# Signature Validation Portal

Organization: DFKI
Contact: Roland Vogt

While the technologies for generation of electronic signatures are well-established in public-key infrastructures, the validation of electronic signatures still shows some open problems. The verification of the validity of an electronic signature requires high degrees of diligence, time and effort for the local configuration and management of trusted certificates and their revocation status. A Signature Validation Portal should be designed and implemented which enables the effective verification of the validity of (qualified) certificates issued within several public-key infrastructures. It manages a collection of trusted (root) certificates and responds to validation requests by building and validating a path of certificates in due consideration of their current revocation status.

SCVP

OCSP

PKI

Trusted Certificates
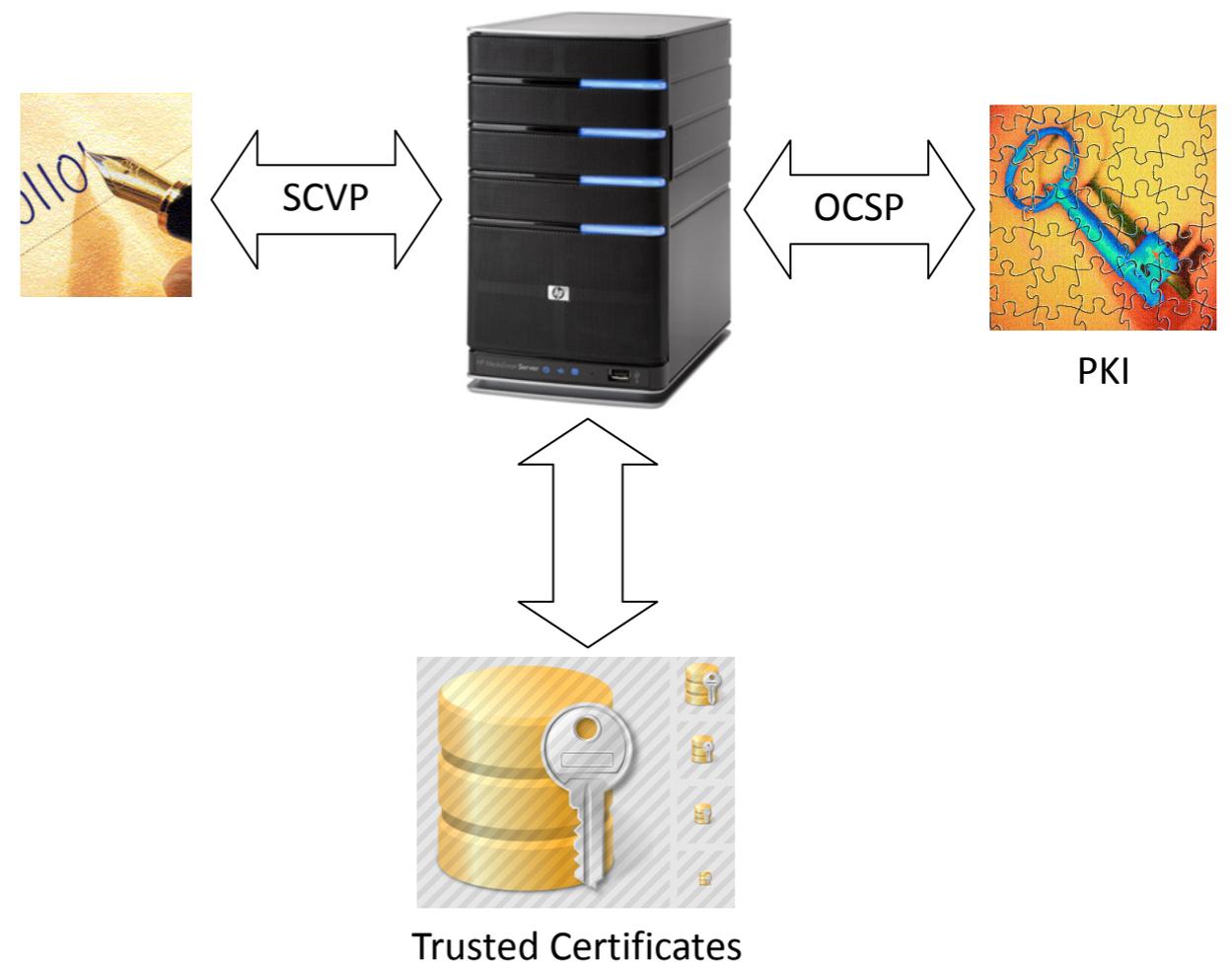
Parsing and Manipulating LaTeX Documents

Organization: MPI
Contact: Timo Kötzing

The main aim of this project is to read in a LaTeX document, and generate an intermediate representation. This representation of the document should allow for manipulations of the document and for being written back to LaTeX code. Possible manipulations are: extraction of document structure, identification and removal of unimportant document parts and automatic cross-referencing.

User-friendly process algebra compiler

Organization: Dependable Systems and Software
Contact: Christian Eisentraut

This project aims intends to extend an existing tool that currently allows to visualize and analyse concurrent processes, specified in the Calculus of Concurrent Systems (CCS). In the extension the user should be able to specify completely new process behaviour by introducing new language constants and operators, together with their semantics. The semantics of the extensions will be provided as a Structural Operational Semantics (SOS). This means that process behaviour is expressed by syntactical transformations of the syntactical process description. In the extension, the user should be able to concentrate on designing new rules with ease, supported by a graphical user interface, which allows to enter rules as if written on plain paper. When a new primitive has been defined in this way, it should be immediately usable in the tool as if it was a built-in CCS expression.

