Property Ownership Formal Modelling Using Event-B and iUML-B

Manar Altamimi[1,2][0000−0002−8789−3950], Nawfal Al Hashimy[2][0000−0002−1129−5217], Asieh Salehi Fathabadi[2][0000−0002−0508−3066], and Gary Wills[2][0000−0001−5771−4088]

[1] College of Computer Science and Information, Information System Department, Princess Nourah Bint Abdulrahman University, Kingdom of Saudi Arabia
[2] School of Electronics and Computer Science, University of Southampton, UK
{mmma2c18, Nawfal , A.Salehi-Fathabadi , gbw}@soton.ac.uk

Abstract. This paper introduces a novel approach to formal modelling and verification of ownership, addressing safety concerns in property transfer processes.The Event-B formal method, graphically represented using iUML-B notation, is used to establish a robust framework for modeling and verifying ownership systems. The verified Event-B model refines and enhances user requirements at the design stage before system implementation. The research focuses on property ownership within the legal framework of the Kingdom of Saudi Arabia, specifically property sales. The research uncovers that, despite conscientious efforts to scrutinise user requirements, the formal model development exposes limitations and inadequacies in the initial specifications. The verification process introduces essential requirements to mitigate potential fraudulent activities, enhancing the security and dependability of ownership claims.

Keywords: ownership · safety · formal methods · Event-B · iUML-B

1 Introduction and Motivation

Proof of ownership is sensitive and valuable information in land registration systems. Land registration systems are complex and comprise numerous interconnected entities [16,11]. The presence of this complexity resulted in a dearth of availability of pertinent property information and a failure to prove ownership. Roughly 70% of the global population lacks access to cost-effective mechanisms for safeguarding their ownership [8]. The procedure of transfer ownership has been developed based on a framework that was conducted by our earlier study [3]. The procedure presents challenges in three primary forms: a potential sale of ownership to multiple owners, an inconsistency of property ownership, and a risk of fraudulent activities, such as identity theft.

Problem Statement: *The oversight of neglecting critical safety considerations within ownership transfer processes leaves the process vulnerable to fraudulent activities that pose typical challenges within the realm of safety-critical cyber-physical systems.*

This paper addresses the challenges arising from the complexity of the legal process in transferring ownership by constructing the Event-B formal model [1]. The model accurately represents the component architecture involved in the process of transferring property ownership, including interactions with stakeholders and other system components. The research question guiding this investigation is: *How can challenges in land registration systems, including double sales, falsification susceptibility, and the risk of fraudulent activities, be effectively addressed?*
The initial step in the process is identifying the approach to construct the model in Section 3. This involves a description of user needs in Section 3, identifying the strategy to construct the model, and modelling the process in Section 4. Lastly, we verify the model against the challenges in Section 5.

2 Related Work

Multiple scholarly inquiries have examined how technology can be used to tackle the difficulties related to property ownership in land registration systems [15,4,10]. The work in [15] asserts that the implementation of a distributed title database in a land registry would offer a theoretically secure and distributed solution to tackle issues. The work in [4] asserts that the primary purpose of electronic records in land registry, facilitated by a digital signature, is to allow users to transfer property ownership online, eliminating the requirement for physical presence. The work in [10] shows that the integration of blockchain technology into property ownership enhanced security and immutability. Nevertheless, these approaches fail to take into account the process associated with improving property ownership.

To enhance the process, [6] has opted iUML-B to address concerns related to safety-critical cyber-physical systems within railway control systems. In [7], have been employed formal methods to enhance safety and security standards in autonomous missions to overcome challenges associated with system specification. While these approaches offer solutions for system specification difficulties, their use cases are often limited to railway or autonomous driving scenarios. This paper, on the other hand, utilises the Event-B formal model to integrate safety requirements into the transfer ownership process, effectively mitigating challenges and tackling the complexity of an industrial system.

