electronics

MDPI

Article

A Unified Forensic Model Applicable to the Database Forensics Field

Amel Ali Alhussan [1], Arafat Al-Dhaqm [2], Wael M. S. Yafooz [3], Abdel-Hamid M. Emara [3,4], Shukor Bin Abd Razak [2] and Doaa Sami Khafaga [1,*]

[1] Department of Computer Sciences, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia; aaalhussan@pnu.edu.sa
[2] School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia (UTM), Johor Skudai 813110, Malaysia; mrarafat1@utm.my (A.A.-D.); shukorar@utm.my (S.B.A.R.)
[3] Department of Computer Science, College of Computer Science and Engineering, Taibah University, Medina 42353, Saudi Arabia; wyafooz@taibahu.edu.sa (W.M.S.Y.); aemara@taibahu.edu.sa (A.-H.M.E.)
[4] Department of Computers and Systems Engineering, Faculty of Engineering, Al-Azhar University, Cairo 11884, Egypt
* Correspondence: dskhafga@pnu.edu.sa

Abstract: The Database Forensics Investigation (DBFI) field is focused on capturing and investigating database incidents. DBFI is a subdomain of the digital forensics domain, which deals with database files and dictionaries to identify, acquire, preserve, examine, analyze, reconstruct, present, and document database incidents. Several frameworks and models have been offered for the DBFI field in the literature. However, these specific models and frameworks have redundant investigation processes and activities. Therefore, this study has two aims: (i) conducting a compressive survey to discover the challenges and issues of the DBFI field and (ii) developing a Unified forensic model for the database forensics field. To this end, the design science research (DSR) method was used in this study. The results showed that the DBFI field suffers from many issues such as the lack of standardization, multidimensional nature, heterogeneity, and ambiguity, making it complex for those working in this domain. In addition, a model was proposed in this paper, called the Unified Forensic Model (UFM), which consists of five main stages: initialization stage, acquiring stage, investigation stage, restoring and recovering stage, and evaluation stage. Each stage has several processes and activities. The applicability of UFM was evaluated from two perspectives: completeness and implementation perspectives. UFM is a novel model covering all existing DBFI models and comprises two new stages: the recovering and restoring stage and the evaluation stage. The proposed UFM is so flexible that any forensic investigator could employ it easily when investigating database incidents.

Keywords: database forensic; digital forensic; design science research; model

Citation: Alhussan, A.A.; Al-Dhaqm, A.; Yafooz, W.M.S.; Emara, A.-H.M.; Bin Abd Razak, S.; Khafaga, D.S. A Unified Forensic Model Applicable to the Database Forensics Field. *Electronics* **2022**, *11*, 1347. https://doi.org/10.3390/electronics11091347

Academic Editor: Prasan Kumar Sahoo

Received: 12 March 2022
Accepted: 17 April 2022
Published: 23 April 2022

1. Introduction

Database Forensic Investigation (DBFI) is a branch of digital forensics that examines database content to confirm database incidents. It is considered a significant field to identify, detect, acquire, analyze, and reconstruct database incidents and reveal intruders' activities [1]. It has suffered from several issues, which have resulted in it becoming a heterogeneous, confusing, and unstructured domain. Examples of these issues include a variety of database system infrastructures, the multidimensional nature of database systems, and domain knowledge effectively being scattered in all directions. Various database system infrastructures with multidimensional natures have enabled the DBFI domain to address specific incidents. Therefore, each database management system (DBMS) has a straightforward forensic investigation model/approach. Consequently, the issues of

different concepts and terminologies in the forensic investigation process and the scattering of domain knowledge have produced other challenges for domain forensic practitioners. Therefore, this study has two contributions:

(i) Conducting a comprehensive review to discover the research directions of the DBFI field;
(ii) Developing a unified forensic investigation process model to solve the complexity, interoperability, and heterogeneity of the DBFI domain.

The novelty of this study lies in the development (and applicability testing) of five simplified conceptual models for the DBFI domain in the unified model called UFM. Each conceptual model of the UFM can work separately. For example, the first sub-model detects the database incidents and the investigation team's required investigation procedures and policies. The investigation team uses the second conceptual sub-model to acquire, preserve, and examine the volatile and non-volatile data from the victim database. The third sub-model allows the investigation team to rebuild, analyze, and document all database events to discover the criminal activities. The fourth model is used to restore and recover the database continuity and mitigate the risk, whereas the fifth model improves the existing investigation process and trains the staff. The fourth and fifth conceptual submodules are new in the DBFI domain and have not been widely reported in the current literature.

The advantages of the proposed model are to provide a clear picture and main investigation reference for database forensic investigators on how to deal with different database incidents. Thus, entire investigation processes and tasks are available in this unified model, reducing the ambiguity and confusion among database forensic investigators.

The remainder of this paper is structured as follows: Section 2 discusses the related work, while Section 3 discusses the methodology. After that, Section 4 gives the results and discussion. Section 5 discusses the advantages of the proposed model. Finally, the conclusion and future directions are given in Section 6.

2. Related Work

A review of the relevant literature can result in several models and frameworks proposed in the DBFI domain. On the other hand, the researchers in [2–6] maintained that the database forensics models might face failure when examining the database systems. Such failure could occur because of the multidimensionality of database systems and the diversity of database management systems (DBMSs). In addition, database forensics concentrates upon a single dimension, i.e., file system. This dimension is mainly hinged on determining, collecting, handling, storing, responding to incidents, and training [2]. In some cases, however, it is not easy to detect database incidents without productive cooperation among digital investigators regarding analyzing a database [2].

Moreover, practices and processes in database forensics do not encompass the transactional database features. The diversity and multidimensionality of DBMSs cause a big challenge to the development of a standardized approach to database forensics. As a result, the present digital forensics models cannot encompass all concepts within database systems [7]. Generally, the mainstream research conducted in the database forensics field focuses on recovering database contents and metadata. This necessitates carrying out different tasks regarding documenting the evidence collected from database incidents [8,9].

Several forensic investigation models in the existing literature concentrate on the Oracle Database. For instance, in [10], a proposed model demonstrated how an Oracle log file could be applied to revealing the attacking events. In that study, the authors investigated the redo logs' binary format, which shows where the required evidence could be found. In addition, they attempted to determine the best way to integrate the evidence into an event timeline. They also suggested how an attacker attempts to cover their tracks by considering an already failed attack and spotting it. The authors in [11] explained how to recover already deleted evidence (in the case of the Oracle objects). This aids examiners in an indirect recovery of evidence from the data files of a compromised server. In addition, a malevolent entity can also drop the objects, although an examiner can use the

Oracle DB Views and Tables (for instance, SOURCE$, IDL_UB1$, OBJ$, IDL_CHAR$, and RECYCLEBIN$ tables) to find the location of the dropped objects. In [12], the researchers introduced a forensic model applicable to capturing the attack-related evidence against authentication mechanisms.

The Listener's log file and the audit trail are leveraged in that model. The log file comprises some details regarding the connection with the database server, e.g., the Internet Protocol (IP) address, the instance name, and the name of the Service Identifier (SID). The audit trail consists of details about the successful and unsuccessful attempts in login and logoff activities. Thus, inspectors can gather evidence from the Listener's log file and the audit trail against the authentication mechanism. This is established by assuming that the audit trail is enabled within the corresponding DB. The authors in [13] proposed another forensic model that focuses on the database servers' disconnection from the network for the obtainment of volatile data. For the recovery of fragile data from the database server, two investigation processes, i.e., the Identification process and the Evidence Collection process, were suggested. During the former, the investigators disconnect the database server from the network and forensic environment and then use techniques to move the already captured data. However, during the Evidence Collection process, the investigator collects volatile data from compromised database servers. There is a need for forensics research to carefully recover and store volatile data and make the data ready for later analyses. This way, forensic investigators could collect non-volatile data in a "human-readable" form; this form of data could be more easily observed than stored binary data. Then, in [14], the detection investigation forensic model was proposed. It highlights how examiners can explore evidence of data theft in the absence of auditing. They modelled the ways to determine an Incident Responder/DBA if an Oracle Database server breach has happened without any audit trail, assuming attackers have attained unauthorized select access to data.

In 2008, the SQL server forensic analysis method was proposed [15], which was applied to collecting and analyzing the evidence obtained from the MSSQL server database. That method comprised four phases: preparation of the investigation, verification of the incident, collection of artefacts, and analysis of the artefacts collected in the previous phase. The method was entirely concentrated on the SQL server database. In addition, the researchers in [16] proposed another database server detection and investigation process model. Their most important goal was to detect the database servers and gather necessary data. Their model consisted of server detection, data collection, and data analysis. On the other hand, its drawback was its incapability to work on volatile artefacts.

