

An Automated Form Building Application Based on Domain-Driven-Design

Zhefei Chen

Department of Computer Science of Nanjing University

November 2, 2020

- 1 What is Domain-Driven-Design
- 2 Apply DDD in our Form Building App
- 3 Conclusion: Inspirations of DDD and Future Work

- 1 What is Domain-Driven-Design
- 2 Apply DDD in our Form Building App
- 3 Conclusion: Inspirations of DDD and Future Work

From Domain Model to Work

The Definition of Domain

A rigorously organized and selective abstraction of knowledge.

The Definition of Domain-Driven-Design

The structure and language of software code (class names, class methods, class variables) should match the business domain.

Examples

If a software processes loan applications, it might have classes such as LoanApplication and Customer, and methods such as AcceptOffer and Withdraw.

The Utility of a Domain Model in Domain Driven Design

- The model and the heart of the design shape each other.
Which means changes of the model will lead to the changes of the design of the software system.

The Utility of a Domain Model in Domain Driven Design

- The model and the heart of the design shape each other.
Which means changes of the model will lead to the changes of the design of the software system.
- The model is the backbone of a language used by all team members.
Using a specialized language tailored for this model can erase the difficulty in the communication between engineers and the project manager(Assume that the project manager knows exactly what the requirements are).

The Utility of a Domain Model in Domain Driven Design

- The model and the heart of the design shape each other.
Which means changes of the model will lead to the changes of the design of the software system.
- The model is the backbone of a language used by all team members.
Using a specialized language tailored for this model can erase the difficulty in the communication between engineers and the project manager(Assume that the project manager knows exactly what the requirements are).
- The model is distilled knowledge.
It reduces the efforts that engineers need to make to communicate with each other because model contains the condensed knowledge we hold about a domain and its components.

- 1 What is Domain-Driven-Design
- 2 Apply DDD in our Form Building App
- 3 Conclusion: Inspirations of DDD and Future Work

The Composition of a Form Building App

- The Basic Knowledge About a Form Building App
- The Domain Model Design & Former Framework
- The New Framework Applied in Development

The Basic Knowledge About a Form Building App

- Main Function: Constructing & Storing Forms
- Special Requirement: Customers may frequently change their ideas, and the form they need has to be modified along with the changes of ideas.

The Domain Model Design & Former Framework

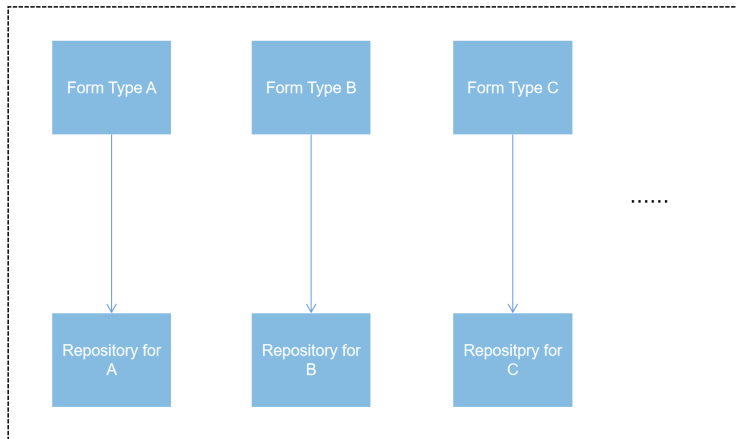
- The Core Knowledge: Forms, Customers and Engineers
- The Domain Model Design: We need to define Classes for Forms, Customers and Engineers, and methods for creating & storing forms and data.

Traditionally we use Java code to define classes, and this method works in most situations. At first we treated the domain model of Form the same as Customer's and engineer's, so we logically define different Form Classes since there could be different kinds of forms facing different requirements.

Former Framework

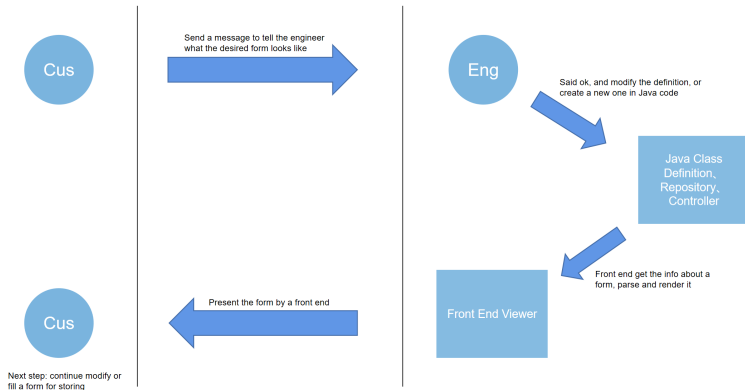
We recognize the domain model of form as classes defined by Java Code.