3 Overview of Modelling Approach

The approach used to simulate the transfer of property ownership involves four distinct stages. The initial stage includes understanding the specifications of ownership and determining the refinement strategy, followed by the construction of the model. Refinement strategy and modeling are iterative processes until an appropriate refinement strategy is achieved. The model is verified using model checking and proof of obligation to ensure compliance with the requirements specification and the consistency of the model. The approach depicted in Figure

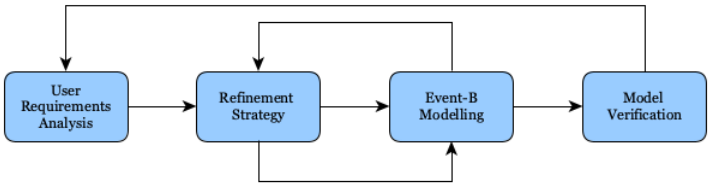


Fig. 1: Model development approach.

1 demonstrates the sequential delivery of each stage. However, it is important to note that this strategy is implemented iteratively. The iterative process plays a crucial role in enhancing our comprehension of ownership by facilitating the selection of an optimal refinement strategy for constructing the model. We give a concise description of the first stage in this Section and describe the following stages in Section 4 and Section 5.

The process of transferring ownership in Saudi Arabia involves four steps: ownership declaration, claiming ownership, the process of transferring ownership, and conveyance. The first step involves identifying the user as a purchaser, owner, or seller. The owner must **claim** their ownership title, which can be transferred to the purchaser through legal means. The process of transferring ownership involves specifying the seller as a seller or allowing the seller to **sell** the property. The purchaser can **request** multiple properties at once, and the seller can accept or withdraw the transaction. The **conveyance** step involves verifying the proof of purchase and ensuring the original and timestamped property information is original to the source. In Figure 2, we give a concise description of the process using an activity diagram.

However, the scenarios have not shown control requirements to avoid double sales, inconsistency in property information, and fraud. Double sales involve consecutive sales by the same seller to different purchasers, requiring stricter control measures to prevent multiple buyers. Inconsistency in information arises from inefficiencies in recording property information and incomplete records. Owners should claim ownership before registering a property and organise it to reflect changes in ownership. Fraud concerns the intentional misuse of assets, such as stealing identities, and requires verification of user identities. Considering the challenges in the process, Event-B modelling contributes to enhancing the process and exposes limitations and inadequacies in the initial specifications.

4 Event-B Model and Refinement Strategy

The refinement strategy is correct-construction [6,5], achieved using Event-B modeling. The modelling is graphically presented in six refinements using iUML-B. The strategy constructs the model gradually, and every refinement addresses one aspect of the process [3].

[3] The complete model can be accessed at: `https://shorturl.at/cfqxS`

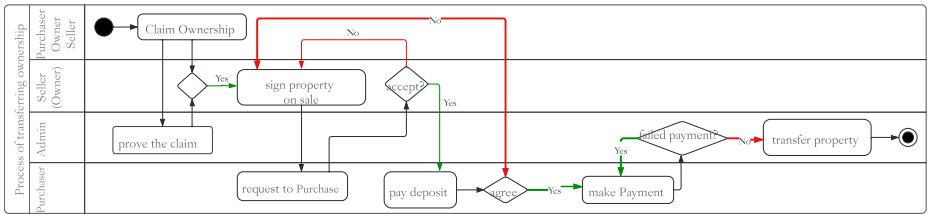
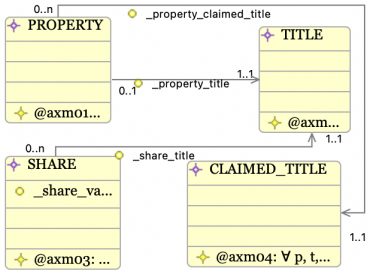
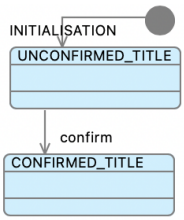


