

УДК 681.327.12.001.362

МЕТОДЫ ПОСТРОЕНИЯ И АНАЛИЗА ДЕСКРИПТОРА ДЛЯ СИСТЕМ МОНИТОРИНГА МОБИЛЬНЫХ ГЕТЕРОГЕННЫХ ОБЪЕКТОВ

А.И. КУЗЬМИЧ*, В.В. КРАСНОПРОШИН, А.Н. ВАЛЬВАЧЕВ

**Белорусский государственный университет информатики и радиоэлектроники
П. Бровка, 6, Минск, 220013, Беларусь*

*Белорусский государственный университет
Независимости, 4, Минск, 220050, Беларусь*

Поступила в редакцию 5 мая 2014

Предложены стандарты, методы построения и анализа дескриптора для мобильных гетерогенных объектов. Представлен пример их использования для сложных объектов с изменяемой структурой. В целом, полученные результаты позволяют сократить время формирования объектов и исключают ошибки в программах мониторинга.

Ключевые слова: мобильные гетерогенные объекты, дескриптор, технические объекты с изменяемой структурой, мониторинг.

Введение

В результате быстрой эволюции промышленных технологий, компьютеров, коммуникаций и искусственного интеллекта появилось множество новых материальных и виртуальных мобильных объектов, состоящих из разнородных компонентов. К первой группе относятся скоростные поезда, безэкипажные корабли, беспилотные самолеты и т.д. [1]. Наиболее сложными из них считаются объекты с изменяющейся структурой и функционирующие под управлением систем с элементами искусственного интеллекта. Во вторую группу входят элементы компьютерной визуализации технологических процессов, экземпляры классов в тренажерах для летчиков, машинистов, диспетчеров, а также составляющие динамических адаптируемых интерфейсов. Здесь наиболее сложными являются тренажеры для пилотов и водителей, визуально приближенные к реальности [2]. Несмотря на столь большое разнообразие, жизненный цикл (ЖЦ) таких объектов включает одни и те же этапы: описание, создание, использование и деструкцию. Поэтому для их изучения целесообразно использовать общий абстрактный термин – «мобильные гетерогенные объекты» (МГО) [3–5].

В данной статье рассматриваются актуальные вопросы описания МГО первой группы для реализации проектов, связанных с перемещением грузов. Важнейшими свойствами современных проектов являются оперативность и разнообразие используемых компонентов с быстро изменяющимися параметрами. Лица, формирующие объекты, вынуждены в кратчайший срок изучить каждый новый проект, выбрать соответствующие компоненты, коммуникации, средство управления (человек, компьютерная система или тандем) и сформировать из них структуру, обеспечивающую достижение цели проекта. Далее результат передается программистам, которые формализуют и используют его в программах мониторинга МГО. В первичном описании могут встречаться ошибки, которые переносятся в программы и могут привести к деструкции МГО. В литературе методы описания МГО крайне специфичны и труднореализуемы [1, 2, 5, 6]. Поэтому актуальным является вопрос построения простого универсального метода описания МГО с использованием интуитивно понятных стандартов, ориентированного на автоматический анализ элементов описания и их применение

в программах поддержки ЖЦ МГО. Предлагается комплексный вариант этих задач, допускающий реализацию на современных языках программирования.

Постановка задачи

Пусть имеется компания, реализующая проекты (proj) с применением МГО (mgo1) в некоторой среде (env). МГО может состоять из компонентов различного типа (например, T200, T201, T202). В сцене реализации ЖЦ МГО участвуют акторы: менеджер (инициирующий проект, С) и, возможно, эксперты; диспетчер (контролирующий траекторию движения МГО, D); исполнитель, непосредственно управляющий МГО и обладающий естественным (nI) или искусственным (aI) интеллектом; возможно, другие МГО (mgo2). Менеджер формирует описание структуры и состава МГО в соответствии с требованиями каждого нового проекта и оценивает сложность МГО на основе собственного опыта и предпочтений, при этом возможны ошибки в типах и реквизитах компонентов. Описание передается программистам, которые формализуют и декомпозируют описание на фрагменты, используемы в различных программах сопровождения ЖЦ МГО, при этом также возможны ошибки с серьезными последствиями. Соответствующая сцена представлена на рис. 1, где символами i1–i5 выделены естественные (визуальные) и технические каналы связи для поддержки ЖЦ МГО. Канал i3 необходим в случае управления МГО диспетчером или менеджером через спутник.

