Computers and Electronics in Agriculture 75 (2011) 327–336

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Original paper

Data modeling to facilitate internal traceability at a grain elevator

Maitri Thakur[a,b,*], Bobby J. Martens[c], Charles R. Hurburgh[a,d]

[a] Department of Agricultural and Biosystems Engineering, Iowa State University, Ames, IA 50011, USA
[b] Department of Industrial and Manufacturing Systems Engineering, Iowa State University, Ames, IA 50011, USA
[c] Department of Logistics, Operations, and Management Information Science, Iowa State University, Ames, IA 50011, USA
[d] Department of Food Science and Human Nutrition, Iowa State University, Ames, IA 50011, USA

ARTICLE INFO

Article history:
Received 20 July 2010
Received in revised form 7 December 2010
Accepted 19 December 2010

Keywords:
Internal traceability
Bulk grain handling
Elevator
Data modeling
ER model

ABSTRACT

Data management in food supply chains to facilitate product traceability has gained importance in recent years. This paper presents a relational database model to facilitate internal traceability at a grain elevator, which is one of the first nodes in a food supply chain. This approach for modeling traceability information in bulk food supply chains has not been studied in past. At an elevator, grain lots (inbound deliveries) are blended to meet buyer specifications, and individual lot identity is not maintained. As a result, an outbound shipment to a customer likely contains grain from many different sources. In a food safety related emergency, tracing the source of a problem or tracking other affected shipments would be nearly impossible. An efficient internal data management system could mitigate these problems by recording all grain lot transformations/activities, including movement, aggregation, segregation, and destruction as well as supplier and customer information. In this paper, a relational database management system is proposed that stores all necessary information, including product and quality information, related to the grain lots in order to enable product traceability. The system can be queried to retrieve information related to incoming, internal and outgoing lots and to retrieve information that connects the individual incoming grain lots to an outgoing shipment. Furthermore, this system can be used both to trace back to the source of a given lot and to track information about previously shipped lots forward. In addition to traceability application, the information stored in this database provides a comprehensive dataset for many applications including mass flow optimization, resource optimization and improved operational efficiency of the grain elevator.

1. Introduction

Tracking and tracing food products throughout the supply chains has gained considerable importance over the last few years (Carriquiry and Babcock, 2007; Jansen-Vullers et al., 2003; Madec et al., 2001; McKean, 2001; Thakur and Hurburgh, 2009). Consumers all over the world have experienced various food safety and health issues. In addition, consumer demand for high quality food and feed products, non-GMO (genetically modified organisms) foods and other specialty products such as organic food has grown in recent years. These factors have led to a growing interest in developing systems for food supply chain traceability, and, as a result, a number of food safety and traceability laws exist in many countries.

The European Union law describes "Traceability" as an ability to track any food, feed, food-producing animal or substance that will be used for consumption, through all stages of production, processing and distribution (Official Journal of the European Union, 2002). Besides ensuring a safe food supply, the USDA (United States Department of Agriculture) Economic Research Service states that use of a traceability system results in lower cost distribution systems, reduced recall expenses, and expanded sales of products with attributes that are difficult to discern (Golan et al., 2004). Thus, in several countries food traceability has become important for reasons other than just the legal obligations. In addition, traceability is important for many other reasons, such as responding to food security threats, documenting chain of custody, documenting production practices, meeting regulatory compliance, and even analyzing logistics and production costs.

Three examples demonstrate how traceability standards are being developed and implemented. The ISO 22005 Food Safety Standard requires that each company know their immediate suppliers and customers based on the principle of one up and one down (International Organization for Standardization, 2007). Food safety is therefore the joint responsibility of all the actors involved. In

* Corresponding author. Present address: Aquaculture Technology, SINTEF Fisheries and Aquaculture, Brattørkaia 17C, 7010 Trondheim, Norway.
Tel.: +47 457 92 572; fax: +47 932 70 701.
E-mail addresses: maitri.thakur@sintef.no, maitri@iastate.edu, maitrithakur@gmail.com (M. Thakur).

0168-1699/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.compag.2010.12.010

absence of internal and chain traceability systems, it is impossible to track the origin of food and to know the production and processing steps it underwent. Since, food safety cannot be ensured without proper documentation of production practices, absence of such systems can have food safety implications. Next, the Bioterrorism Preparedness and Response Act of 2002 (the Bioterrorism Act) requires all food and feed companies to self-register with the Food and Drug Administration and maintain records and information for food traceability purposes (US Food and Drug Administration, 2002). Finally, the GS1 Traceability Standard states that traceability across the supply chain involves the association of flow of information with the physical flow of traceable items. It also states that in order to achieve traceability across the supply chain, all traceability partners must achieve internal and external traceability (GS1 Global Traceability Standard, 2007). Therefore, all the actors involved in the food supply chain are required to store necessary information related to the food product that link inputs with outputs, so that when demanded, the information can be provided to the food inspection authorities on a timely basis as specified by the regulation.

Previous research has emphasized the importance of internal traceability systems. Moe (1998) states that many advantages can accrue from having an internal traceability system from being able to trace the raw material that went into a final product to possibility of improved process control, correlating product data with raw material characteristics and processing data as well as optimization of the use of raw materials for each product type. In order to achieve a fully traceable supply chain, it is important to develop systems for both external supply chain traceability as well as internal traceability. This includes linking, to the best extent possible, units of output with specific units of input. First, each actor must have the ability to externally trace back and track forward product information using the one-up and one-down basis. Then, in order to determine the cause of the problem or to efficiently recall the associated (or contaminated) food products, each supply chain actor should have an internal record-keeping system enabling them to trace back to the input ingredients and track forward to the output products. Therefore, each actor in the supply chain must not only know their immediate suppliers and customers but also maintain accurate records of their internal processes.

Still, traceability in the food industry is lacking because information is often lost or information access is time consuming because of a lack of standardized communication between different systems. This is especially a concern when evaluating supply chains related to bulk grain. In this paper, we present a traceability system for a bulk grain handling scenario. Because of the complexities associated with receiving, storing, and blending bulk grains, a bulk grain handling scenario serves as a good example of how a traceability system can be developed for complex product flows. In this paper, we first describe the functions of a grain elevator, including the complications related to implementing a bulk grain traceability system. Next, traceability literature is highlighted and data management systems are reviewed. Finally, our methodology is discussed and the results of our relational database model, which can be used to facilitate internal traceability at a grain elevator, are offered.

