

# ELESYS North America Speeds Verification of Safety-Critical Embedded Software with PolySpace™ Products for C/C++

While airbags help protect adult passengers in automobile collisions, the force of a deployed airbag can seriously injure younger passengers. To minimize this risk, the U.S. National Highway Traffic Safety Administration requires all vehicle manufacturers to comply with Federal Motor Vehicle Safety Standards (FMVSS) 208 and suppress airbag deployment when a child is detected in the seat.

As a global business leader in automotive safety electronics, ELESYS North America supplies customers such as Honda and General Motors with a wide range of passive and active safety systems, including airbag sensors and controllers, adaptive cruise control, transmission controls, vehicle stability controls, and electronic power steering. ELESYS uses PolySpace™ Client for C/C++ and PolySpace™ Server for C/C++ to ensure the reliability of its systems' embedded software while meeting the company's time-to-market objectives.

"The whole application must execute flawlessly each and every time—lives depend on it. That means that we must ensure that such applications are bug-free," says Svetoslav Stoyanov, software engineering manager for the Software Development and Testing Group at ELESYS North America. "PolySpace allows us to debug our applications much faster, which enables us to go to market faster without the need for additional resources."

**THE CHALLENGE**  
ELESYS developed an airbag suppression system that detects the passenger's weight and disables or enables the airbag as neces-



Airbag deployment during a frontal crash test.

sary to avoid injuries to children. To ensure the safety of this and other systems, ELESYS engineers conducted time-consuming manual reviews to check for the presence of run-time errors.

"The manual code review process was an inefficient use of our resources," says Stoyanov. In addition to being time-consuming, pinpointing errors with this approach was a challenge. Stoyanov adds, "Some run-time errors may only occur under specific operating conditions, which makes them very hard to identify."

ELESYS sought to accelerate run-time testing without adding more testing resources or compromising quality. "We wanted to find a tool that would help us reduce verification time while helping us achieve our stringent software reliability objectives," says Stoyanov.

**THE SOLUTION**  
ELESYS engineers used PolySpace products to conduct an exhaustive analysis of their code. PolySpace analyzed each function and operation for its possible value range and

**THE CHALLENGE**  
To accelerate the verification of airbag suppression systems

**THE SOLUTION**  
Use PolySpace to detect run-time errors in safety-critical embedded software

- THE RESULTS**
- Verification in days, not months
  - Hard-to-find bugs located
  - Testing efficiency increased

“ PolySpace enabled us to dramatically reduce our workload per analysis – from several man-months to a matter of days. ”

Svetoslav Stoyanov, ELESYS North America

then color-coded it according to its reliability, as follows:

- Green. Safe under all operating conditions
- Red. Definite error each time the operation is executed
- Orange. Possible run-time error depending on operating conditions
- Grey. Unreachable code (may indicate a functional issue)

The team focused their resources on the areas that PolySpace had highlighted in red and orange.

ELESYS made PolySpace an integral part of its release procedure. All ELESYS software must be analyzed with PolySpace before it is released, to ensure that run-time errors will not be uncovered during the maintenance phase.

After initially using PolySpace only for software validation, ELESYS engineers now use it throughout development to find and eliminate errors in the early stages of each project.

“During development, whenever we are developing a critical module that embeds complex code, we use PolySpace to quickly identify any potential run-time errors,” says Stoyanov.

## THE RESULTS

### ■ Verification in days, not months.

“On average, a PolySpace analysis on a complete application takes six or seven hours. If we were to use manual code review with the same resources, it would take us at least a month,” says Stoyanov. “Overall, PolySpace enabled us to dramatically reduce our workload per analysis from several man-months to a matter of days.”

■ **Hard-to-find bugs located.** “Some bugs would be very difficult, even impossible, to identify manually or through conventional testing. PolySpace enables us to find these potentially critical errors fast, and provides for an unparalleled level of software reliability. In our line of products, this is a powerful competitive advantage,” says Stoyanov.

### ■ Testing efficiency increased.

“PolySpace is highly selective, which means we did not have to sift through a huge number of baseless warnings before identifying real issues,” explains Stoyanov. “Other tools may tell you where errors lie, but cannot guarantee that other operations are safe. That usually forces you to proceed with redundant tests. PolySpace flags all operations that will never experience run-time errors—no matter the operating conditions. That enables us to invest our resources where they are the most effective.”

To learn more about ELESYS North America, visit [www.elesys-na.com](http://www.elesys-na.com)

## APPLICATION AREAS

- Automotive
- Control design
- Verification and validation

## PRODUCTS USED

- PolySpace™ Client for C/C++
- PolySpace™ Server for C/C++

[www.mathworks.com](http://www.mathworks.com)