Dokumentacja

Samodzielny   
interfejs graficzny  
SSDL SiG

Dorota Kawalec, Błażej Wolańczyk, Jacek Kowalczyk, Kamil Kopryk, Mateusz Brząkała

Spis treści

[Model danych SSDL 2](#_Toc342909830)

[Dane grafu 2](#_Toc342909831)

[Struktura 2](#_Toc342909832)

[Opis komponentów 2](#_Toc342909833)

[Wierzchołek 4](#_Toc342909834)

[Struktura 4](#_Toc342909835)

[Opis komponentów 4](#_Toc342909836)

[Konfiguracje 6](#_Toc342909837)

[Załorzenia 6](#_Toc342909838)

[Uruchamianie wybranej konfiguracji 6](#_Toc342909839)

[Zawartość pliku konfiguracyjnego 6](#_Toc342909840)

[Edycja i dodawanie modułów 7](#_Toc342909841)

[Moduły 8](#_Toc342909842)

[Event manager 8](#_Toc342909843)

[Opis 8](#_Toc342909844)

[Model danych 8](#_Toc342909845)

[Sposób działania modułu 8](#_Toc342909846)

[Arrow 9](#_Toc342909847)

[Opis 9](#_Toc342909848)

[Model danych 9](#_Toc342909849)

[Sposób działania modułu 9](#_Toc342909850)

[Atomic 11](#_Toc342909851)

[Opis 11](#_Toc342909852)

[Model danych 11](#_Toc342909853)

[Sposób działania modułu 11](#_Toc342909854)

[Ustawienia 11](#_Toc342909855)

[History 12](#_Toc342909856)

[Opis 12](#_Toc342909857)

[Model danych 12](#_Toc342909858)

[Sposób działania modułu 12](#_Toc342909859)

[Zdarzenia 14](#_Toc342909860)

[Ustawienia 14](#_Toc342909861)

[contextmenu 14](#_Toc342909862)

[Opis 14](#_Toc342909863)

[Model danych 14](#_Toc342909864)

[Sposób działania modułu 15](#_Toc342909865)

[Zdarzenia 15](#_Toc342909866)

[Deployer 15](#_Toc342909867)

[Opis 15](#_Toc342909868)

[Model danych 15](#_Toc342909869)

[Sposób działania modułu 15](#_Toc342909870)

[Zdarzenia 16](#_Toc342909871)

[Ustawienia 16](#_Toc342909872)

[externalInterface 16](#_Toc342909873)

[Opis 16](#_Toc342909874)

[Sposób działania modułu 16](#_Toc342909875)

[Zdarzenia 16](#_Toc342909876)

[logger 17](#_Toc342909877)

[Opis 17](#_Toc342909878)

[Model danych 17](#_Toc342909879)

[Sposób działania modułu 17](#_Toc342909880)

[Ustawienia 17](#_Toc342909881)

[shortcut 17](#_Toc342909882)

[Opis 17](#_Toc342909883)

[Model danych 17](#_Toc342909884)

[Sposób działania modułu 18](#_Toc342909885)

[Zdarzenia 18](#_Toc342909886)

[zdefiniowane skróty klawiszowe 18](#_Toc342909887)

[specialEvents 18](#_Toc342909888)

[Opis 18](#_Toc342909889)

[Model danych 18](#_Toc342909890)

[Sposób działania modułu 18](#_Toc342909891)

[Zdarzenia 19](#_Toc342909892)

[subgraphNavigator 19](#_Toc342909893)

[Opis 19](#_Toc342909894)

[Model danych 19](#_Toc342909895)

[Sposób działania modułu 19](#_Toc342909896)

[Zdarzenia 19](#_Toc342909897)

[ViewResizer 19](#_Toc342909898)

[Opis 19](#_Toc342909899)

[Model danych 20](#_Toc342909900)

[Sposób działania modułu 20](#_Toc342909901)

[Zdarzenia 20](#_Toc342909902)

[Zdarzenia 21](#_Toc342909903)

[Konwencja 21](#_Toc342909904)

[addBlankNode 22](#_Toc342909905)

[Opis 22](#_Toc342909906)

[Powiązane moduły 22](#_Toc342909907)

[graf przepływu 22](#_Toc342909908)

[atomicParseError 22](#_Toc342909909)

[Opis 22](#_Toc342909910)

[Powiązane moduły 22](#_Toc342909911)

[graf przepływu 22](#_Toc342909912)

[atomicparserperformed 23](#_Toc342909913)

[Opis 23](#_Toc342909914)

[Powiązane moduły 23](#_Toc342909915)

[graf przepływu 23](#_Toc342909916)

[deploySSDL 23](#_Toc342909917)

[Opis 23](#_Toc342909918)

[Powiązane moduły 23](#_Toc342909919)

[graf przepływu 23](#_Toc342909920)

[engineLoaded 24](#_Toc342909921)

[Opis 24](#_Toc342909922)

[Powiązane moduły 24](#_Toc342909923)

[graf przepływu 24](#_Toc342909924)

[enlargeGraph 24](#_Toc342909925)

[Opis 24](#_Toc342909926)

[Powiązane moduły 24](#_Toc342909927)

[graf przepływu 24](#_Toc342909928)

[formSubmitted 25](#_Toc342909929)

[Opis 25](#_Toc342909930)

[Powiązane moduły 25](#_Toc342909931)

[graf przepływu 25](#_Toc342909932)

[historyChanged 25](#_Toc342909933)

[Opis 25](#_Toc342909934)

[Powiązane moduły 25](#_Toc342909935)

[graf przepływu 26](#_Toc342909936)

[historyDeleteBlanknode 26](#_Toc342909937)

[Opis 26](#_Toc342909938)

[Powiązane moduły 26](#_Toc342909939)

[graf przepływu 26](#_Toc342909940)

[initView 27](#_Toc342909941)

[Opis 27](#_Toc342909942)

[Powiązane moduły 27](#_Toc342909943)

[graf przepływu 27](#_Toc342909944)

[onEscape 27](#_Toc342909945)

[Opis 28](#_Toc342909946)

[Powiązane moduły 28](#_Toc342909947)

[graf przepływu 28](#_Toc342909948)

[openForm 28](#_Toc342909949)

[Opis 28](#_Toc342909950)

[Powiązane moduły 28](#_Toc342909951)

[graf przepływu 28](#_Toc342909952)

[SSDLloaded 29](#_Toc342909953)

[Opis 29](#_Toc342909954)

[Powiązane moduły 29](#_Toc342909955)

[graf przepływu 29](#_Toc342909956)

[ssdlParsed 29](#_Toc342909957)

[Opis 29](#_Toc342909958)

[Powiązane moduły 29](#_Toc342909959)

[graf przepływu 30](#_Toc342909960)

[parserGotData 30](#_Toc342909961)

[Opis 30](#_Toc342909962)

[Powiązane moduły 30](#_Toc342909963)

[graf przepływu 30](#_Toc342909964)

[parserPassed 30](#_Toc342909965)

[Opis 30](#_Toc342909966)

[Powiązane moduły 31](#_Toc342909967)

[graf przepływu 31](#_Toc342909968)

[parsertranserror 31](#_Toc342909969)

[Opis 31](#_Toc342909970)

[Powiązane moduły 31](#_Toc342909971)

[graf przepływu 31](#_Toc342909972)

[redeployssdl 31](#_Toc342909973)

[Opis 31](#_Toc342909974)

[Powiązane moduły 32](#_Toc342909975)

[graf przepływu 32](#_Toc342909976)

[reducegraph 32](#_Toc342909977)

[Opis 32](#_Toc342909978)

[Powiązane moduły 32](#_Toc342909979)

[graf przepływu 32](#_Toc342909980)

[selectnone 33](#_Toc342909981)

[Opis 33](#_Toc342909982)

[Powiązane moduły 33](#_Toc342909983)

[graf przepływu 33](#_Toc342909984)

[shortcutBack 33](#_Toc342909985)

[Opis 33](#_Toc342909986)

[Powiązane moduły 33](#_Toc342909987)

[graf przepływu 33](#_Toc342909988)

[shortcutbackall 34](#_Toc342909989)

[Opis 34](#_Toc342909990)

[Powiązane moduły 34](#_Toc342909991)

[Graf przepływu 34](#_Toc342909992)

[shortcutDelete 34](#_Toc342909993)

[Opis 34](#_Toc342909994)

[Powiązane moduły 34](#_Toc342909995)

[graf przepływu 35](#_Toc342909996)

[shortcutEscape 35](#_Toc342909997)

[Opis 35](#_Toc342909998)

[Powiązane moduły 35](#_Toc342909999)

[graf przepływu 35](#_Toc342910000)

[shortcutForward 35](#_Toc342910001)

[Opis 35](#_Toc342910002)

[Powiązane moduły 35](#_Toc342910003)

[graf przepływu 35](#_Toc342910004)

[shortcutForwardAll 36](#_Toc342910005)

[Opis 36](#_Toc342910006)

[Powiązane moduły 36](#_Toc342910007)

[graf przepływu 36](#_Toc342910008)

[ssdlDeployed 36](#_Toc342910009)

[Opis 36](#_Toc342910010)

[Powiązane moduły 36](#_Toc342910011)

[graf przepływu 37](#_Toc342910012)

[switchConsole 37](#_Toc342910013)

[Opis 37](#_Toc342910014)

[Powiązane moduły 37](#_Toc342910015)

[graf przepływu 37](#_Toc342910016)

[switchToCF 37](#_Toc342910017)

[Opis 37](#_Toc342910018)

[Powiązane moduły 37](#_Toc342910019)

[graf przepływu 38](#_Toc342910020)

[switchToDF 38](#_Toc342910021)

[Opis 38](#_Toc342910022)

[Powiązane moduły 38](#_Toc342910023)

