

1. Язык схем для VWML

Мы рассмотрим язык

ShL

(Scheme Language), который будем называть языком схем проектирования VWML-кода.

Этот язык является основой для разработки среды проектирования VWML-кода.

Прежде всего, отметим, что **ShL** включает в себя **VWML**.

Конструкции **ShL** “объемлют” соответствующие конструкции **VWML**.

Мы этот факт запишем так:

$$\text{VWML} \subset \text{ShL}$$

Основным понятием **ShL** является понятие

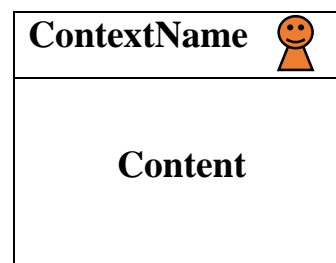
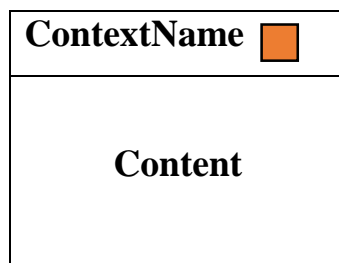
фрейма-контекста.

Такие фреймы подразделяются на два типа (см. рис.1).

а) фрейм-контекст типа 

б) фрейм-контекст типа 

Первый из них не содержит LifetErm'a, а второй – содержит.



Замечание:

Везде жирный шрифт используется для описания **ShL**, т.е. принадлежит метаязыку языка **ShL**.

Отметим, что все конструкции **ShL** “погружены” в некоторый фрейм-контекст. Фрейм-контексты также погружаются в фрейм-контекст, кроме “самых внешних” фреймов.

2. Классификация термов языка VWML

2.1. Множество термов VWML, как обычно, будем обозначать через **T**.

T – множество термов.

Имея в виду обозначения произвольных термов будем писать:

$$\mathbf{t}, \mathbf{t}_1, \mathbf{t}_2, \dots$$

φ-интерпретация терма есть терм:

$$\mathbf{t}_1^\phi = \mathbf{t}_2.$$

2.2. Сущности.

Сущности образуют подкласс **E** класса **T**.

Говоря о сущностях будем пользоваться обозначениями:

$$\mathbf{e}, \mathbf{e}_1, \mathbf{e}_2, \dots$$

Итак, $\mathbf{E} \subset \mathbf{T}$ (от Entity)

$$\mathbf{e}^\phi = \mathbf{e}.$$

2.3. Логические термы.

Класс логических термов **L** также является подклассом класса **T**:

$$\mathbf{L} \subset \mathbf{T} \quad (\text{от Logic}).$$

Говоря о логических термах будем пользоваться обозначениями:

$$\mathbf{l}, \mathbf{l}_1, \mathbf{l}_2, \dots$$

По определению:

$$\mathbf{l}^\varphi = \text{true} \mid \text{false}$$

2.4. Термы, описывающие события.

Класс термов событий является подклассом класса **T**.

Обозначать этот класс будем **Ev**.

$$\mathbf{Ev} \subset \mathbf{T} \quad (\text{от Event}).$$

Говоря о таких термах, будем пользоваться обозначениями:

$$\mathbf{ev}, \mathbf{ev}_1, \mathbf{ev}_2, \dots$$

$$\mathbf{ev}^\varphi = *;$$

Или точнее:

$$\mathcal{M}, \mathbf{ev}^\varphi = \mathcal{M}_1, *$$

где \mathcal{M}_1 рассматривается нами, как модификация \mathcal{M} .

2.5. Термы, описывающие термы, которые интерпретируются как сущности.

