LionWeb: Status Update

LangDev 2024

Jos Warmer, Niko Stotz

TY45

ProxyHands

Recap

LANGUAGE INTERFACES ON THE WEB

Mission

To create an ecosystem of interoperable components for building language-oriented modeling tools on the web.

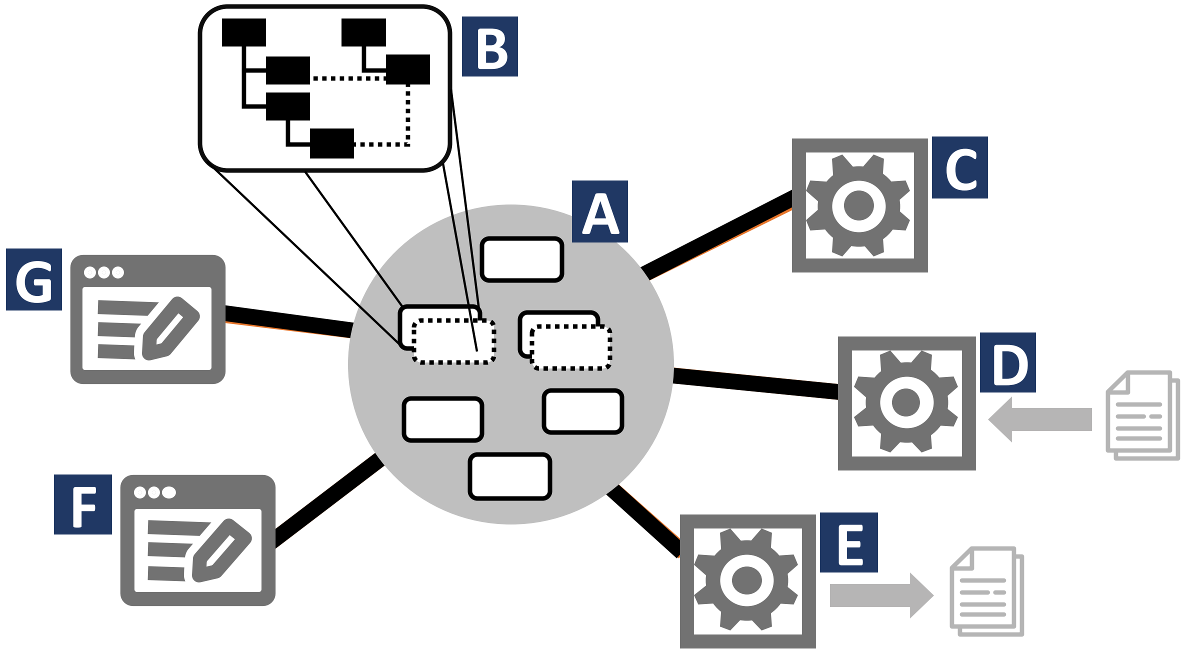
Adopt language engineering

Reduce vendor lock-in

Mix and match components

Foster innovation and incremental improvements

Reference Architecture



Clients

Editors

- F Visual
- G Projectional ✓

Processors

- C Model checker
- D Importer ✓
- E Generator ✓

Repository

Models

- A Original ✓
- B Derived ✓

Protocols (bulk ✓ / delta ✓)

Contents (JSON) ✓

Further info

Introduction

- Talk at LangDev 2023 (slides, video)

Working documents

- Roadmap
- Reference architecture
- Use cases

Usage

LANGUAGE INTERFACES ON THE WEB

Talks and Presentations

Conferences

- LangDev 2023
- SEN Symposium 2024
- MODELS 2024

Meetups

- MPS Community Meetup 2023
- Strumenta Community Panel Session

Universities

- Essen
- Aachen

Real-world Usage

Trumpf

Customers use different software to work with their laser tools. The next generation of software uses LionWeb to assure data consistency, reducing user friction.

Strumenta Code Insight Studio

Presentation yesterday "*Language Engineering for Language Migrations*". Tools for migrating from one programming language to another. LionWeb enabled development of a multi-platform solution for operating on models.

Triradial

A product to assist the staff at clinical trial sites consisting of two parts:

- Freon-based web DSL, using LionWeb for storage, to define, test and generate visualization of the schedule and the code and configuration for the runtime.
- A custom web-based runtime system, guiding them through the work of each visit and study.

