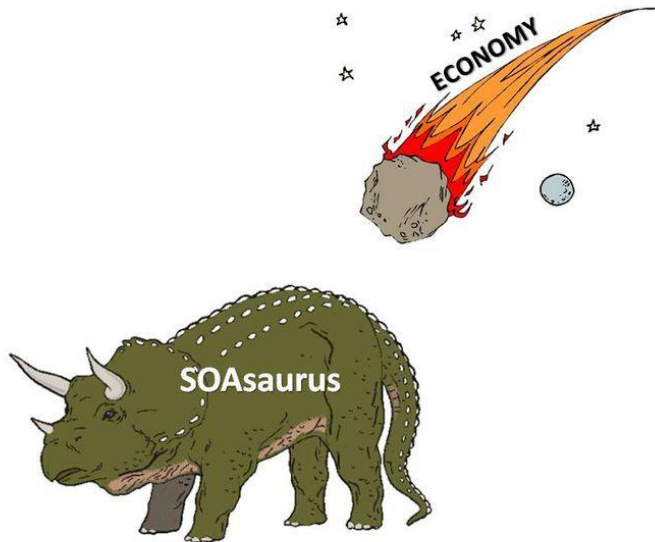


SOA is Dead; Long Live Services

Blogger: [Anne Thomas Manes](#)

Obituary: SOA

SOA met its demise on January 1, 2009, when it was wiped out by the catastrophic impact of the economic recession. SOA is survived by its offspring: mashups, BPM, SaaS, Cloud Computing, and all other architectural approaches that depend on “services”.



Once thought to be the savior of IT, SOA instead turned into a great failed experiment—at least for most organizations. SOA was supposed to reduce costs and increase agility on a massive scale. Except in rare situations, SOA has failed to deliver its promised benefits. After investing millions, IT systems are no better than before. In many organizations, things are worse: costs are higher, projects take longer, and systems are more fragile than ever. The people holding the purse strings have had enough. With the tight budgets of 2009, most organizations have cut funding for their SOA initiatives.

It's time to accept reality. SOA fatigue has turned into SOA disillusionment. Business people no longer believe that SOA will deliver spectacular benefits. “SOA” has become a bad word. It must be removed from our vocabulary.

The demise of SOA is tragic for the IT industry. Organizations desperately need to make architectural improvements to their application portfolios. Service-orientation is a prerequisite for rapid integration of data and business processes; it enables situational development models, such as mashups; and it's the foundational architecture for SaaS and cloud computing. (Imagine shifting aspects of your application portfolio to the cloud without enabling integration between on-premise and off-premise applications.) Although the word “SOA” is dead, the requirement for service-oriented architecture is stronger than ever.

But perhaps that's the challenge: The acronym got in the way. People forgot what SOA stands for. They were too wrapped up in silly technology debates (e.g., “what's the best ESB?” or “WS-* vs. REST”), and they missed the important stuff: architecture and services.

Successful SOA (i.e., application re-architecture) requires disruption to the status quo. SOA is not simply a matter of deploying new technology and building service interfaces to existing applications; it requires redesign of the application portfolio. And it requires a massive shift in the way IT operates. The small select group of organizations that has seen spectacular gains from SOA did so by treating it as an agent of transformation. In each of these success stories, SOA was just one aspect of the transformation effort. And here's the secret to success: SOA needs to be part of something bigger. If it isn't, then you need to ask yourself why you've been doing it.

The latest shiny new technology will not make things better. Incremental integration projects will not lead to significantly reduced costs and increased agility. If you want spectacular gains, then you need to make a spectacular commitment to change. Like [Bechtel](#). It's interesting that the Bechtel story doesn't even use the term “SOA”—it just talks about services.

And that's where we need to concentrate from this point forward: Services.

Posted by [Anne Thomas Manes](#) at 02:02 PM in [SOA](#) | [Permalink](#) [ShareThis](#)

[DIGG THIS](#) [SAVE TO DEL.ICIO.US](#)

Comments

I totally agree with your assessment of the state of the SOA union. Without a commitment to changing the status quo, "SOA" efforts fail. Most efforts focused on the platform stacks and not the enterprise. I've begun to ask "What is your SOA motivation?", if not the business/enterprise then why do it at all.

Posted by: [Tim Vibbert](#) | January 05, 2009 at 07:39 PM



Dave,

Great article, and spot on. In fact, what I hope happens is that people hear "SOA must die, so we can get on with, well... SOA".

I think a some of the confusion can be attributed to companies focusing on the platform to deliver services, instead of the services themselves. And, in trying to be like the Romans and get everyone to do things the same way, all the time.

I think there is beauty and opportunity in the chaos... as you point out, with situational apps, cloud computing, etc... there is a broad horizon for attacking "the business needs" and delivering hugely unlike ever before because of how easy it is becoming to bring information together and present it in standardized ways for the user.

'Nuf said.

David

Progress Software, Actional

Posted by: [David Bressler](#) | January 05, 2009 at 09:46 PM

Anne - excellent summary of my feelings about SOA :) I've actually had conversations with IT professionals about their SOA initiatives where they had trouble defining what SOA stood for, and not understanding that one could have a service-oriented-architecture without buying a "SOA" stack of software.

Another pet-peeve is when every single integration is written as point-to-point with SOA in the middle. Instead of rationalizing the services provided by a system, each time a new client comes along, the SOA team provides a brand new interface with a very use-specific service interface and implementation (possibly even cutting-and-pasting from other implementations against similar parts of the infrastructure). This, to me, defeats the whole point of writing services - that at some point, a new application trying to tie into your services would look at what you've written and largely already have their needs met by what previous projects required from the "SOA" team...

Posted by: [Scott Francis](#) | January 05, 2009 at 09:46 PM

My sincerest apologies, but I must disagree with your assessment of the relationship between the economy and SOA. In fact, it's my feeling that a hurting economy will actually drive MORE adoption of SOA principles across an enterprise, not less. SOA is about three things--effectiveness, efficiency, and agility. If your infrastructure is meeting business needs, and you have lots of money, and the business you're in isn't volatile, then skip SOA and keep doing what you're doing.

However, if your business users are screaming for better capabilities, or your budget has been slashed (or will be slashed), or you're in a volatile line of business where business rules and needs change rapidly, then I suggest that you MUST consider moving, albeit in a measured approach, towards knocking down the vertical application stovepipes and towards an SOA approach.

The one point I totally agree with you is that SOA is disruptive to the organization, but even more so, it will be extremely disruptive to software vendors that continue to bet the farm on large, monolithic applications--SOA will require "disassembling" of these applications into reusable service components.

I tell my associates and clients that a true example of SOA will be the day I can call out to Corel WordPerfect's spellchecker while I'm working in a Microsoft Word document...thanks..r/Chuck Georgo

Posted by: [chuck georgo](#) | [January 05, 2009 at 09:52 PM](#)

The problem that killed SOA is the same now as it was in 1989; no one can decide what services an application provides. Back in those days, I worked on an application development platform for a robotic system in the lab automation market. The chemists insisted that the "primitive operations" provided by a particular syringe pump unit were not just "valve-to-port-1", "valve-to-port-2", "draw" and "expel", but instead were complex and context dependent combinations of these operations (e.g. "wash syringe", and "wash bottle" which assumed a bottle was connected to syringe port 1). The same problem exists now; everyone wants to specify the system in the terms of the problem they are trying to solve, instead of specifying it in terms of the data that the system uses. They do this because they know what their goal is, but not the resources available to apply to the problem, and consequently, the methods needed to apply these resources.

