Day 1: Introduction to Database Theory and Design

Database Theory and Design
Tyler Peterson

International Summer School on Language Documentation and Description
Leiden University Centre for Linguistics, Leiden

November 26, 2011

My Details:

Tyler Peterson	
office:	LUCL Van Wijkplaats 4 Room 205a
telephone:	071-5272059
email:	t.r.g.peterson@hum.leidenuniv.nl trg.peterson@gmail.com (for Google docs)
office hours:	Most afternoons until 18:00

- Please fill out the short survey, and don't hesitate to contact me!

Goals for the Course:

- Databases: what they are, what linguists can use them for.

Goals for the Course:

- Databases: what they are, what linguists can use them for.
  - Databases vs. spreadsheets. Types of databases.

Goals for the Course:

- Databases: what they are, what linguists can use them for.
  - Databases vs. spreadsheets. Types of databases.
  - Different commercial and free database programs: advantages and limitations.

Goals for the Course:

- Databases: what they are, what linguists can use them for.
  - Databases vs. spreadsheets. Types of databases.
  - Different commercial and free database programs: advantages and limitations.
- To familiarize you with the concepts in database **theory** and **design**.

Goals for the Course:

- Databases: what they are, what linguists can use them for.
  - Databases vs. spreadsheets. Types of databases.
  - Different commercial and free database programs: advantages and limitations.
- To familiarize you with the concepts in database **theory** and **design**.
  - **Theory:** Terminology used in database theory; Understanding the design features of a database (entities and attributes).

Goals for the Course:

- Databases: what they are, what linguists can use them for.
  - Databases vs. spreadsheets. Types of databases.
  - Different commercial and free database programs: advantages and limitations.
- To familiarize you with the concepts in database **theory** and **design**.
  - **Theory:** Terminology used in database theory; Understanding the design features of a database (entities and attributes).
  - **Design:** Assessing your goals; planning a database on paper (Entity-Relationship diagrams); Best practices.

Goals for the Course:

- Databases: what they are, what linguists can use them for.
  - Databases vs. spreadsheets. Types of databases.
  - Different commercial and free database programs: advantages and limitations.
- To familiarize you with the concepts in database **theory** and **design**.
  - **Theory:** Terminology used in database theory; Understanding the design features of a database (entities and attributes).
  - **Design:** Assessing your goals; planning a database on paper (Entity-Relationship diagrams); Best practices.
- Looking at the practical implemetation of an analytical database in MS Access.

Goals for the Course:

- Databases: what they are, what linguists can use them for.
  - Databases vs. spreadsheets. Types of databases.
  - Different commercial and free database programs: advantages and limitations.
- To familiarize you with the concepts in database **theory** and **design**.
  - **Theory:** Terminology used in database theory; Understanding the design features of a database (entities and attributes).
  - **Design:** Assessing your goals; planning a database on paper (Entity-Relationship diagrams); Best practices.
- Looking at the practical implemetation of an analytical database in MS Access.
  - Cross tabulation: a front-line analytical tool.

Goals for the Course:

- Databases: what they are, what linguists can use them for.
  - Databases vs. spreadsheets. Types of databases.
  - Different commercial and free database programs: advantages and limitations.
- To familiarize you with the concepts in database **theory** and **design**.
  - **Theory:** Terminology used in database theory; Understanding the design features of a database (entities and attributes).
  - **Design:** Assessing your goals; planning a database on paper (Entity-Relationship diagrams); Best practices.
- Looking at the practical implemetation of an analytical database in MS Access.
  - Cross tabulation: a front-line analytical tool.
  - *Trigger–Target Database for Phonological Processes*

Goals for the Course:

- Databases: what they are, what linguists can use them for.
  - Databases vs. spreadsheets. Types of databases.
  - Different commercial and free database programs: advantages and limitations.
- To familiarize you with the concepts in database **theory** and **design**.
  - **Theory:** Terminology used in database theory; Understanding the design features of a database (entities and attributes).
  - **Design:** Assessing your goals; planning a database on paper (Entity-Relationship diagrams); Best practices.
- Looking at the practical implemetation of an analytical database in MS Access.
  - Cross tabulation: a front-line analytical tool.
  - *Trigger–Target Database for Phonological Processes*
  - *Programma de Fonologia Experimental e Histórica*

5 Day Plan:

- **Day 1:** Introduction to databases: the types, the purposes, the applications.

