

Design patterns for digital twins

Chief Assist. Prof. Mariya Armyanova PhD
University of Economics – Varna, Varna, Bulgaria
armianova@ue-varna.bg

Abstract

Digital twins are virtual counterparts of people, physical objects, processes or systems. They are models based on a very large amount of data about real objects and are used to remotely maintain, monitor, model and predict the behavior of the real object. The development of digital twins is an increasingly widespread practice and their implementation is expected in various spheres. The technologies used to create them are developing very intensively and dynamically. Therefore, the design and architecture of digital twins are not strictly defined, despite the popularity of their application. However, in different spheres, they are applied in different ways and this leads to different architectural solutions. The article examines a pattern-based architecture. For this purpose, design patterns and the possibilities of their use in the development of digital twins are considered.

Keywords: Digital Twin, IoT, Design Patterns, AI

JEL Code: C61, C88

DOI: 10.56065/IJUSV-ESS/2024.13.2.107

Introduction

Дигиталните близнаци се внедряват в различни сфери, а за разработването им се използват технологиите за взаимосвързани системи и мрежи, за събиране и предаване на големи данни, IoT (Интернет на Нещата), съчетан с нарастващите възможности на ИИ (Изкуствения Интелект) и добавената реалност. Те са виртуален двойник на хора, физически обекти, процеси или системи. Те представляват модели, базирани на много голям обем от данни за реалните обекти и се използват за отдалечено поддържане, наблюдение, моделиране и прогнозиране на поведението на реалния обект. Разработването на дигиталните близнаци е все по-широко разпространена практика и се очаква внедряването им в различни сфери. На този фон изпъкват и някои предизвикателства и проблеми свързани с киберсигурността и оперативната съвместимост, защитата на данните и злонамерената употреба, някои етични и правни въпроси, като отговорността за качеството на данните, надеждността, управлението и доверието в технологиите.

Увеличаването на използването на дигиталните близнаци се дължи на многобройните им предимства. Техни приложение са разработени за различни области (фиг. 1). В здравеопазването дигиталните близнаци спомагат за улесняването на прогнозната и персонализирана медицина, за подобряване на резултатите за пациентите и чрез превенция, ранно откриване и лечение намаляват разходите за здравеопазване. Например пациентите използват устройства включени в IoT. На база на събраните здравни данни на индивиди и общества, могат да се правят прогнози за нуждата от здравни грижи и лекарства. Могат да се прогнозират заболявания и да се създават персонализирани и целенасочени лекарства.

В производството и създаването или оптимизиране на продукти дигиталните близнаци помагат на организациите да избегнат потенциални проблеми, да оптимизират производствените и технически проекти и да намалят разходите. Дигиталните близнаци се използват и за дистанционно управление на оборудването и прогнозиране на нуждите му от ремонт. Използват се и за управлението на веригите за доставка и след продажбеното обслужване. Прилагането на математически модели на транспортни задачи осигурява по-ниски общи транспортни разходи (Vasilev et.al., 2023). За нуждите на строителния бизнес се използва IoT и се създават дигитални близнаци, които позволяват да се вземат решения в реално време (Sulova, 2023). Създаването на дигитални близнаци и прилагането на машинното

обучение за събраните данни води до подобряване на стабилността и кредитния портфейл във финансовия сектор (Aleksandrova & Parusheva, 2021).



Figure 1. Сфери на приложение

Друго тяхно приложение е свързано с наблюдението на околната среда и климата. За създаване на прогнози се събират данни от многобройни източници и за разлика от други области, нивото на поверителност на данните в областта е по-ниско. Дигитални близнаци се използват в интелигентните градове, в транспорта и енергийния сектор.

1. Особенности и видове дигитални близнаци

Дигиталният близнак е виртуален модел на реално съществуващ физически обект, система или процес, създаден с помощта на данни и информация, събрани от сензори и други източници. Технологията се основава на идеята, че цифрова информационна конструкция за физическа система може да бъде използвана като самостоятелна единица, често пъти независимо от реалната. Тази цифрова информация се определя като дигитален близнак на информацията, която е вградена в самата физическа система и е свързана с тази физическа система през целия жизнен цикъл на дигиталния близнак (Grieves & Vickers, 2017).

При изграждането на дигиталните близнаци се използват два подхода. В единия случай дигиталният близнак е управляван от модел, а в другия е управляван от данни. При първия подход се използват математически модели за създаване на дубликат на реалните системи, но този подход е много сложен за реализация, поради непрекъснатите промени в реалния обект. При подход за разработка, управляван от данни, се разчита на възможностите за МО (машинно обучение) и се събират големи количества данни, които отразяват състоянието на реалните системи във времето.

Дигиталните близнаци често използват данни в реално време, събрани от сензори и или интернет източници. След това данните се анализират, като се използват експертни системи, машинно обучение или ИИ. Дигиталните близнаци изпълняват три основни функции, като прототипиране, мониторинг и контрол на реалния обект.

