معماری سرویس گرا از نگاه مدیران فناوری اطلاعات

امیر مهجوریان

مشاور، مدیر پروژه و مدرس معماری سازمانی، تحول و سرویس‌گرایی: Mahjoorian.ir

مدیر فنی آزمایشگاه معماری سازمانی سرویس‌گرا : soea.sbu.ac.ir

Amir.Mahjoorian@gmail.com

۱ معرفی معماری سرویس گرا

معماری سرویس گرا (SOA[۱]) رهیافتی است برای ساخت سیستم های توزیع شده که کارکردهای نرم افزاری را در قالب سرویس ارائه می کند. این سرویس ها هم توسط دیگر نرم افزارها قابل فراخوانی هستند و هم برای ساخت سرویس های جدید مورد استفاده قرار می گیرند، این رهیافت برای یکپارچه سازی فناوری ها در محیطی که انواع مختلفی از سکوهای[۲] نرم افزاری و سخت افزاری وجود دارد ایده آل است. خواص معماری سرویس گرا به این شرح است:

- استفاده از استانداردهای مستقل از فناوری و مورد توافق برای ارائه مولفه های نرم افزاری تحت قالب سرویس
- هم موضوعی فنی است و هم نوعی سبک تفکر است
- معرفی کننده یک روش مشخص و مورد توافق برای تعریف و ارتباط بین مولفه های نرم افزاری در قالب سرویس
- تقویت کننده رهیافت سرهم بندی اجزاء از قبل تعریف شده برای ساخت نرم افزارها به جای توسعه و پیاده سازی مجدد
- شاخص های کیفیت سرویس[۳](امنیت، کارائی، سبک یکپارچه سازی و ..) به روشنی برای هر سرویس تعریف و اندازه گیری می شود
- سیستم ها می تواند به سرویس های خارج سازمانی نیز مانند انواع داخلی آن متصل شوند

---

[۱] Service Oriented Architecture

[۲] Platform

[۳] Quality of service

این مقاله در سال ۱۳۸۷ منتشر شده است.

معماری سرویس گرا از دیدگاه های مختلف قابل بررسی است، هر فرد یا ذینفع بر طبق جایگاه خود تصویری از معماری سرویس گرا دارد، در ادامه سه دیدگاه مدیران فناوری اطلاعات، مدیران کسب و کار و طراحان و پیاده سازان سیستم مورد بررسی قرار می گیرد.

- مدیران فناوری اطلاعات: سبکی از معماری که حاوی قوانین، الگوها و ضوابطی است که منجر به ایجاد خصایصی نظیر پیمانه ای بودن[۴]، بسته بندی[۵]، اتصال سست[۶]، استفاده مجدد[۷] و ترکیب پذیری[۸] شده و از نظر ساختاری از یک ارائه دهنده سرویس و یک درخواست کننده سرویس تشکیل شده است.
- مدیران کسب و کار: مجموعه ای از سرویس ها که سازمان مایل به ارائه آنها به مشتریان یا شرکاء خود است.
- طراحان و پیاده سازان سیستم های اطلاعاتی: یک سبک(مدل) برنامه نویسی که از استانداردهائی مورد توافق و مستقل از فناوری استفاده می کند و قابلیت تعامل پذیری بین مولفه های نرم افزاری را بدون توجه به سکو و فناوری پیاده سازی آنها پشتیبانی می کند.

۱,۱ تعاریف معماری سرویس گرا

برای معماری سرویس گرا تعاریف متنوع و بعضا مختلفی ارائه شده که هر کدام از نگاهی به تبیین خصوصیات آن پرداخته اند، برای درک بهتر این مفهوم و آگاهی از کلیه برداشت ها و نگاه های موجود، در ادامه تعدادی از این تعاریف آورده شده است:

- چارچوبی وسیع و استاندارد که سرویس ها در آن ساخته، استقرار و مدیریت می شوند و هدفش افزایش چابکی فناوری اطلاعات در جهت واکنش سریع به تغییرات در نیازهای کسب و کار است.
- سبکی از معماری سیستم های اطلاعاتی که هدف آن دستیابی به اتصال سست در ارتباطات بین مولفه های نرم افزاری است. اتصال سست بین مولفه های نرم افزاری باعث قابلیت استفاده مجدد آنها می شود، سرویس در اینجا به معنای پیاده سازی نرم افزاری یک کارکرد کسب و کار خوش تعریف است که می تواند در فرآیندها یا نرم افزارهای مختلف مورد استفاده و فراخوانی قرار بگیرد.
- معماری سرویس گرا یک محصول نیست بلکه پلی است بین حرفه و فناوری به کمک مجموعه ای از سرویس های متکی بر فناوری که دارای قوانین، استانداردها و اصول طراحی مشخص هستند.