Речь идёт о подклассе **IE** класса **T**.

$$\mathbf{IE} \subset \mathbf{T} \quad (\text{Interpret to E}).$$

Говоря о термах этого класса будем пользоваться обозначениями:

$$\mathbf{ie}, \mathbf{ie}_1, \mathbf{ie}_2, \dots$$

По определению:

$$\mathbf{IE}^\Phi \subset \mathbf{E}$$

т.е.

$$\mathbf{ie}^\Phi = \mathbf{e} .$$

Очевидно, что:

$$\mathbf{E} \subset \mathbf{EI} .$$

2.6. Термы, интерпретация которых суть термы описывающие события.

Речь идёт о классе \mathbf{IEv} , $\mathbf{IEv} \subset \mathbf{T}$.

Говоря о термах этого класса будем пользоваться обозначениями:

$$\mathbf{iev} , \mathbf{iev}_1 , \mathbf{iev}_2 , \dots$$

$$\mathbf{IEv}^\Phi \subset \mathbf{Ev} ; \mathbf{iev}^\Phi = \mathbf{ev} .$$

Таким образом,

$$\mathcal{M} , \mathbf{iev}^{\Phi\Phi} = \mathcal{M}_1 , * .$$

2.7. Термы, интерпретация которых суть логический терм.

Речь идёт о подклассе \mathbf{IL} класса \mathbf{T} ; $\mathbf{IL} \subset \mathbf{T}$.

По определению:

$$\mathbf{IL}^\Phi \subset \mathbf{L} .$$

Если мы говорим о термах этого класса, то будем пользоваться обозначениями:

$$\mathbf{il} , \mathbf{il}_1 , \mathbf{il}_2 , \dots$$

Таким образом:

$$\mathbf{il}^\Phi = \mathbf{l}$$

Но тогда:

$$\mathbf{il}^{\Phi\Phi} = \text{true} \mid \text{false}$$

т.е.

$$IL^{\Phi\Phi} \in \{ \text{true}, \text{false} \}.$$

2.8. Термы высших порядков.

Это классы:

$$I^2E, I^3E, \dots$$

$$I^2L, I^3L, \dots$$

$$I^2Ev, I^3Ev, \dots$$

Эти термы легко определяются по индукции.

Например,

$$I^2E^{\Phi} \subset IE$$

$$I^3E^{\Phi} \subset I^2E \text{ и т.д.}$$

Отметим, что рассмотренная нами классификация термов не является полной.

В то же время, это множество термов, которое достаточно для наших задач описания виртуальных миров.

3. Интерпретационные выражения VWML

Мы говорили о том, что контекст описывается интерпретационными выражениями.

Интерпретационное выражение имеет вид:


$$e \text{ ias } t, \text{ где } e \in E, t \in T.$$

При этом:


$$e^\varphi = e, e \downarrow^\varphi = t^\varphi.$$

Если рассматривать “ \downarrow ” как операцию над термами в контексте \mathcal{M} , то

$$\mathcal{M}, e \downarrow^\varphi = \mathcal{M}, e^{\downarrow^\varphi} = \mathcal{M}, t^\varphi.$$

Если контекст является контекстом типа , то в нём обязательно присутствие LifeTerm’a.

LifeTerm является специальной выделенной сущностью.

В каждом контексте типа  обязательно присутствие LifeTerm-выражения, которое имеет вид:

$$\text{LifeTerm} = ev \quad (*)$$

Здесь LifeTerm – зарезервированная сущность.

$$ev \in Ev - \text{терм типа “событие”}.$$

т.е.

$$\mathcal{M}, ev^\varphi = \mathcal{M}_1, *.$$

Замечание:

Такова конструкция LifeTerm’a в VWML. На ShL эта конструкция обобщается.

4. Схемы и термы языка ShL

В этом коротком разделе мы лишь начнём говорить о схемах.

К различным схемам мы будем обращаться по их именам, которые входят в **ShL**-конструкции.

Будем пользоваться выражением вида:

[**name**] .

Здесь надо иметь в виду, что квадратные скобки принадлежат языку **ShL** и указывают на то, что речь идёт именно о схеме.

name синтаксически представляет собой простую сущность VWML.

