Teaching Case
The Animal Genetic Resource Information Network (AnimalGRIN) Database: A Database Design & Implementation Case

Gretchen Irwin
Computer Information Systems
Colorado State University
Fort Collins, CO USA 80523-1277
Gretchen.irwin@colostate.edu

Lark Wessel
4704 Austin Street, Houston, Texas USA 77004

Harvey Blackburn
USDA, National Center for Genetic Resources
National Animal Germplasm Program
Fort Collins, CO USA 80521

ABSTRACT

This case describes a database redesign project for the United States Department of Agriculture's National Animal Germplasm Program (NAGP). The case provides a valuable context for teaching and practicing database analysis, design, and implementation skills, and can be used as the basis for a semester-long team project. The case demonstrates the broad applicability of database skills to domains outside typical business applications. The functional requirements for the NAGP's information system are documented with a detailed set of use cases, which are commonly used in practice and which provides a link between the database course and the systems analysis and design course in many information systems curricula. The NAGP database redesign project was initiated because the original database design was not flexible or rigorous enough to handle the ways in which the organization's needs grew. The case provides several opportunities to emphasize design flexibility and data integrity controls.

Keywords: Database design & development, Entity-relationship modeling/diagram, Structured query language

1. INTRODUCTION

Organizations rely on databases to provide accurate and timely data to support their operations and decision making. Most university information systems programs include a database course on the fundamental analysis, design, and implementation skills needed to create these databases. In our experience, students gain the most from this course by practicing these skills on multiple examples, both large and small, over an extended period of time. In addition to the examples used in class, small homework assignments, and the textbook, we also assign a semester-long case-based project. The project deliverables correspond to the main course objectives, which state that students will be able to: (1) create a conceptual data model for an information system; (2) translate a conceptual model into a logical database design (in third normal form); (3) implement a database using a database management system (e.g., Microsoft SQL Server); (4) test the data integrity controls on the database; and (5) formulate queries to answer end-user information requests.

This paper presents a case that we have used in undergraduate and graduate database courses. The case is based on one of the authors' experience redesigning a database for the United States Department of Agriculture's National Animal Germplasm Program (NAGP), and it differs from other project-based teaching cases (e.g., Parker 2005, Cappel & Gillman 2011) in several ways. First, the case presents a novel domain for business students. The new domain of animal sciences helps students appreciate the broad applicability of the database skills they learn in the course, and illustrates the importance of learning the users' domain to create conceptual data. While the domain is new, many of the deeper, underlying modeling patterns (Fowler 1997) are found in business database systems as well. This allows us to draw analogies in class from different domains and help students recognize the abstract patterns (e.g., recursive structures) that apply to many situations. The case also provides a clear link to other courses because the system's functional requirements are documented with a detailed set of Use Case Descriptions. Use cases are often taught in systems analysis and design courses and are widely used in practice, so students gain experience working with requirements models that they are likely to see on actual systems development projects.

The NAGP's database was functioning but in need of a complete redesign because the original design was not flexible enough to handle the growth and changes in the data and because the original database did not have sufficient data integrity controls in place. The case provides opportunities to compare alternative designs and discuss business rules that can be implemented with data integrity controls.

The rest of the paper is organized as follows. The following section provides the background on the case and the motivation for the database redesign project. Section three discusses the requirements for the new database system and the business processes that the database must support. Section four outlines the project deliverables for the case. The last section summarizes our experience with the case and how it can be adapted to fit different course objectives.

2. CASE BACKGROUND

In 2007, the United Nations held an International Technical Conference on Animal Genetic Resources to discuss global animal genetic diversity concerns. Researchers reported that more than sixty breeds of livestock had become extinct in the prior decade and twenty percent more of the world's breeds are at risk of becoming extinct (FAO, 2007). The conference emphasized the importance of improving germplasm[i] preservation programs to manage worldwide animal genetic resources. These programs provide the ability to regenerate a breed, preserve valuable livestock breeds, enhance genetic diversity, and research threats to biodiversity such as parasites and diseases that attack specific species and breeds.

The United States Department of Agriculture's Agricultural Research Service started the NAGP in 1999 to collect, preserve, and document germplasm from livestock in the United States (http://www.ars-grin.gov/). Through this program, the US has developed the largest germplasm collection and physical livestock gene repository in the world. An important part of this program is the web-enabled information system called the Animal Genetic Resource Information Network, or AnimalGRIN (www.ars-grin.gov/). AnimalGRIN records information about the germplasm donated to and requested from the repository, and the animals from which the germplasm is drawn. The primary users of AnimalGRIN are breeders and scientists trying to sustain and improve livestock species and breeds.

Version 1 of AnimalGRIN was implemented in 2000 using *Oracle 10g* and *Oracle Forms* and was designed primarily to track cattle breeds. Since then, the database has grown substantially and in unanticipated ways. As of June 2012, the database contained information on roughly 706,000 germplasm samples from over 17,000 animals, representing over three hundred different animal taxonomies[ii], including aquatic species, cattle breeds, pig breeds/lines, goat breeds, sheep breeds, and poultry lines. The database also has information on approximately 10,000 additional animals that are part of the pedigree[iii] of the donor animals.

