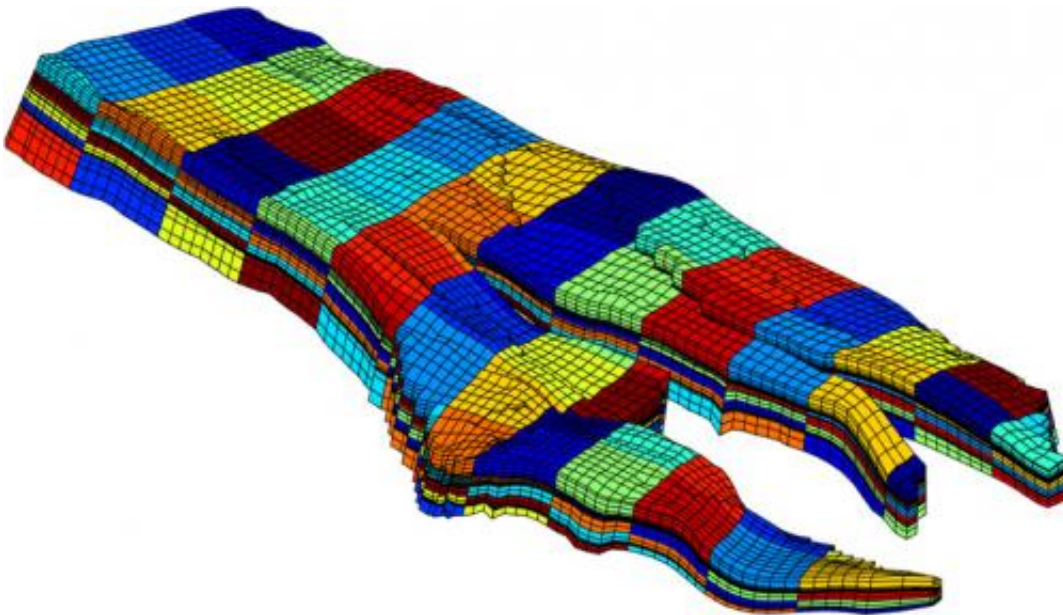


New programming language for fast simulations

January 26 2015



The new language is like the language of mathematics. This allows researchers to subdivide the work process in such a way that reduces the time it takes to develop a simulator. Just like this model which computes fluid pressure in the Norne field. Credit: SINTEF Group

Norwegian researchers have developed an entirely new language for faster programming and simulations.

Programming is a time-consuming process, and it may take many years to develop even a basic simulator. Researchers want to simplify this

process.

They have created a language similar to the language of mathematics. This allows them to subdivide the work process in such a way that reduces the time it takes to develop a simulator. A mathematician can then focus on what he or she knows best – the simulator's area of application. Programming experts, on the other hand, can sit and work with "the translator" in order that the translated code can run faster. They don't have to worry about the application.

Simulating oil and water

At SINTEF's Department of Applied Mathematics, researchers are developing methods and new ways of computing everything from weather forecasts for airports to the tactical formations of Premier League football teams.

In this project, they're concentrating on two [applications](#) – the way in which oil flows within its natural reservoir, and the way flood waves spread across a landscape. The source codes are used to develop simulation tools that can predict what happens to the oil in the reservoir and where the greatest volumes of flood water occur.

"A reservoir is a rigid sponge containing oil", explains SINTEF's André Brodtkorb. "In order to simulate the oil recovery process we divide the reservoir up into small segments or cells and describe how the oil migrates from cell to cell.

These things take time

It is common during the construction of a simulator to involve many people in the development of the programming code which instructs the

computer how to do its job. These include mathematicians, physicists, numericists and programmers.

"The problem is that they all have their own respective core skills", says Brodtkorb. "Those writing the [source code](#) for the simulator aren't necessarily fully aware of what the mathematicians and physicists are doing – and vice versa", he says. "And when people from different fields are trying to reach agreement, misunderstandings arise and things take time", he says.

"The major software companies generally have all the expertise they need in-house. But most research codes are developed by single individuals or small teams – and they need help", says Brodtkorb.

Removing security errors

Making simulators is in itself nothing new. The innovation the researchers want to introduce is the use of a domain-specific language – a precise language that focuses on a closely-defined area of application. Such a language will allow them to develop new simulation programmes and at the same time guarantee that the code is correct.

For example, in the field of programming there is a classic problem linked to selection of the correct units used to describe physical quantities. The Mars Climate Orbiter, launched by NASA to study the Martian atmosphere, crashed on the planet in 1999. The reason was that the company Lockheed Martin had developed a [simulation](#) model using imperial units when NASA was expecting them to deliver in metric.

"The error caused 327 million dollars to go up in smoke", says Brodtkorb. "We've built in safeguards to prevent this kind of thing from happening. For example, we write statements such as 'this unit is in metres' into the source code. If someone then attempts to introduce a

non-SI unit such as 'foot', they will get an error message", says Brodtkorb.

Another problem is how to maintain an overview of all the millions of cells which make up a [simulation model](#). In the middle of a hectic process it's easy to get confused and lose track of the individual cells.

"Even if something goes wrong, the result may nevertheless appear to be correct. If this happens, things can really go haywire and decisions may be taken on false premises", says Brodtkorb. "But in our system the possibility of errors such as these has been eliminated", he says.

The researchers are currently applying for funds to further advance their project.

"We have great faith in the substance of this project", says Brodtkorb. "Our new programming language is easy to use, it takes less time to write programmes, and the sources of error have been removed. Three good sales arguments, don't you think?" he says.

More information: Read more: <http://www.gemini.no>

Provided by SINTEF

Citation: New programming language for fast simulations (2015, January 26) retrieved 24 April 2024 from <https://phys.org/news/2015-01-language-fast-simulations.html>

This document is subject to copyright. Apart from any fair dealing for the purpose of private study or research, no part may be reproduced without the written permission. The content is provided for information purposes only.