**CSFV**

**University of Washington**

**Monthly Technical Report for January, 2013**

Reporting period: 1 January 2013 – 31 January 2013

Date of Report: 15 February 2013

Project Title: Verigames

Contract Number: FA8750-12-C-0174

Program Manager: Dr. Drew Dean, DARPA I2O

Submitted by:

Michael Ernst / Zoran Popović

Computer Science & Engineering

University of Washington

AC101 Paul G. Allen Center, Box 352350

185 Stevens Way

Seattle WA 98195-2350

Fax: (206) 543-2969

E-mail: {mernst, zoran}@cs.washington.edu

**Distribution List**

DARPA/I2O

ATTN: Drew Dean

team-dean@darpa.mil

3701 North Fairfax Drive

Arlington VA 22203-1714

AFRL/RITA

ATTN: Dilia Rodriguez

Dilia.Rodriguez@rl.af.mil

525 Brooks Road

Rome NY 13441-4505

**Unclassified - For Official Use Only**

# Verigames - Project Progress

**1. Expected Progress This Month**

* Oracle is planning to push our Java 8 changes from a special OpenJDK “type annotations branch” to the main branch on January 22.
* Annotations:
  + Get the insert annotations to source part of the Annotation File Utilities (AFU) working for changes to the bytecode.
  + Start working on updating AFU with new compound type representation (this will also fix some bugs in AFU's current handling of compound types).
* Experiment with a unit type system.
* Continue integrating the dataflow framework into Verigames.
* Game client:
  + Continue working through bugs and other issues as Verigames ingests larger software programs.
  + Finish integrating new graphics library to improve performance and support mobile platforms.
* Continue hiring.

**2. Accomplishments This Month**

Our changes to Java appear in build M6 of Java 8, which was released by Oracle on January 31.

Classfile changes are complete in javac, the Checker Framework, and the Annotation File Utilities.

Source code insertions of annotations on compound types now works with the new compound type location syntax. We also fixed some bugs in the way wildcards were handled.

Added AFU support for source code annotation insertions on inner types. This is done in two passes: one pass inserts the receivers/casts and the second pass adds the annotations on the compound types. This is because the code to determine where to insert the annotation for the new annotation syntax requires type information (to know if something is a package, or a static inner class, etc.) This type information won’t be present with a cast insertion. Inserting the cast in the first pass and then re-analyzing the code for type information in the second pass will get the type information needed.

Created additional type systems: OSTrusted, Hardcoded and Random.

Continued non-pipe-centric visual interpretation of level, an approach that will allow for more complex programs to be represented in a game board.

Explored simplification of base mechanics make gameplay more accessible. In particular, we are investigating eliminating the concept of buzzsaws. Conflicts basically play the same role, as long as they are worth a certain amount of negative points. Conflicts caused by different colors or the ones caused by big balls going in small pipes could cost different amounts of points.

We want to make sure that players push conflicts as low down the pipes as possible, such that they impact as few nodes as possible. We can address this if each conflict types are worth a fixed amount of points (Y). Whenever a conflict occurs, it spawns a few conflicts down the road, each worth Y times number of conflicts.

At the same time, we also want programmers to have as few annotations (conflicts) to look at as possible. For example, if 8 nodes with conflicts have the same parent (with only 2 other child nodes devoid of conflicts), the conflict should probably be on the parent. We need to find the right balance in scoring child nodes in order to encapsulate the need for this balance.

Here are a few remaining open questions for visual design:

* Does the new square grid representation handle all the necessary components?
* Can we eliminate the traveling “balls” from the design? Aside from their conceptual use, they do not contribute more than pipe widths on their own.
* Can we represent polymorphism (method inputs/outputs are linked, such that when a pipe changes size, the other one does as well) in some other fashion? i.e. can we just visually link two squares together?
* Is every node affected by a conflict worth the same amount of negative points (linear points for conflicts)?

We prepared presentations for CSFV PI Meeting, February 5-7, 2013 in San Antonio, TX.

**3. Deliverables Submitted**

N/A

**4. Publications Made**

N/A

.

**5. Meetings**

* Weekly UW Verigames full team meeting
* Weekly UW CGS design meeting
* Weekly integration conference call

**6. Issues or Concerns**

Iterative game development cannot start until our IRB is approved. Our game design and development process is highly reliant on gathering data from real players and making adjustments to the game based on perceived patterns of play. Our inability to collect and study data in the short term means that there are a number of game design questions that will remain open for the time being.

**7. Plans for Next Month:**

* Attend and present at CSFV PI Meeting, February 5-7, 2013 in San Antonio, TX.
* Annotations:
  + Complete work on annotations in qualified types, and start working on annotations on compound types in inserted receivers and casts.
  + Determine how AST paths will be stored in the XML.
* Complete integration of the dataflow framework.
* Continue creating type systems: Encrypted, SafeFileType, and OneWayHashWithSalt.
* Game design:
  + Continue investigating game design simplification, especiallycentered around buzzsaw placement and game board graphics.
  + Complete visual development on non-pipe-centric representation of game board.
  + Finish integrating new graphics library.
* Integration:
  + Send list of initial level parameters to TopCoder.
  + Respond to TopCoder mini-site questionnaire.
* Continue hiring.

**8. Financial Summary**

January: Projected expenditures for the month were estimated at $105k. Actual expenditures were $110k. The first Julia Srl invoice has posted; another is pending for February and a third has been received and is being processed. Suzanne Millstein started in January and is paid 50% on this project (and 50% on DARPA APAC SPARTA).