In [17], the authors introduced a new model, i.e., the detection inconsistencies database model, to identify and name the bytes and interpret them for the MySQL database system. This helps users to detect the discrepancies appearing in a database. However, Khanuja and Adane [8] asserted that there is no literature knowledge for multiple log files and cache for more analyses. They used the MySQL database server log artefacts. Furthermore, the authors in [18] proposed a reconstruction model applicable to reconstructing the basic SQL statements from redo logs that restore the formerly deleted or updated values. The drawback of the model was that it was focused on the DML statements and ignored the basic DDL statements. In [19], a practical forensic approach was suggested to reconstruct the basic SQL DDL statements to enhance the previous approach. The authors in [8] built a framework to identify, gather, analyze, validate, and document digital evidence to detect malicious tampering. It was composed of three phases: collection and analysis of non-volatile data; collection, analysis, and reconstruction of volatile data; and comparison of the obtained results. Apart from various database forensic domain knowledge projected for DBMSs, the literature also consists of several forensic tamper detection models and algorithms for the database systems analyses. For example, in [20], the authors introduced discovering methodology and a scenario to detect covert database systems to aid inspectors in the detection of covert database systems. In [21], a new model was proposed that was able to gather digital evidence effectively. It can collect evidence from a database business

environment to be applied to authorized and unauthorized events investigations. The model employs different database features such as replication, triggers, and log file backup. The researchers in [22] scientifically built a forensic tamper detection model that could detect a compromised database audit log using a strong one-way hash function. However, the weak point of their model was its incapability of analyzing intruders' activities, deciding the time tampering occurs, determining which data have been altered, and effectively identifying the adversary. This model was primarily designed to investigate a compromised database management system. The model involves two investigation processes: (1) identification, which is dedicated to the preparation of database forensic layers, methods, and the forensic setting, and (2) collection, which involves the collection of doubted data from the database management system and their transfer to a secure place to be exposed to further forensic inspections. In [23], a model was introduced to gather, preserve, and analyze the database metadata against database attacks. It consisted of four processes: collection and preserving, analysis of the antiforensic attacks, analysis of the database attacks, and preservation of the evidence reports. In [24], the authors introduced a new model capable of creating database events in such a way that the intruder activities could be well recognized. The model comprises two processes: collection and reconstruction of the evidence. The process of collection involves gathering the evidence by replicating sources. On the other hand, the reconstruction process involves reconstructing the users' activities and detecting malevolent activities.

Moreover, the literature consists of many forensic algorithms and tools applicable to the database forensics domain. For instance, with the help of a strong one-way hash function, the researchers in [22] could detect tampering with the database audit log. This way, all compromised database audit logs could be well detected. On the other hand, their proposed algorithm cannot analyse intruder activities, determine the exact time of tampering, recognise the altered data and identify the adversary. For that reason, several researchers have attempted to build effective forensic analysis algorithms to fill this gap. Examples of such forensic algorithms include Red Green Blue (RGB), Monochromatic, Tiled-Bitmap, a3D, and Red Green Blue Yellow (RGBY) algorithms. They offer various capacities for data analysis regarding the time and cost requirements. For instance, the Monochromatic algorithm can detect one corruption event, while RGB is capable of detecting two corruption events. Note that RGBY can detect even more such events but with false alarms.

However, the literature contains only a few forensic tools applicable to the database forensics field, e.g., SQL Profiler (MS SQL Server) [25], ProfilerEventHandler (My SQL) Khanuja and Adane [8], and Log Miner (Oracle DB) [26]. By using SQL Profiler, system administrators would be able to monitor events in an instance of MS SQL Server. This graphical tool can collect and keep information regarding operations/events in a file or SQL Server table for further analysis. The ProfilerEventHandler, a MySQL tool, uses the interface to handle the profiling and tracing of the events [8]. Wright [26] developed the Log Miner tool, which helps a DBA or forensic investigator to rebuild the actions that happened in a database.

In addition, the present study reviewed the existing forensic works concentrating on the NoSQL database systems. For instance, in [7], the authors developed a forensic investigation framework for the document storage of NoSQL DBMS based on its unique features. Their framework contains five phases: preparing, acquiring and preserving, identifying the distributed evidence, examining and analyzing, and finally, reporting and presenting. However, it does not evaluate the scheme of a database or database forensic characteristics, such as collecting logs to evaluate the operations. The authors in [27] designed a forensic tool applicable to investigating the internal structure and data file format of one of the most extensively employed NoSQL DBMSs, i.e., MongoDB. In addition, they attempted to use their tool to find a method for the recovery of already-deleted data. However, the tool cannot support WiredTiger, the default storage engine in MongoDB Versions 3.2 and higher.

In addition to the current work in database forensics, the literature consists of a few review/survey studies. For instance, in [28], the authors reviewed the database forensic investigation processes to provide a resource for database forensics from different perspectives. They also discussed the challenges and limitations of the existing models and offered several solutions to the problems discussed. In another study [29], the research on database forensics from 2009 to 2015 was reviewed. To this end, the authors searched eight search engines, which resulted in only 282 articles. However, their review was not associated with any discussion on challenges, limitations, directions, or solutions to the problems in the domain of database forensics. The authors in [30] carried out a systematic literature review on the same domain from 2015 to 2017. They only employed two search engines to collect required data: IEEE Explore and Science direct. They designed a forensic analysis model that comprised 13 stages: identification, preparation, comparison, recovery, distribution, acquisition, carving, collection, restoring audit log, determination of events, examination and presentation, documentation, and reporting. In [31], the researchers proposed an investigation process model performing definite tasks to explore relevant information on operations carried out on the Oracle Database concepts. They considered four research processes in their model: cancellation of the database operation, collection of data, reconstruction of a database, and fixing the database integrity. Moreover, the authors in [21] designed the Log Miner tool applicable to the Oracle database to reconstruct the actions when the auditing features are turned off.

The present survey discovered that the database forensics field had been discussed in the literature from four perspectives (as shown in Figure 1): database forensics dimensions, database forensics investigation process, database forensics technology, and database forensics knowledge management.

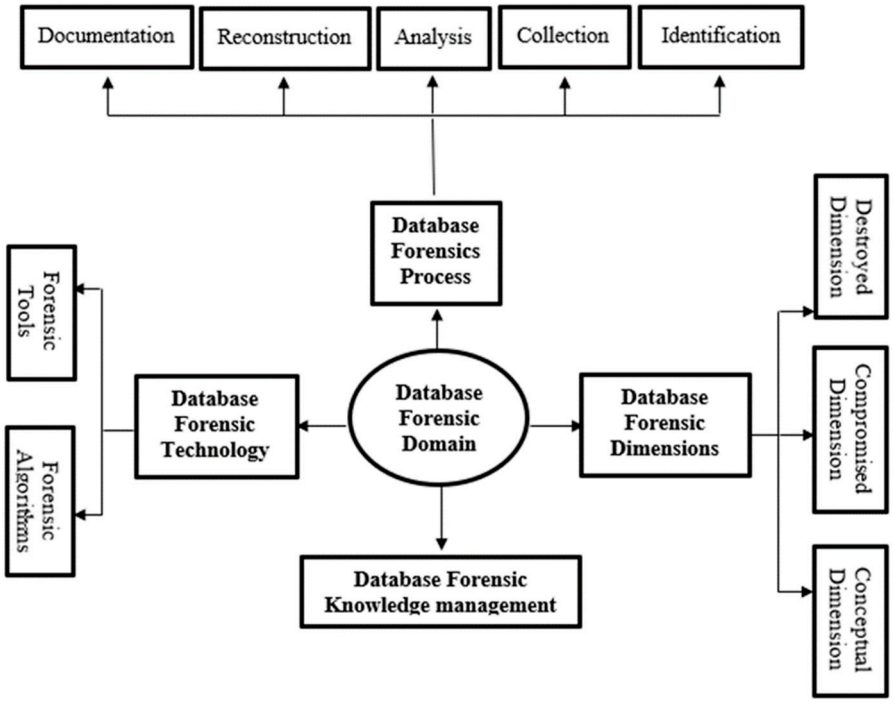

Figure 1. Database forensics perspectives.

3. Methodology

To achieve the objectives of this study, a design science research method was adapted from [32]. It is a suitable methodology for developing a model that contributes to the growth of knowledge in the domain. Thus, the design science research method is used to develop and validate the UFM. It consists of five steps, as shown in Figure 2:

(1) Step 1. Identifying and nominating domain models.

(2) Step 2. Gathering domain processes.
(3) Step 3. Manipulating and combining gathered processes.
(4) Step 4. Identifying the relationships.
(5) Step 5. Validation and Implementation.