Според някои автори дигиталните близнаци се делят на три вида (Grieves, 2023) – дигитален модел, дигитална сянка и дигитален близнак. А според други на четири (Qian et.al.,

2022), като прототип (DTP) (Rasheed et.al., 2020), DT инстанция (DTI) (Haag and Anderl, 2018), DT агрегация (DTA) (Jones, 2020) и DT среда (DTE) (Liu, 2005). Като принципите и възможностите на дигиталния модел и прототип съвпадат. Дигиталната инстанция и сянка съдържат еднопосочна връзка за влияние между реалната система и дигиталния близък, макар и в различни посоки. Дигиталното агрегиране или пълноценния дигитален близък имат подобни функции, но дигиталното агрегиране се отнася до случаи на изготвяне на прототип на даден обект, а не на система от взаимодействащи си обекти. Близъкът на среда се отнася до взаимодействащи си група от реални обекти.

За да бъде определен софтуерът като дигитален модел или DTP, той се създава като цифрово копие на реално съществуващата система, например за наблюдение на интелигентни мрежи. Използва се за симулации и прогнозиране, но дигиталното копие не може да влияе върху реалната система автоматично и измененията в реалната система също не се отразяват автоматично върху дигиталната. Много често дигиталния близък започва съществуването си като модел и в следствие се разширява. За IoT създаването на ново устройство може да е предшествано от създаване на дигитален близък модел, който се използва за симулации преди разработването на реалното устройство. Дигиталният прототип също се използва за подобряване на ефективността на работата на реалната система.

При дигиталната сянка промените в реалната система се отразяват автоматично в дигиталния близък. Използва се за критични системи, за да може да се наблюдават измененията в реалната среда и да се прогнозира бъдещи проблеми, за които да се реагира своевременно. С дигитална сянка може да се осъществява мониторинг на околната среда и да се вземат решения за подобряването ѝ, както и да се разработват стратегии и да се проследи за успешното им изпълнение.

При модела на дигиталната инстанция дигиталният близък може да управлява реалната система, като изпраща прогнози или насоки при промени в околната среда. Използва се например в аграрния сектор.

Пълноценният дигитален близък или агрегацията може автоматично да влияе върху реалната система и да бъде повлиян от нея, т.е. агрегира дигитална сянка и инстанция. Връзката е двупосочна, като всяка промяна в реалната система се отразява в дигиталния близък, а с контролната система на дигиталния близък може да се повлияе на реалната система, за да се постигне желания резултат. Своевременно могат да се внасят корекции и да се задават стратегии.

Дигиталният близък по модела на цифровата среда може да се разгледа като съвкупност от много дигитални близъци, които си взаимодействат помежду си. Дигиталната среда спомага за управлението на всички дигитални близъци, включени в обхвата ѝ и се използва за големи системи като интелигентни домове, градове, здравни системи. Дигиталната среда синхронизира обработката на заявките между множество близъци и съчетава целите на различните дигитални близъци, за да максимизира общата ефективност, например при потреблението на ток от няколко домакинства.

Независимо от голямото разнообразие и области на приложение, може да се открият общи черти на дигиталните близъци, които позволяват да се представи архитектурен модел.

2. Архитектура и технологии на дигитален близък

Повечето автори се придържат към четирислойна архитектура с някои вариации, а някои дори предлагат трислойна архитектура без да идентифицират слой на приложенията за крайния потребител. Придържайки се към предложената четирислойна архитектура може да внесем някои изменения (фиг. 2).

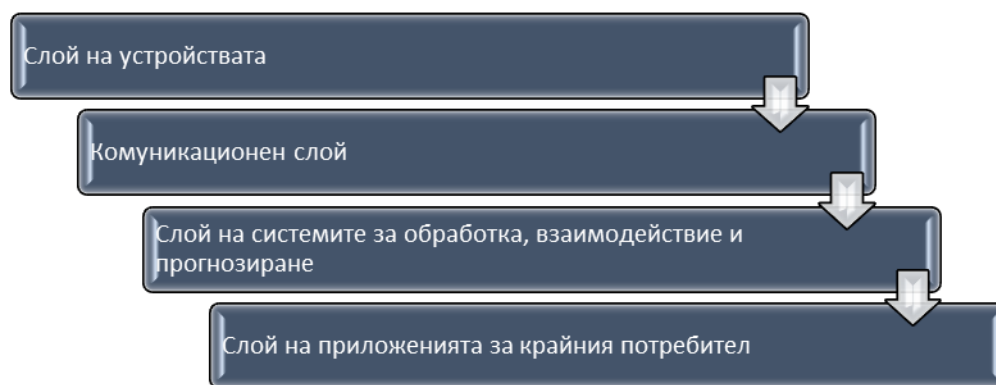


Figure 2. Модел на слоевете за дигитален близнак

Източник: Адаптиран от (Čolaković and Hadzialic, 2018)

Слоят на устройствата, определян и като слой на реалните обекти, се състои от сензорите и IoT устройствата. Тези устройства формират системата на реалния обект. Слоят служи за събиране и изпращане на данни за средата и устройствата, работещи в нея. Събраната информация е ключова за поддържането на съответствието с реалната система. Към този слой се изпращат и управляващите въздействия. Често в слоя се извършва и първоначална обработка на събраните данни, а при федеративното обучение (FL) се изпълняват и някои алгоритми за обучение на IoT устройствата.