Fig. 2: Activity diagram: the process of transferring ownership



(a) Class diagram: main components



(b) State machine: title states

Fig. 3: iUML-B Diagrams: Abstract level

Background knowledge: Event-B [1,9] is a formal method for system development that consists of two types of component: a context and a machine. Contexts represent static data, while machines represent dynamic data. Contexts include carrier sets, constants and axioms (constrain the carrier sets and constants). A machine can access the static part of the model, including s, invariants (constraining variables) and events. An event defines a transition activity with pre-conditions (guards), and actions which modify variables. Event-B is supported by Rodin toolset [2], for modeling, verification. Model verification ensures model correctness and refinement consistency; validation ensures the construction of the right model using model checking. iUML-B [12,13,14] is a diagrammatic modelling notation that provides state machines and class diagrams to represent the Event-B model. It generates Event-B elements automatically and visually presents the model.

Abstract Model:Property The ownership model involves property and title, with each property linked to a title listing all owners' shares, as shown in Figure 3a. The title has two states indicating the status of ownership claims. When owners claim ownership, the status of ownership is confirmed or remains claimed; otherwise, ownership remains unconfirmed or unclaimed, see Figure 3b. These details have not been shown in the analysis. However, they were achieved through the iterative development of modeling.

Forward composition: $p ; q$

$$\forall p,q \cdot p \in S \leftrightarrow T \land q \in T \leftrightarrow U \Rightarrow p ; q = \{x \mapsto y \mid (\exists z \cdot x \mapsto z \in p \land z \mapsto y \in q)\}$$

Range restriction: $r \rhd T$ $r \rhd T = \{x \mapsto y : x \mapsto y \in r \land y \in T\}$
Inverse: $r\sim$ $r\sim = \{y \to x : x \to y \in r\}$

Four sets or classes are specified in the context, as shown in Figure 3a. The relationships show the association between sets, and the cardinality indicates the type of association. Every PROPERTY is associated with one TITLE. The TITLE lists owners' shares SHARE. The CLAIMED_TITLE is a record where owners claim their ownership. These specifications are specified as axioms: @axm01: $\_equivalent\_to = \_property\_claimed\_title^{-1}; \_property\_title$, this ensures that the title and claimed title are associated with the same property.

The title's states are modelled in the machine, Figure 3b. The state moves to claimed or confirmed when all owners claim ownership using the event confirm. The state of the title is safely controlled, specified as an invariant:
@inv01: partition(TITLE, UNCONFIRMED_TITLE, CONFIRMED_TITLE)

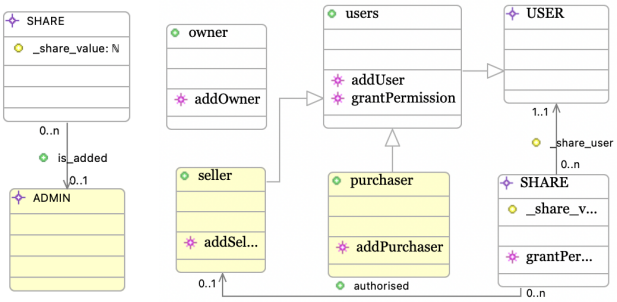
First Refinement: Property Control The abstract context is extended, and a new set, ADMIN, is introduced, Figure 4a. The model considers admin to be the entity responsible for maintaining property information. admin can addTitle and addShares, as well as confirm ownership when the user claims their property.

In the refining machine, the admin users added the TITLE(@inv03) using event addTitle. The same admin added the SHARE(@inv02) using event addShare.
@inv02: $is\_added \in SHARE \pfun ADMIN$
@inv03: $title\_adm \in TITLE \pfun ADMIN$

Further constraints (@inv04 and @inv05) are introduced to control the property: @inv04: $\forall t \cdot t \in CONFIRMED\_TITLE \Rightarrow t \in dom(title\_adm)$
@inv05: $\forall s \cdot s \in dom(share\_title) \Rightarrow s \in dom(is\_added) \land \_share\_title(s) \in dom((title\_adm; is\_added^{-1}) \rhd \{s\})$.