The addition of other species of agriculturally vital animals highlighted the need for a more flexible database design, in part because these diverse species and breeds do not share a common taxonomic structure. There are also other issues—including the addition of new repositories in Canada and Brazil—that are outside the scope of this case, but which emphasized the limitations of the original database. Hence, the NAGP requested a complete redesign to create a database that is more flexible, has more data integrity controls, and is easier for scientists to use.

3. DATABASE REQUIREMENTS

The AnimalGRIN database supports the inventory management function of the NAGP. This includes receiving incoming shipments (called orders) of germplasm into the repository as well as fulfilling requests for germplasm samples from breeders and researchers. The scope of this case is on the germplasm *coming into* the repository, not on the requests for samples removed *from* the repository. The process of receiving germplasm into inventory is very time-consuming and data-entry intensive, and includes documenting information about:

- **Incoming orders** of germplasm (e.g., the date the order is received, who is donating the germplasm);
- **Germplasm donations**, or inventory items, on each order (e.g., the type and quantity of germplasm);
- **Animals** from whom the germplasm came (e.g., the animal's breeder, name, registration number, date of birth, pedigree, and taxonomy); and
- **Inventory location(s)** within the repository assigned to each germplasm donation.

Figure 1 shows a use case diagram that provides an overview of the functional requirements of AnimalGRIN.[iv] Each of the use cases is described below.

3.1 Receiving Orders into Inventory.

The most important requirement for AnimalGRIN is to support the Receive Order into Inventory use case. This use case begins when the inventory specialist receives a packing list for an incoming order of germplasm donations, and opens a new incoming order form in AnimalGRIN. Figure 2 shows a sample packing list with some of the information that must be recorded in AnimalGRIN. The packing lists are often hand-written documents that are incomplete, and the scientist must do additional research in order to fully document the order and the animals whose germplasm is being donated.

When a new order is opened in AnimalGRIN, the system records the received date and assigns a unique order number. The scientist specifies information about the individuals and/or organizations related to the order, such as the germplasm owner (donor) and the shipper. The Manage Person/Organization use case explains how the scientist finds, adds, and updates information about these entities in the database.

Next, the scientist documents the animals whose germplasm is contained in the order, using the Manage Animal use case. The packing list contains some information that helps identify the animal, such as the animal's name (see "sire name" in Figure 2) or registration number. The scientist uses this information to see whether the animal on the packing list already exists in AnimalGRIN, and if so, the scientist "attaches" the animal record to the corresponding order line.

If the animal does not already exist in the database, then the scientist creates a new animal record in the database and enters general information about the animal (e.g., gender, date of birth, birthplace, and breeder) as well as specific identification records (e.g., name, tattoo, and/or registration number). If identification details are not available from the packing list, the scientist researches the animal through the official breed web site or other sources.

