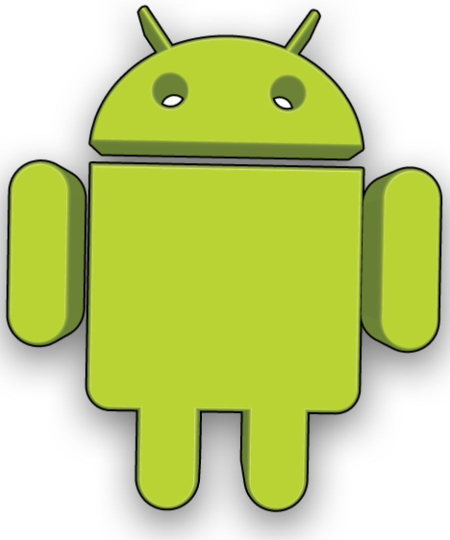
# Regis University

CS493 Senior Capstone

## Scholarship Paper



Intructor: Mr. Allan Rossi

Author: Mr. Jason N Mansfield

# The Object Based Study Application

August 18, 2013

# abstract

The DataFace project is a Android application trying to become a Cheat Sheet or Flow Chart of sorts. This essay describes my attempt to create a application which provides a method for an individual to naturally memorize information using a technique similar to chunking.

# Contents

**August 18, 2013** **6**

# August 18, 2013

## 1 DataFace a Chunking Application for memory

To some individuals memorizing trivial facts comes naturally. Unfortunately, for myself I have the tendency to lose any information I do not deem critical. I have arrived at the conclusion that the only way I can remember any trivial subject is by somehow organizing it in a way that gives it a purpose in my life. One particular approach which caught my attention is called Chunking (Bor, 2012). An example of chunking might be the method you use to remember phone numbers. Memorizing telephone numbers is easy because you already use chucking as an approach:

| Calling code | Area Code | Central Office Code | Subscriber Number |
|---|---|---|---|
| 1 | 757 | 555 | 1234 |

**Figure 1:** *telephony*

In the past few months I have been using various Flash Card programs, such as Anki, for memorizing trivial data. The problem I have been experiencing is correlating items together. For example, I might be studying a piece of history in which a significant event happens in 1851; seven cards later an additional event happens in 1951. It was difficult for me to divide the two event when attempting to recall. However, in a cheat sheet, time line or other method all the dates are in front of you:

event one
1851
event two
1951

**Figure 2:** *flashcards*

Therefore, in an attempt to implement a chunking type approach, I have determined organizing objects together in a viewable format, is the best approach. This will allow the user to not only view the information, but all the related information to it. This approach will create an overall map of information, which allows the learner to connect and reinforce data objects together. If a learner cannot recall an event which happened in 1851, they may in fact remember that an event prior which happened in 1842, and an event that after happened in 1862. This grouping together will allow the learner to use surrounding data to isolate a correct answer.

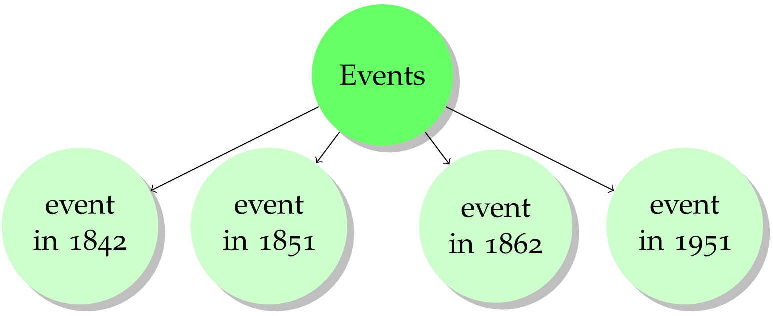


**Figure 3:** *objects*

## 2 Connectionism: neural networks

To be perfectly honest when I began this project I had no backing information to believe my object based approach was relevant. However, as I began to explore I discovered that I'm not the only one to hold this view. Similar to an Object based approach Connectionism theorists believe the brains neural network uses a large number of units which are joined together (Garson, 2012).