1.1. Bulk grain handling

Various lot-activities (transformations) take place as grain moves through the supply chain from the farm to the consumer. These transformations include aggregation, segregation, storage, transfer and destruction (Thakur and Hurburgh, 2009). It is important to be aware of the type and location of each transformation as it is necessary to be able to track and trace the food product through a firm or processing facility (Donnelly et al., 2009; Schwägele, 2005). Grain elevators, which handle bulk commodities like corn and soybeans, are important nodes in the bulk grain supply chan. The elevators buy grain from farmers and store the grain in storage bins (i.e., grain bins or silos) before selling it to the customers. Fig. 1 shows a typical bulk grain handling scenario.

The incoming grain lots from farmers are assigned a unique scale-ticket number, weighed and graded based on quality parameters. These quality parameters include moisture, test weight, damaged material and foreign material. A quality grade is determined based on these parameters and the lot is assigned and transferred to one or more storage bins based on space and quality constraints. Grain is kept in storage bins until it is shipped to a customer. However, while in storage, all or part of the contents of a bin can be transferred to other bins in order to avoid spoilage due to environmental conditions (usually related to increasing temperature inside a bin). This internal movement often goes unrecorded and complicates the lot dynamics due to mixing of previously defined grain lots. In the absence of these internal records, it is impossible to link the incoming and the outgoing lots. Again, just before shipment, grain from different storage bins (i.e., different quality) is blended to meet the customer specifications for quality and to maximize the elevator's profit.

As a result of this grain elevator blending process, one storage bin likely contains grain from many different sources (i.e., original farmer lots), and a specific grain lot shipped to a customer (i.e., food processor or manufacturing plant) may contain grain from multiple sources. Any number of original farmer lots might ultimately comprise a finished food product. If a food related emergency occurred, isolating the source of the problem would be nearly impossible, so a recall of all the finished goods that might possibly have been contaminated would be the only method to ensure the consumer's safety. Such a recall would be time intensive and complex, result in high cost, be damaging to brand names, and add risk to consumers' safety. The following section reviews relevant literature related to traceability and database management systems.

1.2. Traceability and data management systems

The quality management systems (QMS) approach for grain traceability has been previously studied. Hurburgh and Sullivan (2004) demonstrated that a large grain elevator cooperative is able to track raw material through the elevator based on the implementation of a quality management system certified to the ISO 9001 standard. Laux and Hurburgh (2010) present the use of a QMS to create traceability system for a grain elevator. They discuss the use of mock recall procedures to demonstrate if a company meets requirements of the Bioterrorism Act of 2002. These studies however, do not present an electronic system to record and maintain data related to elevator operations. In this paper we present a multifunctional database model that can be used for internal traceability at the grain elevator and also provides a comprehensive dataset for other applications such as mass flow optimization, resource use optimization and overall operational optimization at the elevator. A data model is defined as a coherent representation of objects from a part of reality (Elmasri and Navathe, 2000). A wide range of systems is available for traceability in the food industry, ranging from paper-based systems to IT enabled systems (Food Standards Agency, 2002). Radio Frequency Identification (RFID) technology is also used to develop traceability systems in food supply chains (Natsui and Kyowa, 2004). RFID tags can be used for identification of individual product lots as they move through the supply chain. Information management and database management techniques are also used for developing traceability systems. Niederhauser et al. (2008) presents a conceptual information system for tracking specialty coffee while Jansen-Vullers et al. (2003) present a reference model designed to accommodate support for the registration

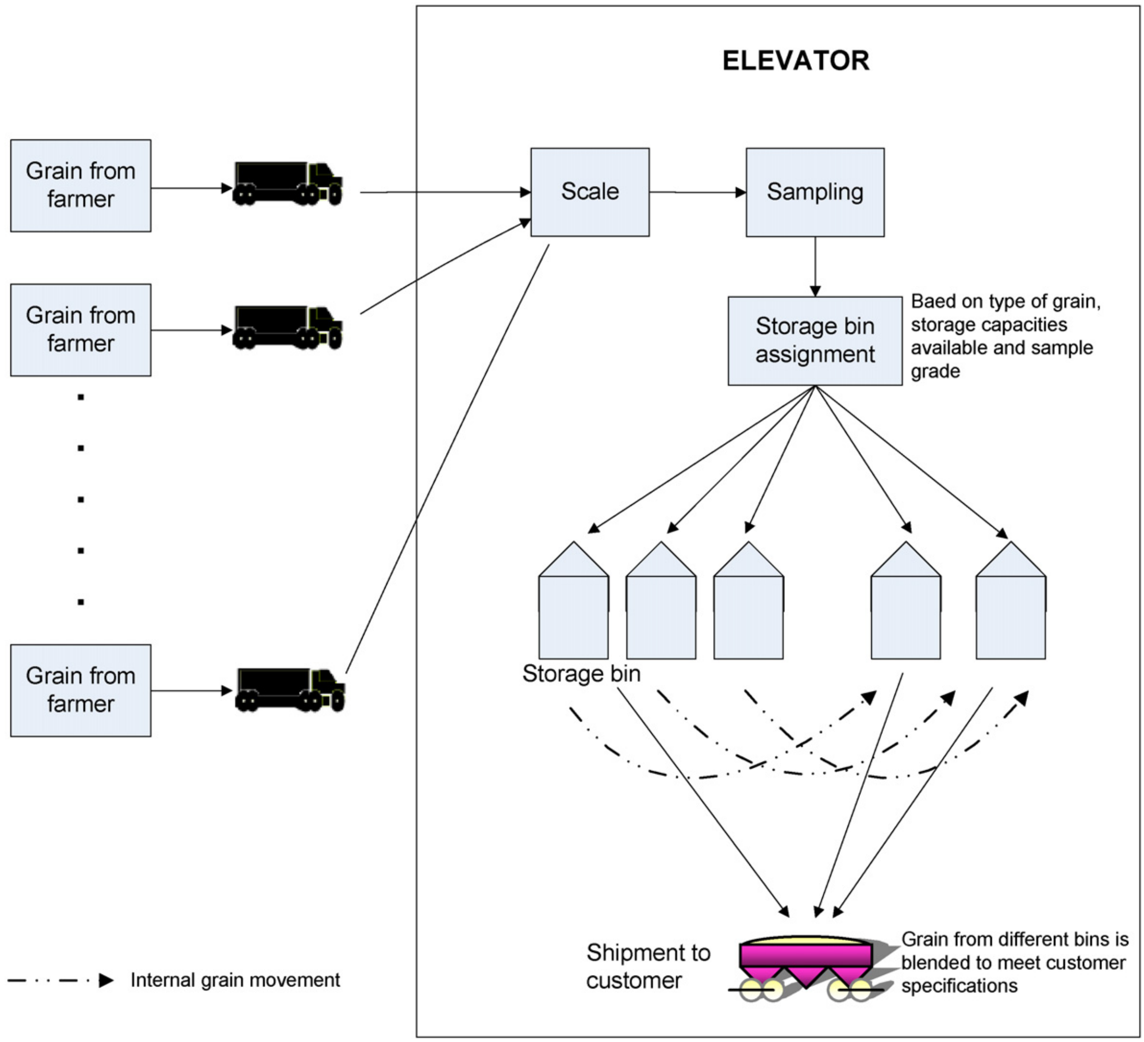

Fig. 1. A typical bulk grain handling scenario.