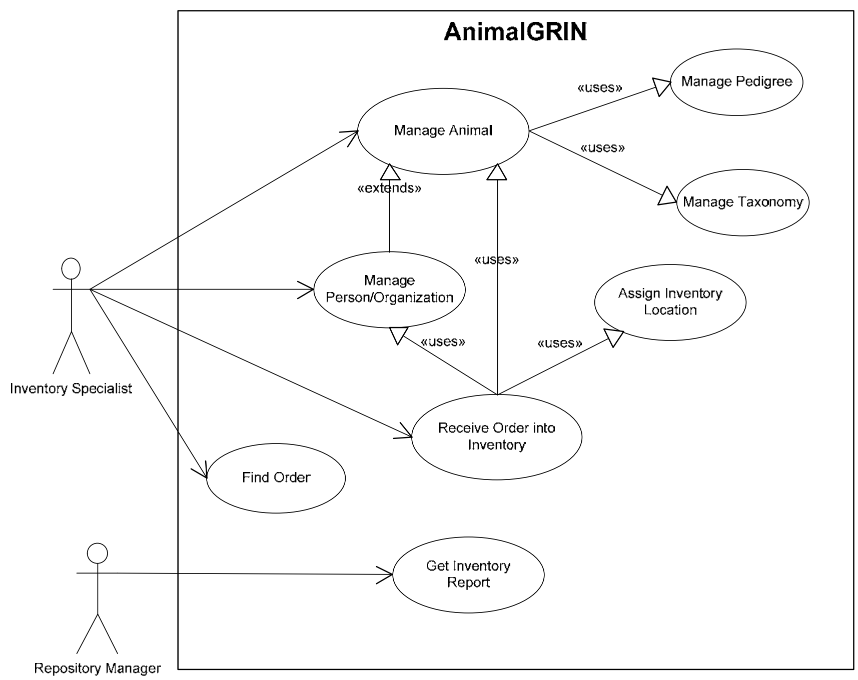


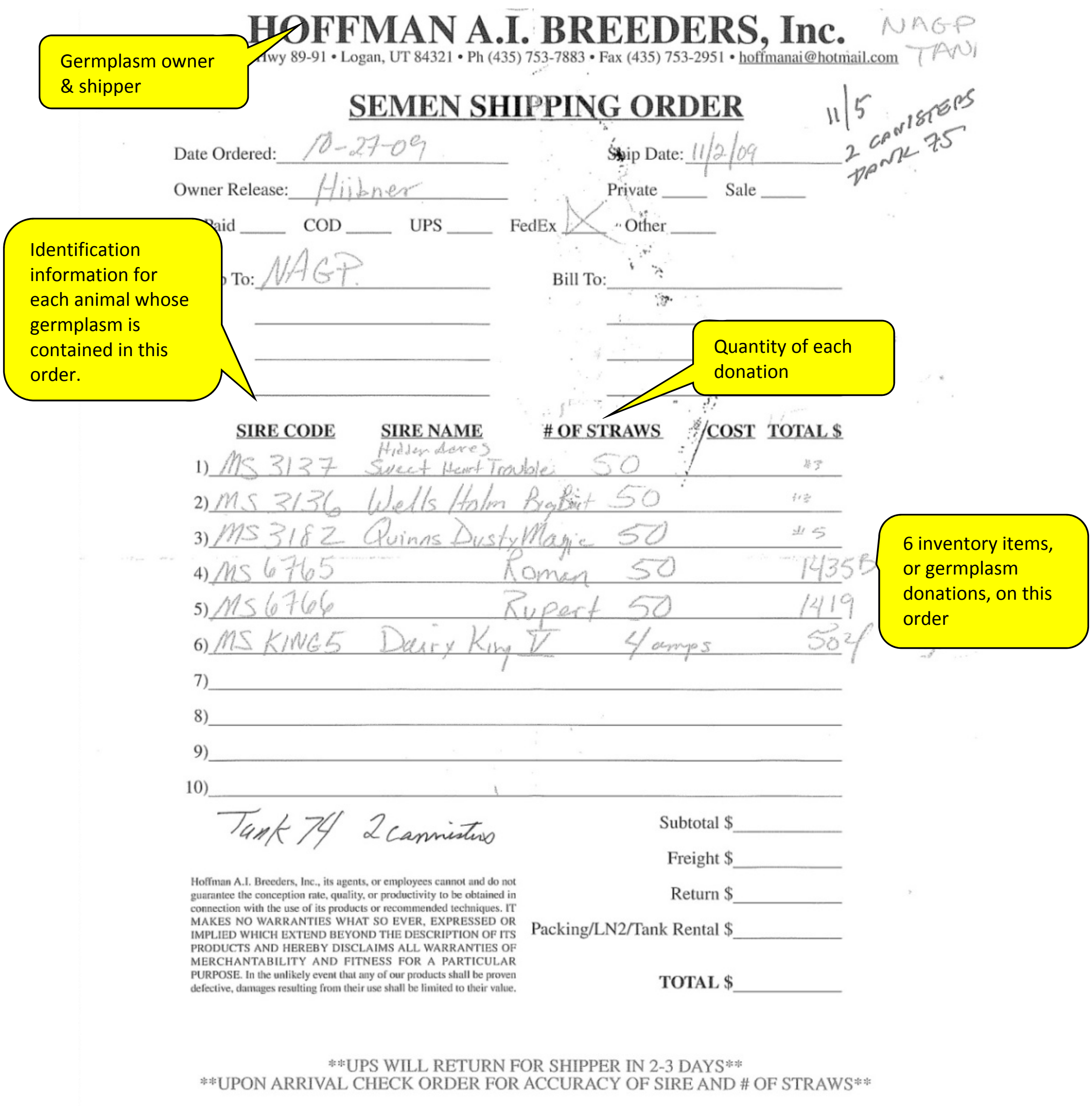
Figure 1: Use Case Diagram for the AnimalGRIN System.

These identifications are critical to matching each animal to the appropriate sire (father) or dam (mother), and to the appropriate donation and germplasm, because the packing lists do not use the ID numbers assigned by the AnimalGRIN system. Without the identification information one Hereford bull in the system would look much like another and it would be nearly impossible to know whether the "right" animal is being attached to a donation. The **Manage Animal** use case (see Figure 1) describes how the scientist finds/updates animals in the database and adds new animals to the database.

Animal documentation is not complete without pedigree and taxonomy details, so that scientists can assess the bio-diversity of a taxonomy within the NAGP repository. If an order contains germplasm from an animal that already exists in AnimalGRIN, then pedigree and taxonomy details should already be specified. However, when the first donation from an animal is received, the **Manage Animal** use case invokes the **Manage Pedigree** and **Manage Taxonomy** use cases (see Figure 1) so that additional documentation can be recorded.

The **Manage Pedigree** use case allows the scientist to create, search, and update an animal's pedigree or family tree. The family tree for one animal is shown in Figure 3. Ideally, every repository animal would have a documented family tree at least as large as the one shown in Figure 3. However, the depth of the tree depends on how much information is available.