5 Day Plan:

- **Day 1:** Introduction to databases: the types, the purposes, the applications.
- **Day 2:** Principles of Database Theory and Design

5 Day Plan:

- **Day 1:** Introduction to databases: the types, the purposes, the applications.
- **Day 2:** Principles of Database Theory and Design
- **Day 3:** Relational Databases I

5 Day Plan:

- **Day 1:** Introduction to databases: the types, the purposes, the applications.
- **Day 2:** Principles of Database Theory and Design
- **Day 3:** Relational Databases I
- **Day 4:** Relational Databases II, examples and implementation

5 Day Plan:

- **Day 1:** Introduction to databases: the types, the purposes, the applications.
- **Day 2:** Principles of Database Theory and Design
- **Day 3:** Relational Databases I
- **Day 4:** Relational Databases II, examples and implementation
- **Day 5:** Implementation

Goals for Today:

The Database as a Concept and Tool

- Understanding what a database is
- Using databases in linguistics

The Database model and its Evolution

- The 'Flat' database
- The Database Management System (DBMS)
- Types of Databases Models

Database Applications

- Choosing the right Database Application
- Linguistic DBMS and Interfaces
- Non-Linguistic DBMS and Interfaces
- References and Suggested Readings

What is a database?

- **The database as a concept:** A structured collection of data, or *structured information:*

What is a database?

- **The database as a concept:** A structured collection of data, or *structured information:*
  - Index cards in a shoebox.

What is a database?

- **The database as a concept:** A structured collection of data, or *structured information:*
  - Index cards in a shoebox.
  - A table in a spreadsheet.

What is a database?

Word	Gloss	Gram.	Morph.
hon	'fish'	N	ROOT
smax	'bear, meat'	N	ROOT
algyax̲	'language'	N	ROOT
sm̓-algyax̲	Gitksan	N	STEM
sm̓-	'true'	A	PREFIX
siipxw	'sick, ill'	A	ROOT
wii-n̓akw	'tall'	A	STEM
wii-	'long'	A	PREFIX
n̓akw	DISTAL		ROOT
n̓akw	EVIDENTIAL		ROOT
x̲-	'consume'	V	PREFIX
iixwt	'fish'	V	ROOT
witxw	'arrive'	V	ROOT
bakw	'arrive'	V	ROOT
litsx̲xw	'read'	V	ROOT
=hl	common noun	Det.	ENCLITIC
=t	proper noun	Det.	ENCLITIC
=tip	plural noun	Det.	ENCLITIC
-y̓	1sg	Agr.	SUFFIX
-n	2sg	Agr.	SUFFIX
-t	3	Agr.	SUFFIX

Table: Structured Information: a Gitksan (Tsimshianic) word list

What is a database?

- The database as an application, or a kind of ‘processor’:

What is a database?

- The database as an application, or a kind of 'processor':
- Different types of processors:

What is a database?

- The database as an application, or a kind of 'processor':
- Different types of processors:
  - Word processor: processes words (!)

What is a database?

- The database as an application, or a kind of 'processor':
- Different types of processors:
  - Word processor: processes words (!)
  - Spreadsheet: processes financial, numerical and statistical information.

What is a database?

- The database as an application, or a kind of 'processor':
- Different types of processors:
  - Word processor: processes words (!)
  - Spreadsheet: processes financial, numerical and statistical information.
  - Database program: processes structured information.

What is a database?

- The database as an application, or a kind of 'processor':
- Different types of processors:
  - Word processor: processes words (!)
  - Spreadsheet: processes financial, numerical and statistical information.
  - Database program: processes structured information.
- The digital presentation of structured information through an application: MS Access; OpenOffice Calc; FileMaker Pro; MySQL with a PHP server; etc.

