

# **Integração de sistemas gerenciadores de bancos de dados relacionais e não relacionais em aplicações web**

*Miguel Salles Florido, Aline Gomes Cordeiro*

Introdução: a maioria dos sistemas de informação são criados com linguagens de programação orientadas a objetos e precisam interagir com sistemas gerenciadores de banco de dados baseados no modelo relacional, gerando problemas devido às diferenças entre os paradigmas. Esse conflito é conhecido como impedância objeto-relacional. Mesmo com as diferenças, ambos os modelos ainda são eficientes e precisam coexistir em diversas aplicações ao redor do mundo. Tendo em vista que é de fácil comunicação e interação entre linguagens orientadas a objetos e bancos de dados relacionais, tem-se como objetivo neste projeto estudar o impacto gerado ao utilizarmos um sistema gerenciador de bancos de dados não relacional (MongoDB), com a intenção de verificar se é possível contornar de forma satisfatória as impedâncias que surgem no mapeamento objeto relacional. Objetivos: investigar quais são as características que sinalizam qual é a forma mais adequada de persistir os dados: SGBD's relacionais ou não relacionais; desenvolver uma aplicação orientada a objetos para utilizar estratégias persistência relacional e não relacional de forma integrada; integrar bancos de dados não relacionais a aplicações desenvolvidas em Java; verificar como frameworks têm dado suporte aos SGBD's não relacionais. Metodologia: pesquisa sobre os tipos e características dos SGBD's não relacionais mais utilizados; levantamento bibliográfico do tema do projeto; desenvolvimento de aplicação de testes migração de bases de dados; documentação do processo anterior para criar um referencial para os estudos posteriores; pesquisa sobre os principais problemas relatados na literatura a respeito do mapeamento objeto não relacional; execução de testes nas aplicações anteriores. Resultados : Ambler (2012) aborda, entre outros aspectos, o problema relativo ao mapeamento de herança (conceito da orientação a objetos) em bancos de dados relacionais. Ambler (2012) descreve ainda 4 estratégias que podem ser empregadas para mapear superclasses e subclasses em uma estrutura relacional de tabelas, cada estratégia apresenta vantagens e desvantagens. Conclusões : foi constatado que os modelos relacional e não relacional possuem características bastante distintas e não é recomendável utilizá-los para solucionar problemas um do outro. O modelo relacional é insubstituível em cenários que demandam integridade e confiabilidade de dados, enquanto o modelo não relacional pode ser mais prático e rápido, porém com restrições como maior armazenamento em disco e menor facilidade de representação de dados com muitos relacionamentos.

*Instituto Federal Fluminense*

*Eixo temático: gerenciamento de banco de dados não relacionais em aplicações orientadas a objetos*

*Fomento da bolsa: IFF*

# **Integration of relational and non-relational database management systems in web applications.**

*Miguel Salles Florido, Aline Gomes Cordeiro*

Introduction: Most information systems are created using object-oriented programming languages and need to interact with relational database management systems, which leads to problems due to the differences between the paradigms. This conflict is known as the object-relational impedance mismatch. Despite the differences, both models are still efficient and need to coexist in various applications around the world. Considering that communication and interaction between object-oriented languages and relational databases are easy, the objective of this project is to study the impact generated by using a non-relational database management system (MongoDB) to determine if it is possible to satisfactorily overcome the impedance that arises in object-relational mapping. Objectives: investigate the characteristics that indicate the most suitable way to persist data: relational or non-relational DBMSs; develop an object-oriented application to use relational and non-relational persistence strategies in an integrated manner; Integrate non-relational databases into Java-based applications; examine how frameworks have been supporting non-relational DBMSs. Methodology: research the types and characteristics of the most commonly used non-relational DBMSs; conduct literature review on the project topic; develop test applications for database migration; document the aforementioned process to create a reference for further studies; Research the main problems reported in the literature regarding object-non-relational mapping; perform tests on the previous applications. Results: Ambler (2012) addresses, among other aspects, the problem related to mapping inheritance (a concept of object orientation) into relational databases. Ambler (2012) also describes four strategies that can be employed to map superclasses and subclasses into a relational table structure, each strategy presenting advantages and disadvantages. Conclusions: it was found that the relational and non-relational models have quite distinct characteristics, and it is not recommended to use them to solve each other's problems. The relational model is irreplaceable in scenarios that require data integrity and reliability, while the non-relational model can be more practical and faster but with constraints such as higher disk storage and less ease of representing data with many relationships.