If the animal's sire (father) and dam (mother) are not in the database, the scientist creates new animal records for them, even if there is no sire/dam germplasm. Thus, some of the animals in the database are *repository* animals (i.e., their germplasm is stored in the NAGP), while other animals in the database are *non-repository* animals (i.e., they are part of the pedigree of one or more repository animals).



HOFFMAN A.I. BREEDERS, Inc.

Hwy 89-91 • Logan, UT 84321 • Ph (435) 753-7883 • Fax (435) 753-2951 • hoffmanai@hotmail.com

NAGP TANI

SEMEN SHIPPING ORDER

11/5 2 CANISTERS TANK 75

Date Ordered: 10-27-09　　Ship Date: 11/2/09

Owner Release: Hibner　　Private ____ Sale ____

Paid ____ COD ____ UPS ____ FedEx X Other ____

Ship To: NAGP　　Bill To: ____

	SIRE CODE	SIRE NAME	# OF STRAWS	COST	TOTAL $
1)	MS 3137	Hidden Acres Sweet Heart Trouble	50		#3
2)	MS 3136	Wells Holm Bugbit	50		#2
3)	MS 3182	Quinns Dusty Magic	50		#5
4)	MS 6765	Roman	50		1435B
5)	MS 6766	Rupert	50		1419
6)	MS KING5	Dairy King V	4 amps		504
7)					
8)					
9)					
10)					

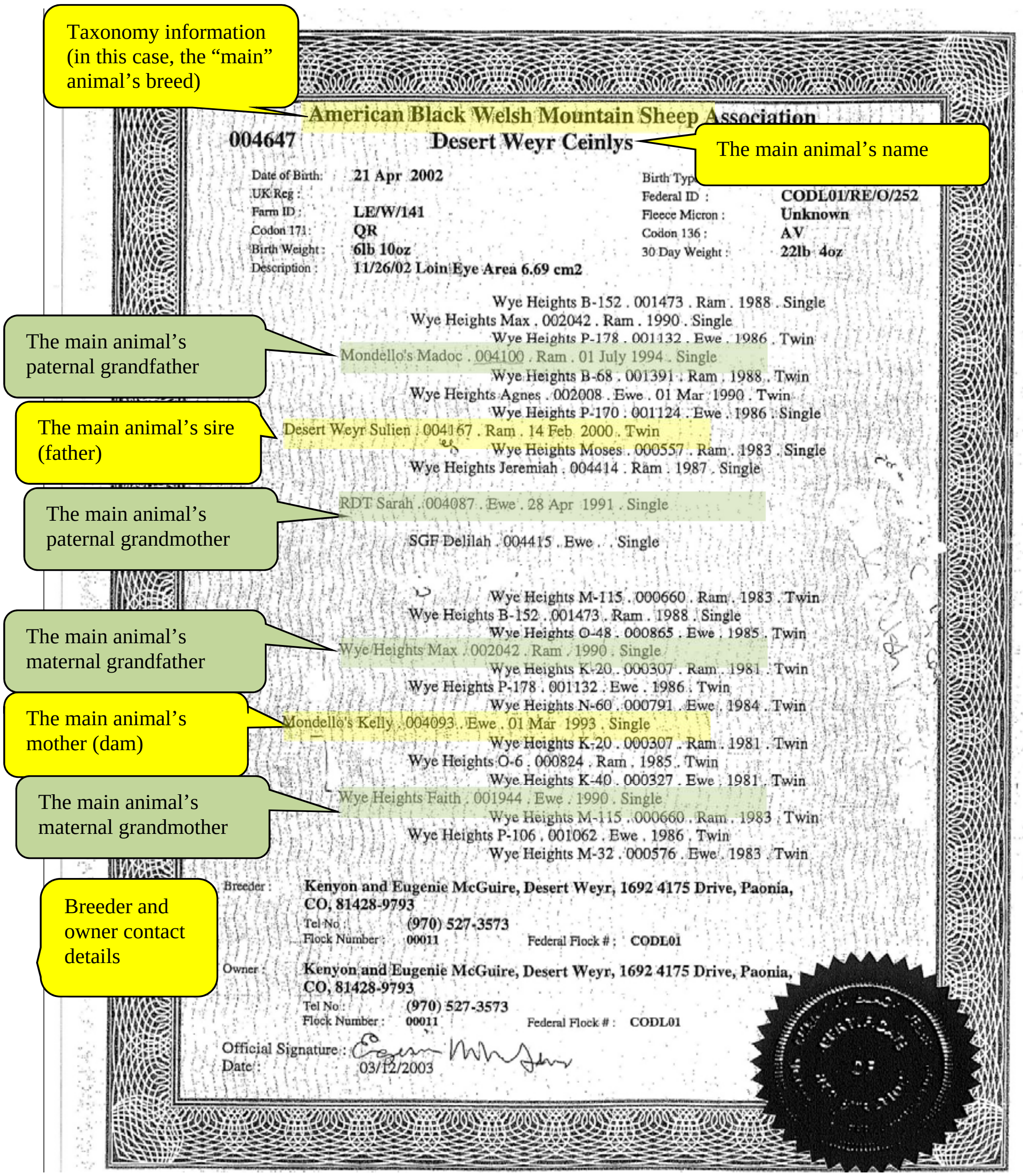
Tank 74　2 Cannisters

Subtotal $____
Freight $____
Return $____
Packing/LN2/Tank Rental $____
TOTAL $____