What is a database?

- Spreadsheets are actually a kind of database: both organize information into tables.

What is a database?

- Spreadsheets are actually a kind of database: both organize information into tables.
- The primary differences between a spreadsheet and database: different types of *queries*.

What is a database?

- Spreadsheets are actually a kind of database: both organize information into tables.
- The primary differences between a spreadsheet and database: different types of *queries*.
  - Spreadsheets use functions to ask questions of numbers. "What's the average daily rainfall for the first six months of this year?"

What is a database?

- Spreadsheets are actually a kind of database: both organize information into tables.
- The primary differences between a spreadsheet and database: different types of *queries*.
  - Spreadsheets use functions to ask questions of numbers. "What's the average daily rainfall for the first six months of this year?"
  - Databases uses functions to ask questions about structured information: "Do we have any books on designing databases in our library? If so, on which shelves are they located?"

What is a database?

- Spreadsheets are actually a kind of database: both organize information into tables.
- The primary differences between a spreadsheet and database: different types of *queries*.
  - Spreadsheets use functions to ask questions of numbers. "What's the average daily rainfall for the first six months of this year?"
  - Databases uses functions to ask questions about structured information: "Do we have any books on designing databases in our library? If so, on which shelves are they located?"
- Retrieval, and presentation: Today's database applications are designed to retrieve and present data through queries through specially designed forms, within a database application, or on the web.

Why use databases in linguistics?

- Linguistic research is a data-rich enterprise:

Why use databases in linguistics?

- Linguistic research is a data-rich enterprise:
  - Archiving massive amounts of language/linguistic data.

Why use databases in linguistics?

- Linguistic research is a data-rich enterprise:
  - Archiving massive amounts of language/linguistic data.
  - Lexicography and dictionary making.

Why use databases in linguistics?

- Linguistic research is a data-rich enterprise:
  - Archiving massive amounts of language/linguistic data.
  - Lexicography and dictionary making.
  - Enables collaboration through client-server applications over a network.

Why use databases in linguistics?

- Linguistic research is a data-rich enterprise:
  - Archiving massive amounts of language/linguistic data.
  - Lexicography and dictionary making.
  - Enables collaboration through client-server applications over a network.
- Database applications are particularily well-suited to linguistic research (cf. Nerbonne 1997; Everaert et al 2009):

Why use databases in linguistics?

- Linguistic research is a data-rich enterprise:
  - Archiving massive amounts of language/linguistic data.
  - Lexicography and dictionary making.
  - Enables collaboration through client-server applications over a network.
- Database applications are particularily well-suited to linguistic research (cf. Nerbonne 1997; Everaert et al 2009):
  - Cross-linguistic and typological research.

Why use databases in linguistics?

- Linguistic research is a data-rich enterprise:
  - Archiving massive amounts of language/linguistic data.
  - Lexicography and dictionary making.
  - Enables collaboration through client-server applications over a network.
- Database applications are particularily well-suited to linguistic research (cf. Nerbonne 1997; Everaert et al 2009):
  - Cross-linguistic and typological research.
  - Tools for verifying and evaluating contrasting empirical and theoretical claims.

Why use databases in linguistics?

- Linguistic research is a data-rich enterprise:
  - Archiving massive amounts of language/linguistic data.
  - Lexicography and dictionary making.
  - Enables collaboration through client-server applications over a network.
- Database applications are particularily well-suited to linguistic research (cf. Nerbonne 1997; Everaert et al 2009):
  - Cross-linguistic and typological research.
  - Tools for verifying and evaluating contrasting empirical and theoretical claims.
  - Specialized queries that can yield new insights into data.

Why use databases in linguistics?

