

## SUPPING WITH THE DEVIL

*Graham Oakes reflects on open source's debt to Microsoft.*

---



**Graham Oakes: openness is fundamentally a good thing, but it brings its own challenges**

Here's a question for you: could the open source movement survive without Microsoft?

On the face of it, Microsoft isn't open source's biggest benefactor. Indeed, many of the movement's enthusiasts have very little time for the 'evil empire' and its spawn. So I'll ask the question another way: if there wasn't a visible near-monopoly to rail against, would the open source movement have accrued as many adherents and as much publicity as it has?

After all, if there wasn't an Empire, Luke Skywalker would probably just be a poor farmer stuck somewhere in the deserts of Tatooine.

Back here on Earth, open source is an increasingly important part of the stack. Most major organisations have pockets of Linux lurking somewhere. Credible enterprise applications are starting to appear – content management has long been an open source stronghold, and players like SugarCRM are starting to compete with the traditional ERP vendors. There are a number of decent options to consider if you want an open source ESB.

This seems to me like a good thing. Competition helps keep prices down. Open code makes for clear interfaces, and gives us reference implementations to learn from.

But is there anything companies should be watching for as they integrate these open source applications into their architecture?

At one level, open source applications are no different to any other. If they've been architected appropriately, they can expose services for others to consume. They can themselves consume services. They need monitoring and management, just like anything else.

At another level, access to the code creates tensions. Developers can be tempted to bypass the overhead of creating clean services: instead they call direct into the inner workings of the application. It's easier to accidentally embed assumptions about how a service is implemented. And it's ever so easy to create new functionality by just tweaking the code – at least until the next version is released and you need to make the tweaks all over again.

There's also a potential mismatch of cultures to be managed. Open source communities lean towards informal collaboration rather than top-down direction. They tend to favour developer-led implementation over architect-led design. There's less emphasis on defining interfaces through documentation or markup language, and more on defining them through code. None of this is necessarily bad, but it may not align to your corporate style.

These are interesting paradoxes. Openness is fundamentally a good thing, but it brings its own challenges. Beware of the obvious truths: open source owes at least part of its success to Microsoft. If your SOA is going to succeed, it too may need to live with contradictions.

● *Enterprise Integration, SOA & Web Services Evaluation Centre Expert Dr Graham Oakes is the principal of content management, product development and customer service strategies consultancy Graham Oakes Ltd. Email: [graham@grahamoakes.co.uk](mailto:graham@grahamoakes.co.uk). Website: [www.grahamoakes.co.uk](http://www.grahamoakes.co.uk).*