[graf przepływu 38](#_Toc342910024)

[validationError 38](#_Toc342910025)

[Opis 38](#_Toc342910026)

[Powiązane moduły 38](#_Toc342910027)

[graf przepływu 38](#_Toc342910028)

[validationFailed 39](#_Toc342910029)

[Opis 39](#_Toc342910030)

[Powiązane moduły 39](#_Toc342910031)

[graf przepływu 39](#_Toc342910032)

[validationPassed 39](#_Toc342910033)

[Opis 39](#_Toc342910034)

[Powiązane moduły 39](#_Toc342910035)

[graf przepływu 39](#_Toc342910036)

[validationPerformed 40](#_Toc342910037)

[Opis 40](#_Toc342910038)

[Powiązane moduły 40](#_Toc342910039)

[graf przepływu 40](#_Toc342910040)

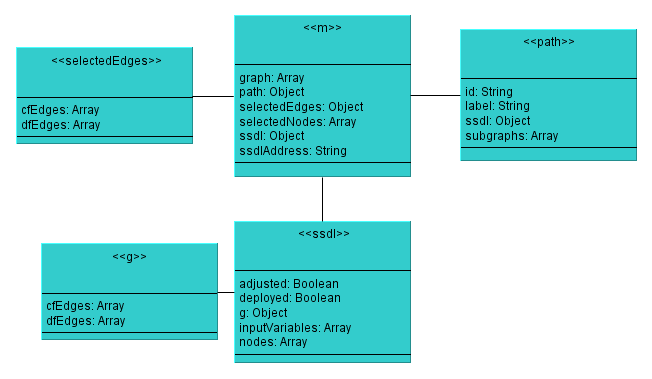
[Wszystkie zdarzenia 40](#_Toc342910041)

# Model danych SSDL

Poniżej przedstawiono model danych używany w aplikacji SSDL\_SIG. Obiekt przedstawiony w tym dokumencie jest przypisany do referencji **m** w aplikacji.

## Dane grafu

### Struktura



### Opis komponentów

|  |  |  |
| --- | --- | --- |
| m | | |
| Nazwa | Typ | Opis |
| graph | Array | Tablica zawiera wszystkie wierzchołki w grafie. Struktura pojedynczego wierzchołka przedstawiona jest poniżej |
| path | Object | Obiekt zawierający ścieżkę dostępu do grafu |
| selectedEdges | Object | Obiekt zawiera referencje do obiektów zaznaczonych strzałek |
| selectedNodes | Array | Tablica zawierające referencje do obiektów **g** wszystkich zaznaczonych wierzchołków |
| ssdl | Object | Obiekt zawierający informacje o całym grafie. Struktura opisana poniżej. |
| ssdlAddress | String | Adres z którego pobrano SSDL z grafem. |

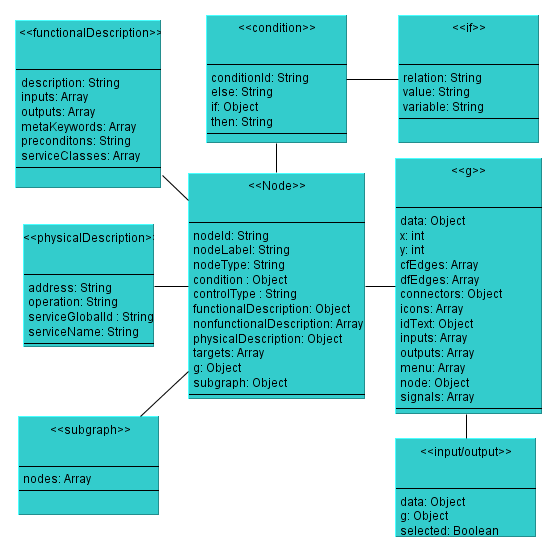
|  |  |  |
| --- | --- | --- |
| path | | |
| Nazwa | Typ | Opis |
| id | String | Unikalny identyfikator grafu |
| label | Srting | Etykieta grafu |
| ssdl | Object | Referencja do obiektu ssdl, opis poniżej |
| subgraphs | Array | Tablica z podgrafami |

|  |  |  |
| --- | --- | --- |
| selectedEdges | | |
| Nazwa | Typ | Opis |
| cfEdges | Array | Tablica z zaznaczonymi strzałkami w trybie control flow |
| dfEdges | Array | Tablica z zaznaczonymi strzałkami w trybie data flow |

|  |  |  |
| --- | --- | --- |
| ssdl | | |
| Nazwa | Typ | Opis |
| adjusted | Boolean | Faga, czy graf został dostosowany |
| deployed | Boolean | Flaga, czy graf został rozmieszczony |
| g | Object | Referencja do obiektu g, zawiera on dwie tablice z referencjami do strzałek w trybie control flow oraz data flow |
| inputVariables | Array | Tablica z wartościami wejściowymi |
| nodes | Array | Tablica zawierająca wszystkie wierzchołki w grafie, opis struktury pojedynczego wierzchołka znajduje się poniżej |

## Wierzchołek

### Struktura



### Opis komponentów

Powyższa struktura przedstawia reprezentację pojedynczego wierzchołka z których zbudowany jest graf. Struktura ta przechowuje informacje które znajdują się w specyfikacji wierzchołka w języku SSDL oraz kilka dodatkowych obiektów odpowiedzialnych za jego prezentację graficzną w systemie SSDL\_SIG. Te dodatkowe obiekty znajdują się w obiekcie g. Są to obiekty utworzone z wykorzystaniem biblioteki Raphael. Ich struktura nie jest przedstawiana w tym dokumencie.

|  |  |  |
| --- | --- | --- |
| Node | | |
| Nazwa | Typ | Opis |
| nodeId | String | Unikalne id wierzchołka |
| nodeLabel | String | Etykieta wierzchołka |
| nodeType | String | Typ wierzchołka |
| condition | Object | Warunek, tylko w przypadku instrukcji warunkowych |
| controlType | String | Rodzaj wierzchołka, tylko dla wierzchołków o nodeType „Control” |
| functionalDescription | Object | Opis funkcjonalny |
| nonfunctionalDescription | Array | Opis niefunkcjonalny |
| physicalDescription | Object | Opis fizyczny |
| targets | Array | Lista nodeId wierzchołków które występują po bieżącym wierzchołku |
| g | Object | Zawiera dane niezbędne do wyświetlania wierzchołka |
| subgraph | Object | Referencja do obiektu podgrafu |

|  |  |  |
| --- | --- | --- |
| condition | | |
| Nazwa | Typ | Opis |
| conditionId | String | Id instrukcji warunkowej |
| Else | String | nodeId wierzchołka który ma zostać wywołany gdy warunek nie zostanie spełniony |
| If | Object | Obiekt warunku |
| Then | String | nodeId wierzchołka który ma zostać wywołany w momencie spełnienia instrukcji w warunku |

|  |  |  |
| --- | --- | --- |
| if | | |
| Nazwa | Typ | Opis |
| Relations | String | Operator warunkowy, możliwe wartości: less, leeq, eq,geq, gt |
| Value | String | Wartość z którą będzie porównywane wejście obiektu |
| Variable | String | Typ wartości |

|  |  |  |
| --- | --- | --- |
| functionalDescription | | |
| Nazwa | Typ | Opis |
| description | String | Pole zawierające słowny opis danego wierzchołka |
| inputs | Array | Tablica z nodeId wierzchołków wchodzących |
| outputs | Array | Tablica z nodeId wierzchołków wychodzących |
| metaKeywords | Array | Tablica zawierająca słowa kluczowe |
| preconditions | String | Warunki wstępne |

|  |  |  |
| --- | --- | --- |
| physicalDescription | | |
| Nazwa | Typ | Opis |
| address | String | Adres usługi |
| operatin | String | Nazwa wywoływanej metody |
| serviceGlobalId | String | Pełen adres wywoływanej metody |
| serviceName | String | Nazwa usługi |

|  |  |  |
| --- | --- | --- |
| subgraph | | |
| Nazwa | Typ | Opis |
| nodes | Array | Tablica zawierająca wierzchołki podgrafu, wierzchołki w podgrafach mają identyczną strukturę jak wierzchołki w grafie |

|  |  |  |
| --- | --- | --- |
| g | | |
| Nazwa | Typ | Opis |
| data | Object | Referencja na obiekt wierzchołka |
| X | Int | Pozycja x wierzchołka |
| y | Int | Pozycja y wierzchołka |
| cfEcges | Array | Tablica zawierająca referencje do obiektów strzałek w trybie control flow |
| dfEdges | Array | Tablica zawierająca referencje do wierzchołków strzałek w trybie data flow |
| connectors | Array | Tablica zawierająca obiekty biblioteki Raphael. Obiekty te to miejsca do których rysowane są strzałki w trybie control flow |
| icons | Array | Tablica zawierająca referencje do obiektów biblioteki Raphael. Obiekty te to ikonki wyświetlane w lewym dolnym rogu wierzchołka. |
| idText | Object | Obiekt biblioteki Raphael. Jest to element wyświetlający nodeId jako tekst na wierzchołku |
| inputs | Array | Tablica zawierająca obiekty wejść. Ich strukturę przedstawia klasa input/output |
| outputs | Array | Tablica zawierająca obiekty wyjść. Ich strukturę przedstawia klasa input/output |
| menu | Array | Tablica zawierająca obiekty biblioteki Raphael. Obiekty te reprezentują ikony menu pojawiającego się z prawej strony wierzchołka po wskazaniu go kursorem. |
| node | Object | Obiekt biblioteki Raphael. Jest to obiekt zawierający główną część wyświetlanego wierzchołka |
| signals | Array | Obiekty biblioteki Raphael. Obiekty sygnalizujące stan wierzchołka, znajdują się w lewym górnym rogu. |

# Konfiguracje

## Załorzenia

Głównym założeniem konfiguracji jest umożliwienie uruchamiania jedynie niektórych modułów aplikacji SSDL\_SIG w zależności od potrzeb. Pliki konfiguracyjne znajdują się w folderze **frontEnd->configs**. Są to pliki tekstowe z rozszerzeniem **.cfg**

## Uruchamianie wybranej konfiguracji

W celu uruchomienia aplikacji w konfiguracji innej niż domyślna należy wywołać aplikację przekazując w zmiennej mode nazwę konfiguracji np.

