



## PROJECT DOCUMENTATION

Project Name:	<b>Synopse mORMot Framework</b>
Document Name:	<b>Design Input Product Specifications</b>
Document Revision:	<b>1.17</b>
Date:	<b>September 9, 2012</b>
Project Manager:	<b>Arnaud Bouchez</b>

### Document License

THE ATTACHED DOCUMENTS DESCRIBE INFORMATION RELEASED BY SYNOPSE INFORMATIQUE UNDER A GPL 3.0 LICENSE.

*Synopse SQLite3/mORMot Framework Documentation.*

Copyright (C) 2008-2012 Arnaud Bouchez.

Synopse Informatique - <http://synopse.info..>

This document is free document; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 3 of the License, or (at your option) any later version.

The *Synopse mORMot Framework Documentation* is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this documentation. If not, see <http://www.gnu.org/licenses..>

### Trademark Notice

Rather than indicating every occurrence of a trademarked name as such, this document uses the names only in an editorial fashion and to the benefit of the trademark owner with no intention of infringement of the trademark.

Prepared by:	Title:	Signature:	Date
Arnaud Bouchez	Project Manager		

## Document Purpose

The *Design Input Product Specifications* document purpose is to create high level description of software specifications for the *Synopse mORMot Framework* project.

The current revision of this document is 1.17.

# 1. System Specifications

---

This document is intended to describe the Design Input Product Specifications.

## 1.1. Definitions

**Added Value** - This level of achievement should be the target of the design team, because achieving this level of performance adds value to the product. However failure to achieve this level does not evoke additional management review.

**Must Have** - This level of achievement must be reached in the final design output. Because of possible negative financial impacts, if this level of performance is not achieved, management review will be triggered.

## 1.2. Project Concept

### 1.2.1. Purpose and Scope

This document focuses on the *Synopse mORMot Framework* library.

The purpose of this *Design Input Product Specifications* (DI) document is to detail the marketing requirements/product specifications for the 1.17 release of the *Synopse mORMot Framework library*. The requirements and specifications found in this document are derived from customer market research, regulatory input and industry common practice.

### 1.2.2. Concept Statement

It was observed that a true JSON and RESTful oriented Client-Server framework was missing in the Delphi programming environment.

Latest versions of Delphi (i.e. Delphi 2010/XE/XE2) provide a JSON and RESTful mechanism named DataSnap (in the *Architect* or *Enterprise* editions), but such a feature could be implemented with previous versions of the Delphi compiler as well, with a more open architecture.

This framework shall use a innovative ORM (Object-relational mapping) approach, based on the RTTI (Runtime Type Information) provided by the Delphi language. It shall expose Server data access and business services to Clients, using JSON over several communication protocols.

After evaluation of most used database engines, the *SQLite3* engine was found out to be secure, fast, and perfectly adapted as a stand-alone database engine for this framework, able to access other

remote database engines using its unique *Virtual Tables* mechanism.

Together with this Client-Server data and business architecture, a set of User Interface components (especially Database Grid and Reporting system) are provided within the framework.

The main approach of this framework is to avoid RAD in the development of projects. RAD has been proved to be a good candidate about prototyping, but is not the best approach for creating a robust and maintainable application. Best practices (as MVC, n-Tier or SOA) shall be used instead.

### 1.3. Expected Use

Any application which need moderate database usage (up to some GB of data) with easy setup and administration, together with a secure ACID behavior in a Client-Server environment should consider using the *Synopse mORMot Framework*.

### 1.4. Requirement Exceptions

This framework was developed in order to run under any Delphi compiler, from version Delphi 6 to version Delphi XE2.

It was conceived so that it could be compatible also with the Free Pascal Compiler, which is more advanced than the Embarcadero Delphi compiler for cross-platform support. This support is not tested, but was taken in account during coding.

#### 1.4.1. License

The framework source code is licensed under a disjunctive three-license giving the user the choice of one of the three following sets of free software/open source licensing terms:

- *Mozilla Public License*, version 1.1 or later (MPL);
- *GNU General Public License*, version 2.0 or later (GPL);
- *GNU Lesser General Public License*, version 2.1 or later (LGPL).

This allows the use of the framework code in a wide variety of software projects, while still maintaining copy-left on code Synopse wrote.

In short:

- For GPL projects, use the GPL license - see <http://www.gnu.org/licenses/gpl-2.0.html>.
- For LGPL license, use the LGPL license - see <http://www.gnu.org/licenses/lgpl-2.1.html>.
- For commercial projects, use the MPL License - see <http://www.mozilla.org/MPL/MPL-1.1.html> - which is the most permissive.

In all cases, any modification made to this source code **should** be published by any mean (e.g. a download link), even in case of MPL. If you need any additional feature, use the forums and we may introduce a patch to the main framework trunk.

You do not have to pay any fee for using our MPL/GPL/LGPL libraries.

But please do not forget to put somewhere in your credit window or documentation, a link to <http://synopse.info>.. if you use any of the units published under this tri-license.

For instance, if you select the MPL license, here are the requirements:

- You accept the license terms with no restriction - see <http://www.mozilla.org/MPL/2.0/FAQ.html>.

for additional information;

- You have to publish any modified unit (e.g. SynTaskDialog.pas) in a public web site (e.g. <http://SoftwareCompany.com/MPL>), with a description of applied modifications, and no removal of the original license header in source code;
- You make appear some notice available in the program (About box, documentation, online help), stating e.g.  
*This software uses some third-party code (C) 2012 Arnaud Bouchez provided by Synopse - <http://synopse.info> - under Mozilla Public License 1.1; modified source code is available at <http://SoftwareCompany.com/MPL>.*

Note that this documentation is under GPL license only, as stated in this document front page.

## 2. Software Design Input

---

The Software *Design Input Product Specifications* (DI) document items follow these main divisions:

- Client Server JSON framework (page 6)
- SQLite3 engine (page 6)
- User interface (page 7)

### 2.1. Client Server JSON framework

#### Design Input 2.1.1 (Initial release)

**Must Have** - The framework shall be Client-Server oriented

#### Design Input 2.1.1.1 (Initial release)

**Must Have** - A RESTful mechanism shall be implemented

#### Design Input 2.1.1.2 (Initial release)

**Must Have** - Communication should be available directly in the same process memory, or remotely using Named Pipes, Windows messages or HTTP/1.1 protocols

#### Design Input 2.1.2 (Initial release)

**Must Have** - UTF-8 JSON format shall be used to communicate

#### Design Input 2.1.3 (Initial release)

**Must Have** - The framework shall use an innovative ORM (Object-relational mapping) approach, based on classes RTTI (Runtime Type Information)

### 2.2. SQLite3 engine

#### Design Input 2.2.1 (Initial release)

**Must Have** - The *SQLite3* engine shall be embedded to the framework

#### Design Input 2.2.2 (Initial release)

**Must Have** - The framework libraries, including all its *SQLite3* related features,

**shall be tested using Unitary testing**

#### **Design Input 2.2.3 (Initial release)**

**Must Have** - The framework shall be able to access any external database, via OleDb, ODBC or direct access for Oracle (OCI) or SQLite3 (for external database files)

### **2.3. User interface**

#### **Design Input 2.3 (Initial release)**

**Must Have** - User Interface and Report generation should be integrated

#### **Design Input 2.3.1 (Initial release)**

**Must Have** - An User Interface, with buttons and toolbars shall be easily being created from the code, with no RAD needed, using RTTI and data auto-description

#### **Design Input 2.3.2 (Initial release)**

**Must Have** - A reporting feature, with full preview and export as PDF or TXT files, shall be integrated