Hoffman A.I. Breeders, Inc., its agents, or employees cannot and do not guarantee the conception rate, quality, or productivity to be obtained in connection with the use of its products or recommended techniques. IT MAKES NO WARRANTIES WHAT SO EVER, EXPRESSED OR IMPLIED WHICH EXTEND BEYOND THE DESCRIPTION OF ITS PRODUCTS AND HEREBY DISCLAIMS ALL WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. In the unlikely event that any of our products shall be proven defective, damages resulting from their use shall be limited to their value.

UPS WILL RETURN FOR SHIPPER IN 2-3 DAYS
UPON ARRIVAL CHECK ORDER FOR ACCURACY OF SIRE AND # OF STRAWS

Figure 2. Sample Packing List for an Incoming Germplasm Order.



American Black Welsh Mountain Sheep Association

004647 Desert Weyr Ceinlys

Date of Birth: 21 Apr 2002
UK Reg :
Farm ID : LE/W/141
Codon 171: QR
Birth Weight : 6lb 10oz
Description : 11/26/02 Loin Eye Area 6.69 cm2

Federal ID : CODL01/RE/O/252
Fleece Micron : Unknown
Codon 136 : AV
30 Day Weight : 22lb 4oz

Wye Heights B-152 . 001473 . Ram . 1988 . Single
Wye Heights Max . 002042 . Ram . 1990 . Single
Wye Heights P-178 . 001132 . Ewe . 1986 . Twin
Mondello's Madoc . 004100 . Ram . 01 July 1994 . Single
Wye Heights B-68 . 001391 . Ram . 1988 . Twin
Wye Heights Agnes . 002008 . Ewe . 01 Mar 1990 . Twin
Wye Heights P-170 . 001124 . Ewe . 1986 . Single
Desert Weyr Sulien . 004167 . Ram . 14 Feb 2000 . Twin
Wye Heights Moses . 000557 . Ram . 1983 . Single
Wye Heights Jeremiah . 004414 . Ram . 1987 . Single
RDT Sarah . 004087 . Ewe . 28 Apr 1991 . Single
SGF Delilah . 004415 . Ewe . . Single
Wye Heights M-115 . 000660 . Ram . 1983 . Twin
Wye Heights B-152 . 001473 . Ram . 1988 . Single
Wye Heights O-48 . 000865 . Ewe . 1985 . Twin
Wye Heights Max . 002042 . Ram . 1990 . Single
Wye Heights K-20 . 000307 . Ram . 1981 . Twin
Wye Heights P-178 . 001132 . Ewe . 1986 . Twin
Wye Heights N-60 . 000791 . Ewe . 1984 . Twin
Mondello's Kelly . 004093 . Ewe . 01 Mar 1993 . Single
Wye Heights K-20 . 000307 . Ram . 1981 . Twin
Wye Heights O-6 . 000824 . Ram . 1985 . Twin
Wye Heights K-40 . 000327 . Ewe . 1981 . Twin
Wye Heights Faith . 001944 . Ewe . 1990 . Single
Wye Heights M-115 . 000660 . Ram . 1983 . Twin
Wye Heights P-106 . 001062 . Ewe . 1986 . Twin
Wye Heights M-32 . 000576 . Ewe . 1983 . Twin

Breeder : Kenyon and Eugenie McGuire, Desert Weyr, 1692 4175 Drive, Paonia, CO, 81428-9793
Tel No : (970) 527-3573
Flock Number : 00011 Federal Flock # : CODL01

Owner : Kenyon and Eugenie McGuire, Desert Weyr, 1692 4175 Drive, Paonia, CO, 81428-9793
Tel No : (970) 527-3573
Flock Number : 00011 Federal Flock # : CODL01

Official Signature :
Date : 03/12/2003

Figure 3. A Pedigree Document for an American Black Welsh Mountain Sheep named Desert Weyr Ceinlys.

The **Manage Taxonomy** use case allows the scientist to find or create an animal's taxonomy. An animal's taxonomy represents its scientific classification, and consists of multiple layers. For example, fish are classified according to their *strain*, which, in turn, belongs to a more general *species*, which, in turn, belongs to a more general *genus*. A rainbow trout might belong to the *House Creek* strain, which belongs to the *mykiss* species, which belongs to the *oncorchynchus* genus. In addition, a category might have a common name in addition to its scientific name. For example, "rainbow trout" is the common name for the *mykiss* species.

Table 1 shows the taxonomic structure for some of the animals in the NAGP repository.

