SESP 2010

Session:	Session: Simulation Kernels (08)
Type:	Concurrent Session
Date:	Tuesday, September 28, 2010
Time:	16:00 - 18:00
Room:	Newton
Chair:	
Co-chair:	
Remarks:	

Seq	Time	Title	Abs No
1	16:00	Tool-based Analysis of Space Domain Simulators	

Alves, T.
University of Minho, Portugal and Software Improvement Group, Netherlands, NETHERLANDS

Based on the ISO/IEC 9126 standard for software product quality, the Software Improvement Group (SIG) developed a tool to automatically analyze software maintainability, the Software Analysis Toolkit (SAT). The SAT is used in the SIG consultancy services and has been applied to software systems in the financial, insurance, government, logistical and IT domain. In this presentation, we will report on the usage of SAT to analyze the maintainability of two simulators used in the space domain: EuroSim and SimSat.

The SAT measures maintainability using a five stars rating: five stars are used for very good quality and one star for very-low quality. The maintainability model is layered and can be decomposed into sub-characteristics, as defined in the ISO/IEC 9126 standard, and then into product properties, defined by SIG. The product properties are volume, duplication, unit complexity, unit size, unit interfacing, and testing. These product properties are automatically measured from the source code. For each product property, a rating is derived by comparing measurements to thresholds, which have been calibrated with a large number of systems. The calibration process ensures that five stars represent 5% of all systems (the best quality), four, three and two stars represent each 30% of the systems, and one star represent the rest 5% of the systems (the worst quality).

To use the SAT, custom configurations were developed. The configurations are tailored to individual projects, defining the programming languages to analyze and how to distinguish production and test code. Custom filters were also defined to ignore generated code, code examples and libraries code. For each system, the SAT was executed generating a database with all the measurements and deploying a web application. The web application displays the key indicators for software quality in a dashboard and allows drilling down to measurements at the lowest level.

The SAT maintainability analysis revealed a rating of three stars for EuroSim and two stars for SimSat. In-depth analysis of both systems revealed problems in most of the product properties. As example, for duplication the SAT measures the number of identical lines of code and identifies clones (contiguous blocks of identical code). Besides showing the existence of the clones, the SAT identifies the files in which the clones are found, and the start and end lines of each clone. For EuroSim, 7.1% of the overall code is duplicated, while for SimSat this

value accounts to 10.4%. For both systems, we detected several identical files and several clones larger than 100 lines of code. Curiously, we discovered that, in both systems, several clones found were due to the (different) implementations of the ESA Simulation Model Portability library (SMP). As additional example, for unit complexity, the SAT measures the number of decisions per function (McCabe metric). For EuroSim, 42% of the overall code contains moderate and (very) high complexity, while for SimSat this values accounts to 20%. However, while in EuroSim the complexity is spread throughout the system, in SimSat it is localized in just a dozen of files.

Additionally, the SAT can be used to support management decisions through the identification and quantification of technical problems. In EuroSim, the complexity analysis helped to uncover that specific teams were responsible for the modules with lower quality. In SimSat, the unusual duplication, size and complexity found for recently developed code, helped to uncover that the team involved with that part of the project lack familiarization with the technology used. In both cases, a decision could be made not to use these teams, or to provide training helping the teams to produce higher quality code.

To conclude, the SAT provides a pragmatic and practically feasible approach to measure product quality, which can be used to support management decisions. The quality model implemented in SAT uses a small set of key metrics whose measurements are aggregated in a five star ranking. By design, the metrics are operational, understandable and generic. The star rating enables comparison and easy communication between all the stakeholders of a project. The analyses of EuroSim and SimSat revealed that although both systems were developed with rigorous software processes, implementing strict standards, quality problems were found. Hence, it is important to have mechanisms to identify and quantify quality problems during both development and maintenance activities.

2 16:30 Real Time Distributed Simulations Using SIMSAT 4.3

Whitty, J.
Terma GmbH, GERMANY

The need for large-scale real-time simulations is growing in demand within many diverse industries. This places ever-increasing performance requirements on the hardware platforms required to run the simulations. Possible solutions to this include the use of multiple core computers and a network of computers to distribute the simulator processing load across many cores and machines.

This paper examines some recent examples of such simulations within the European Space sector, and how SIMSAT 4.3 simulation infrastructure has been extended to satisfy the requirements to run distributed real-time simulations.

The underlying simulation environment used by the European Space Agency (ESA) and the European Space Operations Centre (ESOC) for Spacecraft simulators since the early 1990's is the SIMSAT real time infrastructure. SIMSAT originally ran under VMS, but since then it has been ported to many different Operating Systems, i.e. Windows NT/2000/XP and then most recently SuSE Linux Enterprise Server 9 and 11. It provides the real-time environment for spacecraft simulators to verify the ground segment readiness prior to launch, including the validation of the mission control system, validation of spacecraft procedures and the training of the spacecraft operators. The requirement to develop simulations for large and complex space systems within the European Space Market, for example Galileo, places heavy demands on the simulation platform. This increasing

performance demand has driven the development of a distributed version of the SIMSAT infrastructure.

The paper introduces SIMSAT at a conceptual level, the current SIMSAT Linux 4.3 Software Architecture, and the benefits of its framework "plug-in" design which make it so versatile, flexible and extendable. This plug-in framework allows fast and rapid development of new simulation components that plug-in to the SIMSAT framework, for example, to support the distribution and running of a simulator across a distributed network, or to selectively load various models and assemble different simulated components to run.

The system, network performance, time synchronisation, and configuration requirements of a real-time distributed simulation system are identified. The paper demonstrates how these constraints were solved through the software architectural design principles applied.

The results show that the SIMSAT 4.3 Infrastructure Software meets the system and performance requirements of running distributed models across a dedicated network. The new SIMSAT can be utilised across diverse industries to run large-scale distributed simulations.

3 17:00 EuroSim Strategy and Development

- Not specified -

4 17:30 EuroSim Mk4.2 New Features and Mk4.3 Outlook

- Not specified -