

ATELIER IHM 2023

Fonctions d'utilisabilité dans les studios de
conception de langages dédiés graphiques

Introduction - Qui suis-je ?



Théo GIRAUDET

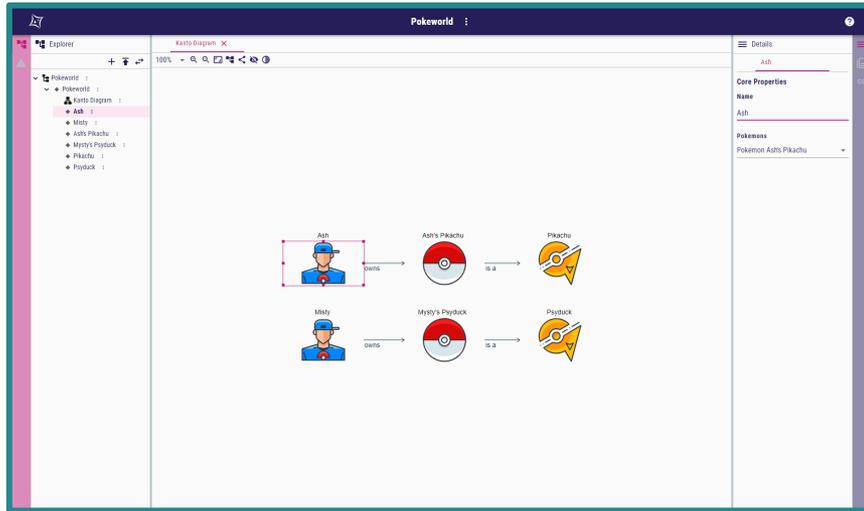
- Doctorant au sein de l'équipe DiverSE à l'IRISA de Rennes
- Thèse réalisée avec l'entreprise Obeo (contrat CIFRE)
- Contributeur de Eclipse Sirius Web
- theo.giraudet@obeo.fr
theo.giraudet@irisa.fr

Introduction - L'entreprise



- PME fondée en 2005
- Spécialisée en ingénierie dirigée par les modèles
- Membre stratégique de la fondation Eclipse
- Développe des produits Open Source
- Vend des surcouches des produits et des services liés à ceux-ci

Sirius Web



Utilisation d'un langage dans Sirius Web par l'utilisateur



Outil pour concevoir des ateliers de modélisation

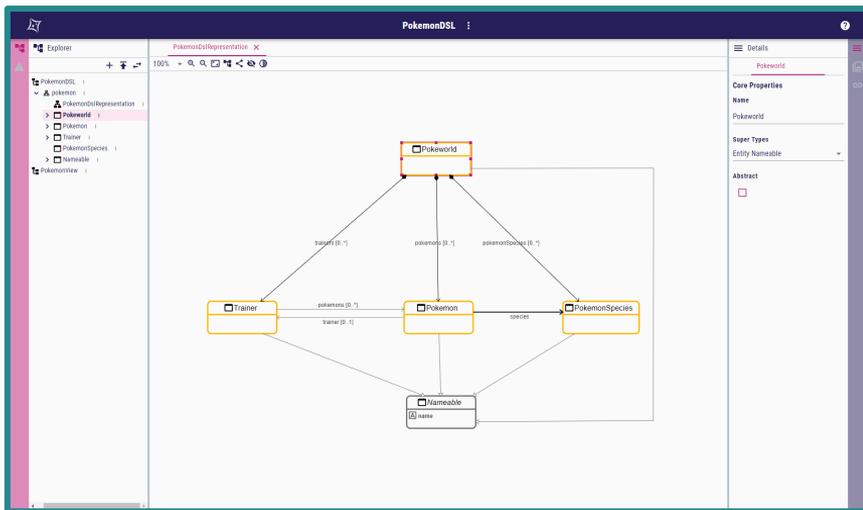


Permet de créer des langages, des représentations pour ceux-ci ainsi que des modèles fondés sur ces langage

