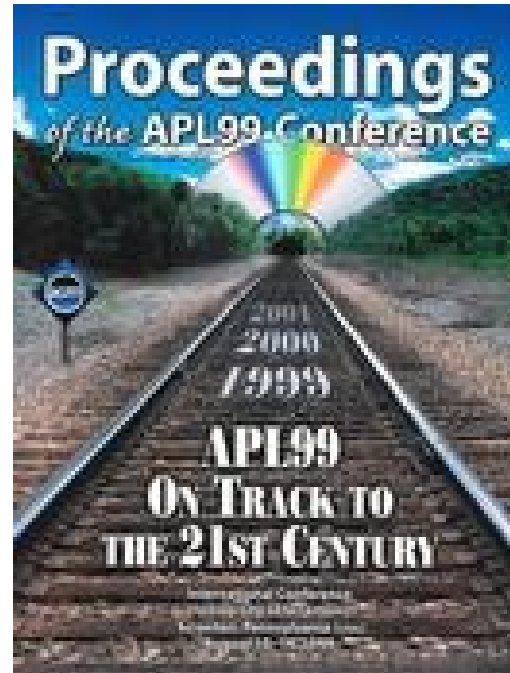




Dyalog North America Meetup, 7 April 2024

Exploring the Dyalog Toolset

Josh David



The APL Advantage

Let me say that again. There is not, nor will there ever be, a programming language that is more productive than APL. Anyone who has used APL and *knows* APL understands this. If another language were as productive as APL, it would be APL. The symbols might be different, and it might execute left-to-right instead of right-to-left, and you might not need a special keyboard, but at its heart, it would still be APL. After all, how can you improve on this:

$$A \leftarrow (+/V) \div \rho V$$

The algorithm for computing the average of a set of numbers is to sum across those numbers and then divide by the number of numbers in that list. APL allows you to perform a direct translation from specification to implementation. You are not sidetracked by DIMENSION statements or by looping control structures. These peculiarities of other languages are inherent weaknesses that reduce productivity. So what do you do?

Gary Bergquist

Leveraging Community Packages

Let's take Dyalog APL out for a spin

- ◆ NYC 2015 Tree Dataset
 - ◆ What's the maximum tree diameter in each borough?