UROP 2010

Automated Services Management over Virtualization

Traditional Data Centers (DCs) use static allocation of resources to ensure satisfaction of Service Level Agreements (SLAs) under peak demand or in worst-case environment conditions (over-provisioning).
Virtualization allows system components to be decoupled from the underlying hardware enabling dynamic allocation of resources, fast replication of virtual machines, live migration of running components, and server consolidation.
In the context of the RESERVOIR FP7 project, USI is working with first-class industrial partners (IBM, SAP, Sun, Telefonica I+D) towards the definition of new generation of Virtualized Data Centers (VDCs).
Services running on VDCs must be appropriately designed to benefit from virtualization, e.g. by automatically redistributing the internal workload when new resources become available. Ad-hoc optimization of existing services and compositions could increase development costs and eventually break existing systems in case of development errors.
The development of frameworks for the automated management of services in VDCs is thus necessary to provide a reliable way to adapt and compose existing services to effectively run over VDCs.
The aim of this work is to extend an existing service composition framework, JOpera, to provide mechanisms for the automated management of services within virtualized data centers.
We are looking for a motivated student with previous knowledge on Java programming, and Web development.
The student will have a chance to improve his/her knowledge on Web2 and Web services, Java, distributed systems, operative systems, virtualization and hypervisors.

The student will be coached while:

- Designing extensions for the JOpera framework for the automated dispatching of requests within VDCs thus permitting to redistribute services invocations according to resources availability and load.
- Designing and running a set of distributed tests using a model-driven tool for experiment automation (Weevil)
- Designing a web service composition to evaluate the performances of the framework in a concrete case study.
- Analysing collected data and produce a brief report

Mentors: Alessio Gambi, Giovanni Toffetti, Cesare Pautasso

Sponsor: Mauro Pezzè

Università della Svizzera italiana | Faculty of Informatics | Information Retrieval Group

2010 UROP Internship Project Proposal

Keywords Information Retrieval, chat mining, crawling, twitter, blogs, IRC

Goal The creation and indexing of a document collection containing chat logs data.

Description As part of our project “ChatMiner”, that aims at the mining of conversational content for topic identification (what is the conversation about?) and author identification (who are the people involved in the conversation?), we need to create a collection of chat dialogs taking places on the Internet. Ideally every online conversation (Facebook wall-to-wall, Twitter-, Skype-, Yahoo-, MSN-, Google- chatting platforms) can be interesting for us, but for the purpose of this UROP we will concentrate on selected sources.

Task for the student

- investigate the literature/the web to find existing collections of the same nature	1-2 weeks
- create the crawler to download the data from two main sources (if not others found in the previous step): Twitter, IRC - format the data	2-3 weeks
- index and preliminarily analyze the collection - retrieve relevant chat conversation	2 weeks
- write a user guide and/or a report	1 week

Program languages:

C/C++ or Java are preferred but we are open to other solutions.

Contact information

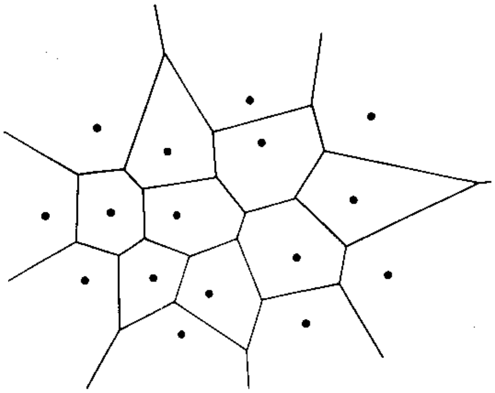
Giacomo Inches
PhD Student
www.ir.inf.usi.ch/urop
giacomo.inches@usi.ch
+41 (0)58.666.4315

Undergraduate Research Opportunities Program

Summer project in Algorithms and Computational Geometry

The farthest line-segment Voronoi Diagram in the max norm.

This project will introduce students to research in the area of discrete algorithms and computational geometry. The Voronoi diagram is a simple but powerful geometric object that encodes nearest neighbor information with numerous applications in diverse areas. Given a set of point sites S in the plane, the Voronoi diagram is a partitioning of the plane into regions, one for each site, such that the Voronoi region of a site s is the locus of points closer to s than to any other site. The following figure illustrates the ordinary Voronoi diagram of a set of points in the Euclidean plane.



Several generalizations of standard Voronoi diagrams have been proposed. For example, the farthest-point Voronoi diagram of a set of point-sites S divides the plane into regions such that the farthest Voronoi region of a site s is the locus of points farthest away from s than from any other site. In this project, we will study the farthest Voronoi diagram of a set of line segments under the simple max metric (instead of the Euclidean distance). The max distance between two points is defined as the maximum between their vertical and horizontal distance. We will study the structural properties of this diagram (this is an open problem) starting with special cases. If successful, we will design efficient algorithms for its construction.

Skills required

Good background in Algorithms and Data Structures. Familiarity with Discrete Mathematics. Ability with the implementation of Algorithms a plus.