Automata Modeling Framework

Organization: Reactive Systems Group
Contact: Markus Rabe

We want an eclipse-based IDE for specifications that allows to easily specify different types of automata and logics. It mainly targets at modeling labeled transition systems, timed automata and logics like LTL, but it should be easily extendable to other types of specifications. The framework shall offer syntax highlighting for logics, and automatic graph-layout for automata and most importantly a well documented interface to other tools like model checkers. There are already several graphical user interfaces for modeling automata (e.g. UPPAAL, Syspect), which are, however, restricted to specific automata classes. We, in contrast, would like to have an extendable and open platform that is not only built for a specific purpose. As an advanced feature, we would like to have a simple script language for batch processing and easy creation of larger models. This project is well suited to be continued as a thesis project or a masters' practical training and we would be enthusiastic to use this framework for our own research.
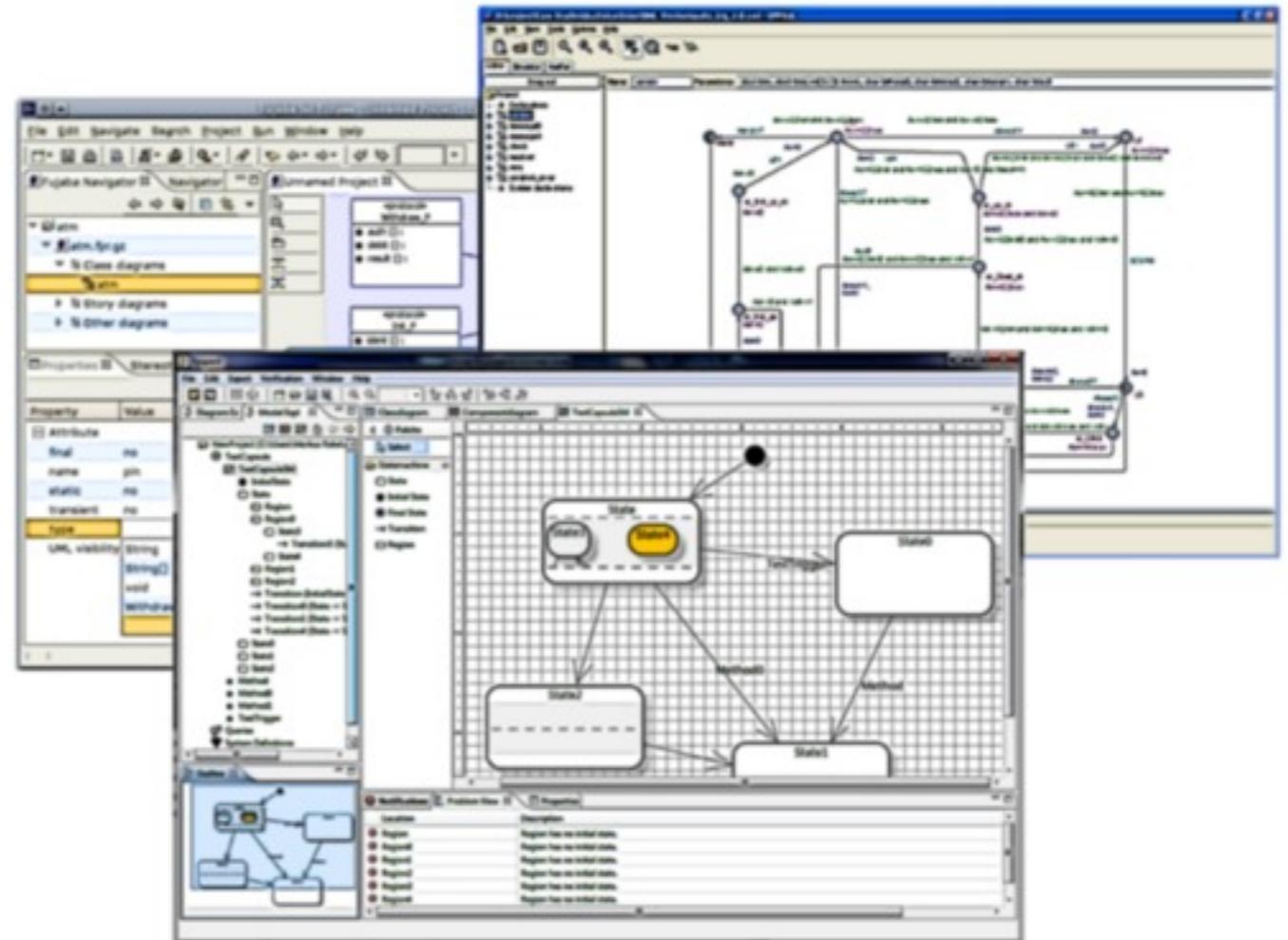
Management Tool for Maintaining Work Package Descriptions

Organization: DFKI
Contact: Alexander Kröner

The definition of interconnected work packages is crucial for setting up a work plan, as it is needed for instance for research proposals. Usually, people from several project partners work at the same time on the actual work package descriptions. Since a complex project easily comprises 100 work packages, establishing relationships between the respective documents can become a tedious task. This should be addressed by a tool, which supports the collaboration of authors in this task.

Intelligent Avatar in Jadex for Virtual 3D Worlds

Organization: DFKI
Contact: Stefan Warwas

Our research group developed the first open research platform ISReal (Intelligent Simulation of Realities) for the 3D Internet, in particular the 3D Web encompassing all virtual 3D worlds that are accessible via the web. These virtual worlds are often inhabited by simple types of avatars acting as alter egos of their users who are exclusively driving their behavior in the 3D scenes. In ISReal, virtual worlds are different: First, they are rather inhabited with intelligent avatars that are avatars driven by intelligent software agents capable of proactive, reactive and autonomous behavior such as by means of goal-driven planning and learning. Second, the semantics of 3D scene objects are described by means of semantic Web technology including formally defined semantic concepts and (animation) services like those for opening and closing of a door associated with a displayed 3D object door. Each intelligent avatar possesses one individual sensor enabling it to perceive objects in the scene and is capable of reasoning upon the object semantics to not only understand its environment but to plan its actions accordingly. A first prototype of the architecture of such an intelligent avatar agent has been implemented for the prominent BDI (Belief-Desire- Intention) agent architecture-based development platform Jack. Your task will be to realize the same BDI agent architecture for the competing multi-agent development platform Jadex (Java, XML) and to validate your solution by means of an avatar that is driven by your intelligent Jadex agent in some use case scenario running on our 3D-Internet research platform ISReal. When working on this task you will get in touch with the exciting technology areas of intelligent agents, semantic Web and 3D Web – and hopefully see your intelligent agent-driven avatar successfully perform in virtual 3D scenes displayed in a given 3D web browser (XML3D-compliant Google Chrome or Firefox).

GUI for Modeling, Simulation, and Verification of Hybrid Automata
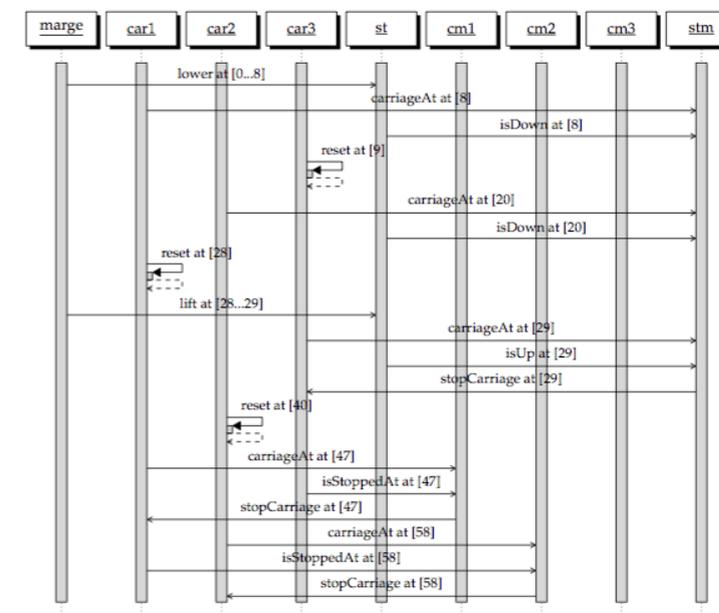
Organization: DFKI
Contact: Christopher Krauß

In our work considering the verification of embedded systems we use the theory of hybrid automata as a formal modeling language. For HAVLE (Hybrid Automata Verification by Location Elimination), the verification tool developed in our group, we now need a suitable graphical user interface with functionality similar to comparable tools (e.g. Uppaal which works with a simpler kind of hybrid automata, so called timed automata). More precisely, the GUI should allow us to graphically model hybrid automata (states, transitions, ...), to enter properties for verification, and to visualize possible behaviors of the modeled system (e.g. as sequence charts). For the visualization of possible behaviors we will also need an API that allows us to construct the components of the diagram during a verifier run. As a result of this project we would like to have a set of eclipse plug-ins (or even better an eclipse RCP application) comprising several views for the different tasks.