- Linguistic research is a data-rich enterprise:
  - Archiving massive amounts of language/linguistic data.
  - Lexicography and dictionary making.
  - Enables collaboration through client-server applications over a network.
- Database applications are particularily well-suited to linguistic research (cf. Nerbonne 1997; Everaert et al 2009):
  - Cross-linguistic and typological research.
  - Tools for verifying and evaluating contrasting empirical and theoretical claims.
  - Specialized queries that can yield new insights into data.
- Consistency and integrity: imposing a structure on information can help reduce inaccuracies and redundancies.

What can databases be used for in linguistics?

- Two broad types of databases in linguistics:

What can databases be used for in linguistics?

- Two broad types of databases in linguistics:
  - A **Linguistic database:** contains data from language research (i.e. words, phonemes, grammatical categories, fundamental frequencies, etc.)

What can databases be used for in linguistics?

- Two broad types of databases in linguistics:
  - A **Linguistic database:** contains data from language research (i.e. words, phonemes, grammatical categories, fundamental frequencies, etc.)
  - A **Metalinguistic database:** contains data about language research (i.e. names of speakers, locations, recording details, etc.)

What can databases be used for in linguistics?

- Two broad types of databases in linguistics:
  - A **Linguistic database:** contains data from language research (i.e. words, phonemes, grammatical categories, fundamental frequencies, etc.)
  - A **Metalinguistic database:** contains data about language research (i.e. names of speakers, locations, recording details, etc.)
- Both are concieved, designed and implemented using the same principles.

A common starting point: the 'Flat' database

Word	Gloss	Gram.	Morph.
hon	'fish'	N	ROOT
smax	'bear, meat'	N	ROOT
algyax̲	'language'	N	ROOT
sm̓-algyax̲	Gitksan	N	STEM
sm̓-	'true'	A	PREFIX
siipxw	'sick, ill'	A	ROOT
wii-n̓akw	'tall'	A	STEM
wii-	'long'	A	PREFIX
n̓akw	DISTAL		ROOT
n̓akw	EVIDENTIAL		ROOT
x̲-	'consume'	V	PREFIX
iixwt	'fish'	V	ROOT
witxw	'arrive'	V	ROOT
bakw	'arrive'	V	ROOT
litsx̲xw	'read'	V	ROOT
=hl	common noun	Det.	ENCLITIC
=t	proper noun	Det.	ENCLITIC
=tip	plural noun	Det.	ENCLITIC
-y̓	1sg	Agr.	SUFFIX
-n	2sg	Agr.	SUFFIX
-t	3	Agr.	SUFFIX

Table: A 'Flat' Database of a Gitksan (Tsimshianic) word list

A common starting point: the 'Flat' database

- Language data in field notes, a numbered arrangement; Possibly transferred onto cards.

A common starting point: the 'Flat' database

- Language data in field notes, a numbered arrangement; Possibly transferred onto cards.
- Enter language data into a word processor (MS Word) or spreadsheet (MS Excel).

A common starting point: the 'Flat' database

- Language data in field notes, a numbered arrangement; Possibly transferred onto cards.
- Enter language data into a word processor (MS Word) or spreadsheet (MS Excel).
- One record in a paper form = One row ("record") in computerized table of data.

A common starting point: the 'Flat' database

- Language data in field notes, a numbered arrangement; Possibly transferred onto cards.
- Enter language data into a word processor (MS Word) or spreadsheet (MS Excel).
- One record in a paper form = One row ("record") in computerized table of data.
- Adequate for a simple applications with not a lot of data or features (i.e. categories).

A common starting point: the 'Flat' database

- Language data in field notes, a numbered arrangement; Possibly transferred onto cards.
- Enter language data into a word processor (MS Word) or spreadsheet (MS Excel).
- One record in a paper form = One row ("record") in computerized table of data.
- Adequate for a simple applications with not a lot of data or features (i.e. categories).
  - Generating word lists.
  - Basic searches.

A common starting point: the 'Flat' database

- Language data in field notes, a numbered arrangement; Possibly transferred onto cards.
- Enter language data into a word processor (MS Word) or spreadsheet (MS Excel).
- One record in a paper form = One row ("record") in computerized table of data.
- Adequate for a simple applications with not a lot of data or features (i.e. categories).
  - Generating word lists.
  - Basic searches.
- A **'flat'** database.

