factory =
    new
JPLDataTableFactory.Builder().addField("name").addField("age",
    JPLFieldTypeManager.INTEGER).build();

// a new simple table
JPLDataTable table = factory.newTable();

Properties props = new Properties();
props.setProperty("name", "james");
props.setProperty("age", "23");

table.addNewRecord(props);

props.setProperty("name", "paul");
props.setProperty("age", "34");

table.addNewRecord(props);

Assert.assertEquals(2, table.size());
Assert.assertEquals(23, table.getRecord(0).getValue("age"));
```

Note:

The accession of any record are done by integer id query. It could be good to give the choice to specify a field as the unique identifier in a next version like in the following prototype:

Data Table

```
        // define the first field as primary key
        factory =
            new
JPLDataTableFactory.Builder().addField("name").addField("age",
            JPLFieldTypeManager.INTEGER).key("name").build();

        ...

        int age = table.getRecord("james").getValue("age");
```