Мы интерпретировали LifeTerm'ы и другие термы языка VWML, обозначая оператор интерпретирования φ .

Мы будем говорить также и об ψ -интерпретировании и об ψ -интерпретаторе. ψ -интерпретатор, в частности применяется к схемам языка **ShL**.

Результат интерпретации будем записывать так:

[**name**] $^{\psi}$

Результатом такой интерпретации являются “две вещи”:

- а) терм языка VWML;
- б) совокупность некоторых интерпретационных выражений языка VWML;
т.е. фрагмент VWML-кода.

Мы часто будем иметь в виду только терм, как результат ψ -интерпретирования, имея в виду, что фрагмент кода мы получаем “на побочном эффекте”.

5. Термы языка ShL

Введем теперь понятие **ShL-терма** языка **ShL**. Обозначим множество таких термов **ShT**.

Отметим, что как простые, так и составные термы VWML (они образуют совокупность термов **T**) являются также термами языка **ShL**.

Иными словами,

$$\mathbf{T} \subset \mathbf{ShT} .$$

Введём сначала понятие простого ShL-терма. Такой терм имеет вид:

$$\mathbf{Class_} < \mathbf{name} > \text{ или } < \mathbf{name} >$$

Class является именем класса термов языка VWML

(однако теперь это символы ShL, и будут записываться в этом документе, не жирным шрифтом).

name – имя простого ShL-терма.

Приведём примеры:

$$E_ < abc > , Ev_ < SomeEvent > , L_ < expr > , \dots$$

Семантика этих термов будет подробно описана ниже. Сейчас же отметим, что, например:

$$E_ < abc >^\psi \in \mathbf{E} ,$$

$$Ev_ < SomeEvent >^\psi \in \mathbf{Ev} ,$$

$$L_ < expr >^\psi \in \mathbf{L} , \dots$$

Таким образом, ψ -интерпретация простого ShL-терма – терм языка VWML.

Введём, наконец, понятие термина (не обязательно простого) языка **ShL**.

Такой терм конструируется так же как и терм языка **VWML**, однако в качестве дополнительного “строительного материала” используются также схемы (конструкции вида $[\textbf{name}]$) и простые ShL-термы (конструкции вида **Class_< name >**).

Приведём примеры:

1. $a \downarrow \text{exe}$
2. $(\text{ev_} \langle \text{event1} \rangle \ (a \ b) \uparrow \ \text{ev_} \langle \text{event2} \rangle)$
3. $([\text{name1}] \ \text{ev_} \langle \text{event} \rangle)$
4. $(\text{Case} \ L_ \langle \text{expr} \rangle) \downarrow \text{exe}$
5. $([\text{name1}] \ [\text{name2}])$

ψ -интерпретация ShL-терма даёт терм языка **VWML**, т.е.

$$\mathbf{ShT}^\psi \subset \mathbf{T}$$

при этом “обычные” термы являются неподвижными точками этого ψ -оператора:

$$\begin{aligned} \mathbf{T}^\psi &= \mathbf{T} , \\ \mathbf{t} \in \mathbf{T} , \text{ тогда } \mathbf{t}^\psi &= \mathbf{t} . \end{aligned}$$

Замечание:

Если ShL-терм содержит в своей записи схемы, то результатом его интерпретации дополнительно также могут являться и интерпретационные выражения **VWML**.

6. Подстановочные выражения

Введём понятие подстановочного выражения языка ShL.

Такое выражение имеет вид:

Простой ShL-терм sub ShL-терм .

Приведём примеры:

$E_{< entity >} \text{sub} (E_{< entity_1 >} a \downarrow) \text{join}$

$Ev_{< event1 >} \text{sub} (a \ b) \uparrow$

$Ev_{< event >} \text{sub} ([\text{transfer}] \ Ev_{< put >})$

$L_{< expr >} \text{sub} [\text{condition}]$

На множестве простых термов ShL можем естественным образом ввести операцию подстановки sub, например:

$E_{< entity >}^{\text{sub}} = (E_{< entity_1 >} a \downarrow) \text{join}$

$Ev_{< event1 >}^{\text{sub}} = (a \ b) \uparrow$

Замечание:

Операцию sub можно рассматривать как аналог операции “ \downarrow ” на сущностях VWML.