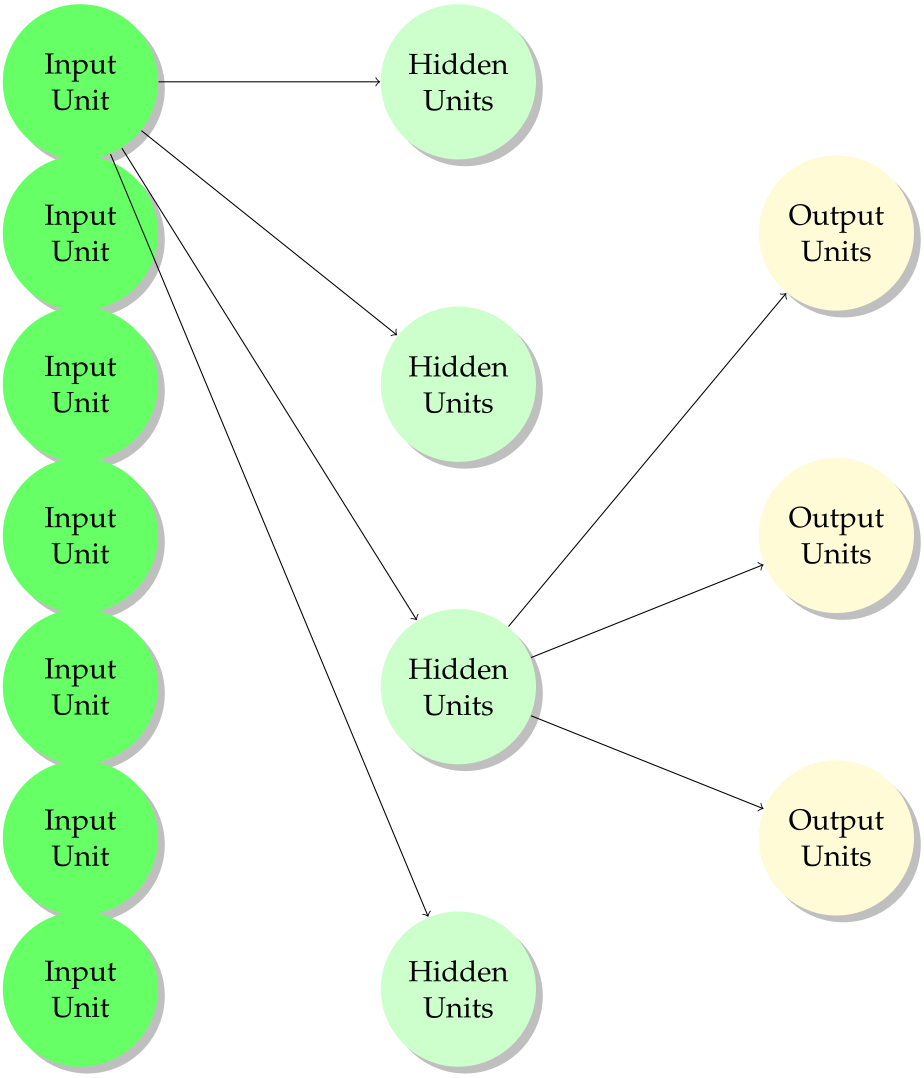


**Figure 4:** *simple neural net*

The object based approach is of course not nearly as intricate as the brains supposed neural network, but I do believe this example drives home the point of using connections. Unlike, episodic memory which is obtained by personal or experiential memory; semantic memory is conceptual and based on learned facts (Sutton, 2012). The key is to create enough connections with factual data objects to pass the information from semantic to declarative memory for future recall.

## 3 Creating Objects

Using a method similar to chunking the DataFace application strives to use small pieces of larger information. The DataFace format allows the creation of a parent Object which can then be populated with child items:

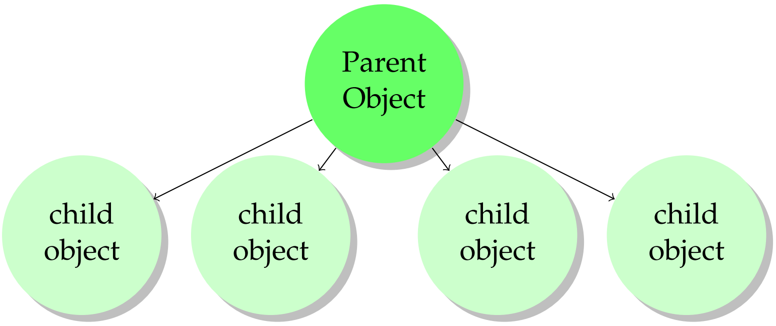


**Figure 5:** *Parent and child objects*

The idea behind this object based application is similar to a flow chart in that the child items can all be seen simultaneously. Additionally, there should be a way to also correlate all the parent objects as well. I selected a Android based platform due to it's open platform allowing me the ability to freely share the final product. Using the Android platform also allows outside users an easy way to use and share the product, if not contribute to its development. Future development should include enhancing the graphical view of the child objects as previously described, add a mathematical markup language such as LaTeX or MathML, and the ability to input parsed data through text documents in formats such as CSV or XML. The first steps of creating a SQLite (*Saving Data in SQL Databases* 2013) database which can manage both Parent and Child objects with a view has been accomplished. However, I am far from satisfied with the child items in a ListView (*ListView* 2013) type format. Once I can create a Layout and View which allows all the child items to be viewed at once, possibly a modified GridView (*GridView* 2013), I feel this particular application will have usefulness.

