

CAS 703 - Software Design Project Winter 2024

McMaster University

Introduction

Welcome to the world of the Carnivals, a vibrant celebration that captivates hearts and sparks joy. Carnivals are festive gatherings that draw people from far and wide, offering a range of entertainment and excitement. In the spirit of embracing this joyful atmosphere, a domain-specific language using Epsilon has been designed to capture the essence of Carnivals through metamodeling.

Metamodelling

To model the dynamic aspects of the Carnival, the diagram-based editor has been used to create the Ecore model. The design encompasses fundamental elements that define the carnival experience, ranging from diverse activities to enthusiastic participants. The metamodelling approach is structured as follows:

Class Diagram

Figure 1 illustrates the ECore Meta-model for the Carnival

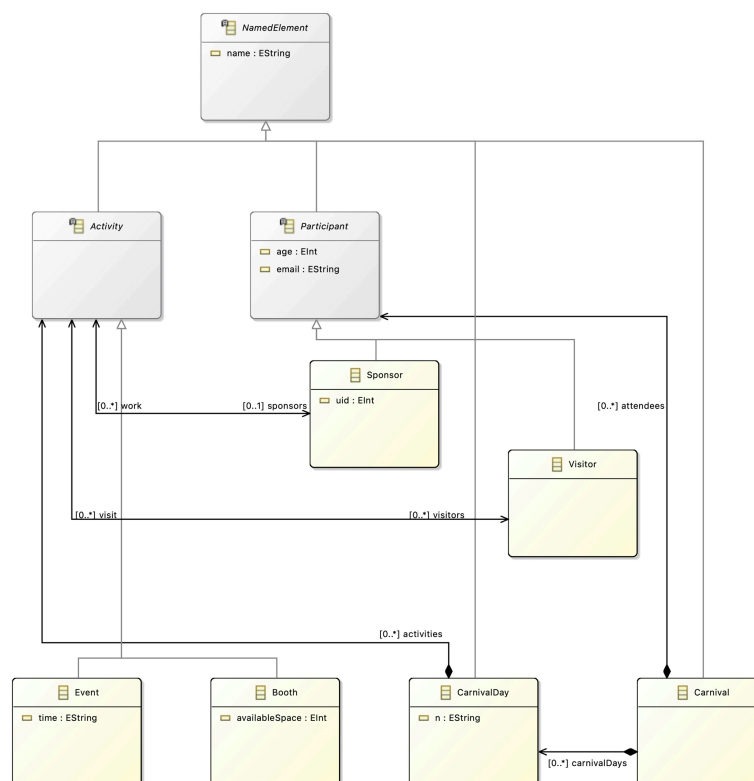


Figure 1: Class Diagram

1. **NamedElement**: An abstract class serving as the foundation for naming various elements within the carnival.
2. **Activity**: An abstract class capturing the diverse range of activities present at the carnival.
3. **Event**: A type of activity open to all carnival attendees.
4. **Booth**: A type of activity with limited resources, offering unique experiences.
5. **Participant**: An abstract class encompassing the different individuals engaged in the carnival.

6. **Visitor:** Class representing attendees who join the carnival as visitors, embracing the fun-filled spirit.
7. **Sponsor:** Class representing folks who actively contribute as sponsors, enhancing the carnival experience.
8. **CarnivalDay:** Class defining a specific day within the carnival, comprising a variety of engaging activities.
9. **Carnival:** Base class encapsulating the entire carnival celebration, composed of carnival days and diverse participants.

This metamodeling structure forms the framework for a comprehensive representation of the Carnival, offering flexibility and customization based on the unique characteristics of your specific carnival celebration. Additional details, associations, and constraints can be incorporated to tailor the model to your project's requirements.

Assumptions

- The sponsors will assume responsibility for event and booth management, thereby eliminating the need for additional helpers or staff.
- Booths remain open continuously throughout the day or until their stocks are depleted.
- Each event takes place at a distinct time during the day, and no two events can happen simultaneously on the same day.

Alternatives

In the alternate design of the Carnival meta model within the context of ECore modeling, the introduction of the "Helper" EClass would enhance the existing structure by providing a dedicated representation for staff or assistants involved in event and booth management tasks. This new EClass would establish a clear distinction between sponsors and the individuals responsible for executing various operational aspects of the Carnival.

The "Helper" EClass would contain attributes defining the roles, skills, and availability of each staff member. Additionally, it would have references to other relevant classes within the meta model, such as events or booths, to indicate the specific areas where helpers are deployed. These references would enable efficient navigation and management of helpers within the Carnival ecosystem.

By introducing the "Helper" EClass, the meta model gains added flexibility and granularity in representing the organizational structure and operational dynamics of the Carnival. Sponsors can now have a dedicated team of helpers under their purview, allowing for more effective coordination and delegation of tasks. However, this addition may necessitate adjustments to the concrete syntax and navigation paths within the model to accommodate the new class and its relationships effectively. Nonetheless, the inclusion of helpers enhances the overall functionality and realism of the Carnival meta model, reflecting the complexities of real-world event management scenarios.

Concrete Syntax & Editor

Semiotic Clarity

In our Carnival meta model design, achieving semiotic clarity was important to ensure that the graphical representations effectively conveyed the underlying model constructs. We first established a one-to-one mapping between model elements and their graphical symbols, ensuring each entity, such as sponsors,

events, and helpers, was distinctly and clearly represented. (Refer to Figure 2: Semiotic Clarity for an illustration of this mapping.)

