

СЪЗДАВАНЕ И ПРИЛОЖЕНИЕ НА ЕДИН СЪБИТИЕН МОДЕЛ

Станимир Стоянов, Иван Стоянов,
Венета Табакова-Комсалова

***Резюме.** В статията е разгледан модел за представяне и работа със събития. Моделът е предназначен да послужи като теоретична основа на кибер-физически-социални системи. Разглеждат се общия подход, необходимостта и приложимостта на такъв вид модели. Освен това, в статията са представени една общата характеристика на модели и включените в модела възможни операции със събития. Дадени са също две класификации на събития, които играят съществена роля за адаптиране на модела за различни приложения. Използваемостта на модела се демонстрира с негова адаптация за разработване на платформа за интелигентно земеделие, наречена ЗЕМЕЛА.*

Ключови думи: изкуствен интелект, събитийен модел, интелигентно земеделие.

Увод

Поради различни причини като напр., промяна на климата, демографски и миграционните потоци, неравномерен икономически растеж земеделският сектор преминава през период на ускорена трансформация, която го поставя пред редица предизвикателства. Дигитализацията на сектора може да стане основно средство за справяне с тези предизвикателства. Подходи, използващи изкуствен интелект и усилен с технологии като интернет на нещата (IoT), кибер-физически-социални системи, блокови вериги и големи данни, предоставят нови възможности за изграждане на нови инфраструктури, подпомагащи селското стопанство (обикновено наречено интелигентно земеделие).

Като важен за икономиката на България сектор, интелигентното земеделие става все по-належаща за решаване задача. В тази връзка през 2021 год. стартира реализацията на Национална научна програма „Интелигентно земеделие“ [1]. В работния план на програмата се предвижда изграждане на платформа за интелигентно земеделие, наречена ЗЕМЕ-

ЛА (ЗЕМЕДЕЛСКИ АСИСТЕНТИ). Теоретичната основа на платформата е събитийен модел (с работно заглавие Event Model).

В тази статия представяме накратко актуалната версия на модела. Идеята за разработване на такъв модел и предишна версия на модела са описани в [2, 3].

Кратък преглед на проблемната област

В специализираната литература се разглеждат различни събитийни модели – някои от най-съществените са обект на този кратък преглед. Още при появата на логическото програмиране Р. Ковалски подчертава в [4] значението на представянето на времето за различни приложения на класическата логика. Тази статия предлага третиране на времето въз основа на понятието събитие. Счита се, че моделът на събитието, предложен в редица статии от изследователи от Калифорнийския университет в Ървайн [5, 6], се състои от отделни аспекти като информационен аспект, структурен аспект, времеви аспект, причинно-следствен аспект, специален аспект и експериментален аспект. В [7] едно събитие може да бъде формално дефинирано като идентичност, включваща обекти, които участват в събитието, набор от действия, които се случват в събитие, среда на събитието, период от време, през който събитието продължава, и твърдения които могат да бъдат предусловие, следусловие и междинно. Напълно завършено смятане на събития е дадено в [8]. В [9] е представен набор от данни Event-Question-Answer за отговаряне на въпроси, ориентирани към събития, върху графи на знания. Инструмент AsyncAPI за автоматизиране на проектирането и внедряването на управлявани от събития инфраструктури е предложен в [10]. В [11] е представена платформа за големи данни за откриване на събития, проследяваща еволюцията в пространството и времето. Обсъжда се също мащабируема архитектура за управление на огромен поток от данни от неструктурирани потоци с цел откриване на гео-социални събития.

Обща характеристика на Event Model

В повечето събитийни модели, разгледани в дадения по-горе преглед, събитията се моделират чрез сложни структури. На нас ни е необходима теоретична основа (рамка), която да позволява виртуализация на физическите обекти, отчитайки също техните времеви, пространствени и събитийни (т.е. тяхното въвличане в различни случвания) аспекти. По