Graphical User Interface for Modeling
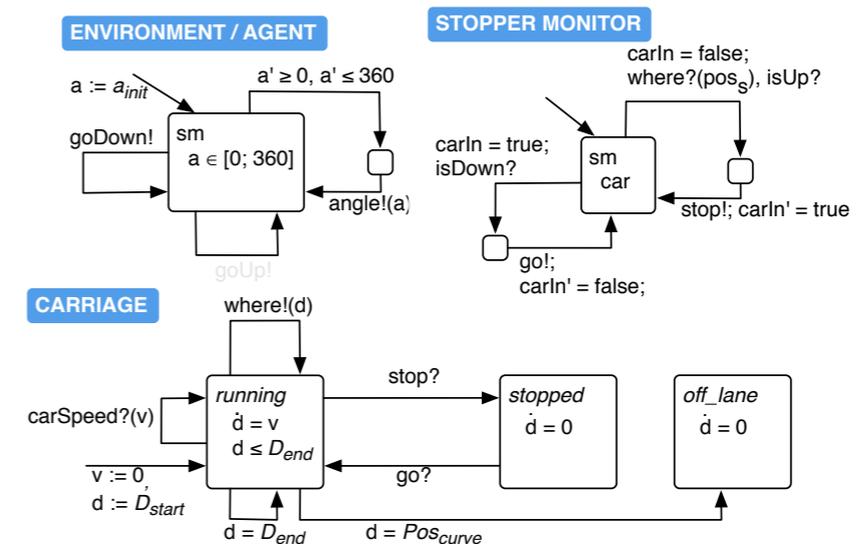
Visualization of Automata Behavior

Tool for Monitoring Resource Consumption in a Smart Home

Organization: DFKI
Contact: Alexander Kröner

Via sensor technology, so-called Smart Homes can capture data about the resource consumption (e.g., water, energy) of their inhabitants. The visualization of such data in a way that can be understood by consumers is often non-trivial task. Furthermore, the sensing capabilities of such an environment may change, e.g., due to the installation of new equipment. This should be addressed by a tool, which supports users in the exploration of resource-related data from several sources, and in adding and removing such sources.

Towards an Agent-Supported Negotiation Framework

Organization: DFKI
Contact: Ingo Zinnikus

Negotiation is in many cases an unstructured process that runs in an ad-hoc manner that is mainly driven by the intuition of the participants. However, in theory communication protocols, which are the basis of negotiation process, are intensively investigated. Up to now there is little support to run such negotiation processes in a computer-aided manner. Our research group at DFKI has largely contributed to the design of mechanisms for game-theoretic negotiation and is currently working toward a framework for such a computer- aided negotiation system based on agent technologies. In particular, we developed a model driven domain specific language for multi-agent systems (DSML4MAS) which builds on the Eclipse Modeling Framework (EMF/GMF). DSML4MAS comes with a modeling environment that is separated into different views. One important view is the so-called Interaction View which includes protocol design: What makes this view special is that it actually disappears when the models are transformed into executable code (at least in the approach that we use for now). You should have this overall picture in mind while investigating model to model transformations of the interaction part of given concrete models. The result of this transformation is a skeleton model that needs to be further integrated with the rest of the model for the multi-agent system. There are two frameworks that support the definition of model to model transformations (declarative ATL and procedural QVT). Both approaches should be investigated and the results compared regarding different criteria like for example readability of code, efficiency etc. If work progresses well, students can progress towards a model to text transformation that transforms the result of the model to model transformations into executable code (at least skeletons for executable code) for a specific execution environment that is already partially implemented.

Mining Software Repositories

Organization: Software Engineering Chair
Contact: Kim Herzig

Software archives contain many valuable information about software projects and the development process of software development teams. There exists a large research community mining and exploring these data sources to develop tools and techniques helping software developers as well as managers to understand flaws in the development of software and to improve future process steps. The software engineering chair at Saarland University is part of this research community and develops strong analysis and recommendation tools helping to detect low quality software artifacts and their corresponding reasons. Currently, the chair is refactoring and merging many of it's tools into a single framework.

Printerface

Organization: Student Council
Contact: Nikolai Knopp

The Fachschaft currently provides students with a printer where they can print a certain amount of pages per month from CIP computers or via the network (intended for lecture slides, readings etc.). The system should be extended such that lecturers can incrementally upload their lecture material to the system, enabling the students to choose and print lecture material directly using the touchscreen next to the printer which currently runs the "Printerface" application. Care must be taken to make the interaction with the system reasonably easy for both students and lecturers, as the success of the extension depends on whether lecturers upload their material or not. Access to the current Printerface sources will be provided.

Improving machine translation using comparable corpora

Organization: DFKI
Contact: Jia Xu / Sabine Hunsicker

In the last decade, increasing attention and incremental improvements have been shown in machine translation that investigates automatic translation of text or speech from one natural language to another. Currently statistical machine translation is a data-driven process, where the quality and quantity of training data is crucial for the system performance. However linguistic resources involve human effort that are expensive and limited. In this project we will employ data mining approaches to build a "data-robot" that automatically detects, crawls multiple language texts from the web and classifies them into different domains. The obtained large amount of comparable corpora will be evaluated in our state-of-the-art translation systems.

Sign Language Translation

Organization: DFKI
Contact: Michael Kipp

In Germany, 80000 people are deaf. 80% of deaf children leave school with considerable reading/writing problems so that written communication (e-mail, internet pages) can be a challenge. Therefore, automatic translation systems with sign language avatars may significantly improve the situation of the deaf. The core of the translation problem is the transition from a written German sentence to a written sign language representation in the form of "glosses" (discrete single signs). The task for this project is to write a "shallow" translator: Given a parallel corpus of German sentences and corresponding gloss sequences, find the simplest possible transformation methods to transform German to glosses. The system can be connected to existing avatar systems to visualize the results. The outcome prototype could become part of a bigger system where simple cases are handled with the shallow approach while more complex sentences are passed on to higher-level systems.