To explore a more typical IT example, an insurance company marketing executive defines a new type of plan to protect individuals from losses caused by theft of their IP services over wireless LANs. He goes to the IT department and asks them to create a web site for it. The insurance company's IT group gets a dozen such requests a day and they are expected to provide instant gratification. Every site request designs the site based on the problem addressed by the plan, and the IT department has to build a whole new system for each one from scratch. The tight delivery deadlines cause the IT department to look for similarities between the sites, and, after a while, they find some; the word "insured" appears in every document, along with words like "damages" and "premium", and they always appear to have the same meaning. After more searching, the IT department discovers that all such sites could be built from a system that exports blocks of data that model the company's idea of an "insured" and a "damage" and a "premium" among other

things. So, just before they start building such a system, they decide to validate their idea with the business experts. That's when the feces hits the fan. The business experts immediately note that a life insurance "insured" is not the same as an automobile insurance "insured". In some states the auto "insured" is the person driving the car, so the "insured" is functionally a car, not a person. The fact that people are named at all is really just a way of identifying the car that is covered ("we insure Jeff's car, not Jeff, but we rate the policy based on Jeff's friends, his neighborhood and education because these things affect who Jeff is likely to let drive his car"). And so it goes, as each type of policy finds a way to prove that its data is unique to its problem. The depressed IT department goes back and looks at the problem again. They find that the relationship between a "policy" and its "insured" is always the same, it's the details of the "insured" that change. And when a policyholder wants to get information about his policy, they always have to find out who the "insured" party is. So, IT solves the problem by replicating source code from project to project, and every time they need a new database, they just take a backup from the last system they built and extend it.

Practices like this prevent any form of "standard insured" from being developed in the insurance industry. No insurance businessman will support this because there are nuances and implications built into the idea of an "insured" that are different for each product type. To see this, what happens when a covered person has no brain activity but is on a life support system; are they dead and does a life insurance policy have to pay, or are they alive and the medical insurance policy has to pay. In reality, life insurance policies assume you're dead when a death certificate is written, because doctors will only sign a document for people they pronounce as "dead". All of the language in a life insurance policy surrounding the word "insured" assumes this to be true. But some medical insurance policies assume you're dead when your organs can be harvested for transplantation, even without a death certificate. And all of the language surrounding their term "insured" works to limit the medical policy's liability within this context. So such a person is neither alive nor dead, and it will take a court case to decide which policy is liable. A software service could only deny all payment given these rules.

These kinds of "business context" assumptions are ultimately coded into the business rules of the systems and they make it very difficult to identify the differences between systems and the data resources the systems manage. As a result, it is difficult to decide what a "service" is and what it should do.

Software developers need to understand that a service can only exist within an operating context; business people need to understand that an address is an address every place it appears. And while software systems might be able to standardize the data of an "insured", and they may be able to standardize the relationship between a policy and its "insured", the business context will always limit how the data is used, and the service will always have to include this business context. This means that every service will either provide just data, or the service will be specific to a business context, but unless the business itself is defined to use standard contexts, the services can never be shared between contexts.

Posted by: Jeff Griffith | [January 06, 2009 at 05:01 AM](#)

Not sure why the traceback did not appear here, but here's a link to my post : SOA ain't dead but it certainly is transforming : <http://blog.dhananjaynene.com/2009/01/soa-aint-dead-but-it-certainly-is-transforming/>

Posted by: [Dhananjay Nene](#) | [January 06, 2009 at 05:01 AM](#)



Thanks for the great postmortem analysis. But it's a Part I only. Part II - discussing the right place of service-orientation.

The roots of SOA problems - people initially misunderstood what SOA stands for. It was crucial for SOA that "services" were substituted by "well-defined interfaces".

Services by the nature are about business and organization structure instead of middleware and applications. And Service-orientation is more about Enterprise Architecture way. The main role of services is glue for business architecture (including organizational and functional views) and IT architecture (from apps to network, servers and storages). There are no such things like business components - there are business services. Business don't know anything like components, but understand what are the services, providers and consumers. It's time to rethink 4+1 approach and to use services instead of artificial business components. It's an only way for IT to be aligned within Business.

SOA as is dead, almost; Long Live Service-Oriented Enterprise Architecture, in any case.

Posted by: [Sergey Orlik](#) | [January 06, 2009 at 05:24 AM](#)



George: I agree with you that the need for application portfolio re-architecture and service-orientation is stronger now than ever before, but [in most organizations] the business guys that hold the purse strings don't understand that, and the IT guys haven't made a compelling business case to support the claim. End result: budgets will get cut and SOA dies.

Posted by: [Anne Thomas Manes](#) | [January 06, 2009 at 05:24 AM](#)

Anne,

When SOA was hot, I would have said SOA was failing because people were talking about it not doing it. Many SOA programs continue to suffer from 'PowerPoint syndrome' and too much emphasis on governance (without execution).

The funny thing is that I've seen SOA move into the engineering groups and out of EA. I'll admit that I have a biased set of projects to survey, but those that I observe have moved towards service orientation in a very, very strong way.

The software developers are thinking about the services early on, often before data models or object models, or in parallel. Finally, I'm seeing SO-Analysis and SO-Design being commonplace. Even my most junior guys will present the clients with an early list of service candidates, usually in a RESTful notation.

What I would say has failed is big, centralized governance programs. From my estimates, you ran about 10+ webinars on BIG SOA GOVERNANCE. It sure would be nice if you'd follow it up in 2009 with 10+ webinars targeted at engineers on the basics of service identification, analysis and design.

SOA didn't fail - however, all attempts by analyst firms to hijack it did.

Long live practical SOA.

Jeff

Posted by: [Jeff Schneider](#) | January 06, 2009 at 06:52 AM



Jeff -- Interesting that you say that. Service modeling is going to be the focus of our SOA research in 2008.

My point is that the term "SOA" is dead -- requests for funding of things called "SOA" will be denied. So as I said, we need to remove "SOA" from our vocabulary.

But service-orientation is more important than ever before. So it's great news that developers are starting to embrace services -- RESTful or otherwise. I haven't seen too many people in the typical large-company development organizations adopt REST yet, but I'm ever hopeful.

Posted by: [Anne Thomas Manes](#) | January 06, 2009 at 07:05 AM

Anne: Absolutely spot on! The gap between business and IT is a significant contributor to SOA's slow/stagnant adoption. And, as an "IT guy, I place the responsibility for this squarely on the shoulders of IT.

If instead of "SOA" had we called it "Generate more revenue and have happier customers by really explaining to the IT staff what it is you really want your IT infrastructure to do without wasting money" then the business side would have probably jumped on it...instead we came up with "service-oriented architecture"...and we're surprised and upset they aren't biting? r/Chuck Georgo ;-)

Posted by: [chuck georgo](#) | January 06, 2009 at 10:17 AM



Ooops, sorry Anne. I saw this post via a tweet from Dave Linthicum, so thought it was his post! Yikes.

Nice post!

David

Posted by: [David Bressler](#) | January 06, 2009 at 10:17 AM

I think Jeff Schneider's comments about practical SOA are spot on. I have said and will continue to say that the technical aspects of good service design were often glossed over or ignored completely. There is still not a wealth of information on this although it is getting a little better.

We went through this same thing with EAI. The promise of the agile, loosely coupled enterprise was highly touted by vendors and analyst alike. But the meat was left out and IT shops went out and bought huge expensive platforms and then proceeded to recreate the same spaghetti they were trying to get away from.

In depth conversations are needed if effective services are ever going to become mainstream for IT organizations.

markg

Posted by: [Mark Griffin](#) | [January 06, 2009 at 10:23 AM](#)

Could be that the enterprises that dismiss SOA, nomenclature or computing shift, will be all the more ready to hand off more of their IT functions to the SaaS and cloud providers that do SOA well and pervasively. In other words, you'll do SOA one way or another ... it's just whether it's your competency or your cloud providers.'

Smartest enterprises will do both and in the correct combination. The rest will have exorbitantly expensive IT that does not respond to change nor exploits the low-cost cloud alternatives. And that spells bankruptcy (sans bailouts).

Posted by: [Dana Gardner](#) | [January 06, 2009 at 12:10 PM](#)

> SOA needs to be part of something bigger.

Agree.

Most of the SOA-Solutions are like classical programs in the web - a static set of tables as background, static pages. Every customer has the same - the business processes has to adapt to the structure of the applications.

Terrible.

The other way is to use a system as background, which allows to create specific, individual tables. So every customer has its own set of tables, its own name conventions, labels etc. The applications can be adapted to the business processes and their changes.

