**Спецификация**

Конечный Автомат

Статус: Черновик

Дата: Ноябрь 8, 2020

Версия: 1.0

|  |  |
| --- | --- |
| **Авторы** | **Компания** |
| Владимир Башев | ПИРФ |
|  |  |

Содержание

[**1.** **Обзор** 4](#_Toc131766119)

[**1.1.** **Введение** 4](#_Toc131766120)

[**1.2.** **Примечание** 4](#_Toc131766121)

[**1.3.** **Ссылки** 4](#_Toc131766122)

[**2.** **Компонент Eco.FSM1** 5](#_Toc131766123)

[**3.** **Интерфейс IEcoFSM1** 6](#_Toc131766124)

[**3.1.** **IEcoFSM1 описание на ECO IDL** 6](#_Toc131766125)

[**3.1.1.** **Функция CreateStateMachine** 6](#_Toc131766126)

[**3.1.2.** **Функция get\_EventProcessing** 6](#_Toc131766127)

[**4.** **Интерфейс IEcoFSM1StateMachine** 7](#_Toc131766128)

[**4.1.** **IEcoFSM1StateMachine описание на ECO IDL** 7](#_Toc131766129)

[**4.1.1.** **Функция get\_Name** 7](#_Toc131766130)

[**4.1.2.** **Функция get\_Id** 7](#_Toc131766131)

[**4.1.3.** **Функция AddState** 8](#_Toc131766132)

[**4.1.4.** **Функция get\_States** 8](#_Toc131766133)

[**4.1.5.** **Функция AddEvent** 8](#_Toc131766134)

[**4.1.6.** **Функция get\_Events** 8](#_Toc131766135)

[**4.1.7.** **Функция AddTransition** 8](#_Toc131766136)

[**4.1.8.** **Функция get\_Transitions** 8](#_Toc131766137)

[**4.1.9.** **Функция AddTransitionById** 9](#_Toc131766138)

[**5.** **Интерфейс IEcoFSM1State** 10](#_Toc131766139)

[**5.1.** **IEcoFSM1State описание на ECO IDL** 10](#_Toc131766140)

[**5.1.1.** **Функция get\_Id** 10](#_Toc131766141)

[**5.1.2.** **Функция get\_Name** 10](#_Toc131766142)

[**5.1.3.** **Функция set\_Name** 11](#_Toc131766143)

[**5.1.4.** **Функция IsInitial** 11](#_Toc131766144)

[**5.1.5.** **Функция set\_Initial** 11](#_Toc131766145)

[**5.1.6.** **Функция IsFinal** 11](#_Toc131766146)

[**5.1.7.** **Функция set\_Final** 11](#_Toc131766147)

[**5.1.8.** **Функция IsTrap** 11](#_Toc131766148)

[**5.1.9.** **Функция set\_Trap** 11](#_Toc131766149)

[**5.1.10.** **Функция get\_Parameter** 12](#_Toc131766150)

[**5.1.11.** **Функция set\_Parameter** 12](#_Toc131766151)

[**6.** **Интерфейс IEcoFSM1Event** 13](#_Toc131766152)

[**6.1.** **IEcoFSM1Event описание на ECO IDL** 13](#_Toc131766153)

[**6.1.1.** **Функция get\_Name** 13](#_Toc131766154)

[**6.1.2.** **Функция get\_Id** 13](#_Toc131766155)

[**6.1.3.** **Функция IsNull** 13](#_Toc131766156)

[**6.1.4.** **Функция set\_Null** 14](#_Toc131766157)

[**6.1.5.** **Функция get\_Parameter** 14](#_Toc131766158)

[**6.1.6.** **Функция set\_Parameter** 14](#_Toc131766159)

[**7.** **Интерфейс IEcoFSM1Transition** 15](#_Toc131766160)

[**7.1.** **IEcoFSM1Transition описание на ECO IDL** 15](#_Toc131766161)

[**7.1.1.** **Функция get\_Name** 15](#_Toc131766162)

[**7.1.2.** **Функция set\_Name** 15](#_Toc131766163)

[**7.1.3.** **Функция get\_Id** 15](#_Toc131766164)

[**7.1.4.** **Функция get\_Parameter** 16](#_Toc131766165)