of operations on lots or batches and support for the registration of associated operation variables and values. This model (Jansen-Vullers et al., 2003) displays the functionality for traceability in manufacturing when production lots or batches are defined. Relational databases are widely used by corporations for operational management programs. The use of these databases for traceability in agricultural industry other than food manufacturing is, however, unheard of by the authors. Support for strategic decisions through analytical databases in the sense of data warehouses, as used and implemented intensively in the industrial sector has thus far not been given serious consideration in the agricultural sector (Schulze et al., 2007). It has been shown that the efficiency of a traceability system depends on its ability to record and retrieve the requested lot-related information (Folinas et al., 2006).

Senneset et al. (2007) state that one of the basic prerequisites of both internal and external supply chain traceability is the unique identification of all raw materials, semi-finished products and finished products. The authors offer three types of operations necessary for obtaining internal traceability:

(1) Recording the unique identities of traceable units. These usually refer to inputs to a process.
(2) Assigning unique identities to new traceable units. These usually refer to outputs from a process.
(3) Linking a set of input unit identities to one or more sets of output identities. These usually refer to transformation of raw materials to finished products.

Based on the concept of unique identification, a traceable unit (TU) is defined as any item with predefined information which may need to be retrieved and which may be priced, or ordered, or invoiced at any point in any supply chain. In practice, a TU refers to the smallest unit that is exchanged between two parties in the supply chain (TraceFood Wiki, 2009). In order to achieve chain traceability and meet the three traceability operations discussed above, efficient internal traceability systems must be in place at each food enterprise (node) in a supply chain. Therefore, it is important to develop systems which record both information related to traceable units and associated transformations occurring internally within each node. Such traceability systems can become complex, especially when TU are not well defined.

Since bulk grain is traded according to grade standards based on quality parameters of the grain lots, it is important to integrate the relevant quality data with the traceable units. Moe (1998) states that traceability can be used in four distinct contexts: product (origin, processing history, distribution and location after delivery), data generated throughout the quality loop, calibration (standards, physical properties, etc.), and IT and programming related to system design and implementation. Jansen-Vullers et al. (2003) suggest the following four elements for traceability:

(1) Physical lot integrity: this includes the lot size and how well the lot integrity is maintained.
(2) Data collection: this includes two types of data; lot tracing data and process data.
(3) Product identification and process linking: to determine product composition.
(4) Reporting: to retrieve data from the system.

Based on these principles, identification of data capture points and the data elements to be recorded at these points is the first step in developing a database management system for traceability.

For efficient grain supply chain traceability, the elevator has a responsibility to maintain data that links inputs (inbound deliveries) and outputs (outbound shipments). When needed, management should be able to retrieve the necessary information from this recorded data. In this paper, we propose the use of a relational database management system (RDBMS) for internal traceability at a grain elevator. The purpose of this database model is to record all the transformations related to incoming and outgoing grain lots as well as the transformations that take place internally at an elevator. Therefore, the objective of this database model is to track and trace individual grain lots through the bulk grain supply chain. The database can be queried to retrieve the relevant information when necessary. However, there are certain factors that create problems in modeling of the bulk grain handling data. The "fluid-like" characteristics of bulk grain distinguish it from other food products and make it very difficult to define a fixed lot-size (or traceable unit) for traceability purposes. The following section describes how these factors were modeled.

2. Methodology

2.1. Traceable units

Defining a lot or a traceable unit (TU) by breaking product flows into discrete units is a way to achieve product differentiation for tracking (Golan et al., 2004; Moe, 1998). However, the definition of a grain lot changes throughout the bulk handling process. In this database model, we use various definitions of a lot of bulk grain at different stages of handling within the elevator and each lot is uniquely identified. The following definitions of a grain lot are used:

1. At the time of purchase, a truckload of grain purchased from a farmer that is identified by a unique scale ticket number is considered a lot. This lot can be assigned to one or more storage bins depending on quality of grain and bin capacities available at that time.
2. In storage, the quantity of grain contained in one bin is considered as one lot. This lot can have multiple sub-lots (different incoming lots identified by unique scale ticket numbers). In storage, each lot is uniquely identified by the storage bin number.
3. For shipment to a customer, one truckload or the shipment load in one railcar is considered as one lot. This outgoing lot might come from several lots (in storage, each bin is a lot) blended together to meet the customer specifications. Each outgoing shipment has a corresponding customer contract and is uniquely identified by a shipment ID.

2.2. Lot transformations

Fig. 1 provides an overview of the lot dynamics at a grain elevator. Three types of activities related to incoming, internal and outgoing grain lots take place at an elevator. Each activity type can be defined by a set of transformations summarized in Table 1. Each lot transformation has a storage bin number associated with it because: (1) incoming grain is assigned to one or more bins, (2) grain can be moved internally from one bin to another and finally, (3) outgoing shipments are prepared by blending grain from different bins in order to meet customer specifications. So, this data model maintains information about lot transformations related to each bin in addition to activity date and time, farmer and customer information, and various grain quality parameters.