*adres-aplikacji.com/?mode=bspe*

## Zawartość pliku konfiguracyjnego

Przykładowa zawartość pliku konfiguracyjnego:

|  |
| --- |
| //system-libraries------------------------------------------------------------------------------>>>  libraries/extension.js  libraries/support.js  libraries/raphael.js  //system-settings------------------------------------------------------------------------------->>>  settings/main.js  settings/CxSE.js  settings/viewSettings.js  settings/meetTheJsons.js  settings/historySettings.js  //key-modules----------------------------------------------------------------------------------->>>  modules/eventMenager.js  modules/shortcut.js  modules/specialEvents.js  //high-priority-modules------------------------------------------------------------------------->>>  modules/viewResizer.js  //standard-modules------------------------------------------------------------------------------>>>  modules/deleter.js  modules/validator.js  modules/deparser.js  modules/mainMenu.js  modules/contextMenu.js  modules/subgraphNavigator.js  modules/deployer4.js  modules/ssdlParser.js  modules/externalInterface.js  modules/tooltipper.js  modules/nodeVisualiser.js  modules/logger2.js  modules/arrow.js  modules/selectorRectangle.js  modules/blankNode.js  modules/resizer.js  modules/formGenerator.js  modules/history.js  modules/atomic.js  modules/atomicrepo.js  //actions--------------------------------------------------------------------------------------->>>  modules/init.js  //end-of-configurable-content------------------------------------------------------------------->>> |

W kolejnych liniach znajdują się ścieżki do poszczególnych modułów które zostaną uruchomione. Niektóre z modułów mają większe znaczenie od innych, dlatego są wypisane w odpowiedniej sekcji.

Opis sekcji:

* system-libraries: Biblioteki zewnętrzne
* system-settings: Pliki zawierające ustawienia początkowe modułów. Nie wszystkie moduły wymagają takich ustawień. Zawartość tych plików to np. ustawienia rozmiarów poszczególnych elementów.
* key-modules: Moduły o kluczowym znaczeniu z punktu widzenia aplikacji. Są one ładowane najwcześniej.
* high-priority-modules: Moduły o podwyższonym priorytecie, ładowane przed standardowymi modułami
* standard-modules: Zestaw standardowych modułów.
* actions: Obsługa zdarzeń wywoływanych przy starcie aplikacji

## Edycja i dodawanie modułów

W celu zamiany jednego z modułów innym wystarczy podmienić ścieżkę starego modułu na nową. Aby dodać nowy moduł wystawczy dopisać ścieżkę prowadzącą do niego w nowej linii.

W przypadku podmiany modułu należy pamiętać, że nowy moduł musi posiadać ten sam zewnętrzny interfejs oraz powinien reagować i wywoływać te same zdarzenia co stary. W przeciwnym wypadku zmiana modułu może mieć wpływ na inne a to z kolei może zakłócić działanie aplikacji.

# Moduły

## Event manager

### Opis

Event manager jest jednym z najważniejszych modułów aplikacji. Ponieważ JavaScript jest językiem sterowanym zdarzeniami do stworzenia aplikacji SSDL\_SIG wykorzystano paradygmat porgramowania sterowanego zdarzeniami. Moduł daje możliwość definiowania wewnętrznych zdarzeń, które mogą być wywoływane w odpowiedzi na reakcję użytkownika, zdarzenie przeglądarki lub w innym dowolnym momencie podczas działania program. Zadaniem event managera jest obsługa wewnętrznych zdarzeń w aplikacji. Działanie większości modułów opiera się na jego działaniu. Jego zastosowanie umożliwia dodawanie nowych modułów bez ingerencji w już istniejące.

### Model danych

Obsługa zdarzeń realizowana jest przez funkcje **invokeEvent()** oraz **attachListener().**

Jako parameter funkcja invokeEvent przujmuje nazwę zdarzenia so wywołania oraz opcjonalne parametry które mają zostać przekazane do funkcji słuchacza.

Funkcja attachListener przyjmuje jako parametry nazwę zdarzenia, funkcję która zostanie wykonana w momencie wystąpienia zdarzenia, nazwę modułu w którym słuchacz został zadeklarowany oraz opis.

Moduł dostarcza również funkcję **detachListener()** która jako parameter przyjmuje id listenera zwróconego przez funkcję attachListener()

### Sposób działania modułu

W momencie wywołania funkcji invokeEvent moduł uruchamia wszystkie funkcje nasłuchujące na dane zdarzenie. Przykładowo podczas startu aplikacji, po załadowaniu silnika, wywoływane jest zdarzenie *initView.* Łańcuch funkcji wywoływanych w reakcji na kego wystąpienie może wyglądać następująco:



Można łatwo zauważyć, że funkcje wywoływane w odpowiedzi na wystąpienie zdarzenia mogą wywoływać kolejne. Pozwala to na tworzenie bardziej skomplikowanych łańcuchów zdarzeń co skutkuje precyzyjną reakcją na wystąpienie konktnego zdarzenia.

## Arrow

### Opis

Zadaniem modułu “arrow” jest rysowanie strzałek łączących wierzchołki w grafie. Strzałki rysowane są poprzez kliknięcie myszką od wyjścia wierzchołka do wejścia wierzchołka w grafie. Ponadto moduł reaguje na zaznaczenie strzałki poprzez zmianę jej koloru

### Model danych

Strzałka dołączana jest do wejść i wyjść wierzchołków, które łączy, a więc nie posiada własnej reprezentacji danych.

### Sposób działania modułu

Moduł obsługuje dwa tryby działania: ControlFlow oraz DataFlow.

**ControlFlow:**

Moduł wyszukuje najkrótsze możliwe połączenie pomiędzy konkretnymi wierzchołkami, pomiędzy którymi narysowana ma zostać strzałka.

**DataFlow:**

Rysuje strzałkę pomiędzy wyjściami i wejściami konkretnych wierzchołków

## Atomic

### Opis

### Model danych

aa

### Sposób działania modułu

aa

### Ustawienia

aa

|  |  |
| --- | --- |
| **Ustawienie** | **Opis** |
|  |  |

## History

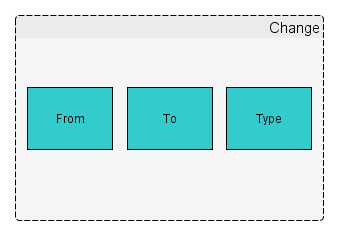
### Opis

Zadaniem modułu historii jest umożliwienie zapamiętania zmian dokonanych przez użytkownika, w celu ewentualnych ich cofnięcia bądź powtórzenia. Po zmianie moduł zapisuje zmienione dane w tablicy. Cofnięcie zmiany lub powtórzenie czynności możliwe jest za pomocą skrótów klawiszowych bądź też poprzez wybór odpowiedniej opcji z menu głównego.

Moduł jest powiązany z innymi modułami za pomocą zdarzeń, które są obsługiwane przez moduł historii.

### Model danych

Dane są przechowywane w tablicy i mają następującą postać:



From: Obiekt, w którym przechowywane są dane historyczne, to znaczy takie które są zachowane przed dokonaniem zmiany przez użytkownika.

To: Obiekt, w którym przechowywane są dane aktualne, to znaczy takie, które są w danym momencie wyświetlane, po zmianach dokonanych przez użytkownika.

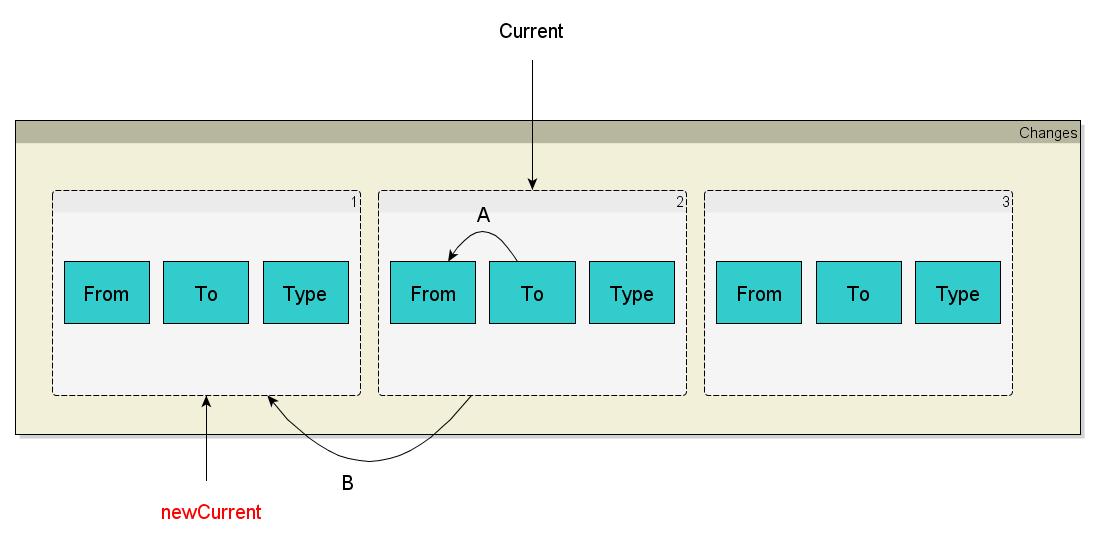
Type: Typ zmiany zapisany w postaci tekstowej pozwalający na rozróżnienie ostatniej dokonanej zmiany.

