

European Commission Helps National Agencies Monitor Fishing Activity using RCP



"The maturity of the RCP framework takes care of the details for me. I feel like I'm sitting on pretty big shoulders."



The world's oceans are in trouble. With some experts estimating that they will be dead from overfishing within thirty years, the need for action is immediate. One group that is stepping into the fray to protect fisheries is the European Commission's Joint Research Center. The JRC works closely with policy makers, conducting applied research in support of EU regulations.

Finding Fishy Business

SUMO, for Search Unidentified Marine Object, is an Eclipse RCP-based tool developed at the JRC and used in collaboration with national agencies to police fisheries.

"Think of it as the intelligence for satellite-based monitoring." said Francois

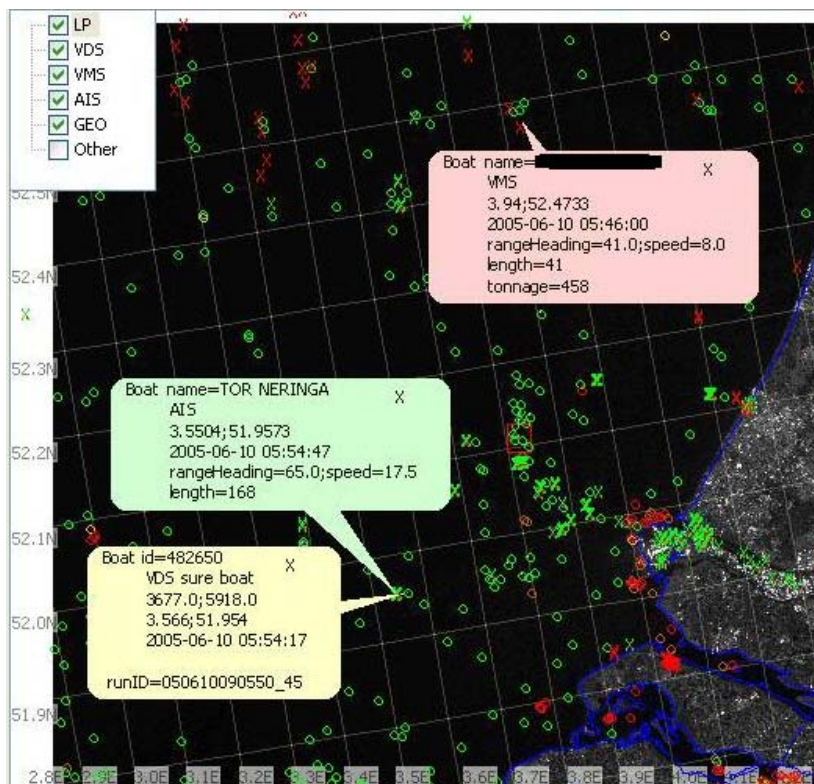
Thoorens, software developer with the JRC. "By

law, fishing vessels are required to retrieve their location from GPS and report it to authorities every two hours. SUMO takes that data and compares it to high-resolution satellite imagery. Within minutes it can tell if activity on the ocean does not match with positions reported."

When this happens, national fisheries inspection authorities such as the coast guard can be dispatched to investigate.

SUMO can also help regulators identify fishing patterns that may be detrimental to a sustainable marine environment and use this information to

guide policy.



SUMO uses RCP to compare reported fishing boat locations with satellite data. Compliance with fishing regulations is easier to monitor.

Finding Boats - not Bugs

Thoorens is the only developer on the SUMO project. Starting with a SWING-based version developed in 2004 with netbeans IDE, he maintained the application himself. He quickly realized that he needed a platform that makes development easier instead of getting in the way. "NetBeans was just too nuts-and-bolts to develop in. It was complicated to learn, and I could see that adding things like reports or new data views would never be simple, even with experience." Thoorens relates.

He also found that Swing was too slow for the intense graphic processing required to work with images that are often around 1GB in size.

So he began looking at Eclipse RCP as an alternative for future development. He liked what he saw and released an RCP-based version late in 2005. "RCP has a lot of advantages," he says. "It is a much easier framework to maintain, so I can concentrate on improving functionality."

For example, he used the JFace UI toolkit to add views and perspectives for overviews, precision picking, detail zooms, 3D overlays and custom algorithms. It gave him the data abstraction he needed to implement each with just a few mouse clicks and a bit of coding. "There was nothing like JFace in NetBeans" states Thoorens, "It really helped me to be effective."

He also points to the general polish of the Eclipse IDE as something that's very helpful for a lone programmer. Small things like more advanced auto-completion make a big difference by helping him code faster and more accurately.

RCP's plug-in architecture helped the JRC work within the European Commission's rules against distributing source code while giving SUMO's users freedom to customize the application to meet their needs. Plug-ins provided a clean, powerful, and open interface for customizations – another detail RCP took care of for them.

While the JRC was not free to share SUMO's source code, Eclipse's open source philosophy has helped Thoorens make the best coding choices without a big investment in time. For

example, to process satellite images most efficiently, he needed to read the data format from the file's headers and dynamically load the best class library for the job. Although achieving this in RCP was not obvious at first, a quick look at the source code showed him how to do it. Now, when more image formats become available from new satellites, he can support them easily.

Straightforward integration with backend services also helped ease SUMO's development. Geographic data such as locations are stored in PostGis, a Postgres SQL-based geographic data management system. Thoorens was able to integrate SUMO with PostGis quickly using standard JDBC. Integrating a map server over an HTTP connection was similarly easy.

Location data comes from various national agencies in all sorts of formats. "We even got a screenshot once!" recalls Thoorens. Sophisticated XML support takes care of much of this information, and he expects that by using Eclipse DTP, or Data Tools Project, in the future, handling new data formats will become a non-issue.

Never Alone with Eclipse

Although SUMO was developed by just one developer, he was never really alone. The Eclipse community proved to be an invaluable support network. A case in point: Thoorens recently added automated reporting using BIRT, Eclipse's Business Intelligence and Reporting Tools. When SUMO identifies an anomaly requiring investigation, it generates a report that can be dispatched to national authorities for follow up.

While BIRT has proven to be a robust toolset and has simplified development, he did find the documentation less detailed than with other Eclipse projects. The user community easily filled in the blanks – answering forum questions quickly and getting him past any technical 'how-to' issues as he ran into them.

Thoorens is developing a reputation as something of a one-man-wonder. "People don't see how I can manage such a large project myself," he relates. "But Eclipse has made it

feasible. The maturity of the RCP framework takes care of the details for me. I feel like I'm sitting on pretty big shoulders.”

[Ron Stone](#) is a technology writer and content management consultant based in Ottawa, Canada.

April, 2007