

The ExESS API

ExESS is an excellent tool for EH&S Data Management. For REACH , ExESS is the best SIMS (Substance Inventory Management System) tool on the Market . For overall Chemicals Management, ExESS can manage Substance Volume Tracking, Safety Data Sheet Authoring, GHS classifications and many other functions. For those reasons, ExESS gives anyone the flexibility to integrate this EH&S Tool with other IT and ERP systems. For some integration work, the ExESS Application Programming Interface (ExESS API) is an interesting and powerfull tool.

- The API has a cost of €2000/year and is delivered with minimum 1 day of training and exercises for €1000/day.

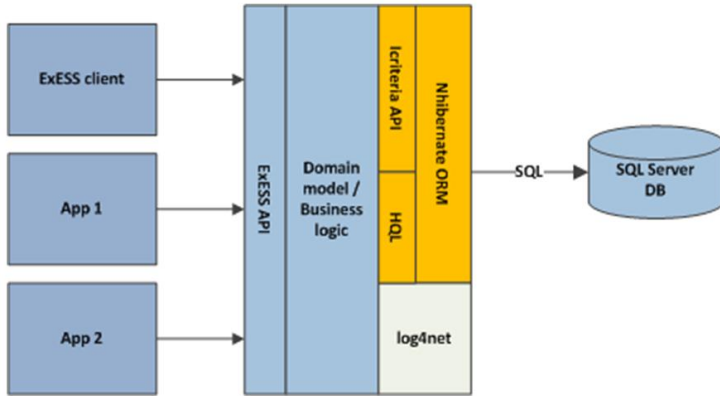


General

The Application Programming Interface (API) is a source code based specification intended to be used as an interface by software components to communicate with Exess. The API includes a description of a set of class definitions, with a set of behaviors associated with those classes. A behavior is the set of rules for how an object, derived from that class, will act in a given circumstance. This abstract concept is associated with the real functionalities exposed, or made available, by the classes that are implemented in terms of class methods (or more generally by all its public components hence all public methods). The API in this case can be conceived as the totality of all the methods publicly exposed by the classes (usually called the class interface). This means that the API prescribes the methods by which one interacts with/handles the objects derived from the class definitions. More generally, one can see the API as the collection of all the kinds of objects one can derive from the class definitions, and their associated possible behaviors. Again: the use is mediated by the public methods, but in this interpretation, the methods are seen as a technical detail of how the behavior is implemented.

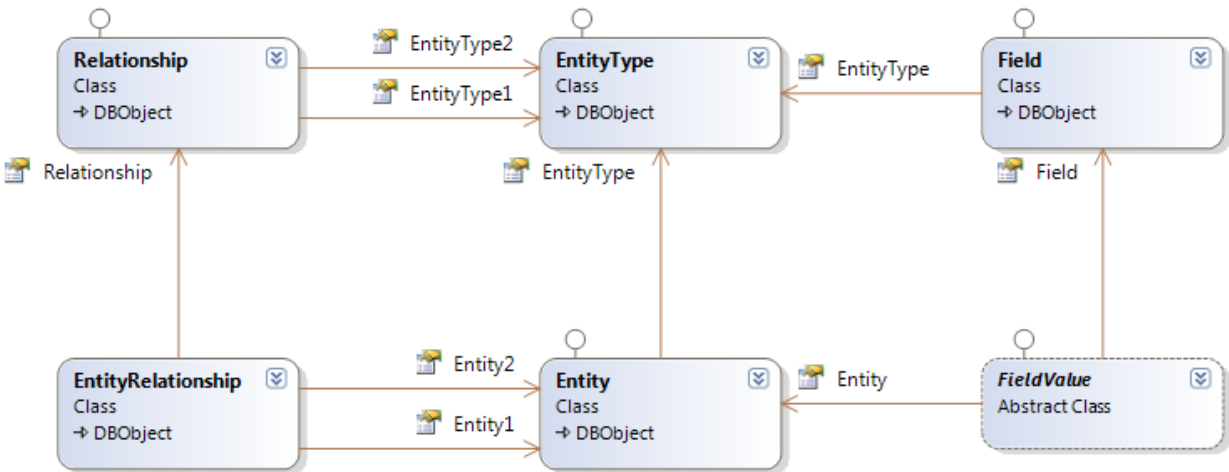
Architecture

The scheme below shows the architectural design of Exess. The Exess API serves as interface between the Exess client and all other third-party applications and the Exess domain model/business logic.



Object model

The object model of Exess shown below illustrates the 6 major classes and their relation towards each other. For an Entity type several Fields can be defined. Entities can be created, having Field values for each of the defined Fields. Between Entity types Relationships can be setup. This way different entities can be related to each other.



Example – searching for entities

The Entity FindEntities method returns all entities of a specified entity type that conform to the given search parameters:

```
public static List<Entity> FindEntities(EntityType entityType,
                                       Field[] searchFields, object[] searchValues)
```

Example: find the products with CAS numbers “107-21-1” or “7726-95-6”

```
using (Session session = ExessManager.DataStore.CreateSession())
{
    foreach (Entity product in Entity.FindEntities(Predefined.EntityTypeProduct,
                                                  new Field[] { Predefined.FieldProductCASNumber, Predefined.FieldProductCASNumber },
                                                  new object[] { "107-21-1", "7726-95-6" },
                                                  LogicalOperator.Or, StringMatchMode.Exact))
    {
        string cas = product.GetFieldValueAsText(session, Predefined.FieldProductCASNumber);
        Console.WriteLine("CAS {0} found!", cas);
    }
}
```

Other Integration tools

ExESS communicates with other IT application using :

- File exchange (XML or other structured format using MS Windows Scheduled task)
- ExESS API
- SAP Environment, Health and Safety - Open Content Connector (EH&S-OCC)



Dirk Stevens and Frans Lemaire
Product Marketing SH&E - ExESS

LISAM SYSTEMS | HEMMIS Office Belgium | Vlaanderenstraat 3A - B-8501 Heule |
www.lisam.com

email : FransL@lisam.com -

Office +32 (0)56 372637 | Fax +32 (0)56 372324 | Mobile +32 (0)477 292519