And the service of implementing these changes quickly (in one, two days) is mission-critical. So a customer may start using four, five tables - and a year later new business activities are coming, so the web solution grows up to 20 or 40 tables.

Posted by: [Juergen Auer](#) | [January 06, 2009 at 12:10 PM](#)



Anne:

My response:

<http://technoracle.blogspot.com/2009/01/soa-is-not-dead-but-complexity-is.html>.

Duane Nickull
Sr. Technical Evangelist, Adobe Systems, Inc.
Chair - OASIS SOA RM Technical Committee

Posted by: [Duane Nickull](#) | [January 06, 2009 at 12:12 PM](#)

Here's my feedback: SOA is not dead, we're still in the early adapter phase
(<http://www.andrejkoelwijn.com/wp/2009/01/06/soa-is-not-dead-were-still-in-the-early-adaptor-fase/>)

Posted by: [andrej](#) | [January 06, 2009 at 07:27 PM](#)

Anne,

Based on the fact that you think SOA is dead, you might really want to consider updating your LinkedIn profile. :-)

Regards,

RonK

Posted by: [Ronk](#) | [January 06, 2009 at 07:27 PM](#)

Hi all,

We are still in a time, where most people assume they have a SOA when they got a web service up and running (due to the word "service" in both terms).

I would assume that 90% using the word just play buzzword bingo and have actually no clue about its meaning.

Because SOA has not even really started yet.

SOA is a prerequisite related to enterprise ESB and BPM. If SOA goes, so do ESB and BPM.

I see the evolution of "Services" as follows

1.) (Web) Services

This is a classical 3-tier architecture

2.) SOA

You break up the static link between services and client by adding some sort of service registry

3.) ESB

You connect all your services to an ESB. Instead of searching the service of your choice yourself you let the ESB Hub decide.

4.) BPM

On top of your ESB you add BPM to design your IT to your business processes. You are able to monitor and change your processes any time.

Web Services are some sort of settled.

SOA is still young technology, but we start to get idea what could be a SOA.

ESB? What is ESB? Nothing but buzzword so far. In the moment just all former EAI vendors relabel their old fat systems with the current buzzword word "ESB". Same for BPM.

I see that IT will be continue to evolve towards BPM. Heaven is, when our services reflect exactly the users business processes and we can change our business processes simultaneously with our IT .

Business Process and IT must be one.

Btw.

If economy affects technology it means you got the wrong consultant!!!

SaaS? Forget it. Its ASP relabeled. Better stick to the old term.

Cheers,
Tobias

Posted by: [Tobias Manthey](#) | January 06, 2009 at 07:27 PM

Hi Anne,

Based on my travel schedule over the past several months, which has been about 90% of my time going to meet with customers who are actively engaged in funded SOA projects, I would say SOA is alive and well. One could argue that there are discretionary projects being put on hold due to budgeting restrictions, but that has little to do with whether they are SOA.

But hey what the heck, I'll join in -

SOA is Dead. Long live service-orientation!and architectural patterns and practices to ensure that an IT organization has consistency across applications that are built using service orientation....Oh and I guess we'll need a means for ensuring that those services and their associated artifacts can be stored, versioned, secured, and accessible from a variety of interface technologies and protocols. And we can't build it all from scratch so we'll need to be able to leverage our investments in existing application assets....then we'll need to compose services together, and perhaps even orchestrate the interactions between them....We'll also need to have a working plan in place to communicate between IT and the business to ensure that the right services are being built at the right level of granularity so that applications, which leverage those services, can have the flexibility and agility to meet the increasing and ever changing demands from the business. And a way to govern that process.

Long live service-orientation! Bravo!

Dave

Posted by: [Dave Chappell](#) | January 06, 2009 at 07:27 PM

As the Zapthink guys keep saying "SOA is something you do, not something you buy". The big problem with big SOA projects has been companies insisting that SOA can be "purchased" and then "installed and configured". Most of the failed SOA to me looks like

nothing more than big integration projects that have gone horribly wrong - They never were SOA in the first place.

Hopefully SOA isn't dead but the chronic misunderstanding in the marketplace about SOA is.

Posted by: Lee | [January 06, 2009 at 07:39 PM](#)

Nice piece. I believe strongly that the over-hype of the term was actually holding back the real practice. The principles are valid and essential, but the expectations and agendas were killers. With the term and its associated distractions dimming, hopefully we can just get on with it.

This phenomenon has many parallels to what happened with AI. As I wrote in <http://drjbutler.wordpress.com/2009/01/06/soa-in-good-eternal-company> the term "AI" became a pariah while its byproducts are now pervasive.

Posted by: Jim Butler | [January 06, 2009 at 07:39 PM](#)

Dear Anne,

What a boat load of regurgitated crap. What does this '...SOA requires disruption to the status quo' mean anyways?! Perhaps no one really sat you down and explained the meaning behind the term SOA. Anyone intelligent could have pointed out that SOA is not about services at all and has never been about services or the technology. That's only a 1/3 of the story. The other two components are: business processes re-structuring and operations re-structuring. Perhaps you were listening to some IT contractor out of a temp-hire agency.

I'm afraid that you've totally missed what knowledgeable practitioners had always been preaching. SOA, as you define it, was only ever limited to people who never truly understood the vision of it all and, instead, saw it as a cure for everything. So, I guess, in essence you've just discovered the concept of 'hype'. Bravo! Well done.

This is precisely why I've come to expect nothing of insight from any of the research vendors. You do a shallow analysis, which often is incomplete and lacks any real substantiative details, and then proclaim it as some newly discovered wisdom.

Truly disappointed,
-harvey

Posted by: Harvey Stage | [January 06, 2009 at 07:39 PM](#)

SOA and "Services" are fundamentally the same thing. The ability to expose an API via web services.

Posted by: [automotive crm guy](#) | [January 06, 2009 at 07:39 PM](#)



IT and Business have been head to head ever since business became frustrated at IT for taking too long to add an extra field to their mainframe screens.

What do IT expect? SOA is about IT doing it right and IT should be doing it right whatever the acronym. However, doing it right takes too long, costs too much, risks complete failure and time has proven, can't always be done at all.

The problem goes around and around but one thing that 2009 is all about is making do with what you already have and making it work a lot better. Shareholders will expect that. If you are laying off people to save money, don't expect to be let off lightly when you lose customers. Make what you have work better by optimizing the employees (end users) desktops and buy yourself time. There are \$trillions of savings to be had TODAY with iterative and agile integration products. Keep planning for the perfect architecture but don't ignore the fact you want to remain in business to see it through - eventually....

SOA is a very long journey. A journey that will see businesses go under or remain uncompetitive unless you look to other solutions to business problems whilst you take that journey. They are out there if you look.

<http://blog.tmcnet.com/desktop-productivity/>

Francis

Posted by: [Francis Carden](#) | January 06, 2009 at 07:39 PM

[Next »](#)



Hi Anne,
Here's my response: <http://www.andrejkoelwijn.com/wp/2009/01/06/soa-is-not-dead-were-still-in-the-early-adaptor-fase/>
Regards,
Andrej

Posted by: [Andrej](#) | January 06, 2009 at 07:39 PM



Lol! SOA is dead, ah!

Let's create one more name for something what is the same, but sounds different, but also with service.

"A, B, C, D...."

"stop"

"L"

"Late, Lost ... hmmm, "

"Long Life Service!"

"Spectacular, spectacular!"

Posted by: [nebupolzar](#) | January 06, 2009 at 07:39 PM



I once went to a prospect who claimed to have implemented a "SOA", but he was complaining everything was now much more fragile: Basically before they had a rudimentary but well-decoupled and robust system, it was possible for sales people to work off-line and then reconcile the day while at home, but it was hard to maintain and integrate new things. The new self-proclaimed SOA was based on the single silly idea of putting synchronous SOAP Ws everywhere, so at the end the whole new system is totally synchronous and fragile, as the failure of one single component means the stop for all the processes (as I said everything has been made synchronous). Of course they now blame SOA for that and want the previous fault tolerance back...

