



Software Product Lines: What's in? What's out?

Media Alert: New Software Product Lines Weblog

With the launch of the new weblog, "Charlie Krueger's Spotlight on Software Product Lines", BigLever Software's CEO will share his practical experiences and perspectives on the emerging field of software product line development. (See below for additional weblog information.)

Message from Charlie Krueger...

A recent posting on the SoftwareProductLines.com discussion forum prompted me to explore the following questions: "What constitutes a software product line approach? When you look at all the different software development practices in the industry, how do you determine "what's in" and "what's out" when it comes to identifying and characterizing software product line (SPL) development approaches?"

These questions are particularly important to those just learning about the emerging topic of software product line development, who want to know how it is similar to and different from what they are doing today. This struck me as a perfect inaugural topic for my new weblog. I've included a snapshot of my thoughts below -- you can view the full posting on the weblog:

> [Charlie Krueger's Spotlight on Software Product Lines](#)

I've talked with hundreds of development organizations about SPL development. From these explorations, two things are eminently clear: (1) The vast majority of companies today develop a portfolio of related and differentiated products - nobody builds just one product "flavor", and (2) The vast majority of development organizations have mechanisms in place to deal with the commonality and variability in their product line.

These two facts might seem to imply that most software development organizations already have software product line development practices in place. But, we have to raise the bar a good bit higher to distinguish those approaches that offer the order-of-magnitude benefits we have come to expect with *true* software product line development.

The techniques most often applied include preprocessors (such as IFDEFs), clone-and-own, configuration management branches, file and directory naming conventions, build scripts, installer scripts, and runtime conditionals with config file or database settings. If you look at the definitions of SPL in publications and books, things like clone-and-own or IFDEFs

Software Product Lines and BigLever

"Data extrapolated from Software Engineering Institute case studies indicates that as much as 50% of development time is eaten up dealing with issues surrounding multiplicity. A software company called BigLever wants to change that by assigning first-class problem status to variations in a software product line."

-- David Rubinstein, SD Times, August 15, 2005

> [See full article.](#)

"Focusing on the software development process for a line of products, rather than just one individual product, BigLever's Gears seeks to cut costs and time to market by leveraging existing software assets like source code, architecture, and processes."

-- Gil Bassak, eeProductCenter, July 27, 2005

> [See full article.](#)

"The [BigLever] tool gives companies a splendid strategy for building a product line by means of reengineering legacy assets... A company using this tool cannot help but think of itself as building variations of a single product. It's the singular product line that evolves over time, by adding new variation capabilities."

-- Paul Clements and Linda Northrop, Software Product Lines: Practice and Patterns, 2001

appear to qualify since predictive and intentional reuse occurs within the product line. However, I don't qualify most of what is going on in the industry as software product line approaches.

I believe that three things must simultaneously occur in order for a development approach to be characterized as a SPL approach:

1. The approach must effectively capitalize on the commonality among the products in a product line in order to avoid duplication and divergence of development effort.
2. The approach must efficiently encapsulate and manage the variation among the products in a product line in order to avoid the labor-intensive combinatoric complexity during development.
3. The majority of the software development effort should be focused on *product line feature* development, with a minority of the effort focused on *product* development.

It is the 3rd item that most organizations in the industry are currently missing. It is not until development organizations shift their perspective 90-degrees -- from product-centric to product line feature-centric -- that they start to experience the extreme benefits that we expect from a "real" SPL approach.

> [See full article.](#)

The New Weblog: Charlie Krueger's Spotlight on Software Product Lines

You can view the weblog using [your web browser](#), or use the RSS feed on the weblog to see new syndicated postings as soon as they become available.

Charles W. Krueger, BigLever Software CEO
512-426-2227
ckrueger@biglever.com
www.biglever.com

> [More about this book.](#)

About BigLever

BigLever Software, Inc. (TM) is a leading provider of software product line development tools and services. BigLever's patent-pending solution, Gears (TM), dramatically simplifies the creation, evolution and maintenance of embedded or standalone software for a product line.

Subscriber Functions:

[Forward](#)
[Subscribe](#)
[Update Info](#)
[Unsubscribe](#)

Contact Info:

BigLever Software, Inc.
10500 Laurel Hill Cove
Austin, TX 78730
512-426-2227
www.biglever.com

Copyright © 2005 - BigLever Software, Inc.

[unsubscribe](#)