Contact

Prof. Evanthia Papadopoulou, SI-209

UROP Project 2010

Prof. Cesare Pautasso, Daniele Bonetta

Process-based Management of Cloud Computing Infrastructure

Background

Cloud Computing is an emerging computing capability that provides an abstraction between the computing resource and its underlying technical architecture (e.g., servers, storage, networks), enabling convenient, on-demand access to a shared pool of configurable virtual computing resources. These can be rapidly provisioned and released with minimal management effort. Examples of modern cloud computing solutions include (but are not limited to) Amazon EC2, Google AppEngine and Microsoft Azure.

Objective

The goal of this project is to design and implement an extension to an existing Web Service Composition Engine (JOpera) able to manage virtual resources deployed within the Cloud, and allow dynamic instantiation, monitoring and migration of virtual machines. The software that will be targeted in this project is the JOpera composition engine as the proposed cloud controller, and the Eucalyptus Private Cloud as IasS infrastructure.

The purpose of this work is to study different ways to access and manage Cloud-based services in order to manage their configuration and control their performance using a process-based composition tool. The idea is to develop a plugin for the JOpera engine providing access to Cloud management web services, and to build example processes to demonstrate the capabilities of the plugin (e.g., the hosting virtual machine, the amount of virtualized resources, etc). At the end of the project a working prototype is expected.

Technologies

With this project you will become an expert in Java, Eclipse plugin development and Eucalyptus-based Clouds management. The work will involve a non-trivial amount of programming (with Java and RESTful Web Services).

Contact

For more information, please contact Daniele Bonetta <daniele.bonetta@usi.ch>,
or Prof. Cesare Pautasso <cesare.pautasso@unisi.ch>

References

Eucalyptus Private Cloud: http://www.eucalyptus.org

UROP Project 2010

Prof. Cesare Pautasso, Daniele Bonetta

Continuous Integration for Eclipse Plugins

Background

The Eclipse software platform is one of the most adopted development environment for programming with the Java language. Its modular, plugin-based architecture makes it easy to extend and customize the platform with additional functionality. Multiple plugins can be collected into features and products which are published on so-called update sites that make it easy to install software updates across the Web.

Objective

The goal of this project is to fully automate the plugin release engineering procedure. In this project you will design and implement a complete Eclipse plugin for compiling, building, testing and deploying other Eclipse plugins. The main benefit of your plugin will be to automate critical tasks in the plugin development lifecycle, which will enable the *continuous integration* of large software projects built on the Eclipse platform. At the end of the project a fully working tool is expected that will be contributed back to the Eclipse open source community.

The idea is to integrate a code repository (e.g., Subversion or CVS) with a continuous integration tool (e.g., Hudson or Cruise Control) so that so-called *nightly builds* of a set of Eclipse plugins can be automatically produced, tested (both with regression JUnit tests, but also with performance benchmarks), and released on a Web site.

Technologies

With this project you will become an expert in Java, Eclipse Plugin development and continuous integration techniques. The work will involve a non-trivial amount of scripting (with ant build files and UNIX shell scripts) in a Linux environment.

Contact

For more information, please contact Daniele Bonetta <daniele.bonetta@usi.ch>,
or Prof. Cesare Pautasso <cesare.pautasso@unisi.ch>

References

Paul Duvall, Steve Matyas, Andrew Glower,
Continuous Integration: Improving Software Quality and Reducing Risk, Addison-Wesley, July 2007,
ISBN 0321336380

Università
della
Svizzera
italiana

Faculty
of Informatics

JOpera
Process Support for Web Services

UROP Project 2010

Prof. Cesare Pautasso, Dr. Achille Peternier, Daniele Bonetta

Performance Analysis and Visualization for Service Composition Profiling

Background

Service Composition is a popular and powerful programming methodology that allows developing complex distributed applications connecting together existing services with coordination technologies called “Service Composition Engines”. In such complex applications performance is a priority issue, and the more complex the composition is, the more complex the performance evaluation becomes.

Objective

The goal of this project is to extend an existing service composition engine, JOpera, with performance analysis and visualization features. In this project you will design and implement a rich Internet application to help evaluate the performance of service compositions. The main benefit of your work will be to allow the identification of performance bottlenecks on composed web services, and to aid the developer with graphical rendering (2D, 3D) of performance statistics and data. At the end of the project a fully working tool is expected that will be included in the JOpera distribution.

The idea is to develop a Web front-end with interactive analysis and visualization features (a good example for the work is the Google Analytics platform).

Technologies

With this project you will become an expert in HTML5 applications, performance evaluation and Web Service composition. The work will involve a non-trivial amount of Web development and rely on data plotting tools such as the Google Charts API or InfoViz.

Contact

For more information, please contact Daniele Bonetta <daniele.bonetta@usi.ch>, or Dr. Achille Peternier <achille.peternier@unisi.ch>

References

Google Analytics Framework: **http://www.google.com/analytics**

Google Charts API: **http://code.google.com/apis/visualization/documentation/gallery.html**

JOpera: **http://www.jopera.org/**

UROP Project 2010

Prof. Cesare Pautasso

JSDot

Background

The goal of JSDot is to provide a native lightweight JavaScript library for browser-based graph visualization. The current version of the library also features basic support for interactive graph editing and is built using SVG.