Следващият слой е комуникационният. Той осигурява инфраструктурата за прехвърляне на данни между реалния и дигиталния близнак. В дигиталните близнаци се осъществява три типа комуникации (Barricelli et.al., 2019): между дигиталния близнак и реалния обект; между дигиталния близнак и останалите дигитални близнаци в средата; и между дигиталния близнак и експертите, които взаимодействат с дигиталния близнак. Целта на така определения комуникационен слой е да предава информацията, като гарантира нейната сигурност и надеждност. В комуникационния слой могат да работят различни шлюзове, които да събират и обобщават информация от сензори и устройства с по-малки изчислителни възможности. Този слой включва и комуникационните протоколи. Поради ограниченията на мрежовата инфраструктура се разработват, различни протоколи, които да ускорят преноса на данните.

Слоят на системите за обработка картографира получените данни от комуникационния слой (Hatcher et.al., 2021), за да може да локализира различните ресурси. След това данните се подлагат на анализ с моделите на ИИ, базирани на машинно или дълбоко обучение. Резултатите от анализа и прогнозите могат да се използват за управление на устройствата или да се предават на слоя на приложенията за крайния потребител. В този слой потребителят изпраща различни заявки до дигиталния близнак, да получи актуална информация и прогнози и чрез дигиталния близнак може да въздейства на реалните устройства.

Дигиталните близнаци се базират на IoT и големите данни, които обработват с помощта на ИИ. Дигиталните близнаци съчетават много технологии, които сами по себе си не са дигитални близнаци. Например симулацията, оптимизацията и дигиталния прототип не е задължително да имат IoT компоненти и да съответстват на реални продукти, за които да се правят тестове в реално време. Машинното обучение и агент-базираното моделиране са технологии на ИИ, но те не винаги създават копие на реално съществуващи елементи. Автономните системи могат да управляват даден процес, но не е задължително и да се подобряват (Sharma et.al., 2022).

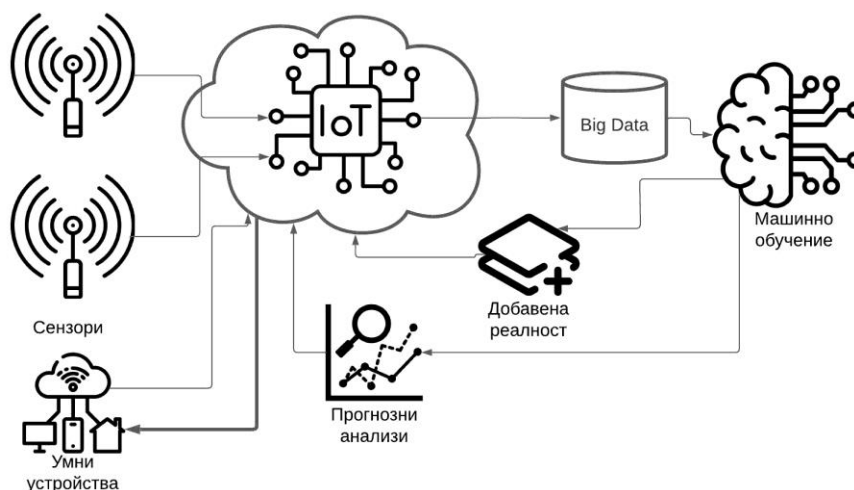


Figure 3. Основни технологии използвани в дигиталните близнаци

Дигиталните близнаци използват IoT устройства за събиране на информация от сензори за реалния обект и крайните устройства (фиг. 3). За да може да се съберат данните и да има съответствие с реалния обект е необходима високоскоростна връзка, която да позволи включване на устройствата в IoT. Разчита се на технологиите за съхранение и обработка на големите данни, за да се съберат данните от IoT устройствата, които се предоставят на системата за МО. МО се използва за прогнози и обратна връзка и за откриване на стратегии за смекчаване на нежеланите резултати. Трябва да се гарантира сигурността на данните, които са събрани, както и сигурността на преноса им между различните компоненти. Разчита се на протоколи за сигурност при споделяне и механизми за оторизация и удостоверяване. При създаване на дигитални близнаци за среда, като транспортна система или интелигентен град се разработват отделни подсистеми за реализацията на дигиталния близнак, които могат да са мрежови, изчислителни и контролни.