Hast du deine **Abschlussprüfung auch hier** im **Betrieb oder woanders** gemacht ?

FRAGE ABSCHLUSS-PRÜFUNG AUCH BETRIEB HIER ODER WOANDERS

FRAGE   BETRIEB   HIER   WOANDERS

Simulate the process of learning the meaning of words from input scenes

Organization: Computational Linguistics and Phonetics
Contact: Afra Alishahi

What is needed is a visual environment, where a sequence of input scenes can be specified as input to the learner. Each scene consists of a sentence (or a list of words), and a collection of images (objects or activities). Over time, the system forms an association between each word and the potential meanings for that word (for example, as a probability distribution over all the objects that have co-occured with that word so far). The exact learning algorithm is not the key here, in fact this system can be used for comparing different algorithms.

Planning Support of Semantic Service-Oriented Business Process Implementation

Organization: DFKI
Contact: Patrick Kapahnke

Service-oriented realization of complex business processes (in service-oriented architectures, SOA) in terms of their orchestration in service workflows still remains a rather tedious hand-crafted work. Each business process may be realized by one or more appropriate Web services (REST or SOAP-based). We focus on SOAP-based web services each of which functionality is described in standard XML-based WSDL (Web service description language) in terms of its operation signature and possibly some mere textual descriptions of its functionality. On the other hand, the semantic Web technology provides you with means of formal ontology-based semantic annotation of Web services that transform them into so-called semantic Web services. Point is that these semantic services can be automatically composed into sequential service plans - hence into workflows of business processes these services are associated with to implement - by means of AI-based semantic service composition planners. Our group has developed a variety of tools for semantic service selection and composition planning including the composition planner OWLS-XPlan.

Building a Semantic Parser for Natural Language Processing

Organization: Computational Linguistics and Phonetics
Contact: Caroline Sporleder

Computing the meaning of a text is one of the biggest challenges in natural language processing. One important subtask involves computing the semantic argument structure of sentences, i.e., determining who did what when to whom. This task is known as "semantic parsing". While there are a number of semantic parsers around, these are limited in that they have a very local focus, i.e., they compute the argument structure of specific verbs in a given sentence rather than that of all words in a text. This has the disadvantage that context effects are not taken into account, which reduces the performance of these tools as well as the usefulness of the output. The aim of this project is to develop a semantic parser that can parse whole texts, take context into account and is able to link local argument structures into a unified representation of text meaning.
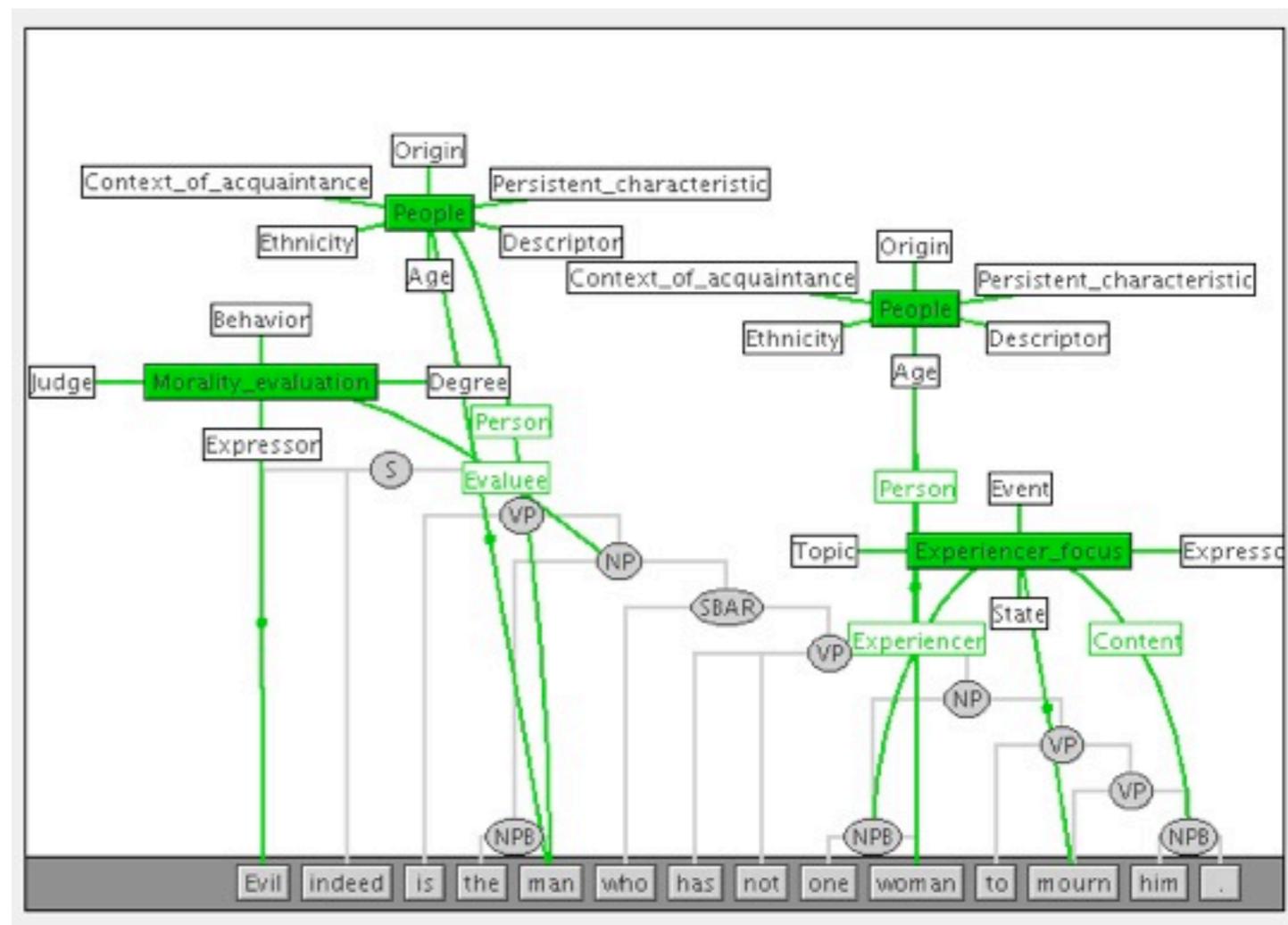
Mutation Testing in Eclipse

Organization: Software Engineering Chair
Contact: David Schuler

Mutation testing seeds artificial defects (mutations) into a program in order to measure the quality of a test suite, and provide clues to improve it. In contrast to traditional coverage metrics mutation testing also judges the quality of oracles used in a test suite. Your task involves the integration of an existing mutation testing tool into the Eclipse-IDE. Amongst others this involves answering the following questions: How to present the mutation testing results to the user? How to detect and track equivalent mutants and mutants that should be ignored? How to efficiently update the results after code changes (e.g by using data flow analysis)?