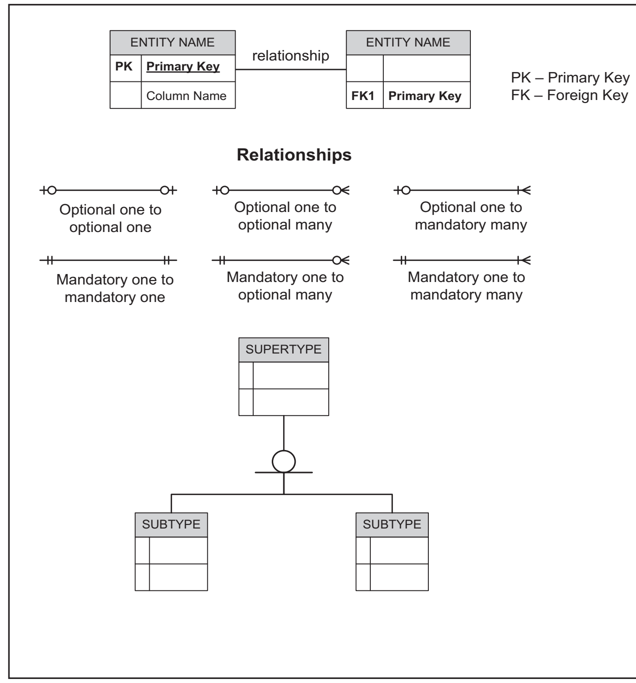

Fig. 2. Symbols used in an E-R model.

2.3. Entity-relationship model (ER model)

The entity-relationship (ER) modeling technique was used to develop the internal traceability grain handling database model. An ER model is a detailed, logical representation of data for an organization or for a business area. The ER model is represented in terms of entities in the business environment, the relationships among those entities, and the attributes of both the entities and their relationships (Hoffer et al., 2006). A relational database management system (RDBMS) is used for constructing and implementing the ER model. The benefits to using a RDBMS come from its ability to store data in a "normalized" format. This concept was originally presented by Codd (1970), who mathematically developed the relational model to provide a better structure for databases. Data normalization is simply a way of organizing data so that it allows for increased efficiency of data storage and retrieval. While spreadsheets can store data in a normalized format, it is very difficult to retrieve in a simple and timely manner. We developed a database designed to facilitate the storage, retrieval and analysis of grain handling data at an elevator. The internal traceability grain handling model was developed using Oracle Database 10g software. The rationale and principles used to develop this database are directly applicable to other commercially available RDBMS software. The design of the relational database adheres to the principles of normalization focusing on data handling efficiency and flexibility.

Table 1
Transformations associated with each grain lot activity.

Activity type	Transformation
Incoming grain purchased from farmer and transferred to a storage bin	1. Transfer: incoming grain lot is transferred to one or more storage bins 2. Aggregation: incoming lot is mixed with grain present in the assigned bin/s 3. Storage: incoming lot is stored in assigned bin/s until next transformation occurs
Grain is transferred internally from one bin to another	1. Transfer: internal grain lot is transferred to one or more storage bins 2. Segregation: a part of an internal lot (storage bin) is transferred to other bin/s 3. Aggregation: the transferred lot is mixed with grain present in the assigned bin/s 4. Storage: the transferred lot is stored in assigned bin/s until next transformation occurs
Grain lots from different storage bins are blended and shipped to the customer	1. Transfer: a part or entire internal lot (storage bin) is transferred from a bin 2. Segregation: a part of an internal lot (storage bin) is drawn from a bin for blending 3. Aggregation: the grain from different bins is blended together

Table 2
Relationship types in an entity-relationship model.

Relationship type	Description
One-to-one	There is exactly one instance in table A that corresponds to exactly one instance in related table B
One-to-many	There is exactly one instance in table A that corresponds to many instances in related table B
Many-to-one	There are many instances in table A that correspond to exactly one instance in related table B

Fig. 2 shows the symbols used in an ER model, which will be used in the later modeling steps. An entity stands for things that can be uniquely identified and characterized by their attributes; whereas relationships represent associations among different entities. Attributes represent information about an entity and relationship types by mapping them into value sets (Patig, 2006). A primary key is an attribute or combination of attributes that uniquely identify an instance in a database while a foreign key is used to link two tables (entities). Typically, a primary key from one table (entity) is inserted into another table (entity), and it then becomes a foreign key. Relationships between two entities work by matching the key columns in two tables. This is usually done by matching a primary key (that provides a unique row/instance) from one table to a foreign key instance in another table. Table 2 describes the different kind of relationships. Such relationships were developed for the grain