---

[4] Modularity
[5] Encapsulation
[۶] Loosely Coupled
[۷] Reusability
[۸] Composability

این مقاله در سال ۱۳۸۷ منتشر شده است.

۱,۲ ویژگی های معماری سرویس گرا

در جدول ۱ مقایسه ای بین رهیافت سرویس گرا با رهیافت های گذشته انجام شده است:

جدول ۱: مقایسه میان معماری سرویس گرا با سایر گزینه ها

معماری سرویس گرا	رهیافتهای گذشته
ارتباطات ارزش آفرین است	ارتباطات هزینه بر بود
مبتنی بر فرآیند	مبتنی بر کارکرد
ساخت برای تغییر	ساخت برای بقا
توسعه تدریجی	تولید یکمرتبه
فدراسیون نرم افزارها	نرم افزارهای تعامل ناپذیر
مستقل از سکو	تک سکو
اتصال سست	اتصال محکم
پیام محور	مولفه محور

در جمع بندی مطالبی که در خصوص تعریف معماری سرویس گرا و ویژگی های آن ذکر شد، می توان خصوصیات معماری سرویس گرا را بصورت خلاصه در شکل ۱ بیان نمود.

شکل ۱: مبانی و خصوصیات معماری سرویس گرا (SOA)

این مقاله در سال ۱۳۸۷ منتشر شده است.

۱,۳ سرویس های وب

معمولا واژه های معماری سرویس گرا و سرویس های وب[۹] اشتباها به جای هم و به صورت معادل استفاده می شوند لذا لازم است این دو مفهوم به صورت دقیق تر بررسی شوند. سرویس های وب را باید نوعی فناوری برای تحقق ویژگی استقلال از سکو در معماری سرویس گرا دانست.

تعریف: یک سرویس وب، نوعی سیستم نرم افزاری است که جهت تعامل ماشین با ماشین در سطح شبکه طراحی شده است و دارای یک تعریف(توصیف) قابل پردازش توسط ماشین با نام WSDL[۱۰] است. دیگر سیستم ها بر طبق این توصیف از قبل مهیا شده با سرویس دهنده تعامل خواهند داشت، پیام ها توسط پروتکل SOAP[۱۱] (ترکیب HTTP با XML[۱۲]) و یا سایر پروتکل های مشابه منتقل می شوند.

سرویس وب باید دارای شرایط زیر باشد:

- در سطح وب در دسترس باشد
- از استاندارد XML جهت تبادل اطلاعات استفاده کند
- به هیچ سکو یا سیستم عاملی وابسته نباشد.
- با سرورها و سیستم ها تعامل دارد و نه کاربران
- خود توصیف باشد
- قابل شناسائی باشد(جهت استفاده سرویس گیرندگان ابتدا باید ثبت و سپس شناسائی شود)

۲ هم نواسازی و هم خوانی در معماری سرویس گرا

دو واژه پر کاربرد در حوزه معماری سرویس گرا، هم نواسازی[۱۳] و هم خوانی[۱۴] نام دارند. هم نواسازی در خصوص ترتیب اجرای سرویس ها در فرآیند بحث می کند، هم نواساز[۱۵] که رهبری گروه را به عهده دارد مجموعه ای از سرویس ها را فراخوانی می کند تا نتیجه مورد نظر حاصل شود و فرآیند تکمیل گردد، ممکن است سرویس های خارج سازمان نیز در این راستا فراخوانی و استفاده شوند، این کار با کمک موتور فرآیند محقق می شود. در عوض هم خوانی به فرآیندهایی گویند که بدون موتور فرآیندی(رهبر ارکستر) اقدام به تبادل پیام کرده و ترتیب و توالی پیام های مبادلاتی را خود بازیگران ثبت و کنترل می کنند(شکل ۲).

---

[9] Web Service
[۱۰] Web service Definition Language
[۱۱] Simple Object Access Protocol
[۱۲] Extensible Markup Language
[13] Orchestration
[14] Choreography
[15] Conductor

این مقاله در سال ۱۳۸۷ منتشر شده است.