Limitations of a Flat database

- You find you need more out of your data:

Limitations of a Flat database

- You find you need more out of your data:
  - Inflexible.
  - Difficult to expand.

Limitations of a Flat database

- You find you need more out of your data:
  - Inflexible.
  - Difficult to expand.
- Many redundant data entries

Limitations of a Flat database

- You find you need more out of your data:
  - Inflexible.
  - Difficult to expand.
- Many redundant data entries
  - Identifying and eliminating incorrect entries.
  - Inconsistency.
  - Unmanageable file size (difficult to transfer), and potential memory problems.

Limitations of a Flat database

- You find you need more out of your data:
  - Inflexible.
  - Difficult to expand.
- Many redundant data entries
  - Identifying and eliminating incorrect entries.
  - Inconsistency.
  - Unmanageable file size (difficult to transfer), and potential memory problems.
- Can become overwhelming complex, and unstable along with the burden of maintaining the database.

Limitations of a Flat database

- You find you need more out of your data:
  - Inflexible.
  - Difficult to expand.
- Many redundant data entries
  - Identifying and eliminating incorrect entries.
  - Inconsistency.
  - Unmanageable file size (difficult to transfer), and potential memory problems.
- Can become overwhelming complex, and unstable along with the burden of maintaining the database.
- **For language data: can obscure potentially meaningful implications, relationships and generalizations.**

Limitations of a Flat database cont.

Word	Gloss	Gram.	Morph.
hon	'fish'	N	ROOT
smax	'bear, meat'	N	ROOT
algyax̲	'language'	N	ROOT
sm̓-algyax̲	Gitksan	N	STEM
sm̓-	'true'	A	PREFIX
siipxw	'sick, ill'	A	ROOT
wii-n̓akw	'tall'	A	STEM
wii-	'long'	A	PREFIX
n̓akw	DISTAL		ROOT
n̓akw	EVIDENTIAL		ROOT
x̲-	'consume'	V	PREFIX
iixwt	'fish'	V	ROOT
witxw	'arrive'	V	ROOT
bakw	'arrive'	V	ROOT
litsx̲xw	'read'	V	ROOT
=hl	common noun	Det.	ENCLITIC
=t	proper noun	Det.	ENCLITIC
=tip	plural noun	Det.	ENCLITIC
-y̓	1sg	Agr.	SUFFIX
-n	2sg	Agr.	SUFFIX
-t	3	Agr.	SUFFIX

Table: A 'Flat' Database of a Gitksan (Tsimshianic) word list

The Solution:

- Separate the flat database into two interacting systems:

The Solution:

- Separate the flat database into two interacting systems:

I. Database Management System (DBMS)

The Solution:

- Separate the flat database into two interacting systems:

I. Database Management System (DBMS)

II. An application to interact with the DBMS.

I. Database Management System (DBMS)

- Keeps data in small, unique chunks

I. Database Management System (DBMS)

- Keeps data in small, unique chunks
  - Efficient storage
  - Maintains 'just enough' redundancy

I. Database Management System (DBMS)

- Keeps data in small, unique chunks
  - Efficient storage
  - Maintains 'just enough' redundancy
- Principle focus: handling data

I. Database Management System (DBMS)

- Keeps data in small, unique chunks
  - Efficient storage
  - Maintains 'just enough' redundancy
- Principle focus: handling data
  - Handles physical details of storing data efficiently
  - Delivers & manipulates data for applications
  - Security and stability

I. Database Management System (DBMS)

- Keeps data in small, unique chunks
  - Efficient storage
  - Maintains 'just enough' redundancy
- Principle focus: handling data
  - Handles physical details of storing data efficiently
  - Delivers & manipulates data for applications
  - Security and stability
- Several "industrial-strength" DBMS:

I. Database Management System (DBMS)

- Keeps data in small, unique chunks
  - Efficient storage
  - Maintains 'just enough' redundancy
- Principle focus: handling data
  - Handles physical details of storing data efficiently
  - Delivers & manipulates data for applications
  - Security and stability
- Several "industrial-strength" DBMS:
  - Oracle
  - Microsoft SQL Server

II. Applications that interact with the DBMS

- A program to retrieve data from a DBMS:

II. Applications that interact with the DBMS

- A program to retrieve data from a DBMS:
  - The DBMS stores data and responds to queries – we don't interact with it directly.

