



## Thesis title: *Source Code Visualization Using the Unity Game Engine*

### Thesis description:

With ever-increasing complexity of software-systems, there is a growing interest in *program comprehension*. Software visualization is a popular program comprehension technique used to support software maintenance, reverse engineering, and analysis of software evolution. One of the most recognized approaches to visualizing source code, at least in the academic community, is the city metaphor, i.e., to depict software systems as three-dimensional cities. Wettel and Lanza developed the pioneering tool CodeCity [1], but several similar solutions have been proposed in the last decade.

Simply presenting appealing visualizations is not the goal of software visualization – instead, the aim is help the user discover actionable insights about the system under study. Visual analytics is defined as "the science of analytical reasoning facilitated by interactive visual interfaces". Our previous research confirmed the importance of *interaction*, as participants in three focus groups stressed the need for operations such as zooming and filtering when exploring prototype visualizations [2].

Game development is all about designing user interaction. By using a game engine to develop software visualizations, state-of-the-art user interaction is provided as an out-of-the-box feature. Furthermore, advances in game development platforms have lowered the entry bar for developers considerably – never has creating high-quality games been so accessible. Unity is one of the most popular contemporary game engines [3], used both in the booming indie scene and for major game titles such as Cities: Skylines and Pokémon GO. Renowned throughout the industry as the multiplatform game engine, Unity allows you to target both desktop and mobile environments.



The objective of the thesis project is to develop an application in Unity that visualizes git repositories as cities. Necessary steps include implementing a source code parser, calculating source code metrics, designing a corresponding visual representation, and dynamically generating assets within Unity. Finally, design an evaluation of your approach, e.g., focus groups or controlled experiments – your work could either be compared to CodeCity or you could assess whether your tool supports a software engineering task that requires program comprehension.

- [1] R. Wettel and M. Lanza. Visualizing software systems as cities, In *Proc. of 4th IEEE Int'l Workshop on Visualizing Software for Understanding and Analysis*, pp. 92-99, 2007.
- [2] E. Engström, M. Mäntylä, P. Runeson, and M. Borg. Supporting Regression Test Scoping with Visual Analytics, In *Proc. 7th Int'l Conference on Software Testing, Verification and Validation*, pp. 283-292, 2014.
- [3] Unity - A multi platform game development tool. <http://unity3d.com>

### Tasks:

- Learn the basics of Unity's 3D engine, e.g., by enrolling in free MOOCs and completing tutorials.
- Study related work on visualizing source code, with a focus on the city metaphor and CodeCity.
- Develop a Unity application (desktop or mobile) that visualizes git repositories.
- Evaluate how the application compares to CodeCity or how it supports a realistic software engineering task.

### Key skills:

- Essential: programming, software engineering
- Preferred: Unity, game development, C#, git, data visualization, user evaluation

Applications should include a brief cover letter, CV, and recent grades. In your application, please provide examples of previous programming experiences or other projects that you consider relevant for the position.

**Expected start time:** January-March 2017

**Location:** SICS Swedish ICT AB, Lund

**Contact person/s:** Dr. Markus Borg ([markus.borg@sics.se](mailto:markus.borg@sics.se))

## About SICS:

SICS Swedish ICT is a leading research institute for applied information and communication technology in Sweden. SICS is a part of RISE, Research Institutes of Sweden, a non-profit research organization owned by the Swedish government and industry. SICS' mission is to contribute to the competitive strength of Swedish industry by conducting advanced and focused research in strategic areas of computer science, and actively promote the uptake of new research ideas and results in industry and society at large. SICS is an active participant in collaborative national, European, and other international R&D programs.

The Software and Systems Engineering Laboratory (SSE) has as its research theme "successful development of software-based systems for the digital society". The focus is on providing relevant solutions for software and systems development in a society that becomes increasingly connected and automated, and frequently updated through continuous deployment. Special attention is given to large-scale cyber-physical systems, where traditional embedded systems are evolving into systems-of-systems through connectivity, using technologies from the Internet of Things. Currently, the research activities in the SSE lab focus on four areas: process evolution, system architecture, software ecosystems, and system qualities.

The RISE institutes SP, Swedish ICT and Innventia are merging in order to create a unified institute sector and become a stronger innovation partner for businesses and society. At the end of the 2016 we will change our name to RISE. Read more at [www.ri.se/en/about-rise](http://www.ri.se/en/about-rise)

SICS Swedish ICT AB,  
Software and Systems Engineering Lab  
Ideon Science Park  
Building Beta 2 3v  
Scheelevägen 17  
SE-223 70 Lund

---