Domain Model of Form



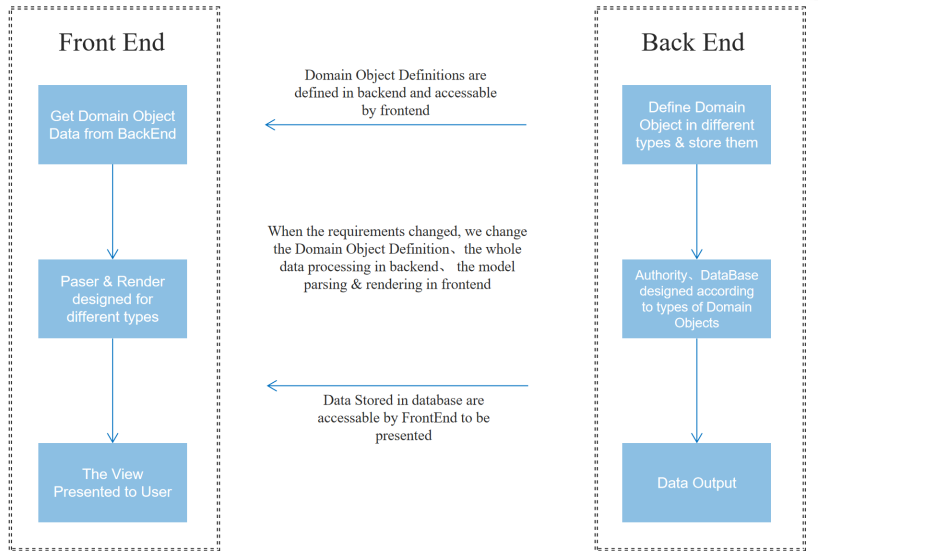
Former Framework

Eventually we think in the following way that engineers passively create or modify the classes defined for forms to satisfy the changing requirement.



Former Framework

The system came as following:



The New Framework Applied in Development

- Problem: How to reduce the cost brought by changes of requirements

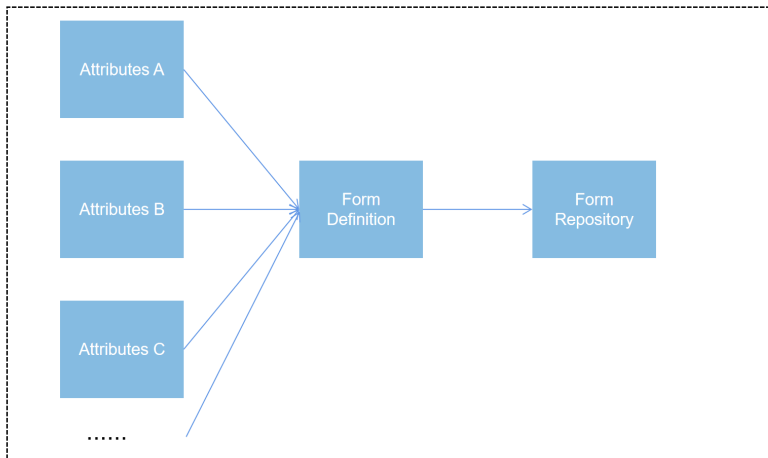
The New Framework Applied in Development

- Problem: How to reduce the cost brought by changes of requirements
- What if we think in a different way when designing the domain model of form?

