

# ODABA Releases TODBMS and Tools 15.0.0

ODABA is a Terminology-Oriented Database Management System (TODBMS) based on standards for object-oriented databases (ODMG 2003). In contrast to other databases that are focused on big data processing, ODABA stands for smart data processing, i.e. it is intended to be used for complex problems and complex data structures in combination with complex processing rules.

The latest version of ODABA has been released on Saturday, Jan 29th, 2018. With ODABA 15.0.0 support aggregation views has been provided and performance has been improved. With version 15 ODABA provides a full P<sub>3</sub> database model by supporting aggregation views.

ODABA 15 provides extended view support. Script-debugging has been improved and performance has been increased. ODABA 15 is ready for being compiled with C++ 14. The ODABA GUI framework works now with Qt 4.8 to Qt 5.10 (tested with 4.8, 5.9 and 5.10). Finally, several bugs have been detected and removed.

More details are described in change logs and in notices delivered with the development databases (ODE tools: **Object/Notices**). Notices delivered with the databases also contain a list of open topics planned for next releases. Notices are stored separately for basic functions (**sos.dev**), database kernel (**opa.dev**), GUI framework (**gui.dev**) and ODE tools (**ode.dev**).



**run Software-Werkstatt GmbH**  
**Weigandufer 45**  
**12059 Berlin**

Tel: +49 (30) 609 853 44  
e-mail: [run@run-software.com](mailto:run@run-software.com)  
web: [www.run-software.com](http://www.run-software.com)

Berlin, October 2012

## Detailed changes (ODABA)

Focus for this release was providing support for aggregation model (P3 database model). In order to support aggregation views, **VIEW** definition has been extended. More over, performance has been improved and several minor extensions have been made. Besides, several bugs have been removed, which are reported in the change log.

### ODABA Database kernel (base)

This version contains an essential number of bug fixes especially for external file access. Functionality for external file access has been extended and improved. In addition, this version provides several improvements and extensions for the kernel functionality:

- Support aggregation views  
The **VIEW** definition has been extended by an aggregation type (default is "no aggregation" for simple views). However, defining an aggregation view will change the view behavior essentially.
- Support C++ 14 (GNU++ 14 and DevStudio 2017)  
Using optimizing compiler (14), `this` pointer checks must not be optimized. Use `-fno-delete-null-pointer-checks` option when using GNU optimizing compiler. By using DevStudio 2017 compiler, performance could be increased by 20%.
- Event handler and Action calls  
In order to improve application performance, event handler and actions are now cached in an action cache.
- Import/export functions for hierarchical file types  
Hierarchical file types are supported.
- Database functions  
In order to improve error analysis, several functions have been provided for displaying instance states and instances by identity and record and analyze database locks (TraceLocks option). This includes function for loading data from a **Binary** object into an instance.
- Locking  
In order to avoid errors when database header is locked, waiting time has been increased from 10 to 100 seconds. Internal locking algorithms have been improved in order to avoid deadlocks.
- Lock analysis  
In order to detect locking problems in an application, a lock trace file may be recorder when setting the file location in the TraceLocks option.
- Option names changed  
In order to provide comprehensible option names, ODABA2 option has changed to ODABA\_TEST and NET changed to SHARE. Old option names are still valid (until version 16)
- Mark virtual OSI interface functions  
ODABA (context interface) functions that are virtual are marked in the OSI interface as such in order to be able to overload those functions with OSI functions.
- File lock functions  
Functions that allow locking file positions are provided, which supports file locking also from within OSI functions.
- Closing property handles  
Closing property handles explicitly with `closeAll` changed for transient properties. Now, the assigned property handle will be closed, but not the transient property handle.
- Data exchange  
Data exchange does not convert data from c-string anymore (i.e. `\n`, `\t` and other escape sequences are imported and exported as they are. Instead, the `REPLACEMENTS` option allows defining more sophisticated user defined data transformation when importing or exporting file data.
- Apply changed access mode to related property handles  
In order to change access mode not only for the property handle hierarchy, but also for related property handles, an additional `accessMode()` function has been provided.
- Attribute initialization

Subordinated attributes of attributes with complex data type had not been initialized with initial value defined in the schema. Now, all attributes will be initialized properly.

## ODABA Application Program Interface (base/opa)

Several extension and minor changes have been made on API functionality.

### Interface changes:

Basic classes (namespace **odaba**)

- **Binary**
  - assign (new)
  - fromInstance (new)
  - toInstance (new)
- **Database**
  - entryStatus (new)
  - unerase (new)
- **ObjectSpace**
  - contextEventHandling (new)
  - get (new)
- **Property**
  - accessMode (changed)
  - collectionLockedForUpdate (new)
  - lockCollectionForUpdate (new)
  - nextKey (changed)
  - unlockCollectionForUpdate (new)
  - value (changed)
- **Value**
  - value (changed)

Service classes (namespace **odaba::utils**)

- **BinaryFile**
  - lastPosition (moved to **odaba::utils::File**)
  - extension(new)
  - tryErase (new)
  - tryEraseDirectory (new)
- **File**
  - lastModified (new)
  - lastPosition (moved from **odaba::utils::BinaryFile**)
  - lock (new)
  - locked (new)
  - unlock (new)
- **XMLString** (functions marked as const)

More details are described in ODABA online documentation: **Reference documentation/ODABA Application Program Interface.**

## **ODABA Script Interface OSI**

Functions implemented in C++ can be marked as virtual now in the ODABA Interface function. This allows overloading virtual system and application (C++) functions with virtual OSI (script) functions defined in the application. Virtual context functions as `input`, `output` or `executeLogin` have been marked as virtual in the OSI interface.

In order to improve performance when testing, reloading OSI functions has been optimized.