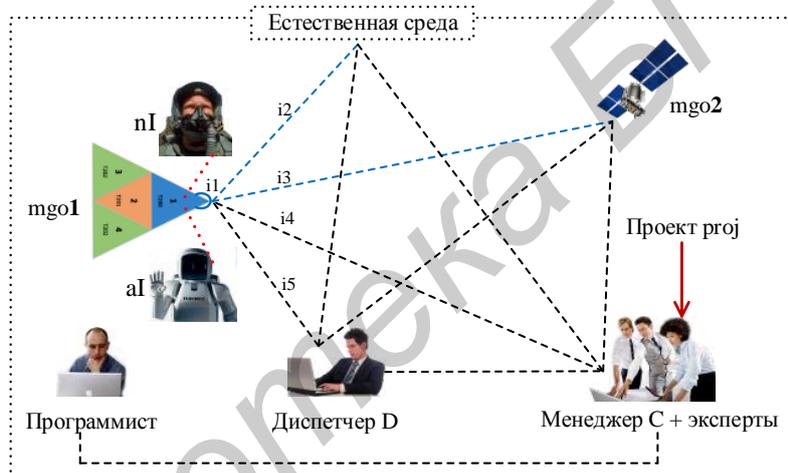


Рис. 1. Сцена реализации ЖЦ МГО

Требуется разработать стандарты для представления компонентов, формат описания (дескриптор) МГО, механизм его декомпозиции, исключающие возможность ошибки.

Основное требование к решению заключается в обеспечении возможности реализации результата на любом современном языке программирования.

Стандарт описания компонентов

Основная проблема построения стандартов заключается в том, что они должны включать все новшества и быть понятны персоналу, который участвует в ЖЦ МГО. Персонал может иметь различный уровень подготовки и говорить на разных языках. Чем выше уровень взаимопонимания, тем ниже вероятность катастроф.

Для выявления значимых сущностей МГО были проанализированы инновационные проекты: «интеллектуального» грузовика Future Truck 2025 (Mercedes-Benz, Daimler team), автомобиля без водителя «Self-Driving Car» (Google), безэкипажных кораблей Self-Driving Ships (Rolls-Royce, Blue Ocean team), скоростного поезда CRH380 High-Speed Train (Hong Kong Polytechnic University). В результате определено, что важнейшей сущностью МГО является тип компонентов, т.к. все задачи МГО решаются с учетом его свойств. На основе статистического анализа определены элементы типа, которые присутствуют практически во всех указанных выше проектах. Соответствующий стандарт можно представить кортежем:

$$Tcomp = (T, Id, Dt, kDt, X, Gt, Ng, I, com, tr), \quad (1)$$

где T – тип компонента; Id – идентификатор; Dt – датчики, фиксирующие значения диагностических параметров; kDt – количество датчиков; X – диагностические параметры; Gt – графический образ типа (jpg-файл); Ng – назначение; I – тип интеллекта, применяемого для управления; com – коммуникации для обмена информацией; tr – траектория движения.

Для практического применения (1) необходимо классифицировать элементы стандарта. Датчики в зависимости от пользователей классифицируются на две группы: $DtnI$ – для биологических объектов (людей) и $DtnoI$ – для технических объектов. Интеллект в зависимости от носителя подразделяется на три группы: nI – естественный (носитель – человек); aI – искусственный (носитель – программа); noI – отсутствие интеллекта (носитель – грузовые платформы и т.п.). Коммуникации, используемые акторами для обмена информацией, классифицируются в зависимости от масштаба на четыре группы: локальные ($com1$) для общения в рамках структуры МГО; интернет ($com2$) для общения в рамках глобальной общедоступной инфраструктуры; телефонные ($com3$) для общения по телефонным и специальным каналам; спутниковые ($com4$) для общения по каналам спутников связи. Траектория классифицируется в зависимости от среды на четыре группы: наземная (trE), подземная ($trUE$), водная (trW), воздушная (trA), космическая (trC).

При формировании элементов стандарта необходимо учитывать и согласовывать мнение технических специалистов и психологов. Для построения корпоративного стандарта на основе (1) можно использовать СУБД (MySQL и др.), таблицы (Excel и др.) или сохраненный (в .Net – Serialize) на внешнем носителе как XML-файл и затем прочитанный (Deserialize) экземпляр соответствующего класса [7]. Пример стандартов для описания типов в компании, использующей в бизнес-проектах железнодорожный транспорт, представлен в таблице.