[**7.1.5.** **Функция set\_Parameter** 16](#_Toc131766166)

[**7.1.6.** **Функция get\_Event** 16](#_Toc131766167)

[**7.1.7.** **Функция get\_Source** 16](#_Toc131766168)

[**7.1.8.** **Функция get\_Target** 16](#_Toc131766169)

[**8.** **Интерфейс IEcoFSM1EventProcessing** 17](#_Toc131766170)

[**8.1.** **IEcoFSM1EventProcessing описание на ECO IDL** 17](#_Toc131766171)

[**8.1.1.** **Функция get\_StateMachine** 17](#_Toc131766172)

[**8.1.2.** **Функция ResetStateMachine** 17](#_Toc131766173)

[**8.1.3.** **Функция get\_CurrentState** 18](#_Toc131766174)

[**8.1.4.** **Функция isInitialState** 18](#_Toc131766175)

[**8.1.5.** **Функция isFinalState** 18](#_Toc131766176)

[**8.1.6.** **Функция InvokeByEvent** 18](#_Toc131766177)

[**8.1.7.** **Функция InvokeById** 18](#_Toc131766178)

[**8.1.8.** **Функция InvokeByName** 18](#_Toc131766179)

[**9.** **В разработке …** 19](#_Toc131766180)

[**Приложение А** 19](#_Toc131766181)

1. **Обзор**

Данный документ описывает требования к реализации компонента Eco.FSM1 (Конечный Автомат).

* 1. **Введение**

Описание.

* 1. **Примечание**
* Ключевые слова в документе
  1. **Ссылки**

Данный параграф содержит ссылки на информацию, помогающую понять данный документ:

[] – наименование ссылки

Доступен по: <http://адрес>

1. **Компонент Eco.FSM1**

Компонент, имеет следующие описание:

1. **Интерфейс IEcoFSM1**
   1. **IEcoFSM1 описание на ECO IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.h" | | | |
| [  object,  uguid(7ACB8256-EEB3-40FA-8E83-9A9AF03464D5),  ] | | | |
| interface IEcoFSM1 : IEcoUnknown { | | | |
|  | | | |
| IEcoFSM1StateMachine\* | | ***CreateStateMachine*** | ([in] char\_t\* Name); |
|  | | |  |
| IEcoFSM1EventProcessing\* | | ***get\_EventProcessing*** | ([in] IEcoFSM1StateMachine\* StateMachine); |
|  | | | |
|  | | | |
| } | |  |  |

* + 1. **Функция CreateStateMachine**

Функция создает экземпляр конечного автомата с заданным именем ***Name*** и возвращает указатель на интерфейс конечного автомата ***IEcoFSM1StateMachine\*.***

* + 1. **Функция get\_EventProcessing**

Функция создает обработчик событий для конечного автомата, переданного как входной параметр ***StateMachine*** и возвращает указатель на интерфейс обработчика событий конечного автомата ***IEcoFSM1EventProcessing\*.***

1. **Интерфейс IEcoFSM1StateMachine**
   1. **IEcoFSM1StateMachine описание на ECO IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.h" | | | |
|  | | | |
| [  object,  uguid(6B73BD3F-22F6-462E-AC29-F5C1214E775E),  ] | | | |
| interface IEcoFSM1StateMachine : IEcoUnknown { | | | |
|  | | | |
| char\_t\* | | ***get\_Name*** | ([in] void); |
|  | | |  |
| int16\_t | | ***get\_Id*** | ([in] void); |
|  | | | |
| IEcoFSM1State\* | | ***AddState*** | ([in] char\_t\* Name); |
|  | | | |
| IEcoList1\* | | ***get\_States*** | ([in] void); |
|  | |  |  |
| IEcoFSM1Event\* | | ***AddEvent*** | ([in] char\_t\* Name, |
|  | |  | [in] int16\_t Id, |
|  | |  | [in] bool\_t isNull); |
|  | | | |
| IEcoList1\* | | ***get\_Events*** | ([in] void); |
|  | |  |  |
| IEcoFSM1Transition\* | | ***AddTransition*** | ([in] IEcoFSM1Event\* Event, |
|  | |  | [in] IEcoFSM1State\* Source, |
|  | |  | [in] IEcoFSM1State\* Target); |
|  | | | |
| IEcoList1\* | | ***get\_Transitions*** | ([in] void); |
|  | |  |  |
| IEcoFSM1Transition\* | | ***AddTransitionById*** | ([in] int16 EventId, |
|  | |  | [in] int16 SourceStateId, |
|  | |  | [in] int16 TargetStateId); |
|  | | | |
| } | |  |  |