Tablica zmian zawierać może w sobie wiele powyższych obiektów. Ich liczba ustalana jest w osobnym pliku ustawień. Do wskazania na ostatnią dokonaną zmianę służy wskaźnik zdefiniowany w module.

### Sposób działania modułu

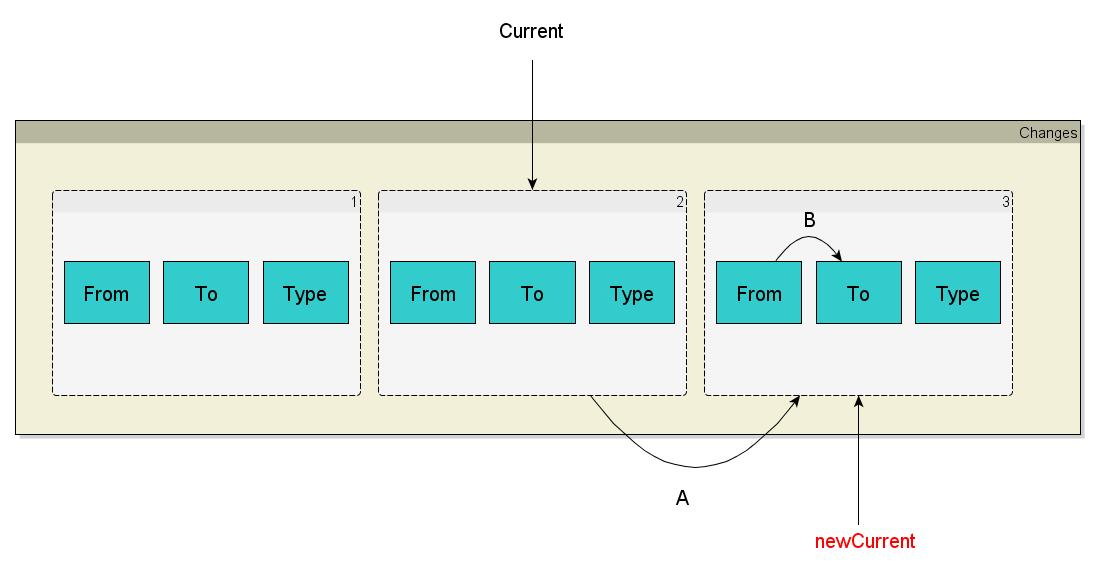
**Cofnięcie zmiany:**

Wskaźnik “Current” ustawiony jest na konkretnej pozycji. Przy cofnięciu zmiany zmieniają się dane, które są aktualnie wyświetlane. Dane, które należy podmienić determinowane są przez typ “Type”. Zmianę aktualnych danych reprezentuje na rysuku strzałka A. Następnie wskaźnik przesuwany jest na poprzednią pozycję w tablicy: strzałka B.



**Powtórzenie zmiany:**

Wskaźnik “Current” ustawiony jest na konkretnej pozycji. Przy powtórzeniu zmiany najpierw zmienia się położenie wskaźnika, co reprezentowane jest na rysunku przez strzałkę A, a następnie podmieniane są wyświetlane dane. Dane, które należy podmienić determinowane są przez typ “Type”. Zmianę aktualnych danych reprezentuje na rysuku strzałka B.

**Cofnięcie wszystkich zmian:**

Polega ono na przejściu z aktualnej zmiany wskazywanej przez wskaźnik do początku tablicy, co odbywa się na takiej samej zasadzie jak dla jednej zmiany, z tym że podmiana danych wykonywana jest w pętli aż osiągnięty zostanie początek tablicy.

**Powtórzenie wszystkich zmian:**  
  
Polega ono na przejściu z aktualnej zmiany wskazywanej przez wskaźnik na koniec tablicy zmian, co odbywa się na takiej samej zasadzie jak dla pojedynczej zmiany, z tym że podmiana danych dokonywana jest w pętli, aż osiągnięty zostanie koniec tablicy.

### Zdarzenia

**Zdarzenia obsługiwane przez moduł “history”:**

|  |  |
| --- | --- |
| **Zdarzenie** | **Opis działania** |
| historyChanged | Dodaje obiekt zmiany do tablicy zmian |
| shortcutBack | Cofa zmianę |
| shortcutForward | Powtarza zmianę |
| shortcutBackAll | Cofa wszystkie zmiany |
| shortcutForwardAll | Powtarza wszystkie zmiany |
| historyDeleteBlankNode | Usuwa wierzchołek |

**Zdarzenia wysyłane przez moduł “history”**

|  |  |
| --- | --- |
| **Zdarzenie** | **Opis działania** |
| refreshView | Odświeża aktualnie wyświetlany widok |
| replaceSsdl | Podmienia wyświetlany graf. |

### Ustawienia

Ustawienia przechowywane są w oddzielnym folderze ustawień. Plik ustawień modułu *historySettings*:

|  |  |
| --- | --- |
| **Ustawienie** | **Opis** |
| historySize | Maksymalny rozmiar, jaki osiągnąć może tablica zawierająca zmiany w module historii. |

## contextmenu

### Opis

Moduł odpowiedzialny jest za wizualizację menu kontekstowego po kliknięciu przez użytkownika prawym przyciskiem myszy na konkretnym obiekcie, do którego podłączone jest menu kontekstowe. Istnieje także możliwość podłączenia menu kontekstowego do lewego przycisku myszy wykorzystując specjalny parametr

### Model danych

Menu kontekstowe zbudowane jest na zasadzie drzewa. Posiada ono własny model danych. Istnieje globalna tablica menu, do którego dodawany jest obiekt menu kontekstowego, co pozwala na zamknięcie wszystkich obiektów menu poprzez kliknięcie lewym przyciskiem myszy na dowolnym miejscu poza obiektem menu.

Każdy obiekt menu kontekstowego zawiera 3 dane:

1. Obiekt wywołujący menu
2. Korzeń struktury drzewiastej przechowującej opcje.
3. Zbiór podmenu, które sa aktualnie otwarte.

### Sposób działania modułu

Funkcja contextMenu buduje obiekt menu, która jako parametry przyjmuje nasłuchiwany obiekt i opcjonalnie możliwość dodanie akcji po kliknięciu, a także możliwość odblokowania propagacji zdarzenia kliknięcia.

### Zdarzenia

**Zdarzenia obsługiwane przez moduł “contextMenu”:**

|  |  |
| --- | --- |
| **Zdarzenie** | **Opis działania** |
| onEscape | Zamyka wszystkie menu |

**Zdarzenia wysyłane przez moduł “contextMenu”**

|  |  |
| --- | --- |
| **Zdarzenie** | **Opis działania** |
| Zdarzenie zdefiniowane przy inicjalizacji menu | Zdarzenie zdefiniowane przy inicjalizacji menu |

## Deployer

### Opis

Moduł rozmieszczający graf na kanwie.

### Model danych

Moduł posiada własny parser, ktróry przygotowuje tablicę węzłów, która posiada referencje rodzic- dziecko pomiędzy węzłami.

Na potrzeby działania modułu konieczne jest również zdefiniowanie zmiennych:

1. level- określająca poziom położenia wierzchołka w hierarchii
2. column- numer kolumny w siatce grafu liczony od lewej do prawej. Obrazuje kolejność węzłów na danym poziomie hierarchii.

W późniejszym etapie działania modułu tworzona jest macierz grafu, która zawiera obiekty reprezentujące węzły.

### Sposób działania modułu

Stosując macierz grafu moduł tworzy optymalną strukturę hierarchiczną, a an jej podstawie wylicza współrzędne połorzenia konkretnych wierzchołków grafu.

Moduł może działać w dwóch trybach:

1. Korzystający z metody “bruteforce” (graf poniżej 18 węzłów)
2. Korzystający z algorytmu genetycznego

### Zdarzenia

**Zdarzenia obsługiwane przez moduł**

|  |  |
| --- | --- |
| **Zdarzenie** | **Opis działania** |
| deploySsdl | Rozmiescza graf, wpisuje współrzędne wierzchołków i dopasowuje rozmiar kanwy. |
| redeploySsdl | Ponownie rozmieszcza graf. Ponadto wysyła zdarzenie historyChanged, które umozliwia nawigację po historii zmian. |

**Zdarzenia wysyłane przez moduł ”**

|  |  |
| --- | --- |
| **Zdarzenie** | **Opis działania** |
| ssdlDeployed | Informuje o zakończeniu działania rozmieszczania grafu. |

### Ustawienia

Ustawienia zawarte są w oddzielnym pliku

|  |  |
| --- | --- |
| **Ustawienie** | **Opis** |
| nodeW | Szerokość wierzchołka |
| nodeH | Wysokość wierzchołka |

## externalInterface

### Opis

Dodaje zestaw funkcji komunikacji zewnętrznych do modelu danych zawartego w silniku.

### Sposób działania modułu

Moduł służy za pośrednika przy wywoływaniu funkcjonalności innych modułów takich jak:

1. Parser
2. Validator
3. AtomicServices
4. Deparser

### Zdarzenia

**Zdarzenia wysyłane przez moduł ”**

|  |  |
| --- | --- |
| **Zdarzenie** | **Opis działania** |
| responseFormatError | Wysyłane w momencie błędnego format odpowiedzi |
| unknownAjaxError | Wysyłane w moemencie błędu w komunikacji porpzez Ajax |

## logger

### Opis

Moduł wypisujący informacje zwrotne do użytkownika. Można je zobaczyć poprzez kliknięcie na przycisk otwierający konsolę.

### Model danych

Moduł posiada własny model dancyh, który przechowuje zbiór wiadomości. Wiadomość ma następujący format:

1. Type- typ wiadomości
2. Title- opcjonalny nagłówek wiadomości
3. Message- treść wiadomości

Ponadto przechowywane są liczniki dla poszczególnych typów wiadomości oraz ustawienia ich wyświetlania.

