

DataMgr 2.1 Alpha

Posted At : May 1, 2007 1:26 PM | Posted By : Steve

Related Categories: DataMgr

I realize that DataMgr 2.0 hasn't been out long, but I thought of a few features that I wanted near the end of the 2.0 development cycle. Since they required a fair bit of extra effort, I thought it best to hold them over for a "2.1" release.

Although I have been given some good ideas for future versions of DataMgr, the 2.1 version will probably be the last major release for at least a few months - I have some other projects that I want to give some attention.

DataMgr 2.1 is still in alpha, so the feature set may change (unstable features may be removed), but the features in the alpha are:

- Optional "sort-field" attribute for "list" relation field to dictate order of list
- Optional "delimiter" attribute for "list" relation field
- DataMgr throws error for more than one "Increment" field in one table
- Report datasource name in loadXML() error
- Cascading Relation Fields (see below)
- **Optional bidirectional self-relational many-to-many-relationships**
- Custom Relation Fields (see below)
- Optional "advsql" argument in getRecords (see below)
- Fixed bug in saveSortOrder for MySQL

Most of these features should be relatively obvious, but some could use some explanation.

Cascading Relation Fields

DataMgr has several types of relation fields (see [docs](#) or any [recorded presentation](#)). In DataMgr 2.0, a relation field could use any field from any other (related table). As of DataMgr 2.1, it can also use any relation field from any other (related table) - except "list" relation fields.

For example, you could concatenate a label relation field with an aggregate relation field or perform an aggregate calculation on an aggregated value from another table.

Custom Relation Fields

Although the built-in relation field types should cover most purposes, you can expand on them if you want by using type "custom". Then you can pass in a "sql" attribute holding the value of the SQL for the subquery (or simple SQL statement) that will return a value for this relation field.

If you provide a "CF_Datatype" attribute in the "relation" element, then you can also have DataMgr filter by the value, just as it does with any other field (no saving, though).

If you don't provide a "sql" attribute, DataMgr will return an empty string as the value for that column.

advsql

Enhancing the flexibility of DataMgr further, you can pass an "advsql" argument to getRecords() as a structure. This structure can have keys including SELECT, FROM, WHERE, ORDER BY. The value of the key will be the sql that you want to append to the given portion of the query.

Conclusion

Note that although DataMgr_Sim is included in the zip file, it hasn't yet been updated for the new functionality - hopefully I will get to that before the first beta.

If you have the time, download [DataMgr 2.1 Alpha 1](#) and try it out. Let me know when you find bugs so I can get them worked out for the beta.