Now is that the failure of SOA of the plain defeat of intelligence?

Posted by: [Maurizio](#) | [January 06, 2009 at 08:04 PM](#)



I am not sure why this blog doesn't take trackbacks; our response is at <http://www.vosibilities.com/soa/giving-soa-a-terminology-niptuck/2009/01/06/>.

Alex Neihaus
Active Endpoints, Inc.

Posted by: [Alex Neihaus](#) | [January 06, 2009 at 08:04 PM](#)



SOA is not dead, just change the name. If you ask me which name, I don't know. I agree with many of the comments, SOA is a very particular way to redefine your application portfolio and more: Change the way of IT and business units are related and exchange information. Maybe the flow between this two guys are the complexity of SOA and generate bad feelings about SOA.

Posted by: [R.Sierra](#) | [January 06, 2009 at 09:15 PM](#)



Hi Anne. I don't think SOA as a set of core principles is dead. Sure the term has been hijacked by many people (analysts included), but that doesn't mean it's dead. That's not what I'm seeing in the field either. Anyway, I thought I'd add something here <http://markclittle.blogspot.com/2009/01/soa-is-dead-again.html>

Posted by: [Mark Little](#) | [January 07, 2009 at 05:52 AM](#)



I would say that SOA might not be dead yet but I am hearing of alot of SOA projects being delayed or killed off. It might be more appropriate to say that SOA may be going into hibernation for a bit.

Posted by: [Craig](#) | [January 07, 2009 at 05:52 AM](#)



Trackback also not worked for the following post on the ARIS BPM Blog:

<http://www.arisblog.com/2009/01/07/technical-soa-is-dead/>

Posted by: [Sebastian](#) | January 07, 2009 at 05:52 AM



Trackbacks work -- I just have to approve them. -- Anne

Posted by: [Anne Thomas Manes](#) | January 07, 2009 at 05:54 AM



SOA is dead because a long time ago, engineering discipline died.

The death of engineering discipline is what led to people believing that deploying technology solved architectural problems, that vendors are the experts and know it all and that there's no need to worry about maintaining clear definitions of terms and acronyms such as SOA or Web 2.0

IT has long sought to have any warm body at a keyboard, sweeping aside the idea of appropriate qualification, experience and expertise in favour of being up-to-date with whatever are the "hot" technologies, architecture in a box such as J2EE and the idea that cranking code as fast as possible is all that matters.

IT has once again shot itself in the foot, blamed the tools and run over the hill to the next shiney thing. I'm betting that'll happen all over again real soon....

Posted by: [Dan Creswell](#) | January 07, 2009 at 07:08 AM



What a stupid article. Only nonsense. I wanted to write an answer but then I say the reply from Harvey. I totally agree with him. There are too many people discussing about this topic who have absolutely no idea. SOA was always a shift in thinking, never a technology. Just because some did it wrong this does not mean that SOA is dead. In fact it is not what I can confirm from my day to day practice.

Posted by: [Jochen](#) | January 07, 2009 at 09:35 AM



A bittersweet emotion came over me when I heard that some of the most renowned of industry experts, even those heavily invested into the SOA movement, have now declared that SOA officially died on 1 Jan 2009...

more:

<http://geekswithblogs.net/MSARCH/archive/2009/01/07/128458.aspx>

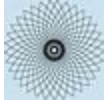
Posted by: [Jim Kita](#) | [January 07, 2009 at 09:35 AM](#)



harvey said: "You do a shallow analysis, which often is incomplete and lacks any real substantiative details, and then proclaim it as some newly discovered wisdom."

...and, like sheep, many form a flock and follow along without question.

Posted by: [Erik](#) | [January 07, 2009 at 11:00 AM](#)



I think what Anne means is ESB or service bus is dead. Long live the service cloud.

Posted by: [vikas](#) | [January 07, 2009 at 07:45 PM](#)



Anne, as a Service Owner, I will offer you my support of your premise. I run a service owning team in a large enterprise and have spent a lot of wasted time evaluating, reevaluating, and re-reevaluating SOA-branded products that solve many imaginary problems. Its cost my team thousands of hours of distractions to work with the SOA-driven outside influences, usually ignoring the context and purpose of the actual Services. Many of the products add dubious value but cost significant time and money.

Much of 2006-2008 we've spent building up our service inventory using a manufacturing paradigm migrating into a managing and operating paradigm (indecently, we've chosen Software Product Lines as our manufacturing paradigm), we've built unit cost models and proven reuse savings with little interest from our executive leadership who are typically more enamored with the SOA vendor product rather than the details about the Service. From a manufacturing standpoint, it only makes sense to focus on the actual product, LEAN manufacturing prescribes removing the waste factors, focus on the basics necessary to manufacture and manage the core product first, then move into the more exotic areas later. SOA started backwards, pay for all the exotic infrastructure, repositories, 'buses', etc.. first.

As you stated the "why you're doing it?" part was never just to do SOA. We built our Services specifically with other goals in mind, we were retiring older integration technology, leveraging price brakes on new technologies, decoupling between user applications and Core data owning systems, adding alternate availability architecture options, etc... We have been very successful at all those goals, hence we believe we have success. Added benefits are that new delivery efforts move a lot faster using services then older integration methods, and we hope next generation projects can leverage BPM and Mashup technologies.

However, we did waste a lot of time on the shiny new SOA things. Had we invested in the service inventory instead of the shiny-new things we would have a more robust inventory.

Overall...good advice...focus on the Service first.

Let the exotic come later.

Posted by: [Carl Parziale](#) | [January 07, 2009 at 07:45 PM](#)



Since trackback didn't work, please see my response
here: http://blogs.gartner.com/nick_gall/2009/01/06/long-live-the-web/

Posted by: [Nick Gall](#) | January 07, 2009 at 07:45 PM



Anne,

My post "Understanding SOA" from last April sheds some light on a few of the issues raised in this thread. It discusses:

SOA definition
ARCHITECTURE
SOA ARCHITECTURE PRINCIPLES
TECHNOLOGIES
SERVICE DISCOVERY
IDENTITY MANAGEMENT
CONCLUSION

<http://www.keystonesandrivets.com/kar/2008/04/understanding-s.html>

PJW

Posted by: [Paul Wallis](#) | January 07, 2009 at 07:49 PM



SOA has a different context in the commercial world than it does in the Federal space. SOA within the DoD includes but not limited too: The adoption of Open Source Methodologies, Best Patterns and Practices, and Standards. The fall of the economy forces the Government to analyze how and where they can begin to cut cost. The cost is reduced dramatically when SOA is applied to Software Projects within the DoD. What a concept, expose all data and apps via services between agencies. SOA is not dead within the DoD as it may be within the commercial space. We still have a long way to go within the Federal Space.

Posted by: [Mike](#) | January 07, 2009 at 07:51 PM



Anne, I see your recession, raise to a depression and still contend SOA will remain important. Happy New Year! :-)

http://blogs.jboss.com/blog/pfricke/2009/01/06/Economic_Depression_and_the_Rise_of_Open_Source_SOA_and_Business_Rules.txt

Posted by: [Pierre Fricke](#) | January 08, 2009 at 06:19 AM



Good discussion! Here's an article I wrote on the true value of SOA - its business services:
"Your SOA needs a business case"

<http://www.via-nova-architectura.org/files/magazine/Baarda.pdf>

Posted by: Piet Jan Baarda | January 08, 2009 at 06:19 AM



Interesting article, however like many here I fail to understand how SOA can be dead when you promulgate the raise of service. If you have services you need architectural models to use and manage them in.

So yes, maybe changes in the way it's sold and evolution of its structure (let's face it we'll never stop arguing over the pros and cons of architectural models or technologies), but death of an IT domain is a bit stretch... for the death of a buzz, only time will tell us.

Posted by: Pierre de Leusse | January 08, 2009 at 06:19 AM



'Anne' is a new word in dictionary meaning 'insight + courage'.

<http://aminsblog.wordpress.com/2009/01/08/soa-is-dead-long-live-service-oriented-architecture/>

Posted by: Amin Abbasopour | January 08, 2009 at 09:24 AM



Ah! yes! A decent new resolution for all!