### Sposób działania modułu

Moduł podłączony jest do widoku za pomocą funkcji, które wywołują dodanie konkretnego typu wiadomośći do danych. Moduł posiada także zdefiniowane funkcje, które otwierają i zamykają konsolę.

### Ustawienia

aa

|  |  |
| --- | --- |
| **Ustawienie** | **Opis** |
|  |  |

## shortcut

### Opis

Moduł obsługuje skróty klawiszowe, za pomocą których mogą być wywoływane odpowiednie funkcjonalności interfejsu.

### Model danych

Moduł wykorzystuje zewnętrzną bibliotekę do potrzeb działania interfejsu.

### Sposób działania modułu

Za pomocą odpowiedniej funkcji kontrolera skrót klawiszowy jest dodawany do silnika. Można go również usunąć za pomocą silnika. Wciśnięcie skrótu klawiszowego inicjuje zadane zdarzenie, które zdefiniowane jest przy dodawaniu skrótu klawiszowego do silnika.

### Zdarzenia

Moduł nie wywołuje żadnych zdarzenń ani nie nasłuchuje na żadne zdarzenia

### zdefiniowane skróty klawiszowe

|  |  |  |
| --- | --- | --- |
| **Skrót** | **Moduł definiujący** | **Wysyłane zdarzenie** |
| Escape | SpecialEvents | shortcutEscape |
| Ctrl+S | SpecialEvents | shortcutSave |
| Ctrl+A | SpecialEvents | shortcutSelectAll |
| Ctrl+C | SpecialEvents | shortcutCopy |
| Ctrl+X | SpecialEvents | shortcutCut |
| Ctrl+V | SpecialEvents | shortcutPaste |
| Ctrl+Z | SpecialEvents | shortcutBack |
| Ctrl+Y | SpecialEvents | shortcutForward |
| Delete | SpecialEvents | shortcutDelete |
| Shift+N | SpecialEvents | shortcutNew |
| Shift+Q | SpecialEvents | switchConsole |
| Shift+C | SpecialEvents | switchToCF |
| Shift+D | SpecialEvents | switchToDF |
| Shift+R | SpecialEvents | redeploySsdl |
| Shift+Z | History | shortcutBackAll |
| Shift+Y | History | shortcutForwardAll |
| Shift+I | Resizer | enlargeGraph |
| Shift+O | Resizer | reduceGraph |
| Shift+U | Resizer | restoreGraph |

## specialEvents

### Opis

Dopisuje zdarzenia odpowiedzialne za inicjację przepływu najważniejszych zdarzeń, które muszą wykonać się jako pierwsze przy uruchomieniu interfejsu.

### Model danych

Moduł nie posiada modelu danych

### Sposób działania modułu

Przypisuje do zdarzeń JavaScript zdarzenia silnika. Ponadto definiuje większość skrótów klawiszowych.

### Zdarzenia

**Zdarzenia wysyłane przez moduł ”**

|  |  |
| --- | --- |
| **Zdarzenie** | **Opis działania** |
| onEscape | Wywołuje zamknięcie wszystkich elementów wizualnych, które zostały wcześniej otworzone. Zdarzenie przypisane jest do JavaScriptowej funkcji “onclick” |

## subgraphNavigator

### Opis

Zapewnia możliwość przechodzenia po grafach oraz podgrafach, które mogą zawierać poszczególne wierzchołki.

### Model danych

Moduł nie posiada własnego modelu danych

### Sposób działania modułu

Modul przy każdym odświeżeniu widoku buduje nową ścieżkę grafu, który jest aktualnie wyświetlany w interfejsie graficznym. Pozwala na przechodzenie pomiędzy dowolnymi elementami drzewa jego podgrafów lub grafów macierzystych. Korzysta przy tym z funkcjonalności modułu contextMenu do wyświetlania drzewa grafu i podgrafów

### Zdarzenia

**Zdarzenia obsługiwane przez moduł**

|  |  |
| --- | --- |
| **Zdarzenie** | **Opis działania** |
| initView | Inicjalizuje część graficzną modułu |
| refreshView | Pokazuje aktualną ścieżkę grafów wyświetlanym drzewie |

**Zdarzenia wysyłane przez moduł ”**

|  |  |
| --- | --- |
| **Zdarzenie** | **Opis działania** |
| onEscape | Zamyka wyświetlane elementy graficzne |
| historyChanged | Dodaje zmianę podmiany grafu do historii |
| replaceSsdl | Przeładowuje ssdl oraz inicjalizuje przeładowanie widoku |

## ViewResizer

### Opis

Inicjuje zmienne ustawień odpowiedzialne za konfigurację przeglądarki użytkownika wymagane do pracy z interfejsem oraz rozmieszczenia obiektów w interfejsie.

### Model danych

Moduł nie posiada modelu danych

### Sposób działania modułu

Definiuje zestaw funkcji odpowiedzialnych za przekształcenia okna przeglądarki dostosowując interfejs do napotkanych warunków (np rozdzielczość).

Dołącza się głównie do zdarzeń języka JavaScript.

### Zdarzenia

**Zdarzenia obsługiwane przez moduł**

|  |  |
| --- | --- |
| **Zdarzenie** | **Opis działania** |
| initView | Dopasowuje interfejs do rozmiaru okna. Definiuje podstawowe zmienne |
| refreshView | Wykonuje operacje na kanwie: dostosowuje ją do wyświetlanego grafu. |

# Zdarzenia

### Konwencja

Do Opisu przepływu danych między zdarzeniami przyjęto następującą konwencję:



Module1: Elipsa symbolizująca moduł wywołujący zdarzenie

InvokeEvent: Strzałka wychodząca z elipsy, wchodząca do prostokąta symbolizuje wywołanie zdarzenia przez moduł

Event: Prostokąt symbolizujący zdarzenie. Etykieta jest nazwą zdarzenia

ListenEvent Strzałka wychodząca z prostokąta, wchodząca do elipsy symbolizuje reakcję modułu na zdarzenie, na które moduł ten nasłuchuje.

Module2: Elipsa symbolizująca moduł obsługujący zdarzenie, na które nasłuchuje.

## addBlankNode

### Opis

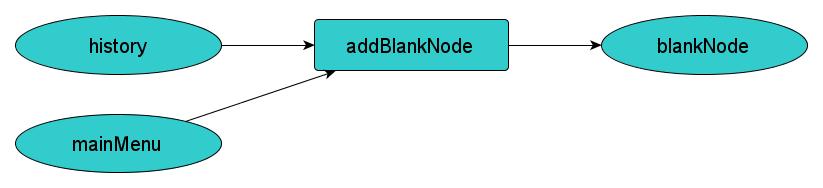
Zdarzenie wywoływane jest przez użytkownika za pośrednictwem menu głównego bądź za pośrednictwem modułu historii. Zdarzenie obsługiwane rzez moduł “blankNode” dodaje do grafu pusty obiekt i wywołuje moduł “nodeVisualiser”, który rysuje dany wierzchołek.

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| History | Wywoływacz |
| mainMenu | Wywoływacz |
| blankNode | Słuchacz |

### graf przepływu



## atomicParseError

### Opis

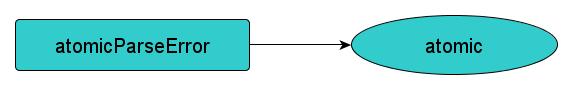
Zdarzenie wywoływane w momencie, gdy połączenie Ajax pobierające dane o usługach atomowych zwróci błąd

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| Atomic | Słuchacz |

### graf przepływu



## atomicparserperformed

### Opis

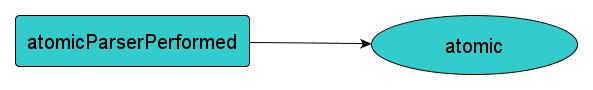
tODO

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| Atomic | Słuchacz |

### graf przepływu



## deploySSDL

### Opis

Zdarzenie wywoływane przez moduł “ssdlParser”, informujące o sparsowaniu danego grafu do postaci danych wykorzystywanych w interfejsie, które obsługiwane jest przez moduł “deployer” znajdujący położenie wierzchołków grafu

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| ssdlParser | Wywoływacz |
| Deployer | Słuchacz |

### graf przepływu



## engineLoaded

### Opis

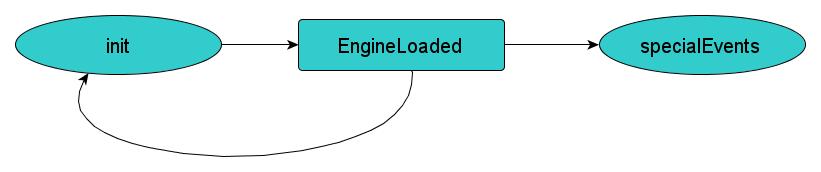
EngineLoaded jest to zdarzenie, które wywoływane jest przy inicjalizacji interfejsu. Ładuje SSDL.

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| Init | Wywoływacz, Słuchacz |
| specialEvents | Słuchacz |

### graf przepływu



## enlargeGraph

### Opis

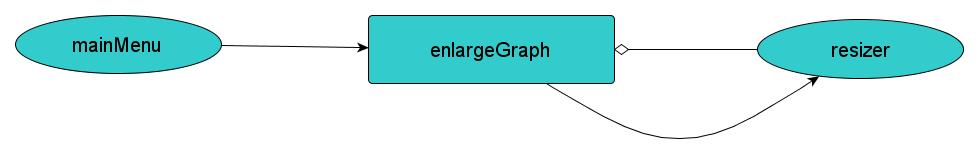
Zdarzenie w momencie, gdy konieczne żądane jest powiększenie grafu. Obsługiwane jest przez moduł “resizer”, który zmienia ustawienia rysowania wierzchołków grafu, a następnie komunikuje się z modułem “nodeVisualiser”, aby graf został przerysowany. Ponadto moduł “resizer” definiuje skrót klawiszowy, za pomocą którego graf może być powiększony.