در هم نواسازی، یک کنترل کننده مرکزی، جریان گردش کارها را بین چندین عامل(سرویس،کارگر، سیستم و..) تقسیم می کند. یکی از کاربردهای این مفهموم در شکستن فرآیندهای بزرگ به اجزاء کوچکتر است بطوریکه این اجزاء تحت نظارت هم نواساز اصلی عمل نموده و نتیجه آنها برای همان هم نواساز ارسال شود. این کار پیچیدگی کار را کاهش می دهد، بدین ترتیب منطق جریان کار بصورت جداگانه نگهداری می شود (با نماد BPEL[۱۶]) و بسط و تغییر آن ساده تر می شود. اجزاء(سرویس) نباید دانشی از منطق جریان کار اصلی داشته باشند، آنها فقط به درخواست های هم نواساز پاسخ می دهند و هر جزء یک واحد خود شمول و مستقل به حساب می آید.

شکل ۲: هم نواسازی و هم خوانی

هم نواسازی به سه طریق باعث چابکی فناوری اطلاعات می شود:

- منطق گردش کار که بصورت جدا تو سط هم نوا ساز نگهداری می شود، ساده تر قابل بسط و اصلاح است و تغییر آن تاثیری بر پیاده سازی سرویس های فراخوانده شده نخواهد داشت.
- انجام هم نواسازی باعث می شود منطق گردش کار و حالت مربوط به هر جزء از آن خارج شده و لذا این اجزاء شانس بیشتری برای استفاده مجدد در دیگر هم نواسازی ها داشته باشند.
- استفاده از هم نواسازی در وسعت گسترده و فراگیر برای سازمان هایی که نوع کسب و کار آنها بصورت فدرالی (سازمان هایی که خود متولی ارائه خدمت به مشتریان نبوده، بلکه وظایف را بین مجموعه ای از شرکت ها یا پیمانکاران تقسیم می کنند) همراستایی و انعطاف پذیری بالایی برای فناوری اطلاعات به ارمغان می آورد.

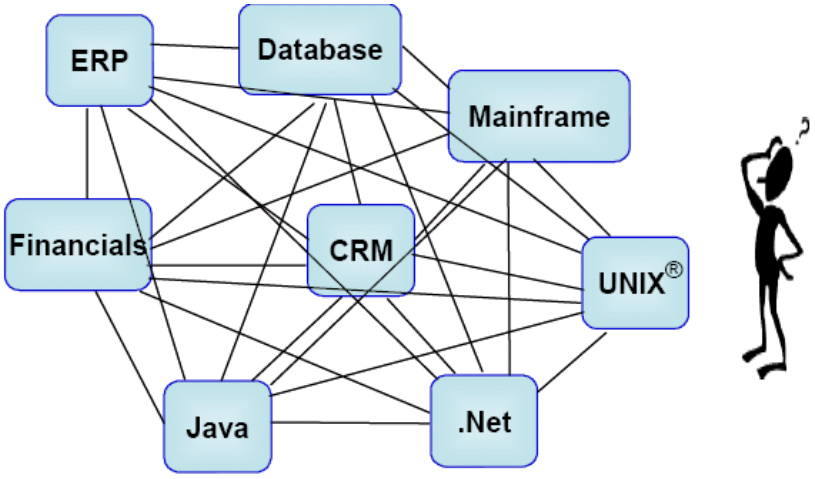
[۱۶] Business Process Execution Language

این مقاله در سال ۱۳۸۷ منتشر شده است.

۳ کاربرد های معماری سرویس گرا

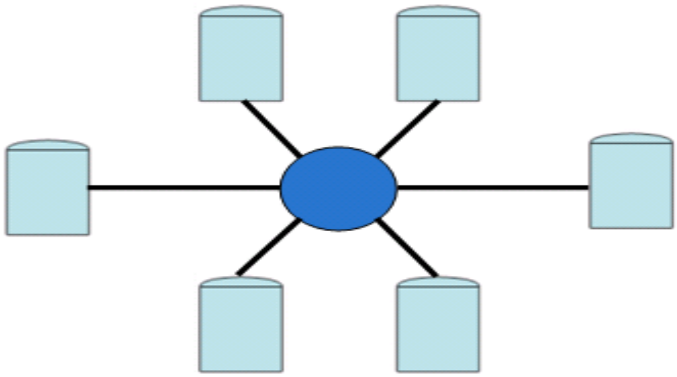
۳,۱ یکپارچه سازی سیستم های اطلاعاتی

راه حل معماری سرویس گرا برای یکپارچه سازی سیستم های اطلاعاتی، ارتباط بین سیستم های اطلاعاتی به کمک وب سرویس است. از اواخر دهه ۹۰ برای چالش تعامل پذیری سیستم های اطلاعاتی رهیافت هائی ارائه شد که معروفترین آنها اتصال نقطه به نقطه[۱۷] و یکپارچگی مبتنی بر یک مترجم مرکزی بود. در حالت نقطه به نقطه(شکل ۳) برای هر تعامل بین دو سیستم اطلاعاتی می بایست که استاندارد و مسیر ارتباطی خاصی فراهم گردد. طبیعی است که چنین رهیافتی بسیار هزینه بر و دست و پا گیر خواهد بود.