Seriously, however, most organization fail on SOA due to the "wishful" thinking of "one-stop shopping" solution. There are two main areas for SOA to succeed: (1) Service alignment between business and IT, (2) Service development within IT. And... we all know about the "openness" or lack thereof of the folks in the two disciplines to collaborate. If that hurdle can be overcome, then the service contracts hit the IT department just to find that IT may not be ready to complete it. Such simple things as UDDI registry or SOAP stack etc. are not there just yet - remember IBM came out with WSSR due to their claim of UDDI's inflexibility in an agile and actionable SOA environment. So now... Anne is right then? Even with its demise, SOA still left a decent framework for IT - don't you think? Ask Mr. Erl?

I like Tobias' breakdown of the components for a "pre-SOA" environment, but prefer in the following sequence:

- (1) BPM (most platforms seem mature.)
- (2) Services (web or not)
- (3) ESB (for legacy systems)
- (4) maybe SOA - at least something close to it.

At least these are my 2009's resolution!

-Duc

Posted by: Duc Nguyen | January 08, 2009 at 11:01 AM



SOA is not dead though it's reputation has been tarnished by exploitation and misguided attempts at implementation. See my blog "SOA Is Not Dead--Just Misunderstood" at http://www.eds.com/sites/cs/blogs/eds_next_big_thing_blog/archive/2009/01/06/soa-is-not-dead-just-misunderstood.aspx?CommentPosted=true#commentmessage

Posted by: Fred Cummins | January 08, 2009 at 02:47 PM



Trackback: <http://www.arisblog.com/2009/01/07/technical-soa-is-dead/>

Posted by: Sebastian | January 09, 2009 at 05:33 AM

[« Previous](#) | [Next »](#)



SOA is as SOA does. Great article, and timely. My company has embarked on a major SOA initiative, which (don't laugh) inspired me to write a song about SOA. This might, in fact, be the first rock-and-roll song ever about SOA. I call this new genre "Nerd Rock".

Have a listen: <http://kompoz.com/6618>

-Raf

Posted by: Raf | January 09, 2009 at 05:33 AM



I have conducted a Service Oriented Autopsy, including the important question "Was SOA Bunburying?".

<http://stevenimmons.org/blogs/steve-nimmons/09012009/soa-service-oriented-autopsy>

Posted by: Steve Nimmons | January 09, 2009 at 08:09 AM



Here's another one on SOA during a recession: A RECESSION IS A BLESSING IN DISGUISE !!

http://eng.dya.info/Images/Recession_Proof_SO_V13a_Berg_Baarda_%20Jan_2009_UK_tcm14-50724.pdf

Posted by: Piet Jan Baarda | January 09, 2009 at 08:09 AM



SOA is the underlying implementation that enables Services. The goal has always been to create business services that are flexible enough to respond to changing business requirements without having to engage an army of programmers. I suggest readers take a

moment to review the Open Group Service Integration Maturity Model <http://www.opengroup.org/projects/osimm/doc.tpl?CALLER=index.tpl&dcat=14&gid=17990> or just google OSIMM. OSIMM helps to provide the technology, business and governance requirements for successfully implementing services.

Posted by: Andras Szakal | [January 09, 2009 at 09:34 AM](#)



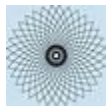
So... The point is that the three letter acronym 'SOA' is 'dead', but everything that it stands for is right.. I don't see the need to waste time talking semantics..

I also don't understand why people think REST is some god-send, it's just Web Services without the SOAP Envelope and hijacking the HTTP protocol (HTTP PUT? Cmon) instead of just clearly defining your interfaces via WSDL.

Anyway, the real reason that projects (SOA/Non-SOA) fail is that IT consistently fails to talk about projects in a manner that the business can understand. There's a lack of correlation between "If we, the business, give you \$x for this project, what will the ROI be in x years?".

It's a larger IT issue, not one thats constrained to a particular approach.

Posted by: Dan Zrobok | [January 09, 2009 at 09:34 AM](#)



Anne is correct about SOA being dead. It failed for the same reason that DCE failed - and the reasons are not technical. The reason for its failure is based on internal politics, culture and business rules complexity.

Everyone agrees that SOA requires big changes in culture and the internal workings of the company in terms of the way IT services are delivered. This translates to a big impact in terms of resources and \$'s as well as a general consensus between IT and business units on the correct way forward.

So what does a company do when it says "we can not afford this and we actually want IT to reduce their budget next year by 10% or we will look at outsourcing"?

Smart IT people will recognize that using terms that few people really understand e.g. SOA, web services etc., is not going to sell well in 2009. Especially when the huge bulk of the companies prod applications use traditional 3GL's (Cobol, Basic, PL/1, VB6 etc).

Smart IT people will recognize that it will be much more efficient to sit down with the key staff of each of their BU's to really understand their issues and challenges. Once is done, then the IT staff need to determine, with open minds, whether it is better to use traditional methods (web enablng existing apps or create new ways with OO technologies (.Net, J2EE) or perhaps just plain and simple 3GL programming.

Its back to basics.

Posted by: Kerry | [January 11, 2009 at 11:14 AM](#)



Anyone who believes "SOA is dead" needs to check out www.cio.com/document/pdfs/2009_state_of_the_cio_highlights.pdf
Looking at this report it is clear that SOA or "business process management" or the need to "align IT with business" are some of the top priorities of the CIO community.

Posted by: IK | [January 11, 2009 at 11:14 AM](#)



Hi Anne,

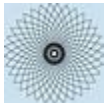
Well written. I agree. You have inspired me to blog about this again. In my last SOA post, What is SOA, Really.... A Sacred Omnipotent Acronym, I touched on a few of the reasons that SOA will die.

<http://www.thecepblog.com/2008/11/10/what-is-soa-really-a-sacred-omnipotent-acronym/>

SOA might not be "pronounced dead" yet, and there are always a handful of analysts and seller who will continue CPR on the corpse-of-soa.

Yours faithfully, Tim
www.thecepblog.com
www.unix.com

Posted by: Tim Bass | [January 11, 2009 at 11:14 AM](#)



Hi Anne,

OK, thanks for the motivation! Here is my follow-on one your excellent post!

SOA in Cardiac Arrest, Long Live Services

<http://www.thecepblog.com/2009/01/11/soa-in-cardiac-arrest-long-live-services/>

Enjoy!

Yours faithfully, Tim
www.thecepblog.com
www.unix.com

Posted by: Tim Bass | [January 11, 2009 at 02:37 PM](#)



We have come up with ever more expensive and elaborate ways of doing procedure calls.

Please indulge me for a moment. The first subroutine library I ever used was made up of individually coiled rolls of paper tape hanging on coat hooks beside the computer. You wanted to use one? Take it off the hook, inline it into your own program on tape, coil it back up and replace it on the hook.

It's been downhill ever since! We have found more and more ways of invoking these - pushing the binding later and later, making them location/implementation independent, but all the time essentially using a call/return paradigm.

That simply isn't how the world works - at least for the most part. In most business and human interaction we work asynchronously. I tell you that something has happened - you take action (appropriately, I hope). We seem to use request/response to get information - "What's my balance?" - I want some info and am prepared to wait for it. Most of the time, however, while I may want things to happen, I don't necessarily want to stand there arms folded doing nothing else while it happens. Especially not using the kind of approach like, "Please go and tell our foreign exchange student to clean his room and I will wait here until he is done" - something I may have to do when I don't speak enough of the exchange student's language.

There's far too much friction in request/response and therein lies the major problem with most SOA experiments. Yes there have been lots of successes, but tightly coupling to a predefined interface and then using a heavyweight RPC is probably not what many had in mind when they started on the SOA journey.

So if we now step back and look at what we have been trying to do with SOA, we should divorce ourselves from the implementations - not think about SOA as the alphabet soup of SOAP/WSDL/WS* and refocus on business and business conversation. Using frameworks (thinking frameworklike VPEC-T) to help us understand how to get the business and IT to focus on the same ends, determine the proper means is where we ought to be taking SOA. Remember the A in SOA is architecture.

