

2.5.2. Najít v repozitory browseru

Pokud si uživatel například vyhledá jakýkoliv artefakt skrz „Vyhledávání“, tak se mu zobrazí jeho detail, ale neuvidí jeho pozici v repozitory tree.

Díky možnosti „Najít v repozitory browseru“ má uživatel možnost, aby se mu rozbalil repozitory tree až k příslušnému artefaktu.

Funkcionalita je dostupná v kontextovém menu pod symbolem stroměčku



The screenshot displays the Dataprojekt application interface. On the left, a sidebar titled "Repozitář" (Repository) shows a tree structure. A blue arrow points to a tree icon in the sidebar's navigation menu. The main area shows a "C# Model" diagram with classes like Account, Transaction, Order, ShoppingBasket, LibItem, and StockItem. The diagram includes attributes, methods, and relationships between these classes. At the bottom, there is a table with columns: Název (Name), Typ (Type), Stereotyp (Stereotype), and Verze (Version). The table shows the C# Model as a Logical type, version 1.0. The footer contains the copyright information: "Dataprojekt s.r.o. © 2020 - 7.3.0.6 - Licencováno pro: Dataprojekt, a.s. (unlimited)"

Po kliknutí na ikonku „Najít v repozitory browseru“ se rozbalí repozitory tree až na pozici dotyčného artefaktu.

EaExample 1 Vyhledávání 1 Repozitář 1 Publikace Správa repozitáře 1 Správa portálu 1 Bedřich Mravenec (bmravenec) Odhlášení

Project Management
Testing
Maintenance
Publishing
Reporting
Database Modeling
Schema Modeling
Geospatial Modeling
Systems Engineering
Visual Execution Analysis
Software Engineering
Implementation Model - With Behaviors
C# Model
C# Model - No Attributes
Account
LinItem
Order
ShoppingBasket
StockItem
Transaction
OrderStatus
EJB Entity
EJB Session
Java Model
Java Model With Behaviors
User Interface
Component Model
Deployment Model
Environment Model
Model Transformation
Model Simulation
Pro Cloud Server

C# Model

This Class diagram represents the MDA transform generated from the Abstract Class model PIM.

For more information on MDA transforms see:

MDA Transforms

This Class model can be forward engineered to produce C# code.

For more information on processes of code generation, reverse engineering of source code and synchronization between the source code and model - see:

Code Engineering

```
classDiagram
    class Account {
        - billingAddress: string
        - closed: bool
        - deliveryAddress: string
        - emailAddress: string
        - name: string
        + createNewAccount(): void
        + loadAccountDetails(): void
        + markAccountClosed(): void
        + retrieveAccountDetails(): void
        + submitNewAccountDetails(): void
        + validateUser(string, string)
    }
    class ShoppingBasket {
        - shoppingBasketNumber: string
        + addItem(): void
        + createNewBasket(): void
        + deleteItem(): void
        + processOrder(): void
        + property
        + LinItem() LinItem
    }
    class Order {
        - date: Date
        - deliveryInstructions: string
        - orderNumber: string
        + checkOutStanding(Order): void
        + property
        + date() Date
        + deliveryInstructions() string
        + LinItem() LinItem
        + orderNumber() string
        + status() OrderStatus
    }
    class Transaction {
        - date: Date
        - orderNumber: string
        + loadAccountHistory(): void
        + loadOpenOrders(): void
        + property
        + account() Account
        + date() Date
        + LinItem() LinItem
        + orderNumber() string
    }
    class LinItem {
        - quantity: int
        + property
        + item() StockItem
        + quantity() int
    }
    class StockItem {
        - Author: string
        - catalogNumber: string
        - author: number
        - author: number
        - author: string
        + property
        + Author() string
        + catalogNumber() string
        + author() number
        + author() number
        + title() string
    }
    Account "1" -- "1" Order : account
    Account "1" -- "1" Transaction : -account
    Order "1" -- "1" LinItem : -item
    LinItem "1" -- "1" StockItem : -item
    ShoppingBasket "1" -- "1" LinItem : -item
```

Název	Typ	Stereotyp	Verze
C# Model	Logical		1.0

Dataprojekt s.r.o. © 2020 - 7.3.0.6 - Licencováno pro: Dataprojekt, a.s. (unlimited)

Repozitory tree se nám úspěšně rozbalil a našel hledaný artefakt.

Revision #2

Created 30 May 2022 11:42:50

Updated 15 January 2024 10:17:52