تاثیر معماری سرویس‌گرا بر الگوریتم زمان‌بندی در محاسبات ابری

اعظم السادات نوربخش

هیات علمی گروه کامپیوتر دانشگاه آزاد اسلامی واحد لاهیجان

Nourbakhsh@liau.ac.ir

شاهد کاظمی شریعت پناهی

دانشجو کارشناسی ارشد مهندسی نرم افزار کامپیوتر دانشگاه آزاد واحد علوم تحقیقات گیلان

kazemi.shariatpanahi@gmail.com

ساسان برهلیا

دانشجو کارشناسی ارشد مهندسی نرم افزار کامپیوتر دانشگاه گیلان

Berehlia@msc.guilan.ac.ir

مسئول مکاتبات: شاهد کاظمی شریعت پناهی

چکیده

محاسبات ابری، طی سال‌های اخیر رو به محبوبیت گذارده و شرکتهای بزرگ فناوری اطلاعات سراسر دنیا شروع به طراحی زیرساخت آن نموده‌اند. در دنیای فناوری امروزی، برخی حوزه‌ها همانند SOA در محاسبات ابری نیز مطرح گردیده اند. مقاله حاضر، پژوهشی در مورد الگوریتم‌های زمان‌بندی محاسبات ابری با استفاده از SOA، مقوله‌هایی در زمینه پیاده‌سازی الگوریتمی جدید و ایده آل را مورد بررسی قرار داده و پیشنهاد می‌کند که این الگوریتم-ها با توجه به مقوله SOA با سایر گونه‌های دیگر آن به‌کار گرفته شود. افزون بر این، SOCCA طراحی سطح بالایی را به‌منظور حمایت بهتر از ویژگی‌های چندگانه عاریه‌ای از محاسبات ابری پیشنهاد می‌دهد. با این وجود، تفسیرهای گوناگونی از آن چه که محاسبات ابری می‌نامند، وجود دارد. حتی اگر برخی از ویژگی‌های اساسی محاسبات ابری از طریق تلاش‌های علمی و پژوهشی به مرحله تحقق رسانده شوند، این زمینه هنوز در ابتدای راه پیشرفت خود قرار دارد. همچنین همان گونه که گفته شد با در نظر گرفتن همپوشانی در میان SOA و محاسبات ابری در SOCCA می‌توان به آنتولوژی‌ها و پارامترهایی دست یافت که در آینده سبب بهبود در ساختار الگوریتم زمان‌بندی در محاسبات ابری شود و نو آوری جدیدی را پیش روی آورد. این مقاله به الگوریتم زمان‌بندی محاسبات ابری با توجه به‌تاثیر مقوله SOA پرداخته و روشی ایده‌آل را برای بهبود الگوریتم زمان‌بندی در محاسبات ابری بیان می‌کند.

واژگان کلیدی: معماری سرویس‌گرا، محاسبات ابری، الگوریتم زمان‌بندی، معماری لایه‌ای SOCCA

۱. مقدمه

می‌توان محاسبات ابری را به معنی به کارگیری قابلیت‌های کامپیوتری بر مبنای اینترنت درنظر گرفت. مزیت‌های محاسبات ابری در مقایسه با محاسبات سنتی شامل چابکی، کاهش هزینه، مستقل بودن از تجهیزات، عدم وابستگی به مکان و میزان پذیری است. معماری سرویس‌گرا، روشی برای طراحی و پیاده‌سازی نرم‌افزارهای گسترده سازمانی است تا بتوانند به وسیله ارتباط بین سرویس‌هایی که دارای خواص اتصال سست، دانه درشتی و قابل استفاده مجدد هستند، به هدف خود برساند. از این معماری، معمولاً برای ساخت سیستم‌های توزیع شده که کارکردهای نرم‌افزاری را در قالب سرویس ارائه می‌دهند، و نیز برای ساخت سرویس‌های جدید بهره جویی می‌شود. در این معماری، تلاش بر یکپارچه‌سازی فناوری‌ها در محیطی که انواع مختلف از سکوهای نرم-افزاری و سخت‌افزاری وجود دارد، مدنظر است. با توجه به بررسی‌های گذشته در این حیطه، می‌توان محاسبات ابری و SOA را به طور مستقل و همزمان مورد بررسی قرار داد، به‌گونه‌ای که محاسبات ابری و سرویس پیشنهادی بتوانند زیر بنای ارزشمندی برای دستاوردهای SOA را فراهم کنند. در ادامه این مقاله، به مراحلی برای بهبود پیشرفت محاسبات ابری با کمک معماری سرویس‌گرا و دیگر موارد آن اشاره خواهد شد. با توجه به بحث‌های گفته شده، در این مقاله سعی شده است که نخست به معرفی و بررسی و سپس به ویژگی‌ها و لایه‌ها و پروتکل‌های استاندارد، که برای ترکیب با این دو مقوله سروکار دارند، پرداخته شود. در ادامه زمان بندی منابع ابرها در تئوری مصرف کننده و تهیه کننده مورد بررسی قرار گرفته اند. در این بخش نیازهای مشتری، منابع

