3.1. Academic works

Automatic programming and software generation are two fields where the scientists and ICT technician have worked in the past years.

These techniques are applied to different fields in real life and they do no only refer to user interface and web application generators.

3.1.1. User interface generation

A common problem in software developing is represented by the user interface. It must be standard through all the different views of the application and can be automatized. HTML or XUIL are perfect examples for the standardization of the user interfaces, each component of the same type is rendered in the same way, based on the operating system and the browser version.

There are studies that addresses this problem which propose a different approach based on software mining.

Software mining is a branch of data mining focused on mining software artefacts such as source files and database schema for useful information related to the characteristics of a system. The main idea consists in analyzing the software metadata and inspect the back-end architecture before creating a native UI suite.

It has been implemented with the project Metawidget, a framework that inspects the backend software of different technologies and allows to create user interfaces for a variety of client applications and technologies.

This topic has been addressed even some years ago for the UI of desktop applications. This guaranteed a faster developing in software engineering.

Early examples of model-based tools include Cousin [XXX-- “Design Alternatives for User Interface Management Systems Based on Experience with COUSIN] and HP/Apollo’s Open-Dialogue [“ADM-A Dialogue Manager,” in Proceedings SIGCHI'85: Human Factors in Computing Systems] that provided a declarative language in which the designer listed the input and output of the user interface. The system then generated the dialogs to display and request the data. These evolved into model-based systems, such as Mike [XXX- “Mike: The Menu Interaction Kontrol Environment], Jade [XXX- Automatic, Look-and-Feel Independent Dialog Creation for Graphical User Interfaces], UIDE [XXX-- A Second Generation User Interface Design Environment: The Model and The Runtime Architecture], ITS [XXX- ITS: A Tool for Rapidly Developing Interactive Applications.], and Humanoid [XXX-- Beyond Interface Builders: Model-Based Interface Tools].

These systems used techniques such as heuristic rules to automatically select interactive components, layouts, and other details of the interface, leading to some difficulties in control.
Developers were required to learn a new language for defining models, not helping the technique-s diffusion.

3.1.2. Web-based generation

Another branch of the automatic software generation is related to web based applications.

The main idea is to analyze the model of the application in order to be able to generate the server-side software and the related views: the software must be allow to insert, editing, validating and managing entities.

Data-entry applications are typical examples of this pattern.

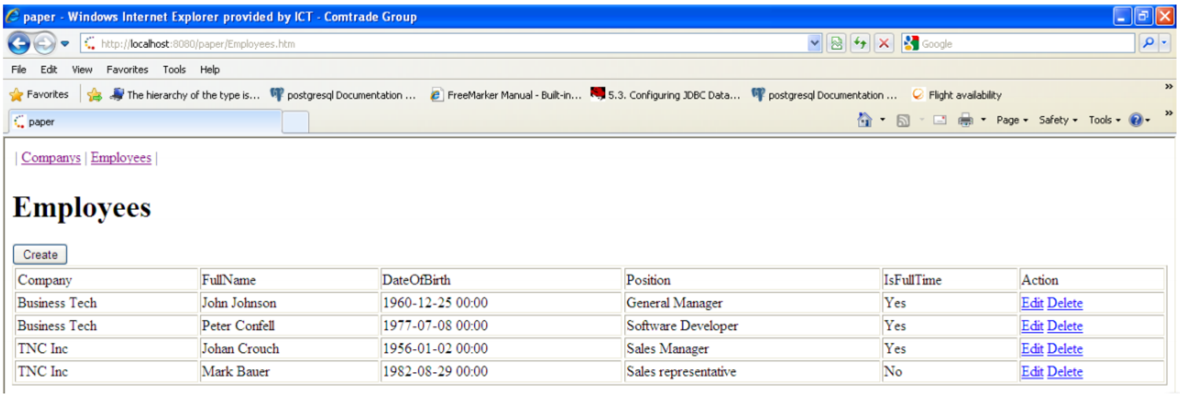
Business logic and presentation layer should be completely separated, that's why these kinds of application make use of different exchanging format for communications.

The difference between the various projects is the metamodel's specification used.

Some makes use of XML for defining elements but most are based on the Eclipse EMF framework.

EMF project is an Eclipse modeling framework and code generation facility that operates on source XMI models and produces Java classes. The input of the model may be a set of UML diagrams on top of which the framework applies model transformation in order to obtain the desired result.

A different approach [XXXX- A Generator of MVC-based Web Applications] is given by HibernateTools toolset to analyze a database's schema metadata. By obtaining the meta information it is possible to generate the required output. It uses Freemarker ad rendering technology.


paper - Windows Internet Explorer provided by ICT - Comtrade Group
http://localhost:8080/paper/Employees.htm
Google
File
Edit
View
Favorites
Tools
Help
Favorites
The hierarchy of the type is...
postgresql Documentation ...
FreeMarker Manual - Built-in...
5.3. Configuring JDBC Data...
postgresql Documentation ...
Flight availability
paper
Page
Safety
Tools

| Companys | Employees |

Employees

Create

Company	FullName	DateOfBirth	Position	IsFullTime	Action
Business Tech	John Johnson	1960-12-25 00:00	General Manager	Yes	Edit Delete
Business Tech	Peter Confell	1977-07-08 00:00	Software Developer	Yes	Edit Delete
TNC Inc	Johan Crouch	1956-01-02 00:00	Sales Manager	Yes	Edit Delete
TNC Inc	Mark Bauer	1982-08-29 00:00	Sales representative	No	Edit Delete