Extended API functions have been provided at OSI function interface.

## **Open Document Support**

No changes but a few bugs have been removed.

## Detailed changes (ODE and GUI framework)

The ODABA GUI framework works now with Qt 4.8 to Qt 5.10 (tested with 4.8, 5.9 and 5.10). Several extensions and bug-fixes have been made. Beside supporting the aggregation model (extended **VIEW**), several minor extensions and bug-fixes have been made:

- **Complex table headers**  
The GUI framework supports vertical header columns as well as multiple line headers for tables. Algorithms for resizing auto-size columns has been changed in order to support resizing for multiple header columns. After resizing an auto-size column explicitly, it becomes a fixed size column.
- **Changing columns at run-time**  
The GUI framework supports adding and removing columns to/from regions and columns. Adding columns to a table/tree region allows passing a control name for defining column layout. Moreover, sub columns may be added to columns in order to create multiple table headers at run-time.
- **Label manipulation features**  
Extensions have been made in order to change label layout separate from region's cell layout..
- **Frequent data storing**  
Better support for shared databases (multiple user applications) is provides by saving and refreshing instance data when changing focus (leave and enter) - `saveDataEnabled()`.
- **Support common event handler**  
In order to provide event handlers handling all GUI events submitted by application, event handling notification may be enabled (`eventNotificationEnabled()`).
- **Lock analysis**  
In order to trace and analyze database locks, scripts have been provided for loading lock trace files (`TRACE_LOCKS` option) into `ObjectLocks` extent. (`eventNotificationEnabled()`).

### GUI Framework (gui)

GUI framework functionality has been extended. Moreover, GUI framework now runs with Qt4.8 and Qt 5.9. In order to adopt Qt 5.9, some minor changes become necessary (conditional compilation depending on Qt version). Besides, several bug-fixes have been made. Running GUI framework with Qt higher than 5.0 does not support user controls provided by odaba.

### ODE tools (ode)

ClassEditor and Designer have been extended. Most important extensions are:

- **Support aggregation model in GUI applications**  
In order to define an aggregation model in **ClassEditor**, one has to define a **VIEW** with an appropriate aggregation type. For standard views, the aggregation type is "no aggregation".  
GUI applications also allow defining aggregation trees supporting drill down in aggregation models.
- **Define multi-line column headers**  
In order to define multi-line column headers, style options for controls got a new option `multi-line` indicating a multi-line table header. Controls not supporting this feature will ignore this option. The `sorted` option, which has no effect on style level, has been removed.
- **Overload GUI actions**  
In order to overwrite standard settings for default action layout, application specific actions with default action name may be defined in **Designer**.
- **Check expressions in ad-hoc filters or actions**  
The standard form supporting ad-hoc expressions provides checking functionality and member selection.

## ODABA GUI Application Program Interface (gui/ode)

Additional functions in GUI context interface have been provided

- **AligmentTypes** (new enumeration)
- **Column**
  - addColumn (new)
  - isValid (new)
- **ControlContext**
  - addColumn (new)
- **Layout**
  - labelColur (new)
  - labelFont (new)
  - labelTextColur (new)
- **ProjectContext**
  - eventNotification (new)
  - eventNotificationEnabled (new)
  - saveDataEnabled (new)
- **Region**
  - addColumn (new)
  - columnId (new)
  - isValid (new)
  - regionId (new)

## ODABA Documentation

Extensive documentation has been added for the aggregation model and is available at

[http://www.run-software.com/inhalt/documentation/odaba/documents/opa/HierarchyTopics/OUG\\_Aggregation\\_schema.html](http://www.run-software.com/inhalt/documentation/odaba/documents/opa/HierarchyTopics/OUG_Aggregation_schema.html) or via [online documentation](#) (ODABA base) at **OnlineDocumentation/User's Guide / Spoeical Features / Aggregation schema**.

Documentation has been updated and new features have been documented.

## Installing ODABA

ODABA, including applications and libraries, is available for free under Open Source licenses (GPL). ODABA runs on various hardware configurations, operating systems and works on many desktop environments. ODABA can be obtained as source code distribution and in various binary formats from <http://sourceforge.net/downloads/odaba/>.

Several features require third party components, which have to be installed before installing ODABA. When the corresponding libraries are not available, one may install ODABA, but the features referenced below will not work.

- libzip - required for LibreOffice document generation
- zlib - required for data compression and database backup and restore)
- curl - required for enhanced email support)
- hunspell - required for spell check in ODE tools, like terminus

Using optimizing compiler (14), `this` pointer checks must not be optimized. Use `-fno-delete-null-pointer-checks` option when using GNU optimizing compiler.

## Previous Releases

With the release of ODABA 15.0.0 we declare the end of live for all previous released ODABA versions less than version 14.0.0. Bug fixes on 13.2.x version are provided on demand.

System model has been changed (extended), but no version upgrade is required. However, development tools do not run with old resource database (`ode.dev`).

Changes affect view definitions. Views have been moved from functional model (`ODC_View`) to database model and data type had been renamed to `SDB_View`. When views have already been stored in a resource database, the resource database has to be updated:

- exporting view definitions to an XML file (using old **ode.dev** database)
- rename `ODC_` to `SDB_` in xml file
- import updated XML file (using new **ode.dev**)

## System Requirements

In order to get the most out of this release, we recommend to use a recent computer with at least 1 GB of memory and 2 GHz CPU or better. In order to install the binaries, about 100 MB are required. Installing sources requires about 50 MB. 80 MB are required in addition, when installing the documentation locally.

## About RUN-Software

RUN-Software develops database management system ODABA and tools since 1994. Besides general and particular software solutions, RUN-Software publishes theoretical works about database theory and terminology in connection with data modeling.

See also: [www.run-software.com](http://www.run-software.com)