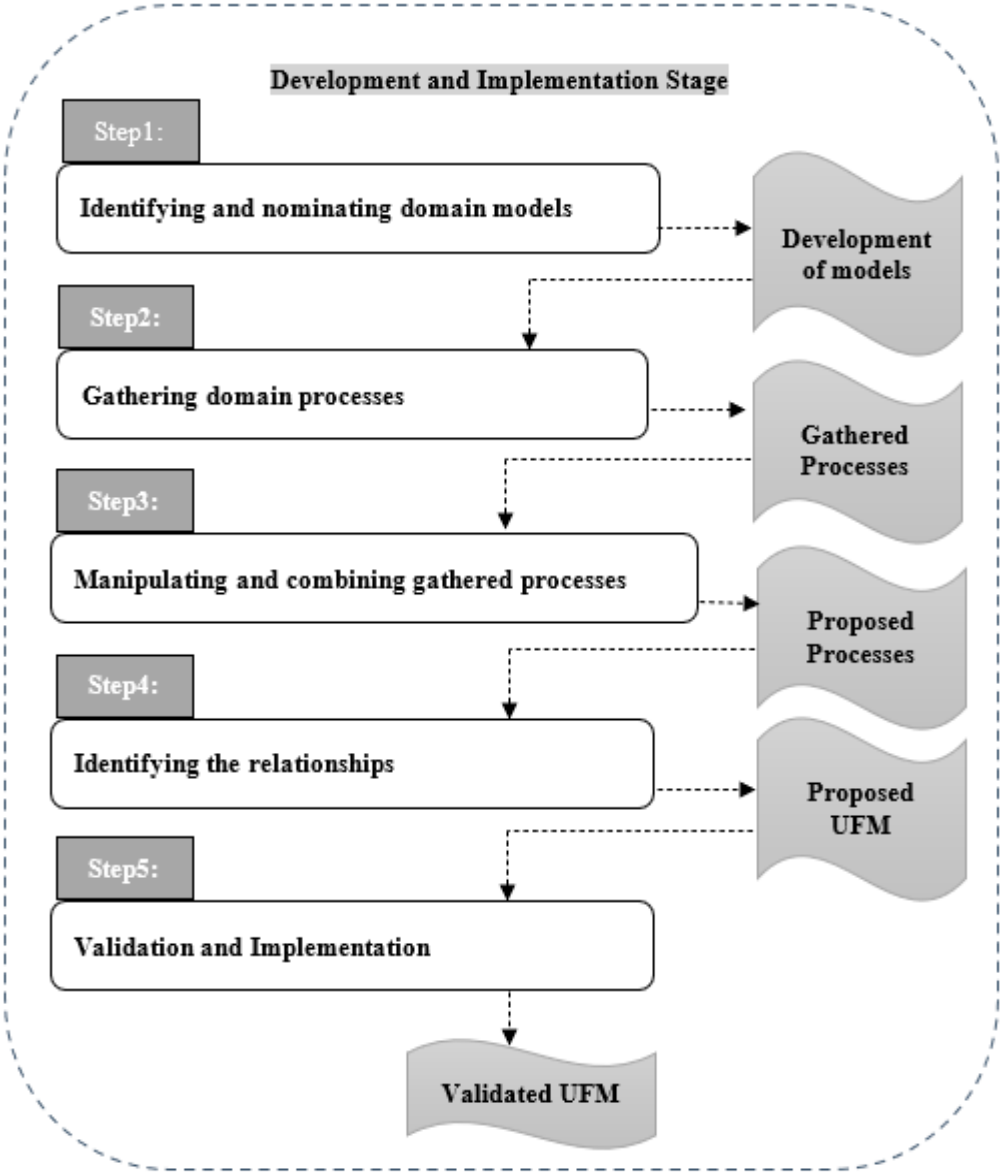

Figure 2. The design science research method.

(1) Step 1. Identifying and nominating the domain models: This step involves identifying and selecting the DBFI models for development and validation purposes. Based on the coverage metrics [28,33], eighteen models were identified and nominated for development and validation purposes, as shown in Table 1.

Table 1. Identified development and validation Models.

ID	Nominated Models	Extracted Processes
1.	[34]	Detection process, Collection process, Reconstruction process, Restoring, and Recovering
2.	[35]	Authentication, System Explanation, Evidence Collection, Timeline Construction, Analysis process, Recovering process, and Searching.
3.	[14]	The preparation process, The ollection process
4.	[36]	Preparation process, Incident Confirmation, Collection process, Analysis process
5.	[37]	Preparation of Database Environment, Extraction process, Investigation process
6.	[38]	Acquirement process, Investigation process, Financial Analysis
7.	[9]	The extraction process, Restoration process
8.	[16]	Server Discovery process, Gathering process, The examination process
9.	[39]	Investigation process, Collection process, Analysis process
10.	[8]	Identification process, Acquiring process, Investigation process, Reporting process
11.	[5]	Crime Reporting process, Examination process, Physical Examination process, Digital Examination process, Documentation process, Postinvestigation process, and Postinvestigation, Analysis Process
12.	[6]	Preparation process, Defining/Gaining process, Artefact Collection process, Volatile Collection process, Nonvolatile Collection process, Preservation process, Analysis process
13.	[35]	Collection process, Analysis process
14.	[4]	The preparation process, Collection process
15.	[23]	Collection process, Preservation process, Analysis process
16.	[24]	Gathering process, Examination process
17.	[36]	Investigation process, Rebuilding process
18.	[40]	The rebuilding process, Recovering process
19.	[34]	Initial Analysis process, Implementation process, Analysis process

(2) Step 2. Gathering the domain processes: Investigation processes of the identified models will be gathered at this step based on the following criteria adapted from [41,42]:

The Investigation processes should be gathered from the main model text or diagram; The Investigation process should have a meaning or definition to recognize the semantic meaning of the investigation process.

A total of 64 investigation processes were extracted from 19 DBFI models shown in Table 1. Most of these 64 Investigation processes are redundant and need to be combined in order to produce unified forensics for the DBFI domain. The next step discusses the combining process.

(3) Step 3. Combining and proposing common processes: The mapping process [43] and harmonization process [44] were used in this study to propose common investigation processes. Thus, five main stages were proposed: Initialization stage, Acquiring stage, Investigation stage, Restoring and Recovering stage, and Evaluation stage. Each stage has several investigation processes. For example, the Initialization stage can cover the whole Investigation preparation for any investigation task. Table 2 displays the proposed stages and their processes.

Table 2. Proposed investigation stages for DBFIs.

Existing Processes	Proposed Stages	Proposed Process for Each Stage
Detection process	Initialization Stage	Preparation process
Authentication, System Explanation		
Preparation process		
Preparation process, Incident Confirmation		
Preparation of Database Environment		
Acquirement process		
Server Discovery process		
Preparation process		Detection process
Preparation process		
Crime Reporting process		
Preparation process		
Preparation process		
Investigation process		
Initial Analysis process		
Collection process	Acquiring Stage	Acquisition process
Evidence Collection		
Collection process		
Collection process		
Extraction process		
Investigation process		
Extraction process		
Collection process		
Collection process		
Collection process		
Physical Examination process, Digital Examination process		Preservation process
Defining Gaining process, Artefact Collection process, Volatile Collection process, Nonvolatile Collection process, Preservation process		
Collection process		
Collection process		
Collection process, Preservation process		
Gathering process		
Implementation process		Examination process

Table 2. *Cont.*

Existing Processes	Proposed Stages	Proposed Process for Each Stage
Timeline Construction, Analysis process	Investigation Stage	Reconstruction process
Analysis process		
Investigation process		
Financial Analysis		Analysis process
Investigation process		
Analysis process		
Analysis process, Reporting process		
Analysis process		
Analysis process		Reporting process
Analysis process		
Examination process		
The investigation process and Rebuilding process		
Rebuilding process		
Analysis process		
Restoring and recovering	Restoring and Recovering Stage	Restoring process
Recovering process, and searching		
Restoration process		Recovering process
Recovering process		
Postinvestigation process	Evaluation Stage	Training Staff
		Evaluation of Existing Investigation process

Nominating common processes from extracted processes is based on the similarities in meaning or functioning regardless of naming [45]. Therefore, to nominate common processes that vary in naming, synonyms, definitions, and meaning is laborious and may lead to incorrect results. For this purpose, this study used three techniques to assist in filtering and proposing the common processes from the extracted processes. The techniques are:

- ✓ Synonyms check using the Wordnet2 technique;
- ✓ Synonyms check using the Thesaurus.com (accessed on 11 March 2022) technique;
- ✓ Extraction of semantic functioning or meaning of each concept.

The first and second techniques that were used in the selection process to nominate common processes from extracted processes are synonyms check using Wordnet2 and Thesaurus.com (accessed on 11 March 2022) techniques. WordNet2 is the richest lexical database of English words that are gathered into several cognitive synonym sets, and the Thesaurus.com (accessed on 11 March 2022) technique is a large and widely used free online thesaurus [46]. WordNet2 technique is a lexical–semantic resource that groups together words based on their meanings or functioning [47]. For example, Figures 3 and 4 illustrate how the WordNet2 and Thesaurus.com (accessed on 11 March 2022) techniques are used to group the candidates of common processes.

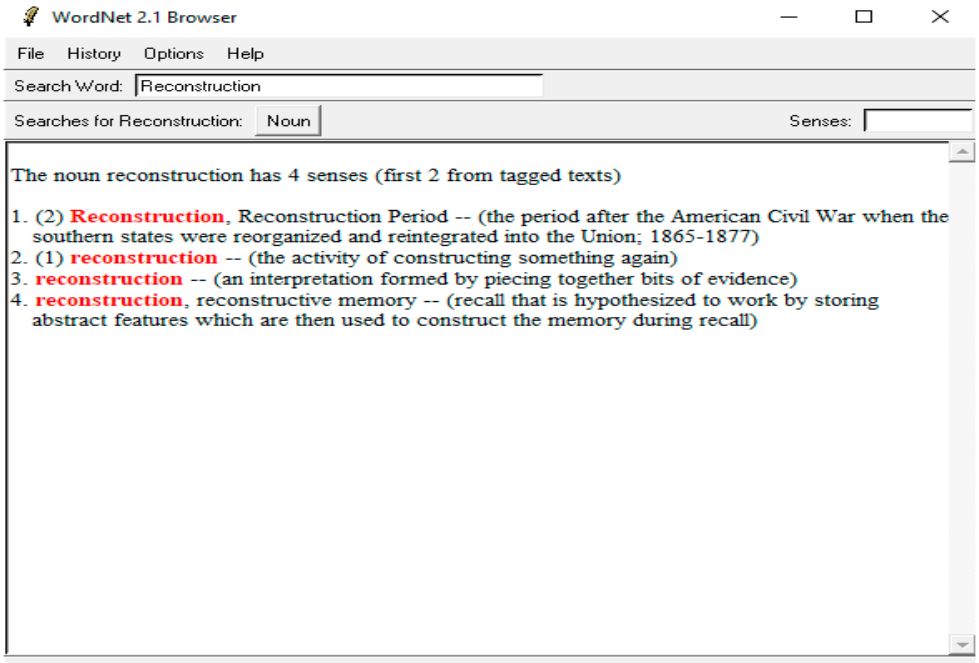

Figure 3. The Synonyms of Reconstruction process using the WordNet2 technique.

