A Web Application Is a DSL*

David H. Lorenz
Open University of Israel; & Technion, IIT

Boaz Rosenan
University of Haifa

App-DSL Correspondece

DSLs

DSL		Application
Abstract Syntax	⟷	Database Schema
Concrete Syntax + Editor	⟷	User Interface
Semantics	⟷	Business Logic
Program	⟷	**State**

Applications

External DSLs

- Implemented in a PL **external to the DSL**
- Implemented as **compilers / interpreters**
- **Limited reuse** between DSLs
- Mostly **imperative**
- Flexible **syntax / semantics**

State-of-the-Art Web Apps

- Implemented in a PL **external to the application**
- Typically implemented as extensions to a **Web server**
- **Limited reuse** between Apps
- Mostly **imperative**
- Flexible **functionality**

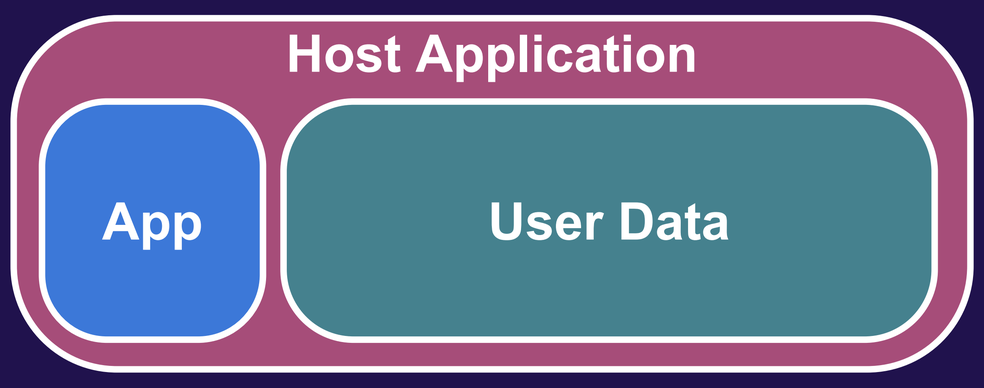
Internal DSLs (DSL Embedding)

- Implemented in a **host language**
- Implemented as a **software library / module**
- **Reuses the implementation** of the host language
- Often **declarative**
- Syntax / semantics **limited** by the host language

Application Embedding

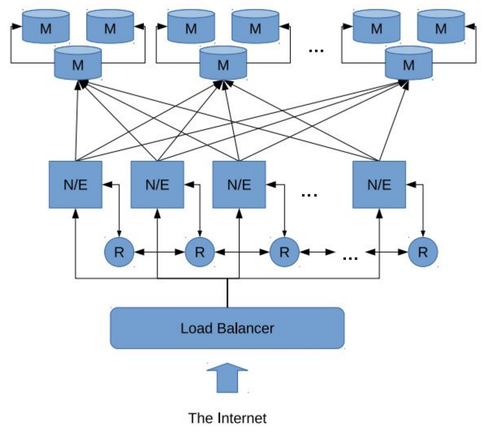
New!

- Implemented in a host application
- Implemented as a data in the database
- **Reuses the implementation** of the host application
- Often **declarative**
- Functionality (potentially) **limited** by the host language



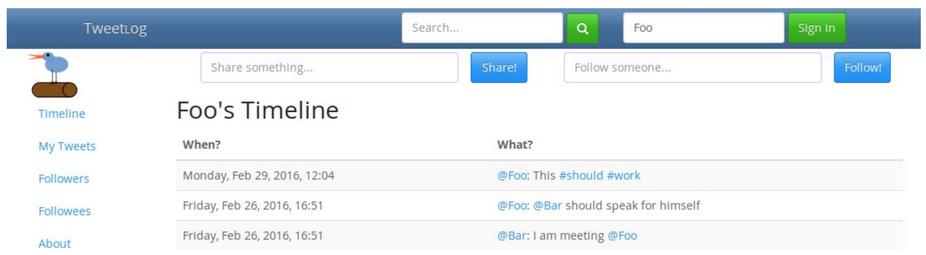
Proof-of-Concept: FishTank

We implemented a proof-of-concept implementation, named **FishTank**.
Implemented using a typical application stack: MongoDB, Node.JS & Express.JS, Angular.JS



Case-Study Application: Tweetlog

A **Twitter-like** application built on top of **FishTank**



Business-logic implemented as **declarative rules**.

1. axiom *U* tweeted *T* at *τ* { replaceText (*T* , raw ***Text*** , tokenized (***X***) , ***T*** ′) , tokens (*X* , tweetCtx) parses *Text* } → procTweet (*U* , *τ* , *T*)
2. axiom procTweet (***User*** , ***Time*** , ***Tweet***) { replaceText (*Tweet* , tokenized (***Tokens***) , _ , _) , ***Token*** :: < tweetCtx > ∈ *Tokens* }
   → searchIndex (*Token* , *Time* , *User* , *Tweet*)
3. axiom procTweet (@ ***A*** , *Time* , *Tweet*) → searchIndex (@ *A* , *Time* , @ *A* , *Tweet*)
4. axiom ***U*** ₁ follows @ ***U*** ₂ since _ →
   searchIndex (@ *U* ₂ , *Time* , ***U*** ₃ , *Tweet*) →
   timeline (*U* ₁ , *Time* , tweet (*U* ₃ , *Tweet*)) :- ⊤

Conclusion

The **App-DSL correspondence** illuminates the fact that **application embedding** is unexplored territory.
We found application embedding as a way to improve reuse and ease the development of web applications.

Website:

*This research was supported in part by the *Israel Science Foundation (ISF)* under grant No. 1440/14.