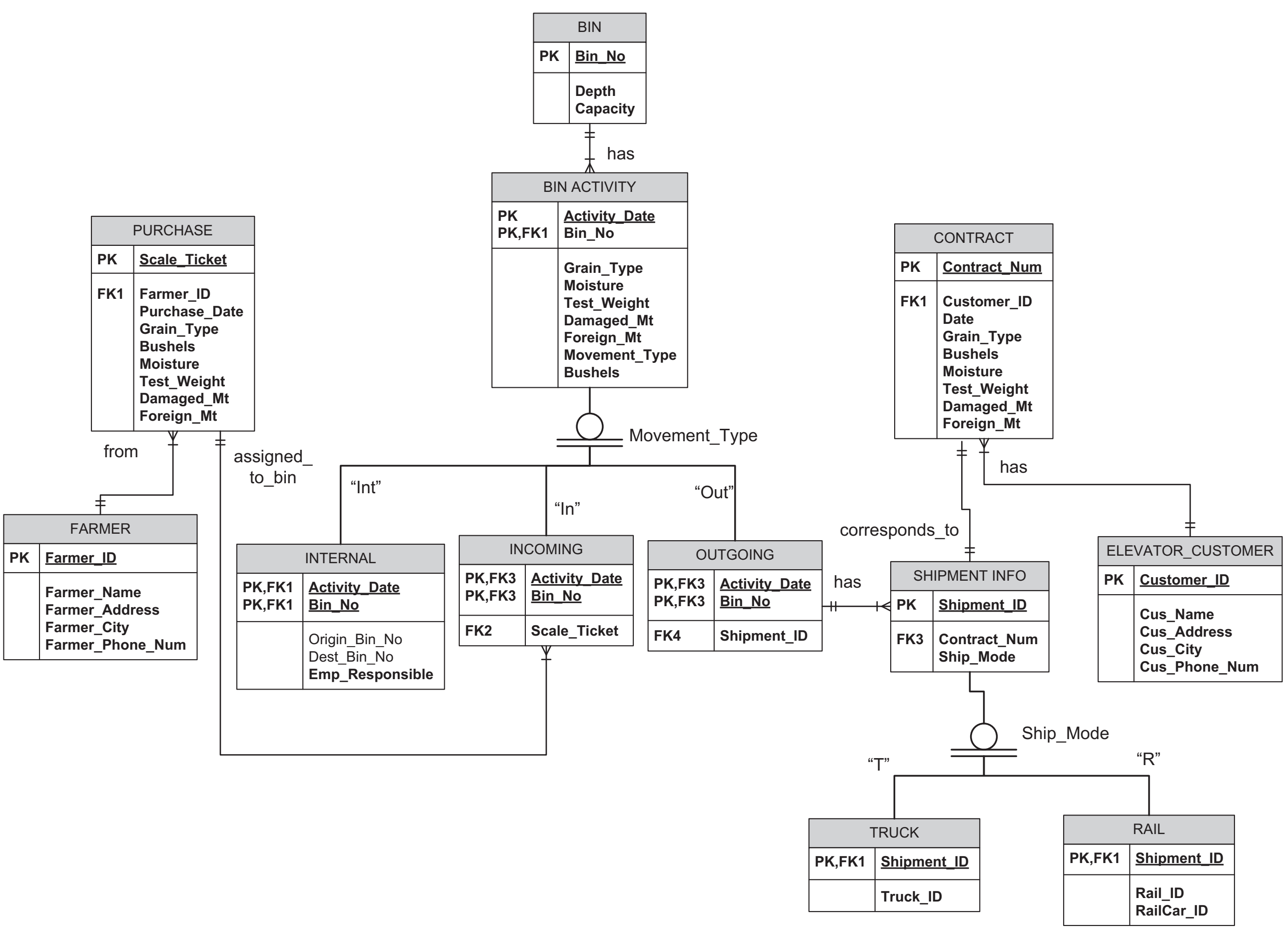

Fig. 3. Entity-relationship diagram for internal traceability at a grain elevator.

```
Trigger TRG_ACTIVITY_TYPE

CREATE OR REPLACE TRIGGER trg_activity_type
AFTER INSERT ON bin_activity
FOR EACH ROW
BEGIN
        IF :new.movement_type = 'Int' THEN
          INSERT into Internal(activity_date, bin_no) VALUES (:new.activity_date,
:new.bin_no);
        ELSIF :new.movement_type = 'In' THEN
          INSERT into Incoming(activity_date, bin_no) VALUES (:new.activity_date,
        :new.bin_no);
        ELSE
          INSERT into Outgoing(activity_date, bin_no) VALUES (:new.activity_date,
:new.bin_no);
        END IF;
END;

Trigger TRG_SHIP_MODE

CREATE OR REPLACE TRIGGER trg_ship_mode
AFTER INSERT ON shipment_info
FOR EACH ROW
BEGIN
        IF :new.ship_mode = 'R' THEN
          INSERT into Rail(shipment_ID) VALUES (:new.shipment_ID);
         ELSE
          INSERT into Truck(shipment_ID) VALUES (:new.shipment_ID);
  END IF;
END;
```

Fig. 4. Database triggers used for entities *bin_activity* and *shipment_info*.

lot activities/transformations and associated quality characteristics.

Fig. 2 also represents supertype and subtype entities. A supertype entity is used to represent two or more entities when they are viewed as the same entity by other entities. A subtype entity is an entity that is a special case of another entity, created when attributes or relationships apply to only some instances of an entity. The subsets of instances to which the attributes or relationships apply are separated into entity subtypes. When an attribute applies only to some occurrences of an entity, the subset of occurrences to which it applies should be separated into entity subtypes.

The common data elements are put in the supertype entity and the specific data elements are placed with the subtype to which they apply. All attributes of the supertype must apply to all subtypes. Each subtype contains the same key as the supertype. Database triggers can be used to automatically transfer data from supertype tables to subtype tables. A database trigger is a procedural code that is automatically executed in response to certain events on a particular table in a database (Hoffer et al., 2006). The Structured Query Language (SQL) was used to develop a functional model that can be implemented in a real elevator setting. Some sample reports and queries are discussed in the following sections.

3. Results

Fig. 3 shows the ER model for the internal traceability database at a grain elevator. Table 3 provides a description of each entity and the related attributes. Every time a transformation (aggregation, segregation, storage, transfer, etc.) takes place, the quality factors of moisture, test weight, foreign material and damaged material are recorded. A scale ticket number is assigned to the grain lots purchased from the farmers. Each incoming lot is tested for quality and transferred to one or more storage bins (that may already contain previous lots) depending on grain type (corn or soybeans), space availability and grain quality. The information related to the farmer and the activity dates are also recorded. Similar information is recorded when grain is moved internally at the elevator and for shipments to the customers (see Fig. 3 for details). The *bin_activity* entity has three sub-types, one each for the *internal*, *incoming*, and *outgoing* grain movement corresponding to every storage bin. Similarly, the *shipment_info* entity has two sub-types, *truck* and *rail*. The data is recorded in each table depending on the mode of transportation of the outgoing shipment. Database triggers were created for automatic data transfer to the sub-type tables.

By utilizing the relational database design, the proposed model can store, manage, retrieve all grain handling data and run calculations for aggregated quality of the blended products. The integration of all these functions makes this model unique from the existing spreadsheet based inventory control programs for grain elevators. This model combines inventory information, grain handling and grain quality information as well as the grain blending process in one centralized location.

3.1. Database triggers

A trigger is a named set of SQL statements that are considered (triggered) when a data modification (such as INSERT, UPDATE, and DELETE) occurs. If a condition stated within the trigger is met, then a prescribed action is taken (Hoffer et al., 2006). Triggers are commonly defined as ***On*** event ***If*** condition ***Then*** action (Dayal et al., 1988; Hanson, 1989; Kotz et al., 1988; Widom and Finkelstein, 1990). Triggers were used for two entities, namely, b*in_activity* and *shipment_info* to automatically transfer data from the supertype

```
SELECT p.farmer_ID, f.farmer_name, p.purchase_date, p.grain_type, p.bushels
FROM purchase p, incoming i, farmer f
WHERE p.scale_ticket = i.scale_ticket
AND f.farmer_ID = p.farmer_ID
AND bin_no = '9';
```

FARMER_ID	FARMER_NAME	PURCHASE_DATE	GRAIN_TYPE	BUSHELS
F0001	John Smith	16-Mar-08	Corn	2124
F0001	John Smith	16-Mar-08	Corn	1508
F0001	John Smith	16-Mar-08	Corn	3200
F0003	Pat Torreson	16-Mar-08	Corn	4205
F0003	Pat Torreson	16-Mar-08	Corn	3025
F0003	Pat Torreson	16-Mar-08	Corn	4850

Fig. 5. Sample query and report generated for incoming lot information.

Table 3
Description of entities in the ER model.