Category Type:	Category Name:	Common Name(s):
Genus: Species: Breed: Line:	Bos taurus Hereford Prospector	Species Common Name: cattle
Genus: Species: Breed:	Bos taurus Angus	Species Common Name: cattle
Genus: Species: Breed:	Sus scrofa Hereford	Species Common Name: pig
Genus: Species: Breed:	Capra hircus Myotonic	Species Common Name: goat Breed Common Names: Fainting Goat, WoodenLeg
Genus: Species: Breed:	Ovis aries Rambouillet	Species Common Name: sheep
Genus: Species: Breed:	Ovis aries Black Welsh Mountain	Species Common Name: sheep
Genus: Species: Breed: Variety: Line:	Gallus gallus Plymouth Rock White Low 8 Week Body Weight	Species Common Name: chicken
Genus: Species: Variety:	Meleagris gallopavo Jersey Buff	Species Common Name: turkey
Genus: Species: Strain:	Salmonidae trutta Lestijoki	Species Common Name: brown trout

Table 1. Example Taxonomic Structures of Animals in the NAGP.

Not all animals have the same taxonomic structure, as shown in Table 1. The taxonomies vary in terms of the number of classification levels and the names of those levels. For example, while the taxonomy for fish consists of *genus*, *species*, and *strain*, the taxonomy for cattle consists of *genus*, *species*, *breed*, and (optionally) *line*. Some chickens are classified according to *genus*, *species*, *breed*, *variety*, and *line*. In addition, each level in a taxonomy may or may not have a common name (or names) associated with it. For example, the common name of the *taurus* species is cattle. If the animal's taxonomy does not exist in the database (e.g., because it is the first of a new genus or species or line of animal in the NAGP repository), then the scientist must create the appropriate hierarchical structure.

It is possible for an animal to be attached to a taxonomy at any level below the species level. For example, one cattle animal might belong to the Hereford breed, and thus belong to the taxonomy *Bos* → *taurus* → *Hereford*. Another cattle animal might belong to the Prospector line, and thus belong to the taxonomy *Bos* → *taurus* → *Hereford* → *Prospector*. The scientist attaches the animal to the lowest-level of the taxonomy possible, based on the information available.

For each level of a taxonomy structure, the database needs to store the category or level (e.g. Genus, Species, Breed, or Strain), the name of the taxonomy at that level (e.g. Bos, taurus, Hereford, or Lestijoki), the level's common name(s), if applicable, and the purpose(s) of the level (e.g. meat, milk, fiber, etc.), if applicable. The database also needs to store the relationship between different levels, so that it can generate the complete taxonomy structure for each animal.

Once the animal documentation is complete, the scientist returns to the packing slip to record information about the germplasm donations themselves (from the **Receive Order into Inventory** use case in Figure 1). An order contains one or more germplasm donations. Typically, one donation consists of a set of straws with semen from one animal—this is defined as one inventory item in AnimalGRIN. An order may contain donations from multiple animals. It is also common to receive germplasm from one animal in many separate orders over time. In any case, for each inventory item, the scientist needs to specify details such as the germplasm type (e.g., semen or blood), whether it is fresh or cryogenically frozen upon receipt, and the size and number of vessels containing the germplasm (e.g., five 1.5 fluid-ounce straws).

Finally, when the animal and order details are documented, the scientist assigns the germplasm donations items to specific locations in the repository. The **Assign Inventory Location** use case explains how the scientist finds available locations and allocates new inventory items to one or more locations in the NAGP physical repository. Locations are organized in a hierarchical structure, as shown in Figure 4 below. A donation is assigned to a visotube (the left-most picture in Figure 4), which is located within a goblet, which is located within a canister, which is located within a pie, which is located within a tank. The visotube is the smallest location, and inventory items are assigned at this level.

The system must record the entire inventory structure—from visotube to tank—assigned to each donation. Each tank is identified by a number which is unique across all tanks in a repository. Each pie within a tank is assigned a number between one and four, and each canister within a pie is also assigned a number. Each goblet within a canister is assigned a number-letter combination (e.g., 1A, 1B) and each visotube within a goblet is assigned a letter (A-L). While tank numbers are unique across all tanks in a repository, the smaller location units are assigned numbers or letters that are unique only within their parent location unit. Table 2 shows examples of location numbers and their corresponding meanings.

About twenty straws of germplasm may be stored in each visotube. Recall that one donation, or order line, consists of a set of straws filled with germplasm from one animal. One donation, then, may be allocated to one or more visotubes—one or more inventory locations—depending on the number of straws on the order line and space available in the visotube. Donations from multiple order lines may be stored in the same goblet—but not the same visotube—depending on the number of straws per order line and the space available in the goblet. Once the items are assigned to locations, the **Receive Order into Inventory** use case is successfully completed.