7. Фреймы простых термов

Class_< term >
ShL - term

Это просто краткая фреймовая запись подстановочного выражения.

Саму рамку можно рассматривать, как знак “**sub**”.

8. Интерпретационные выражения ShL-языка

Мы рассматривали интерпретационные выражения как конструкции языка VWML. Теперь мы расширим совокупность интерпретационных выражений. Мы будем рассматривать интерпретационные выражения **ShL**-языка.

В частности, интерпретационные выражения VWML являются интерпретационными выражениями **ShL**.

Будем обозначать:

IV – множество интерпретационных выражений языка **VWML**;

IS – множество интерпретационных выражения языка **ShL**.

В силу сказанного выше:

$$\mathbf{IV} \subset \mathbf{IS} .$$

Говоря об интерпретационных выражениях будем пользоваться обозначениями:

ivexpr , **ivexpr1** , **ivexpr2** , ...

isexpr , **isexpr1** , **isexpr2** , ...

Интерпретационное ShL-выражение имеет вид:

ShL_term1 ias **ShL_term2** .

Как мы уже отмечали:

$$\mathbf{ShL_term}^{\Psi} \subset \mathbf{T} .$$

Для левой части интерпретационного ShL-выражения потребуем, чтобы:

$$\mathbf{ShL_term1}^{\Psi} \subset \mathbf{E} .$$

Т.е. результат ψ -интерпретации здесь является сущностью языка VWML.

Определим теперь ψ для интерпретационных выражений.

Пусть,

$$\mathbf{isexpr} = \mathbf{ShL_term1} \ \underline{\mathbf{ias}} \ \mathbf{ShL_term2}$$

Тогда:

$$\mathbf{isexpr}^\psi = \mathbf{ShL_term1}^\psi \ \underline{\mathbf{ias}} \ \mathbf{ShL_term2}^\psi$$

При этом, мы полагаем, что:

$$\mathbf{ivexpr}^\psi = \mathbf{ivexpr} \ .$$

Т.е. интерпретационные выражения языка VWML являются неподвижными точками интерпретатора ψ .

9. Фреймы схем

Введём сначала понятие схемного выражения языка **ShL**.

Это выражение имеет вид:

$$[\text{Sh_name}] = \text{ShL-term} , \\ \{ \text{isexpr1} , \text{isexpr2} , \dots \} ;$$

При этом,

$[\text{Sh_name}]$ – имя схемы (часто будем говорить просто схема);

ShL-term – терм схемы;

$\{ \text{isexpr1} , \text{isexpr2} , \dots \}$ – тело (или окрестность) схемы.

Имея в виду ψ -интерпретацию схемы, будем писать:

$$[\text{Sh_name}]^{\psi}$$

При этом, полагаем, что:

$$[\text{Sh_name}]^{\psi} = \text{ShL-term}^{\psi} , \\ \{ \text{isexpr1}^{\psi} , \text{isexpr2}^{\psi} , \dots \} .$$

Приведём пример:

Пусть,

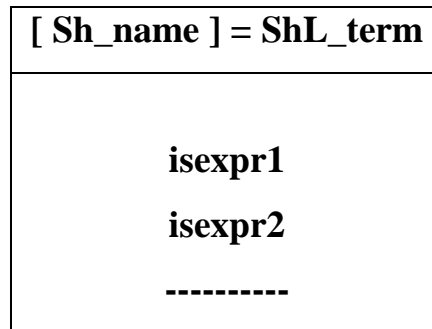
$$[\text{Sh_example}] = (a \ E_{< \text{entity} >}) \downarrow \text{exe} \\ \{ b \ \underline{\text{ias}} \ Ev_{< \text{event} >} \}$$

Тогда,

$$[\text{Sh_example}]^{\psi} = (a \ E_{< \text{entity} >}^{\psi}) \downarrow \text{exe} \\ \{ b \ \underline{\text{ias}} \ Ev_{< \text{event} >}^{\psi} \}$$

Точно оператор ψ будет определён в следующем разделе.

