



BIGLEVER NEWSLETTER: **From the PLE Frontline**

Paul's Three Surprises: Part 3

Greetings from Paul Clements:

In the last two newsletters, I've been writing about the biggest surprises I've encountered with product line engineering in the six months or so since I joined BigLever. The first one was that, among the shared assets in a product line, software does not occupy a privileged position, and may not occupy any position at all. (See [Paul's Three Surprises: Part 1.](#)) The second was that the business relationships supporting the configurator are as important as the technology inside it. (See [Paul's Three Surprises: Part 2.](#))

My third surprise has been the role of architecture in product lines.

Product line architecture. Or not.

A great deal has been written about product line architecture. In the academic and research worlds, there are conference workshops devoted to it, and tall stacks of journal papers about it. Books on software product line engineering devote chapters to it. It's treated as the king of the core assets, and good luck fielding a product line if you don't have one.

Now, counting second and third editions, there are six books with "software architecture" in the title that have my name on the cover. I have believed and continue to believe in the fundamental importance of architecture in the successful development of any software-intensive system.

I continue to believe in the usefulness of architecture evaluation as a standard lifecycle activity, and in the criticality of clear and stakeholder-focused architecture documentation. I continue to wish that more organizations would follow this advice. Did I mention the six books? They're full of it. (Full of the advice, that is.)

Having said that, though, I've been surprised at the small marginal differences I'm seeing between product architectures and product line architectures. In fact, the latter is almost just a conceptual term that stands for the aggregation of the former. The sampling of organizations I've seen so far here do not seem to be investing in centralized product-line-wide projects to invent the product line architecture.

The lesson seems to be: Architect your products, not your product line.

■ BigLever at IBM Innovate

As an IBM partner, BigLever will provide insight into the latest advances in product line engineering at the following Innovate conference sessions:

Innovate 2012

June 3 - 7, Orlando, Florida

Panel Discussion: The Convergence of Engineering Disciplines in Modern Systems Engineering

June 4, 4:15 PM - 5:45 PM

Panelists:

Dr. Charles Krueger, BigLever
Randy Skelding, Pratt & Whitney
Brian Wells, Raytheon
Tony Baer, Ovum
Mark Holt, IBM

Case Study Presentation:

Advances in PLE for Aerospace and Defense: Featuring Lockheed Martin and the Aegis Product Line

June 5, 10:30 AM - 12:00 PM

Presenters:

Dr. Paul Clements, BigLever
Susan Gregg, Lockheed Martin

Mega-Scale Product Line Requirements Engineering at General Motors

June 6, 4:15 PM - 5:15 PM

Presenters:

Dr. Charles Krueger, BigLever
Rick Flores, General Motors

Come see us at Innovate! Visit BigLever on the showfloor, booth 516, in the Ready for Rational Pavilion.

■ About BigLever and Product Line Engineering

As an industry thought leader, BigLever offers a variety of informational resources regarding

Then manage the architectures in a product line fashion by (just like every other kind of asset) capturing their differences by building in variation points in the assets that represent the architectures – documents, design models, etc.

Now, within a product line, the more alike those product architectures are, the easier it will be to define and build the variation points. (Again, that's true for all assets.) But do the similarities in product architectures emerge from (a) the fact that the products are all solving similar problems and so turn to similar solution structures (or perhaps borrow each other's), or (b) a gifted product line architect with experience and foresight to design a works-for-the-product-line solution? Perhaps the right answer is (c) a combination of the two. A product line emerging from an already-existing family of separately-developed systems would certainly favor (a), but the re-engineering of those systems during the emergence could nudge them towards a common solution, à la (b).

In any case, product line architectures are important, because product architectures are important.

But, surprise! Paying attention to the latter can take you much farther down the road to the former than I would have imagined.

I hope you've enjoyed this newsletter series. You're welcome to send feedback, or tell me your biggest surprises about product line engineering. If there are enough interesting responses, I'll compile them and write about them in a future newsletter.

Best Regards,

Dr. Paul Clements

BigLever Software Vice President of Customer Success

pclements@biglever.com

Don't miss your Newsletter!

Please help us make sure that you continue to receive the BigLever Newsletter by confirming your subscription with us. This ensures that future newsletters will be successfully delivered to your Inbox and not misplaced into your junk mail or spam folder. To confirm, simply click the "confirm" link in the white bar below. After confirming, you can unsubscribe from our newsletter distribution at any time.

new generation product line engineering solutions, methods, and real world deployments.

- [Overview: Product Line Engineering Solutions for Systems and Software](#)
- [White Paper: The Systems and Software Product Line Engineering Lifecycle Framework](#)
- [White Paper: The Pragmatic 3-Tiered Product Line Engineering Methodology](#)
- [Case Study: General Dynamics & U.S. Army](#)
- [Analyst Report: Managing Quality in Product Line Engineering – An Automotive Story](#)

■ [Getting Started](#)

It can be difficult to decipher where to begin with your transition to product line engineering practice. The important question is... where do you start next Monday? BigLever has a practical, proven approach – and a special package offer – to help you get started.

>> [Learn more.](#)

■ [About BigLever](#)

BigLever Software, Inc.TM is the leading provider of systems and software product line engineering framework, tools and services. BigLever's patented GearsTM solution enables organizations to reduce development costs and bring new product line features and products to market faster, enabling businesses to more reliably target and hit strategic market windows.

BigLever Software, Inc.
10500 Laurel Hill Cove
Austin, TX 78730
www.biglever.com
512-777-9552
info@biglever.com