# Stratiform: Dissemination of Computation

Srinivasan, Venkat

October 16, 2016

**ABSTRACT**

Stratiform is a framework for the dissemination of computational burden among peers in a temporarily connected mobile ad-hoc network. The idea behind Stratiform is to take a socialized approach to computational burden in a mobile community; when there is an abundance of idle devices and a computationally over-burdened peer within the ad-hoc network, Stratiform would step in and share the burden among all of the connected peers, allowing for quicker turn-around time for the requested computation. On the active device, Stratiform takes a parallelizable task and splits it into smaller tasks that are easier to compute; once the split is done, Stratiform disseminates the tasks to connected idle devices. The idle devices finish their smaller computations and transmit the individual results back to the requesting instance of Stratiform, which combines the parts into the full solution. This behavior takes advantage of every device in the network, allowing for the maximization of computation throughput of the entire network. Stratiform shines when it comes to problems that are more quickly solved in a massively-parallel architecture, such as a graphics processing unit (GPU). Rather than limiting the computation of such problems to just one device and one GPU, Stratiform seeks to treat the entire network like a large GPU consisting of individual GPUs, allowing for such tasks to finish more efficiently.

## 1 INTRODUCTION

Stratiform is an extension to Disseminate [1], which is research done in the field of opportunistic data dissemination intended to reduce burden on 3G infrastructure. Disseminate seeks to introduce the concept of delay-tolerance in data acquisition; Internet resources, such as web pages, are sent incomplete, with random chunks of data missing. Once the initial chunk is received by a node, missing chunks are opportunistically traded between connected nodes allowing for everyone to receive every missing piece, re-creating the complete resource eventually. Disseminate seeks to reduce redundant data transmission over 3G networks to the same geographical area. Stratiform uses a similar network architecture to share computational resources, instead of raw data. Stratiform seeks to treat the entire network as a parallel computation resource. It seeks to divide tasks into smaller chunks that get executed individually in each node and later recombined in order to get the full result in a more efficient manner, similar to how Disseminate seeks to divide and re-combine data.

## 2 DESCRIPTION OF PROJECT

Stratiform utilizes WiFi Direct in order to create a local ad-hoc network between Android devices. Stratiform monitors the status of all devices in this network, and registers devices when they connect into the network. These devices shall henceforth be referred to as “nodes”, or as “peers” within the network. Stratiform provides a dual-layered approach, where the user-task runs on-top of the Stratiform layer. Each user task provides Stratiform with a Map-Reduce architecture that Stratiform utilizes in order to break down and recombine user work, similar to the paradigm for massive parallelization in a Hadoop cluster [2]. When Stratiform receives a task, it utilizes a provided “Map” function to breakdown the task into smaller chunks that get sent to each individual peer. The master starts executing its own chunk. As each node finishes its task, it re-transmits the solution back to the requesting node. The Stratiform task on the requesting node asynchronously receives the messages and starts recombining the solutions, using the provided “Reduce” function, after the requesting node finishes its own execution.

### 2.1 Implementation

The current implementation of Stratiform can be broken down into three asynchronous and parallel-running layers: the lowest-level networking layer, the Stratiform layer, and the user-task layer. A top-down approach of the Stratiform implementation can be seen in Figure 1. The networking layer takes care of WiFi Direct connections and communications between nodes. The networking layer consists mainly of the WiFi Direct driver code, along with interfaces that aid in sending and receiving broadcast messages over the network. The Stratiform network protocol works similar to how the Controller Area Network (CAN) protocol does [3]. Every node always broadcasts when transmitting, albeit specifying a destination node ID if the broadcast is meant for one node alone; each receiving node filters out broadcasts not meant for it, but asynchronously puts messages intended for it into its First-In-First-Out (FIFO) queues. Stratiform accesses these queues periodically and takes out messages that are received. There are two kinds of messages: the

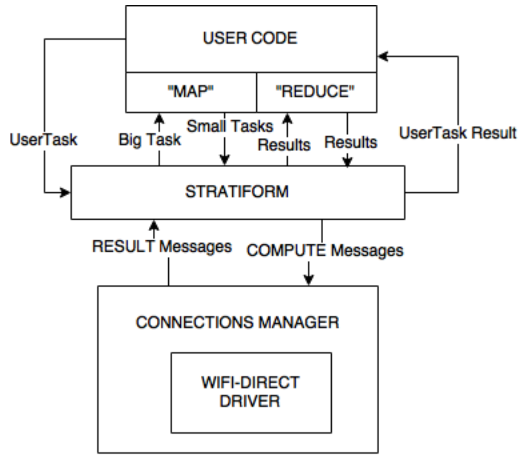


Figure 1: Stratiform Architecture

compute messages and the result messages. Compute messages are transmitted by the requesting device for other nodes to compute; receive messages are the results that are passed back from the other nodes back to the requesting node. The Stratiform layer uses the user-level "Map" function to break down a user task into smaller Compute tasks that get sent out to individual peers; it also uses "Reduce" function to recombine the received Result messages into the overall solution, which is then delivered to the user task. This entire architecture is implemented in the Android Operating System, using native WiFi Direct.