Схемное выражение может быть представлено в виде фрейма-схемы.



“Верхнюю часть” фрейма будем называть заголовком фрейма-схемы.

Каждый фрейм-схема погружается в фрейм-контекст.

10. Ψ -интерпретация

Мы здесь рассмотрим понятие Ψ -интерпретации термов, подстановочных выражений, интерпретационных выражений и схемных выражений языка ShL.

Ψ -интерпретация определяется рекурсивно следующими правилами.

1. Ψ -интерпретация простых ShL-термов.

$$\mathbf{Class_< term >^{\Psi} = Class_< term >^{\text{sub } \Psi}}$$

2. Пусть в ShL-терм конструируется из элементов языка VWML, а также с помощью:

item1, item2, ... , item_n, ...

которые являются либо именами схем, либо простыми ShL-термами, т.е. либо имеют вид:

[name]

либо:

Class_< term > .

Сказанное выше запишем так:

ShL-term = Expr (item1, item2, ... , item_n) .

Но тогда:

$$\mathbf{ShL-term^{\Psi} = Expr (item1^{\Psi}, ... , item_n^{\Psi})}$$

Например,

Пусть $\mathbf{ShL_term} = (a \downarrow [b] E_{<c>}) \downarrow exe$.

Тогда $\mathbf{ShL_term}^\Psi = (a \downarrow [b]^\Psi E_{<c>^\Psi}) \downarrow exe$.

3. ψ -интерпретация интерпретационных ShL-выражений.

Пусть дано интерпретационное выражение:

$$\mathbf{isexpr} = \mathbf{ShL_term1} \ \underline{ias} \ \mathbf{ShL_term2}$$

Тогда по определению:

$$\mathbf{isexpr}^\Psi = \mathbf{ShL_term1}^\Psi \ \underline{ias} \ \mathbf{ShL_term2}^\Psi$$

4. ψ -интерпретация схемы.

Пусть дана схема:

$$\begin{aligned} [\mathbf{Sh_name}] &= \mathbf{ShL_term} , \\ \{ \mathbf{isexpr1}, \mathbf{isexpr2}, \dots \} \end{aligned}$$

Тогда по определению:

$$\begin{aligned} [\mathbf{Sh_name}]^\Psi &= \mathbf{ShL_term}^\Psi , \\ \{ \mathbf{isexpr1}^\Psi, \mathbf{isexpr2}^\Psi, \dots \} \end{aligned}$$

5. Терминальные ψ -интерпретации

Пусть $\mathbf{t} \in \mathbf{T}$, т.е. \mathbf{t} – терм языка VWML.

Тогда $\mathbf{t}^\Psi = \mathbf{t}$.


Отсюда следует, что если \mathbf{ivexpr} – интерпретационное выражение языка VWML, то

$$\mathbf{ivexpr}^\Psi = \mathbf{ivexpr} .$$

11. Интерпретация контекста

Мы говорили, что все конструкции ShL-языка погружены в некоторый контекст. Контекст также может погружаться в контекст.

Сейчас мы лишь отметим, что ψ -интерпретация контекста сводится к ψ -интерпретациям погруженных в него объектов. Точные определения дадим позднее.

Отметим лишь, что контекст типа  содержит LifeTerm. На языке ShL мы имеем в таком контексте специальное выражение:

$$\text{LifeTerm} = \text{ShL_Term}$$

ψ -интерпретация этого выражения даёт:

$$\text{LifeTerm} = \text{ShL_Term}^\psi .$$

Здесь, LifeTerm – сущность VWML.