

CASE STUDY

The most technologically advanced rover ever built, the Curiosity Rover is a mobile laboratory about the size of a small SUV.

Curiosity's 17 cameras, robotic arm, and suite of specialized laboratory-like tools and instruments are controlled by more than 2 million lines of software.



GRAMMATECH

Mars Curiosity Rover Searches for Signs of Life with the Help of GrammaTech's CodeSonar

After its eight month journey spanning 352 million miles, NASA's Mars Curiosity Rover completed a spectacular landing with the help of a giant parachute, a jet-controlled descent vehicle, and a bungee-like apparatus called a "sky crane." Due to the time required for messages to travel from Mars to Earth and back, the landing procedure was completely controlled by software. To boost the reliability of the software, NASA needed advanced static analysis.



OTHER CUSTOMERS IN THE AEROSPACE & DEFENSE INDUSTRY INCLUDE:

AIRBUS France

BAE Systems

Boeing

Booz Allen Hamilton

GE Aviation

Honeywell

Lockheed Martin

Northrop Grumman

Raytheon

Sypris

Developing Zero-Defect Software

Spacecraft have become increasingly reliant on software to carry out mission operations. Curiosity relies on more software than all the previous missions to Mars combined. Even the fault protection systems on a spacecraft are software-based.

For the two years leading up to the launch, NASA focused on developing mission critical, zero-defect software. Every single line of code was analyzed and scrubbed using advanced static analysis tools, including CodeSonar® from GrammaTech. "NASA's Jet Propulsion Laboratory used CodeSonar to check for bugs in the Curiosity software on a nightly basis," said Michael McDougall,

senior scientist at GrammaTech, who worked closely with NASA.

Advanced static analysis finds serious software errors such as buffer overruns, race conditions, null pointer dereferences, and resource leaks. It also finds subtle inconsistencies such as redundant conditions, useless assignments and unreachable code. "Because static analysis is a compile-time process, it can find bugs while the software is being developed," said Paul Anderson, Vice President of Engineering at GrammaTech. "The tools examine paths and consider conditions and program states in the abstract. By doing so, they can achieve much higher coverage of code than is usually feasible with testing alone."

“For NASA, this was a huge accomplishment and we’re pleased that we were able to contribute to their success.”

— Michael McDougall
Senior Scientist,
GammaTech

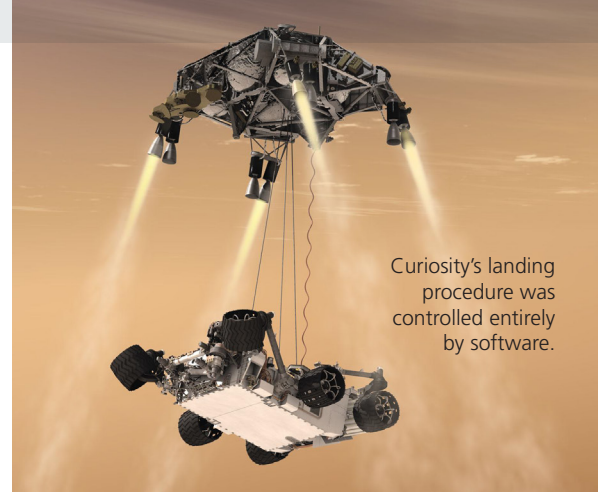
Curiosity’s Software Upgrades

Upon landing on Mars, Curiosity underwent a four-day major update to delete the landing software, and install the surface operations programs designed for roaming the red planet. NASA designed the mission to be able to upgrade the software as needed for different phases of the mission. Software upgrades are necessary in part because Curiosity’s computing power is relatively low compared with what we’re used to on Earth. However, the RAD750 PowerPC microprocessor built into the rover’s redundant flight computers was chosen because it is virtually impervious to high-energy cosmic rays that would quickly cripple a smart phone or laptop computer. Additionally, the rover’s main computers have only about 4 gigabytes of storage capacity, compared with 32 gigs or so for a smartphone.

New software for upcoming phases of the mission is being developed on an on-going basis. Because a single error could result in loss of contact with the rover and jeopardize the mission, every software upgrade must work perfectly the first time it runs.

Coding Standard for Mission Critical Software

NASA has an excellent track record for producing high quality software and follows a number of best practices. As part of its rigorous development process, the Mars Curiosity mission follows “The Power of 10: Rules for Developing Safety-Critical Code,” developed at the Laboratory for Reliable Software (LaRS) at NASA’s JPL. GammaTech worked with NASA to extend its CodeSonar static analysis tool to automatically enforce the Power of 10 rules, as well as automatically flag generic programming defects.



Curiosity’s landing procedure was controlled entirely by software.

In developing the coding guidelines, JPL looked at the types of software related anomalies that had been discovered in missions during the last few decades and came up with a short list of problems that seem to be common across almost every mission. This led to the idea of defining a very small set of rules that could easily be remembered, that clearly related to risk, and for which compliance could mechanically be verified. The 10 rules are designed to reduce risk for mission critical software, and have evolved into the JPL Institutional Coding Standard for the Development of Flight Software. Rule ten specifies that advanced static analysis tools should be used aggressively throughout the development process.

“I’m extremely proud of the part GammaTech played in the successful landing of Curiosity,” said McDougall. “For NASA, this was a huge engineering accomplishment and we’re pleased that we were able to contribute to their success.”

Curiosity continues to make new discoveries. Each day 100 engineers and researchers write commands to keep the rover productive and gathering science. With the ability to reliably upgrade software as needed, NASA can answer new questions about Mars and expand our understanding of the planet.



For more information:
www.grammatech.com
Email: info@grammatech.com

GammaTech Headquarters:
6903 Rockledge Drive, Suite 1250
Bethesda, MD 20817

U.S. sales: 888-695-2668

International sales:
+1-607-273-7340

Email: sales@grammatech.com