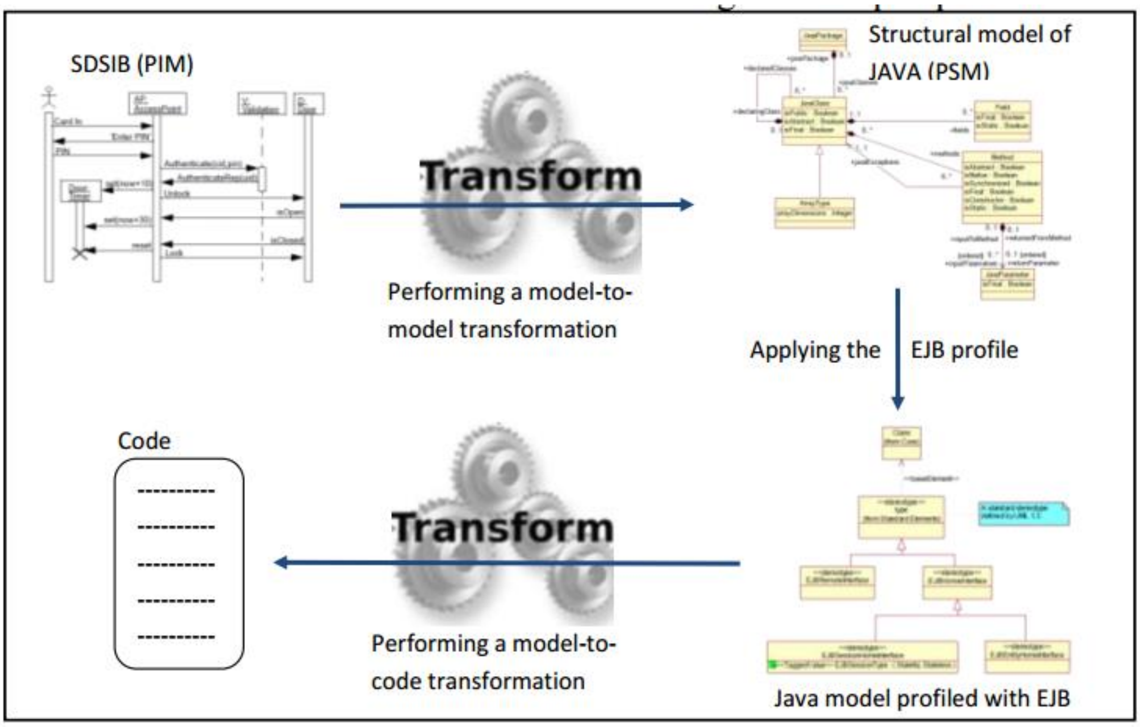
The technologies adopted are different as some researches have been developed.

[XXXX- TOWARD AUTOMATIC GENERATION OF MVC2 WEB APPLICATIONS] Is based on a source metamodel built with UML and applies transformation that lead to the target model, which is a simplified version of the relational databases's schema. Implementation of the mapping rules is based on EMF with a programming approach.

The output of the method is an XML file containing all actions, forms and forward jsp pages.

3.1.3. Code generation from sequence diagram

Starting from a UML sequence diagram is a different approach to code generation.
Sequence diagram can represents the MVC pattern and it is possible to obtain a model transformation from the source model to the target model, by means of a Java PSM as intermediate model.
Some project follows this concept [XXXXX- Automatic code generation by model transformation from sequence diagram of system's internal behavior
] and are able to generate controllers that satisfy the source use case.


SDSIB (PIM)
Transform
Performing a model-to-model transformation
Structural model of JAVA (PSM)
Applying the EJB profile
Code
Transform
Performing a model-to-code transformation
Java model profiled with EJB

The first phase consists in transforming the source model into a set of Java classes, annotated in order to support relationships. On top of this EJBs it is built an interface to support CRUD operations. The transformations follow the EMF framework.

3.1.4. Other types of code generation

Code generation is used in many application’s field above Crud software.

First use of the code generation comes together with the definition of the first programming languages. Programming languges has been developed from the first generation (machine language) till the fifth-generation (constraint-based), passing through assembly language, high level language and declarative languages [XXXX- Programming Languages: History and Fundamentals (Automatic Computation)]. Coding a programming language is the first example of code generation.

The AUTOPASS project (AUTOPASS: An Automatic Programming System for Computer Controlled Mechanical Assembly) refers to a system able to translate English-like sentences in software. It uses a geometric database generated prior to compilation and updated during the compilation, that represents the state of the worls at each assembly step.

Other project [XX – Special purpose automatic programming for 3D model-based vision] applies automatic programming to recnognizing software. Given a description of a 3D object it generates a software able to recognize the object in images, without restrictions on the orientation of the object in space.

A generator for interactive voice response applications [XXXX-https://www.google.com/patents/US6456699] has been patent in 2002. The software is able to inspect the menu section of a website and produce a IVR menu.

Automatic programming has been applied also in the robotics field [XXX- Automatic Programming of Robots using Genetic Programming]. The proposal of the project is to generate a computer program that enable an autonomous mobile robot to perform simple tasks. It does not apply reinforcement learning algorithms due to their computational and knowledge issues. It is built with the genetic programming paradigm.