Objective

The goal of this project is to continue the development of JSDot by reviewing the existing code and graph model and by implementing the following set of features. The result should be a generic graph viewer and editor, which can be enriched by plugins which can add a more specific semantic to the graph, giving more features or applying some restrictions. It should become possible to embed multiple graph widgets in the same page.

- Extend JSDot API to support generic event handlers for editing tools and views
- Associate nodes and edges with css styles and stencils
- Extend arrow shapes with decorators on both ends and bendpoints
- Allow users to select and operate on multiple nodes
- Use HTML5 local storage to save and retrieve graph JSON models

Additionally the project will implement some of the following layout algorithms:

- Align vertically, horizontally or on a straight line
- Interactive Rotate/Stretch
- Grid
- Hierarchical, rank-based
- Force-based

Technologies

With this project you will become an expert in HTML5, SVG, and advanced JavaScript.

Contact

For more information, please contact Cesare Pautasso <cesare.pautasso@usi.ch>

References

JSDot: **https://code.google.com/p/jsdot/**

Università
della
Svizzera
italiana

**Facoltà
di scienze
informatiche**

UROP Project 2010

Prof. Cesare Pautasso Marcin Nowak

Architectural Knowledge Visualization

Modern software systems are often very complex. In order to manage this complexity, designers need useful design abstractions - such as ones found in software architecture. A process leading to development of successful software architectures requires a significant amount of domain specific and generic knowledge, which can be only obtained from the collaboration of multiple experts. This project focuses on the development of the novel methods of visualizing Architectural Knowledge. Visual design aids aim at making body of knowledge easily accessible (browsing) and building right intuition about its properties (complexity for example).

In the context of research work on Software Architectural Decisions, web-based tool named Software Architecture warehouse is being actively developed. The aim of this project is includes the development of client-side JavaScript extensions implementing visual aids for navigation, processing and analysis of architectural Knowledge.

Project objectives include:

- implementation and performance tuning of the gradient interpolation algorithm for metrics visualization,
- implementation of design space (graph) overview visualization,
- development of custom modes of visualization adopted for:
  - single-user, local, interactive decision making,
  - multi-user, collaborative decision making and arbitration.

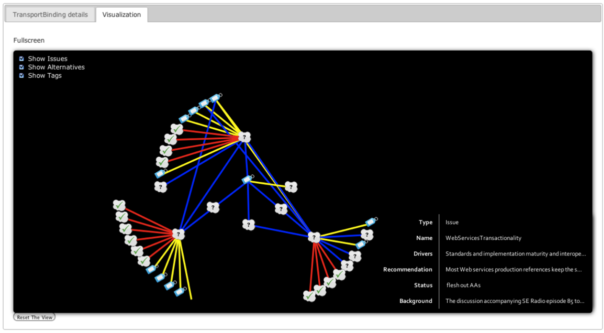
More information can be found here:

- Software Architectural Warehouse project website (http://saw.inf.unisi.ch),
- SHARK 2010 paper on new challenges and opportunities in the field (http://saw.inf.unisi.ch/homepages/nowak/publications).

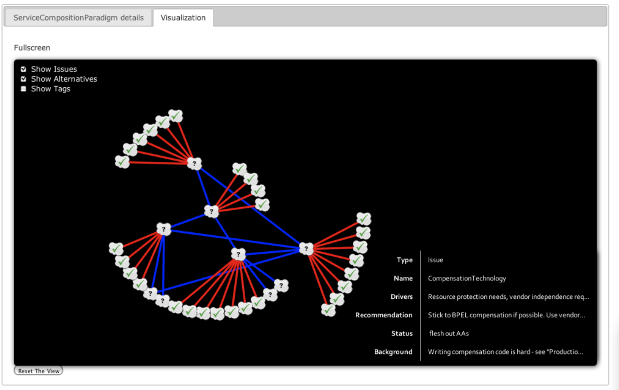
We are looking for student with interest in modern dynamic-web technologies. Strong JavaScript skills will be advantageous.

Keywords: 2D/3D graphics, metrics,

Technologies: HTML5, JavaScript



Type	Issue
Name	WebServicesTransactionality
Drivers	Standards and implementation maturity and interope...
Recommendation	Most Web services production references keep the s...
Status	flesh out AAs
Background	The discussion accompanying SE Radio episode 85 to...



Type	Issue
Name	CompensationTechnology
Drivers	Resource protection needs, vendor independence req...
Recommendation	Stick to BPEL compensation if possible. Use vendor...
Status	flesh out AAs
Background	Writing compensation code is hard - see "Productio...

Contact person: Marcin Nowak (Marcin.Nowak@usi.ch)

Università
della
Svizzera
italiana

**Facoltà
di scienze
informatiche**

UROP Project 2010

Prof. Cesare Pautasso Marcin Nowak

Collaborative Decision Making

Modern software systems are often very complex. In order to manage this complexity, designers need useful design abstractions - such as ones found in software architecture. A process leading to development of successful software architectures requires a significant amount of domain specific and generic knowledge, which can be only obtained from the collaboration of multiple experts.

In the context of research work on Software Architectural Decisions, a new web-based knowledge management tool named Software Architecture Warehouse is being actively developed. The Aim of this project includes the development of client-side JavaScript and HTML5 extensions implementing online-collaboration scenarios.