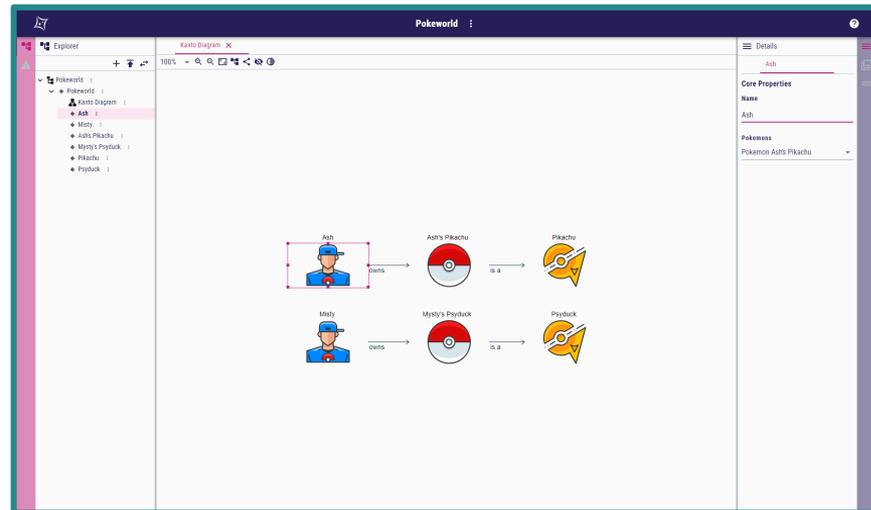


Application Web et collaborative

Sirius Web



Définition du domaine d'un langage (syntaxe abstraite)
dans Sirius Web par le **spécifieur**



Utilisation d'un langage dans Sirius Web par l'**utilisateur**

Explorer

PokemonDslRepresentation

Details

Pokeworld

Core Properties

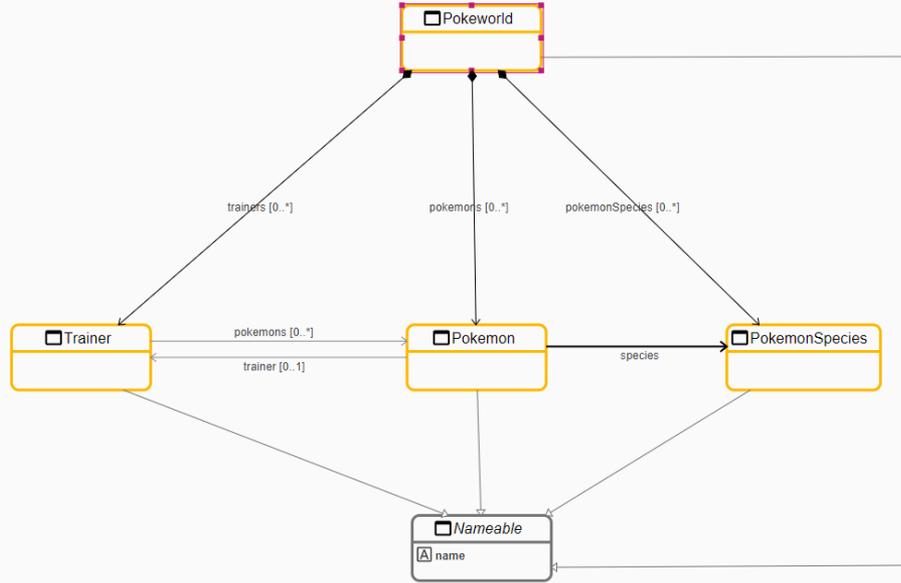
Name

Pokeworld

Super Types

Entity Nameable

Abstract



Explorer

- PokemonDSL
- PokemonView
 - View
 - PokemonDslDiagram
 - DiagramPalette
 - TrainerNode**
 - NodePalette
 - FreeFormLayoutStrategyDescription
 - #E5F5F8
 - PokemonNode
 - PokemonSpeciesNode
 - pokemonSpeciesEdge
 - pokemonsEdge

Create a new Model

Select the model to create

- Flow
- Domain
- View
- Papaya Domain
- Papaya View
- Robot Flow
- Big Guy Flow (17k elements)
- Others...