Необходимостта от осигуряване на надеждност, точност, бързина и сигурност при преноса и обработката на непрекъснато нарастващия обем данни поставя много високи изисквания към комуникационните и изчислителните ресурси. Проблемът се усложнява от изискването за ефективност и мащабиране на обработките. В опит да се решат тези предизвикателства се въвежда идеята за разпределена, а не централизирана обработка на данните и децентрализирано обучение. Федеративното обучение (FL) е разновидност на МО, което се опитва да разреши предизвикателствата пред преноса и обработката на големи данни в IoT. Целта е да се използват подобряващите се възможности на устройствата, като моделите за МО се обучават с ресурсите на устройствата или на специализиран сървърен слой, без да се налага данните събрани да се изпращат на централизирано място. По този начин се гарантира поверителността и сигурността на данните, които не напускат устройствата в суров вид или дори изобщо не се прехвърлят към централизирано хранилище. Особено необходима е тактиката за сфери, които използват чувствителна информация. Чрез намаляване на необходимостта от преместване на данни, FL помага да се избегнат забавянния и намалява търсенето в мрежите, което прави възможно анализирането на данните бързо и локално. Освен това FL може да работи с различни видове и количества данни на много устройства, като помага на IoT системите да се мащабират ефективно, като същевременно запазва информацията за потребителите поверителна и защитена (Yadav et.al, 2022). FL има потенциал да се използва, защото повишава ефективността, надеждността и гъвкавостта.

За всяка една технология има специфична група от шаблони, които подпомагат реализирането им. За IoT се предлагат редица шаблони, дори от самите доставчици на облачни услуги, например има шаблони за AWS.

3. Предизвикателства пред дигиталните близнаци и възможности за решаването им с шаблони за проектиране

Дигиталните близнаци се съчетават с технологии за ИИ. Използването им дава възможност да се повиши интелигентността, като се отчитат всички промени в реалната система. За да има смисъл обаче от работата на дигиталния близнак, прехвърлянето и обработката на данните трябва да става в реално време и да е точно. И тук възниква въпроса за ефективно използване на изчислителни ресурси, мрежовите ресурси и ресурсите за съхранение.

Основните предизвикателства пред създаването на дигиталните близнаци са свързани със своевременната синхронизацията с реалния обект. При това могат да възникнат проблеми с хардуера или мрежови повреди, които да намалят надеждността. Включването на голям брой потребители предполага възникване на проблеми с претоварването на мрежите и високата сложност на алгоритъма за управление. Възможно решение е използване на механизми за FL, както и осигуряване на надеждна комуникация между различните системи.

Друг проблем е необходимостта от обвързване на дигиталния близнак със съществуващите софтуерни системи (Grieves, 2019). Често се използва разнообразен софтуер за реализацията на бизнес процесите и е трудно да се постигне съгласуваност с всички системи. Възможно решение е да се използват технологиите на ИИ, за да се създаде модел, който да открие оптималните възможности за взаимодействие със съществуващите системи, което обаче поставя проблема за високите начални разходи. За решаването на проблемите за интеграцията със съществуващите софтуерни системи могат да се ползват шаблони за комуникация (Peter, 2016) и шаблони за комбиниране на устройствата (Qanbari et al., 2016).

При разработването на дигитален близнак форматът, пълнотата и качеството на събраните данни рядко са еднакви. Затова е необходима допълнителна обработка, която да подготви данните за използване. Съхранението, предаването и обработката на големи количество данни, създава значителни разходи за компютърната инфраструктура, за да се постигне висока производителност. Затова се търси метод за събиране на минимум данни при това с различно качество, без значително и отрицателно въздействие върху ефикасността, например разчитайки на устройствата за FL. Друга възможност е създаване на допълнителен съвършен слой, в който да се извършва началната обработка на данните, след което да се изпращат към централизираното хранилище. Компютърните възли участващи в този слой може да разчитат на блокчейн технологията за възнаграждения за приноса им в обработката и за отчитане на надеждността им при съпоставяне с крайния резултат от обединяването на отделните резултати от обученията.

Като всяка технология работеща с големи данни важен е въпросът за киберсигурността за гарантиране на сигурността на данните на организацията, личните данни и данните на партньорските ѝ организации при преноса, събирането и съхранението на данните. Шаблоните за киберсигурност помагат за решаването на тези проблеми. Тъй като приложенията функционират в облачна среда, най-голямо приложение имат шаблоните от категорията шаблони за сигурност в облачните системи (Erl, 2015).

Осигуряването на непрекъснатата свързаност на IoT устройствата е важно за работата на дигиталния близнак. Възможно е обаче да се избегнат прекъсванията поради спиране на захранването, софтуерни проблеми или други непредвидени обстоятелства. При предаването на данните, трябва да се отчетат не само особеностите на текущата комуникационна среда, но и пространството за съхранение на данни на устройството. Могат да се използват шаблони за

IoT (Koster, 2014; Chandra et al., 2016), които могат да осигурят надеждна обвързаност на устройствата в IoT.

Много изисквания към качеството на работа, като бързодействие, наличност и надеждност, безопасност, мащабируемост са въведени в редица облачни шаблони. Други особености на облаците, като заплащане единствено за използване и ниските изисквания към индивидуалните ресурси на клиентите, също изискват управление на системата в реално време. Могат да бъдат реализирани с помощта на други облачни шаблони (Fehling, 2015; Varia, 2010).