Archive of SID

www.SID.ir

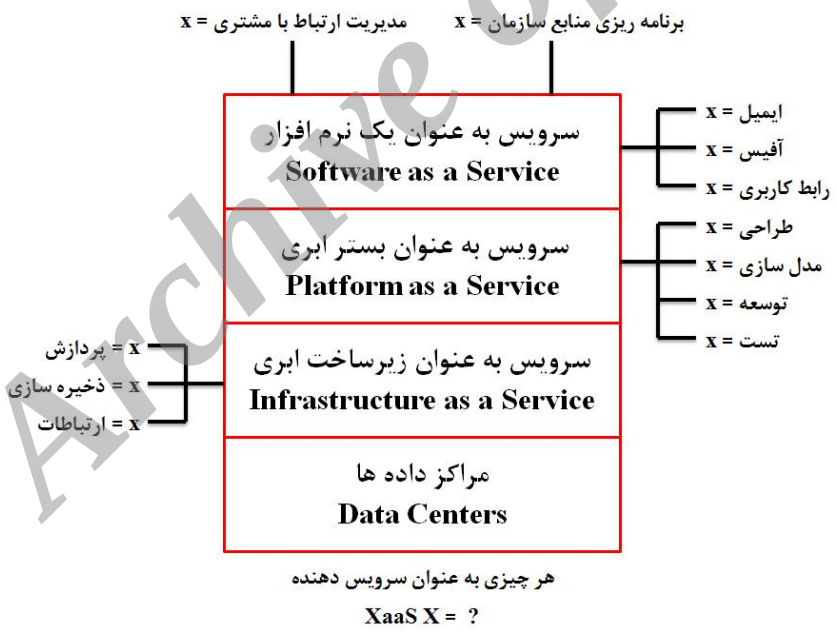
فروشنده‌ها، قیمت مورد توجه قرار گرفته و سپس هر کدام با توجه به مقوله SOA مدل شده اند. در بخشی دیگر الگوریتم مصرف انرژی که محاسبات ابری بیشتر به قسمت لایه واسطه‌ای در SOA مربوط می‌شود تحلیل شده و پس از این موارد، در قسمتی دیگر الگوریتمی جدید که سبب ترکیب SOCCA شده و با توجه به مقوله سازمانی SOA را با خود به همراه دارد تشریح شده و در نهایت روشی ایده‌ال را بیان می‌شود که با توجه به SOA، به بهبود الگوریتم زمان-بندی در حیطه محاسبات ابری کمک شایانی می‌کند.

۲. مروری بر محاسبات ابری و معماری سرویس‌گرا

معماری سرویس‌گرا و محاسبات ابری به یکدیگر مرتبطاند، SOA الگوی معماری است که راه حل‌های کسب و کاری برای ایجاد، سازماندهی و مصرف دوباره‌ی اجزای محاسباتی را هدایت می‌کند. به عبارت دیگر SOA و محاسبات ابری با هم همزیستی دارند و مکمل یکدیگرند و هم دیگر را پشتیبانی می-کنند (Tsai, Sun, & Balasooriya, 2010)، (Barry & Dick, 2013)، (شمس و مهجوریان، ۱۳۸۹).

۱.۲. معرفی محاسبات ابری

محاسبات ابری مدلی است برای ایجاد دسترسی همگانی و به صورت آسان و بنا به سفارش شبکه. ابرها مجموعه‌ای از منابع محاسباتی پیکربندی پذیر مانند شبکه‌ها، سرورها، فضای ذخیره‌سازی، برنامه‌های کاربردی و سرویس‌ها اند که بتوانند با کم‌ترین نیاز به تلاش‌های مدیریتی یا تعامل با ارائه دهنده سرویس، به سرعت فراهم شده یا آزاد و رها شوند (National Institute of Standards and Technology, 2011). کاربرد محاسبات ابری برای کاربران و سایر سازمان‌ها سعی بر این داشته است که به جای نصب نرم‌افزار روی چند سیستم، تنها یک نرم‌افزار بر روی یک سرویس دهنده یک بار اجرا و بارگذاری کند که بقیه بتوانند به آن دسترسی داشته باشند. برخی از ویژگی‌ها مانند سرویس مبتنی بر تقاضا [1]، دسترسی وسیع شبکه [2]، ائتلاف منابع [3] و انعطاف پذیری سریع [4] سبب افزایش دسترسی به مقوله SOA در ترکیب با محاسبات ابری شده است (Dillon, Wu, & Chang, 2010)، (Rittinghouse & Ransome, 2010). محاسبات ابری دارای لایه‌های سرویس مختلفی می‌باشد که در شکل ۱ مشاهده می‌شود. هر لایه زیر بنایی برای لایه بالایی خود است.