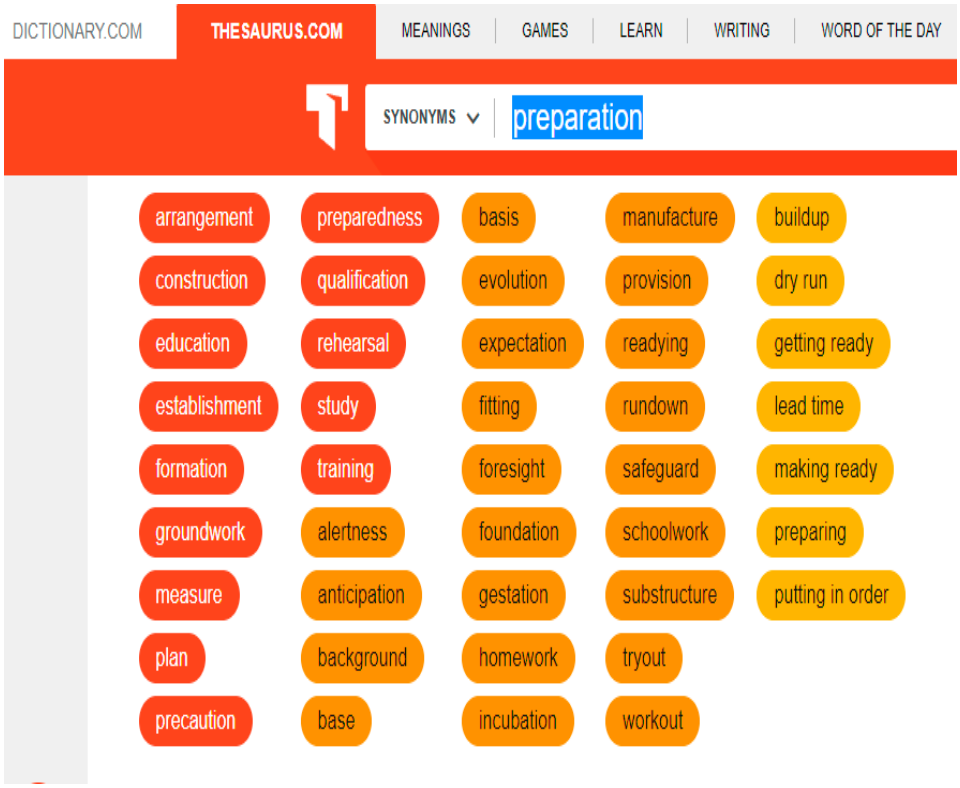

Figure 4. The Synonyms of preparation using the Thesaurus.com (accessed on 11 March 2022) technique.

Therefore, the common processes which have similar meanings or functioning, regardless of their names or synonyms, are grouped into the same stage, as illustrated in Table 2. The processes which have initialization or preparation meaning are combined under the Initialization Stage. For example, the Preparation process has the same meaning along with different synonyms such as Initial Analysis process, Preparation of Database

Environment, Incident Confirmation, Crime Reporting process, Acquirement process, and Investigation process. Similarly, the detection process has the same meaning along with different synonyms such as Authentication, System Explanation, Incident Confirmation, and Server Discovery process. Thus, the redundant processes which have similar meanings of initialization or preparation have been combined under one abstract stage called Initialization Stage.

A similar process is applied for other proposed stages. The proposed Acquiring Stage, Investigation Stage, Restoring and Recovering Stage, and the Evaluation Stage are combined under redundant Investigation processes which have similar meanings or functioning regardless of their names or synonyms. Based on the techniques described above, 64 common processes were categorized into five stages. Each stage has similar processes, either in semantic meaning or functional meaning.

(4) Step 4. Identifying the relationships: Based on discovering the relationships among the proposed investigation stages and processes in the literature, the UML relationships were used to draw the proposed model. Association and aggregation (composition) relationships were used to draw the proposed model. For example, the first stage, i.e., the Initialization stage, was linked to the second stage using association relationships, where the aggregation (composition) relationship between the Initialization stage and their investigation was used. Figure 5 shows the proposed unified investigation process model for the DBFI field.

- Initialization stage: This stage involves two processes: preparation and detection (see Figure 6). The Preparation process aims to prepare an investigation team and trust forensic tools, policies, and procedures for the investigation phase. The investigation team must comply with agency/company policies and procedures in doing the investigation, and it must follow the laws and regulations. Then, the team detects and verifies the database incident using specific forensic tools. The main resources for investigations are the OS log files, application logs, database logs, trace files, and alert files. When the database incident is detected, the investigation team moves to acquire stage to gather the data.
- Acquiring Stage: The main purpose of this stage is to gather, preserve, and examine the data to identify and reveal the database incidents. It consists of three main processes, as shown in Figure 7, which are: the Acquisition process, Preservation process, and Examination process. The Acquisition process is used to gather/acquire volatile and nonvolatile data from different resources. The acquired data need to be protected in terms of their integrity. The Preservation process is used to protect the acquired data using proper techniques, e.g., hashing algorithms. The investigation team should take a backup of the original data and hashed data in case tampering happens. The Examination process is used to check the authentication of the acquired data. Thus, the investigation team needs to rehash the gathered data and check the consistency of the data; in case of no matching, the investigation team should go back and take another copy of the original data. The investigation stage is required if the authenticity of the data is correct and exposed to no tampering.
- Investigation Stage: This is the main stage, which rebuilds and analyzes the timeline events, interprets, reports, documents, and presents the database incident. It consists of three main processes, as shown in Figure 8: Reconstruction process, Analysis process, and Reporting process. The timeline of the acquired data will be rebuilt to analyze and interpret to find similar patterns of the crime. Then, the chain of custody of the evidence is gathered and structured in robust documents. Finlay, the report should be prepared and submitted to the court. Investigators need to present the result in court and reply to all judges' questions. This is the final stage of a real investigation; however, one of the main points, which is often neglected by investigators, is restoring and recovering data for business continuity. Therefore, the next stage is considered in the proposed model to fill this gap.

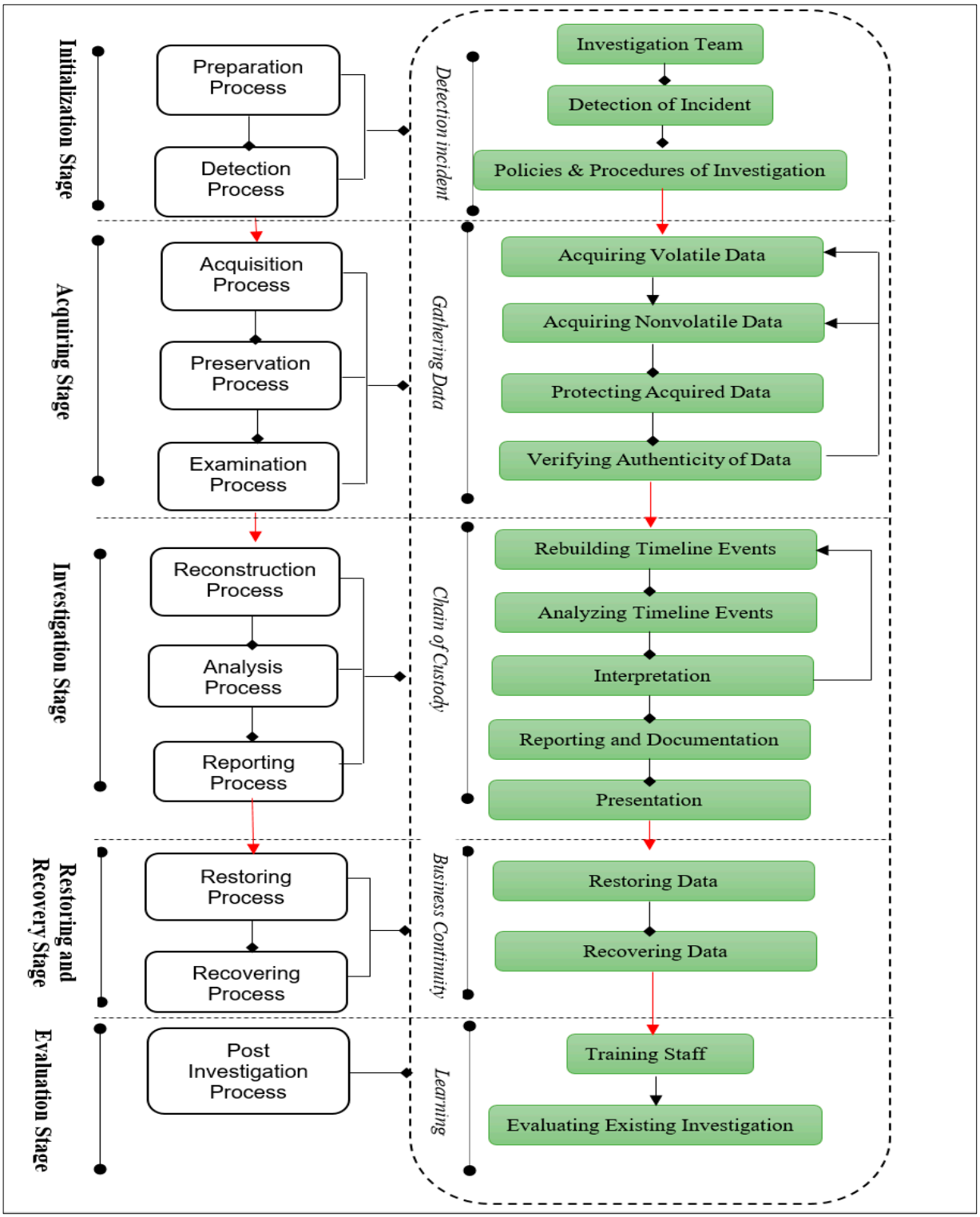

Figure 5. Unified Forensics Model for Database Forensics Field.

