OVERVIEW

Habanero is an open source Enterprise Application Framework for .Net that provides the tools to produce an agile application from the data layer through to the front-end. Released for free under the LGPL license, it persists data to and from the database using object relational mapping (ORM), manages business objects to support a rich domain model, and generates user interfaces at runtime to display an edit the mapped data.

POWERFUL AND FLEXIBLE DOMAIN MODEL

Habanero provides a developer with everything that is needed to develop highly efficient, scalable, and easily maintainable software. At its core is a highly optimised Object Relational Mapping (ORM) layer that handles all interactions of your domain model objects with the data store of your choice. All of the mappings and definitions of your domain model objects are defined in a Class Definition object model. By default, these Class definitions are loaded from an XML definition file at runtime, but can also be created programmatically which allows for some very flexible solutions. As a result the creation of your object model at runtime in order to implement 4th normal form solutions becomes as easy as implementing any other solution.

PERSISTENCE AGNOSTIC

Persistence to most mainstream relational databases (Oracle, SQL Server, MySQL, FireBird, MS Access, PostgreSQL, and SQLite), as well as files (XML, binary, etc.) and even a highly efficient in-memory data store (perfect for testing and delayed persistence) has been built into Habanero for you. Furthermore, if your required persistence strategy is not supported, you can create your own custom strategy that plugs seamlessly into Habanero, enabling the required persistence support.

In Habanero, the persistence and domain layers have been carefully separated and cleverly architected so that you can write an entire application without having to settle on a persistence layer until the time when you need to do final testing and deployment of the finished product. The only change that is required to move your Habanero application to a different database vendor is a simple setting in your application configuration file. The beauty of this is that it allows the developer to create the application in a truly Agile fashion, not weighed down by dependencies on a particular backend data store and structure.

When using Habanero, you will notice that you never have to think about SQL, or files, or any other persistence strategy. All of this is done for you. The Habanero SQL engine handles everything, from querying your database to inserting, updating and deleting your data records. Due to the intelligence built into the Habanero object model, you are able to query across relationships without having to think about SQL Joins and you can do efficient updates of the data in a table for only the fields that have been modified.

All of the major persistence concerns of a developer are addressed by Habanero so that more time can be spent on the actual problem domain. Habanero handles the persistence technology, concurrency control, transactions and audit logging for you, without any additional effort.

EASY TO USE

A major component of Habanero’s power lies in its simplicity. The object model of the framework just makes sense. Even though there is a bit of a learning curve, as there is with all frameworks, the time to get to your first working solution is considerably less. Once the developer has got to grips with the domain driven approach that Habanero promotes, the most complex tasks become much easier to implement.

A tool called FireStarter is included with Habanero to provide a visual interface for the creation of your domain model objects and rules as well as your application code. Using FireStarter, a developer is able to iteratively generate all of the domain model code (as well as supporting Test Fixtures) for his application, allowing him to focus on the application logic and flow instead. FireStarter also supports the ability to reverse engineer an existing database and create a domain model that matches the database’s structure and rules. The use of FireStarter with Habanero makes the development and maintenance of applications faster, easier, and more reliable.

SECURITY AND VALIDATION AT DOMAIN LEVEL

Security and validation of the object model are supported at the domain model level of the application, so that accessing the same object from multiple User Interfaces does not give inconsistent access and validation rules. The validation rules on the domain model can be defined within the Class Definitions, or as custom code. The domain model security checks are not bound to a particular security provider, so that you can implement this for your particular environment and security requirements.

USER INTERFACE AUTOMATION AND AGNOSTICISM

One of the notable features of Habanero is the dynamic generation of user interfaces for editing and managing your domain model objects. Multiple views can be defined for one object, from which dynamic controls can be created at runtime. These dynamic controls provide you with a user interface that automatically handles the binding and synchronisation of the domain model objects’ properties to the corresponding controls. Additionally, custom controls and mappers can be created by the developer and used in the dynamic user interfaces to fulfil a specific user interface requirement.

Habanero’s UI implementation does not bind itself to a specific user interface technology but rather uses a Factory pattern for the creation of all of the user interface controls. Out of the box Habanero supports Windows forms and Visual WebGui user interface client types. One of the major advantages of the control factory is that you can convert a Windows forms application into a Visual WebGui application with considerably little effort. This also allows for the creation of user interfaces that are much more testable.

STABILITY

Habanero is a very stable framework with many solutions successfully deployed all over the world. The framework has been developed with a Test Driven Development approach ensuring that the code base is of a very high quality. The framework is currently covered by more than 7500 unit and acceptance tests and the core domain layer library has test coverage of almost 90%. There is also a very active community working on the framework with very deliberate focus on establishing Habanero as the standard for high quality application development.

SUMMARY

So, why Habanero? There are so many other frameworks out there all achieving pretty much the same thing. It is Habanero’s Object Model and flexibility that makes it stand out above the rest.

.......