## 4 The current state of the DataFace application

Although, the DataFace application has not reached a mature level, the structure seems to be intact. The parent objects allow for a Title and a short description. The child items are identical with the exception that they are encapsulated within the parent object. I was able to research and implement the following items into my application:

**Activities** I discover how to communicate between activities using intents (*Intent* 2013) and bundles (*Bundle* 2013).

**SimpleCursorAdapter** The SimpleCursorAdapter (*SimpleCursorAdapter* 2013) was one adapter that I used during development, and is the remaining adapter now due to its ability to take SQLite column data and create a proper ListView.

**Constructors** DataFace contains two constructors (Tamada, 2012), one for the Parent object and One for the Child Objects.

**SQLite3** SQLite3 matched with Android makes an excellent environment for development. I have created web applications before, but creating this Java based application using sqlite3 (Vogel, 2013) on Android was by far the most rewarding experience.

## 5 The current DataFace Source Code

The education I have received at Regis University has certainly opened my eyes to new possibilities. Through classes such as CS375 Advanced Programming and Algorithms, and MT320 Introduction to Discrete Mathematics; I have learned to have a unprejudiced view towards certain programing languages, but instead concern myself with the underlying logic. My favorite classes MT360A and MT360B Calculus I and II, prepared me for the discipline I needed to take a real world problem and break it down into a solution. Clearly, the knowledge I gained taking CS442 Database programming was helpful during the development of DataFace with its use of SQLite3. Although, CS434 Object Oriented Programming using Java was an excellent class, I still had to do some research to recall some basic Object Oriented methods. In retrospect I wish I had spent less time on creating a Fragment view and simply made a orginial effort to understand the basics of an Activity. I began the project believing the implementation of a Fragment (*Building a Dynamic UI with Fragments* 2013) was critical. Overall, I was able to accomplish what I thought could be done within the time alloted. I'm fairly certain there are still quite a few beginner development flaws within the application which will require re-writing over time. I currently have two sqlite databases being used due to a semi-stable delete function. Eventually, there will likely only be a single database with two tables. The more I learn about managing the sqlite database using the Android API the better my application will become. Additionally, as previously mentioned a GridView might just be the trick needed for the child views, but only time and future efforts will make that determination. This was a tremendously satisfying project for me and I am excited to see what I can accomplish in the future with the knowledge handed to me by my Instructors at Regis. I am truly grateful for teaching I have recieved at Regis University.

# Appendix

To view the current DataFace project you may download the latest APK to use on your Android based device: Dataface-release-unsigned.apk

To view the latest development source code of this product, please take a look at the online GitHub repository: DataFaceProject

To clone this project using git:
git clone https://github.com/trentonknight/DataFaceProject.git

To comment or see status updates on this project view the wiki here: DataFace: An attempt to create a Object based learning tool.

# Bibliography

Bor, Daniel (2012). *The ravenous brain : how the new science of consciousness explains our insatiable search for meaning*. New York: Basic Books. ISBN: 046502047X.

*Building a Dynamic UI with Fragments* (2013). Android Open Source Project. URL: http://developer.android.com/training/basics/fragments/index.html.

*Bundle* (2013). Android Open Source Project. URL: http://developer.android.com/reference/android/os/Bundle.html.

Garson, James (2012). "Connectionism". In: *The Stanford Encyclopedia of Philosophy*. Ed. by Edward N. Zalta. Winter 2012.

*GridView* (2013). Android Open Source Project. URL: http://developer.android.com/reference/android/widget/GridView.html.

*Intent* (2013). Android Open Source Project. URL: http://developer.android.com/reference/android/content/Intent.html.

*ListView* (2013). Android Open Source Project. URL: http://developer.android.com/reference/android/widget/ListView.html.

*Saving Data in SQL Databases* (2013). Android Open Source Project. URL: http://developer.android.com/training/basics/data-storage/databases.html.

*SimpleCursorAdapter* (2013). Android Open Source Project. URL: http://developer.android.com/reference/android/support/v4/widget/SimpleCursorAdapter.html.

Sutton, John (2012). "Memory". In: *The Stanford Encyclopedia of Philosophy*. Ed. by Edward N. Zalta. Winter 2012.

Tamada, Ravi (2012). *Android SQLite Database Tutorial*. URL: http://www.androidhive.info/2011/11/android-sqlite-database-tutorial/.

Vogel, Lars (June 6, 2013). *Android SQLite Database and ContentProvider - Tutorial*. URL: http://www.vogella.com/articles/AndroidSQLite/article.html.