- **Name :** Loan Rosseeuw

- **Name of your Level 1 : L1_IslamSouissi**

- **Paper title : User Interface Code Generation For EJB-Based Data Models Using Intermediate Form Representations**

- **Source :** scholars.google.com

- **Keywords specific to the paper :** Code generation / Data modeling

- **Summary of the main contributions :**

The use of the J2EE platform simplifies data-intensive application development by providing standard middleware services. EJB components are commonly used for implementing business logic, particularly to offer the same application functionality across various client types. This paper proposes a method for automating the generation of user interface components for data stored in EJB-based data models. These components, represented as coarse-grained forms, allow basic operations on sets of entity beans and can be deployed across multiple environments such as standalone GUI applications, web browsers, and wireless devices.

The key to this approach is the use of Intermediate Form Representation (IFR), which abstracts the functionality and layout of forms into XML documents. By leveraging IFR, developers can create equivalent user interfaces for different environments without rewriting code. The process involves generating session bean facades for each entity bean in the data model, which expose basic data operations to client forms. Additionally, IFR documents can be customized to tailor the functionality and layout of generated forms.

The paper reviews related work in model-based user interface development environments, emphasizing the lack of methods specifically tailored for EJB-based models. It also discusses the architectural design of EJB-based systems and existing tools for EJB component generation.

The generation process includes the creation of session bean facades and the structure of IFR representations, which define form functionality and layout. Forms support standard operations on beans such as addition, removal, update, copying, and query-by-form searching. Customization options include modifying form layouts and implementing custom code for additional operations.

The paper concludes by highlighting the benefits of the proposed code generation system, including rapid prototyping of applications and support for legacy relational databases through CMP entity bean layers. Future research directions include designing a visual, multi-environment IFR document editor to further accelerate application development in the J2EE realm.