Table name (entity)	Attribute name	Contents
Bin	Bin_No	Grain storage bin number
	Depth	Bin depth (ft)
	Capacity	Bin capacity (bushels)
Bin_activity	Activity_Date	Bin activity date
	Bin_No	Grain storage bin number
	Grain_Type	Type of grain moved (corn or soybeans)
	Moisture	Average Moisture content of grain in the bin (%)
	Test_Weight	Average test weight of grain in the bin (lb/Bu)
	Damaged_Mt	Average percentage of damaged grain in the bin (%)
	Foreign_Mt	Average percentage of foreign material in the bin (%)
	Movement_Type	Type of movement (internal, inbound or outbound)
	Bushels	Quantity of grain moved in bushels
Internal	Activity_Date	Bin activity date
	Bin_No	Grain storage bin number
	Origin_Bin_No	Grain origin bin number
	Dest_Bin_No	Grain destination bin number
	Emp_Responsible	Name of employee responsible for moving grain
Incoming	Activity_Date	Bin activity date
	Bin_No	Grain storage bin number
	Scale_Ticket	Scale ticket number of inbound grain in elevator
Outgoing	Activity_Date	Bin activity date
	Bin_No	Grain storage bin number
	Shipment_ID	ID of outbound shipment
Shipment_info	Shipment_ID	ID of outbound shipment
	Contract_Num	Contract number of shipment
	Ship_Mode	Shipment mode (truck or rail)
Truck	Shipment_ID	ID of outbound shipment
	Truck_ID	ID of truck for outbound shipment
Rail	Shipment_ID	ID of outbound shipment
	Rail_ID	ID of rail for outbound shipment
	Railcar_ID	ID of railcar for outbound shipment
Elevator_customer	Customer_ID	Customer ID
	Cus_Name	Customer name
	Cus_Address	Customer address
	Cus_City	Customer city
	Cus_Phone_Num	Customer phone number
Contract	Contract_Num	Contract number – outbound shipment
	Customer_ID	Customer ID for shipment
	Contract_Date	Date of contract
	Grain_Type	Type of grain
	Bushels	Quantity of grain required in bushels
	Moisture	Max. moisture content of grain required on contract (%)
	Test_Weight	Min. test weight of grain required on contract (lb/Bu)
	Damaged_Mt	Max. allowable damaged grain on contract (%)
	Foreign_Mt	Max. allowable foreign material on contract (%)
FARMER	Farmer_ID	Farmer ID
	Farmer_Name	Farmer name
	Farmer_Address	Farmer address
	Farmer_City	Farmer city
	Farmer_Phone_Num	Farmer phone number
Purchase	Scale_Ticket	Scale ticket number of inbound grain in elevator
	Farmer_ID	Farmer ID
	Purchase_Date	Date of purchase
	Grain_Type	Type of grain purchased (Corn or Soybeans)
	Bushels	Quantity of grain purchased in Bushels
	Moisture	Moisture content of grain purchased (%)
	Test_Weight	Test Weight of grain purchased (lb/Bu)
	Damaged_Mt	Damaged matter in grain purchased (%)
	Foreign_Mt	Foreign matter in grain purchased (%)

entity to the respective subtype entities based on the response (i.e., the type of activity). SQL code for these database triggers is shown in Fig. 4. It can be noted that data is added to the respective subtype entities using the triggers based on the type of movement and the type of shipment mode, respectively, for the two supertype entities.

3.2. Queries and reports

Once the data is stored in the database, the manipulation is accomplished through the use of queries written using the Structured Query Language (SQL). The set of queries presented in this section act as a start for basic data retrieval, but the WHERE clauses should all be changed to match specific data requirements. Once written these queries can be saved and easily executed at a later date but would return varying results based on the changes made to the data set during that time. Some sample reports are shown in this section of the paper. The main purpose of this database is to be able to connect the incoming grain lots with the outgoing grain lots. This information is vital in case of a food safety related emergency. Reports can be generated from the database to answer queries such as:

- Which farmers supplied the grain contained in a specific storage bin?

```
SELECT DISTINCT o.activity_date, c.contract_num, c.customer_id, o.bin_no, t.truck_ID,
b.bushels
FROM contract c, outgoing o, shipment_info s, truck t, bin_activity b
WHERE c.contract_num = s.contract_num
AND s.shipment_ID = o.shipment_ID
AND t.shipment_ID = o.shipment_ID
AND b.activity_date = o.activity_date;
```

ACTIVITY_DATE	CONTRACT_NUM	CUSTOMER_ID	BIN_NO	TRUCK_ID	BUSHELS
02-MAY-08 10.21.00 AM	CA031708	C0004	2	20001	1500
02-MAY-08 10.21.00 AM	CA031708	C0004	11	20001	1500
02-MAY-08 02.25.00 PM	CG040608	C0005	9	20002	4000

Fig. 6. Sample query and report generated for outgoing lot information using truck as transportation mode.

- Which bins were used to blend grain for a specific outgoing shipment?
- Which incoming lots contributed to a specific outgoing shipment?

Fig. 5 shows the SQL code and sample report generated to display the farmer information, purchase date, grain type and quantity purchased that was transferred to storage bin number 9.

Fig. 6 shows the SQL code and sample report generated to display the outgoing shipments using truck as transportation mode. The report includes the activity date (shipment date), contract number, customer ID, the bin number/s from where the grain is drawn for blending, truck ID and the quantity shipped on each truck in bushels. Similarly, Fig. 7 shows the code and report generated to display the outgoing shipments using rail as transportation mode.

The ability to link the outgoing lot (shipment) information to the incoming lots is important to trace back the source of problem in case of a food safety emergency. Fig. 8 shows the SQL code and sample report generated to display the incoming grain lot information corresponding to outgoing shipments to Company A. The query is created so that the report includes the scale ticket number of the incoming lots, purchase date, farmer name, quantity purchased in bushels, bin number assigned to the incoming lot, activity date (shipment date), contract number, bin number/s from where the grain is drawn, and the quantity shipped on each railcar in bushels. This report displays the incoming lots that are present in an outgoing shipment. The grain lots are divisible so a part or an entire incoming grain lot may be present in an outgoing lot. This information can be used to trace back the origin of grain (back to a farmer or a group of farmers) present in an outgoing shipment.

```
SELECT DISTINCT o.activity_date, c.contract_num, c.customer_id, o.bin_no, r.rail_id,
r.railcar_id, b.bushels
FROM contract c, outgoing o, shipment_info s, rail r, bin_activity b
WHERE c.contract_num = s.contract_num
AND s.shipment_ID = o.shipment_ID
AND r.shipment_ID = o.shipment_ID
AND b.activity_date = o.activity_date;
```

