Return of the Table

Michael Vitrano and Jared Pochtar
4115 Final Project
Professor Stephen Edwards
Problem Area

• Working with relational databases can be difficult using traditional SQL queries, which makes querying certain subsets of data difficult.

• Additionally, learning certain statistics about a dataset can be impossible to do solely in SQL.

• The traditional way of computing these statistics is to issue queries to the database, parse the information through a database driver and use a second language to compute statistics.
RT’s Solution

• Relational tables and records become first-class language objects
• Driver for interfacing with database is built-in to the language
• Filter, Join and Map operations allow the programmer to easily create any composite dataset from the original database or filter to any subset of the original database
• Reporting generated datasets in a neat and organized fashion is easy
• Committing data to the database is also easy!
• RT code is translated into Java source code and automatically compiled to .class files
• rtc.sh and rt.sh are scripts that manage compiling and running RT programs respectively
• rtc, the RT translator, is written in OCaml and is responsible for Scanning, Parsing, Semantic Analysis and Code Generation
• Generated Java source is linked against RTLib, which provides the framework for all table related features
Demonstration

• **Problem:**
  - Business has information regarding customer purchases in their database and want to know who are their best customers
  - Also want to know which store location and item sold are the most popular

• **Steps:**
  - Join customer, item, store and purchase data
  - Calculate each customer’s total amount purchased, favorite store, amount spent at that store, favorite item and the amount spent on that item
  - Sort customers based on total amount purchased
  - Display results and commit to database
Lessons Learned

• Quit talking about the project and just start writing code
• Work with people who you trust to do work on time and correctly
• Divide the project into separate units and assign each to a team member, make sure everyone knows how the parts interface
• SVN is a necessity, but can cause headaches if you’re lazy about using it
Advice to Future Groups

• Start early, this can not be stressed enough
• This is an iterative process and must be done steps at a time. It is impossible to do all at once
• Pick an end point to your project and stick to it
• Test cases are important for two reasons, they both make sure you’re doing it right and keep you focused on what the language is trying to do
• Choose a problem area that interests you