شکل ۱- سلسله مراتب محاسبات ابری (Tsai et al., 2010)

[1] On demand self service
[2] Broad network access
[3] Resource pooling
[4] Rapid elasticity

Archive of SID
www.SID.ir

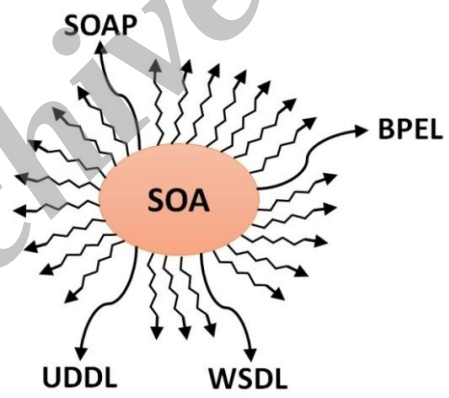
سرویس نرم‌افزار ابری سرویس‌های برنامه‌های کاربردی ابری یا نرم‌افزار به عنوان نرم‌افزار به صورت سرویس SaaS روی اینترنت قرار می‌گیرند. لایه سرویس به عنوان بستر ابری، به مصرف کننده این اجازه را می‌دهد تا بر روی محیطی که برنامه بر روی آن است تنظیماتی انجام داده و با کنترل و نظارت داشته باشد. در لایه سرویس زیرساخت ابری، مصرف کننده می‌تواند از منابع محاسبات پایه همانند قدرت پردازش، فضای ذخیره سازی و اجزای شبکه، میان-افزار استفاده کند. زیر بنای پردازش ابری، مراکز داده است. مراکز داده پایه‌ای برای ذخیره ابرها است (Tsai et al., 2010)، (Rittinghouse & Ransome, 2010)، (Barry & Dick, 2013)، (Magoulès, Pan, & Teng, 2012).

۲.۲. معماری سرویس‌گرا (SOA)[5]

معماری سرویس‌گرا بخشی نوظهور از معماری نرم افزار است. این بخش شامل روش‌های مختلف نامطمئن بوده و تاکید بیشتر بر آزمون و خطا ضروری است. SOA یک پدیده جدید بوده و برای پیاده سازی روش‌های آن نیاز به شناخت و تشریح بیشتر می‌باشد. خدمات از طریق ESB کشف و ارائه شده است. بنابراین ESP[6] به عنوان یک کانال ارتباطی بین خدمات عمل می‌کند. این توابع عواملی از قبیل مسیریابی، امنیت، قابلیت اطمینان، خدمات مدیریت و ورود به سیستم را فراهم می‌کند. خدمات به دو بخش خدمات عمومی و خدمات فرآیند طبقه بندی می‌شوند (Waris, Khan, & Fakhar, 2013). معماری سرویس-گرا دارای چهارچوبی استاندارد است، همچنین صفات و خصوصیاتی را در ذات خود دارد که سبب ایجاد قابلیت‌هایی از قبیل ساخته شدن، استقرار و یا مدیریت دیگر سرویس‌هاست. برای نمونه عدم امکان تعامل پذیری و یکپارچگی تمامی سیستم‌های اطلاعاتی سازمانی، در جهت استفاده از مؤلفه‌های نرم-افزاری مشترک، یکی از دلایل منطقی در معماری سرویس‌گرا است و همچنین می‌توان تفاوت میان دیدگاه‌های کارشناسان IT با کارکنان کسب و کار را از جمله آن موارد ذکر کرد (شمس و مهجوریان، ۱۳۸۹)، (Williams, 2014). شکل ۲ نمونه‌ای از پروتکل‌ها و استانداردها را در این زمینه نشان می‌دهد.

مهم‌ترین اهداف ارئه معماری سرویس‌گرا شامل موارد زیر اند: (شمس و مهجوریان، ۱۳۸۹)، (Williams, 2014):

- ایده‌آل کردن قابلیت استفاده مجدد و انعطاف پذیری
- انعطاف‌پذیری فناوری اطلاعات در پاسخ به تغییرات مداوم کسب و کار
- ایجاد استاندارد سازی و یکپارچگی سکوها و زیرساخت‌های فناوری اطلاعات
- هم‌راستایی کسب و کار با IT
- ایجاد تعامل پذیری بهتر در بین سازمانی



