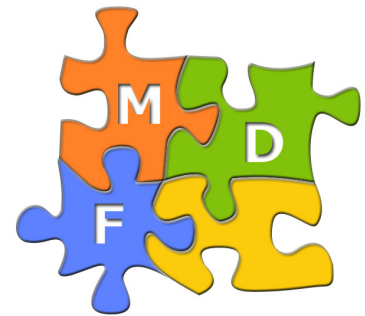


# Getting started with Mdf

Mdf – MES Development Framework

Workshop



Welcome to the workshop: Getting started with Mdf

Today we will see some fundamental concept on how to develop complex MES applications with the Mdf library according to the ISA-95 standard.

## Mdf - MES Development Framework

A collection of assemblies:

- EquipmentModel (ISA 95)
- MaterialModel (ISA 95)
- ProductionSchedule (ISA 95)
- PersonnelModel (ISA 95)
- ... (ISA 95)
- OrderModel
- SecurityModel
- Utility

The Mdf acronym stands for MES Development Framework.

This library is composed by different dot net assemblies that allow you to develop a MES application, quickly and with little effort and in compliance with industry standards.

In this example we are going to demonstrate:

- The Equipment Model, that allows you to define the model of the plant.
- The Material Model, that allows you to manage all materials from raw materials to finished goods, lots, and handling units.
- The Production Schedule, that allows to manage the process orders and scheduling.
- The Personnel Model, that gives you an interface to manage the personnel with specific skillsets in the shop floor operations.

We supply assemblies for every model described in the ISA-95 standard.

The library includes several external assemblies, for example:

- the order model to manage order like incoming or shipping orders.
- the security model to manage authorization in the MES application.
- several utilities to manage sequences, parameters, constants and so on.

## Mdf - MES Development Framework

Develop complex MES solutions faster

- Integrated in Visual Studio .NET 2008
  - Addin
  - Solution template
- Custom Class Generator
  - Table generator
  - Class generator (c# code)
- Compiled code for high performance

The Mdf library is integrated into the Visual Studio dot net 2008 IDE as an add in and a c sharp project template.

The main characteristic of the add in is the CCG: Custom Class Generator. This tool, starting from the data configuration, generates a series of classes. The result will be only compiled code, which optimizes the performance.

The CCG tool also generates several custom tables to allow for a high performance storage layer.

ISA 95 Compliant - [www.isa-95.com](http://www.isa-95.com)

- Three categories of resources
  - Equipment
  - Material (and Energy)
  - Personnel
- Combined resources
  - Process Segments
  - Capability Definition (by time & by segment)
- Production-Process-Product
  - Product Definition
  - Production Schedule
  - Production Performance

Now let us take a look to the ISA-95 standard.

The ISA-95 is an international standard for the integration between enterprise and control systems. ISA-95 consists of models and terminology.

These can be used to determine which information has to be exchanged among systems for sales, finance, and logistics, and systems for production, maintenance, and quality.

This information is organized in UML models, which are the basis for the development of standard interfaces between ERP and MES systems.

The ISA-95 standard consists of several models, for example models of the physical structure of the enterprise, models of production procedures, models of information flows, and object models.

A model is a useful representation of a specific attribute or behavior.

Models are useful because they describe or mimic reality without dealing with every detail of it.

They typically help people analyze a situation by combining ideas with information about that specific situation.

Models help us make sense of the world's complexity.

In this slide we can see the main models:

- Equipment model
- Material model
- Personnel model

These three models are called the resources model

The combined resources are modelled by:

- Process segment model.
- Process segment capability model.
- Production capability model.

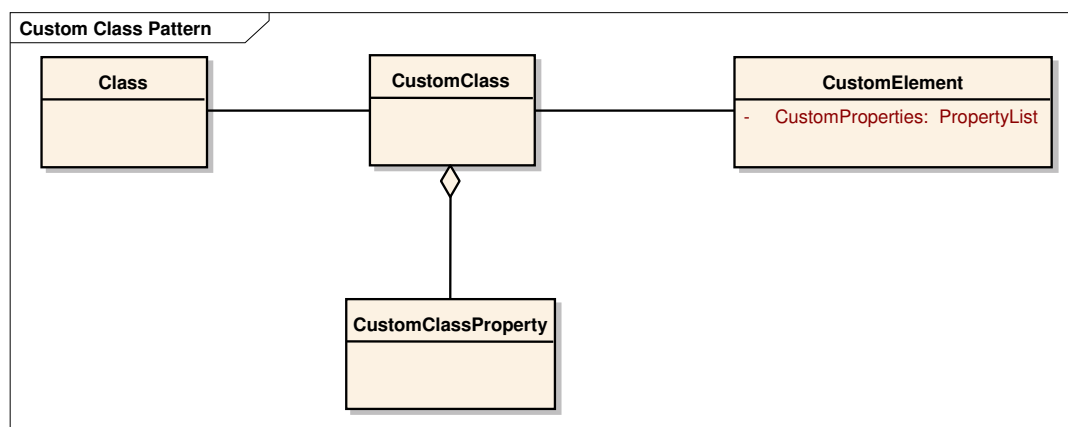
There are at least three models for Production Process or Product:

- Product Definition model
- Production Schedule model
- Production Performance model

For any further information take a look at [www.isa-95.com](http://www.isa-95.com).

