**CSFV**

**University of Washington**

**Monthly Technical Report for April 2013**

Reporting period: 1 April 2013 – 30 April 2013

Date of Report: 15 May 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**

* Continue integration of the game with the Resource Allocator.
* Integration of the dataflow framework into the Game Solver.
* Determine how AST paths will be stored in the XML.
* Continue creation of type systems.
* Continue iteration on new representation of game levels.
  + Explore possibility of letting players alter level layouts themselves.
  + Decide on design of map.get in new representation.

**2. Accomplishments This Month**

* We are still working through bugs and remaining edge cases in the dataflow framework.
* Improved readability of GameSolver, splitting several methods into smaller ones, and did some other miscellaneous cleanup items.
* We decided to punt on map.get representation for now, as we have some ideas for how it will work but need to see real programs regularly pushed through our system first before we can be certain it is correct.
* We are continuing work on the type systems for the checker, specifically the Lock type system.
* Subcontractor Julia srl continues to work on the implementation of security type inference.
  + Instantiated the generic information flow framework to @Random, @NotHardCoded, @OsTrusted and @Trusted.
  + Implemented providing manual security annotations for return values, method and constructor parameters and fields.
  + Generated jaif files for the four security properties already implemented for hadoop-common and hadoop-auth, and attempted to annotate the source code using the inferred jaif and the annotation file utilities (AFU). This was not successful yet, due to a problem in what we believe is either the AFU or the compiler.
* Game functionality and features were expanded.
  + Reworked grid level zooming/panning/bounds and improved panning and zooming experience to approximate that of Google Maps—an important interface convention for the large boards we expect to display.
  + A preliminary scoring system was added. Grid boxes are labeled according to possible points.
* Continued iteration on new representation of game levels.
  + Currently, the game represents all incoming and outgoing lines as being the same width as the box they connect to, but that is an overly conservative representation. We believe there are going to be cases (even after we implement flow sensitivity analysis) where one set of linked classic “pipes” can have more than one type of input (in Classic: a small ball can enter a green pipe in one location, but a large ball could enter the green pipe in a separate location). We had been hoping that only one type of input could ever enter a linked pipe, but we were able to come up with several cases where that is not true even with flow sensitivity present. As a result, we will need to continue to think about how we represent the relationship between incoming and outgoing lines on the boxes, since they will be related in certain cases.
  + We are also considering semi-automation of certain optimizations across the board, akin to actions like “shake” and “wiggle” in Foldit. The expected large size of boards created by real-world programs leads us to imagine that simply turning on or off boxes will not be engaging enough to drive return visits. Therefore, the gameplay focus may shift to giving players a variety of tools to change large parts of the board at once.
* Worked with TopCoder on game integration.
  + Ongoing integration with authentication system and Resource Allocator.
  + Agreed on features to be developed by integrator versus TA1 for the mini-sites.
* A new programmer, Doug Hoeger, was hired into the Programming Languages group and will work on the Annotation File Utilities full-time.
* Both the UW and the Air Force have now approved our application for experimentation with human subjects.

**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**

N/A

**7. Plans for Next Month:**

* Begin testing the game with larger boards— much larger than previously tested.
* Continue integration of the game with authentication server and Resource Allocator.
* Integration of the dataflow framework into the Game Solver.
* Finish Lock type system.
* Continue iteration on new representation of game levels.
  + Begin generating edges for every pipe segment, not just for every unique pair of joined edge sets.
  + Begin process of linking incoming/outgoing pipes of subnetwork nodes to their inputs/outputs.
  + Change XML format for level generation for new Gridworld representation.

**8. Financial Summary**

April: Projected expenditures for the month were originally estimated at $142k. Actual expenditures were $61k, however, this includes a negative adjustment to indirect costs that were over charged last month on our subcontract.

Julia Srl continues contributions to the project, and the total PO for their subcontract has been increased to $148,756. No invoice posted this month, but one is pending for next month.

Staff and grad student funding remains the same.