* + 1. **Функция get\_Name**

Функция возвращает имя конечного автомата***.***

* + 1. **Функция get\_Id**

Функция возвращает идентификатор конечного автомата***.***

* + 1. **Функция AddState**

Функция добавляет состояние в конечный автомат с заданным именем ***Name*** и возвращает указатель на интерфейс состояния конечного автомата ***IEcoFSM1State\*.***

* + 1. **Функция get\_States**

Функция возвращает список всех состояний конечного автомата в виде указателя на интерфейс для работы со списком ***IEcoList1\*.*** Элементом списка является указатель на интерфейс состояния конечного автомата ***IEcoFSM1State\*.***

* + 1. **Функция AddEvent**

Функция добавляет событие в конечный автомат с заданными именем ***Name***, уникальным идентификатором ***Id***, флагом пустого события ***isNull*** и возвращает указатель на интерфейс события конечного автомата ***IEcoFSM1Event\*.***

Если ***Id*** равен нулю, то идентификатор события назначается автоматически.

Если ***Id*** не равен нулю и совпадает с существующим ***Id*** событием в списке, то событие не добавляется в список, возвращая нулевой указатель на интерфейс.

* + 1. **Функция get\_Events**

Функция возвращает список всех событий конечного автомата в виде указателя на интерфейс для работы со списком ***IEcoList1\*.*** Элементом списка является указатель на интерфейс события конечного автомата ***IEcoFSM1Event\*.***

* + 1. **Функция AddTransition**

Функция добавляет переход в конечный автомат с заданными событием ***Event***, исходным состоянием ***Source***, конечным состоянием ***Target*** и возвращает указатель на интерфейс перехода конечного автомата ***IEcoFSM1Transition\*.***

* + 1. **Функция get\_Transitions**

Функция возвращает список всех переходов конечного автомата в виде указателя на интерфейс для работы со списком ***IEcoList1\*.*** Элементом списка является указатель на интерфейс перехода конечного автомата ***IEcoFSM1Transition\*.***

* + 1. **Функция AddTransitionById**

Функция добавляет переход в конечный автомат с заданными идентификаторами события ***EventId***, исходного состояния ***SourceId***, конечного состояния ***TargetId*** и возвращает указатель на интерфейс перехода конечного автомата ***IEcoFSM1Transition\*.***

1. **Интерфейс IEcoFSM1State**
   1. **IEcoFSM1State описание на ECO IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.h" | | | |
|  | | | |
| [  object,  uguid(6242C2A9-31D9-4FB9-A1CB-8AC693AD6EB4),  ] | | | |
| interface IEcoFSM1State : IEcoUnknown { | | | |
|  | | | |
| int16\_t | | ***get\_Id*** | ([in] void); |
|  | |  |  |
| char\_t\* | | ***get\_Name*** | ([in] void); |
|  | | |  |
| void | | ***set\_Name*** | ([in] char\_t\* Name); |
|  | | | |
| bool\_t | | ***IsInitial*** | ([in] void); |
|  | |  |  |
| void | | ***set\_Initial*** | ([in] bool\_t isInitial); |
|  | | | |
| bool\_t | | ***IsFinal*** | ([in] void); |
|  | |  |  |
| void | | ***set\_Final*** | ([in] bool\_t isFinal); |
|  | |  |  |
| bool\_t | | ***IsTrap*** | ([in] void); |
|  | |  |  |
| void | | ***set\_Trap*** | ([in] bool\_t isTrap); |
|  | |  |  |
| void\* | | ***get\_Parameter*** | ([in] void); |
|  | |  |  |
| void | | ***set\_Parameter*** | ([in] void\* value); |
|  | | | |
| } | |  |  |

* + 1. **Функция get\_Id**

Функция возвращает идентификатор состояния конечного автомата***.***

* + 1. **Функция get\_Name**

Функция возвращает имя состояния конечного автомата***.***

* + 1. **Функция set\_Name**

Функция устанавливает(назначает) имя состояния конечного автомата***.***

* + 1. **Функция IsInitial**

Функция проверяет состояние конечного автомата и возвращает значение равное единице «1» в случае, если состояние является начальным. В остальных случаях возвращает ноль «0»***.***

* + 1. **Функция set\_Initial**

Функция устанавливает(задает) значение начального состояния конечного автомата в случае, если значение равно единице «1»***.*** В остальных случаях, принимает значение равное ноль «0».