Accordingly, the confirm, addTitle, and addShare events are not satisfied with the new invariants. Therefore, guards are added to ensure that only the admin can confirm the property and to make the current refinement consistent.
event addShare: @grd01: $share \in dom(is\_added)$
event confirm: @grd01: $title \in dom(title\_adm)$

Second refinement: Ownership This refinement entails modelling ownership and its association with owners. This ensures consistency of property information with owners that reflects changes over time. Ownership represents the proportion of ownership that each owner possesses. An owner could claim ownership of their property by providing proof of their claimed title. After confirming ownership, the title becomes attainable for sale. This refinement describes the procedure for proof of ownership, as mentioned in the previous analysis, before conducting any conveyance transaction. The context is extended in Figure 5. New sets are introduced: a USER associate it with SHARE, specified as the axiom: @axm04: $\_unconfirmed\_ownership = \_share\_title^{-1}; \_share\_user$
Only owners can claim ownership if and only if they have a copy of the claim title that is equivalent to the title:
@axm05: $\forall t, ct, p \cdot p \in PROPERTY \land t \in ran(\_share\_title) \land ct \in CLAIMED\_TITLE$



(a) First refinement (b) Third refinement

Fig. 4: Class diagrams: yellow: new classes, white: abstract classes

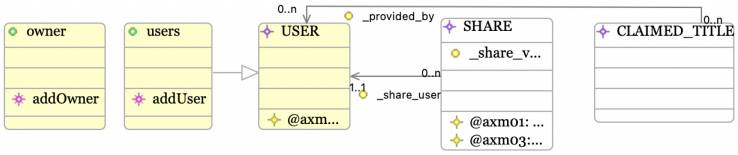
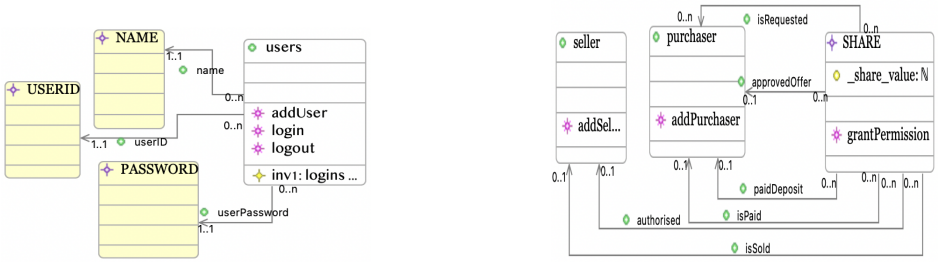


Fig. 5: Second refinement: part of iUML-B class diagrams

$\wedge\ p \mapsto t \in \_property\_title \wedge ct \in dom(\_equivalent\_to) \wedge \_equivalent\_to(ct) = t$
$\Rightarrow \_provided\_by[\{ct\}] = ran(ran(\_share\_title \rhd \{t\}) \lhd \_unconfirmed\_ownership)$
The machine is refined to support the specification in this context. The ownership structure consists of three components: share_title, share_user, and ownership. We define users according to USER type to represent system users. share_title is the number of shares. share_user is an association of users with shares. ownership is an association title with users. One type of user role, the owner, is introduced at this level: ran(ownership). An archive of ownership inv06 should be kept as part of the requirements, specified as an invariant:
@inv06: archiveOwnership $\in$ share_title $\leftrightarrow$ users
inv07 is an archive information is a timestamp of transfer ownership:
@inv07: archiveDate $\in$ archiveOwnership $\rightarrow$ DATE
We model this by initially assigning all shares to the model's user and then updating to the transfer date.

Third refinement: User Type The context remains unchanged while the machine is refined to introduce new variables, purchaser and seller to represent the roles of users at different states in the model Figure 4b. Although the user can play all roles, they cannot simultaneously be the seller and purchaser in the same transaction. When an owner intends to sell the property, they must grant permission grantPermission either to themselves or someone else:
@inv08: authorised $\in$ SHARE $\nrightarrow$ seller
purchaser and seller can be added by using events addPurchaser and addSeller, respectively. We explicitly define the purchaser and seller as types of users to maintain the consistency of the model.