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| mainMenu | Wywoływacz |
| Resize | Słuchacz |

### graf przepływu



## formSubmitted

### Opis

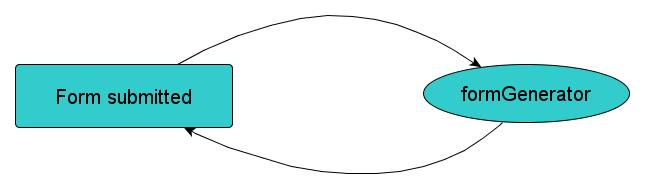
Zdarzenie wywoływane jest w momencie kliknięcia przycisku zapisz znajdującego się pod formularzem. Przekazuje dane z formularza zmodyfikowanego przez użytkownika do wierzchołka i zamyka formularz. Następuje również wyczyszczenie danych w celu uniknięcia ich nadpisania przy następnym wykorzystaniu formularza.

### Powiązane moduły

**Zdarzenia obsługiwane przez moduł**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| formGenerator | Wywoływacz, Słuchacz |

### graf przepływu



## historyChanged

### Opis

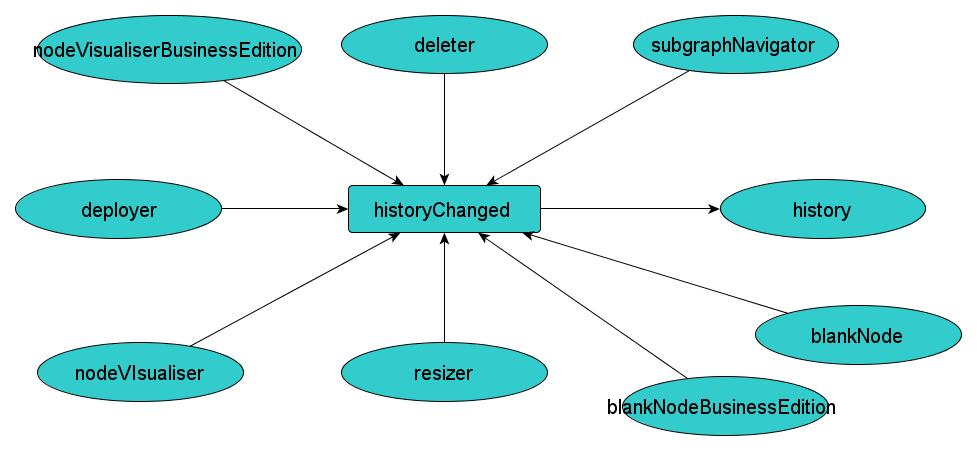
Zdarzenie wywoływane przez szereg różnych modułów, które zapewnia komunikację z modułem historii. Moduł historii obsługując zdarzenie dodaje zmianę do historii zmian, przez co umożliwia cofnięcie bądź powtórzenie zmian.

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| Deployer | Wywoływacz |
| nodeVisualiserBusinessEdition | Wywoływacz |
| Deleter | Wywoływacz |
| subgraphNavigator | Wywoływacz |
| blankNode | Wywoływacz |
| resizer | Wywoływacz |
| nodeVisualiser | Wywoływacz |
| History | Słuchacz |

### graf przepływu



## historyDeleteBlanknode

### Opis

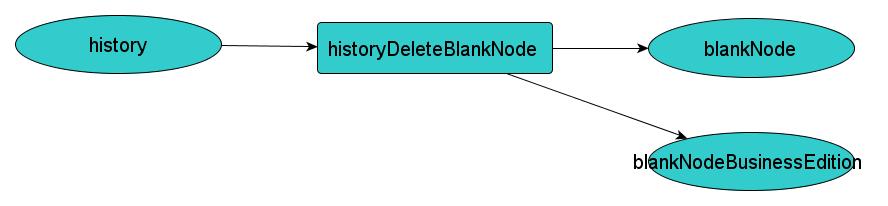
Zdarzenie wywoływane przez moduł historii w celu umożliwienia przechowania danych przez moduł “blankNode“, który tworzy nowy obiekt wierzchołka na potrzeby ewentualnego przekazania go później do modułu historii przy zmianach wprowadzanych przez użytkownika.

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| History | Wywoływacz |
| blankNode | Słuchacz |
| blankNodeBusinessEdition | Słuchacz |

### graf przepływu



## initView

### Opis

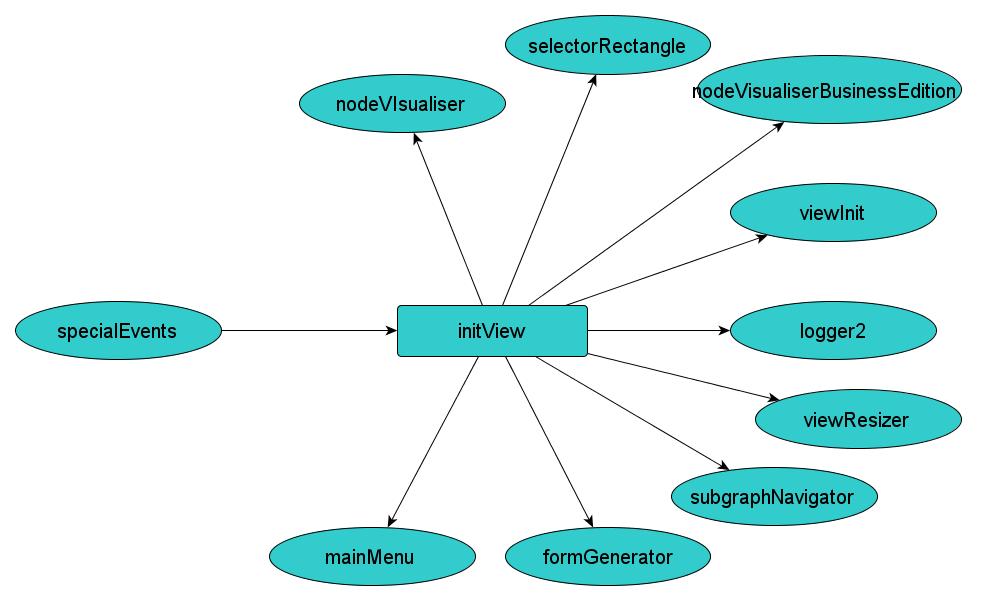
Zdarzenie specjalne wywoływane na początku działania interfejsu. Informuje o załadowaniu widoku.

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| specialEvents | Wywoływacz |
| nodeVisualiser | Słuchacz |
| selectorRectangle | Słuchacz |
| nodeVisualiserBusinessEdition | Słuchacz |
| viewInit | Słuchacz |
| Logger2 | Słuchacz |
| viewResizer | Słuchacz |
| subgraphNavigator | Słuchacz |
| formGenerator | Słuchacz |
| mainMenu | Słuchacz |

### graf przepływu



## onEscape

### Opis

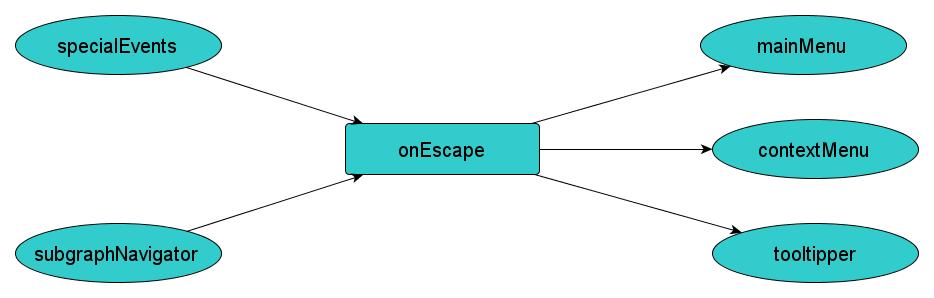
Zdarzenie wywoływane w momencie kliknięcia myszką na na pustym polu znajdującym się na kanwie pozwalające na informację o potrzebie ukrycia elementów widoku pokazywanych przez dane moduły

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| specialEvents | Wywoływacz |
| subgraphNavigator | Wywoływacz |
| mainMenu | Słuchacz |
| contextMenu | Słuchacz |
| tooltipper | Słuchacz |

### graf przepływu



## openForm

### Opis

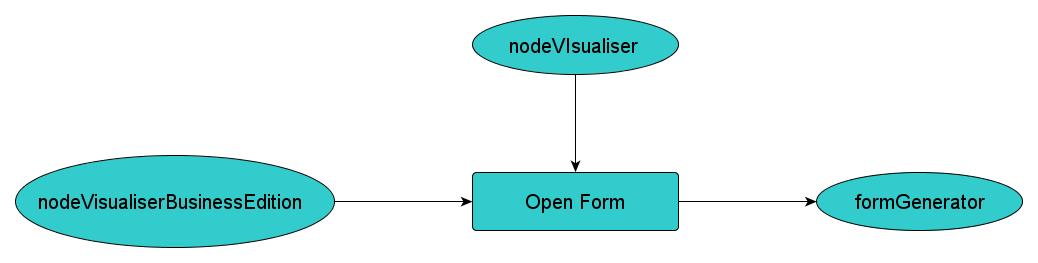
Zdarzenie reaguje na akcję otwarcia formularza. Akcja ta może być wywołana przez użytkownika za pośrednictwem modułów obsługujących wizualizację wierzchołków.

