ARTICLE 2

The document titled "Model Differences in the Eclipse Modeling Framework" discusses the importance of recording and managing design-level structural changes in software modeling and development. The authors introduce EMF Compare, an approach for calculating and representing model differences in the Eclipse modeling framework. They highlight the need for a clear separation between the calculation, representation, and visualization of model differences.

The paper starts by emphasizing the increasing complexity of software development and the shift towards model-driven engineering. Models serve as descriptions of real or imaginary phenomena and play a crucial role in software development. However, their effectiveness lies in their ability to be manipulated through automated transformations to produce various artifacts. Understanding the evolution of software systems and detecting differences between models are important for development and management practices.

The problem of model differences is complex and requires algorithms and notations that can fully utilize the potential of model-driven engineering. The authors present a conceptual framework that outlines the requirements for representing model differences as first-class entities. They propose that a solution should have a clear separation between calculation, representation, and visualization.

The calculation of model differences has been extensively researched, with various approaches ranging from text comparisons to model differencing techniques. Specialized methods have been developed for comparing UML diagrams, either based on persistent identifiers or similarity metrics. The paper highlights an approach based on structural similarity, which can compare models conforming to arbitrary metamodels.

Representation of model differences is crucial for their exploitation. The effectiveness of representation is often affected by factors such as the calculation method or the intended scope. Different representation notations, such as edit scripts or model-based representations, have their advantages and limitations. A proper representation should contain all the necessary information and enable further analysis and manipulation.

Visualization of model differences requires presenting the information in a way that is relevant to the intended goal. Concrete syntax is used to highlight specific pieces of information and provide a human-readable representation of the differences.

The paper concludes by discussing EMF Compare, which addresses the challenges of model difference calculation, representation, and visualization in the Eclipse modeling framework. The approach is metamodel independent, flexible, and interoperable. It represents the outcome of model comparison as a model, enabling further transformations.

Overall, the document provides an overview of the problem of model differences in software development and presents the EMF Compare approach as a solution within the Eclipse modeling framework.