Touch-screen solution for coffee accounting

Organization: Software Engineering Chair
Contact: Clemens Hammacher

At the software engineering chair, there are a lot of people drinking coffee, and exactly one coffee machine. We established a paper-based accounting system, where people buy a "coffee card" with a fixed amount of check boxes, and they have to strike out one box per coffee, two per cappuccino or similar. Additionally, there is a bonus system for those who clean the coffee machine, and there is again a piece of paper where they have to sign. The problem with this solution is that the whole manual bookkeeping is very error-prone. The solution that we have in mind is a fully digital solution, which is integrated into the existing chair's LDAP system. The system should be controlled via a touch screen close to the coffee machine. The machine provides a serial interface where an accounting system can be connected. Unfortunately, the protocols used are not publicly available.

SPASS-YAGO Web Frontend

Organization: MPI
Contact: Patrick Wischnewski

SPASS-YAGO is based on the automated theorem prover SPASS and provides a reasoning engine that is in particular designed for the YAGO ontology. The YAGO ontology is a knowledge base consisting of 20 million facts extracted from Wikipedia. An example of a fact from YAGO is: "Angela Merkel is born in Hamburg." SPASS-YAGO can answer sophisticated queries using the knowledge from YAGO. For example, it can answer the following query in 0.6 seconds: "Who is born in the same city where all his children are born?" This query cannot easily be answered using conventional search engines à la Google. In order to make SPASS-YAGO available to the public we would like to have a web frontend and a version of SPASS-YAGO that can communicate with the frontend and can handle queries concurrently.

SPASS–YAGO

YAGO ⟺ SPASS — YOUR PROJECT

Who is politician and physicist and born in Germany?

Who is born in the same city where all his children are born?
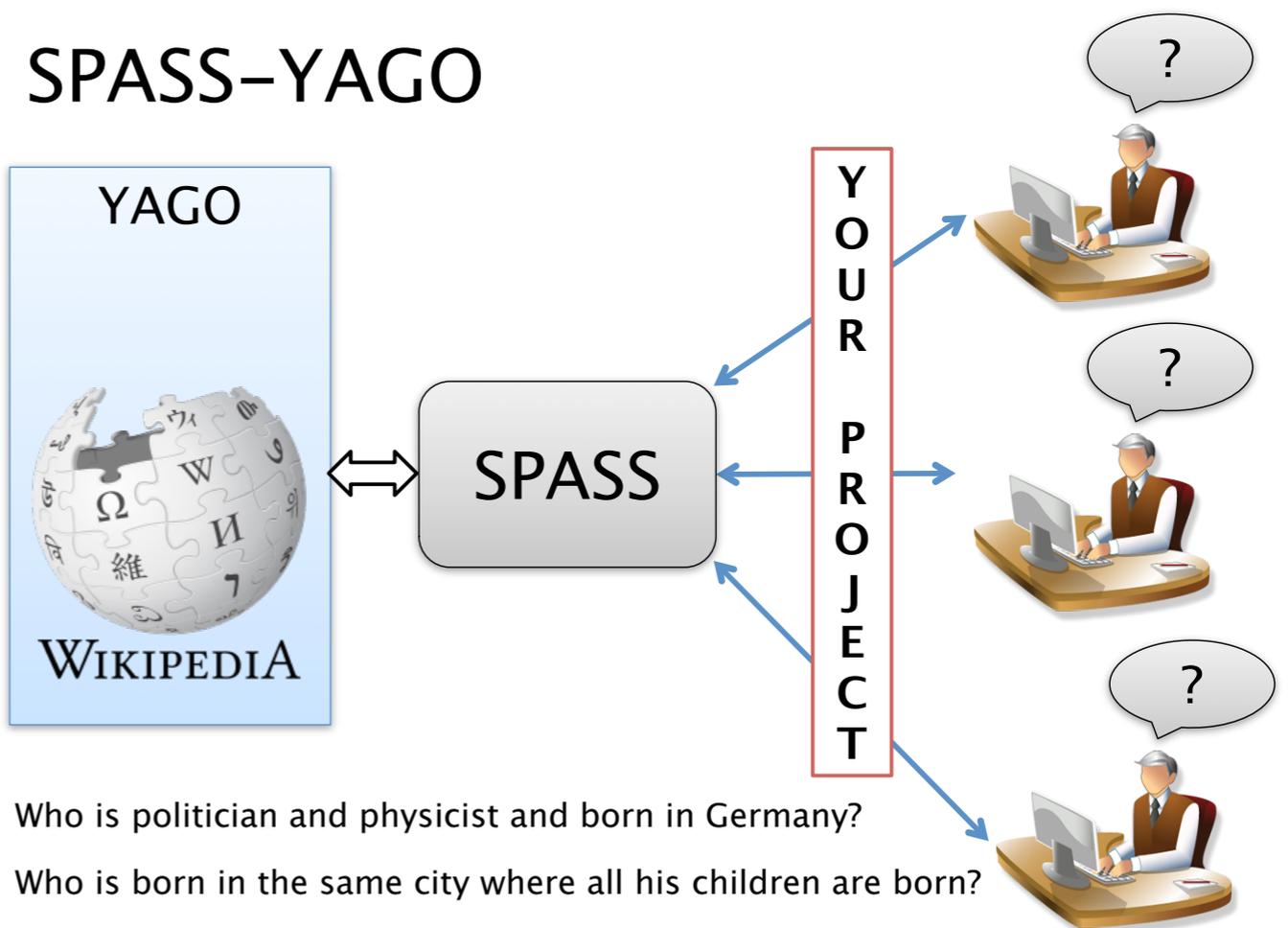
Visualizing Exam Distributions

Organization: English Department of Saarland University
Contact: Hank Rademacher

It is often helpful to see what happens to the so-called "bell curve" when one alters the thresholds between grades on a major test. For example, what will happen to the distribution if one moves the score needed for an "1,0" down by two points? This app allows instructors to enter scores for a major exam and set the borders between the grades on an instructor-defined scale, for example, 1,0 / 1,3 / 1,7 / 2,0 and so on. A graphical user interface allows the instructor to see "live" how changing these borders affects the distribution. Statistics such as mean, median, max and min are of course part of the readout as well.

Planning and Capacities: Bachelor, Masters, Lehramt

Organization: English Department of Saarland University
Contact: Hank Rademacher

This app allows instructors/planners/advisers to visualize the progress of large cohorts of students through a set of academic programs (courses of study) in order to better plan how many of each course to offer in each semester. The app takes into account (1) the number of students in each program (Bachelor, Bachelor Nebenfach, Lehramt, Masters, etc. and their particular semester; (2) the modules and "Wahlpflichtelemente" that students in each of these programs have to complete; (3) the number of CPs involved; (4) the range of time during which students are required to complete them ("Regelstudiensemester"); and the prerequisites involved for any courses that have them. The app allows instructors/planners/advisers to better visualize the modules and the students moving through them, and produces readouts on how many staff members are needed in any given semester for any given course. The app is dynamic in the sense that one can add and remove students at various stages of the process. In other words, one can enter up-to-date student statistics at all levels and receive in return a "dynamic flow chart" of sorts.