شکل ۲- پروتکل‌ها و استانداردها در SOA

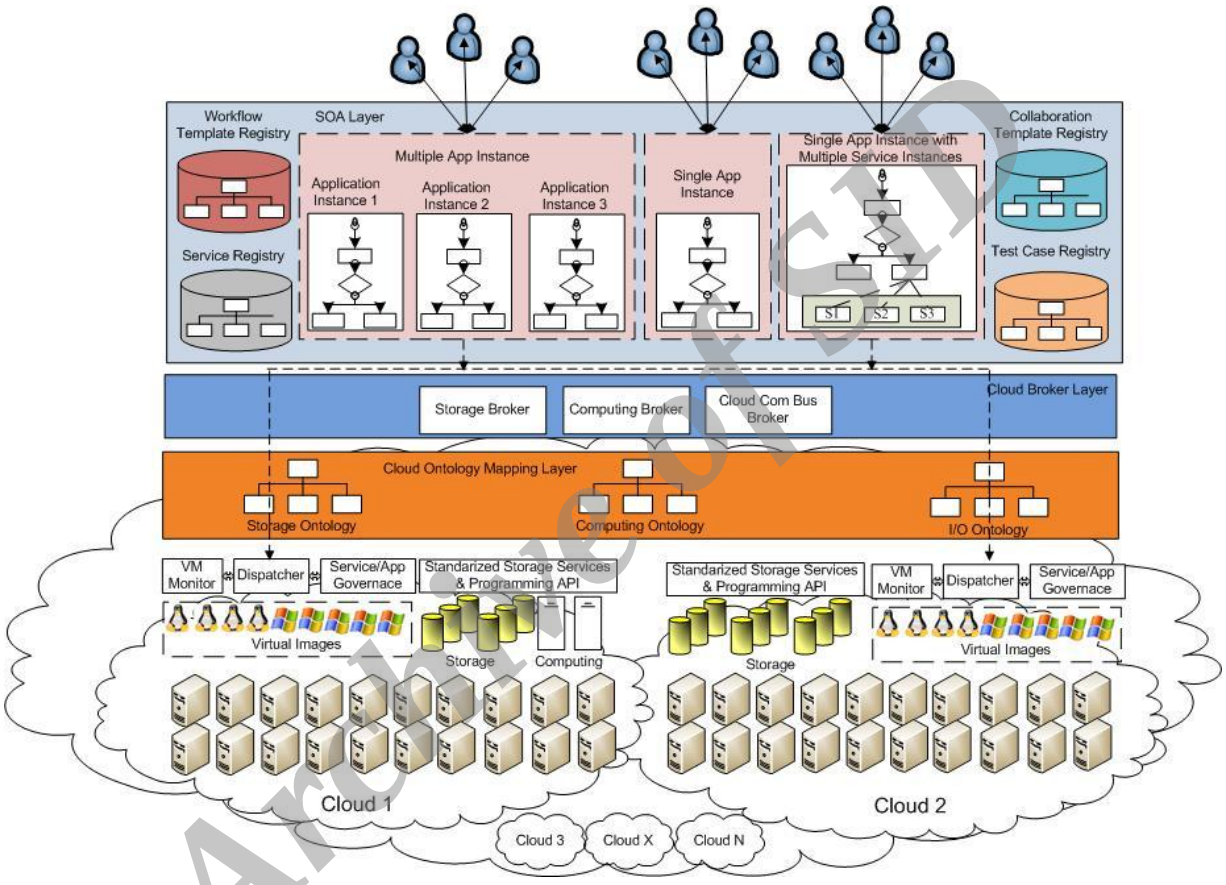
پروتکل‌ها و استانداردها در SOA بسیار منحصر به فرد اند. SOAP ساختاری برای تبادل پیام‌ها در قالب XML دارا است. WSDL زبانی مبتنی بر XML است که هدف آن توصیف ویژگی عملیاتی سرویس‌های وب می‌باشد. UDDL واسطی برای انتشار و شناسایی سرویس‌های وب است و شامل یک مخزن می‌شود که ارائه دهندگان به انتشار و تبلیغ سرویس خود می‌پردازند تا دیگران بتوانند آن را شناسایی کنند. BPEL زبانی برای مشخص کردن فرآیند-های اجرایی کسب و کار است (شمس و مهجوریان، ۱۳۸۹)، (Williams, 2014)، (Barry & Dick, 2013).

[5] Service oriented architecture
[6] Enterprise Service Bus

Archive of SID
www.SID.ir

۳. رهیافتی برای ترکیب میان محاسبات ابری با SOA

پردازش ابری و SOA دارای نقاطی مهم و مشترک هستند. بیشترین اهمیت، در نزدیکی بالای پشته مربوط به پردازش ابری دیده می‌شود که اجزای کاربردی قابل دسترسی شبکه و سرویس‌های نرم‌افزاری هستند و مانند سرویس‌های همزمان شبکه پردازش ابری و معماری سرویس‌گرا، مفاهیم کلی سرویس‌گرا را به اشتراک می‌گذارند. SOA از منظر حیطه، عملکرد، اجرا و پیاده‌سازی سرویس‌های نرم‌افزاری، حائز اهمیت است. پردازش ابری بر روی جنبه-هایی از پشته پردازش IT تمرکز می‌کند و می‌تواند آن را به محصول مصرفی تبدیل نمایند، که دریافت آن، به طور گام به گام از ایجادگرهای ابری ممکن باشد. به همین نحو، یک سکو برای اجرای کاربردهای شبکه گرا می‌تواند از مراکز مازاد بر احتیاج داده در ابر به اجاره گرفته شود. در شکل ۳ وابستگی این دو مقوله را می‌توان به وضوح مشاهده کرد.



شکل ۳- معماری سرویس‌گرای محاسبات ابری (Tsai et al., 2010)

۱.۳. معماری لایه‌ای SOCCA

۱.۱.۳. لایه ایجاد گر ابری تکی

این لایه شبیه محاسبات ابری جاری است. هر ایجادگر ابری، مراکز اطلاعاتی خودش را می‌سازد و قدرت این را دارد که ابر به مراکز اطلاعاتی، خدمات (سرویس) ارائه کند. هر ابر ممکن است تکنولوژی‌های مجازی تحت تملک خودش را داشته باشد ویا تکنولوژی‌های مجازیOpen source مانند Eucalyptus[۷] را بهینه سازی نماید. در این لایه ایجادگرهای ابری منابع و مجازی سازی را مراقبت می‌کنند (Tsai et al., 2010).