II. Applications that interact with the DBMS

- A program to retrieve data from a DBMS:
  - The DBMS stores data and responds to queries – we don't interact with it directly.
  - DBMSs are used with a "client" application: MS Access, FileMaker Pro etc. These create a graphical user interface to interact with the data through *forms* and *reports*.

II. Applications that interact with the DBMS

- A program to retrieve data from a DBMS:
  - The DBMS stores data and responds to queries – we don't interact with it directly.
  - DBMSs are used with a "client" application: MS Access, FileMaker Pro etc. These create a graphical user interface to interact with the data through *forms* and *reports*.
- A language to query data from a DBMS:

II. Applications that interact with the DBMS

- A program to retrieve data from a DBMS:
  - The DBMS stores data and responds to queries – we don't interact with it directly.
  - DBMSs are used with a "client" application: MS Access, FileMaker Pro etc. These create a graphical user interface to interact with the data through *forms* and *reports*.
- A language to query data from a DBMS:
  - *Structured Query Language* (SQL): a standardized language that uses user-defined functions to query the data.

II. Applications that interact with the DBMS

- A program to retrieve data from a DBMS:
  - The DBMS stores data and responds to queries – we don't interact with it directly.
  - DBMSs are used with a "client" application: MS Access, FileMaker Pro etc. These create a graphical user interface to interact with the data through *forms* and *reports*.
- A language to query data from a DBMS:
  - *Structured Query Language* (SQL): a standardized language that uses user-defined functions to query the data.
  - Generates reports in form of a table or *pivot table*.

Four types of database models:

- A Flat database
  - Made of a single table, or "file".
  - Each row corresponds to some object (e.g., a language) being described, and each column represents a property (*attribute*), such as name, location, or word order etc..

Four types of database models:

- A Flat database
  - Made of a single table, or "file".
  - Each row corresponds to some object (e.g., a language) being described, and each column represents a property (*attribute*), such as name, location, or word order etc..
- A Relational database
  - Consists of several tables (*relations*) linked to each other.

Four types of database models:

- A Flat database
  - Made of a single table, or "file".
  - Each row corresponds to some object (e.g., a language) being described, and each column represents a property (*attribute*), such as name, location, or word order etc..
- A Relational database
  - Consists of several tables (*relations*) linked to each other.
- A Hierarchical database
  - Not as a table but as a tree structure, similar to folders and subfolders in an operating system: each unit "belongs" to some larger unit, and contains smaller units.

Four types of database models:

- A Flat database
  - Made of a single table, or "file".
  - Each row corresponds to some object (e.g., a language) being described, and each column represents a property (*attribute*), such as name, location, or word order etc..
- A Relational database
  - Consists of several tables (*relations*) linked to each other.
- A Hierarchical database
  - Not as a table but as a tree structure, similar to folders and subfolders in an operating system: each unit "belongs" to some larger unit, and contains smaller units.
- An Object-Oriented database database
  - Data are modeled as *objects* of various types that share or inherit properties according to their type
  - For example, a database about word classes could let objects of the type *transitive verb* inherit properties of the type *verb*.

Two types of database applications:

- **Stand-alone desktop databases:** MS Access

Two types of database applications:

- **Stand-alone desktop databases:** MS Access
- **The network database:** WordNet

Stand-alone desktop databases

- Suitable for the one-person research project.

Stand-alone desktop databases

- Suitable for the one-person research project.
- A stand-alone software with a graphical user interface for both the database configuration, and to create forms and queries.