Дигиталните близнаци използват много технологии и за всяка една от тях могат да се открият подходящи шаблони, които да се използват за създаването на отделни подсистеми за дигиталните близнаци. Могат да се използват шаблони за IoT, киберсигурност, големи данни, ИИ. Затова специфичните шаблони за дигиталните близнаци са тези, които ги отличават от другите видове софтуерни решения. Те не са задължително свързани с конкретна технология, а определят софтуерът, който се разработва като дигитален близнак. Първата група от такива шаблони са от най-високо ниво на абстракция, т.е. архитектурните шаблони (Tekinerdogan and Verdouw, 2020).

Концептуалният модел на архитектурата показан на фигура 1 не е достатъчен, за да представи напълно системната архитектура на дигиталния близнак. Моделът е много общ и може да се използва за подпомагане на разбирането на различни системи, които се базират на многослойната архитектура. Обаче моделът не представя достатъчно характеристики, за да се определи типът на дигиталния близнак или необходимите му елементи за разработка.

В зависимост от системите, за които се използва, като например за мониторинг, за контрол или автономна система, дигиталният близнак може да е от различен тип. В съответствие с това има създадени различни архитектурни шаблони за проектиране, които са полезни за проектиране на система с дигитален близнак (Tekinerdogan and Verdouw, 2020).

Първият шаблон е за дигитален близнак **Модел**. Използва се в началните етапи на създаване на устройства за IoT или системите близнаци. По същество при този модел все още няма създаден реален обект, а моделът на дигиталния близнак се използва за симулации и по особеностите на близнака се разработва реалният физически обект. Дигиталният близнак може да се използва за създаване и на общ интерфейс за многократна употреба (абстрактен клас) и да се създадат различни обекти дигитални близнаци, които да се свързват с него (фиг. 4). Така общият интерфейс може да се използва за подобни, но различни физически обекти.

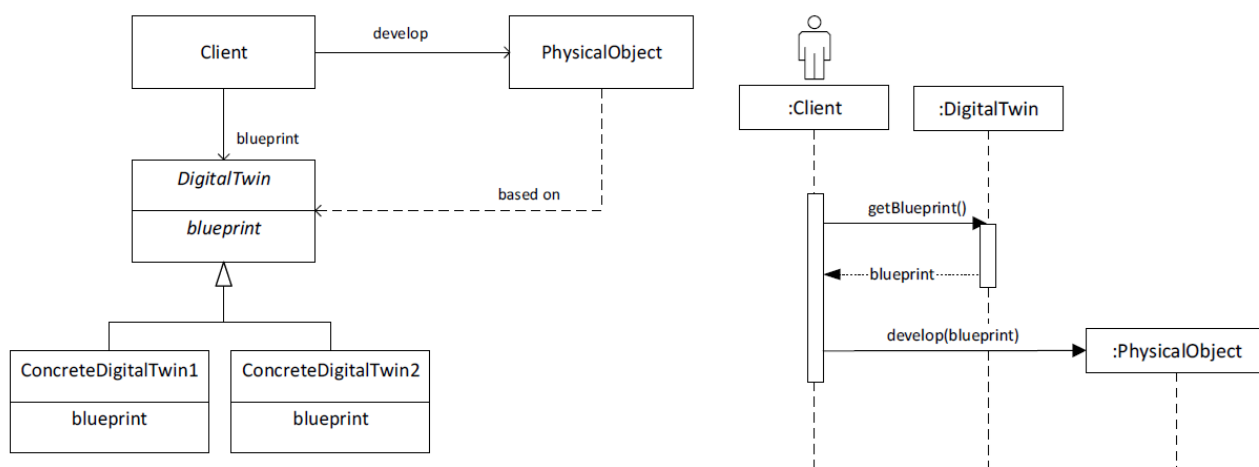


Figure 4. Диаграма класовете и последователността за шаблон на дигитален близнак Модел (Tekinerdogan and Verdouw, 2020)

Друга специфична архитектура е за дигитален близък **Генератор** (Производител), който също като модела дефинира възможността за производството на реалния физически обект въз основа на модела съдържащ се в дигиталния близък. Разликата е, че не е необходимо човешка намеса, а производството е автоматично. За да може да стане това дигиталният близък трябва да включва и техниките за разработка, с които се генерира обекта. Може да се използва за производството на нов продукт според желанията на клиентите. Като основа може да се използват техниките използвани в производството на сходен продукт.

Шаблонът Дигитална **Сянка** отразява характеристиките на реален физически обект. В зависимост от изискванията и целите могат да се дефинират различни дигитални близкаци за един физически обект. Дигиталният близък може да се разработи като модел, т.е. като се въведат данни за физическия обект от разработчиците или автоматично като се генерира модел въз основа на събрани данни със сензори от физическия обект. Подобна разработка се базира на възможностите на ИИ за извличане на модели от събраните данни. Данните могат да се събират от един реален обект или от различни сходни физически обекти.

Шаблонът Дигитално **Съответствие** е създаден дигитален модел, който се използва за откриване на физически обекти, които съответстват на свойствата заложи в модела. Обикновено подобни приложения се използват за разпознаване и класификация на обекти в машинното обучение или дълбокото обучение. Процесът на откриване на физическите обекти или отхвърляне е автоматичен. Използват се алгоритми за търсене, класификация и разпознаване на обекти.