[7] Eucalyptus System. http://www.eucalyptus.com

Archive of SID
www.SID.ir

۲.۱.۳. آنتولوژی[8] در مسیر دهی لایه‌ای ابری

ایجادگرهای ابری ممکن است با استانداردهای ذکرشده دقیقاً مطابقت نداشته باشد: همچنین این امکان وجود دارد که آنها ویژگی‌های اضافی را که شامل این استانداردها هستند، پیاده‌سازی نکنند. سیستم‌های مهم هستی شناسی که مورد نیازند به شرح زیر هستند (Tsai et al., 2010):

۱- آنتولوژی ذخیره‌ای: به مفاهیم و واژه‌هایی اطلاق می‌شود که با دستکاری داده‌ها در ابرها مرتبط اند همانند داده‌های به روز، افزودن داده‌ها، حذف داده‌ها، انتخاب داده‌ها و غیره.

۲- آنتولوژی محاسباتی: به مفاهیم و واژه‌های اطلاق می‌شود که با محاسبات توزیع شده درون ابرها مرتبط اند، همانند چارچوب‌های Map/Reduce.

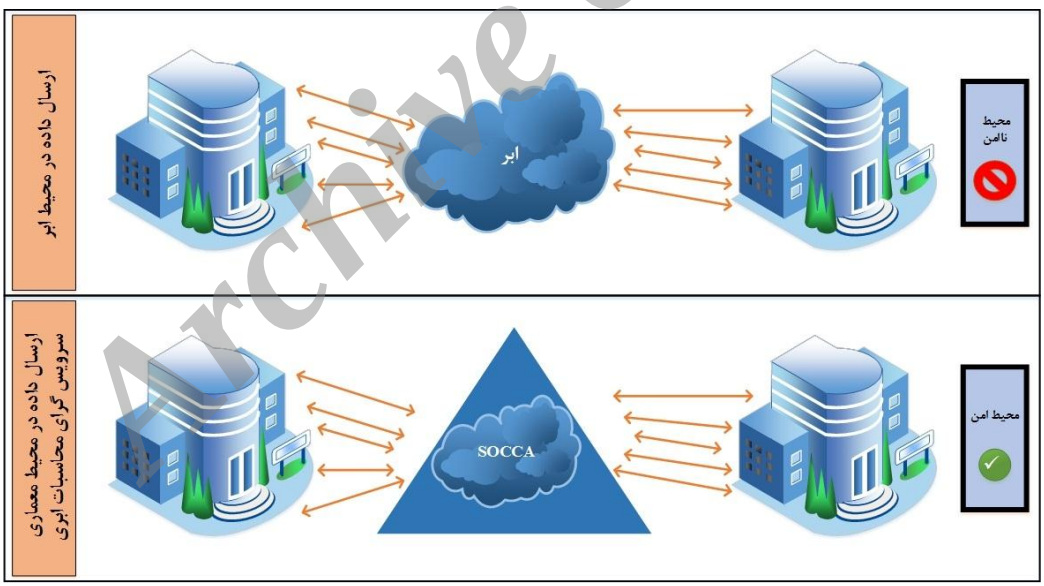
۳- آنتولوژی ارتباطی: شامل تعریف مفاهیم و گرایش مربوط به الگوی ارتباطات از جمله ابرها، همانند کد گذاری الگو و مسیر پیغام‌ها هستند.

آنتولوژی‌ها باعث می‌شوند که همه ایجادگرها، با یک چارچوب[9] به منظور تشکیل ابر، موجودیت پیدا کنند، اما در عوض هر ایجادگر ابری در ابرها چارچوب‌هایی را می‌سازد- مانند آن چه که در CCOA وجود دارد - و هر ایجادگر ابری اطلاعاتی را به روش یکسان به اشتراک می‌گذارد (Tsai et al., 2010).

۳.۱.۳. لایه کارگزار (واسطه‌ای) ابری[10]

این لایه، با اطلاعاتی سروکار دارد که به ایجادگر[11](تامین کننده) ابری مربوط است. این لایه شامل اطلاعات ایجادگر ابری است که اطلاعاتی از قبیل قیمت گذار، سخت افزار، نرم افزار و سرویس‌های ایجاد شده عناصر هسته‌ای را شامل می‌شود. هر ایجادگر، رتبه بندیِ نحوه درجه بندی مشتریان مختص به خود را داراست. این مورد با مقایسه‌ی قیمت‌ها، قابلیت اطمینان و مرور دیدگاه افرادی که از آن ابر استفاده کرده اند، در ارتباط است. واسطه‌های ابری به عنوان عامل‌های مابین ایجادگر فردی ابری و لایه SOA عمل می‌کنند. هر سرویس عمده‌ی ابری، نوعی سرویس واسطه‌ای وابسته به خود را دارد و به طور کلی واسطه‌های ابری باید وظایف خاصی را انجام دهند (Tsai et al., 2010).