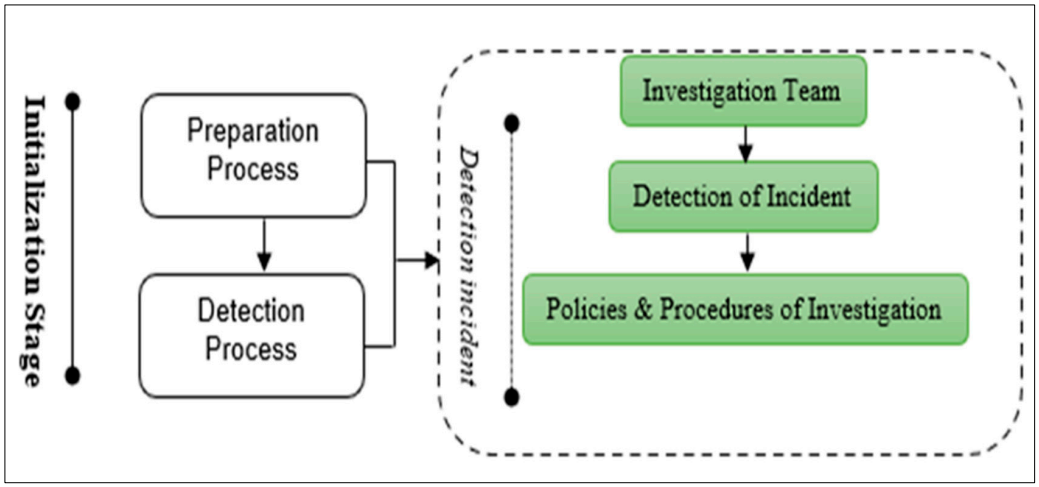

Figure 6. Initialization stage.

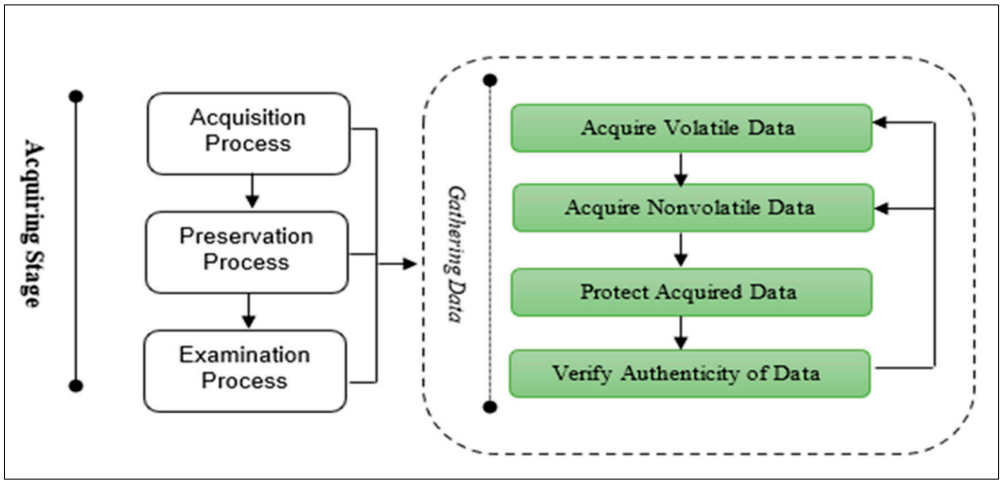

Figure 7. Acquiring stage.

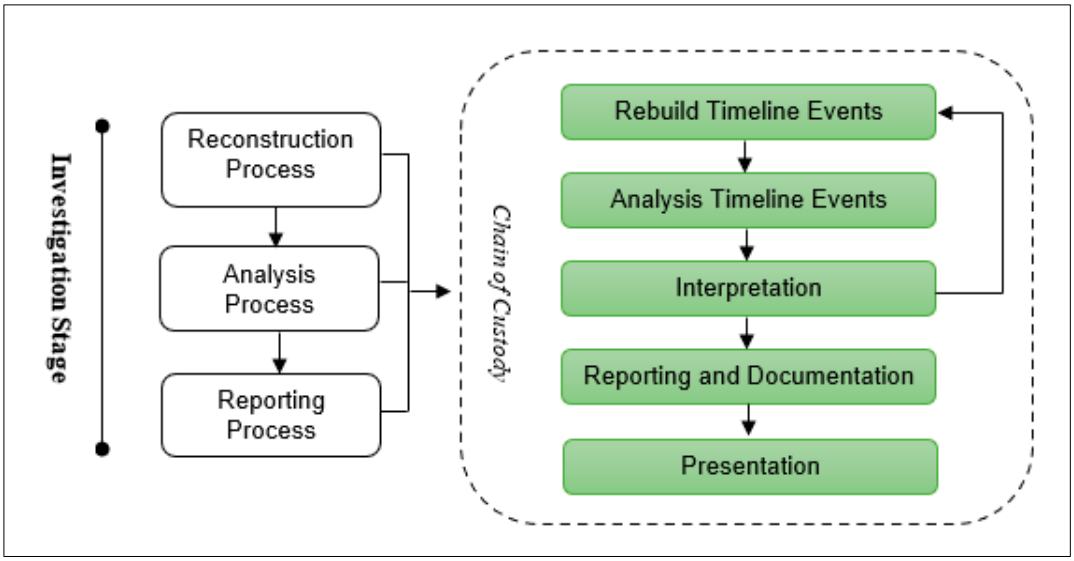

Figure 8. Investigation Stage.

- Restoring and Recovering Stage: This stage aims to restore and recover the deleted/damaged data to the new database environment. This stage consists of two main

processes, as shown in Figure 9: restoring and recovering data. Data recovery is the process of restoring data that has been lost, corrupted, or made inaccessible for any reason or accidentally deleted.

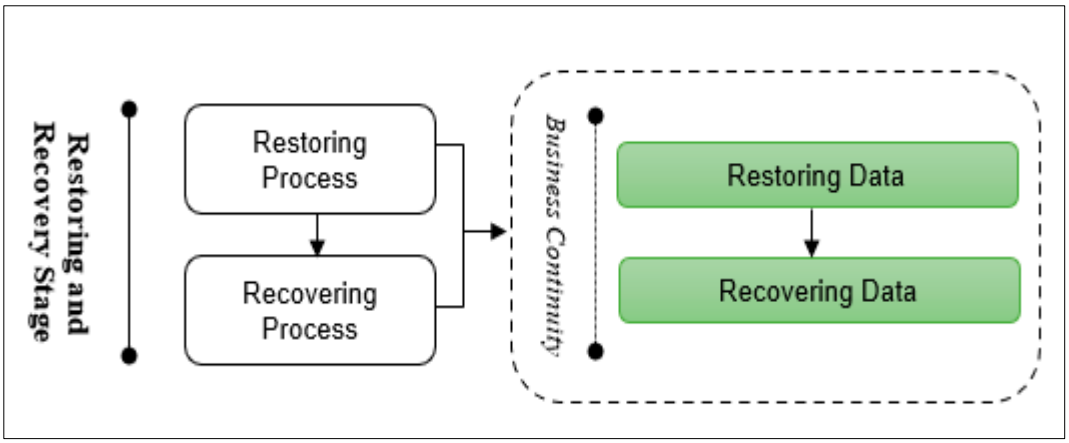

Figure 9. Restoring and recovering stage.

- Evaluation Stage: This stage needs to evaluate the investigation process to improve it and avoid any problems. This stage consists of the postinvestigation process, as shown in Figure 10. It is used to teach the staff the principles of the investigation and how they can deal with or face any database incidents. In addition, it is used to improve the whole investigation stage.

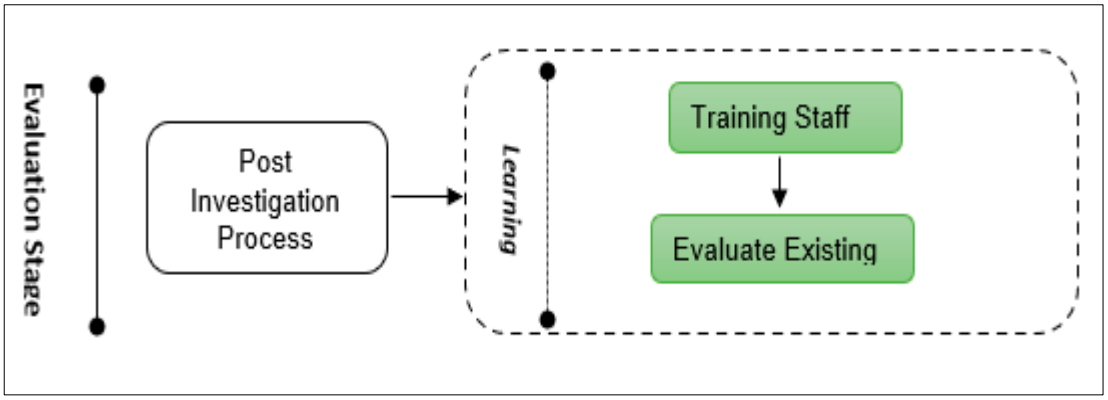

Figure 10. Evaluation Stage.