Create a new Representation

Select the representation to create on TrainerNode

- New Representation
- New Representation
- New Representation

Open an existing Representation

Select the representation to open

- PokemonDslRepresentation

Details

TrainerNode

Core Properties

Name
TrainerNode

Domain Type
pokemon::Trainer

Semantic Candidates Expression
aql:self.eContents()

Precondition Expression
Precondition Expression

Synchronization Policy
 SYNCHRONIZED
 UNSYNCHRONIZED

Label Expression
aql:self.name

Collapsible

Reused Child Node Descriptions

Reused Border Node Descriptions

User Resizable

Explorer

Kanto Diagram

100%

Details

Ash

Core Properties

Name

Ash

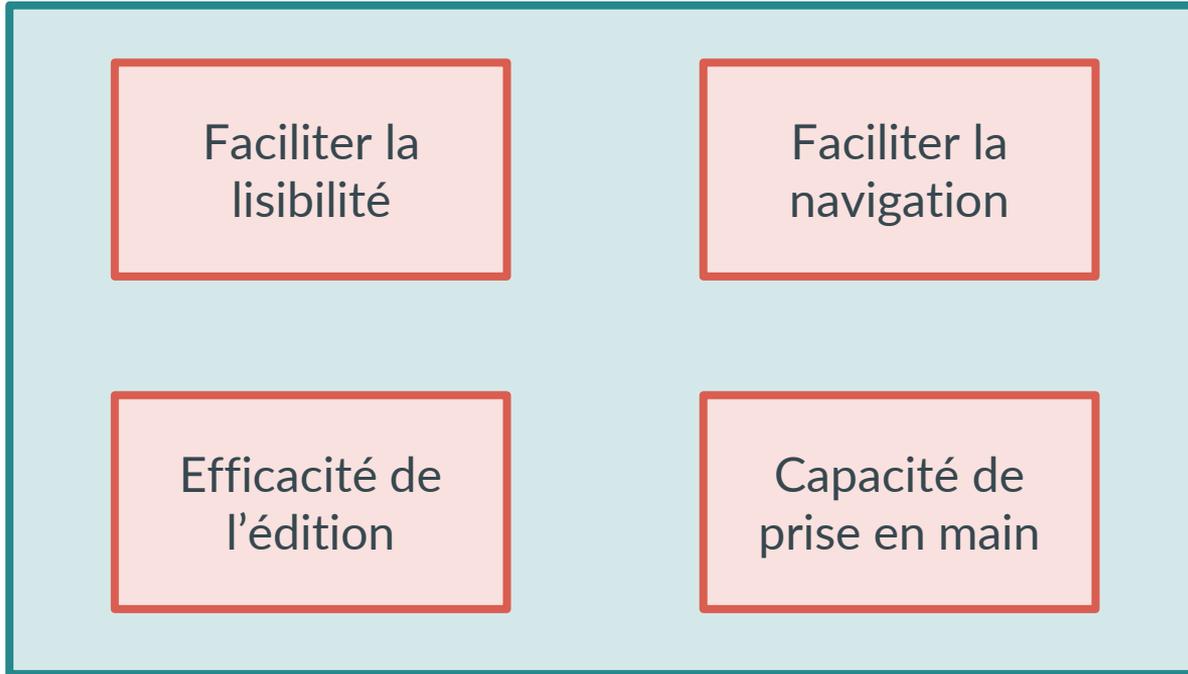
Pokemons

Pokemon Ash's Pikachu

- Pokeworld
 - Kanto Diagram
 - Ash
 - Misty
 - Ash's Pikachu
 - Misty's Psyduck
 - Pikachu
 - Psyduck