Шаблонът за **Прокси** (Заместител) дигитален близък се използва като прокси за физическия обект, т.е. цялата комуникация за физическия обект минава през дигиталния. Дигиталният близък действа и отговаря от името на физическия обект. Той може да обработва предварително информацията, или да извърши последваща обработка или да осигури някои услуги, извършвани от физическия обект. Може да се използват няколко дигитални близкаци, които да осигуряват различни услуги. Прокси може да се използва за подобряване на сигурността, защитата от неоторизиран достъп, подобрена ефективност, обслужване на повече клиенти и други. Проксита може да е статично или динамично, като отразява промените в състоянието на физическия обект. Може временно да замести физическия обект, като поеме някои обработки.

Шаблонът Дигитално **Възстановяване** се използва при необходимост от възстановяване на предишно състояние на физически обект. Такава необходимост може да възникне поради повреда, загуба или привеждане в съответствие със състоянието на други физически обекти или в съответствие със средата. Състоянието на физическия обект в определени контролни точки се съхранява в дигиталния близък и при необходимост въздействието може да е обратно към физическия обект. По-този начин може да се осигури по-голяма надеждност на работата и по-бърза реакция на различни стимули от реалния обект.

Шаблонът за Дигитален **Мониторинг** се използва за наблюдение на физическия обект. Един дигитален близък може да наблюдава група от сходни обекти и физическият обект също може да бъде наблюдаван от няколко дигитални близкаци. Наблюдението може да е непрекъснато или на определени времеви интервали. Може да се използва за системи за мониторинг на замърсяването на въздуха в градска среда, мониторинг на показателите на пациенти, непрекъснати производства и др.

Шаблонът за Дигитален **Контрол** се основава на шаблона за мониторинг, но изпълнява и функции за управление, за да адаптира състоянието на физическия обект. Използват се различни системи за вземане на решение често с ИИ, следят се контролни параметри и се избира действие, което се прилага върху физическия обект. Действието може да се извърши от оператор, а не да се извърши автоматично вследствие на команда от дигиталния близък. Физическият обект съдържа и изпълнителен механизъм, който приема външни въздействия.

Обикновено операторът управлява дигиталния близък и извършва въздействието на реалния обект от разстояние.

Шаблонът за Дигитална **Автономия** може да използва възможностите за въздействие на дигиталния близък върху реалния обект без нужда от човешка намеса. Дори контролните параметри могат да се определят от дигиталния близък и да не изискват човешка намеса. С коригиране на контролните параметри може динамично да се отговори на промените в средата. С други думи дигиталният близък може да се учи, като в зависимост от резултата адаптира своите контролни действия.

На база на целите на архитектурните шаблони, те могат да бъдат разделени според етапите на жизнения цикъл на реалния обект (фиг. 5). Шаблоните Модел, Сянка и Производител се използват при разработката на реалния обект, а останалите шаблони по време на използването му.

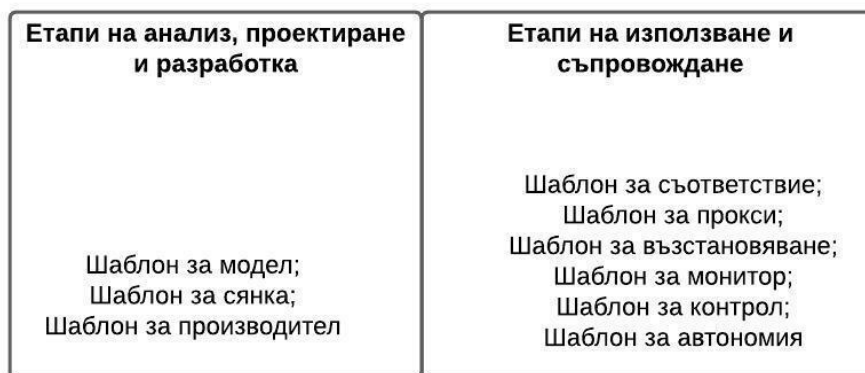


Figure 5. Архитектурни шаблони за дигитални близнаци

Други шаблони са предложени от (Khail and Schnizer, 2024). Те са шаблони за проектиране и решават проблеми от по-ниско ниво на абстракция, защото се използват за постигане на определени характеристики в дигиталните близнаци. Тези шаблони са достатъчно общи и могат да се приложат към дигитални близнаци в различни сфери на приложение. Техните решения се основават на класическите шаблони за проектиране на GoF и принципите на обектноориентираното програмиране, но авторите ги модифицират за целите на дигиталните близнаци.

Такъв шаблон е **Фасада** на дигиталния близък. Той се отнася до интерфейса, чрез който потребителите взаимодействат с дигиталния близък. Шаблонът се базира на шаблонът за проектиране Фасада на GoF. Необходимостта от прилагането му е в непрекъснатото разширяване на системата с нови подсистеми, които може да са с различен и дори объркващ интерфейс за потребителите. Решението на шаблона Фасадата е да предложи на потребителя достъп през унифициран API. Този API представя един и същ интерфейс, независимо от използваната подсистема. Шаблонът скрива спецификата на това как работи всяка подсистема, предоставяйки на потребителя прост, последователен начин за взаимодействие с дигиталния близък. Шаблонът не определя механизмите за маршрутизация на данните между подсистемите. Затова разработчиците трябва да проектират механизми за вътрешна комуникация, така че стойностите да се изпращат до съответните подсистеми.