ACTIVITY_DATE	CONTRACT_NUM	CUSTOMER_ID	BIN_NO	RAIL_ID	RAILCAR_ID	BUSHELS
25-MAR-08 10.25.00 AM	C032208	C0001	8	10001	1	2000
25-MAR-08 10.25.00 AM	C032208	C0001	2	10001	1	5000
28-APR-08 11.30.00 AM	A042508	C0002	11	10001	11	6000
25-MAR-08 10.25.00 AM	C032208	C0001	2	10001	1	2000
25-MAR-08 10.25.00 AM	C032208	C0001	8	10001	1	5000
28-MAR-08 10.25.00 AM	CG040908	C0005	2	10003	12	664
29-APR-08 09.25.00 AM	G042808	C0003	9	10002	2	5000

Fig. 7. Sample query and report generated for outgoing lot information using railcars as transportation mode.

```
SELECT DISTINCT p.scale_ticket, p.purchase_date, f.farmer_name, i.bin_no, p.bushels,
o.activity_date, c.contract_num, e.cus_name, o.bin_no, s.ship_mode, b.bushels
FROM purchase p, farmer f, incoming i, contract c, outgoing o, shipment_info s, elevator_customer e,
bin_activity b
WHERE c.contract_num = s.contract_num
AND s.shipment_ID = o.shipment_ID
AND b.activity_date = o.activity_date
AND p.farmer_id = f.farmer_id
AND p.scale_ticket = i.scale_ticket
AND i.bin_no = o.bin_no
AND c.customer_id = e.customer_id
AND cus_name LIKE '%Company A%'
Order by o.bin_no;
```

SCALE _TICKET	PURCHASE _DATE	FARMER _NAME	BIN _NO	BUSHELS	ACTIVITY _DATE	CONTRACT _NUM	BIN _NO	BUSHELS
1011	15-Mar-08	Ron Penning	2	1564	28-MAR-08 10.25.00 AM	CG040908	2	664
1010	15-Mar-08	Ron Penning	2	2200	28-MAR-08 10.25.00AM	CG040908	2	664
1019	16-Mar-08	John Smith	9	1508	02-MAY-08 02.25.00 PM	CG040608	9	4000
1020	16-Mar-08	John Smith	9	2124	02-MAY-08 02.25.00 PM	CG040608	9	4000
1018	16-Mar-08	John Smith	9	3200	02-MAY-08 02.25.00 PM	CG040608	9	4000
1046	16-Mar-08	Pat Torreson	9	3025	02-MAY-08 02.25.00 PM	CG040608	9	4000
1047	16-Mar-08	Pat Torreson	9	4205	02-MAY-08 02.25.00 PM	CG040608	9	4000
1045	16-Mar-08	Pat Torreson	9	4850	02-MAY-08 02.25.00 PM	CG040608	9	4000

Fig. 8. . Sample query and report generated to connect incoming and outgoing lot information.

4. Conclusions

Development of data management systems to facilitate product traceability in food supply chains has gained importance in the past years. The ability to track and trace individual product units depends on an efficient supply chain traceability system which in turn depends on both internal data management systems and information exchange between supply chain actors. In this paper, we present a relational database model to facilitate internal traceability at a grain elevator.

Grain elevators handle bulk commodities marketed against generic grade standards that are based on physical attributes. Different lot-activities take place as the grain moves through the supply chain from the farm to the consumer. At an elevator, grain lots (inbound deliveries) are commingled to meet buyer specifications, and lot identity is not maintained. As a result, an outbound shipment to a customer can contain grain from many sources. In a food safety related emergency, it would be almost impossible to trace back the source of problem and to track (forward) other affected lots. This process is very time intensive, increases the recall costs, and can lead to a tainted brand name for the company. The problem can be mitigated by an efficient internal record keeping system that would document all grain activities (transformations). The proposed database system stores product identity and transformation information related to grain lots (traceable units) and can be queried to retrieve information related to all incoming, internal and outgoing lots.

Definition of a lot size or a traceable unit was an important step in developing a data management system since all the information has to be linked to a unique entity, which in general is a specific lot size. But, grain is handled in bulk and defining a lot size is a complex task. So, instead of a strict definition of a lot, we use several definitions and explain how the lot size changes as grain moves through an elevator. Each receipt from a farmer (usually, a truckload) is assigned a unique scale ticket number and considered as one lot. When in storage, a grain bin is considered as one lot which in turn can contain grain from different farmer deliveries (scale tickets). This implies that a storage bin can contain many sub-lots. Again, when the grain is shipped to a customer, an outgoing shipment is prepared by blending grain from different storage bins in order to meet customer specifications. For an outgoing shipment, a railcar or a truckload (depending on the transportation mode) is considered as one lot.

The entity-relationship modeling technique was used to develop the database management system for internal traceability. All the information related to the grain lot activities/transformations and associated quality characteristics were recorded in this database. An important feature of the ER model is the use of supertype and subtype entities. Two entities, the type of grain lot movement and the mode of transportation were modeled as supertype entities. This feature simplified the database design and information retrieval. Depending on the type of movement; whether it is an incoming grain activity, internal activity or an outgoing activity, the information is stored in the corresponding tables. This design was used because these entities (different movement types) share some common attributes. The common attributes such as the quality parameters are placed in the supertype entity *bin_activity* while the specific attributes *scale_ticket*, *shipment_ID*, etc. are placed in the subtype entity to which they apply. Another feature of this model is the use of database triggers. Triggers were used to automatically transfer data from the supertype entity to the subtype entities.

The database can be queried to retrieve information related to any grain lot activity (transformation). It can be used to trace back the source of a given lot or track forward the information related to the shipped lots. The information that connects the individual incoming grain lots to an outgoing lot can also be retrieved using this system as is shown by some sample queries in Section 3.

This paper demonstrates that using a relational database management approach for recording all lot activities (transformations) is an effective way to link the incoming and outgoing grain lots at an elevator.

The next steps in this work include the development of a graphical user interface to enable the users to enter data in the database. The model also needs to be implemented in a real elevator setting and tested for performance based on the response time of information retrieval in case of a product recall.

5. Practical implications

Currently, the grain lot data presented in this paper is handled by several stand-alone applications at the elevator and sometimes just in paper format. This makes it difficult to access and compile the data in order to analyze elevator business functions using a system perspective, which is necessary for operational optimization. This model can integrate information from several stand-alone applications in one central system connected to unique entities, grain lots in this case. In addition to traceability application of this model, the information stored in this database provides a comprehensive dataset for many applications including mass flow optimization, resource optimization, improved operational efficiency as well as environmental impact analysis of the grain elevator.