Posted by: [Chris Bird](#) | [January 11, 2009 at 06:47 PM](#)



Thanks for this. Provocative, but necessary. IT executives will need many such clouts over the head to come to their senses, because the essence of SOA as you put it is just too scary: "SOA ... requires redesign of the application portfolio ... a massive shift in the way IT operates."

In a way this "demise" reminds me of the disillusionment we went through with software reuse, many years ago. Then, we dreamed of building applications from modular business components. Today, we dream of building applications from modular business services.

In both cases, we allowed ourselves to become consumed with the technology aspects of the problem and let the far thornier management and organizational issues sort themselves out, which of course they never did.

Reuse and SOA both demand dramatic shifts in organizational structure, project sponsorship / funding / management, and product planning. And they demand comprehensive, up-front domain analysis and disciplined, long-term planning for reuse, notions that are out of step with business managers' endless one-off demands of "Build this app for me right now."

Such a waste of education. So many lessons from the days of software reuse are applicable to SOA, and we dismiss them because the problem looks different enough to be new.

A tad more detail here, if

interested: <http://bluecollararchitect.blogspot.com/2009/01/soas-funny-hat.html>

Posted by: Jonathan Ross | [January 11, 2009 at 09:58 PM](#)



To Dan Zrobok: Every business gets the IT that it deserves! With business traditionally only focussed on very specific solutions without any regard for overall coherence, consistency and quality only very mature organizations will get SOA benefits. Don't feel guilty as an IT person. Immature businesses will chop off your head if you talk about coherence and architecture in general. Well... they got it coming. Business grow up!

Posted by: PJ | [January 12, 2009 at 09:29 AM](#)



I scanned the blog by Anne, SOA is not dead, in contrary I believe people will leverage it to look into optimizing their business processing by leveraging SOA practices to monitor events related to processes and business situations. I expect there will be greater focus on re-engineering to save cost, and to re-engineer they would need know the current situation and be more proactive. The new suite of technologies on business activity and events monitoring will assist the executives to have a dashboard to make those decisions proactively.

Posted by: [Kadeer Beg, Prolifics CTO](#) | [January 12, 2009 at 09:29 AM](#)



"SOA needs to be part of something bigger."

In other words, SOA doesn't "require" anything? For example, the view is "we need to restructure our business, and we've decided to use SO principles as the way to guide that restructuring." As opposed to "we want to 'do SOA' therefore we are going to restructure our business."

SOA isn't the goal. Its the means to an end. Right?

Posted by: Rob Eamon | [January 12, 2009 at 12:33 PM](#)



Shouldn't the dinosaur be labeled IBM?

Posted by: [Robert S. Robbins](#) | [January 12, 2009 at 12:33 PM](#)



First of all - don't worry - if Anne Thomas is saying SOA is dead, she means that SOA has already reincarnated under different 'Avatar' ...whole response here: <http://blog.sumbera.com/2009/01/13/soa-reincarnation/>

Posted by: [Stanislav Sumbera](#) | [January 13, 2009 at 09:20 AM](#)



SOA is dead. So, if we are lead to believe that something is dead, first we must try and understand what it is that is actually dead, then pursue the reasons as to why it died.

Too often we have been lead to believe that unless you pursue SOA with an enterprise-driven approach focused on building reusable services, then you are not doing SOA.

Thankfully, in the real world, there is a single way to do SOA - many customers start out with SOA with a Project-Driven or a Infrastructure-Driven approach to SOA.

So it may perhaps be true that the Enterprise-driven approach to SOA may be less common today. However, thankfully, the enterprise-driven approach is not the only way to do SOA, and so SOA is not dead afterall!

For more about the different approaches to SOA, you may want to read our article in SOA World Magazine entitled "Going Beyond Project-Driven SOA"

<http://soa.sys-con.com/node/698981>

Posted by: Mohamad Afshar | [January 13, 2009 at 10:13 AM](#)



Rsponding to Mohamad Afshar: Our research shows that project-driven SOA initiatives fail as frequently as enterprise-driven ones. Although you may execute a number of successful projects (typically SOI rather than SOA), from an overall standing, the IT department still does not gain significantly reduced costs or increased agility. Within 18-24 months, these project-driven initiatives tend to stall out because the business fails to recognize significant value from the effort. As I said, the only truly successful SOA initiatives are the ones that are part of a much larger IT transformation effort.

Posted by: [Anne Thomas Manes](#) | [January 13, 2009 at 10:24 AM](#)

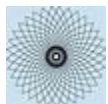


Hi, Anne!

Excellent article, and one that (as you probably know better than most) expresses my sentiments about SOA as well. I've written a reply at <http://broadcast.oreilly.com/2009/01/soa-is-dead-its-about-time.html>.

Kurt Cagle

Posted by: [Kurt Cagle](#) | [January 13, 2009 at 11:54 AM](#)



This has been a great discussion with excellent comments, my favorite so far this year! I have recently posted something along the order of Capability as a Service (CaaS) is the "New SOA" in,

IT Infrastructure: Capability as a Service

<http://www.thecepblog.com/2009/01/14/it-infrastructure-capability-as-a-service/>

Yours faithfully, Tim
www.thecepblog.com
www.unix.com

Posted by: [Tim Bass](#) | [January 14, 2009 at 02:28 PM](#)



Hi Anne,

You are dead-on right and I feel that the paradigm was never understood and rather misused by all including implementers and vendors, since I was involved with one of the first "Good" implementations in the industry that is still going strong on ROI. As you said, coming to think of it we were lucky since SOA was part of a big picture for an urgent hunger for integration thus not being fixated on SOA alone.

SOA might die as a buzz word but the services architectural approach it brought along is revolutionary and will become irreplaceable for businesses in the future. The economic recession will rather force firms to use these concepts given the rising mergers & acquisitions, the need to integrate the enterprises, and the competitive edge business will seek with faster than ever product delivery requirements due competition in the global economy.

I think this is the time for the visionary organization to invest in moving towards this capability and be ready for the onslaught when the recession eases, which we know has to. My take towards this is to always start grounds up putting in place the basics of service architecture capitalizing on existing investments ensuring at a minimum your services across systems can facilitate enterprise integration and can plug and play as the architecture matures for the enterprise, instead of top heavy investments into middleware vendors, protracted governance models, etc.

Posted by: [Anupam](#) | [January 14, 2009 at 03:52 PM](#)

Hi Anne,

I can't believe that you are the same person that used to raise millions of VC in your former role as executive of Systinet. But that's the beauty of being an analyst. You have to re-assess the market again and again and change your mind if the market (hype) changed.

There are many indications that Dinosaurs died in a big bang of climate change. And that's my major point: The change of the economy and the climate change that killed the dinosaurs had nothing in common. So what has the SOAsaurus and the real dinosaurs in common then?

The dinosaurs and the SOAsaurus are significant proof of concepts for major innovations. The dinosaurs have been the first very large vertebrate with a backbone or spinal cord, a brain case, and an internal skeleton. Even though the direct further "development" of a pure dinosaur failed, the concept was a major breakthrough and paved the way for the vertebrates including human beings. (I am not a biologist; please take this as an analogy...). I believe SOA has not suddenly died. Everybody who still believed (until your blog) that SOA

would bring a cost saving or business value as a stand alone initiative missed the evolution of SOA of the past two years. SOA proofed that it is the outstanding concept of integration across application silos, across technology tanks and vendors. I totally agree with you that pure SOA has no return of invest, but this comes from the BPM, ERP, Business Event, BI and other systems that are directly related to business value contributions.

How many muscles would an upright walking animal or human being require without having a skeleton and backbone? Bones, Muscles, and a well protected brain, that's the concept that survives in symbiosis. Therefore SOA will decline the hype - no question - but continue to be a major ingredient of future enterprise software concepts.

Stefan from Forrester

Posted by: [Stefan Ried, Ph.D.](#) | [January 15, 2009 at 03:14 AM](#)



The picture will be better reflecting the situation if the "SOAsaurus" will be changed to "ESBsaurus". It is quite unfortunate that the SOA - which is the pattern of EA, became a synonym of ESB in the heads of CIOs.