Interested in Integration

Jetbrains WebMPS

Metadev Daga

ModelingValueGroup Dclare

TU Delft Spoofax

TypeFox Langium

- https://www.typefox.io/blog/lionweb-langium/

TypeFox Typir

Other talks about LionWeb @ LangDev 24

Title	Speaker	Affiliation
Language Engineering for Language Migrations Thursday 18:00 Track 1: Salon de Grados	Federico Tomassetti	Strumenta
GenFPL: DSL-embeddable functional programming languages Friday 10:00 Track 1: Salon de Grados	Meinte Boersma	DSL Consultancy
LionWeb and Kolasu: an integration story Friday 11:30 Track 1: Salon de Grados	Alessio Stalla	Strumenta
Implementing LionWeb in Rascal Friday 12:00 Track 1: Salon de Grados	Ulyana Tikhonova	FIRE
A Case Study: Execution of LionWeb nodes in Truffle Language Framework Friday 12:30 Track 1: Salon de Grados	Erkan Diken	FIRE

Spec Work

LANGUAGE INTERFACES ON THE WEB

How We Work

Specification documents (public)

- hosted on GitHub: github.com/LionWeb-io/specification
- published at lionweb.io/specification
- refers to issues for details

Discussion issues (public)

- same GitHub repo: 125 open / 129 closed
- keep them focused
- document alternatives and rationale

Weekly core team meetings

- keep discussions running (more than 2 years already!)
- "rough consensus and running code"
- meeting logs and video recordings

Implementation mostly by team members (public)

- partially in paid projects
- guided by real-world needs

Finished

Released as LionWeb 2023.1

- JSON serialization format
- meta-metamodel (M3)

Scheduled for LionWeb 2024.1

- bulk protocol
- added `StructuredDataType`
- removed `JSON` PrimitiveType
- several test suites

In Progress

Working draft

- delta protocol (issues)
- formalizing model correctness incl. language (PR)

Under discussion

- derived models (issues)