* + 1. **Функция IsFinal**

Функция проверяет состояние конечного автомата и возвращает значение равное единице «1» в случае если состояние является конечным, в остальных случаях возвращает ноль «0»***.***

* + 1. **Функция set\_Final**

Функция устанавливает(задает) значение конечного состояния конечного автомата в случае, если значение равно единице «1»***.*** В остальных случаях, принимает значение равное ноль «0».

* + 1. **Функция IsTrap**

Функция проверяет состояние конечного автомата и возвращает значение равное единице «1» в случае если состояние является ловушкой, в остальных случаях возвращает ноль «0»***.***

* + 1. **Функция set\_Trap**

Функция устанавливает(задает) значение ловушки состояния конечного автомата в случае, если значение равно единице «1»***.*** В остальных случаях, принимает значение равное ноль «0».

* + 1. **Функция get\_Parameter**

Функция возвращает указатель на параметры пользователя***.***

* + 1. **Функция set\_Parameter**

Функция устанавливает(сохраняет) указатель на параметры пользователя.

1. **Интерфейс IEcoFSM1Event**
   1. **IEcoFSM1Event описание на ECO IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.h" | | | |
|  | | | |
| [  object,  uguid(1BFBB309-B33F-4E86-AABE-0C41ED5E256C),  ] | | | |
| interface IEcoFSM1Event : IEcoUnknown { | | | |
|  | | | |
| char\_t\* | | ***get\_Name*** | ([in] void); |
|  | | |  |
| int16\_t | | ***get\_Id*** | ([in] void); |
|  | |  |  |
| bool\_t | | ***IsNull*** | ([in] void); |
|  | |  |  |
| void | | ***set\_Null*** | ([in] bool\_t isNull); |
|  | |  |  |
| void\* | | ***get\_Parameter*** | ([in] void); |
|  | |  |  |
| void | | ***set\_Parameter*** | ([in] void\* value); |
|  | | | |
| } | |  |  |

* + 1. **Функция get\_Name**

Функция возвращает имя события конечного автомата***.***

* + 1. **Функция get\_Id**

Функция возвращает идентификатор события конечного автомата***.***

* + 1. **Функция IsNull**

Функция проверяет событие конечного автомата и возвращает значение равное единице «1» в случае если событие является пустым, в остальных случаях возвращает ноль «0»***.***

* + 1. **Функция set\_Null**

Функция устанавливает(задает) пустое значение события конечного автомата в случае, если значение равно единице «1»***.*** В остальных случаях, принимает значение равное ноль «0».

* + 1. **Функция get\_Parameter**

Функция возвращает указатель на параметры пользователя***.***

* + 1. **Функция set\_Parameter**

Функция устанавливает(сохраняет) указатель на параметры пользователя.

1. **Интерфейс IEcoFSM1Transition**
   1. **IEcoFSM1Transition описание на ECO IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.h" | | | |
|  | | | |
| [  object,  uguid(00C3032A-56F2-4474-B4FD-6F837D907909),  ] | | | |
| interface IEcoFSM1Transition : IEcoUnknown { | | | |
|  | | | |
| char\_t\* | | ***get\_Name*** | ([in] void); |
|  | |  |  |
| void | | ***set\_Name*** | ([in] char\_t\* Name); |
|  | | |  |
| int16\_t | | ***get\_Id*** | ([in] void); |
|  | |  |  |
| void\* | | ***get\_Parameter*** | ([in] void); |
|  | |  |  |
| void | | ***set\_Parameter*** | ([in] void\* value); |
|  | | | |
| IEcoFSM1Event\* | | ***get\_Event*** | ([in] void); |
|  | |  |  |
| IEcoFSM1State\* | | ***get\_Source*** | ([in] void); |
|  | |  |  |
| IEcoFSM1State\* | | ***get\_Target*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **Функция get\_Name**