тази причина беше разработен така нареченият събитийен модел, който да послужи за теоретичната основа на платформата. Какъв е нашият подход? За създаване на модела използваме един аналитичен и . . . подход. Анализирайки различни задачи, които трябва да решава една платформа за интелигентно земеделие идентифицираме различни възможни събития. Стигнахме до извода, че различните процеси и сценарии в интелигентното земеделие могат да се представят като последователности от събития. Освен това, събитията могат да бъдат основен механизъм за активиране на тези сценарии и действия, както и за синхронизиране на работата на активните компоненти в платформата. Като събитие ние приемаме нещо, което се случва или се смята за случващо се на определено място и в даден интервал от време и което оказва влияние върху работата на платформата. В ZEMELA, тъй като събитията могат да се случват както във физическия, така и във виртуалния свят, ние трябва да предложим общо символично представяне във виртуалния свят (т.е. събитията във физическия свят са виртуализирани). Намирането на подходяща презентация е предизвикателство поради голямото разнообразие от събития с техните специфики за интересуващата ни област. Моделът е една нотация, която би трябвало да позволява тяхното обобщено и абстрактно представяне.

Разработването на платформата ЗЕМЕЛА се извършва на следните нива:

1. Най-абстрактното ниво е събитийният модел.
2. Второто ниво е представително ниво – представяне на събитията като знания с общ смисъл (правила, фрейми, онтологии).
3. Програмно-техническо ниво – програмната реализация на събитията и тяхната обработка.

Обобщавайки, нашият събитийен модел може да се характеризира накратко, както следва:

- Опростено представяне на събития – много от реферираните публикации използват изключително сложни и трудни за обработка представяния на събития;
- Йерархия на събитията – позволява да се представят различни зависимости и връзки между събитията;
- Рекурсия – моделът предоставя възможност за представяне на

сложни събития чрез рекурсивни дефиниции;

- Проактивност – когато настъпят определени събития, динамично се генерира съответен автономен и проактивен агент, който поема инициативата да уведомява заинтересованите компоненти.

Определение и класификации на събитията

Определение на едно събитие. Като събитие приемаме нещо, което се случва или се смята за случващо се на определено място и в даден интервал от време и което оказва влияние върху работата на платформата. В ZEMELA, тъй като събитията могат да се случват както във физическия, така и във виртуалния свят, ние трябва да предложим общо символично представяне във виртуалния свят (т.е. събитията във физическия свят са виртуализирани). Намирането на подходяща презентация е предизвикателство поради голямото разнообразие от събития с техните специфики за интересуващата ни област.

Класификации. В модела на събитието са дефинирани две основни класификации. Първият използва йерархичен подход, при който събитията от по-високо ниво се третират като предварително определени от събития от по-ниско ниво. По този начин нашият модел на събития разграничава следните три нива на събития:

- Основни събития – обикновено това са атомарни събития, които могат да инициират различни събития в платформата и като такива обслужват предимно други събития (те приписват събития). Следните основни събития са твърдо дефинирани: дата, час, местоположение и (екстремални) стойности на сензорите. Поради особеностите на домейна, който представлява интерес, могат да бъдат посочени допълнителни основни събития. Този вид събития заемат най-ниското място в йерархията.
- Системни събития – този тип събития представят различни събития в системната инфраструктура на платформата като генериране/премахване на динамични компоненти и изпращане/получаване на съобщения. Системните събития обикновено се използват за управление на взаимовръзката между личния асистент и другите компоненти на платформата. Този проблем ще бъде разгледан по-подробно по-долу.

- Домейн-събития – представят събития, типични за интересуващата ни област и заемат третото най-високо ниво в йерархията на събитията.

Втората класификация е свързана с това как протичат събитията във времето. В този смисъл разграничаваме две големи групи събития, които сме назовали:

- Планови – тези събития се очакват и могат да се появят в определен момент (или интервал) във времето. Личният асистент обикновено работи с планирани събития.
- Случайни – събитията възникват случайно в определено време или интервал от време. Тези събития обикновено се идентифицират от Event Engine.

Между двете класификации съществува тясна зависимост (връзка). За да я разкрием ще разгледаме по-подробно базовите събития на модела:

- Time – най-общо времето може да бъде реално или относително, представено като часове, дни, месеци или години. Това събитие е основополагащо за работа с темпоралните аспекти на моделираните физически обекти, процеси или сценарии. Този тип събития обикновено са съставна част на плановите домейн-събития.
- Location – най-общо това базово събитие се използва за представян (моделиране) на пространствените аспекти на интересуващите ни физически обекти, процеси или сценарии. Евентуално може да бъде съставна част на двата типа домейн-събития.
- Extreme value – този тип базови събития служат за идентифициране на неочаквано възникнали домейн-събития. Като такива те обикновено са съставна част на инцидентните домейн-събития.