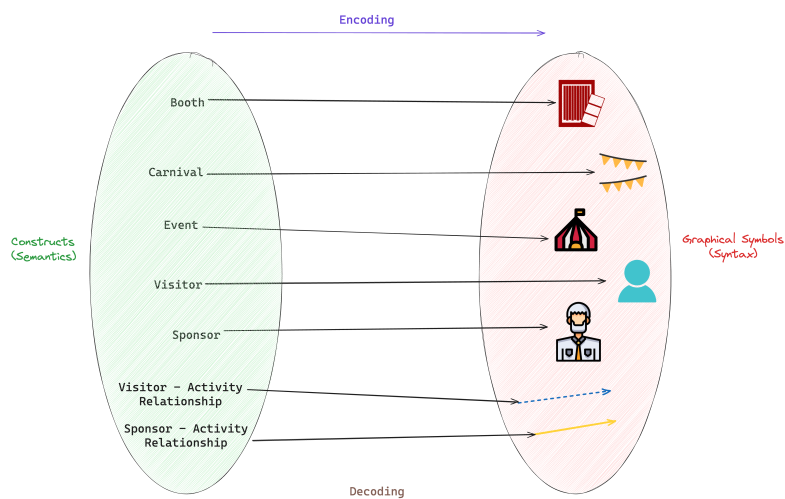


Figure 2: Semiotic Clarity

The GMF Editor

We employed Eugenia, an Eclipse-based tool built on the Graphical Modeling Framework (GMF), to create the Concrete Syntax and Editor The Eugenia generated editor was run as a plugin on Eclipse, using which we created a sample carnival model illustrated in Figure 3 below. The resulting graphical editor provides stakeholders with an intuitive and easily interpretable visual representation of the meta model's structure and relationships.

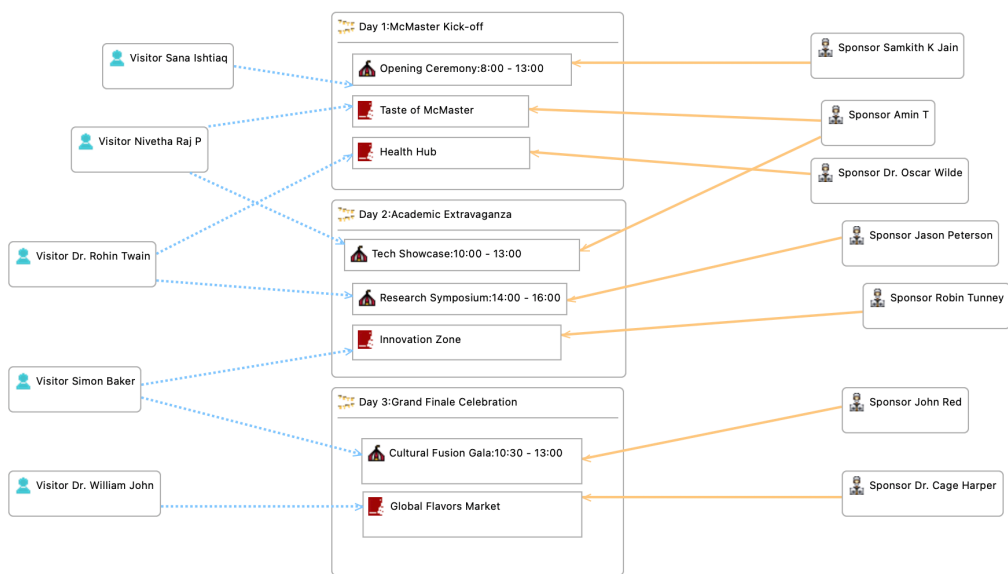


Figure 3: GMF Editor / Model Instance

Advantages

Using GMF (Graphical Modeling Framework) for the Carnival Meta-model offers several advantages. One major advantage is its ability to provide a customizable and visually appealing graphical editor tailored specifically to the domain of the Carnival Meta-model. This enables stakeholders to intuitively create, edit, and visualize the model's elements and relationships, enhancing understanding and collaboration. Additionally, GMF automates much of the underlying infrastructure required for creating graphical editors, saving time and effort in development.

Disadvantages

The alternative design incorporating a set of helpers/staff, while adding functionality, poses challenges in terms of clutter and symbol flooding. Introducing additional elements like helpers/staff may lead to a denser diagram with an increased number of graphical symbols, potentially making it more difficult to discern and interpret the model's structure. This issue can diminish the clarity and usability of the graphical representation, counteracting the benefits of using GMF for visual modeling. Thus, careful consideration and balancing of functionality and clarity are essential when incorporating such complex features into the graphical representation of the Carnival Meta-model using GMF.

Validation

Model to Text Transformation

Software and Dependencies

- **Eclipse Modeling Tools Version - 2023-12 (4.30.0):** [Download](#)
- **Graphical Modeling Framework (GMF) - Tooling 3.2.1:** [Download](#)
- **Epsilon 2.1:** [Download](#)
- **Emfatic 1.0.0:** [Download](#)

References

- **EMF:** [Eclipse Modeling Framework Documentation](#)
- **Ecore:** [Eclipse Modeling Framework Core Tutorial](#)
- **GMF:** [Graphical Modeling Framework Tutorial](#)
- **EVL:** [Epsilon Validation Language Documentation](#)
- **EGX:** [Epsilon Generation Language Documentation](#)
- **OCL:** [Object Constraint Language Documentation](#)
- **EGL:** [Epsilon Generation Language](#)

Contributors

- Amin Toloekashefpakdel (toloekaa@mcmaster.ca)
- Nivetha Raj Pappuraj (pappuran@mcmaster.ca)
- Samkith K Jain (kishors@mcmaster.ca)
- Sana Ishtiaq (ishtis3@mcmaster.ca)