(5) Step 5. Validation and Implementation Stage: Validation and Implementation are two significant processes used to assess the applicability of the proposed model to real investigations with real scenarios or real case studies. Thus, the proposed model was evaluated from two perspectives (as mentioned earlier): The Completeness perspective and the Implementation perspective, which are explained as follows:

- Completeness perspective: In this perspective, the focus was on the validation in case the proposed UFM was completed against the available DBFIs models. To finalize this process, a comparison of the UFM was made against other models [1]. The comparison was made to verify if the proposed UFM is effective and whether it can entirely translate and fit into the existing domain models. Table 3 shows the models used in the comparison. The proposed UFM is more comprehensive, and it incorporates activities that have been identified in the previous models. Table 3 shows that all the processes of the previously proposed models are covered in the proposed UFM.

Table 3. Comparison between exiting DBFI models with the proposed model.

	Existing Database Forensics Investigation Models																		
	M 1	**M 2**	**M 3**	**M 4**	**M 5**	**M 6**	**M 7**	**M 8**	**M 9**	**M 10**	**M 11**	**M 12**	**M 13**	**M 14**	**M 15**	**M 16**	**M 17**	**M 18**	**M 19**
Initialization stage	√	√	√	×	√	√	×	√	√	√	√	×	√	√	×	×	×	×	√
Acquiring stage	√	√	√	×	√	√	√	√	√	√	√	√	√	√	√	√	×	√	×
Investigation stage	√	√	×	√	×	√	√	√	×	√	√	√	√	×	√	√	√	√	√
Restoring and Recovering Stage	√	√	×	×	×	×	×	√	×	×	×	×	×	×	×	×	×	√	×
Evaluation Stage	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×

- Implementation perspective (experiments environment): The applicability of the proposed model to the real investigation of the database incidents is evaluated in this stage. To this end, FTK Imager and HashMyFiles forensic tools are used. The FTK Imager tool is used to capture the data, and the HashMyFiles tool is used to preserve the captured data. To do this, the authors used the following scenario: "We received a complaint from a customer called Arafat. He said that his credit card is not working, and he cannot login to his account". For this scenario, the authors will use the environment of the following experiment, which is illustrated in Figure 11. The first three stages of the proposed UFM are used: the Initialization, Acquiring, and Investigation stages to capture and reveal malicious activities:

✓ Initialization stage: The investigation team should review the policies and procedures of the organizations and the database investigation procedures before starting the investigation. They should prepare the trusted investigation tools. In this case, we prepared FTK Imager and HashMyFiles tools. The investigation team interviewed with DBA and the IT staff to gather the information needed to verify the database incident. The information (database files locations, passwords, accounts, users, IP, etc.) should be gathered through interviews. The version of the database is Oracle 11.2.0. The DBA informed us that he discovered that the account number of the customer was locked, and his secret key was changed. Then, we discovered that the database had been compromised. In this case, the volatile information is very important to detect the attack and find the path of the attack. For this purpose, the investigation team should move to the acquiring stage.

✓ Acquiring stage: In the second stage, the investigation team uses FTK Imager and HashMyFiles tools to capture and preserve volatile data. The investigation team should capture volatile and non-volatile data (in order), as shown in Figure 12. The captured data should be moved to an external flash disk to avoid any problem and then duplicated. The captured data should be protected from any tampering/updating. Thus, the HashMyFiles tool should be used for this purpose. The HashMyFiles tool is used to produce hashed values for the captured data, as shown in Figure 13. To check the consistency of the data before moving to the analysis procedure, the authentication of the captured data needs to be performed using the FTK Imager tool. Thus, the file "memdump", which is already hashed, is verified,

as shown in Figure 14. The authentication is conducted, the value is correct, has no tampering, and it is ready to move to the next stage of the investigation.

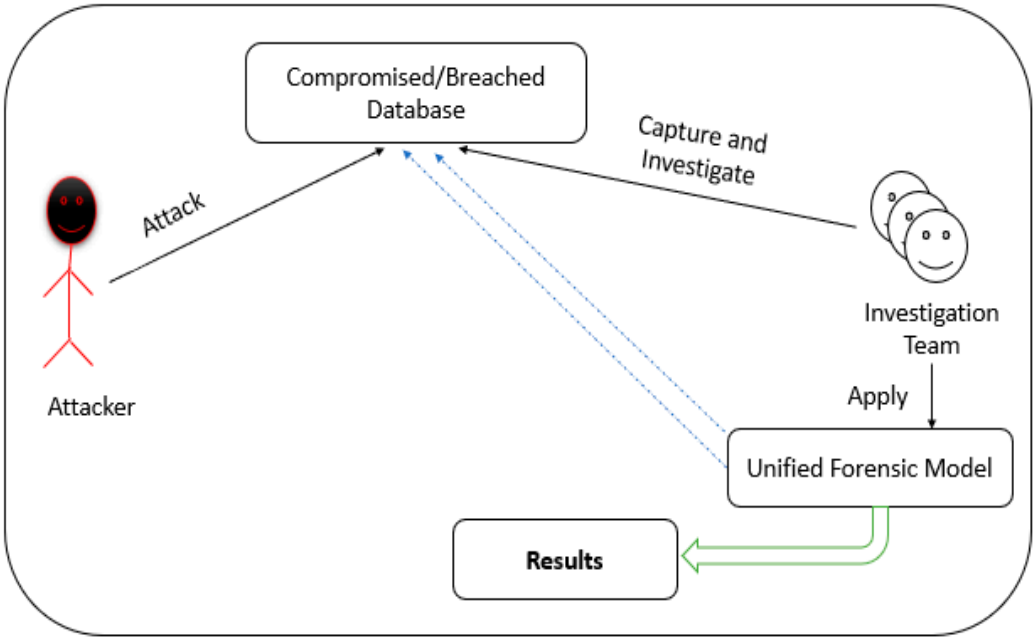

Figure 11. Experiments environment.

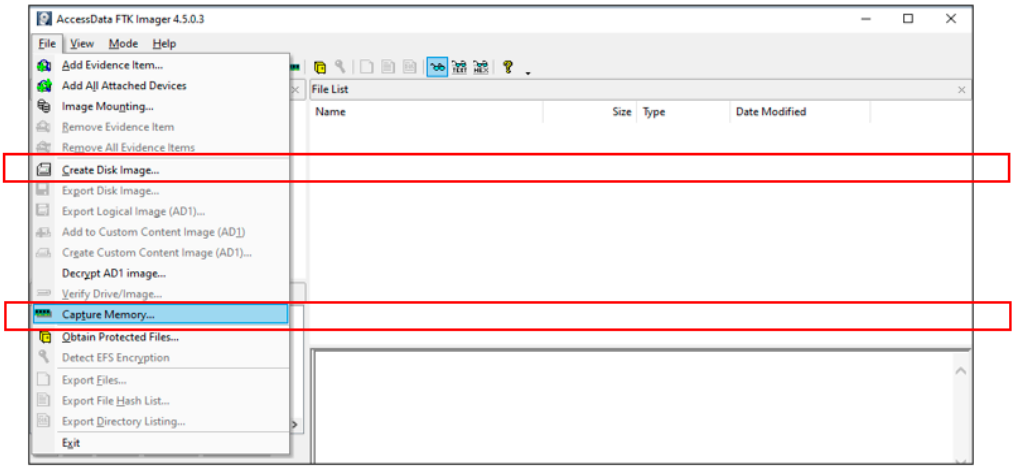

Figure 12. Capturing volatile and nonvolatile data.

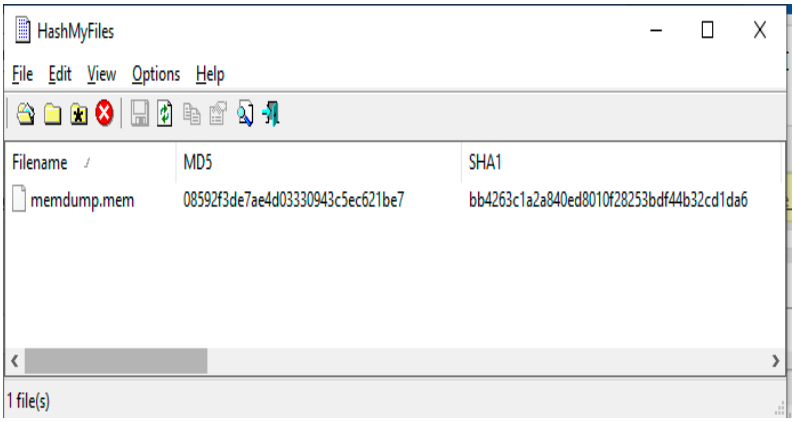

Figure 13. Hashing the captured data.

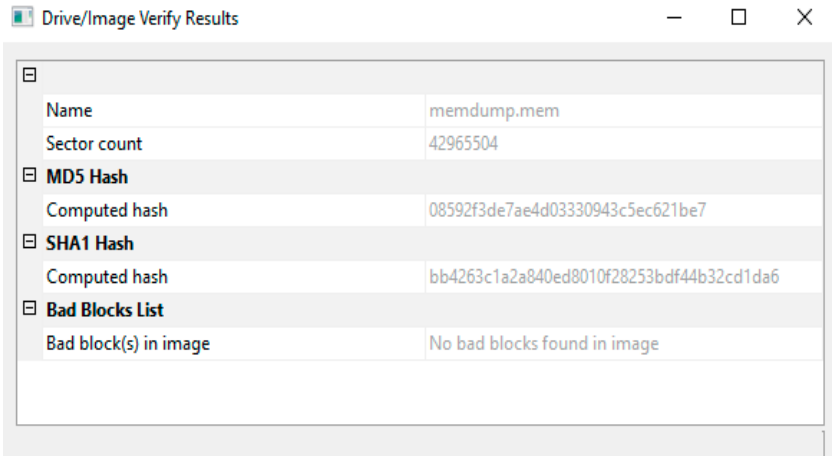

Figure 14. Results of verification.