Формален модел

Представяне на събития. За формалното представяне на модела ще въведем следните означения:

- E – множеството на всички събития, случващи се в рамките на платформата.
- $ei \in E$ – едно отделно събитие.
- $e = \langle eid, etype, epar \rangle$ – всяко събитие представяме като тройка

от следните елементи: eid – идентификатор на събитието, $etype$ – тип на събитието, $epar$ – списък с атрибути на събитие, който може да бъде празен.

- Едно събитие ei наричаме зависимо, ако съществува събитие ej с $ej \in epar(ei)$.

Оператори. В събитийния модел дефинираме следните три операции върху домейн-събитията:

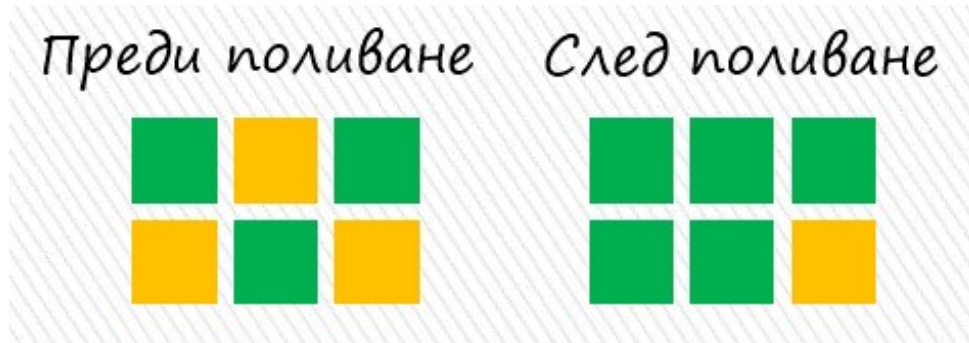
- $ei \uparrow ej$ – възникването на ei предизвиква случването на ej .
- $ei \downarrow ej$ – възникването на ei прекратява събитието ej .
- $accu(e)$ – акумулиране на събития.

Коментар за алгебричните свойства на операциите. За да можем да използваме модела на събитието като основа за работата на двата модула на оперативното подпространство (РА и Event Engine), изискваме и двата оператора да отговарят на следните изисквания:

- Некомутативност – $ei \uparrow ej \neq ej \uparrow ei$, т.е. ако събитието ei предизвиква случването на събитие ej , от това не трябва да следва, че ej може да предизвика случването на ei . Аналогично за оператора \downarrow .
- Ляво-асоциативност за \uparrow – от досегашния анализ изглежда, че операторът \uparrow е ляво-асоциативен, т.е. $ei \uparrow ek \uparrow ej$ може да се интерпретира като $(ei \uparrow ek) \uparrow ej$ или събитие ei предизвиква възникване на събитие ek , което от своя страна води до възникване на събитие ej .
- Дясно-асоциативност за \downarrow – по подобен начин изглежда, че операторът \downarrow е дясно-асоциативен, т.е. изразът $ei \downarrow ek \downarrow ej$ трябва да се интерпретира като $ei \downarrow (ek \downarrow ej)$ или събитие ek трябва първо да прекрати събитие ej преди то да бъде прекратено от събитие ei .
- Транзитивност – домейн-събитията са транзитивни.

С един малък пример ще се опитаме да мотивираме изискването за некомутативност на операторите. Да предположим, че персоналният асистент е идентифицирал домейн-събитието „засушаване“. Това събитие от своя страна предизвиква събитието „напояване“. Некомутативността на този оператор ни осигурява претенции за прецизност на поливната

система (включително на нейния софтуер). Нека допуснем, че операторът \uparrow е комутативен. Тогава, ако след напояването останат сухи парцели (т.е. „напояване“ \uparrow „засушаване“) приемаме, че поливната система коректно е завършила напояването (фигура 1).