With a free subscription you can download tons of documentation about this standard.

# Custom Class Pattern



Now we can see more in depth the most important tool of the MDF library.

The Custom Class Generator.

The CCG. automatically generates the classes defined by the user-developer using the Custom Class Pattern.

Here is an example.

The equipment model is where you define how the MES is divided as equipment hierarchy.

The equipment is an entity defined by the ISA-95 standard, but the standard defines only the identifier, the description, and the hierarchy of the units of equipment.

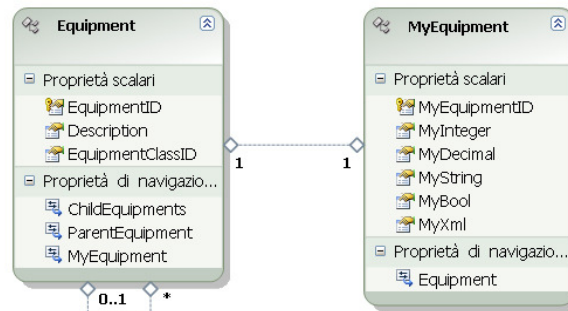
In real world Manufacturing Execution System applications this is not enough.

For example, let's say we have a generic equipment class named My Equipment that can be a work cell (according to the standard terminology).

Our class will be an ISA-95 equipment and the custom class will be, in this example, my equipment with the following custom attributes:

- My Integer
- My Decimal
- My String
- My Bool
- My Xml

## Custom equipment model



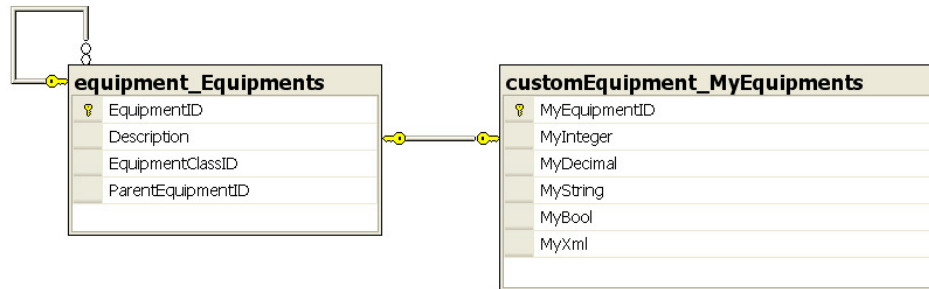
Using the definition that the user-developer has set, the custom class generator will generate a new class named “my equipment” with several properties:

- My Integer of integer type
- My Decimal of decimal type
- My String of string type
- My Bool of boolean type
- My Xml of xml type

Similarly, the custom class generator will create a table named custom equipment underscore my equipment that will have the following fields:

- My Integer
- My Decimal
- My String
- My Bool
- My Xml

## Custom equipment DB



As a last step, the custom class generator will create an entity model to ensure that a single entry point allows access to the storage layer. All this will be done using the entity framework that Microsoft has released at the end of 2008.

All the entities have characteristics of the ISA-95 standard and attributes defined by the user-developer.

## Partial class and Partial method

The Custom Class Generator implements:

- Partial class
- Partial method
  - On before add
  - On after add
  - On before modify
  - On after modify
  - On before delete
  - On after delete

All the classes generated by the custom class generator are partial classes and expose several partial methods.

Using this interesting technique, introduced by Microsoft with c sharp 3.0, it will be possible to add custom properties and custom methods, as well to add some special control during the add, modify and delete operations.

All this code can be stored in separated files so the custom class generator will not overwrite them.

## Mdf Solution Design

- Storage Layer (SQL Server 2005/2008)
- Data Access Layer (Entity Framework)
- Business Layer (Mdf, Mdf.BusinessEntity)
- Extended Business Logic (Workflow foundation / Web Services)
- Presentation Layer (Web)
- Distributed Integration (Data Services)
- Reporting (SQL Reporting Services)

The solution architecture for a real world manufacturing execution system application based on the Mdf library will be composed by the following components:

- a storage layer with SQL server 2005 or 2008. We are working to integrate our library with Oracle too.
- the data access layer. It will be implemented by the entity model using the entity framework.
- a c sharp project, Mdf business entities, to implement the business layer.
- several Web services to develop the business logic of the MES application that can be integrated with the workflow foundation.
- a web presentation layer with extensive use of the Ajax technology to manage data and supervise the production area.
- the distributed integration. It will be possible through the ADO data services.
- for reporting, we will supply several templates for SQL server reporting services that can be modified by the user-developer.

## Future work

- Allow to choose between SQL Server 2008 and Oracle 11g RDBMS
- Develop a tool for aiding the building of the presentation layer
- Develop several use cases and demos

Now we are at the end of this workshop. I'd like to talk about some future works. Our main goal is to develop an Oracle version of the framework to use Oracle as Storage Layer.

We want to supply some tools integrated in the Visual Studio dot net to improve the productivity during user interface development.

We are working on several use cases to show how our framework can be used.

Thanks for your time. Contact us for any further information, suggestion, comments and critiques to our e mail [info@empystudio.com](mailto:info@empystudio.com) or our web site [www.empystudio.com](http://www.empystudio.com)

We will be glad to develop a demo with your specification to show how our library can be used in a real world application.

Thanks and to the next workshop.