✓ Investigation Stage: In this stage, the captured information is analyzed, and malicious activities are identified. The FTK Imager tool is used to search for malicious transactions. We started digging in the captured image and used some keywords based on the previous DBA's information and some log files such as "Credit Card", "CreditCard", "Update", "Delete", and "Secret Key". After trying all these keywords, it was found that the attacker had changed the victim's secret key and changed it to 222, as shown in Figure 15.

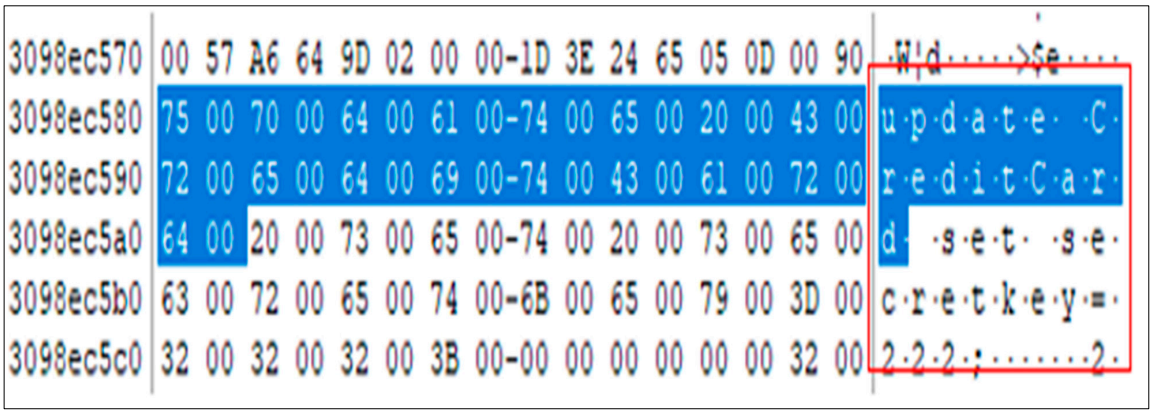

Figure 15. Analyzing acquired data.

4. Results and Discussion

This study proposed a new unified forensics model, UFM, applicable to the database forensics field. It unified all models and frameworks existing in the database forensics field. UFM consists of five stages and 12 investigation processes, as shown in Figure 5. Thus, the redundant/overlapping investigation processes were harmonized and combined in one abstract model. For example, the initialization stage covered all existing processes with similar meanings, such as Detection process Authentication, System Explanation Preparation process, Preparation process, Incident Confirmation, Preparation of Database Environment, Acquirement process, Server Discovery process, Crime Reporting process, Defining Investigation process, and Initial Analysis process. Thus, this stage allows investigation teams to prepare policies, procedures, trusted forensics tools, and trusted forensics environments, conduct interviews, detect database crime, and conduct search warrants, authentications, and authorizations.

Acquiring stage covered all similar exiting investigation processes with similar meanings, for example, Collection process, Collection Evidence process, Extraction Process,

Investigation process, Extraction process, Physical Examination process, Digital Examination process, Defining Gaining process, Artefact collection process, Volatile collection process, Non-volatile collection process, Preservation process, Gathering process, and implementation. This stage allows investigation teams to capture, preserve, and examine volatile and non-volatile data.

The existing analysis and reconstruction processes were harmonized and gathered under one abstract process, called the "Investigation stage"; they include Timeline Construction, Analysis process, Investigation process, Financial Analysis, Investigation process, Reporting process, Examination process, Investigation Process, the Rebuilding process, Restoring and recovering, and Recovering process.

The Recovering and Restoring stages were proposed in UFM, which gives the proposed model the ability to restore and recover deleted files and save time for business continuity. The final stage, i.e., the Evaluation stage, is dedicated to learning and improvement. It consists of post-investigation processes.

The validation process results showed that the proposed UFM is a comprehensive model and can work with different scenarios in the database systems. UFM covered 64 investigation processes. In addition, the implementation results showed that UFM could be implemented effectively even by investigators who are new to the field.

5. Advantages of the Proposed Unified Forensic Model

In this study, a novel model called Unified Forensic Model was developed to solve the heterogeneity, interoperability, and complexity of the DBFI domain. The interoperability of the DBFI domain was solved by developing the UFM. The general processes and concepts used in the DBFI domain are identified and combined as required. This process involves analysing the DBFI domain frameworks, models, and processes in the DBFI domain. The study generalizes the design science approach to creating the UFM. Two validation techniques have been utilised to ensure that the UFM can be interoperable in many database systems: a comparison against other models and a case study. The comparison against other model techniques has been used to ensure that the developed UFM can cover all DBFI domain models. The case study/scenario technique has been used to ensure the applicability of the UFM in the DBFI domain. Thus, the UFM can be interoperable along with any database system.

Furthermore, the heterogeneity of the DBFI domain has been solved by the proposed five common investigation stages. The proposed five common investigation processes covered 64 investigation processes of the DBFI domain.

Therefore, the benefits of the proposed UFM for the domain of forensic practitioners are in:

(1) Simplifying common communication amongst different DBFI domain practitioners through a common representation layer that includes all the processes, concepts, tasks, and activities that must exist in the DBFI domain;
(2) Providing guidelines and new model-developing processes that assist domain practitioners in managing, sharing, and reusing DBFI domain knowledge;
(3) Solving the heterogeneity and ambiguity of the DBFI domain; Generality and reusability of common processes.

6. Conclusions

Researchers attempt to collect, preserve, identify, analyze, reconstruct, and document database incidents in the database forensics field. To this end, different database forensics models and frameworks have been proposed in the literature. In this study, a new unified model, called UFM, was proposed for the database forensics investigation (DBFI) field. The redundancy in the investigation processes, which causes ambiguity and heterogeneity among domain forensic practitioners, was addressed in UFM by combining all the existing models and frameworks of the database forensics field. The exiting processes were grouped and combined in five stages.

Furthermore, two completely new investigation stages, i.e., Restoring and Recovering stage and the Evaluation stage, were incorporated into UFM. The designs science approach was adopted to carry out this study, developing the resulting procedure to establish a unified forensic baseline for the database forensics field. Five stages, i.e., Initialization stage, acquiring stage, Investigation stage, Restoring and Recovering stage, and Evaluation stage were included in the proposed model. This model was compared with other models proposed previously in this field to evaluate the proposed model's completeness. The proposed model allows domain forensic practitioners to identify, capture, preserve, reconstruct, and analyze volatile and non-volatile data from suspicious databases based on trusted forensic tools. Furthermore, it may be used as a guide to enhance the forensic database stages and database security measures. The following recommendations are potential areas for future research in the database forensics field: (1) developing a semantic metamodeling language that manages, organizes, structures, and shares investigation knowledge, (2) developing an investigation repository for the retrieval and storage of field knowledge to help the forensic practitioners gain quick access, and (3) applying the proposed UFM to the real-life scenarios.

Author Contributions: Conceptualization, A.A.-D., W.M.S.Y. and A.-H.M.E.; methodology, A.A.-D., W.M.S.Y., S.B.A.R. and A.-H.M.E.; software, A.A.-D.; validation, A.A.-D., W.M.S.Y., A.-H.M.E., D.S.K. and A.A.A.; formal analysis, A.A.-D., S.B.A.R. and W.M.S.Y.; investigation, A.A.-D., A.-H.M.E., D.S.K. and A.A.A.; resources, A.A.-D. and W.M.S.Y.; data curation, A.A.-D. and A.-H.M.E.; writing—original draft preparation, A.A.-D., W.M.S.Y. and A.-H.M.E.; writing—review and editing, A.A.-D., W.M.S.Y. and A.-H.M.E.; visualization, A.A.-D., W.M.S.Y., S.B.A.R., A.-H.M.E., D.S.K. and A.A.A.; supervision, A.A.-D., W.M.S.Y. and A.-H.M.E.; project administration, A.A.-D.; funding acquisition, D.S.K. and A.A.A. All authors have read and agreed to the published version of the manuscript.

Funding: This project is funded by Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2022R308), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Acknowledgments: Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2022R308), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Al-Dhaqm, A.; Razak, S.; Ikuesan, R.A.; R Kebande, V.; Hajar Othman, S. Face Validation of Database Forensic Investigation Metamodel. *Infrastructures* **2021**, *6*, 13. [CrossRef]
2. Al-Dhaqm, A.; Razak, S.A.; Othman, S.H.; Nagdi, A.; Ali, A. A generic database forensic investigation process model. *J. Teknol.* **2016**, *78*. [CrossRef]
3. Fasan, O.M.; Olivier, M. Reconstruction in database forensics. In Proceedings of the IFIP International Conference on Digital Forensics, Pretoria, South Africa, 3–5 January 2012; pp. 273–287.
4. Beyers, H.Q. Database Forensics: Investigating Compromised Database Management Systems. Ph.D. Thesis, University of Pretoria, Pretoria, South Africa, 2014.
5. Susaimanickam, R. A Workflow to Support Forensic Database Analysis. Ph.D. Thesis, Murdoch University, Murdoch, Australia, 2012.
6. Fasan, O.M.; Olivier, M.S. On Dimensions of Reconstruction in Database Forensics. In Proceedings of the International Workshop on Digital Forensics and Incident Analysis, WDFIA, Crete, Greece, 6–8 June 2012; pp. 97–106.
7. Yoon, J.; Jeong, D.; Kang, C.; Lee, S. Forensic investigation framework for the document store NoSQL DBMS: MongoDB as a case study. *Digit. Investig.* **2016**, *17*, 53–65. [CrossRef]
8. Khanuja, H.K.; Adane, D.S. A framework for database forensic analysis. *Comput. Sci. Eng.* **2012**, *2*, 27–41. [CrossRef]
9. Olivier, M.S. On metadata context in database forensics. *Digit. Investig.* **2009**, *5*, 115–123. [CrossRef]
10. Delfanti, R.L.; Piccioni, D.E.; Handwerker, J.; Bahrami, N.; Krishnan, A.P.; Karunamuni, R.; Hattangadi-Gluth, J.A.; Seibert, T.M.; Srikant, A.; Jones, K.A.; et al. Glioma Groups Based on 1p/19q, IDH, and TERT Promoter Mutations in Tumors. *N. Engl. J. Med.* **2018**, *372*, 2499–2508. [CrossRef]
11. Litchfield, D. *Oracle Forensics Part 1: Dissecting the Redo Logs*; NGSSoftware Insight Security Research (NISR), Next Generation Security Software Ltd.: Sutton, UK, 2007.
12. Litchfield, D. *Oracle Forensics Part 2: Locating Dropped Objects*; NGSSoftware Insight Security Research (NISR) Publication, Next Generation Security Software: Sutton, UK, 2007.