Stand-alone desktop databases

- Suitable for the one-person research project.
- A stand-alone software with a graphical user interface for both the database configuration, and to create forms and queries.
- Many tasks are automated; customizable templates.

Stand-alone desktop databases

- Suitable for the one-person research project.
- A stand-alone software with a graphical user interface for both the database configuration, and to create forms and queries.
- Many tasks are automated; customizable templates.
- Everything fits in one file or folder, and can be backed up, sent by email, etc.

Stand-alone desktop databases

- Suitable for the one-person research project.
- A stand-alone software with a graphical user interface for both the database configuration, and to create forms and queries.
- Many tasks are automated; customizable templates.
- Everything fits in one file or folder, and can be backed up, sent by email, etc.
- The ony requirement is a desktop computer with the database application; software is easy to install or already present, and it is not necessary to set up a server.

Stand-alone desktop databases

- Suitable for the one-person research project.
- A stand-alone software with a graphical user interface for both the database configuration, and to create forms and queries.
- Many tasks are automated; customizable templates.
- Everything fits in one file or folder, and can be backed up, sent by email, etc.
- The ony requirement is a desktop computer with the database application; software is easy to install or already present, and it is not necessary to set up a server.
- Internet collaboration possible but not required.

Stand-alone desktop databases

- Suitable for the one-person research project.
- A stand-alone software with a graphical user interface for both the database configuration, and to create forms and queries.
- Many tasks are automated; customizable templates.
- Everything fits in one file or folder, and can be backed up, sent by email, etc.
- The ony requirement is a desktop computer with the database application; software is easy to install or already present, and it is not necessary to set up a server.
- Internet collaboration possible but not required.
- MS Access, FileMaker Pro, OpenOffice Calc.

Network databases

- Ideal when multiple people must collaborate on data entry.

Network databases

- Ideal when multiple people must collaborate on data entry.
- A modular system of three parts:

Network databases

- Ideal when multiple people must collaborate on data entry.
- A modular system of three parts:
  - A web-based interface (i.e. a web browser)
  - A server. (running PHP to manage the queries and generate the web pages)
  - The database. (MySQL)

Network databases

- Ideal when multiple people must collaborate on data entry.
- A modular system of three parts:
  - A web-based interface (i.e. a web browser)
  - A server. (running PHP to manage the queries and generate the web pages)
  - The database. (MySQL)
- Most of the same functions with stand-alone databases can be used in network databases.

Comparing Pros and Cons

- Stand-alone databases
  - **Pros:** Can be implemented quickly and easily.
  - **Cons:** Can be expensive and proprietary.

Comparing Pros and Cons

- Stand-alone databases
  - **Pros:** Can be implemented quickly and easily.
  - **Cons:** Can be expensive and proprietary.
- Network databases
  - **Pros:** Free, with more or less the same functionality as a stand-alone, proprietary database.
  - **Cons:** Extensive computer knowledge required (i.e. setting up a server, making the connections, knowledge of HTML)

Criteria

- General:
  - Who produced the software, which platforms the software runs on? Is other software needed?
  - Is it easy to use? Is it well-supported/documented? Cost?

Criteria

- General:
  - Who produced the software, which platforms the software runs on? Is other software needed?
  - Is it easy to use? Is it well-supported/documented? Cost?
- Technical:
  - Ability to import and export data (i.e. text, XML files).
  - Are the pre-defined and/or user-defined options helpful? Can they be easily modified?
  - Is the application scalable?
  - Is it relational?

Criteria

- General:
  - Who produced the software, which platforms the software runs on? Is other software needed?
  - Is it easy to use? Is it well-supported/documented? Cost?
- Technical:
  - Ability to import and export data (i.e. text, XML files).
  - Are the pre-defined and/or user-defined options helpful? Can they be easily modified?
  - Is the application scalable?
  - Is it relational?
- Linguistic:
  - Unicode compatibility, special character input methods, and the ease of character input.
  - Ability to handle texts and texts, interlinearized material.
  - Allows you to follow the best practices for archiving linguistic data (i.e. XML, E-MELD emeld.org).