Posted by: [Dr. Yigal Gur](#) | [January 15, 2009 at 10:52 AM](#)



Anne,

I couldn't resist jumping on this thread. The term "SOA" ended up being an overly-hyped term created by software and analyst firms to get some air time and sell products.

Service Oriented design and approaches have been around way before the term SOA became popular. I published an article comparing object and service oriented approaches back in 2000.

The important aspect of SOA is the loosely coupled design that facilitated reuse. This takes experience, discipline, business understanding and some foresight to understand what will make a service reusable. It is not something you can buy off the shelf.

If someone sold their business an SOA initiative promising all the benefits that came along with the hype, then they were investing in technology for the wrong reasons and were setup to fail from the beginning because there is no way to deliver on the overinflated expectations.

I hope no one out there is selling their business something under the "Web 2.0" banner. IT professionals getting leading their business partners to invest in this manner are investing in technology for technology sake and plays into the already poor perception business users have of IT professionals as overbuilding propeller heads.

Anne, I think you were trying to say something similar but they way in which you say it also plays into the hype of what is a solid architecture practice. Now, I have to deal with business users who say we should not be doing SOA because it is dead.

Mike

Posted by: [Mike Alvarez](#) | [January 15, 2009 at 12:48 PM](#)



Anne,

An excellent, thought provoking article. But, whilst sadly the phrase SOA might be somewhat tarnished, I cannot accept that "SOA has failed to deliver its promised benefits". Possibly what you and I mean by SOA are two different things?

Whatever, I think that SOA suffers very much from the fact that it contains the word Architecture. This leads one immediately to think an SOA system simply (!) requires a new architecture.

That is not the case. SOA is a fundamental shift in the way most enterprises co-operate internally, and the reason SOA is perceived to have failed is because few enterprises have had the balls to change their management and communication structures to allow SOA to do what it does; deliver greater agility through a tighter coupling of business and IT. Yes, in this case, tighter coupling is a good thing!

I believe that companies typically deliver computer systems that broadly reflect their internal communication structures. Whilst these structures remain unchanged it is hardly surprising that SOA is perceived to have failed to deliver. I believe the truth is few have understood it and even fewer have implemented it.

Kind regards,

Phil Bowker

Posted by: Phil Bowker | [January 15, 2009 at 01:07 PM](#)

[« Previous](#) | [Next »](#)



Anne,

Interesting post. After thinking through this for a few days, I decided to perform an autopsy and determine the cause of SOA's death:

<http://blog.dennyboynton.com/post/An-Autopsy-on-SOA.aspx>

Posted by: [Denny Boynton](#) | [January 15, 2009 at 02:12 PM](#)



I posted this on InfoQ also but for those who don't get over there:

I Disagree

Seems like every job description out there now has SOA in it, I don't see SOA as being dead at all. I do see issues with customers trying to implement large green solutions provided by vendors. As happens with any significant technology change, it takes the vendors time to perfect the solution in terms of quality, documentation, etc. As I wrote in an article here awhile ago, I'm not sure the ESB was a great idea but the ESB is not SOA so to pronounce

SOA dead is not making sense to me right now. If you want to make statements due to the economy then just pronounce technology dead, no?

Posted by: John Harby | [January 15, 2009 at 02:40 PM](#)



My response: Please don't ask for more buzz-words...

<http://xpediantsolutions.blogspot.com/2009/01/response-ojt-is-dead-long-live-training.html>

Posted by: Bill Berger | [January 16, 2009 at 08:06 AM](#)



Anne,
Your article title is misleading.

Your niggle appears to be with an acronym that has been abused, rather than the architectural paradigm itself.

You go to some length to show how the concept has been misrepresented by vendors and misunderstood by the business community. You remind us that the tail (technology) continues to wag the dog (business). But you agree that "the requirement for service-oriented architecture is stronger than ever."

Rainer Thiel

Posted by: Rainer Thiel | [January 16, 2009 at 08:32 AM](#)



Finally!! Can we go back to writing out CSV files for other applications now? Why not just scrap the computers too, and go back to paper - that would definitely reduce the IT budget!

Posted by: Steve Van Ausdall | [January 18, 2009 at 08:33 PM](#)



Hi Anne,

Interesting article, but what is a SOA? If you ask 10 technologists, you will get 10 different answers. Worse yet, how do you ever explain it to a business person (i.e. the one with the money) and more importantly, why should they care? If you look at the Wikipedia definition of a SOA, it is some 10 pages long - ridiculous!

Fodder for another blog post: succinctly explain a SOA intended for a business audience in one paragraph so that business folks could understand what it means and why they would care. I don't think it can be done. And that is why the TLA SOA is dead.

Having been involved in many "services" designs and implementations over the years, we never used that TLA because it would have been the end of the project as everyone would

have hung themselves on it trying to figure out what it meant and why. As a software development professional, I am embarrassed that our industry came up with that TLA. I really hope it dies ;-)

Keep up the good writings!

Posted by: [Mitch](#) | [January 19, 2009 at 12:03 AM](#)



SOA is to Information systems what cheap loans and low interest mortgages were to the financial world. Just hypes and lies aimed at misleading ignorants. Time to come back to earth. Pragmatism must replace all the hypes.

Posted by: Bob | [January 24, 2009 at 03:21 PM](#)



Is not better instead of kill the acronym just to fix the causes of the failure ? Otherwise, we are going to change the name every year, and the problem remain unsolved...

Posted by: [Glauco Reis](#) | [January 25, 2009 at 02:33 PM](#)



- better late than never ;) -

I don't think 'services' is the solution.

We should more look into the direction of Model Driven SOA!

See <http://www.theenterprisearchitect.eu/archive/2009/01/26/soa-is-dead-long-live-model-driven-soa>

Posted by: [Johan den Haan](#) | [January 26, 2009 at 12:23 PM](#)



Great analysis.

Service providers and technology vendors tailor definitions of SOA to suit their particular market offerings. Many vendors describe their technology offerings as “SOA” simply because they have implemented a web service API. Similarly the acquisition of an enterprise message bus does not mean that the organisation now has SOA architecture. Web services and enterprise message buses are technologies SOA is a design philosophy, you cannot go out and buy a SOA nor can any vendor sell you one.

SOA is (or should be) a business initiative. The multiplicity of definitions has lead to unrealistic expectations.

Paraphrasing another contributor - lets scrap SOA and get on with SOA

Posted by: [Selwyn Akintola](#) | [January 27, 2009 at 04:47 AM](#)



SOA has fallen, victim of a civil war - evidenced in this very blog - among its own flagbearers. Like all civil wars, there were zealots and warriors who formed multiple opposing armies, each group ever arguing religion: "what's the best service," "bottom-up" versus "top-down," "granularity," "where is the business logic?" "SOAP" vs. "REST." Debate and purity are fine, but not when they get to the point of bringing real-world business initiatives to a standstill or, worse, bankrupt. In the end, no side won, and the SOA flag now has fallen into the dust. In my experience, it was the business people who always understood quite readily the value of SOA for Business Services, and who pleaded with the technology teams to please get on with it and deliver in a timely, cost-effective way. We failed them. We failed our companies and clients. If it's not too late to pick up that SOA flag again, let's hope we can do better the next time around and meet that essential need business was trying to communicate to us, and on-time and within budget.

Posted by: Sowa | [January 27, 2009 at 02:47 PM](#)



oh, please ms manes! stop listening to the salesmen! go ask the technocrats! this argument is too shallow!

Posted by: george O | [January 27, 2009 at 11:50 PM](#)



IMHO it comes down to the question, How do you eat an elephant? The larger the infrastructure/organization (re-)engineering project, the greater the risk of choking. So take small bites. And chew well.

Transitioning to SOA can happen service-by-service, minimizing risk. Lessons learned from service-enabling one process, system or store is then used to accelerate bringing new ones online, identifying orchestration requirements and shaping pragmatic, achievable design goals as the infrastructure evolves.