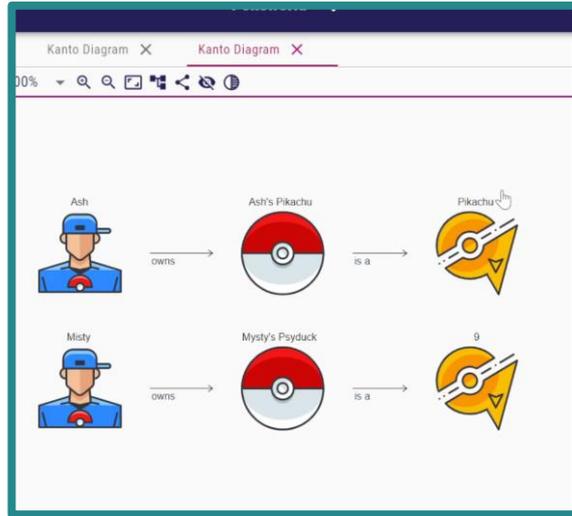


Fonctions d'utilisabilité

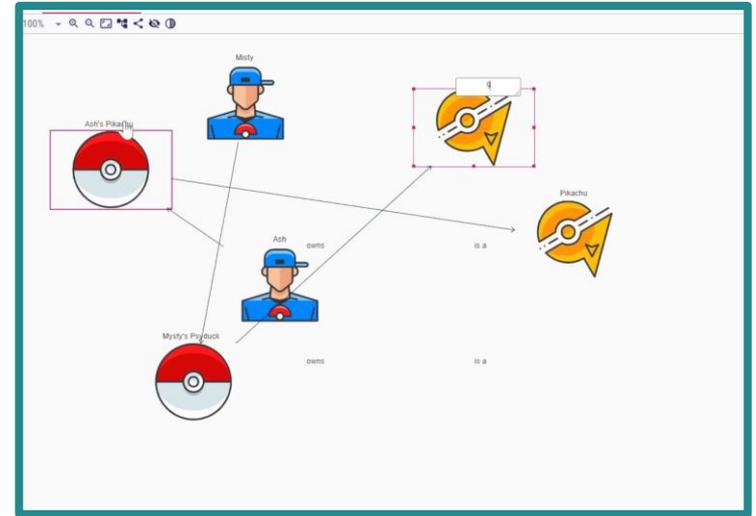


Classes de fonctions d'utilisabilité pour une représentation diagramme (*non exhaustif*)