۲.۳. همپوشانی در محاسبات ابری و SOA



شکل ۴- مقایسه‌ای از ارسال داده در محیط ابر و SOCCA

محاسبات ابری به نسبت SOA مفهوم وسیع‌تری را پوشش می‌دهد که از طریق نرم‌افزار لایه‌ای ارائه‌ی پشته، کلیه پشته، حتی سخت‌افزار، را در بر می‌گیرد. با توجه به بحث ترکیب محاسبات ابری با SOA باید این مسئله را در نظر گرفت که SOA در بعضی از حوزه‌ها قوی‌تر از محاسبات ابری عمل می-

[8] Ontology
[9] Framework
[10] Broker Layer
[11] Provider

Archive of SID
www.SID.ir

کند و همینطور بالعکس؛ یعنی محاسبات ابری مواردی را در بر می‌گیرد که در آن قوی‌تر از SOA است. SOA و پردازش ابری می‌توانند به طور مستقل یا همگام، از روش دیدگاه مکملی، مورد توجه IT قرار گیرند. مساله مشخص شدن سرویس‌ها در ابعاد ابرها و SOA بسیار مهم است. که البته با توجه به پروتکل‌هایی نظیر REST، UDDI و SOAP،WSDL این امر ساده‌تر انجام می‌شود. بطور کلی مشخص شدن سرویس‌ها در داخل ابرها بحثی بسیار ضروری است.

در بحث ترکیب معماری SOA و محاسبات ابری می‌توان، دید سازمانی به مقوله محاسبات ابری داشت. لایه‌های کسب‌وکار در سیستم‌های IT دربرگیرنده این مقوله اند. با نگرش در سلسله مراتب محاسبات ابری و لایه SaaS می‌توان به این نتیجه رسید که معماری سرویس‌گرا به انعطاف‌پذیری بین مقوله‌های مدیریت ارتباط با مشتری[12]، برنامه ریزی منابع سازمان[13] و غیره به این فناوری کمک می‌کند. به طوری که با تغییر چالش‌های بین سازمانی و با کمک پروتکل‌های SOA تعامل پذیری در بین سازمان‌ها به سهولت انجام شود (Zhang & Zhou, 2009). همانطور که درحیطه مجازی‌سازی بحث شد، مجازی شدن، خلق نسخه مجازی از سیستم عامل است که وسیله ذخیره بر روی سکوی سخت‌افزاری است (He & Liang, 2012).

SOA دارای نظارت عمیقی در سیستم‌های خود است که این امر در محاسبات ابری به خوبی انجام نشده است و این خود یکی از مواردی است که امکان پسرفت فناوری را در محاسبات ابری به همراه دارد (شمس و مهجوریان، ۱۳۸۹)، (Williams, 2014)، (Marks, 2008). ولی این امر در SOA به صورت بسیار کامل و کارا رعایت می‌شود. توانایی تعیین خط مشی‌ها، در رابطه با خدمات و مدیریت تغییرات اعمال شده بر روی این سرویس‌ها، نشان دهنده قدرت در SOA است. SOA به خوبی می‌تواند تغییرات را در سیستم مدیریت کند. در این میان محاسبات ابری با ارائه منابع مورد نیاز به SOA کمک می‌کند تا قدمی سریع در پیشرفت فناوری بردارد. SOA استانداردهای خود را به همراه سرویس‌هایی برای برنامه‌های کاربردی تحت وب قرار می‌دهد. ولی محاسبات ابری سعی در سرویس دهی به کاربرانی دارد که از طریق ابر به نرم‌افزار دسترسی دارند. بوسیله SOA و محاسبات ابری می‌توان، یک همکاری سازمانی موثر و بسیار پویا را امکان پذیر ساخت.

وقتی در یک محیط ابری داده‌ها بین چندین سازمان ارسال می‌شوند، ممکن است یک محیط ناامن و سرشار از خطا داشته باشیم که باعث از بین رفتن درصدی از اطلاعات و داده‌ها شود. اما وقتی داده‌ها در محیط ابری مبتنی بر معماری سرویس‌گرا در بین سازمان‌ها رد و بدل می‌شوند، محیطی امن خواهد بود. معماری سرویس‌گرا، با توجه به فاکتورهایش نظیر استفاده از پروتکل‌ها، مفهوم سراسری، مفهوم برون‌سپاری[14] و ... محیط امنی را برای ما در پی خواهد داشت که داده‌ها در آن بدون چالش خاصی بین سازمان‌ها در ارتباط هستند. این مقوله در شکل شماره ۴، با توجه به دو محیط ابری و SOCCA به نمایش درآمده است.

۴. بررسی الگوریتم زمان‌بندی با توجه به مساله مشتری خدمتگزار

در این بخش، الگوریتم‌های زمان‌بندی‌ای تشریح خواهند شد که نیازهای کاربر را برطرف نموده و همچنین محدودیت‌های محاسبات ابری را از بین می‌برند. ابتدا استفاده از تئوری فروشگاه (مصرف کننده و سرویس دهنده) در زمان‌بندی محاسبات ابری مطرح می‌شود تا نیازهای کاربر معرفی گردد. در ابعاد دیگری از این موضوع، یک الگوریتم ارزش پایه زمان‌بندی ابری ارائه شده است.