Project objectives are:

- implementation of the asynchronous content push-update mechanism for the web-clients.
- implementation of the generic and specialized widget generators,
- solution performance and scalability evaluation.

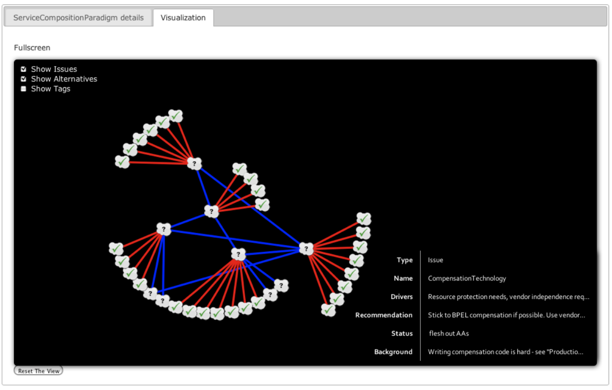
More information can be found here:

- Software Architectural Warehouse project website (http://saw.inf.unisi.ch),
- SHARK 2010 paper on new challenges and opportunities in the field (http://saw.inf.unisi.ch/homepages/nowak/publications).

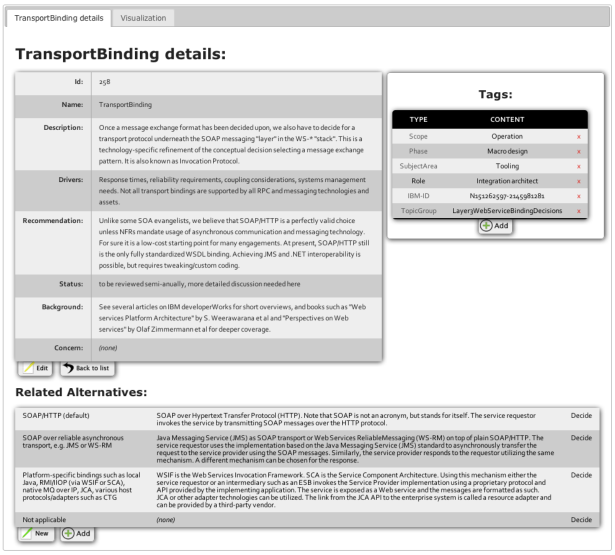
We are looking for student with interest in modern web technologies. Coding skills in JavaScript and/or Ruby would be advantageous but are not a must.

Keywords: software architecture, knowledge sharing, non-linear development,

Technologies: Ruby on Rails, JavaScript, HTML5



Type	Issue
Name	CompensationTechnology
Drivers	Resource protection needs, vendor independence req...
Recommendation	Stick to BPEL compensation if possible. Use vendor...
Status	flesh out AAs
Background	Writing compensation code is hard - see "Productio...



TransportBinding details:

Id:	258
Name:	TransportBinding
Description:	Once a message exchange format has been decided upon, we also have to decide for a transport protocol underneath the SOAP messaging "layer" in the WS-* "stack". This is a technology-specific refinement of the conceptual decision selecting a message exchange pattern. It is also known as Invocation Protocol.
Drivers:	Response times, reliability requirements, coupling considerations, systems management needs. Not all transport bindings are supported by all RPC and messaging technologies and assets.
Recommendation:	Unlike some SOA evangelists, we believe that SOAP/HTTP is a perfectly valid choice unless NFRs mandate usage of asynchronous communication and messaging technology. For sure it is a low-cost starting point for many engagements. At present, SOAP/HTTP still is the only fully standardized WSDL binding. Achieving JMS and .NET interoperability is possible, but requires tweaking/custom coding.
Status:	to be reviewed semi-anually, more detailed discussion needed here
Background:	See several articles on IBM developerWorks for short overviews, and books such as "Web services Platform Architecture" by S. Weerawarana et al and "Perspectives on Web services" by Olaf Zimmermann et al for deeper coverage.
Concern:	*(none)*

Tags:

TYPE	CONTENT	
Scope	Operation	x
Phase	Macro design	x
SubjectArea	Tooling	x
Role	Integration architect	x
IBM-ID	N151262597-2145981281	x
TopicGroup	Layer3WebServiceBindingDecisions	x

Related Alternatives:

SOAP/HTTP (default)	SOAP over Hypertext Transfer Protocol (HTTP). Note that SOAP is not an acronym, but stands for itself. The service requestor invokes the service by transmitting SOAP messages over the HTTP protocol.	Decide
SOAP over reliable asynchronous transport, e.g. JMS or WS-RM	Java Messaging Service (JMS) as SOAP transport or Web Services ReliableMessaging (WS-RM) on top of plain SOAP/HTTP. The service requestor uses the implementation based on the Java Messaging Service (JMS) standard to asynchronously transfer the request to the service provider using the SOAP messages. Similarly, the service provider responds to the requestor utilizing the same mechanism. A different mechanism can be chosen for the response.	Decide
Platform-specific bindings such as local Java, RMI/IIOP (via WSIF or SCA), native MQ over IP, JCA, various host protocols/adapters such as CTG	WSIF is the Web Services Invocation Framework. SCA is the Service Component Architecture. Using this mechanism either the service requestor or an intermediary such as an ESB invokes the Service Provider implementation using a proprietary protocol and API provided by the implementing application. The service is exposed as a Web service and the messages are formatted as such. JCA or other adapter technologies can be utilized. The link from the JCA API to the enterprise system is called a resource adapter and can be provided by a third-party vendor.	Decide
Not applicable	*(none)*	Decide