Myxobase Explorer

Organization: Department of Pharmaceutical Biotechnology
Contact: Rachana Gundluru

To develop a database with a GUI providing comprehensive information of biological compounds (including basic compound property description, information of the producing organisms, experimental analytical information, corresponding scientific literature, etc...) The main task is organizing a big amount of information into relational database with all necessary linkages and the design of a well functioning graphic user interface. The database should be a repository for a fast look up of all the relevant data of a compound including intelligent storage of data and effective search options. Efficient input and retrieve of the data with a user friendly interface is to be developed for the effective functioning and usability of the database.

Framework to integrate different multi-touch applications

Organization: DFKI
Contact: Johannes Schöning

The goal of the project is the planning / development of a framework that enables the integration of various different multi-touch applications that are based on different techniques (Flash, TUIO, Python-applications). On the one hand, a graphical interface should be designed that gives the possibility to choose among the integrated applications and then triggers the execution of an application that handles the camera data (and derived information from it, e.g. blobs, touch coordinates) and sends this data in the appropriate form to the 'external' application. Therefore, it is necessary to examine which ways of communication are possible and needed and in which way the raw data has to be processed. Furthermore, it is necessary to get in touch with standardized protocols like TUIO to enable a corresponding realization. Besides, there is the need to implement an application starter that enables the use of applications that are written in Python by using the multi-touch capable library 'LibAVG'.

Raw camera data / touch points → Framework-Application → Multitouch-capable Application

Multitouch-Application (Flash)

Multitouch-Application (TUIO)

Multitouch-Application (libAVG)
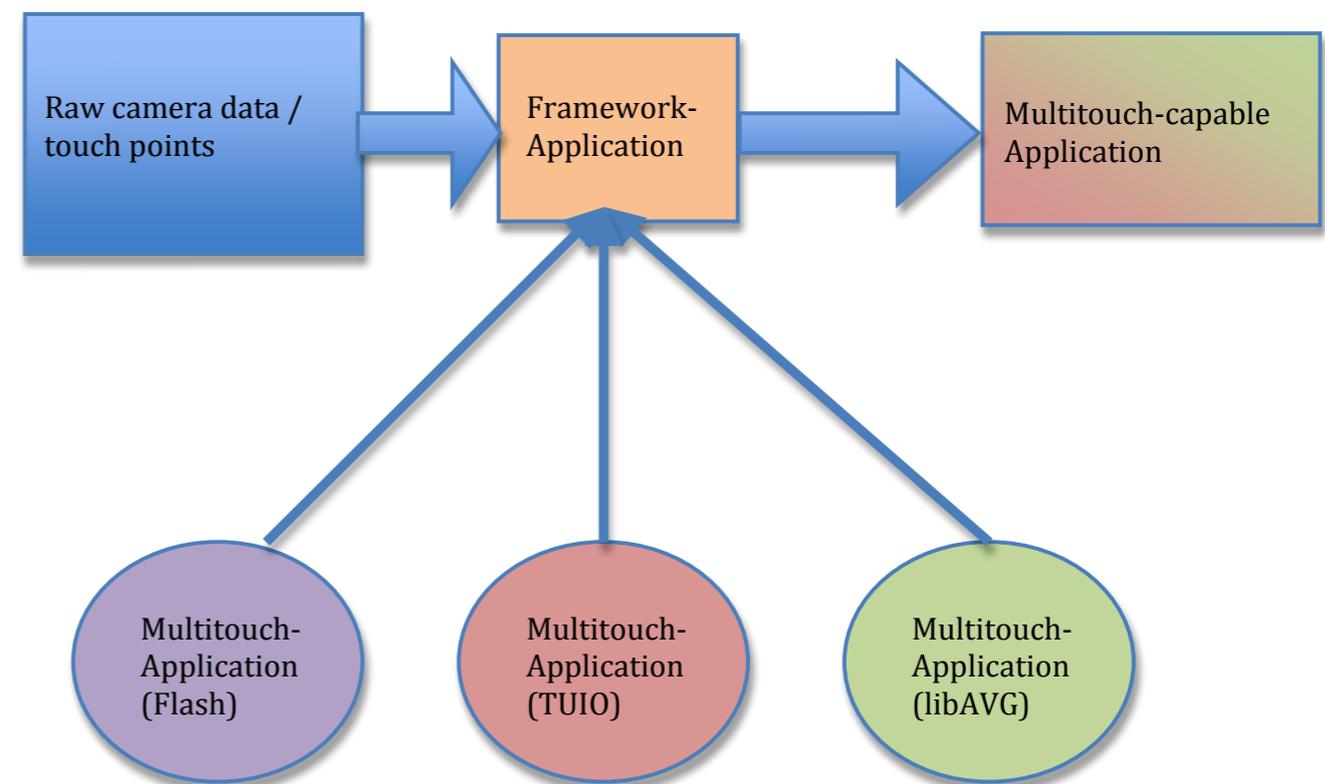
Library Management System (LMS)

Organization: DFKI
Contact: Constantin Houy

The LMS must enable the management of library book collection. LMS users must be able to search for books by criteria like title and author and have the possibility of lending them. There should be some sort of control panel by which user accounts and book collection can be managed. The LMS user interface should be web-based. If the prototype will be satisfactory it will replace existing system. Because of that, it is rather expected that the system won't have many advanced functions but will be fully implemented.

Visio-Online

Organization: MPI-SWS
Contact: Rupak Majumdar

I would like to have a library to have a Microsoft-Visio like interface on the web. Currently, there are some Java libraries which could in principle be used as applets, but they are quite slow. I think this can be done through javascript and html5 in the browser, and use json as a backend representation. I would like the API to support specific components (just as Visio supports libraries for circuits, etc); for example, I would like support for drawing Simulink-like block diagrams. I think such an interface will be quite useful in a number of applications: my interest is in having a visual language front end for describing systems and components for a verification backend.

ProjectTracker

Organization: MPI-SWS
Contact: Rupak Majumdar

I would like to have a software engineering toolbox that connects requirements, design, engineering, and verification tasks. Currently, feature requests for software are mostly kept in Excel sheets. These become tasks in some intermediate tool (probably a tool to manage iterations, such as Rallydev in the agile world), and at some point, moved over to a ticket on trac for developers. A separate tool may be used for managing tests. I would like to have a tool which integrates feature requests, trac, and test management, and more importantly, have a notion of provenance that is tracked in the system, so that any particular engineering task can be traced back to the customer requests that initiated it, and for any feature request, there is a way to track what tests test the functionality for the request. There are other applications of the provenance information, e.g., to track how customer requests move through the system. The system could start as a trac plugin, but might require a more thorough redesign as well as UI changes. (Customer feature requests are usually uploaded by sales and marketing, who might not like the same user interface as developers).