شکل ۳ : رهیافت اتصال نقطه به نقطه برای ارتباط بین سیستم های اطلاعاتی سازمان

در حالت مترجم مرکزی نیز میان افزاری[۱۸] به عنوان مترجم بین همه سیستم های اطلاعاتی عمل می کرد به گونه ای که مانند یک هاب مرکزی تمامی پیام های ارسالی به این واسط ارجاع می شد و پس از ترجمه به پروتکل و فناوری مربوط به سیستم دوم، ارسال می گشت(شکل ۴).



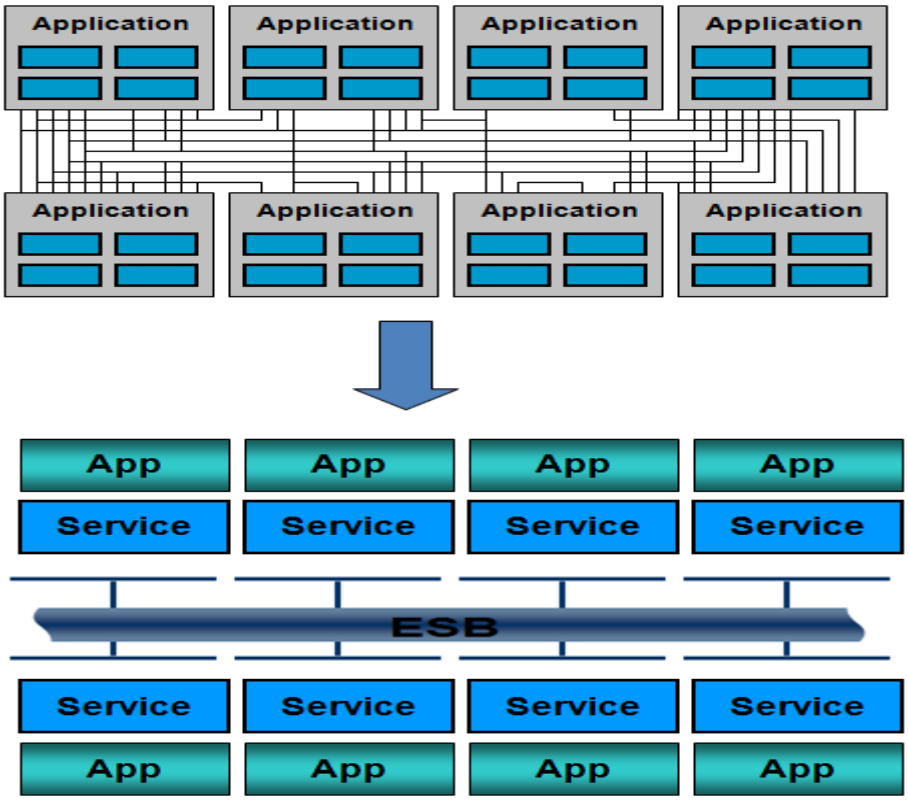
[۱۷] Peer-to-Peer

[۱۸] Middle-Ware

این مقاله در سال ۱۳۸۷ منتشر شده است.

شکل ۴ : رهیافت مترجم مرکزی برای ارتباط بین سیستم های اطلاعاتی سازمان

این گزینه نیز با دشواری هائی همراه بود که مهمترین آنها وجود انواع پروتکل های ناهمجور و عدم جامعیت بود. اما در معماری سرویس گرا اصل بر این است که همه سیستم های اطلاعاتی با یک واسط استاندارد و مورد توافق جهانی تعامل داشته باشند. این واسط سرویس نام دارد(شکل ۵) و پروتکل های مورد استفاده آن نیز شامل SOAP،WSDL،XML می شود، تبادل سرویس بین سیستم های اطلاعاتی به کمک گذرگاه سرویس سازمان(ESB[۱۹]) انجام می شود.