Друг шаблон е предложен за **актуализацията** на дигиталния близък. Той се базира на принципа на проектирането за отделяне на данните от функционалността с помощта на свойства, които капсулират достъпа до конкретни променливи. По този начин, потребителят няма да е принуден да следи непрекъснато променящите се стойностите на сензорите на устройствата, а може да зададе желани резултати без да се налага да подава множество

стойности за различни параметри. Освен това прехвърлянето на стойността на едно свойство, вместо множество отделни стойности на параметри, опростява и ускорява комуникацията между дигиталния и реалния обект.

Друг шаблон е шаблонът за **Синхронизация** на състоянията на дигиталния близък. Много пъти от дигиталният близък се очаква да изпълнява повече от една роля при взаимодействие с реалния обект. Може да проследява само състоянието или да оказва въздействие върху реалния обект. Авторите предлагат проблемът да се реши с извикване на методи, които да прехвърлят състоянието на параметрите към или от реалния обект, или ги записват във файл. Разбира се за изпълняване на различните методи, разработчиците трябва да предвидят много фактори, но целта на шаблоните е да предостави насока за решение на проблема, която след това трябва да се специализира според нуждите на конкретната система.

Други групи шаблони са от по-ниските нива на абстракция, т.е. за конкретни приложения и затова са силно зависими от областта на приложение на дигиталният близък. Например шаблоните в производствената сфера могат да се обединят около няколко цели:

- изследват данните и се стремят да идентифицират самостоятелно проблеми с качеството на производството в сравнение със заложеното в стандартите;
- предупреждават операторите чрез подходящи алармени системи за необичайни или извън допустимите ситуации по време на работа;
- анализират и съпоставят симптомите и причините за повредите и дефектите в производството;
- подпомагат избора на коригиращи действия за отстраняване на откритите повреди и дефекти.

В някои критични ситуации дигиталният близък използва шаблонът за автономия, за да отговори много бързо и пи минимални измененията в показателите на производството. Някои автори (Semeraro et. al., 2023) предлагат отделни шаблони за проектиране, които следят процеса на производство и могат да предприемат действия при възникване на проблем. Такъв е шаблон за контрол на **микроспиране**, който се използва в непрекъсваеми производствени процеси. При възникване на микроспиране или забавяне системата може да предприеме действия по синхронизация, които да не нарушат производствения процес.

Шаблонът за **рестартиране** на устройство се използва за стартиране на производства, които изискват извършване на няколко последователни или паралелни действия, за да може да се стартира даден процес безопасно.

Друг специфичен шаблон е шаблонът за **контрол на затягащ механизъм**, който изисква съчетанието на два фактора – сила на затягане и налягане. Шаблонът предполага подаване на параметри, които отговарят на корелацията между параметрите.

Както се вижда и от последния шаблон, той е доста специфичен и свързан с работата на конкретен тип производство. Затова разработчиците първо трябва да определят предназначението на близка и след това да открият подходящи шаблони. Редица платформи за облачни услуги също предлагат свои шаблони, за да улеснят използването си.

Conclusion

Предизвикателствата пред дигиталните близки се дължат на необходимост от интеграция на много технологии за реализацията им и се пренасят проблемите от разнообразните технологии. Тези проблеми вече са намерили решения в шаблоните за проектиране за тези технологии. Шаблоните подпомагат синхронизация с реалните обекти, мониторинга на мрежова инфраструктура, киберсигурност, работата на устройствата в IoT. Те реализират и допълнителни цели, като бързодействие, наличност и надеждност, безопасност, мащабируемост и ефективно използване на ресурсите.

Шаблоните съдържат решения, които са доказали своята ефикасност. Те въвеждат подходящи мерки, като дублиране на сензори, предаване на предварително обработени и

криптирани данни, съобразени с възможностите на комуникационните канали, прилагане на автентификация, механизми за сигурност и откриването на аномалии, за да се гарантира целостта на данните и киберсигурността. Основната трудност е в откриването и избора на подходящ шаблон, защото няма единна тяхна класификация. Освен това целите на някои шаблони са противоречиви и е трудно да се съчетаят в единна система.

