Subject: [FEATURE REQUEST] Add Jsonize() to S_ structs
Posted by omari on Wed, 26 Jul 2023 17:11:16 GMT
View Forum Message <> Reply to Message

Hi,
I need to serialize a recordset using json, i use this code:

```
Array<S_USER> a;
Sql sql;
//...
String js = StoreAsJson(a);
```

in order to let this work, S_USER (generated from sch file) need to have Jsonize().

this patch add Jsonize to generated S_TABLE structs.

```
diff --git a/uppsrc/Sql/S_info.cpp b/uppsrc/Sql/S_info.cpp
index 0b89e8236..b47c99e27 100644
--- a/uppsrc/Sql/S_info.cpp
+++ b/uppsrc/Sql/S_info.cpp
@@ -78,6 +78,23 @@ void S_info::Set(const void *s, const ValueMap& m) const
        }
 }

+void S_info::Jsonize(const void *s, JsonIO& json) const
+{
+       if(json.IsStoring()) {
+               for(int i = 0; i < column.GetCount(); i++) {
+                       Value v = Get(s, i);
+                       json(column.GetKey(i), v);
+               }
+       }
+       else {
+               for(int i = 0; i < column.GetCount(); i++) {
+                       Value v;
+                       json(column.GetKey(i), v);
+                       Set(s, i, v);
+               }
+       }
+}
+
 SqlSet S_info::GetSet(const String& prefix) const
 {
        SqlSet set;
diff --git a/uppsrc/Sql/SqlSchema.h b/uppsrc/Sql/SqlSchema.h
index d4a9f797b..4d222c5d5 100644
```

```
--- a/uppsrc/Sql/SqlSchema.h
+++ b/uppsrc/Sql/SqlSchema.h
@@ -171,6 +171,8 @@ struct S_info {
       void     Set(const void *s, const SqlId& id, const Value& v) const;
       void     Set(const void *s, const ValueMap& m) const;

+       void     Jsonize(const void *s, JsonIO& json) const;
+
       SqlSet   GetSet(const String& prefix) const;
       SqlSet   GetOf(const SqlId& table) const;

diff --git a/uppsrc/Sql/sch_header.h b/uppsrc/Sql/sch_header.h
index 3ae50ccfc..a15c4a241 100644
--- a/uppsrc/Sql/sch_header.h
+++ b/uppsrc/Sql/sch_header.h
@@ -37,6 +37,7 @@ public: \
       void                    Set(int i, const Upp::Value& v)       { return info->Set(this, i, v); } \
       void                    Set(const Upp::SqlId& id, const Upp::Value& v)  { return info->Set(this, id,
v); } \
       void                    Set(const Upp::ValueMap& m)                   { return info->Set(this, m); } \
+       void                    Jsonize(JsonIO& json)              { return info->Jsonize(this, json); } \
       \
       operator                  const S_info&() const              { return *info; } \
       \
```