Hasan Sözer

Is it possible to just carry on watching a football game, while your TV fixes a software error in the background? The fault-tolerant software that Hasan Sözer has developed isolates errors and corrects them without the user noticing it .

Recovery at runtime

"Even with consumer electronics, you can already see that software is defining functionality to a high degree. That's true of the latest generation of TVs as well as mobile devices. And the complexity and scope of the software is only increasing. Given the short time-to-market, it's simply not possible to test the system for every possible situation. A system might also have to react to a whole range of external factors, for example, if it operates as part of a network. This makes it increasingly difficult to prevent errors occurring. The extreme requirements specified for the software in an airplane or a nuclear power station means that you're permitted the choice of having certain tasks executed in duplicate by different programs. For consumer electronics, that's just too costly."

"I therefore researched whether it was possible to develop fault-tolerant software, that is, you accept that errors can occur, but you build in routines to handle them. Fault-tolerant software includes procedures to enable a system to recover at run time. The well-known method for dealing with a computer that's hanging is to switch it off and then back on. That can be a source of annoyance, certainly if the computer hangs a lot. With a TV that's unacceptable. In the middle of an exciting film, you don't want to have to turn the set off and then on again because the software has made a processing error."

"Another challenge is to design the system so that it just carries on working while recovering from an error situation. This requires a thorough analysis as its starting point, because you want to know the location of the critical components in the system. Redesigning a system from scratch is not feasible, but you can certainly restructure the existing software. Using SARAH, our new method for performing reliability analyses, we can analize errors and assign them a priority. SARAH combines scenario-based analyses with reliability techniques. 'Reliability' is generally related to safety, but you can also use it in connection with the quality a user expects. This is our basic premise."

"Using the above-mentioned priorities, we can divide a system into separate components. We simply cannot make an entire system fault tolerant, that would be far too expensive. We know what the critical components are, so that's where we begin. By compartmentalizing the system, we can prevent the effect of an error spreading throughout the system. We isolate the error and invoke a local recovery procedure. In a TV, for example, you can decouple the software for the picture streaming from the software that controls the other processes."

'Fault-tolerant software includes procedures to enable a system to recover at run time.'

"When indentifying these individual components, their interaction is an important factor of course. Where precisely do you place the boundaries? Part of the decision-making is a cost-benefit analysis. The framework FLORA I've developed assists in defining recoverable units and provides the coordination and communication necessary for recovering from an error situation locally."

"I expect the need for this type of application to keep increasing. Systems are simply becoming so complex that something has to be done. Although I've focused specifically on applications for the software in TVs, I think more applications will follow. One example is the mobile phone, with its ever-growing number of functions."