Contact person: Marcin Nowak (Marcin.Nowak@usi.ch)

UROP 2010

A web interface for a novel direct manipulation development environment

Advisors: Prof. Michele Lanza & Fernando Olivero

Abstract:

The goal is to develop a web interface front-end to the Gaucho environment [1] we are building. Gaucho is development environment, that introduces a novel experience for developing Object-oriented programs. The user interface of Gaucho is based on direct manipulation and visual programming techniques, opposed to traditional text based integrated development environments. Gaucho is written in Smalltalk using the Pharo IDE [2].

The user interface of Gaucho is currently developed using a pixel based framework called Morphic, therefore we are missing the infrastructure to implement fundamental features, for example the possibility of zooming in and out of any portion of the system. A web based vectorial interface for Gaucho would allow us to greatly improve the current usability of the environment

The web user interface framework provides both the presentation and the connection layer to the underlying system. Since Gaucho is written in Smalltalk, we propose building the web front end using Seaside [3], a dynamic web development environment also written in Smalltalk.

References:

[1] http://gaucho.inf.usi.ch/
[2] http://www.pharo-project.org
[3] http://www.seaside.st/

UROP 2010

Understanding past development sessions

Student: TBD

Advisors: Prof. Michele Lanza & Lile Hattori

Abstract:

You start a new programming session. How long does it take you to remember what were you doing in the previous session? It actually depends on how long was the break between this and the previous coding session. And if the break is long enough, you might just not remember anything.

You are working on a team. How do you find out what were your colleagues coding in the last two weeks while you were away?

Imagine that you can replay step by step (Similar to Google Wave) the changes that you or your colleagues have done in a given time period as if you were watching a movie of a past development session. It will help you to recover your memory about what you were working on, to understand what your colleagues have been doing, even to spot changes that introduced defects on the program.

This is already possible. Syde [1] Replay is an Eclipse plug-in that replays past changes to a software system. The changes are collected as the developers work and are stored for future use. Currently the Replay feature of Syde uses the standard diff editor provided by Eclipse.

Your goal in this project is to enhance this plug-in to better support developers in understanding what changed between subsequent snapshots of a class.

Some of the enhancements we envision are:

- building a new editor to replay past changes and add intuitive visual cues (such as colors) to help developers to spot the changes;
- adding support for watching all the changes the team members made in a given period;
- adding support for watching how a set of classes changed in a given period;
- creating a report that summarizes the changes of a given period.

References:

[1] http://syde.inf.usi.ch/

Università
della
Svizzera
italiana

**Faculty
of Informatics**

"What's up?" – Building Reactive Public Displays that Facilitate Casual Encounters With Friends

A UROP Project proposal at the University of Lugano

Public display systems are increasingly becoming part of the urban landscape, with systems being deployed in venues such as railway stations, shopping malls and city squares. However, most public display systems are simple one-way communication channels enabling controllers of the system to broadcast content to the general public – perhaps with limited interaction via SMS.

These displays have the potential to bring spatially separated friends and colleagues together. Until now, chance encounters with friends meant running into them on the street, shop, café, or work, i.e., it implied physical co-location. But with proliferation of public displays, such chance encounters could occur in front of physically separate, yet virtually connected display "places".

The aim of this project is to develop a network of **reactive public displays** that are able to **detect "friends" in their vicinity** and **open up an ad-hoc audio-visual communications channel** for them.

At the outset, this project explores the use of passive IR sensing to detect when someone steps in front of the (e.g., wall-mounted) display and combines it with active Bluetooth scanning, to identify users based on the Bluetooth address of their mobile phone. The event of "encountering a friend" will be achieved through the *Spark* Instant Messaging (IM) client: IM friends will be detected near the display with their Bluetooth addresses and the system will open up a communication channel using an attached Webcam. While the initial prototype will only need to be able to link two fixed displays, the system should provide hooks for integrating a more generic display selection system in the future, e.g., based on the detected Bluetooth IDs at multiple display endpoints.

The system should be implemented on a Mac OS X, with a potential view towards cross-platform use (e.g., using a scripting language such as Python). Passive IR sensing will rely on the *BTnodes* sensor node platform, which is programmed in C.[1] The *Spark* IM client will be running through an *Openfire* server for "friends" detection, and a *red5* server will be used for the audio/video communication. The programming language for *Openfire*, *Spark*, and *red5* is Java. Audio/video connection for *red5* requires some programming in Flash.

Project Milestones

The project should address the following milestones:

1st step (2 weeks): The first and the most essential part of the project is movement detection and Bluetooth address acquisition. BTNode (or nodes) will be connected to a Mac Mini for movement detection. The built-in Bluetooth module on the Mac should be used to pro-actively acquire Bluetooth addresses and make them available upon movement detection.

[1] Fully functional code examples available.

