All programs are data, and algorithms that operate on data

* Computer stores data as 1 and 0s
* Binary Files and all programs we run are secretly 0's and 1's, all instructions and structs and arrays (Show hexdump)

Creating Data Abstractions

We should create abstractions for benefits such as…

* Greater semantic meaning
* Memory efficiency
* Lookup/other operation efficiency
* Persistant file storage
* Archiving of files

Note: This is a non academic approach and I’m sure I missed something

This requires us to create a standard, and a parser/marshaler to make it real.

**Note: While files and program data are conceptually the same (just bytes) - You will tend to need a different spec for files should you want to store more than 1 data type**

* Example: Write an array to disk
* Example 1: Create a struct of a person, and Write a marshaller to convert to a file

<https://www.sqlite.org/fileformat.html>

Examples:

* Semantics: Character, Integer, Float, Boolean, Struct
* Memory Efficiency: Zip (Compression)
  + Reduce redundancy
* Lookup and other operation efficiency: Hashmap, Tree, Graph, Linked List
  + Utilize references (Pointers), data formatting and algorithms
* Usage in programs or in files
  + Data tends to need to be stored a little differently in files, even though it's still bytes
* Archiving: Gzip
  + Combination of many files into 1 (Then even better compression)

Storing single data types: How to

Storing Multiple Data types: How to

* Spec for what goes where
* OR label each data or section of data something that states

Pointers

Lets add the final key to data power: Pointers

Pointers for relational data

Lookup and operation speed: Data Structures

Example: Create a linked list as files!

How might you implement Graphs or Trees in file format:

<https://github.com/thinkaurelius/faunus/wiki/GraphSON-Format>

<https://en.wikipedia.org/wiki/Newick_format>

\* The data formats are quite extensive, so I would suggest reading more into this.

Storing and modeling lots of different data

* How can we store multiple data types in a file?
* Tend to want some sort of delimiter or schema

THERE are a lot of ways to model data>>> Some common ones are tabular data and key-value data.

2d Array, hashmap + array

CSV Files + JSON Files

relationships to other data:

Same ideas as data structures: Create pointers

Examples

* JSON or CSV with references to data within or outside
* Data structures
* eg. Java Class file format

=====================================================

Understand translating between file and program data

Create parsers

Ways of structuring data and why (Semantic meaning, Compression, relational data, tabular, graph, lookup speed)

How pointers helps in structuring data

* Plain Text Files (ASCII): Super important for convenience
  + But not as fast or memory efficient
  + We tend to see data formats built off of