References

1. Aleksandrova, Y., & Parusheva, S. (2021) Optimizing financial results for credit risk prediction in peer to peer lending platforms using machine learning. *Tenth International Conference on Intelligent Computing and Information Systems (ICICIS)*, Cairo, Egypt, 2021, pp. 369-374.
2. Barricelli, B., Casiraghi, E. and Fogli, D. (2019) A Survey on Digital Twin: Definitions, Characteristics, Applications, and Design Implications. *IEEE Access*. 7. pp. 167653-167671.
3. Chandra G., & Mahindra T. (2016) *Pattern language for IoT applications*. Conference on Pattern Languages of Programs (CPLoP). Available from: <https://hillside.net/plop/2016/papers/writinggroup/17.2.pdf> [Accessed 12/08/2019].
4. Čolaković, A., & Hadzialic, M. (2018) Internet of Things (IoT): A Review of Enabling Technologies, Challenges, and Open Research Issues. *Computer Networks*. 144.
5. Erl, T. & Cope, R. & Naserpour, A. (2015) *Service-Oriented Architecture: Concepts, Technology, and Design*. н.м.: Prentice Hall.
6. Fehling, C., Barzen, J., Breitenbücher, U. & Leymann, F. (2015) *A Process for Pattern Identification, Authoring, and Application*, Institute of Architecture of Application Systems. н.м.: University of Stuttgart, Germany.
7. Grieves, M., & Vickers, J. (2017) Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems. In: Kahlen, J., Flumerfelt, S., Alves, A. (eds) *Transdisciplinary Perspectives on Complex Systems*. Springer, Cham. pp. 85–113.
8. Grieves, M. (2019) Virtually intelligent product systems: Digital and physical twins, in: *Complex Systems Engineering: Theory and Practice*. American Institute of Aeronautics and Astronautics. pp. 175–200.
9. Grieves, M. (2023) *Digital Model, Digital Shadow, Digital Twin*. Available from: https://www.researchgate.net/publication/369830792_Digital_Model_Digital_Shadow_Digital_Twin [Accessed 2/09/2024].
10. Haag, S., & Anderl, R. (2018) Digital Twin – Proof of Concept. *Manufacturing Letters*. 15. pp. 64–66.
11. Hatcher, W. G. et.al. (2021) Towards Efficient and Intelligent Internet of Things Search Engine. *IEEE Access*, 9, pp. 15778-15795.
12. Jones, D., Snider, C., Nassehi, A., Yon, J., & Hicks, B. (2020) Characterising the Digital Twin: A systematic literature review. *CIRP Journal of Manufacturing Science and Technology*. 29. pp. 36–52.
13. Khail, W. S., & Schnizer, P. (2024). Patterns in Digital Twin Development. *Proceedings of the 29th European Conference on Pattern Languages of Programs, People, and Practices*, pp. 1-8.
14. Koster, M. (2014) *Design Patterns for an Internet of Things*. [Online] Available from: <http://community.arm.com/groups/internet-of-things/blog/2014/05/27/design-patterns-for-an-internet-of-things> Available from: <http://iot-datamodels.blogspot.com/2014/05/design-patterns-for-internet-of-things.html> [Accessed 12/08/2019].
15. Liu, Z. (2005) Reading behavior in the digital environment: Changes in reading behavior over the past ten years. *Journal of Documentation*. Bradford. 61(6), pp. 700–712.
16. Peter W. (2016) Communication Patterns for the Internet of Things, June 1. Available from: <https://software.intel.com/en-us/articles/communication-patterns-for-the-internet-of-things> [Accessed 12/08/2019].

17. Qanbari, S., Pezeshki, S., Raisi, R., Mahdizadeh, S., Rahimzadeh, R., Behinaein, N., Mahmoudi, F., Ayoubzadeh, S., Fazlali, P., Roshani, K., Yaghini, A., Amiri, M., Farivarmoheb, A., Zamani, A., & Dustdar, S. (2016) *IoT Design Patterns: Computational Constructs to Design, Build and Engineer Edge Applications*. IEEE First International Conference on Internet-of-Things Design and Implementation (IoTDI), pp. 277-282.
18. Qian, C. et.al. (2022) Digital Twin—Cyber Replica of Physical Things: Architecture, Applications and Future Research Directions. *Future Internet*. 14(2):64.
19. Rasheed, A., San, O., & Kvamsdal, T. (2020) Digital twin: Values, challenges and enablers from a modeling perspective. *IEEE Access*. 8, pp. 21980–22012.
20. Semeraro, C., Lezoche, M., Panetto, H., & Dassisti, M. (2023) Data-driven invariant modelling patterns for digital twin design, *Journal of Industrial Information Integration*, Vol. 31,100424.
21. Sharma, A., Kosasih, E., Zhang, J., Brintrup, A., & Calinescu A. (2022) Digital Twins: State of the art theory and practice, challenges, and open research questions. *Journal of Industrial Information Integration*.
22. Sulova, S. (2023) A conceptual framework for innovative digital data processing in the construction business. The 2st International Conference on Emerging Technology Trends on the Smart Industry and the Internet of Things January 24-25th 2023. pp. 48-53.
23. Tekinerdogan, B., & Verdouw, C. (2020). Systems Architecture Design Pattern Catalog for Developing Digital Twins. *Sensors*, 20(18), 5103.
24. Vasilev, J., Nikolaev, R., & Milkova, T. (2023) Transport Task Models with Variable Supplier Availabilities. *Logistics*. 7(3), p. 45. <https://doi.org/10.3390/logistics7030045>
25. Yadav, S.P., Bhati, B.S., Mahato, D.P., & Kumar, S. (2022) *Federated Learning for IoT Applications*. Springer.