

# Can Infopipes Facilitate Reuse in a Traffic Application?

Emerson Murphy-Hill, Chuan-kai Lin, Andrew P. Black, and Jonathan Walpole  
Maseeh College of Engineering & Computer Science, Portland State University

1900 SW 4th Avenue  
Portland, OR 97201  
503-725-4036

{emerson,cklin,black,walpole}@cs.pdx.edu

## ABSTRACT

Infopipes are building blocks for creating data streaming applications, and it is claimed that Infopipes facilitate code reuse. To evaluate this claim, we have built a significant traffic application in Smalltalk using Infopipes. In this paper and accompanying poster, we present a short introduction to Infopipes, a traffic problem and algorithmic solution, and the types of reuse Infopipes facilitate in our implementation of said traffic algorithm.

## Categories and Subject Descriptors

D.2.13 [Software Engineering]: Reusable Software

## General Terms

Design

## Keywords

Infopipes, Traffic Analysis, Truck Volume, Streaming Applications

## 1. INTRODUCTION

Streaming applications are notoriously difficult to build because they often have to be constructed from the ground up. Infopipes seek to alleviate this difficulty by providing an abstraction on which streaming applications can be written [1]. Infopipes allow the programmer to define streaming components that focus on data flow aspects of applications, rather than control flow. We will show that many Infopipe components are reusable through a number of means.

## 2. A TRAFFIC APPLICATION

To verify the claim about the reusability of Infopipes, we have implemented a recent traffic algorithm using Infopipes. This algorithm was first described by Kwon and colleagues in 2003[2]. Implementing this algorithm gives us the experience of implementing a real-world streaming application, so that we may determine the degree of reusability offered by Infopipes.

The algorithm allows truck volume to be estimated on highways using existing equipment. While truck volume can easily be measured using modern traffic equipment, the cost of deploying such equipment can be prohibitively high. The truck volume algorithm presented by Kwon and colleagues leverages existing, widely deployed highway equipment. Figure 1 depicts a variation of this equipment.

## 3. IMPLEMENTATION

We implemented the algorithm in Squeak and Smalltalk/X, two flavors of Smalltalk. The implementation uses a number of general purpose Infopipes, as well as new, traffic-specific Infopipes. The implementation defines about 50 classes. An instance of the application that analyzes a 4-lane highway encompasses more than 100 Infopipes. Different pieces of the application have varied data flow characteristics and may run in separate threads, operating system processes, or on different machines. We believe this application is representative of a modern streaming application.



Figure 1. Induction loops on a typical US highway in Portland, Oregon.

## 4. FINDINGS

We have identified three main types of reuse that Infopipes help facilitate. Each type is illustrated in Figure 2 by the `HighwayVehicleClassifier` which encompasses a large number of other Infopipes. The first type of reuse is class inheritance. Because Infopipes are defined as object-oriented classes, we may reuse Infopipe behavior by subclassing existing Infopipes and specializing behavior. For instance, `VariablePipe`, shown in Figure 2, reuses `Buffer` via subclassing. The second type of reuse is parameterization. Some Infopipes take data-processing parameters, making these Infopipes useful in different data-processing contexts. For instance, Figure 2 contains a `FunctionPipe` which applies a velocity correlation function. The third type of reuse is simply instantiation. We can reuse Infopipes by simply instantiating them in different contexts. In Figure 2, a `VehicleLengthCalculator` is instantiated  $n-1$  times, where  $n$  is the number of traffic lanes the application runs on.

## 5. CONCLUSIONS

Infopipes facilitate a high degree of reuse in our implementation. We were able to reuse Infopipes from earlier traffic experiments, have reused Infopipes repeatedly in this implementation, are currently reusing Infopipes from this implementation in other traffic applications, and anticipate reusing current Infopipes in future applications. In sum, we feel that the Infopipe abstraction facilitates reuse in a real-world streaming application.

## 6. REFERENCES

- [1] R. Koster, A. P. Black, J. Huang, J. Walpole and C. Pu. Infopipes for Composing Distributed Information Flows. In Proceedings of the ACM Multimedia Workshop on Multimedia Middleware, Ottawa, Canada, October 2001.
- [2] J. Kwon, P. Varaiya, and A. Skabardonis. Estimation of Truck Traffic Volume from Single Loop Detector Outputs Using Lane-to-lane Speed Correlation. Presented at TRB 2003 and forthcoming in *Transportation Research Record*.

