OriginalDatabaseActions

The **Original Database Actions** are the first attempt to integrate DatabaseActions support in Cocoon. Provides several Actions that automated the basics database operations over individual records (insert, update and delete). This Actions are oriented to use request parameters as input. The output is provided using request attributes and sitemap variables.

It is possible to create components that are capable of performing these activities for any row in any table, assuming that they are parameterised by a description of the database schema.

Cocoon achieves this using an additional configuration file, similar to how FormValidationUsingCocoon uses a configuration file to define the form parameters and validation rules.

If you are starting in Cocoon, use better the new ModularDatabaseActions

If you are looking for the supported DataypesOfDatabaseActions

---

Main Disadvantage

1. No support for autoincrement column type
2. The description of the database structure is split over several files, because this Actions attempt to use all tables in the provided description.

Usage

Describing the Database - descriptor.xml

The database is described using an XML document. The root element of the document is ignored, and can have any name. In theory this allows a single document to contain both database description and form validation configuration.

- `connection` – identifies a database connection pool, see ConnectionPooling.
- `table` – describes a complete table. It's `name` attribute identifies the table. The file can contain several table elements.
- `keys` – contains database primary key information
- `key` – describes a key.
- `values` – contains information about non-key columns in table
- `value` – describes a non-key column

The `key` and `value` elements share the following attributes:

- `param` – the name of the posted form parameter
- `dbcol` – the corresponding database column name
- `type` – the DataypesOfDatabaseActions of the column.

An example:

```
<root>
  <connection>personnel<connection>
  <table>
   <keys>
    <key param="id" dbcol="id" type="int"/>
   </keys>
   <values>
    <value param="name" dbcol="name" type="string"/>
    <value param="department" dbcol="department_id" type="int"/>
   </values>
  </table>
</root>
```

The Database Actions

- DatabaseAddAction – inserts a record; there's a specialisation for Oracle, the `OraAddAction`
- DatabaseUpdateAction – updates a record
- DatabaseDeleteAction – deletes a record

The actions can be combined into an ActionSet, or can be referenced directly from a pipeline. If the action succeeds then it's contents are evaluated.