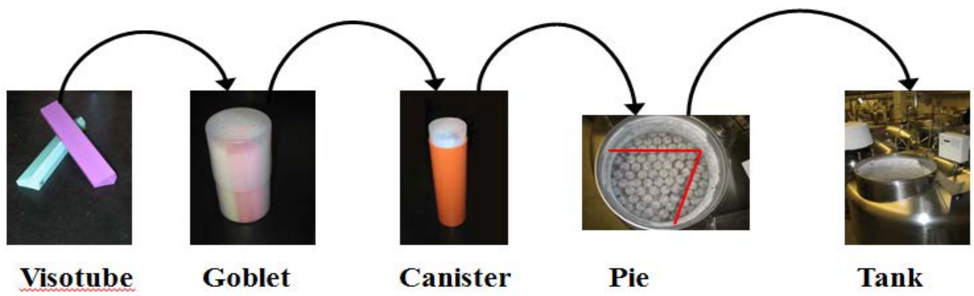


Figure 4. Inventory Location Structure

Location ID	Fully-Specified Location
1-2-1-1A-B	Tank #1 Pie #2 (within tank 1) Canister #1 (within pie 2) Goblet 1A (within canister 1) Visotube B (within goblet 1A)
5-2-3-1A-A	Tank #5 Pie #2 (within tank 5) Canister #3 (within pie 2) Goblet 1A (within canister 3) Visotube A (within goblet 1A)

Table 2. Example Inventory Locations.

3.2 Querying the AnimalGRIN Database.

In addition to entering data about incoming orders, AnimalGRIN users also need to retrieve data about orders and animals. For example, the inventory specialist responds to requests for information about orders by initiating the **Find Order** use case. This use case is expected to grow considerably over time, to capture various queries about existing orders. For example, the user may want to see summary information for a specific order, such as the number of germplasm straws on an order. Or, the user may need a list of those orders that have not yet been completely assigned to specific inventory locations in the system.

The repository manager also needs to retrieve information from AnimalGRIN. This information is typically summarized, such as the total number of orders were received in a given time period, or the total number of animals and breeds represented in the NAGP at any given time. The last use case in Figure 1, **Get Inventory Report**, explains how the manager interacts with AnimalGRIN to obtain some of this summary information.

Appendix 1 (available at http://jise.org/Volume23/23-1/Irwin-23-1Appendix.pdf) provides detailed use case descriptions (Cockburn 2001) to accompany Figure 1. The AnimalGRIN database must track all of the information needed to support these use cases.

4. PROJECT DELIVERABLES

This case may be used in several ways, depending on the learning objectives for the course and the project. We use the case as the basis for a semester-long (15-week), team-based project that covers conceptual data modeling, relational database design, database implementation, and query formulation. We use Microsoft SQL Server to support the implementation and query formulation portions of the course. Table 3 outlines each deliverable along with its corresponding course objective and completion time.

5. FINAL COMMENTS

We have used this case in multiple semesters and for both undergraduate and graduate database courses. Informal feedback from students indicates that the case is both challenging and rewarding, which is consistent with our observations. From a data modeling perspective, the case provides a novel domain to which students can apply their skills. The domain—managing animal germplasm donations and family trees—differs from the mainstream transaction processing database examples used in many textbooks. This difference allows us to illustrate the importance of understanding the user's domain, analyzing use cases and business rules, and creating an accurate conceptual data model. At the same time, we are able to help students see that even in very different domains, some of the same database design "patterns" apply. For example, in the AnimalGRIN case, germplasm donations are stored in visotubes, which are stored in goblets, which are stored in canisters, which are stored in refrigerated tanks. This may initially be complex and confusing to students, but we can help them see that it is analogous to a business situation where an employee has a supervisor, who in turn has a manager, who in turn has a senior manager, etc. Students then learn that these different examples can be modeled and designed in the same way, i.e., with a one-to-many recursive structure. The case provides ample opportunity to discuss modeling problems and solutions that occur (at an abstract level) across different domains and fosters students' analytical and critical thinking skills.

We have also found this case to be useful for linking database development to other courses. The functional requirements for AnimalGRIN are documented with a Use Case Diagram and Use Case Descriptions, so that students can work with systems analysis artifacts that they are likely to see on systems development projects, and in other courses in the information systems curricula. The user interface component to AnimalGRIN, which is outside the scope of our database course, can be directly related to a course on web application development. A systems analysis and design course could model the current business process of receiving donations into inventory—which still begins, oftentimes, with the physical shipment and manually-written packing slip—and consider technology-enabled business process improvements.

Course Objective	**Project Deliverable**	**Approximate Time**
Create a conceptual data model of the information requirements for an information system.	Conceptual Entity-Relationship Diagram (ERD) and corresponding data dictionary for the AnimalGRIN case.	3 weeks
Create a logical database design for an information system.	Translation of the Conceptual ERD into a relational schema that is in third normal form, with a corresponding data dictionary.	2 weeks
Build a relational database to implement a logical design with appropriate data integrity constraints.	Script of SQL statements to create tables with appropriate data integrity constraints.	1-2 weeks
Formulate simple SQL statements to populate a database with sample data.	Script of SQL insert statements to populate the tables from the previous deliverable with the sample data.	1 week
Formulate SQL statements to test data integrity constraints.	Script of INSERT, UPDATE, and DELETE statements to test various integrity constraints, and a completed testing matrix.	1 week
Formulate SQL queries (or SQL dynamic views) in response to "plain English" requests for information.	Script of the queries and/or views to support the **Find Order** and **Get Inventory Report** use cases.	2-3 weeks

Table 3. Project Deliverables.

