User Voices

Russell Winder

Version 1.0, 2015-10-01

Table of Contents

What users say about GPars

Prakash Viswanathan — Java and Groovy Developer

GPars library is powerful, very simple to use, well designed and it brings advanced threading concepts like DataFlows, Agents and Actors formerly only sparsely available in Scala, Clojure to the Groovy world.

I use **GPars** in my project for a billion dollar product and services company in North America. Our application performs 14 to 20 million transactions a day which involves web service, complex computations and database calls.

❗ Our application performs 14 to 20 million transactions a day !

GPars library is solid and works like a charm. With this library, our application code is much shorter, clean and easier to maintain.

My work life is much easier after using Agents for monitoring and collecting stats on large batch processing jobs, and using Data Flow Queues and Tasks for adding and consuming jobs with just a few lines of code. And we got awesome support from the GPars team.

John Rudnik

GPars is awesome. Generally dealing with threading and concurrency is a real pain. With **GPars** you can get some very usable concurrency going in 3 lines of code.

Aurelien Maury — Xebia France — Tech Lead

We used **GPars** Actors to scatter HTTP requests in our Grails project to multiple backends, gather results and stream HTTP chunks of results as soon as

> ❗ It was just plain fun

they came back from backends. It was just plain fun, no Thread management boilerplate, just business rules optimizations. Would definitely re-use.

Andrzej Grzesik — eCircle, development env lead

Yes, I use **GPars**, it rocks. We've hacked a rdbms → hbase migration application, **GPars** helped to improve speed by a huge factor, and it was easy to use and didn't give us problems. - www.ecircle.com

Adrian Nakon

GPars is quite awesome - I’m building a multi-threaded App that collects metric data across a large number of Cisco switches and Linux servers (using Groovy and Java), and **GPars** is working like a dream. The concept of stateful Actors is very nice. Well done! :)

Robin Bramley

On a data migration exercise from *SugarCRM* to *Salesforce*, some of the entity migrations could be performed in parallel as they weren’t order dependent. **GPars** was chosen for sheer simplicity of *GParsPool* and *eachParallel* - only requiring 3 trivial new lines of code and the addition of 'Parallel' to the each collection iteration.

GPars dramatically reduced the time taken to migrate the data by parallelising the processing of the database result set and the subsequent web service calls.

leanjavaengineering.wordpress.com

Dan Fraser

GParallelizer is very cool. I had to collect information from 200 machines using JSCH (www.jcraft.com) and 3 lines of code made it 10x faster.

Contact: twitter.com/gblack

Jeff Gortatowsky — Software Architect

Been having fun with **GPARS** because it is so easy to experiment with. It lets me concentrate more on the solving the problem at hand rather than worrying about all the Java mechanics involving with coding the details of thread lifecycle management. Plus it is more expressive!

Feel free to add your own comments

add your user voice through our form

We will be happy to add you and your project to the list of happy GPars users.