Fonctions d'utilisabilité - Faciliter la lisibilité

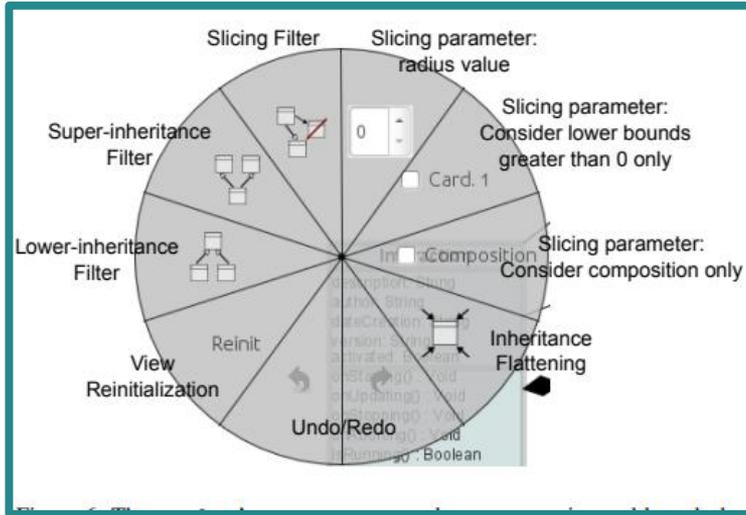


Disposition libre

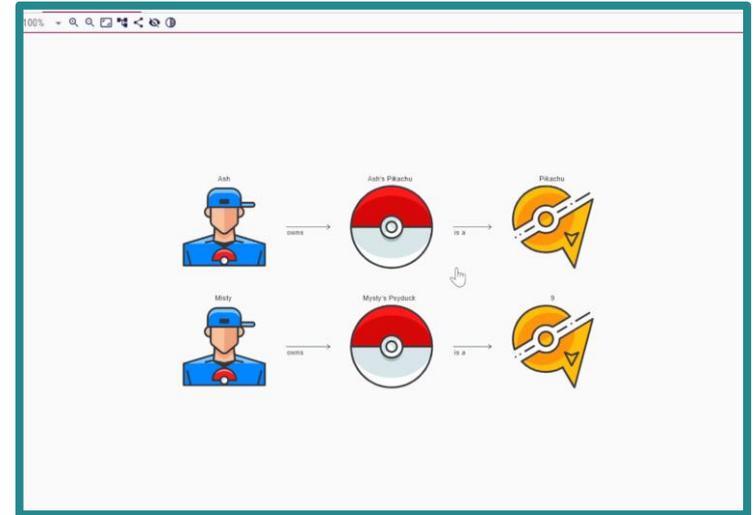


Disposition automatique

Fonctions d'utilisabilité - Faciliter la lisibilité

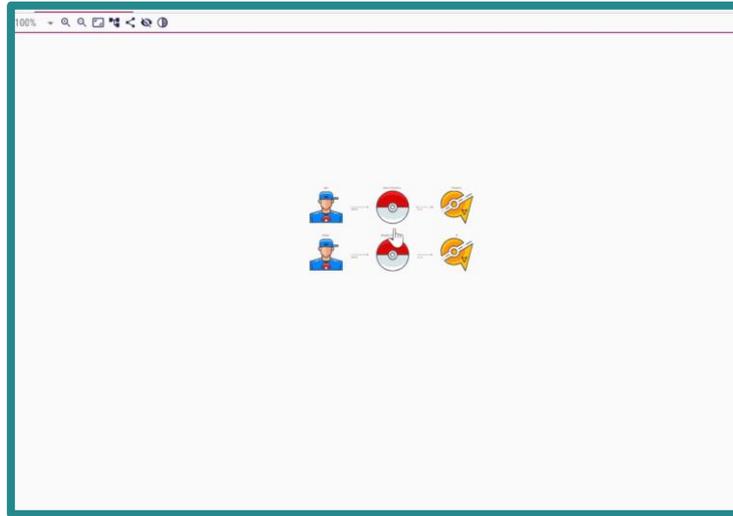


Zoom sémantique [1]