13. Litchfield, D. *Oracle Forensics Part 3: Isolating Evidence of Attacks against the Authentication Mechanism*; NGSSoftware Insight Security Research (NISR) Publication: Sutton, UK, March 2007.
14. Litchfield, D. *Oracle Forensics Part 4: Live Response*; NGSSoftware Insight Security Research (NISR), Next Generation Security Software Ltd.: Sutton, UK, 2007.
15. Fowler, K. *SQL Server Forenisc Analysis*; Pearson Education: Indianapolis, IN, USA, 2008.
16. Son, N.; Lee, K.; Jeon, S.; Chung, H.; Lee, S.; Lee, C. The method of database server detection and investigation in the enterprise environment. In Proceedings of the FTRA International Conference on Secure and Trust Computing, Data Management, and Application, Loutraki, Greece, 28–30 June 2011; pp. 164–171.
17. Frühwirt, P.; Huber, M.; Mulazzani, M.; Weippl, E.R. InnoDB database forensics. In Proceedings of the 2010 24th IEEE International Conference on Advanced Information Networking and Applications, Perth, WA, Australia, 20–23 April 2010; Volume 386, pp. 1028–1036. [CrossRef]
18. Frühwirt, P.; Kieseberg, P.; Schrittwieser, S.; Huber, M.; Weippl, E. Innodb database forensics: Reconstructing data manipulation queries from redo logs. In Proceedings of the 2012 Seventh International Conference on Availability, Reliability and Security, Prague, Czech Republic, 20–24 August 2012; pp. 625–633.
19. Frühwirt, P.; Kieseberg, P.; Schrittwieser, S.; Huber, M.; Weippl, E. InnoDB database forensics: Enhanced reconstruction of data manipulation queries from redo logs. *Inf. Secur. Tech. Rep.* **2013**, *17*, 227–238. [CrossRef]
20. Lee, G.T.; Lee, S.; Tsomko, E.; Lee, S. Discovering methodology and scenario to detect covert database system. In Proceedings of the Future Generation Communication and Networking (FGCN 2007), Jeju Island, Korea, 6–8 December 2007; Volume 2, pp. 130–135.
21. Azemović, J.; Mušić, D. Efficient model for detection data and data scheme tempering with purpose of valid forensic analysis. In Proceedings of the 2009 International Conference on Computer Engineering and Applications (ICCEA 2009), Manila, Philippines, 6–8 June 2009.
22. Snodgrass, R.; Yao, S.; Collberg, C. Tamper Detection in Audit Logs. In Proceedings of the Thirtieth International Conference on Very Large Data Bases, Toronto, ON, Canada, 31 August–3 September 2004; pp. 504–515. [CrossRef]
23. Khanuja, H.; Suratkar, S.S. Role of metadata in forensic analysis of database attacks. In Proceedings of the 2014 IEEE International Advance Computing Conference (IACC), New Delhi, India, 21–22 February 2014; pp. 457–462.
24. Frühwirt, P.; Kieseberg, P.; Krombholz, K.; Weippl, E. Towards a forensic-aware database solution: Using a secured database replication protocol and transaction management for digital investigations. *Digit. Investig.* **2014**, *11*, 336–348. [CrossRef]
25. Chopade, M.S.D.; Bere, S.S.; Kasar, M.N.B.; Moholkar, M.A. V SQL Query Recommendation Using Collaborative Query Log: A Survey. *Int. J. Recent Innov. Trends Comput. Commun.* **2004**, *2*, 3715–3721.
26. Wright, P.M. Oracle database forensics using LogMiner. In *June 2004 Conference*; SANS Institute: London, UK, 2005; pp. 1–39.
27. Yoon, J.; Lee, S. A method and tool to recover data deleted from a MongoDB. *Digit. Investig.* **2018**, *24*, 106–120. [CrossRef]
28. Al-Dhaqm, A.; Abd Razak, S.; Othman, S.H.; Ali, A.; Ghaleb, F.A.; Rosman, A.S.; Marni, N. Database forensic investigation process models: A review. *IEEE Access* **2020**, *8*, 48477–48490. [CrossRef]
29. Hauger, W.K.; Olivier, M.S. The state of database forensic research. In Proceedings of the 2015 Information Security for South Africa (ISSA), Johannesburg, South Africa, 12–13 August 2015; pp. 1–8.
30. Bria, R.; Retnowardhani, A.; Utama, D.N. Five stages of database forensic analysis: A systematic literature review. In Proceedings of the 2018 International Conference on Information Management and Technology (ICIMTech), Jakarta, Indonesia, 3–5 September 2018; pp. 246–250.
31. Wong, D.; Edwards, K. System and Method for Investigating a Data Operation Performed on a Database. U.S. Patent 10/879,466, 29 December 2005.
32. Al-Dhaqm, A.; Razak, S.; Siddique, K.; Ikuesan, R.A.; Kebande, V.R. Towards the Development of an Integrated Incident Response Model for Database Forensic Investigation Field. *IEEE Access* **2020**, *8*, 45018–145032. [CrossRef]
33. Kelly, S.; Pohjonen, R. Worst practices for domain-specific modeling. *IEEE Softw.* **2009**, *26*, 22–29. [CrossRef]
34. Ogutu, J.O. A Methodology To Test The Richness of Forensic Evidence of Database Storage Engine: Analysis of MySQL Update Operation in InnoDB and MyISAM Storage Engines. Ph.D. Thesis, University of Nairobi, Nairobi, Kenya, 2016.
35. Khanuja, H.K.; Adane, D. Forensic analysis of databases by combining multiple evidences. *Int. J. Comput. Technol.* **2013**, *7*, 654–663. [CrossRef]
36. Adedayo, O.M.; Olivier, M.S. Ideal log setting for database forensics reconstruction. *Digit. Investig.* **2015**, *12*, 27–40. [CrossRef]
37. Lee, D.; Choi, J.; Lee, S. Database forensic investigation based on table relationship analysis techniques. In Proceedings of the 2009 2nd International Conference on Computer Science and Its Applications, CSA 2009, Jeju Island, Korea, 10–12 December 2009; p. 5404235.
38. Choi, J.; Choi, K.; Lee, S. Evidence investigation methodologies for detecting financial fraud based on forensic accounting. In Proceedings of the 2009 2nd International Conference on Computer Science and Its Applications, CSA 2009, Jeju Island, Korea, 10–12 December 2009; p. 5404202.
39. Tripathi, S.; Meshram, B.B. Digital evidence for database tamper detection. *J. Inf. Secur.* **2012**, *3*, 113–121. [CrossRef]
40. Wagner, J.; Rasin, A.; Grier, J. Database forensic analysis through internal structure carving. *Digit. Investig.* **2015**, *14*, S106–S115. [CrossRef]

41. Caro, M.F.; Josyula, D.P.; Cox, M.T.; Jiménez, J.A. Design and validation of a metamodel for metacognition support in artificial intelligent systems. *Biol. Inspired Cogn. Archit.* **2014**, *9*, 82–104. [CrossRef]
42. Bogen, A.C.; Dampier, D.A. Preparing for Large-Scale Investigations with Case Domain Modeling. In Proceedings of the Digital Forensics Research Conference, DFRWS, New Orleans, LA, USA, 17–19 August 2005.
43. Selamat, S.R.; Yusof, R.; Sahib, S. Mapping process of digital forensic investigation framework. *Int. J. Comput. Sci. Netw. Secur.* **2008**, *8*, 163–169.
44. Al-Dhaqm, A.; Razak, S.; Othman, S.H.; Choo, K.-K.R.; Glisson, W.B.; Ali, A.; Abrar, M. CDBFIP: Common Database Forensic Investigation Processes for Internet of Things. *IEEE Access* **2017**, *5*, 24401–24416. [CrossRef]
45. Ali, A.; Abd Razak, S.; Othman, S.H.; Mohammed, A.; Saeed, F. A metamodel for mobile forensics investigation domain. *PLoS ONE* **2017**, *12*, e0176223. [CrossRef] [PubMed]
46. Haghighi, P.D.; Burstein, F.; Li, H.; Wang, C. Integrating social media with ontologies for real-time crowd monitoring and decision support in mass gatherings. In Proceedings of the Pacific Asia Conference on Information Systems, Jeju Island, Korea, 18–22 June 2013.
47. Akinyemi, J.A.; Clarke, C.L.A.; Kolla, M. Towards a collection-based results diversification. In Proceedings of the 9th international conference on Adaptivity, Personalization and Fusion of Heterogeneous Information, Paris, France, 28–30 April 2010; pp. 202–205.