Databases designed for linguistics

- Stand-alone: SIL Shoebox 5.0 with Toolbox 1.2
  - Runs on both Windows and Mac. Proprietary, but not too expensive.
  - Not very well supported, problems exporting XML files.
  - A native environment for text interlinearization and analysis.
  - **Uses filter-type searches, not structured queries.**

The Players

- Stand-alone, relational databases:

The Players

- Stand-alone, relational databases:
  - MS Access: powerful and customizable form and query tools. Proprietary and not cheap.

The Players

- Stand-alone, relational databases:
  - MS Access: powerful and customizable form and query tools. Proprietary and not cheap.
  - FileMaker Pro: also with customizable form and query tools. Proprietary and not cheap.

The Players

- Stand-alone, relational databases:
  - MS Access: powerful and customizable form and query tools. Proprietary and not cheap.
  - FileMaker Pro: also with customizable form and query tools. Proprietary and not cheap.
  - OpenOffice Calc: less features than Access or FileMaker, but has the same core functionality. Open source (free), but somewhat unstable.

The Players

- Stand-alone, relational databases:
  - MS Access: powerful and customizable form and query tools. Proprietary and not cheap.
  - FileMaker Pro: also with customizable form and query tools. Proprietary and not cheap.
  - OpenOffice Calc: less features than Access or FileMaker, but has the same core functionality. Open source (free), but somewhat unstable.
- Network: MySQL (http://www.mysql.com/); Apache server with PHP; Google Chrome – all free.

The Players

- Access, FileMaker Pro, and Calc are all suitable databases for linguistic analysis: they are all **relational** and can handle **SQL queries**.

The Players

- Access, FileMaker Pro, and Calc are all suitable databases for linguistic analysis: they are all **relational** and can handle **SQL queries**.
- Some more pros/cons and comparisons:

The Players

- Access, FileMaker Pro, and Calc are all suitable databases for linguistic analysis: they are all **relational** and can handle **SQL queries**.
- Some more pros/cons and comparisons:
  - Access, FileMaker Pro, and Calc do not handle texts well.

The Players

- Access, FileMaker Pro, and Calc are all suitable databases for linguistic analysis: they are all **relational** and can handle **SQL queries**.
- Some more pros/cons and comparisons:
  - Access, FileMaker Pro, and Calc do not handle texts well.
  - Access is more constrained than FileMaker or Calc – less possibility for introducing errors or inconsistencies.

The Players

- Access, FileMaker Pro, and Calc are all suitable databases for linguistic analysis: they are all **relational** and can handle **SQL queries**.
- Some more pros/cons and comparisons:
  - Access, FileMaker Pro, and Calc do not handle texts well.
  - Access is more constrained than FileMaker or Calc – less possibility for introducing errors or inconsistencies.
  - FileMaker and Calc are suited to smaller, less-complex projects.

The Players

- Access, FileMaker Pro, and Calc are all suitable databases for linguistic analysis: they are all **relational** and can handle **SQL queries**.
- Some more pros/cons and comparisons:
  - Access, FileMaker Pro, and Calc do not handle texts well.
  - Access is more constrained than FileMaker or Calc – less possibility for introducing errors or inconsistencies.
  - FileMaker and Calc are suited to smaller, less-complex projects.
  - All are XML compatible and network ready.

- There are countless resources on the web on database design, theory, and implementation.
- Specific references on linguistic databases:
  - Ferrara, M. & Moran, S. 2004. Review of DBMS for Linguistic Purposes. *Proceedings of E-MELD 2004*. Online publication, at http://www.linguistlist.org/emeld/workshop/2004/proceedings.html.
  - Nerbonne, John. 1998. *Linguistic Databases*, CSLI, Stanford.
  - Everaert, Musgrave, Dimitriadis (eds) 2009. The Use of Databases in Cross-Linguistic Studies. *Empirical Approaches to Language Typology (EALT) 41*. Mouton de Gruyter.