The information recoded in this format can be extracted and used to meet both operational and analytical requirements of the business. The operational requirements of an enterprise's business processes generally include short-term decision making while analytical requirements refer to long-term decision making based on historical and aggregated data. The historical data recorded over long term using a relational database system could be analyzed to study the grain handling practices of the elevator. Elevators move grain from one bin to another and between different elevator locations based on space and quality constraints. Availability of historical data would allow the elevator management to analyze their grain handling practices and to define new procedures in order to optimize the logistics costs and to minimize the food safety risk by optimizing their blending practices.

Currently, sustainability in food supply chains is gaining significant importance. The data stored in the proposed model can be used to perform environmental impacts analysis by linking energy consumption to each activity at the grain elevator, for instance, by linking this information to internal grain movement as well as shipment to the customers. Since, each process is tracked separately, it would be possible to identify and optimize processes with high energy consumption.

The model presented in this paper is not just another process modeling mechanism but is a multi-functional tool that can be used for various applications within the grain industry.

If such database systems are used by all actors in the supply chain, there would be no information loss internally at an enterprise and this information exchanged between the actors would be complete.

Acknowledgement

The authors would like to thank Mr. Butch Hemphill of Farmers Cooperative Company, Iowa for his help in understanding the elevator operations.

References

Carriquiry, M., Babcock, B.A., 2007. Reputations, market structure and the choice of quality assurance systems in the food industry. American Journal of Agricultural Economics 89, 12–23.

Codd, E.F., 1970. A relational model of data for large shared data banks. Communications of the Association of Computing Machinery 13 (6), 377–387.

Dayal, U., Blaustein, B., Buchmann, A., Chakravarthy, U., Hsu, M., Ledin, R., McCarthy, D., Rosenthal, A., Sarin, S., Carey, M.J., Livny, M., Jauhari, R., 1988. The HiPAC project: combining active databases and timing constraints. ACM SIGMOD Record 17 (1), 51–70.

Donnelly, K.A., Karlsen, K.M., Olsen, P., 2009. The importance of transformations of traceability – a case study of lamb and lamb products. Meat Science 83, 68–73.

Elmasri, R., Navathe, S.B., 2000. Fundamentals of Database Systems, 3rd ed. Addison-Wesley, Reading, MA, USA.

Folinas, D., Manikas, I., Manos, B., 2006. Traceability data management for food chains. British Food Journal 108 (8), 622–633.

Food Standards Agency, 2002. Traceability in the Food Chain; A Preliminary Study. Food Chain Strategy Division, Food Standards Agency.

Golan, E., Krissoff, B., Kuchler, F., 2004. Food traceability: one ingredient in a safe and efficient food supply, economic research service. Amber Waves 2, 14–21.

GS1 Global Traceability Standard, 2007. Business Process and System Requirements for Full Chain Traceability. http://www.gs1.org/traceability/gts.

International Organization for Standardization, 2007. New ISO Standard to Facilitate Traceability in Food Supply Chains. ISO 22005:2007.

Hanson, E.N., 1989. An initial report on the design of Ariel: a DBMS with an integrated production rule system. SIGMOD Record 18 (3), 12–19.

Hoffer, J.A., Prescott, M., McFadden, F., 2006. Modern Database Management. Prentice Hall, New Jersey.

Hurburgh, C.R., Sullivan, T., 2004. An ISO-based system for quality management and traceability in the US grain handling industry. In: International Quality Grains Conference: A Global Symposium on Quality-assured Grains and Oilseeds for the 21st Century. Indianapolis. U.S. Grain Quality Research Consortium (NC-213).

Jansen-Vullers, M.H., van Dorp, C.A., Buelens, A.J.M., 2003. Managing traceability information in manufacture. International Journal of Information Management 23, 395–413.

Kotz, A.M., Dittrich, K.R., Mülle, J.A., 1988. Supporting semantic rules by a generalized event/trigger mechanism. In: Proceedings of the International Conference on Extending Database Technology: Advances in Database Technology, March 14–18, pp. 76–91.

Laux, C., Hurburgh, C.R., 2010. Meeting FDA traceability requirements with quality management systems. Journal of Industrial Technology 26 (3), 1–10.

Madec, F., Geers, R., Vesseur, P., Kjeldsen, N., Blaha, T., 2001. Traceability in the pig production chain. Revue Scientifique Et Technique (International Office of Epizootics) 20 (2), 523–537.

McKean, J.D., 2001. The importance of traceability for public health and consumer protection. Revue Scientifique Et Technique (International Office of Epizootics) 20 (2), 363–371.

Moe, T., 1998. Perspectives on traceability in food manufacture. Trends in Food Science and Technology 9, 211–214.

Natsui, T., Kyowa, A., 2004. Traceability system using RFID and legal issues. WHOLES, A multiple view of individual privacy in a networked world. www.sics.se/privacy/wholes2004/papers/takato.pdf.

Niederhauser, N., Oberthür, T., Kattnig, S., Cock, J., 2008. Information and its management for differentiation of agricultural products: the example of specialty coffee. Computers and Electronics in Agriculture 61 (2), 241–253.

Official Journal of the European Communities, 2002. Regulation (EC) No. 178/2002 of the European Parliament and the Council of 28 January 2002.

Patig, S., 2006. Evolution of entity-relationship modeling. Data and Knowledge Engineering 56 (2), 122–138.

Schulze, C., Spilke, J., Lehner, W., 2007. Data modeling for precision dairy farming within the competitive field of operational and analytical tasks. Computers and Electronics in Agriculture 59 (1–2), 39–55.

Senneset, G., Forås, E., Fremme, K.M., 2007. Challenges regarding implementation of electronic chain traceability. British Food Journal 109 (10), 805–818.

Schwägele, F., 2005. Traceability from a European perspective. Meat Science 71 (1), 164–173.

Thakur, M., Hurburgh, C.R., 2009. Framework for implementing traceability in the bulk grain supply chain. Journal of Food Engineering 95 (4), 617–626.

TraceFood Wiki, 2009. http://www.tracefood.org.

US Food and Drug Administration, 2002. The Bioterrorism Act of 2002.

Widom, J., Finkelstein, S.J., 1990. Set-oriented production rules in relational database systems. In: Proceedings of the 1990 ACM SIGMOD International Conference on Management of Data, May 23–26, Atlantic City, New Jersey, pp. 259–270.