شکل ۵ : تاثیر استفاده از معماری سرویس گرا در یکپارچه سازی سیستم های اطلاعاتی

۳,۲ یکپارچگی اتوماسیون فرآیندهای سازمان در قالب ارکستریشن

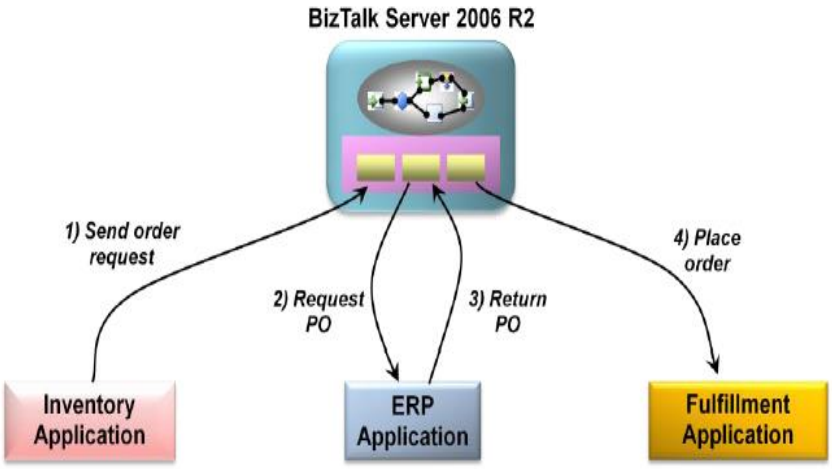
معماری سرویس گرا برای مدیریت و اجرای فرآیندهای سازمان از مفهوم ارکستریشن(هم نواسازی) کمک گرفته است، در این رهیافت منطق و جریان کار فرآیند از فعالیت های آن جدا می شود، به گونه ای که جریان گردش فرآیند در قالب BPEL مدیریت می شود ولی هر کدام از فعالیت های فرآیند می توانند توسط سیستم های اطلاعاتی مختلف پیاده سازی شوند. بدین ترتیب امکان تغییر جریان کار بدون نیاز به تغییر سیستم های

[۱۹] Enterprise Service Bus

این مقاله در سال ۱۳۸۷ منتشر شده است.

پشتیبان میسر می شود که این امر کمک شایانی به انعطاف پذیری فناوری در پاسخ به تغییرات حرفه می کند.

در شکل ۶ چگونگی پیاده سازی یک فرآیند که با سه سیستم اطلاعاتی ارتباط دارد توسط ابزار BizTalk Server نشان داده شده است. در این مثال هر سیستم دارای فناوری و استانداردهای مخصوص به خود است. نکته جالب اینکه سیستم های اطلاعاتی به کار گرفته شده برای اتوماسیون فرآیند مزبور از وجود فرآیند(گردش و قوانین) اطلاع ندارند، آنها تنها به پیام هائی که ارسال شده جواب می دهند(جداسازی منطق فرآیند از اتوماسیون فعالیت های آن)، این موضوع باعث سهولت پیاده سازی فرآیندهای جدید و تغییر فرآیندهای موجود می شود که ارزش بالایی برای سازمان ها دارد.



شکل ۶ : پیاده سازی فرایندهای حرفه به کمک ارکستریشن سرویس ها

۳,۳ تعامل پذیری بین سازمانی

اگرچه ارتباط و یکپارچگی سیستم های اطلاعاتی سازمان ضروری است اما از آن مهمتر(سخت تر) تعامل پذیری بین سازمانی است، چراکه میزان تنوع فناوری ها و پروتکل ها در بین چند سازمان به مراتب بیشتر از میزان آن در بین سیستم های داخل سازمانی است. در شرایط اقتصادی و تجاری جدید، سازمانها نیاز دارند که بصورت موثر از اطلاعات یکدیگر استفاده کنند، از طرف دیگر "فرآیندهای بین سازمانی" در حال گسترش هستند. تجارت که در گذشته سازمانی و حوزه ای بود اکنون به سمت جهانی شدن و اکوسیستمی پیش می رود و تعیین مرز و اندازه جغرافیائی برای سازمانها دشوار شده است. در چنین شرایطی نیاز به تبادل اطلاعات بین سازمانها به شدت احساس می شود.

راه حل معماری سرویس گرا برای این حوزه استفاده از وب سرویس های استاندارد است که در شبکه اینترنت قابل شناسائی و فراخوانی هستند، برای استفاده از این وب سرویس ها قبل از هر چیز آنها باید توسط

این مقاله در سال ۱۳۸۷ منتشر شده است.