And significantly, business procedures and processes change step-wise as services become available. This lessens disruption of operations and allows assimilation of changes and fine-tuning of services/processes before continuing. And as mentioned above, re-aligning goals as reality is better understood.

In general, and whatever you call them, changes to the IT firmament should be approached with humility. Infrastructures are products of evolution - survival of the highest switching cost - and are as such a custom fit. Besides, what's in a name? JDBC by any other name is still not the be-all-end-all data access language it was harked as in the last millennium. But it's still a pretty good idea.

-Eddie

Posted by: [Eddie Hartman](#) | [January 29, 2009 at 09:41 AM](#)



1. SOA strategy has got nothing to do with underlying technology and systems.
2. SOA has got nothing to do with webservices either. You can have one without the other.
3. You dont become SOA enterprise by just creating webservices.
4. SOA has to be thought in terms of business process building blocks and not IT systems centric.
5. Not every webservice artifact in an enterprise needs to be part of SOA. You need to add layers of abstraction to a webservice before you can think of making it part of SOA. Business groups/units would be the best folks to identify what they think as a "service" or step in a service.
6. SOA is not a one time implementation thing.
7. SOA needs time, money and resources.
8. If you start thinking in terms of IT systems while creating your SOA strategy, you are not thinking SOA.

Posted by: Praveen Jhurani | [January 29, 2009 at 11:22 PM](#)



<http://apsblog.burtongroup.com/2009/01/soa-is-dead-long-live-services.html#comment-144224970>

I think that Jeff Griffith hit the nail on the head. This is my findings and observations as well.

Basically a lack of understanding of the business aspects to determine what is reusable and what is not is a fundamental problem as SOA attempts to centralize business functionality and reduce integration points - and not knowing when that is right or wrong at a service level is difficult for technical people to communicate and business to conceptualize.

Business people think of the specific problem solution as opposed to IT's view of how the solution might be respresented in the business domain.

Posted by: Allen Guest | [February 03, 2009 at 01:34 PM](#)



I just blogged that I think "Big SOA is Dead; Little SOA is Thriving" at: <http://tinyurl.com/soa-today2> . Ok, maybe Big SOA isn't "dead", but certainly struggling to convince companies to invest in BPM, BAM, ESB (Big SOA) in today's economic climate is a tough, academic sell when they can go Little SOA with positive ROI. Organizations want rapid results- they want SOA Today and not 6-9 months down the line!

Posted by: [jordan braunstein](#) | [February 24, 2009 at 05:54 AM](#)



I want to propose an alternative to an architecture based on co-operating services - "The Integrated Solution (tm)". With "The Integrated Solution (tm)", each component calls the public API of the component it wants communicate with. There is no potential network interaction protocol to define, no overhead of serialization, and minimal latency cost. The cost of connecting two functions with "The Integrated Solution (tm)" is lower than the cost in a service based approach.

This leads to lower IT cost, tighter integration, shorter time to market, higher performance, and lower system complexity.

Successful implementation of a system using "The Integration Solution (tm)" does not require a disruption to the status quo. Integration is simply a matter of deploying proven technologies - modules, components, API's, Software Design, Project Management, good HR practices. There is no required shift in the way IT operates, and no new technologies to learn. There is no need for a low cost, quickly delivered, high performance, application delivered with "The Integrated Solution (tm)" to be part of something bigger to deliver ROI.

"The Integration Solution (tm)" can be a journey, or not, depending on your business needs. "The Integration Solution (tm)" is something you do, not something you buy.

Posted by: ArtSrc | [March 09, 2009 at 03:28 PM](#)



From the trenches:

Ann Thomas is SPOT on!!

Even the reply by the guys at ORACLE to her dissing SOA (i.e. the grandiose vision of SOA) only provide project examples that are integration in nature ONLY.

Unfortunately EVERY single project i have consulted on, where SOA is "supposed" to be the underlying architecture/ethos/whatever is only a means to integration.

The grandiose SOA concept requires the FULL cooperation of an enterprise, extensive planning and then coordination.

Sizeable organization simply do not have the wherewithal to carry out such an exercise.

Management continually look for the I.T. silver bullet, whether it be off-shoring, on-shoring, SOA...it simply does not exist.

Posted by: kambiz shahri | [March 10, 2009 at 12:20 AM](#)



My response is here: <http://architecture-soa-bpm-eai.blogspot.com/2009/03/is-soa-dead-nope.html>

Posted by: [Tushar](#) | [March 29, 2009 at 10:53 PM](#)



This kind of technology is always difficult for companies. Mostly because it involves introducing a lot of change. Change is important, but it also adds risk. Something most companies like to try and avoid.

Posted by: [Japanese words](#) | [April 02, 2009 at 10:26 PM](#)



"SOA is dead" - Nietzsche, "Nietzsche is dead" - SOA "No, SOA is getting killed by a lack of SOA talent." - Linthicum "No, SOA is getting killed by a lack of SOA interest" - SOAFacts.

-- SOAFacts.com

Couldn't resist a SOAFacts reference, which I think represents in humor the nugget of your observation. All the hype, marketecture, and focus on big-dollar platforms has hid the value in services-thinking, and disillusioned many technologists.

Posted by: Peter Pascale | [April 09, 2009 at 11:43 AM](#)



Good article. Thing is that the arcronim SOA turned into a brand for selling products and technologies. I don't have a problem with this but it does making things difficult when examining products with all the spin off terminology and concepts from different vendors. For me SOA is what Object Orientation was to the 90s. A great style of architecture which helps me build better software. When considering the costs, I think the question to be asked is not "do we need SOA" it's "are we selecting the right vendor/produts to meet our solution."

Posted by: Stephen Dougall | [May 15, 2009 at 07:23 AM](#)



Anne - This seems to be an over agressive statement. SOA benefits are being underrated today because of its slow ROI. And if SOA journey begins with the right note it can bring lot of value to an enterprise.

http://www.infosysblogs.com/oracle/2009/05/top_down_or_bottom_up_soa.html

Posted by: Gaurav Sharma | [May 31, 2009 at 11:19 PM](#)



Services are like parts. It should be a simple matter for me to make some nails, hammers, buckets, springs, and bricks, and greatly improve my bottom line. Oh gee, something missing here? What fraction of a whole is comprised of loose parts? Half? A tenth? The discussions above use the words 'enabler' and 'mashup' amongst others. There is a gigantic gap in the thought processes that focus on development of parts, leaving for later the nature of the whole. This particular technology (It is not productive to name it) promotes wrong-headed emphasis on trivial characteristics, and the failures to succeed while using it

(that I am aware of) grew directly out of its misbegotten philosophy. Value comes from a different place.

Posted by: Dave M | [June 03, 2009 at 02:44 PM](#)



SOA is dead? Was it ever alive in the first place? SOA, in my humble opinion is the stuff of pipe dreams. For the vast majority of companies it is not achievable or realistic, rather it being the new plaything of the super-rich...and who is super rich anymore. I'm not saying the concept wasn't good (I even blogged on it myself <http://www.theprocessninja.com/process/2009/04/mr-potato-head-explains-soa-bpm.html>) but it just wasn't practical for the majority of companies...

Posted by: [The Process Ninja](#) | [June 28, 2009 at 08:48 PM](#)

[«_Previous](#) | [Next_»](#)



Excellent post.

My opinion: The major product vendors once again got in the way with overcomplicated SOA stacks and got in with so much focus on technology (product) implementation that budgets were spent before legacy business functions could be migrated to a SOA-compliant architecture.

Posted by: Philip Stander | [September 03, 2009 at 08:57 AM](#)



One of my favorite examples of this type of thinking is a question a professor posed for me in grad school. Is CRM something new or something that we have been doing since the first retailer had a product to sell. Obviously you need the label to make sense of what it is you are trying to accomplish. You must realize that a lot of it relates to using modern tools to address a problem what you are trying to accomplish relates to using a new tool for an old problem.

Posted by: [ITIL V3](#) | [January 05, 2010 at 10:30 PM](#)

[« Previous](#)