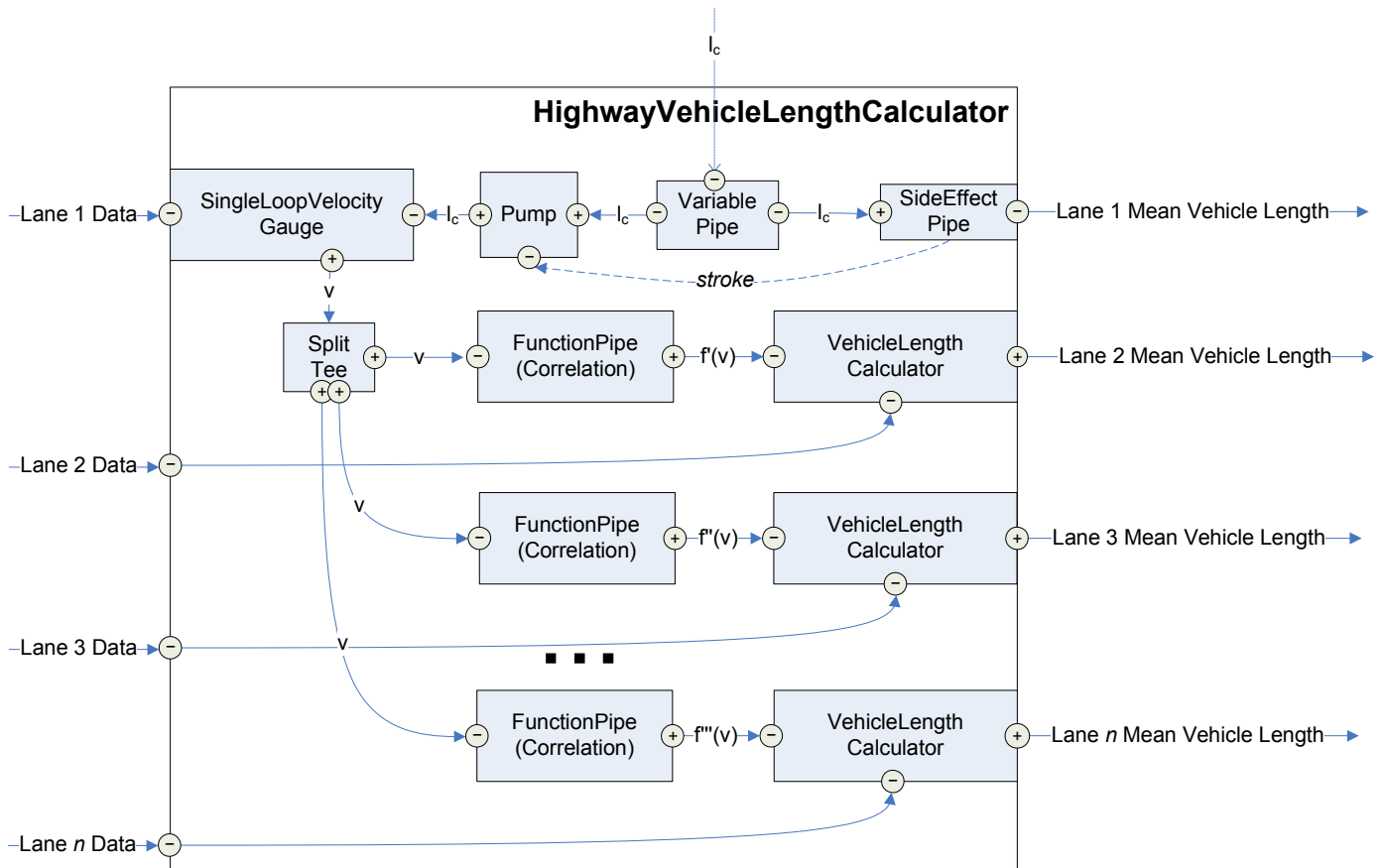


Figure 2. The `HighwayVehicleClassifier`, the heart of the truck volume estimation algorithm.