متقاضیان شناسائی شوند. بدین منظور دایرکتوری ثبت و شناسائی سرویس ها (UDDI[۲۰]) ایجاد شده است. ارائه دهندگان سرویس مشخصات سرویسهای خود را در این دایرکتوری ها ثبت می کنند و متقاضیان نیز با جستجوی سرویس مورد نظرشان (مانند موتورهای جستجوری صفحات وب) می توانند از این سرویس ها در سازمان خود بهره ببرند. نتیجه این امر امکان تعامل موثر با دیگر سازمانها (شرکاء، رقبا، مشتریان) است که توسط سرویس های متنوع فراهم شده است(شکل ۷).

شکل ۷ : تعامل پذیری بین سازمانی به کمک وب سرویس ها

۴ مزایا و نتایج معماری سرویس گرا

- سیستم های چابک: معماری سرویس گرا سازمان را قادر می سازد تا به سرعت سیستم های خود را تغییر دهد. این چابکی هم از جهت کارکردهای سیستم و هم از جهت تغییر جغرافیائی یا ارتقاء سکوها و حتی تغییر تامین کننده فناوری است.
- استفاده مجدد: استفاده مجدد از کد برنامه یا سیستم ها، از گذشته مورد توجه روش های تولید و توسعه نرم افزار بوده است، معماری سرویس گرا قابلیت استفاده مجدد را هم در سطح کارکردی(سرویس) و هم در سطح داده ها مهیا می کند.

[۲۰] Universal Description Discovery and Integration

این مقاله در سال ۱۳۸۷ منتشر شده است.

- یکپارچگی آسان با شرکاء داخلی و خارجی: می توان گفت قابلیت یکپارچگی سیستم ها و سکوها یکی از نکات کلیدی است که معماری سرویس گرا به آن پرداخته است.
- بهبود بازگشت سرمایه: معماری سرویس گرا مجموع هزینه صرف شده برای فناوری اطلاعات و سرویس های کسب و کار را به دو روش کاهش می دهد. اول با حذف هزینه های میان افزارها و فناوری های اختصاصی و جایگزین کردن آن با فناوری های استاندارد مانند وب سرویس و دوم با ترکیب کارکردهای کسب و کار در قالب سرویس هائی که توسط واحدهای مختلف قابل استفاده باشد.
- همراستایی فناوری اطلاعات با کسب و کار: هدف معماری سرویس گرا پیاده سازی نرم افزاری یک کارکرد کسب و کار (سرویس کسب و کار) با قابلیت استفاده مجدد و انعطاف پذیر است، لذا سرویس های نرم افزاری در معماری سرویس گرا چیزی نیستند مگر عینیت بخش همان سرویس های کسب و کار در بستر فناوری اطلاعات.
- قابلیت انعطاف و تغییر آسان از یک ارائه دهنده سرویس به دیگری: موضوع انعطاف در معماری سرویس گرا برای هر دو مورد سرویس های داخل سازمانی و خارجی صدق می کند.

۵ مراجع و منابع

[1] Erl, T., SOA: Principles of Service Design, Prentice Hall, 2008.
[2] Service Oriented Architecture (SOA) in the Real World, Microsoft Publication, 2007.
[3] Erl, T., Service-Oriented Architecture: Concepts, Technology, and Design, Prentice Hall, 2007.
[4] Khoshafian, S., Service Oriented Enterprises, Auerbach, 2007.
[5] Pulier, E. and Taylor, H. 2006, Enterprise SOA, Manning, 2006.
[6] Chatterjee, Sandeep and Webber, Developing Enterprise Web Services: An Architect's Guide, Upper Saddle River, Prentice Hall, 2004.
[7] Knipple, R., Service Oriented Enterprise Architecture, MS Thesis, IT-University of Copenhag, 2005.
[8] Manes, A.T., Web Services: Manager's Guide, Addison-Wesley, 2003.
[9] Krafzig, D., Banke, K. and Slama, D., Enterprise SOA: Service-Oriented Architecture Best Practices, Prentice Hall, 2004.
[10] Lawler, J. and Howell-Barber, H., Service-Oriented Architecture: SOA Strategy, Methodology, and Technology, Auerbach, 2007.
[11] Sims, O., Services Oriented Architecture: Part 3 - Federation, CBDI Forum, 2003.

این مقاله در سال ۱۳۸۷ منتشر شده است.