The New Framework Applied in Development

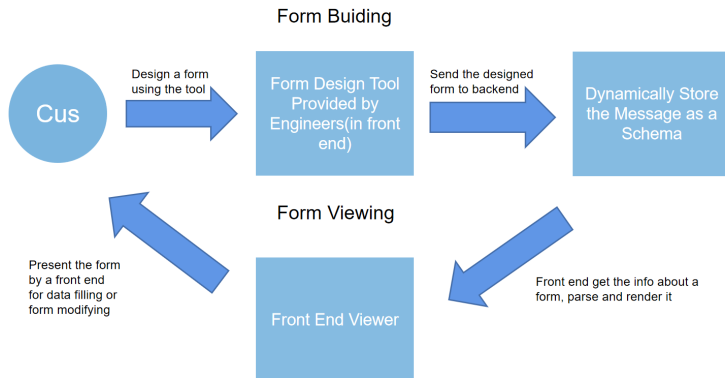
We separate a form definition into different attributes, thus the process of defining a kind of form becoming the combination of attributes.

Domain Model of Form



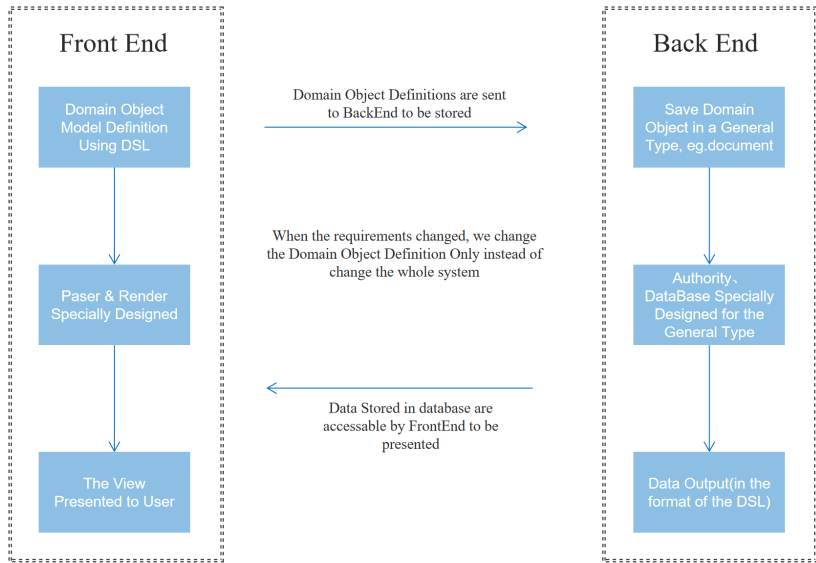
The New Framework Applied in Development

And the whole process becomes that customer constructs a form he needs on his own, engineers are only responsible for storing the form designed correctly.



The New Framework Applied in Development

The new framework appears:



- 1 What is Domain-Driven-Design
- 2 Apply DDD in our Form Building App
- 3 Conclusion: Inspirations of DDD and Future Work

Conclusion: Inspirations of DDD

- Applying Domain-Driven-Design helps us build project in a more systematical way, which explicitly explains what a software is made for and what it consists of.

Conclusion: Inspirations of DDD

- Applying Domain-Driven-Design helps us build project in a more systematical way, which explicitly explains what a software is made for and what it consists of.
- A good design of Domain Model can reduce software engineers' burden when changes of requirements occur.

Conclusion: Inspirations of DDD

- Applying Domain-Driven-Design helps us build project in a more systematical way, which explicitly explains what a software is made for and what it consists of.
- A good design of Domain Model can reduce software engineers' burden when changes of requirements occur.
- We can integrate Domain-Driven-Design with a specified DSL to come up with a specially designed framework to satisfy the complexity of requirements.

Though we improved the performance of the app by change the details of domain model, there are still some problems need to do.

- Using Json Schema as a DSL has its demerits when we need to build a large form with many options. In that situation, the schema text of a form could be to long to be accepted by engineers.

Solution: We are now working to develop a specified DSL to overcome the drawbacks of Json Schema.

Though we improved the performance of the app by change the details of domain model, there are still some problems need to do.

- Using Json Schema as a DSL has its demerits when we need to build a large form with many options. In that situation, the schema text of a form could be to long to be accepted by engineers.
Solution: We are now working to develop a specified DSL to overcome the drawbacks of Json Schema.
- Now we have a front end for form creating and filling, a back end for storing and searching data, but the Authorization part of back end is not finished yet, we still need to work on to complete the whole framework for a DDD Form Building App.

Ideally we will finally present a DDD framework for this kind of Form building App, including not only the part of form constructing and storing, but also a complete back end with authorization and database access.

Thanks For Listening!
Q&A