Функция возвращает имя перехода конечного автомата***.***

* + 1. **Функция set\_Name**

Функция устанавливает(назначает) имя перехода конечного автомата***.***

* + 1. **Функция get\_Id**

Функция возвращает идентификатор перехода конечного автомата***.***

* + 1. **Функция get\_Parameter**

Функция возвращает указатель на параметры пользователя***.***

* + 1. **Функция set\_Parameter**

Функция устанавливает(сохраняет) указатель на параметры пользователя.

* + 1. **Функция get\_Event**

Функция возвращает указатель на интерфейс события конечного автомата ***IEcoFSM1Event\*.***

* + 1. **Функция get\_Source**

Функция возвращает указатель на интерфейс исходного состояния конечного автомата ***IEcoFSM1State\*.***

* + 1. **Функция get\_Target**

Функция возвращает указатель на интерфейс конечного состояния конечного автомата ***IEcoFSM1State\*.***

1. **Интерфейс IEcoFSM1EventProcessing**
   1. **IEcoFSM1EventProcessing описание на ECO IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.h" | | | |
|  | | | |
| [  object,  uguid(A768CE5B-8B4C-4873-80F9-8FCC5365A855),  ] | | | |
| interface IEcoFSM1EventProcessing : IEcoUnknown { | | | |
|  | | | |
| IEcoFSM1StateMachine\* | | ***get\_StateMachine*** | ([in] void); |
|  | | |  |
| IEcoFSM1State\* | | ***ResetStateMachine*** | ([in] void); |
|  | | | |
| IEcoFSM1State\* | | ***get\_CurrentState*** | ([in] void); |
|  | | | |
| IEcoFSM1State\* | | ***IsInitialState*** | ([in] void); |
|  | |  |  |
| IEcoFSM1State\* | | ***IsFinalState*** | ([in] void); |
|  | |  |  |
| IEcoFSM1State\* | | ***InvokeByEvent*** | ([in] IEcoFSM1Event\* Event); |
|  | | | |
| IEcoFSM1State\* | | ***InvokeById*** | ([in] int16\_t Id); |
|  | |  |  |
| IEcoFSM1State\* | | ***InvokeByName*** | ([in] char\_t\* Name); |
|  | | | |
| } | |  |  |

* + 1. **Функция get\_StateMachine**

Функция возвращает указатель на интерфейс конечного автомата ***IEcoFSM1StateMachine\*.***

* + 1. **Функция ResetStateMachine**

Функция сбрасывает конечный автомат в начальное состояние и возвращает указатель на интерфейс начального состояния конечного автомата ***IEcoFSM1State\*.***

* + 1. **Функция get\_CurrentState**

Функция возвращает указатель на интерфейс текущего состояния конечного автомата ***IEcoFSM1State\*.***

* + 1. **Функция isInitialState**

Функция проверяет текущее состояние конечного автомата и возвращает указатель на интерфейс состояния конечного автомата ***IEcoFSM1State\**** в случае если состояние является начальным, в остальных случаях возвращает нуль «null»***.***

* + 1. **Функция isFinalState**

Функция проверяет текущее состояние конечного автомата и возвращает указатель на интерфейс состояния конечного автомата ***IEcoFSM1State\**** в случае если состояние является конечным, в остальных случаях возвращает ноль «null»***.***

* + 1. **Функция InvokeByEvent**

Функция обрабатывает событие по заданному указателю на интерфейс события конечного автомата ***Event*** и возвращает указатель на интерфейс состояния конечного автомата ***IEcoFSM1State\**** после обработки***.***

* + 1. **Функция InvokeById**

Функция обрабатывает событие по заданному идентификатору события конечного автомата ***Id*** и возвращает указатель на интерфейс состояния конечного автомата ***IEcoFSM1State\**** после обработки***.***

* + 1. **Функция InvokeByName**

Функция обрабатывает событие по заданному имени события конечного автомата ***Name*** и возвращает указатель на интерфейс состояния конечного автомата ***IEcoFSM1State\**** после обработки***.***

1. **В разработке …**

# **Приложение А**