### Powiązane moduły

**Zdarzenia obsługiwane przez moduł**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| nodeVisualiserBusinessEdition | Wywoływacz |
| nodeVisualiser | Wywoływacz |
| formGenerator | Słuchacz |

### graf przepływu



## SSDLloaded

### Opis

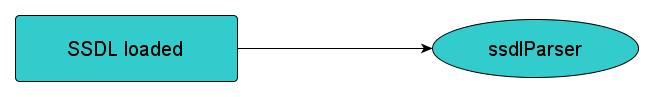
Zdarzenie wywoływane automatycznie przodczas zdarzenia EngineLoaded. Informuje, że ssdl został załadowany.

### Powiązane moduły

**Zdarzenia obsługiwane przez moduł**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| ssdlParser | Słuchacz |

### graf przepływu



## ssdlParsed

### Opis

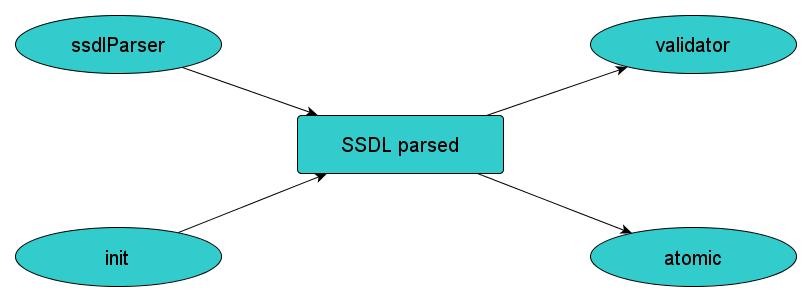
Zdarzenie informujące o fakcie przeparsowania SSDL

### Powiązane moduły

**Zdarzenia obsługiwane przez moduł**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| ssdlParser | Wywoływacz |
| Init | Wywoływacz |
| Validator | Słuchacz |
| Atomic | Słuchacz |

### graf przepływu



## parserGotData

### Opis

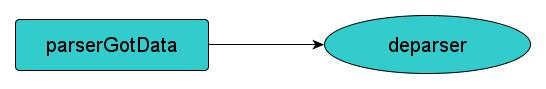
Zdarzenie wywoływane w momencie, gdy todo

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| Deparser | Słuchacz |

### graf przepływu



## parserPassed

### Opis

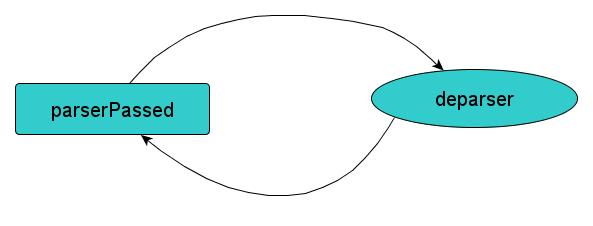
Zdarzenie wywoływane w momencie, gdy połączenie Ajax pobierające dane o usługach atomowych zakończy się powodzeniem

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| deparser | Wywoływacz, słuchacz |

### graf przepływu



## parsertranserror

### Opis

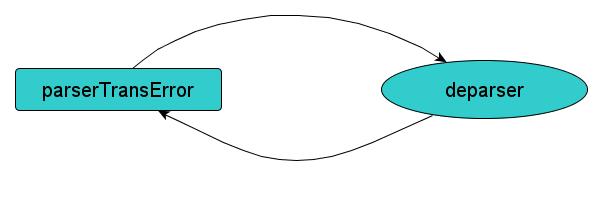
todo

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| Deparser | Wywoływacz, słuchacz |

### graf przepływu



## redeployssdl

### Opis

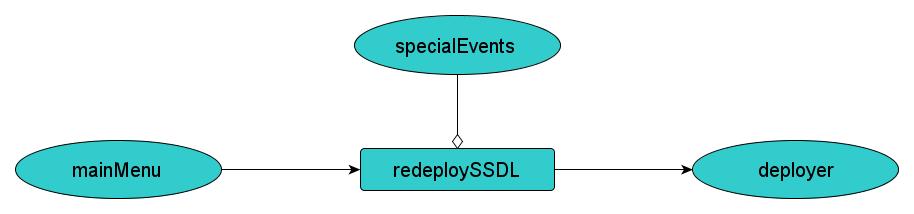
Zdarzenie wywołwane za pomocą menu głównego, bądź skrótu klawiszowego zdefiniowanego w module secialEvents, które obsługiwane jest przez moduł deployer, który rozmieszcza graf.

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| mainMenu | Wywoływacz |
| deployer | Słuchacz |

### graf przepływu



## reducegraph

### Opis

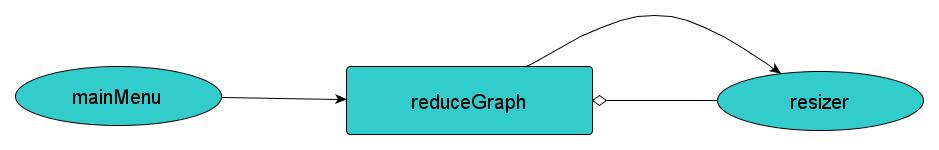
Zdarzenie w momencie, gdy konieczne żądane jest pomniejszenie grafu. Obsługiwane jest przez moduł “resizer”, który zmienia ustawienia rysowania wierzchołków grafu, a następnie komunikuje się z modułem “nodeVisualiser”, aby graf został przerysowany. Ponadto moduł “resizer” definiuje skrót klawiszowy, za pomocą którego graf może być pomniejszony.

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| mainMenu | Wywoływacz |
| resizer | Słuchacz |

### graf przepływu



## selectnone

### Opis

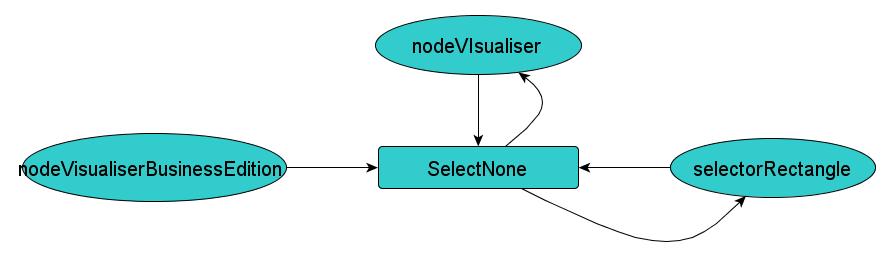
Zdarzenie odpowiadające za odznaczenie zaznaczonych elementów grafu

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| nodeVisualiser | Wywoływacz, słuchacz |
| nodeVisualiserBusinessEdition | Wywoływacz, słuchacz |
| selectorRectangle | Wywoływacz, słuchacz |

### graf przepływu



## shortcutBack

### Opis

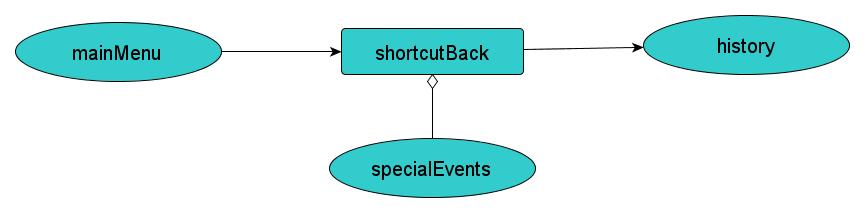
Zdarzenie wysyłane w momencie żądania przez użytkownika cofnięcia ostatniej dokonanej zmiany. Obsługiwane jest przez moduł historii, a skrót klawiszowy do wywołania tego zdarzenia zdefiniowany jest w module “specialEvents”

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| mainMenu | Wywoływacz |
| History | Słuchacz |

### graf przepływu



## shortcutbackall

### Opis

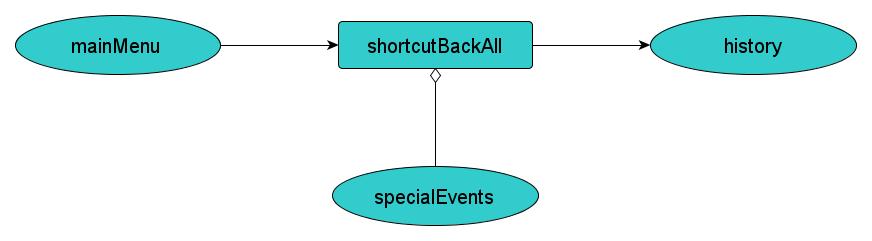
Zdarzenie wysyłane w momencie żądania przez użytkownika cofnięcia do początku historii zmian. Obsługiwane jest przez moduł historii, a skrót klawiszowy do wywołania tego zdarzenia zdefiniowany jest w module “specialEvents”

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| mainMenu | Wywoływacz |
| History | Słuchacz |

### Graf przepływu



## shortcutDelete

### Opis

Zdarzenie wywoływane w momencie żądania przez użytkownika usunięcia zaznaczonego wierzchołka poprzez wciśnięcie klawisza “delete”.