### 2.2 Connection Alternatives

The theoretical implementation of Stratiform requires a connected network on which all phones were able to communicate with each other. Stratiform envisions itself as a platform that does not require external infrastructure to work. To stay true to this spirit of Stratiform, the team decided on using a true ad-hoc network, instead of more traditional WiFi/Ethernet networks that require external infrastructure, such as a router [4]. Thus, the choice was between Bluetooth Low-Energy (BLE) and WiFi Direct. BLE has significantly lower data throughput, capped at a theoretical maximum of about 260 kilobits per second (Kbps) [5]; Stratiform leads itself to larger data bandwidth requirements, thus the team opted for WiFi Direct as the medium of communication. WiFi Direct boasts speeds up to 250 megabits per second (Mbps) [6].

### 2.3 Architecture Alternatives

The current iteration of Stratiform uses a variation of the MapReduce paradigm for breaking down problems and reconstructing solutions. We considered other ways of implementing this

Table 1: Stratiform Performance Evaluation

| Matrix Size (N) | Number of Nodes | Centralized Time Taken (ms) | Distributed Time Taken (ms) |
|---|---|---|---|
| 4 | 2 | 1.345 | 15.746 |
| 50 | 2 | 24.765 | 35.652 |
| 100 | 2 | 38.455 | 39.324 |
| 100 | 4 | 37.342 | 35.342 |
| 100 | 5 | 37.234 | 32.234 |

breakdown-reconstruction behavior. One of the other serious approaches we considered is delegating the actual assignment of work to the upper levels. Instead of Stratiform itself breaking down the task and assigning work to each connected node, the user-level task would receive a list of peers from the Stratiform layer. The user level code would break down the task and call an "assign" function from the Stratiform layer that would transmit the task blindly. An asynchronous thread would receive completed tasks and dump it into a container that the user-code then would read from. We decided to drop this implementation and instead opt for the MapReduce architecture mainly due to the enhanced abstraction that the latter provides. There is a very clean divide between the user-task and the Stratiform layer which the former solution does not provide. Utilizing MapReduce architecture hides the number of peers from the user-level code, as well as the inner workings of Stratiform. This clearer abstraction delineation provided by the MapReduce architecture convinced us to adopt it into our solution.

## 3 PERFORMANCE RESULTS

We opted to evaluate the performance of Stratiform using a matrix multiplication example. A $NxM$ and a $MxA$ matrix, where $N$, $M$ and $A$ are specified by the user, is multiplied using Stratiform. Our examples all make $N = M = A$. The time taken for a solution to be found for the matrix multiplication for various sized matrices and for various number of nodes is featured in Table 1. Stratiform shines when the amount of data is large and the network contains a large number of peer nodes. The overhead for transmission only allows Stratiform to be faster than locally processing the data if the matrix is over 100x100. The specified performance rates was taken in a low-traffic area. **Latency rates, and thus Stratiform results, might vary depending on where Stratiform is used.**

## 4 CONCLUSION

Stratiform is a revolutionary way of approaching computation in a connected world. Having a network of connected peers running Stratiform allows the entire network's computational throughput to increase, as idle devices within the network take-on computational burden of active devices. Stratiform's architecture allows versatility for what type of tasks it can execute, as the user is given control of how tasks are split and how results are recombined. Giving the user a large level of control, along with making the internals versatile, gives Stratiform the capability of being a very powerful tool that could change the way distributed processing is done in the world today.

## REFERENCES


[1] V. Srinivasan, T. Kalbarczyk, and C. Julien, “Demo abstract disseminate: A demonstration of device-to-device media dissemination,” *IEEE International Conference on Pervasive Computing and Communications*, 2015.

[2] International Business Machines Incorporated. What is MapReduce? [Online]. Available: https://www-01.ibm.com/software/data/infosphere/hadoop/mapreduce/

[3] National Instruments. Controller Area Network (CAN) Overview. [Online]. Available: http://www.ni.com/white-paper/2732/en/

[4] M. Chambers. The Hardware Needed for Basic Ethernet Networks. [Online]. Available: http://www.dummies.com/how-to/content/the-hardware-needed-for-basic-ethernet-networks.html

[5] Robin Heydon, “Bluetooth low energy,” Global Standards, CSR plc, Tech. Rep. [Online]. Available: https://www.bluetooth.org/DocMan/handlers/DownloadDoc.ashx?doc_id=227336

[6] WIFI Alliance. How fast is Wi-Fi Direct? [Online]. Available: http://www.wi-fi.org/knowledge-center/faq/how-fast-is-wi-fi-direct