۱.۴. برنامه زمان‌بندی منابع ابری

از آنجا که موجودیت منابع در خوشه‌های[15] ابری تغییر می‌یابد، دو فاز الگویی برای زمان‌بندی منابع ابری[16] ارائه شده است: بر طبق کاربرد، انتشار منابع ابری، موقعیت منابع و غیر قابل پیش بینی بودن. برای ذخیره منابع قبل از اجرای یک ماشین مجازی[17]، به منابع ابری نیاز است. الگوی زمان‌بندی‌ها در پروسه‌های زمان‌بندی باید اتمیک باشد، یعنی یا کامل اجرا شود یا اصلا اجرا نشود و لغو شود. برنامه زمان‌بندی منابع ابری می‌تواند مانند مشتری (مصرف کننده)[18] و فروشنده (تهیه کننده)[19] ارائه شود که به آن تئوری فروشگاه[20] می‌گوییم. ابر، یک مکان فروش برای مشتری‌ها و فروشنده‌هاست. ما نیاز و پیشنهاد مشتری‌ها را ارائه می‌کنیم. با توجه به ۳ فاکتور نیاز مشتری، پیشنهاد ارائه شده و هدف فرمول نهایی، به نتیجه بررسی الگوریتم زمان‌بندی با توجه به مساله مشتری خدمتگزار در شکل ۵ دست می‌یابیم (Yang, Yin, & Liu, 2011).

[12] Customer Relationship Management (CRM)
[13] Enterprise resource planning (ERP)
[14] Form of outsourcing
[15] Cluster
[16] Cloud Resource Scheduling
[17] Virtual Machine
[18] Consumer
[19] Supplier
[20] Market

Archive of SID
www.SID.ir

نیاز مشتری:	پیشنهاد:	فرمول نهایی:
$\{N_i *\{\alpha_i,\beta_i,\gamma_i\}\}, i=1..n$	$P_{\min} \le P \le P_{\max}$	$\sum_{i=1}^{n}\delta_{ji}*\alpha_i \le K_{j1},$ $\sum_{i=1}^{n}\delta_{ji}*\beta_i \le K_{j2},$ $\sum_{i=1}^{n}\delta_{ji}*\gamma_i \le K_{j3},$ $\sum_{i=1}^{n}\delta_{ji} = N_i,$ $P_{\min} \le P \le P_{\max}$

شکل ۵- فرمول‌های برنامه زمان‌بندی (Yang et al., 2011)

سه شرط برای رسیدن به بهترین هدف در این تئوری وجود دارد (Yang et al., 2011):

۱. P کمتر از Pmin
۲. P مشتری را راضی میکند
۳. P بیشتر از Pmax

در شکل ۶، این سه شرط به همراه کل برنامه زمان‌بندی ابری به نمایش درآمده است.

بررسی شرط اول: به این معنی است که P از خواسته‌های کاربر بهتر است، ولی کاربر می‌خواهد به بالاترین قیمت منابع برسد که بالاترین اجرا را دارد. بنابراین می‌توانیم فرایند زمان‌بندی را با ارائه فروشنده با کم‌ترین قیمت تطبیق دهیم. از آنجایی که ما اول N1 بعد N2 و ... را چک می‌کنیم، این یک تطبیق جزئی است. در جریان تطبیق، ما تفاوت بین جدیدترین موارد Ni و قدیمی‌ترین آنها را محاسبه می‌کنیم. اگر قیمت نهایی P به اضافه تفاوتش بیشتر از Pmin باشد حلقه شکسته می‌شود. توزیع باقی مانده از Ni+1 به Nn بدون تغییر باقی می‌ماند. این تطبیق می‌تواند زمان زیادی را ذخیره کند.

بررسی شرط دوم: برای شرط دوم ما به راه حل درست می‌رسیم. بنابراین زمان‌بندی کامل است.

بررسی شرط سوم: شرط سوم بدین معنی است که اگر منابع قیمت پایین همچنان موجود نباشد، سعی می‌کنیم بهترین محاسبه را داشته باشیم. بنابراین قیمت تمام موارد نمی‌تواند راضی کننده باشد ؛ حتی اگر فروشنده‌های معمول فروشگاه را راضی می‌کند.

- یک حلقه برای: مقدار ۱ تا n مشتری
- یک حلقه برای: مقدار ۱ تا n فروشنده
- قیمت فروشنده برای آیتم هایی که داشته است (ریشه‌ها، حافظه‌ها، دیسک‌ها)، با توجه به INSTJ
- اتمام حلقه / ایجاد حلقه جدید برای: مرتب کردن قیمت واحد
- اگر فروشنده راضی باشد، در INSTJ قرار بگیرد
- پایان دو حلقه.
- محاسبه قیمیت نهایی P، اگر P < PMax بود، به کاربر گزارش میدهد قیمت نهایی کمتر از P باشد.
- اگر P > PMin بود، در INSTJ قرار بگیرد
- ایجاد یک حلقه جدید: جهت رضایت فروشنده
- اگر فروشنده راضی باشد، در INSTJ قرار بگیرد
- پایان حلقه
- در نهایت اگر P < Pmin باشد، گزارش می‌دهد به کاربر که حدقل قیمت خیلی زیاد است و پیشنهاد می‌دهد که کاربر حداقل قیمت را کم کند.
- پایان برنامه: بهترین حالت این است که P مشتری را راضی کند.