(a) Fourth refinement (b) Fifth refinement.

Fig. 6: Part of iUML-B Diagrams: class diagrams

Fourth refinement: User Identity This refinement focuses on preventing fraudulent activity by enforcing a security policy. Each user is given NAME, USERID, PASSWORD, see Figure 6a. A safety property is added to the model to verify the user's identity. This can be modelled using the variable logins and adding two events: login and logout. The logins variable is a subset of user. When the user logs in, they should declare their userid and password. logout is basically removing the user from logins set.

Fifth refinement: Ownership Process This refinement involves the modelling of the property acquisition process while preventing double sales in the process, Figure 6b shows the static part, while 7 shows the transition part. The process commences when the state of title is CONFIRMED_TITLE. CONFIRMED_TITLE can only exist in four states. The initial state is NOT_FOR_SALE once the title is confirmed by ADMIN using event confirm. Each transition is carried out by multiple users, each fulfilling their assigned roles. For example, a seller engages in the activities of sell and acceptedOffers. Given that a confirmed title cannot be associated with several states throughout the purchasing process, it is crucial to specifically define the user's role and their connection to ownership. This will ensure that distinct users are identified for different states. For instance, multiple purchasers can make requests to acquire a property (@inv09), but only one request is accepted (@inv10):

@inv09: $isRequested \in SHARE \leftrightarrow purchaser$

@inv10: $approvedOffer \in SHARE \nrightarrow purchaser$

Safety invariants are included to assure the safety process, some of them are:

@inv11: $dom(isRequested) \subseteq dom(isSold)$

@inv12: $approvedOffer \subseteq isRequested$

@inv11 and @inv12 ensure that any title that is requested should be on sale and only an approved offer is being requested, respectively.

5 Model Verification

This section summerises theorem proving and model checking effort of the presented Event-B model of the ownership supported by the Rodin toolsets.

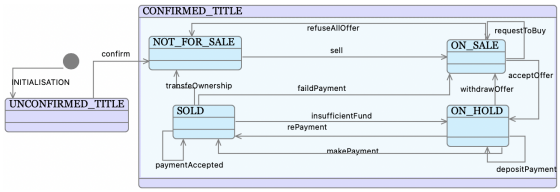


Fig. 7: Fifth refinement: iUML-B state machine

Theorem Proving The Event-B theorem proving technique uses invariant preservation (e/v/INV) Proof Obligation (PO) and guard strengthening (e/g/-GRD) PO to ensure concrete guards are stronger than abstract ones. A model of six machines yielded 233 POs, with 86% of cases automatically proven using the Rodin prover. However, there are some POs, mostly generated to prove invariant preservation, that are not discharged automatically and require interactive (manually) proving. The majority of POs occur in the last refinement, with 65 POs manually proved. An example of invariant preservation manually proved is $\mathsf{isSold} \in \mathsf{SHARE} \rightarrow \mathsf{seller}$ when executing sell. This invariant is discharged when we ensure that the share of ownership is not on sale by adding a guard to the event. The invariant prevents the resale a share that is already on sale.

ProB Model Checking ProB in Event-B is an effective tool for assessing probabilistic behaviours in system models. It helps capture and evaluate uncertainties within the model, allowing for exploration of potential system states and behaviors. This tool captures both deterministic and stochastic aspects, fostering a more realistic representation of complex systems during the construction phase. For example, (...grantPermission, sell, requestToBuy...) are sequences of events that demonstrate the scenario to address double sales for the same seller and property by ensuring that the seller cannot resell the ownership multiple times:

at time ^{i}: sell and requestToBuy are not active.
at time $^{i+1}$: grantPermission and requestToBuy are not active.
at time $^{i+2}$: grantPermission and sell are not active.