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| deleter | Słuchacz |

### graf przepływu



## shortcutEscape

### Opis

Zdarzenie wywoływane w momencie wciśnięcia przez użytkownika klawisza “Escape” pozwalające na zamknięcie formularza powiązanego z modułem “formGenerator”

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| formGenerator | słuchacz |

### graf przepływu



## shortcutForward

### Opis

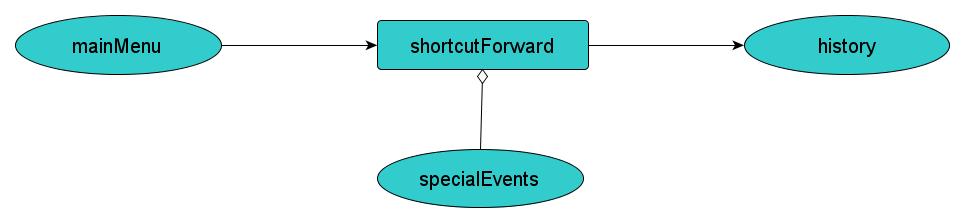
Zdarzenie wysyłane w momencie żądania przez użytkownika powtórzenia ostatniej dokonanej zmiany. Obsługiwane jest przez moduł historii, a skrót klawiszowy do wywołania tego zdarzenia zdefiniowany jest w module “specialEvents”

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| mainMenu | Wywoływacz |
| history | Słuchacz |

### graf przepływu



## shortcutForwardAll

### Opis

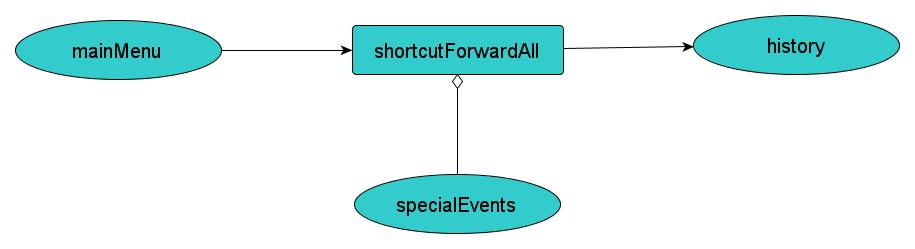
Zdarzenie wysyłane w momencie żądania przez użytkownika powtórzenia zmian do ostatniej dokonanej zmiany. Obsługiwane jest przez moduł historii, a skrót klawiszowy do wywołania tego zdarzenia zdefiniowany jest w module “specialEvents”

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| mainMenu | Wywoływacz |
| History | Słuchacz |

### graf przepływu



## ssdlDeployed

### Opis

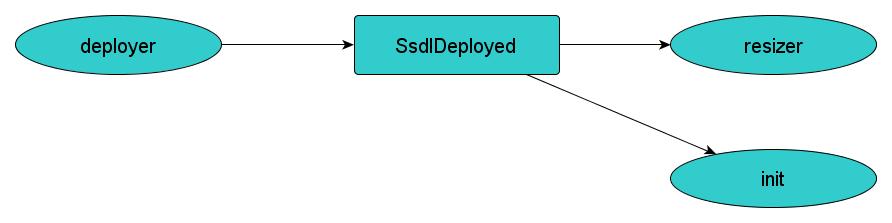
Zdarzenie wykorzystywane do inicializacji modułu resizer, inforumuje o rozmieszczeniu grafu na kanwie.

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| deployer | Wywoływacz |
| resizer | Słuchacz |
| init | Słuchacz |

### graf przepływu



## switchConsole

### Opis

Zdarzenie odpowiadające za pokazywanie widoku konsoli. Skrót klawiszowy zdefiniowany jest w module “specialEvents”

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| logger2 | Słuchacz |

### graf przepływu



## switchToCF

### Opis

Zdarzenie wywoływane w momencie żądania zmiany trybu wyświetlania na tryb przepływu kontroli. Skrót klawiszowy do wywołania zdarzenia zdefiniowany jest w module specialEvents

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| nodeVisualiser | słuchacz |

### graf przepływu



## switchToDF

### Opis

Zdarzenie wywoływane w momencie żądania zmiany trybu wyświetlania na tryb przepływu danych. Skrót klawiszowy do wywołania zdarzenia zdefiniowany jest w module specialEvents

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| nodeVisualiser | słuchacz |

### graf przepływu



## validationError

Zdarzenie wywoływane w momencie, gdy wystąpi błąd podczas walidacji

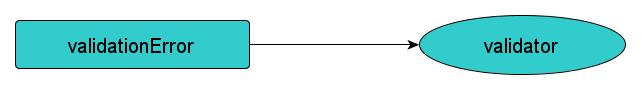
### Opis

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| validator | Słuchacz |

### graf przepływu



## validationFailed

### Opis

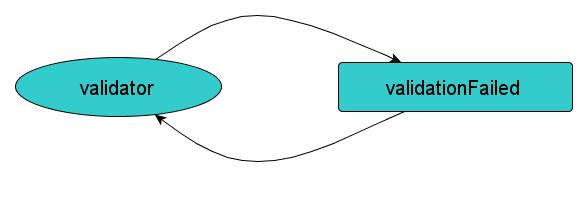
Zdarzenie wywoływane za pomocą Ajax poprzez moduł validator.Wyświetla wiadomości o błędach, które wystąpiły przy walidacji

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| validator | Wywoływacz, słuchacz |

### graf przepływu



## validationPassed

### Opis

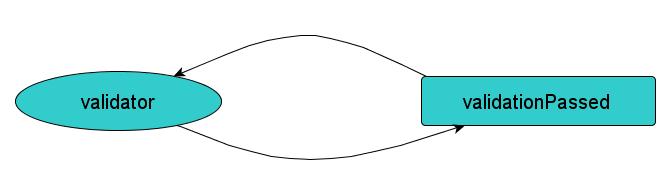
Zdarzenie wywoływane w momencie, gdy walidacja przebiegła pomyźlnie

### Powiązane moduły

**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| validator | Wywoływacz, słuchacz |

### graf przepływu



## validationPerformed

### Opis

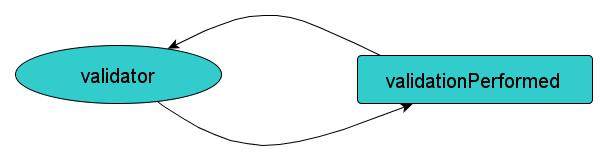
Zdarzenie wywoływane w momencie zakończenia walidacji. Obsługiwane jest przez moduł “validator” , który przetwarza wyniki walidacji.

### Powiązane moduły

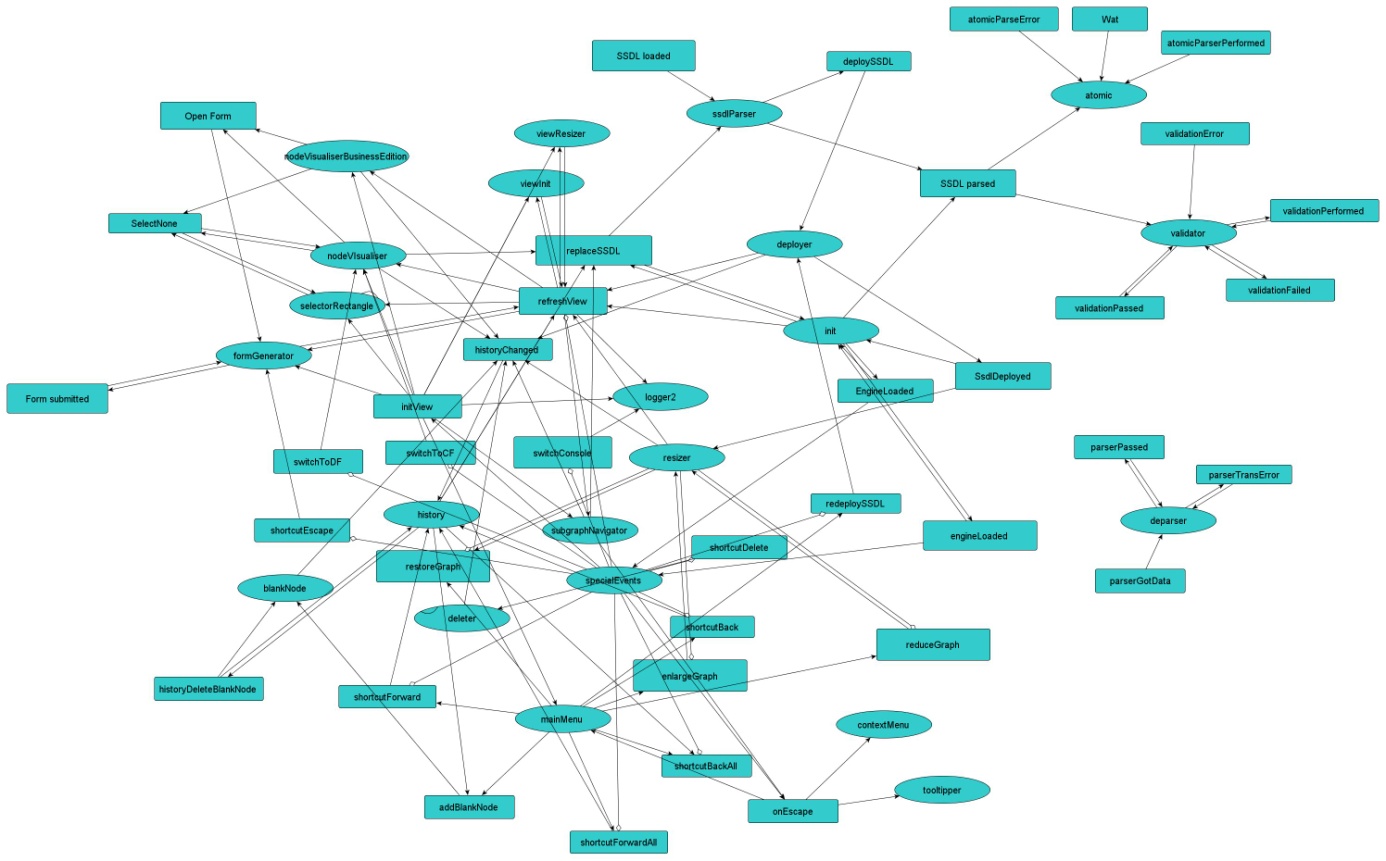
**Zdarzenia powiązane z modułem**

|  |  |
| --- | --- |
| **Moduł** | **Rola** |
| validator | Wywoływacz, słuchacz |

### graf przepływu



## Wszystkie zdarzenia

Poglądowa mapa zdarzeń wygląda następująco:  


Graf zdarzeń w wysokiej rozdzielczośći dostępny jest pod adresem:  
TODO