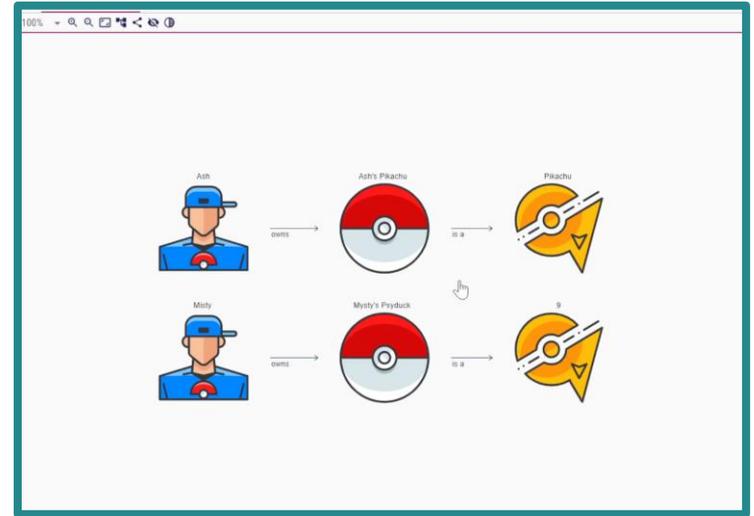


Cacher/estomper des éléments

Fonctions d'utilisabilité - Faciliter la navigation

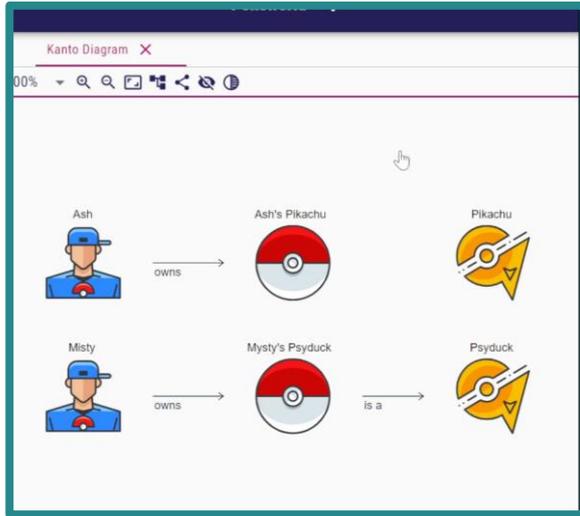


Déplacements dans le diagramme

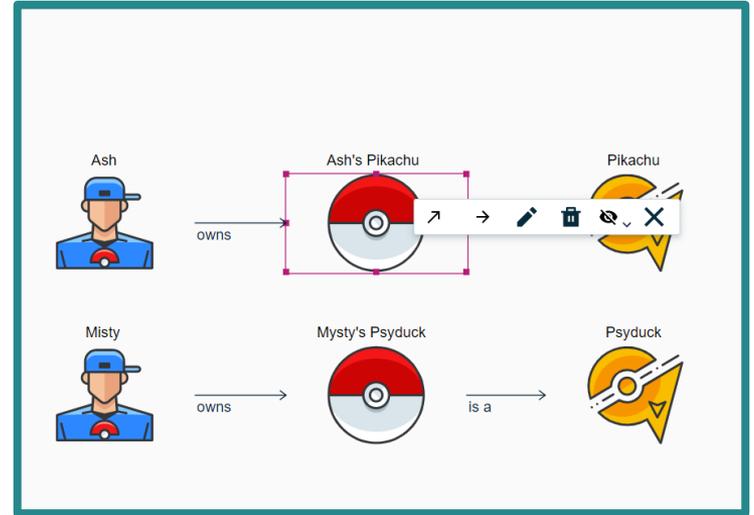


Feedback lors de l'interaction avec un élément

Fonctions d'utilisabilité - Efficacité de l'édition

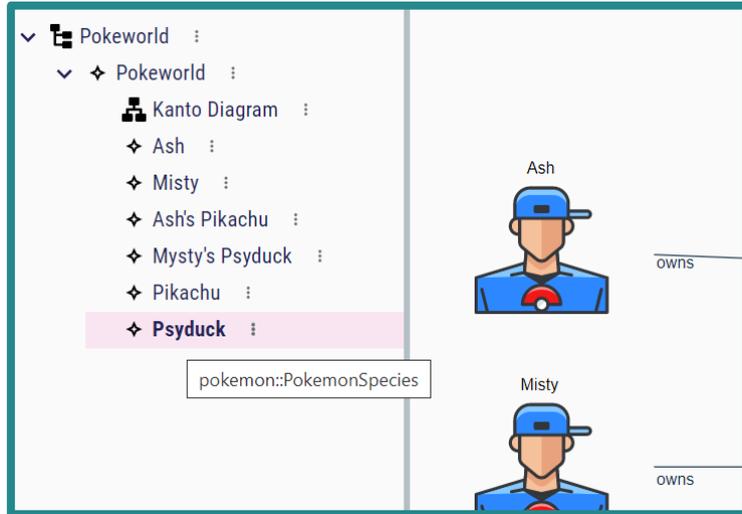


Connecteur magique



Palette d'outils

Fonctions d'utilisabilité - Capacité de prise en main



Tooltips

Name:

Instance Class Name:

Instance Type Name:

Abstract

Interface

Bouton d'aide

Défis

1 Toolkits incomplets

- Peu de bibliothèques pour faire des diagrammes
- Celles existantes se limitent à de simples opérations
- Très peu de supports de ces fonctions d'utilisabilité

2 Développer des fonctions d'utilisabilité génériques et modulaires

- Fonctions agnostiques du domaine modélisé
- Fonctions qui doivent pouvoir être adaptées pour le domaine

Défis

- 3 Trade-off entre traitements côté serveur et besoins clients pour faire de l'utilisabilité
 - La quasi-totalité des traitements sont fait sur le serveur
 - Besoins d'infos riches côté client pour certaines fonctions d'utilisabilité

Références

[1] Arnaud Blouin, Naouel Moha, Benoit Baudry, Houari Sahraoui, and Jean-Marc Jézéquel. 2015. Assessing the Use of Slicing-based Visualizing Techniques on the Understanding of Large Metamodels. *Information and Software Technology* 62, 0(2015), 124 – 142.
<https://doi.org/10.1016/j.infsof.2015.02.007>

Merci de votre attention !

N'hésitez pas à poser vos questions !