Bulk API Semantics

LANGUAGE INTERFACES ON THE WEB

Basic Commands

List/create/delete partitions

- straightforward CR_D

Retrieve

- get me subtree of node X with depth Y

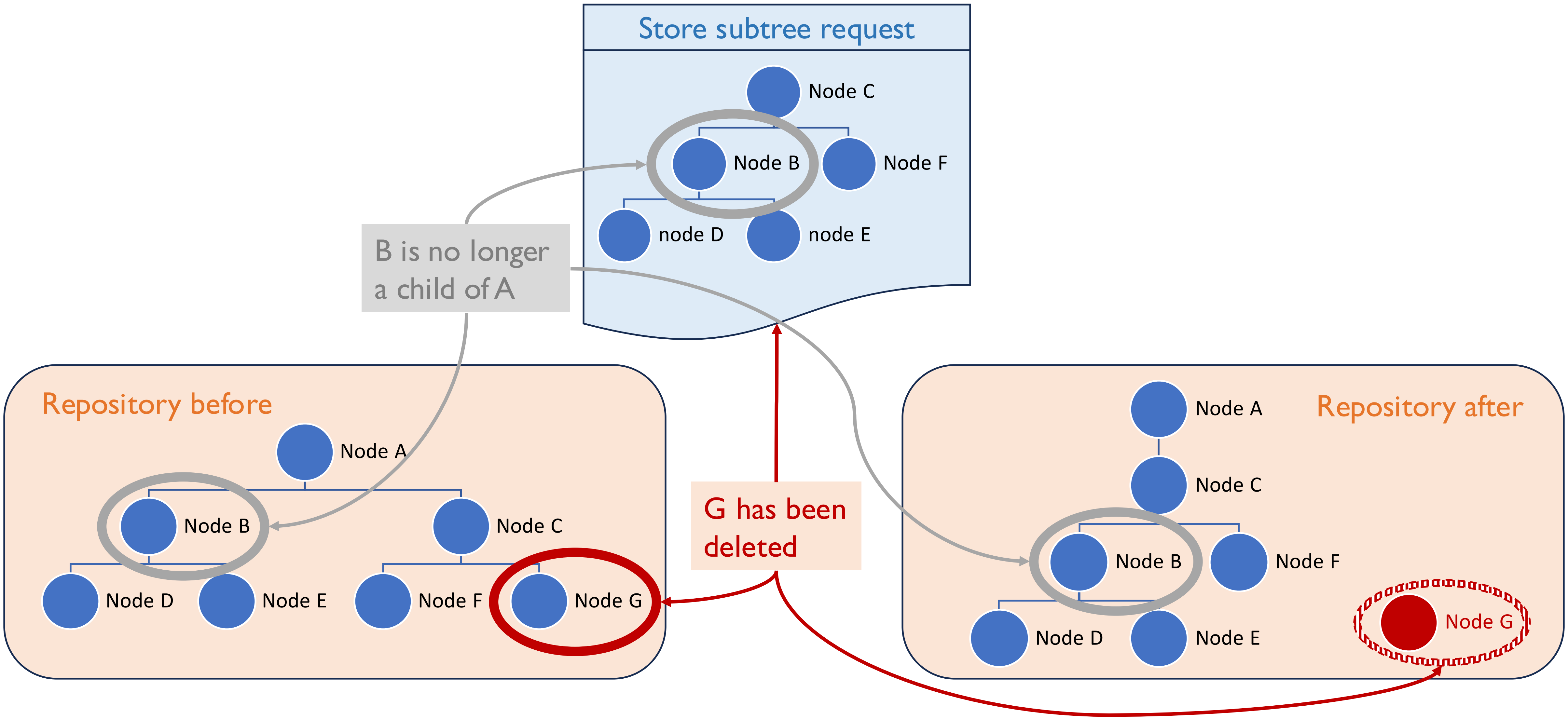
Store

- make the repository represent the subtree(s) I send
- guarantees repository contents are proper tree:
  - each node has exactly one parent (except for partition nodes)
  - unique node ids

Ids

- give me guaranteed free node ids

Intricacies of Store



Usable Code

LANGUAGE INTERFACES ON THE WEB

Repository App

Implements bulk API

Guarantees repository contents are proper tree

- language agnostic

Supports multiple repositories

With or without history

Optimized methods for uploading huge models

- millions of nodes

https://github.com/LionWeb-io/lionweb-repository

Repository App: Implementation

Implementation

- PostgreSQL backend
- written in TypeScript
- reuses LionWeb TypeScript implementation

Provides TypeScript repository client

No authentication

- out of scope

Distributions

- Docker image
- source code

Freon LionWeb Core Editor App

M3 in Freon

Web editor to edit LionWeb languages

- load/store JSON

https://github.com/LionWeb-io/lionweb-freon-m3

C# Library

Streaming, memory-efficient M1/M2 (de)serializer with custom error handling

Low-level serialization API, high-level semantic-enforcing node API

Annotations, dynamic & lenient nodes

Roslyn-based generator LionWeb M2 → C# types

Node and language navigation utils

Advanced utilities

- cloner
- comparer
- textualizer
- reference finder

Extensive test coverage

https://github.com/LionWeb-io/lionweb-csharp

Java Library

Memory-efficient M1/M2 (de)serializer with pluggable workers

- additional binary wire formats

Low-level serialization API, high-level node API

Annotations, dynamic & proxy nodes

Node and language navigation utils

Advanced utilities

- model comparator
- validator

EMF M1/M2 Importer + Exporter

https://github.com/LionWeb-io/lionweb-java

Kotlin Library

Idiomatic Kotlin adaptions of Java APIs

Generator LionWeb M2 → Kotlin types (together with StarLasu)

Repository client

- bulk API
- huge model upload API
- compression

https://github.com/LionWeb-io/lionweb-kotlin

TypeScript Library

M1/M2 (de)serializer

Low-level serialization API, high-level semantic-enforcing node API

Annotations, dynamic nodes

Generator LionWeb M2 → TypeScript types

Command line utilities

- validator
- differ
- textualizer
- diagram generator for languages (PlantUML/Mermaid)
- measure
- repair
- sort
- infer language from JSON instance

https://github.com/LionWeb-io/lionweb-typescript

MPS Plugin

M1 (de)serializer, M2 serializer

Annotations, custom extensions for node/concept descriptions

MPS language attributes for LionWeb infos

Command line interface to serialize M2

Experimental

- M2 deserializer
- bulk-like repository API
- bulk-like repository client

https://github.com/LionWeb-io/lionweb-mps

Modelix Plugin

Bulk repository API

- partial support for references

https://github.com/LionWeb-io/lionweb-modelix-adapter

Rascal Plugin

M1 (de)serializer, M2 deserializer

Annotations, generalizations, references

Custom-made pointers and lionspace for LionCore languages and nodes

Rascal terminal to invoke transformations

https://github.com/UlyanaTikhonova/lionweb-rascal

TY45

Team and Contact

Team

Meinte Boersma	Freelancer	**Alex Shatalin**	Jetbrains (MPS)
Norman Koester	itemis (Modelix)	**Niko Stotz**	FIRE
Sergej Koscejev	Freelancer	**Federico Tomassetti**	Strumenta
Sascha Lisson	itemis (Modelix)	**Markus Voelter**	Freelancer
Eugen Schindler	Canon Production Printing	**Jos Warmer**	Freelancer (Freon)

Contact

Want to work with us:
info@lionweb.io

GitHub:
https://github.com/LionWeb-io

Slack to follow us, use, or implement LionWeb:
https://join.slack.com/t/lionweb/shared_invite/zt-2k1k5bsv7-EvC2IABIgSxWrqJNVOZ0HQ