Computer science paper corpus and concordancer software

Organization: MPI-SWS
Contact: Rose Hoberman

I would like to develop software to construct a corpus of well-written computer science papers, as well as concordancing software that students could use to analyze the corpus. The software would have to extract all text from PDF, PS, or latex files and properly tag headings, math, and figures. The software would then need to part-of-speech tag the text, build an index, and then use that index to provide functions for searching and analyzing the corpus. The standard KWIC (key word in context) search would be useful, but I can imagine many other types of searches and analyses that would also be helpful for students. Of particular interest would be analyses to compare statistical properties of a student's own paper with the papers in the corpus. An easy-to-use interface and powerful and efficient search capabilities (including regular expressions and part-of-speech tags) would be essential.

Concordances for *determined* with associated = 10

```
1   f the staff.     Staff availability should be determined by the work to be done.  The most rigorous ef
2   , who had  arrived with a strong squad and a determined look in their eyes.  The Liverpudlians should
3   wedish super star has arrived  in Suffolk in determined mood.     One of the first top riders to sign
4   ced pleating  arrangement because of the pre-determined size of the pleats,  thereby leaving a larger
5   ratic venture  but, at the time, it made him determined to escape from Auden's  artistic and personal
6   ia as an early Christmas present  which I am  determined to keep healthy. Any handy tips for good,  st
7   ried for fouling up on the big occasion was  determined to make no mistake this time, racing to the f
8    jealous  of Osiris's power and prestige and determined to seize the throne  for himself. When Osiris
9    the larger  theatres. But, of course, I was determined to take him into the  company and his first r
10  and payments on account.    If cost has been determined using FIFO, LIFO, weighted average  cost or a
```

Right collocates for 'determined'

to	5
look	1
mood	1
size	1
by	1

WordNet entries for determined

Visual Logging Framework

Organization: Programming Systems Lab
Contact: Sigurd Schneider

The purpose of this project is to build an Eclipse plug-in for Java developers that aids them in setting up industrial strength logging. The vision is that logging is set up via a GUI interface that lets the user define logging logic in terms of primitives called sections and schemes. A section is a block of code (including a user-defined name), for which entry and exit (and possibly parameters and locals) get logged. A scheme is associated with a class, and defines for every method calls, parameters and fields should be logged. The events in the log must allow unique identification of the executing thread and all the objects they are referring to. The GUI should offer an option to hide/show all logging logic such that the code is not cluttered. Finally, logging must work out of the box given the definitions of sections and schemes.

Hochschulsport Team Organizer

Organization: Hochschulsport
Contact: Jörg Schad

Many activities organized by the Hochschulsport consist of more or less fixed teams. So it would be nice to have a platform where teammates can exchange ideas, organizes tournaments and other event. Other ideas might include a training management (like freizeitmanager) where training sessions could be organized (e.g. team mates can register for a session and hence the coach can consider the group size before training).

Improving the security and the data privacy of social networks

Organization: Language-Based Security group
Contact: Kim Pecina

We have a project on improving the security and the data privacy of social networks, namely we provide fully anonymous authentication and pseudonymous authentication besides the usual friend and friend-of-a-friend mechanisms native to Facebook. A very basic prototype implementation is already in place but it needs to be redesigned, a better integration with Facebook, and a user-friendly graphical user interface. If possible, we would also like to investigate the option to fully decentralize the social network to a peer-to-peer architecture. A background in cryptography is helpful but not necessary as we provide all the required cryptographic primitives for this project.

Transform old 'List & Label' designs into modern jasperreports!

Organization: Agfa Healthcare
Contact: Stefan Großmann

ORBIS is the Clinical Information System (CIS) of Agfa Healthcare that integrates all administrative and clinical information in health care facilities. It is used by more than 400,000 people every day and worldwide. Some old modules in ORBIS are still using List & Label to print patient specific reports, which have been created in an integrated designer. Most of these reports are additionally customized by the customers to satisfy their individual needs. Currently the reporting platform in ORBIS is based on the open source framework Jasperreport. The Goal of the project should be a concept that shows whether it is possible to convert the old List & Label reports into the jrxml design of Jasperreports. As a proof of concept, a prototype should be developed in addition.
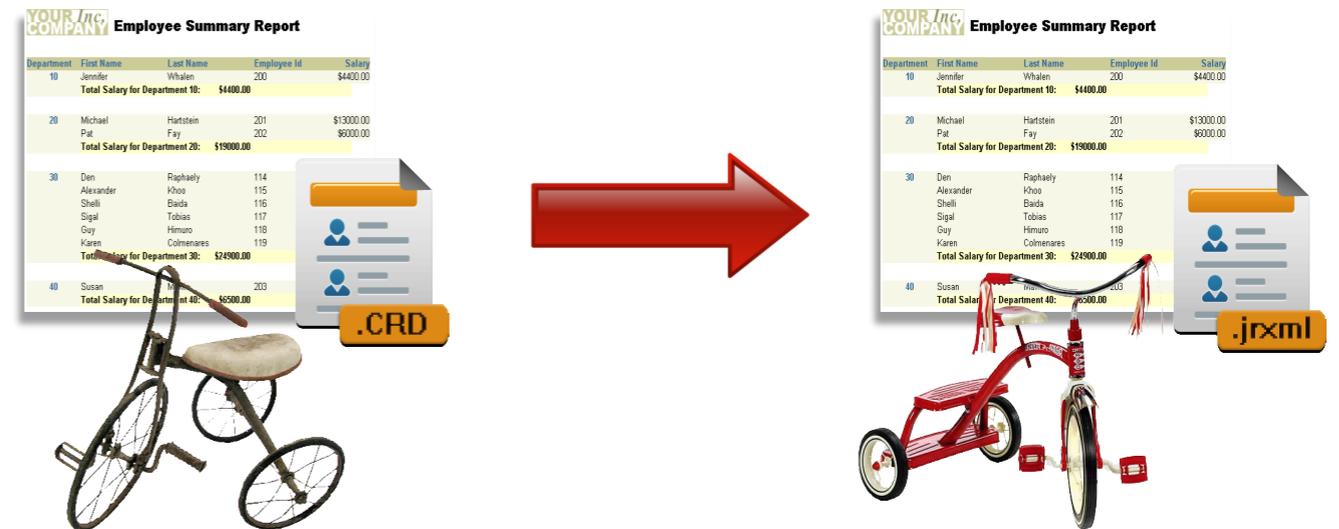
Transform old „List & Label" designs into modern jasperreports!