Таблица 1. Стандарт для описания компонентов МГО

Тип (T)	Идентификатор типа компонента (Id)	Датчики (Dt)	Количество датчиков (kDt)	Параметры (X)	Граф. файл (Gf)	Вид	Интеллект (nI/aI)	Траектория (Tr)
$T1$	Локомотив	Dt_{T1} (nI, noI)	kDt_{T1}	X_{T1} (nI, noI)	Lc.jpg		nI/aI_{T1}	Tr_{T1}
$T2$	Цистерна	Dt_{T2} (noI)	kDt_{T2}	X_{T2} (noI)	Vt.jpg		noI	Tr_{T2}
$T3$	Товарный вагон	Dt_{T3} (noI)	kDt_{T3}	X_{T3} (noI)	Vk.jpg		noI	Dt_{T3}

После создания такой базы данных на сервере, все другие программы, включая мониторинг и визуализацию результатов, могут пользоваться актуальной и подтвержденной экспертами информацией. В случае изменения значений полей другие программы будут использовать новый вариант без коррекции кода.

Дескриптор МГО

Под термином дескриптор будем понимать компактное описание состава и структуры МГО, обеспечивающие выделение составляющих для применения в программах поддержки ЖЦ МГО. Основная сложность построения дескриптора заключается в разнообразии типов компонентов, множестве вариантов их отношений, используемых коммуникаций и уровня риска во время движения по траектории. Первый и третий фактор был преодолен введением стандартов (табл. 1). Для преодоления второго фактора классифицируем отношения между компонентами по признаку связности на две группы: физические (F), где акторы жестко связаны физически и перемещаются по одной траектории (например, вагоны в метро); логические (L), где акторы не связаны и могут перемещаться по разным траекториям (например, группа кораблей).

Четвертый фактор – риск (R) – определяет менеджер, используя собственный опыт и знания экспертов. В большинстве случаев достаточно классификации риска на три класса: слабый ($R1$), средний ($R2$), сильный ($R3$). Фактически уровень риска является интегральной характеристикой, отражающей непосредственно риск и уровень неопределенности в процессе реализации ЖЦ МГО.

Используя результаты классификации, сформируем формулу описания МГО:

$$Dmgo = (proj, id, FL, N, \langle i. T. q. \rangle, com, R), \quad (2)$$

где N – общее количество акторов; i – порядковый номер актора; q – количество акторов данного типа; «,» – разделитель, который можно заменить на любой другой.

В качестве примера построим дескриптор для описания железнодорожного состава VRP11.4, сформированного для проекта Сургут, состоящего из локомотива, четырех цистерн и трех товарных платформ, использующих локальную и специальную связь: $mgo = (Сургут, RVP11.4, F, 8, 1, T1.1, 2, T2.4, 3, T4.3, com1, com3, R1)$

Для практического применения дескриптора необходимо разработать механизм разбора строки на фрагменты, которые будут использоваться в других программах.

Алгоритм анализа дескриптора

Для алгоритма анализа строки дескриптора используем классическую идею разбора последовательности символов по некоторому признаку. В данном случае в качестве признака будем использовать символ «,». Ниже представлен краткий (без try-catch-проверки корректности входной информации) консольный вариант соответствующей программы на языке C# [7]:

```
using System;
namespace ConsoleParsing
{ class Program
  { static void Main (string[] args)
    { Console.WriteLine("Введите строку дескриптора: ");
      string mgo = Console.ReadLine ();
      Console.WriteLine ("Результат анализа:");
      Parsing (mgo);
      Console.ReadKey (); }
  static void Parsing(string mgoin)
  {
    string[] parts = mgoin.Split (new char[] { ',' });
    foreach (string s in parts) { Console.WriteLine (" " + s);
    } } }
```

В данном случае для представления фрагментов используется тип string. Работа программы аналогична синтаксическому анализу предложения с целью выделения лексем. Полученные фрагменты легко преобразуются в другой тип (например, методами класса convert [7]) или в дерево, которое отражает структуру входной последовательности и хорошо подходит для дальнейшей обработки.

Методика применения дескриптора

Практическое применение дескриптора включает следующие шаги.

1. Анализ нового проекта и формулирование требований для его выполнения.
2. Определение типов, количества, отношения компонентов и коммуникаций, обеспечивающих выполнение требований и достижения цели проекта.
3. Старт программы построения дескриптора и ввод менеджером соответствующей строки.
4. Декомпозиция строки на элементы.
5. Использование элементов в других программах сопровождения ЖЦ МГО.

Необходимым условием успеха применения методики является работоспособность каналов связи $i1, i4, i5$.

Экспериментальная проверка методики

Для проверки правильности методики решим задачу из области применения инновационных МГО для доставки груза в различные точки труднодоступных территорий.

Пусть для проекта Z требуется описать МГО drone, включающий четыре беспилотника трех типов (1 – $T200$, 1 – $T201$, 2 – $T202$), использующие для связи локальные коммуникации. На взлете и части траектории для уменьшения сопротивления и экономии топлива в МГО используются отношения F , затем они меняются на L и компоненты по отдельности решают собственные задачи, после чего восстанавливается отношение F и осуществляется возврат в точку вылета (рис. 2). Полет осуществляется в грозовых условиях при максимальном риске. Требуется построить соответствующий дескриптор и провести его анализ.