6 Conclusion and Future Work

The paper addresses the research question of how effectively tackling challenges at different refinement levels, focusing on property ownership inconsistency, reducing fraudulent activities, and double sales. It uses formal methods to understand the legal complexity of transferring ownership, reveal inadequate system specifications, and introduce missing requirements to mitigate potential fraudulent activities. These requirements improve coherence and clarify user scenarios, ultimately enhancing the land registration systems.

The formal modelling and verification approach for ownership can be expanded to identify, analyse, and mitigate security risks in systems. Further research could generalise the model to cover jointly owned assets like luxury jewellery and harvesters, enhancing its potential for enhancing system security.

References

1. Abrial, J.R.: Modeling in Event-B: system and software engineering. Cambridge University Press (2010)
2. Abrial, J.R., Butler, M., Hallerstede, S., Hoang, T.S., Mehta, F., Voisin, L.: Rodin: an open toolset for modelling and reasoning in Event-B. International journal on software tools for technology transfer **12**(6), 447–466 (2010)
3. Altamimi, M., Al Hashimy, N., Wills, G.: Expert review of the land registration framework in the kingdom of saudi arabia. International Journal of ICT Research in Africa and the Middle East (IJICTRAME) **11**(1), 1–18 (2022)
4. Brennan, G.: Defining Title Registration. In: The Impact of eConveyancing on Title Registration, pp. 115–151. Springer International Publishing, Cham (2015)
5. Dghaym, D., Hoang, T.S., Turnock, S.R., Butler, M., Downes, J., Pritchard, B.: An stpa-based formal composition framework for trustworthy autonomous maritime systems. Safety science **136**, 105139 (2021)
6. Dghaym, D., Poppleton, M., Snook, C.: Diagram-led formal modelling using iuml-b for hybrid ertms level 3. In: Abstract State Machines, Alloy, B, TLA, VDM, and Z: 6th International Conference, ABZ 2018, Southampton, UK, June 5–8, 2018, Proceedings 6. pp. 338–352. Springer (2018)
7. Dghaym, D., Turnock, S.R., Butler, M.J., Downes, J., Hoang, S., Pritchard, B.: Developing a framework for trustworthy autonomous maritime systems. Proceedings of the International Seminar on Safety and Security of Autonomous Vessels (ISSAV) and European STAMP Workshop and Conference (ESWC) 2019 (2020), https://api.semanticscholar.org/CorpusID:209056678
8. Group, W.B.: Enhancing public sector performance: Malaysia's experience with transforming land administration. World Bank (2017)
9. Hoang, T.S.: An introduction to the Event-B modelling method. Industrial Deployment of System Engineering Methods pp. 211–236 (2013)
10. Peck, M.E.: Blockchain world - do you need a blockchain? this chart will tell you if the technology can solve your problem. IEEE Spectrum **54**(10), 38–60 (2017). https://doi.org/10.1109/MSPEC.2017.8048838
11. Rizal Batubara, F., Ubacht, J., Janssen, M.: Unraveling transparency and accountability in blockchain. In: ACM International Conference Proceeding Series. pp. 204–213. 20th Annual International Conference on Digital Government Research (dg.o 2019) (2019). https://doi.org/10.1145/3325112.3325262, https://doi.org/10.1145/3325112.3325262
12. Said, M.Y., Butler, M., Snook, C.: A method of refinement in uml-b. Software & Systems Modeling **14**(4), 1557–1580 (2015)
13. Snook, C.: iuml-b statemachines new features and usage examples. In: Proceedings of the 5th Rodin User and Developer Workshop. University of Southampton (2014), https://eprints.soton.ac.uk/365301/
14. Snook, C., Butler, M.: Uml-b: Formal modeling and design aided by uml. ACM Transactions on Software Engineering and Methodology (TOSEM) **15**(1), 92–122 (2006)
15. Szabo, N.: Secure Property Titles with Owner Authority. Nakamoto Institute pp. 1–5 (1998), https://nakamotoinstitute.org/secure-property-titles/
16. Zevenbergen, J.: A Systems Approach to Land Registration and Cadastre. Nordic Journal of Surveying and Real Estate Research **1** (2004)