For example, what if breeders were able to submit packing slips online, so that the AnimalGRIN would know what was coming and when it had been shipped and by whom? That would dramatically decrease the NAGP scientist's work, especially if completeness and error checking was provided with the online packing slips. However, breeders may be resistant to adopting and using this new feature, or increasing the workload on their end.

Finally, the case can be adapted to focus on a narrower but deeper set of database skills, as might be done for an advanced or graduate database source. For example, SQL skills could be emphasized by starting with the logical design or data definition script, and adding more complex query tasks and triggers for data integrity controls (e.g., triggers to verify that all fathers are male, all mothers are female, or that an animal's parents and grandparents are of the same breed). More modeling complexity can be added by relaxing some of our assumptions or adding new business rules. For example, students can consider how their designs would change if one animal has multiple donations (e.g., one donation of semen and another of blood) on the same order, or if there is more than one repository (e.g., a repository in Canada or in Brazil).

6. ENDNOTES

[i] Germplasm is the hereditary material found in the egg or sperm cell of an organism (www.Merriam-Webster.com). For animals in these preservation programs, germplasm is usually in the form of semen, blood, or embryos.

[ii] Taxonomy, in this case, refers to the scientific classification of animals, such as the genus, species, and line to which the animal belongs.

[iii] **Pedigree** refers to the family tree of an animal, which begins with the animal's sire (father) and dam (mother).

[iv] There are good sources for details on use case diagramming notation and constructs, for example, the Object Management Group's formal UML specification (OMG, 2011) or books such as Armour & Miller (2001).

7. REFERENCES

Armour, F. and Miller, G. (2001) *Advanced Use Case Modeling: Software Systems*. Addison-Wesley Object Technology Series.

Cappel, James J. and Gillman Jr., Jason R. (2011) "Teaching Case: Jay's Collectibles." *Journal of Information Systems Education,* 22(2), pp. 111-116.

Cockburn, Alistair (2001). *Writing Effective Use Cases*, Addison-Wesley, Upper Saddle River, NJ.

Food and Agriculture Organization (FAO). (2007) *The State of the World's Animal Genetic Resources for Food and Agriculture*. Rischkowsky, B. and Pilling, D. (eds.), Rome, Italy.

Fowler, Martin (1997). *Analysis Patterns: Reusable Object Models*, Addison-Wesley, Upper Saddle River, NJ.

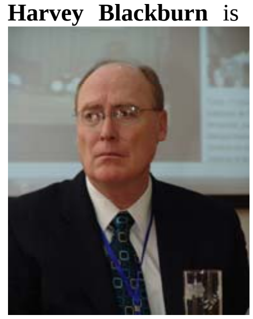
Object Management Group (OMG) (2011). *OMG Unified Modeling Language Superstructure, Version 2.4.1*. http://www.omg.org/spec/UML/2.4/Superstructure.

Parker, Kevin R. (2005) "Lost River Wind Riders: A Project for Teaching Database Design," *Communications of the Association for Information Systems*: Volume 16, Article 24. Available at: http://aisel.aisnet.org/cais/vol16/iss1/24.

AUTHOR BIOGRAPHIES

Gretchen Irwin is an associate professor in the Computer Information Systems Department at Colorado State University. Gretchen holds a PhD and a Masters of Science in Information Systems from the University of Colorado. Gretchen's primary research interest is in systems development, specifically in understanding how individuals learn and master tools, technologies, and approaches for systems analysis and design. Gretchen's research has appeared in the *Communications of the ACM*, the *Journal of MIS*, the *Journal of the AIS*, *IEEE Transactions on Professional Communication*, and *Human-Computer Interaction*.

Lark Wessel is an Associate Technical Project Manager at Elsevier in their HESI Assessment division. In her role on the product team, she acts as liason between the functional users and the technical team, and performs user acceptance testing of new development projects related to the online nursing exam products and support systems. Lark holds a Masters of Science in Computer Information Systems from Colorado State University. Her thesis project included the redesign of the database for the USDA Department of Agriculture National Animal Germplasm Program. Lark also holds a Bachelors of Business Administration in Management Information Systems from Ohio University.



Harvey Blackburn is the National Coordinator for the National Animal Germplasm Program, part of the United States Department of Agriculture's Agricultural Research Service. He has a PhD in Animal Breeding and Genetics and an MS in Animal Science. For the past decade, his research has involved all aspects of gene banking livestock genetic resources, which include: information systems, genetic structure of populations and cryobiology

STATEMENT OF PEER REVIEW INTEGRITY

All papers published in the Journal of Information Systems Education have undergone rigorous peer review. This includes an initial editor screening and double-blind refereeing by three or more expert referees.

ISSN 1055-3096