Фигура 1. Некомутативност на операторите

Заклучение

В статията е представен накратко събитийен модел, който теоретична рамка за разработване на платформата за интелигентно земеделие ЗЕМЕЛА. Оперативният компонент на платформата, включващ персонален асистент и модула Event Engine, оперират също като интерпретатори на възникнали събития. В момента се разработва първи прототипни версии на двата софтуерни модула.

Благодарности

This work is supported by the Bulgarian Ministry of Education and Science under the National Research Program “Smart crop production” approved by Decision of the Ministry Council No. 866/26.11.2020 and partially supported by the Scientific Research Fund under Grant No. KP-06-X36/2, project BG PLANTNET “Establishment of National Information Network GenBank – Plant genetic resources”.

Литература

- [1] Bulgarian Ministry of Education and Science under the National Research Program “Smart crop production”, Grant agreement No D01-65/19.03.2021 approved by Decision of the Ministry Council No. 866/26.11.2020.
- [2] S. Stoyanov, A. Stoyanova-Doycheva, V. Ivanova, V. Tabakova-Komsalova, An Event Model for Smart Agriculture, *2021 IEEE Inter-*

- national Conference Automatics and Informatics (ICAI)*, 30 September – 2 October 2021, Varna.
- [3] S. Stoyanov, A Virtual Space Supporting eLearning, *Proceedings of the Forty Fifth Spring Conference of the Union of Bulgarian Mathematicians Pleven*, April 6–10, 2016, 72–82.
 - [4] R. Kowalski, M. Sergot, A Logic-based Calculus of Events, *New Generation Computing* 4 (1986), 67–94, 1986, Ohmsha, Ltd.
 - [5] R. Jain, EventWeb: Developing a Human-Centered Computing System, *Computer*, February 2008, 42–50.
 - [6] U. Westermann, R. Jain, Toward a Common Event Model for Multimedia Applications, *IEEE Multimedia*, January-March 2007, 19–29.
 - [7] W. Liu, Z. Liu, J. Fu, R. Hu, Z. Zhong, Extending OWL for Modeling Event-oriented Ontology, *2010 International Conference on Complex, Intelligent and Software Intensive Systems*, DOI10.1109/CISIS.2010.88, 581–586.
 - [8] E. Mueller, *Commonsense Reasoning. An Event Calculus Based Approach*, Morgan Kaufmann, 2015.
 - [9] T. Costa, S. Gottschalk, E. Demidova, Event-QA: A Dataset for Event-Centric Question Answering over Knowledge Graphs, *The 29th ACM International Conference on Information and Knowledge Management*, October 19–23, 2020, Virtual Event, Ireland.
 - [10] A. Gómez, M. Iglesias-Urkia, A. Urbieto, J. Cabot, A model-based approach for developing event-driven architectures with AsyncAPI, *MODELS'20*, ACM, October 1–23, 2020, Montreal, QC, Canada.
 - [11] I. Afyouni, A. Khan, Z. Al Aghbari, Spatio-Temporal Event Discovery in the Big Social Data Era, *IDEAS 2020*, ACM, August 12–14, 2020, Seoul, Republic of Korea.

Станимир Стоянов^{1,*}, Иван Стоянов², Венета Табакова-Комсалова³
^{1,2,3} Пловдивски университет „Паисий Хилендарски“

Факултет по математика и информатика,

бул. „България“ 236, Пловдив, България

^{1,2,3} ИИКТ – Българска академия на науките, София, България

Автор за кореспонденция: stani@uni-plovdiv.bg

CREATION AND APPLICATION OF AN EVENT MODEL

Stanimir Stoyanov, Ivan Stoyanov, Veneta Tabakova-Komsalova

Abstract. *This paper discusses a model for presenting and working with events. The model is intended to serve as a theoretical foundation of cyber-physical-social systems. The general approach, necessity and applicability of such models are also discussed. Furthermore, a general characteristic of models and the possible operations with events are described in the article. Two classifications of events are also given, which play an essential role in adapting the model for various applications. The usability of the model is demonstrated by its adaptation to a smart farming platform known as ZEMELA.*

Key Words: artificial intelligence, event model, smart agriculture.