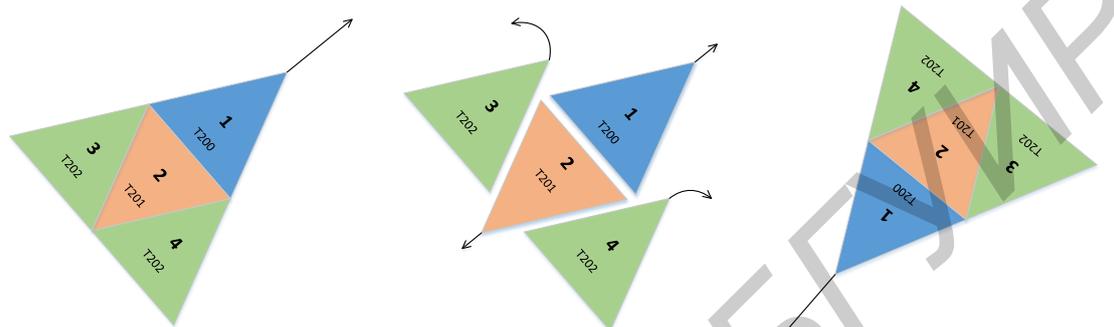


Рис. 2. МГО с изменяющейся структурой

Решение по шагам.

Шаг 1. Главное требование – компактное описание МГО drone.

Шаг 2. Типы компонентов, отношения и коммуникации даны в условии.

Шаг 3. Старт программы и ввод строки дескриптора

$mgo = (Z, drone, F.L.F, 4, 1.T200.1, 2.T201.1, 3.T202.2, com1, R3)$.

Шаг 4. Декомпозиция строки на элементы производится автоматически после ее передачи в метод Parsing. На экране появляется результат:

Введите строку дескриптора:

Z,drone,F.L.F,4,1,T200.1,2.T201.1,3.T202.2,com1,R3

Результат анализа:

Z
drone
F.L.F
4
1.T200.1
2.T201.1
3.T202.2
com1
R3

Шаг 5. Выделенные фрагменты при необходимости преобразуются в любой тип из строкового и используются в других программах, включая мониторинг, оценку шансов выполнения проекта и др. Очевидно, что результат соответствует поставленной задаче. Аналогичным образом можно построить и провести анализ дескриптора для любого МГО, характеристики компонентов которого представлены в стандарте.

На основе вышеизложенного, можно утверждать, что предложенный вариант дескриптора и алгоритм его анализа в полной мере соответствует сформулированным выше требованиям, обеспечивает компактное описание МГО и возможность автоматической обработки его элементов, что минимизирует возможность ошибок при формировании структуры и состава МГО.

Заключение

Разработаны стандарты для описания МГО, интуитивно понятные персоналу, участвующему в ЖЦ МГО. Предложен вариант дескриптора для описания МГО с фиксированной и изменяемой структурой с использованием стандартов. Описан алгоритм разделения дескриптора на элементы, необходимые для решения задач сопровождения ЖЦ МГО. Представлены результаты эксперимента по использованию дескриптора и алгоритма для описания и выделения элементов сложного МГО с переменной структурой. В целом, полученные результаты способствуют сокращению времени формирования МГО и исключению ошибок формализации, вызывающих сбои работы систем мониторинга.

METHODS FOR DESIGN AND ANALYSIS OF DESCRIPTORS FOR MONITORING MOBILE HETEROGENEOUS OBJECTS

A.I. KUZMICH, V.V. KRASNOPROSHIN, A.N. VALVACHEV

Abstract

Standards, methods of construction and analysis descriptor for mobile heterogeneous objects are proposed. An example of their use for complex objects with variable structure is given. Overall, these results allow us to reduce the formation of objects and eliminate errors in the monitoring programs.

Список литературы

1. Distributed Computing and Artificial Intelligence: 10th International Conference. New York, 2013.
2. *Kuhl F., Weatherly R.* Creating computer simulation systems: an introduction to the high level architecture. New Jersey, 1999.
3. *Cai Y., Hua K., Cao G.* // IEEE Int. Conf. Mobile Data Management. 2004. P. 27–38.
4. *Pasco A., Adziew V., Cominos P.* Heterogeneous objects modelling and applications. Springer, 2008.
5. *Кузьмич А.И., Краснопрошин В.В.* // Информатика. 2013. № 4. С. 45–53.
6. *Валькман Ю.Р.* // Матер. междунар. конф. «Искусственный интеллект». 2008. С. 310–318.
7. *Троелсен Э.* Язык программирования С# и платформа.NET 4.5. М. 2013.