

## Hungarian technology company reduces cost and developer time while increasing software performance and safety for Hungarian railways



### Executive Summary

**Customer Name:** EvoPro Innovation  
**Industry:** Application and Technology Company  
**Location:** Budapest, Hungary  
**Company size:** 250+ staff

#### Challenge

- Develop faster and safer solutions for railway sensor diagnosis across Hungary
- Reduce development time of multi-core solution
- Reduce time-to-market for solution
- Reduce training and staffing needs

#### Solution

- Integrate ParaFormance tool-kit into development environment
- Identify opportunities to parallelise source code automatically using ParaFormance
- Automate code rewriting to enable for multi-core using ParaFormance refactoring
- Ensure refactored code is safe using automated dynamic and static safety

#### Results

- 1 week of manual implementation effort reduced to around 3 hours
- 3500 lines of code analysed and refactored using ParaFormance
- High performance, reliable and parallelized multi-core solution for railway diagnostics

### Challenge

EvoPro Innovations LLC (EvoPro) is a specialist software development and professional services firm, based in Budapest and serving an international client base. EvoPro operates in the fields of embedded systems, software development, mobile computing and large-scale control systems. A recent product developed by EvoPro is a railway diagnostics system for the Hungarian railway network.

The Railway Diagnosis application (eRDM) is a dynamic railway diagnostic system that is able to measure the load of each wheel, axle and carriage of the train passing over the system at operating speed. The system also provides diagnostic features: it is able to detect unbalanced rail-cars, wheel flat spots, damaged bogies and suspension problems. Thus the system allows real-time freight transportation monitoring, helps to increase railway transportation safety and to reduce rail/carriage maintenance costs.

Parallelising this application, which consists of around 3500 lines of complex code, took the team of specialised developers at EvoPro 1 week of effort, and this does not include the time to design and implement the solution in the first place.

In common with other software firms, EvoPro has struggled to recruit the developers it needs who are skilled in parallelism. Very few engineers learn about parallelism, including patterns, during their university studies. It can take around 1 year to bring

a software engineer up to the required standard of writing parallel code. This is clearly both expensive and time consuming.

**“The ParaFormance tool is very powerful.”** says Zsolt Szepessy, CEO EvoPro Innovations, Hungary.



*Figure 1: Example of Railway Sensors*

### **Solution**

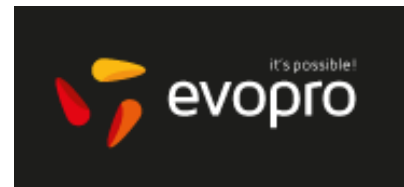
First, EvoPro ran their 3500-line application through the ParaFormance pattern discovery tool. After just a few minutes, ParaFormance analysed the entire source code structure and reported back 15 potential sources of parallelism. This pattern discovery is a key feature of the ParaFormance tool set. It analyses the application and reports to the programmer where the multi-core opportunities lie within the code. Finding the sources of parallelism in an application is often a lengthy and very time consuming process. EvoPro found that it took 2 days for a team, already familiar with the code and experts in writing parallel software, to do this job manually, and furthermore, through the manual process, only finding 2 or 3 potential sources of parallelism. It would be a very daunting task indeed for a junior or non-specialised developer!

Secondly, EvoPro took the most likely candidates, as presented by the pattern discovery process and ran them through the next stage of refactoring, using the ParaFormance refactorer. The Refactoring feature automatically rewrites C++ code into parallel code at the touch of a button. Moreover, the refactoring tool does not just rewrite code, but it applies very detailed and sophisticated checking algorithms to ensure that the code being written is both sensible to rewrite for multi-core and also does not have any properties associated with it that can introduce runtime bugs when the multi-core version would go live<sup>1</sup>. These are all very time consuming and very specialized activities for a developer to implement manually. After this refactoring of the code, ParaFormance introduces all of the multi-core “business logic” into the source code automatically. Moreover, using ParaFormance, the EvoPro team were able to try out three different parallel implementations. Normally that would require a major effort of learning three different parallel libraries, but also implementing and understanding the parallel libraries. After deploying the application, EvoPro found that their application had reduced its runtime overhead by 7 times. That is a 7 times performance increase at the touch of a button.

Finally, the application was deployed and the safety checking feature of ParaFormance automatically found a crucial error in one of the parallel implementations automatically. This feature performs very detailed analysis of the code to ensure that it is thread-safe, and that running the multi-core version of the application will be free from bugs that only occur on multi-core technology. This task is often overlooked by software houses when writing multi-core code and it is typically the task that takes the longest time to complete. The ParaFormance tool finished in minutes and highlighted potential thread safety violations as well providing hints to the developer for repairing them.

---

<sup>1</sup> Race conditions, deadlocks, etc.



## Results

EvoPro evaluated ParaFormance on their Railway Diagnostic Application, eRDMA, an extensive and important application that reads very large amounts of railway sensor information in order to determine in real-time whether or not rolling stock is faulty. It does this in real-time while that rolling stock is travelling over the railway lines. Once this data is processed, if a pattern is found, then the train could be stopped and inspected quickly and efficiently, and before any accident could occur.

To achieve the required performance levels, the eRDM application needed to be parallelized. This is a difficult and very expensive task for EvoPro because it is hard to recruit the software developers needed because there are so few developers with the necessary skills and expertise.

Using ParaFormance solved the problem quickly. From start to finish, going from a single-core legacy 3500 line C++ application to one that is optimised and ready for multi-core, took about 2-3 hours. (That included the time to understand and install the ParaFormance tool). This is a process that would take a specialist developer around 1 week of manual effort, equating to approximately £2-3k of employment cost. Because the code ParaFormance generates is faster and because hidden errors are identified and corrected, the railway is potentially safer, too. Return on investment from ParaFormance was very fast and represented excellent value to the company.