2nd step (2 weeks): The second step involves integration between the BTNodes scanner system and the Spark IM client. Each user will have a Spark IM account with a Bluetooth device attached to it (i.e., the device's address). If two Spark "friends" are detected near the displays, the system shows a simple presence message on the screen, e.g., "Dave is at ETH" to John, and "John is at Lugano" to Dave, if Dave and John are Spark friends. The Spark IM clients will connect to an Openfire server.

3rd step (2 weeks): The last step is the integration of the BTNodes scanner system and the Spark IM Client with the red5 server, enabling audio and video communication between the displays.

4thstep (3 weeks): Once the system is developed, it should be deployed and tested for approximately 2 consecutive weeks. The deployment locations will be two research groups at the ETH Zurich[2] and the University of Lugano, respectively.

Project Agreement

The student hereby declares that he has read the above project description and agrees to the project topic outlined above. He agrees to meet regularly with his supervisor[3] and to inform him faithfully of his progress. After each meeting, the student will send a short summary with the topics discussed and the decisions taken to the supervisor.

The supervisor hereby declares that he will meet regularly with the student in order to provide concrete advice, to the best of his knowledge and within reasonable bounds. The supervisor will provide the students with the hardware and software necessary for the completion of the project.

Lugano,___________________________________
Date

___________________________________ ___________________________________
Student: Supervisor: Nemanja Memarovic

____
Professor: Marc Langheinrich

[2] No physical travel required.
[3] Typically weekly meetings, unless arranged otherwise with supervision.

UROP 2010

Title: Media Access Control (MAC) for e-puck robots
Supervisor: Alexander Förster
Assistant: Anna Förster / Frederick Ducatelle
Sponsor: Luca Gambardella / UROP
Keywords: MAC protocol, e-puck, micro-controller, programming language C, infra-red based, range and bearing
Prerequisites: Programming in C

Abstract: The aim of the project is to design and implement a MAC (Media Access Control) protocol for the infra-red based communication system for the e-puck robots.

The e-puck robots are small-size mobile robots, equipped with a infra-red communication system, called also range and bearing system. The robots can not only exchange packets through it, but also measure the distance and orientation to other robots.

The goal of this project is to implement a reliable MAC protocol to control the shared access to the infra-red communication medium by all robots. The main objectives of a MAC protocol are avoiding and handling of packet collisions and enabling high data traffic.

The e-puck robots have a well-designed programming and debugging interface (in C), which simplifies a lot the task. Extensive documentation, programming examples and a mailing list are available. The project will be under the supervision of Alexander Förster from IDSIA/USI and Anna Förster from USI. Together with them you will learn how to use the e-puck robots, how to program and debug them and will discuss the project details.

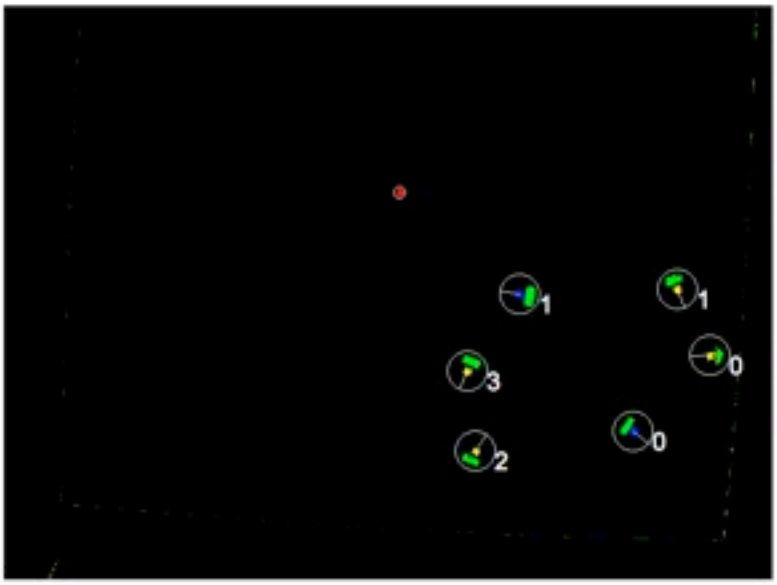
For more information, contact the advisors directly and/or check the following sources:

www.e-puck.org, http://www.rbzrobotdesign.com/epuck/
http://en.wikipedia.org/wiki/Media_Access_Control
http://zytrax.com/tech/wireless/802_mac.htm
http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Linux.Wireless.mac.html

UROP 2010

Title: Camera Grid for Robot Tracking
Supervisor: Alexander Förster
Sponsor: Luca Gambardella / UROP
Keywords: programming language C++, Qt-framework, Matlab, computer vision
Prerequisites: Programming in C/C++ and Matlab

Abstract: The aim of the project is to extend an existing real time color based object tracking program. The current version of the software is tracking objects in the image space of one camera. The system has to be extended in some ways: A post-processing step to convert the image coordinates to world coordinates has to be integrated into the real time process. Secondly, the software has to be extended to handle multiple cameras and information about the object positions have to exchanged between the camera tasks. Thirdly, feedback information from the objects has to be integrated into the system for a very robust tracker.



The current tracking system is programmed with the Qt framework from Nokia and the DC1394 library to handle FireWire cameras in a operating system independent way. The software was uses previously by many students for their different projects (master theses, undergraduate internships, and course labs).