Course Dependency Graph

Organization: Student Council
Contact: Matthias Höschele

Last semester the students council discussed the idea of a system that can help students to decide which courses to take. The system should work like a special purpose social network in which students can rate and comment on courses. Students should also be able to point out which courses they consider to be prerequisites and worthwhile followups and therefore create a graph of dependencies between courses. For current courses there should also be an option to give direct anonymous feedback to enable lecturers to quickly react to problems and wishes of the participants.

B-HiP

Organization: DFKI
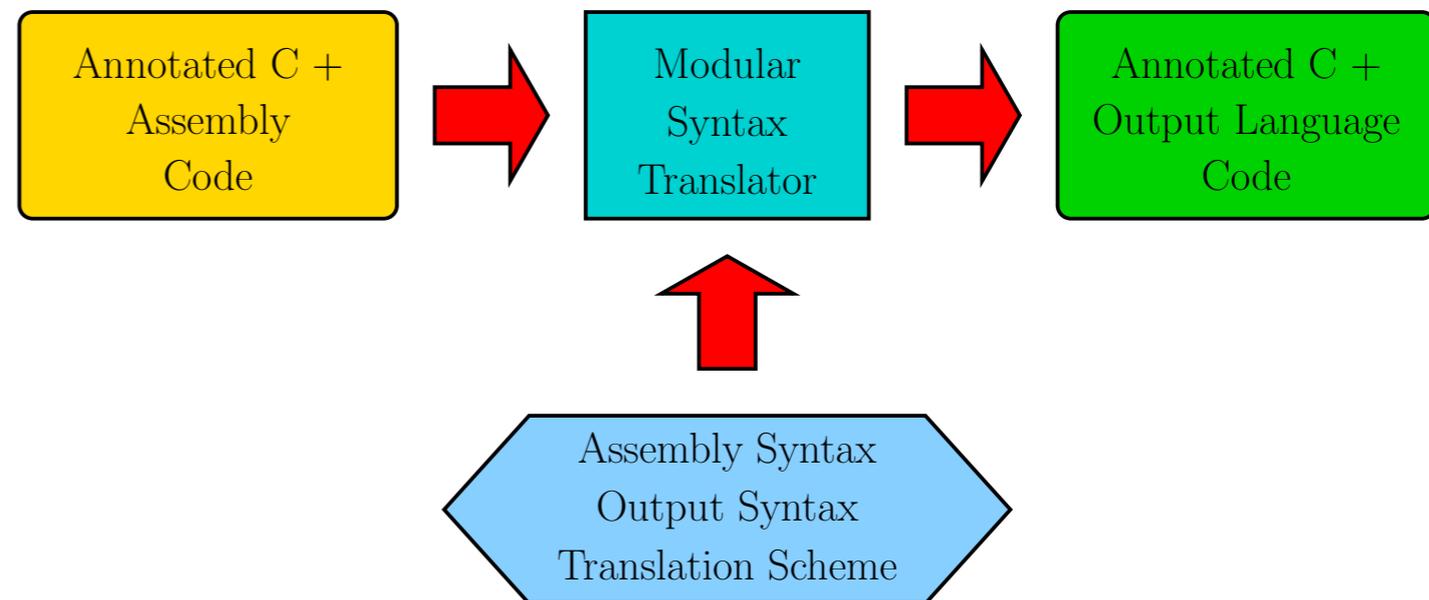Contact: Alexandra Chapko

In the project B-HiP (Business 4 High-Performance Computing) an event-driven architecture has to be implemented, which processes interactions and transactions and automatically builds semantic models in OWL. As a use case demonstrator, an iPad application has to be developed in Objective C which acts as a personal mobile assistant for a business man. According to current events (changes in train / flight schedules, postponed meetings), recommendations for complex task management should be created. Therefore the event-driven, semantic mechanisms of the Java EE backend should be used. The project should follow an iterative software development methodology, and should create testable prototypes for each iteration. Complex computations for reasoning and forecasting can be tested within the Future SOC (Service-oriented Computing) Lab of the Hasso-Plattner-Institut (HPI) in Potsdam, which offers a high-performance computing infrastructure for these purposes.

Modular Syntax Translator for Assembly Verification

Organization: Institute for computer architecture and parallel computing
Contact: Christoph Baumann

In the current projects of the chair we focus on the verification of operating system code using an automated C verifier (Microsoft Research's VCC). To handle the assembly portions we developed an approach that translates the assembly instructions into an appropriate representation consisting of C and VCC annotations. Now we would like the students to write a command line tool which automates this translation. The main difficulty is, that it should support various architectures (currently we envision x64, PowerPC and VAMP ASM) and translate to a provided output language. Hence the students will have to find a modular way to represent assembly code and define an interface for the exchange of input and output syntax as well the translation scheme. Moreover, to be able to argue about the translation step in our meta proofs justifying the overall verification approach, the students would also need to deliver an exact (formal) definition of the translation process implemented by their tool.

Annotated C + Assembly Code → Modular Syntax Translator → Annotated C + Output Language Code

Assembly Syntax Output Syntax Translation Scheme

uRun

Organization: DFKI
Contact: Alexandra Chapko

The idea is to develop a system which provides personalized and situation-aware recommendations to a runner. A is user equipped with a smart phone which collects information about the user's context (e.g. vital parameters, location or interests). Via an internet connection the information is sent to a backend where it is analyzed. Depending on the results of the analysis, different recommendations (e.g. running routes or speed) / services (e.g. advertisement or paying service) are provided to the user which support the user before, during and after the training. Furthermore, the system has to be able to learn from the behavior of a user's behavior so that the quality of the recommendation improves over time. The system should be implemented using Java/Java EE, mobile frontends (JSF, PHP), web services (Axis), semantic web technologies (OWL, RDF/S). An existing system has to be modified and extended in an iterative approach.

Prototype of a Portal-based workflow engine

Organization: DFKI
Contact: Markus Reiter

Objective of the project is the implementation of a workflow system prototype, which is based on open source technology. The required platform is the Liferay Portal server and the JBoss jBPM framework. The implementation shall demonstrate the feasibility to implement a workflow system on a portal server solution. The functionalities shall include, but are not limited to the following generic workflow features: loading predefined workflow models, define roles and responsibilities for these models, execute workflows and assign tasks to defined roles, trace the execution, and report the number of successfully executed workflows / errors. The special interest is the development of the workflow systems in a portlet-based environment, which is provided by the open source Liferay Portal server technology.

Driving school

Organization: Fahrschule Auffenberg
Contact: Dietmar Auffenberg

The project deals with the interactive preparation for the theoretical driving school test. So the job is to develop an Internet platform, on which the learner is able to learn simply for his theory exam. Therefor is a clear user interface to be create in which the learners can quickly find his way. There also should be several methods for learning: for example, realistic testing, learning only difficult issues, only personal selection of questions etc. Driving instructors should be able to enter the new students into the system and manage their own students. The driving instructor allows it to be feedback (statistics) to get on the learning progress of their students, so he knows when his students are ready for testing.