For more information, contact the advisors directly and/or check the following sources:

http://qt.nokia.com/
http://damien.douxchamps.net/ieee1394/libdc1394/
http://www.idsia.ch/~alexander/robotlab.html

UROP 2010

Title: Sonic Based Robot Localization
Supervisor: Alexander Förster
Sponsor: Luca Gambardella / UROP
Keywords: e-puck, micro-controller, programming language C, robot localization, sound source recognition
Prerequisites: Programming in C

Abstract: The aim of the project is to evaluate localization capabilities bases on the sonic noise equipment of the e-puck robots. The environment is a 10m x 10m open space with two loudspeakers on side of the room. The robots are equipped with 3 microphones. A central computer can communicate with the robots and generates the sound signal for the loudspeaker. The localization task should be processed on the robot's digital signal processor and send over bluetooth connection to the central computer.

The e-puck robots are small-size mobile robots, equipped with a bunch of sensors, actuators and communication modules: infra-red proximity sensors, CMOS camera, loudspeaker, IR remote control, bluetooth.

The goal of this project is to implement a sonic based localization system with static emitters and moving receivers. The same idea is uses by GPS with electro-magnetic waves instead of sonic waves and by small robotic systems with ultra-sonic modules instead of signal in the human hearable range.

The e-puck robots have a well-designed programming and debugging interface (in C), which simplifies a lot the task. Extensive documentation, programming examples and a mailing list are available. You will learn how to use the e-puck robots, how to program and debug them.

For more information, contact the advisors directly and/or check the following sources:

www.e-puck.org
http://www.eurobot.org

UROP 2010

Computational kernels mapping and execution on a Coarse Grained Reconfigurable Array Template

Professor: Laura Pozzi.
Assistant: Giovanni Ansaloni.
Period: July 15th - August 31st

Abstract:
Coarse Grained Reconfigurable Arrays are a promising class of architectures conjugating the flexibility of reconfigurable hardware with high computational density. To investigate their performance, Hw/Sw Co-simulation framework has been implemented to study systems comprising a CGRA accelerator and a host processor.
Goal of the project is to analyze and map intensive loops, derived from benchmark applications, on instances of such systems to monitor their performance (data transfer, configuration and execution time).

Knowledge and Skills:
The project requires a good knowledge of VHDL and a basic knowledge of C. Previous experience with SOPCBuilder and Modelsim is a plus. During the project, the student will get accustomed to Hw-Sw Co-design and novel reconfigurable systems; he/she will leverage advanced features of Altera's system-level design suite.

Teaching

Teaching activities
Student projects

Program an Interpolation-Based Model Checker

Implementing a Model Checker often boils down to implement a reachability procedure: a property is satisfied by an automaton when it's negation is not reachable from a set of initial states. Reachability procedures can be implemented symbolically by means of encodings into SAT. A SAT-Solver can be then used to test whether a state is reachable or not.

In this project we will implement a Model Checker that uses a state-of-the-art reachability technique based on the notion of *interpolants*. The technique was introduced in the paper Interpolation and SAT-based Model Checking few years ago.

We provide an implementation of a decision procedure that already compute interpolants. The tool is OpenSMT, a decision procedure toolkit developed in our group. Therefore *the student should essentially implement the algorithm described at page 7* (of Interpolation and SAT-based Model Checking), assuming that OpenSMT is used for storing, checking, and interpolate formulae.

OpenSMT provides an API through which all these tasks can be easily set up. You are not required to know all the details about interpolation or SAT-Based Model Checking, as they require several weeks to be mastered; we only expect the implementation of a tool, and we will be happy to support the student not only on the "practical" but also on the "theoretical" side when things are not clear.

What we expect from a student:

- C/C++ experience;
- Being able to understand an algorithm and to turn it into practice;
- Willing to program a tool which should be tested for correctness and tuned for performance.

What we can give to a student:

- Knowledge in fields of decision procedures and formal methods;
- Practical application of theoretical notions into efficient, useful, and practical tools.

The project will be conducted under a supervision of the Lab members, mainly Roberto Bruttomesso and Aliaksei Tsitovich.

Contact: roberto.bruttomesso@usi.ch or aliaksei.tsitovich@usi.ch

The Lab is funded by Swiss National Science Foundation, Foundation Tasso, Hasler Foundation, EU-COST initiative and EU FP7.

Teaching

Teaching activities
Student projects

Paralleling OpenSMT Towards Cloud Computing

The success of OpenSMT, open-source SMT-solver born at USI two years ago, motivates us to explore new ideas in its development: now we want it to become parallel! Although, the general problem of paralleling of SAT- and SMT-computation is known to be very hard, we believe OpenSMT can do it and thus benefit from the modern cloud-computing resources. We propose two projects to USI students that target this challenge.

Project #1:

Recent work at Microsoft Research shows that the massive speedup can be achieved from running the same formula on slightly different versions of the solver (ManySat, Z3). Theoretically not very efficient, the approach benefits from the fact that the solver instances *share* the facts they learned during computation. Student is expected to implement clause sharing in OpenSMT and investigate the speedup and the overhead of this sharing. Besides the infrastructure to run multiple instances of the solver with different search strategies should be established.

Project #2:

We propose to explore existing computation-intensive parts of the OpenSMT algorithms, e.g. linear rational arithmetic solver (LRA), from the paralleling point of view. Various instructions of the algorithm are not limited with sequential ordering and can be executed simultaneously. The student is expected to build and implement the parallel version of the algorithm (for multi-core and multi-node running environments) and investigate the speedup and the overhead of it.

Background:
OpenSMT is a compact and open-source SMT-solver written in C++ with the main goal of making SMT-solvers easy to understand and use as a computational engine of formal verification (more details are available at the OpenSMT project page).

What we expect from a student:

- C/C++ experience (OpenMP/MPI experience is a big advantage);
- Willing to learn and understand new frontiers;
- Passion to open-source development.

What we can give to a student:

- Experience in C++ development for clouds/multi-core machines;
- Understanding of how complex industrial problems are solved with "state-of-the-art" research techniques;
- Knowledge in fields of decision procedures and formal methods.

Both projects will be conducted under a supervision of the Lab members, mainly Roberto Bruttomesso and Aliaksei Tsitovich. Don't think that it is too complex. Yes, it is, but we will help you to "capture this flag".

Contact: aliaksei.tsitovich@usi.ch

The Lab is funded by Swiss National Science Foundation, Foundation Tasso, Hasler Foundation, EU-COST initiative and EU FP7.

Università
della
Svizzera
italiana

Faculty
of Informatics

**Institute of
Computational
Science
ICS**

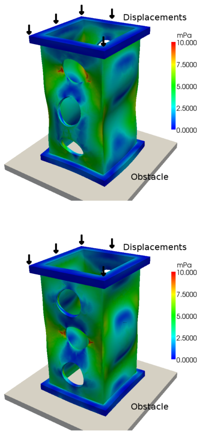
State of the Art Solution of Nonlinear Programming Problems

Prof. Dr. Rolf Krause, Dr. Christian Groß

1 Project Description

The solution of nonlinear programming problems is of crucial importance in various applications, e.g. for the simulation of the behavior of soft-tissues, large-deformations or for shape optimization. Nonlinear programming problems are optimization problems of the kind: find an $x \in \mathbb{R}^n$ such that $f(x) = \min!$ for some, possibly nonconvex, objective function f. As the objective function might be (locally) nonconvex the solution of such an optimization problem becomes very complicated since the well-known Newton-Raphson method might not converge to the solution.

Therefore, one must think about alternative methods to solve nonlinear programming problems. Beginning from the 1950's two standard methods called Trust-Region and Linesearch methods were developed, which are in theory – but also in practice – able to find a local solution of problems like $f(x) = \min!$. Furthermore, in the last decades more efficient variants of these strategies have been developed and investigated where one of these will stand in the focus of this research project, the so-called *adaptive cubic overestimation method.*



The computation of these deformations was carried out by solving nonlinear programming problems.

In the first part of this project, the applicant will implement the standard Trust-Region method in Matlab. Then, in the second part, we use the knowledge won during this implementation process to also implement the adaptive cubic overestimation algorithm in Matlab. In the third part of the project we will compare the behavior of both methods for the solution of more complicated nonlinear programming problems.

2 Applicant's Skills

- Due to the character of this project, the student must have good programming and perhaps testing skills.
- It is of importance that the applicant has a reasonable understanding of Analysis and Linear Algebra. This is due to the fact that understanding the underlying numerical analysis will accelerate the implementation. In particular, it will help the student to trace errors, understand the algorithms and to successfully implement these.

Università
della
Svizzera
italiana

**Faculty
of Informatics**

Meex – Assessing the impact of user contributed fingerprinting to interval labeling based wifi positioning.

Location fingerprinting has many advantages when it comes to indoor localization, as it can leverage existing infrastructures. However, the main problem of location fingerprinting is that it requires the collection of (large) sets of fingerprints, which is time consuming and costly. One way to overcome this problem is using a crowdsourcing approach, i.e., the users themselves can add fingerprints to the system at any time, continuously training the system as they are using it.

In order to improve accuracy, recent research has investigated the use of "interval measurements" instead of just instants, i.e., instead of just taking one measurement at a certain time, the system may continue recording measurements for hours. However, in a crowdsourcing system, this still requires users to actively provide fingerprints to the system, albeit in a slightly different mode.

In previous work, we developed an App for the iPhone called 'Meex'. Meex is a chat application that allows users to virtually check-in to their current location and find other users who are checked-in nearby, a concept similar Apps like 'Foursquare' or 'Gowalla'. When users check-in, they have to confirm their current location. Hence, Meex can take WiFi measurements in the background and update the model used for positioning. This data is then used to train a Support Vector Machine (SVM). Although all of the above software components have been developed, the long-term effect of using an application like Meex on the accuracy has not been analyzed.

The goal of this project is twofolds. First we want to conduct a user study to investigate how people actually use an application like Meex. Second, using the data from the user study, we want to examine the effect of adding a large number of WiFi measurements to the accuracy of the underlying SVM.

For more information visit http://www.repdpin.org -- which is the underlying fingerprinting engine used in the project, and/or contact Marc Langheinrich.