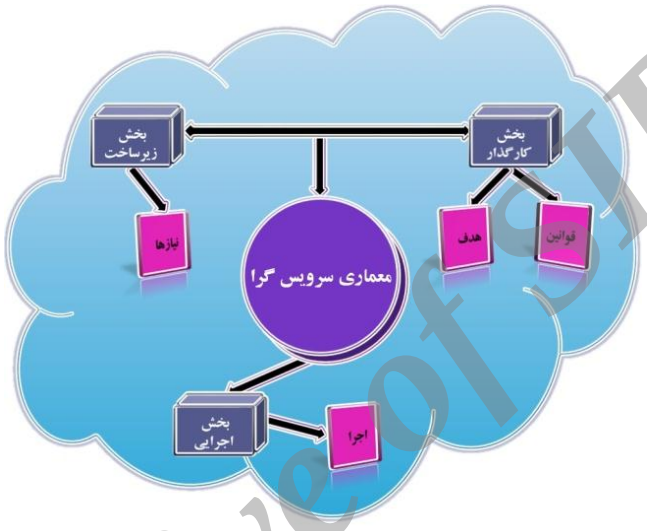
شکل ۶- برنامه پایه زمان‌بندی ابری (Yang et al., 2011)

Archive of SID

www.SID.ir

۵. ارائه یک روش ترکیبی برای بهبود الگوریتم زمان‌بندی در محاسبات ابری با توجه به حیطه SOA

ارائه خدمات محاسبات ابری درمورد روش جامع برای هر مدل تخصیص منبع، از اهمیت خاصی برخوردار است. لذا باید هر دو مورد منابع محاسبه و منابع شبکه را برای ارائه دقیق در الگوریتم زمان‌بندی در نظر بگیریم. محاسبات ابری در ابتدا برای شروع کارهای محاسبه‌ای تحت شبکه ایجاد شده‌اند. در این بحث به حیطه منابع مشتریان از لحاظ دسترسی و از لحاظ چگونگی منابع فیزیکی با توجه به لایه بندی در معماری SOCCA، اشاره می‌شود. تاخیرهای انتقال داده یکی از موانع اصلی است که با توجه رشد انبوه مشتری با آن مواجه می‌شویم. به عبارتی دیگر می‌توان گفت که وقتی مقدار زیادی داده به محیط تخصیص می‌یابد، کارایی خدمات شبکه مربوط به سیاست‌های کلی در قبال فرآیندهای دیگر در رابطه با زمان‌بندی خواهد بود. در نتیجه گفتنی است که مشتریان خصوصیات SOA را در ابر به عنوان عامل اصلی در تصمیم‌شان برای انتقال در نظر خواهند گرفت. مدل پیشنهاد شده در این بحث، با توجه به پارامترهای ارائه شده، همراه با چالش‌های تخصیص منابع مشتری/خدمتگزار و براساس دیدگاه لایه‌ای SOCCA و لایه کارگزار آن است.



شکل ۷- ساختار عمومی گره، الگوریتم زمان‌بندی محاسبات ابری مبتنی بر SOA

تئوری مشتری/خدمتگزار با توجه به الگوریتم‌های زمان‌بندی محاسبات ابری که مبتنی بر معماری‌ای برگرفته از معماری سرویس‌گرا است - همان گونه که در شکل ۷ به نمایش درآمده است- به صورت زیر بررسی می‌شود:

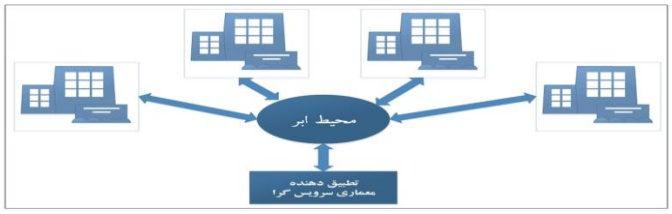
در ابتدا باید نیاز مشتری بررسی شود. یکسری فاکتورها برای اجاره کردن درخواست می‌شود. $N_i . \{N_1, N_2, ... N_n\}$ تعداد ورود شمارنده i است. که منابع مورد نیاز شامل: ریشه‌ها a_i، حافظه‌ها B_i و دیسک‌ها Y_i است. نیاز مشتری با استفاده از فرمول ۱ بررسی می‌شود:

$$\{N_i * \{\alpha_i, \beta_i, \gamma_i\}\}, i = 1..n \quad (۱)$$

سپس باید یک هدف کلی از کار ارائه شود. که این هدف می‌تواند از سوی مشتری پیشنهاد شود که در محدوده زیر است:

$$P_{\min} \le P \le P_{\max} \quad (۲)$$

مکان‌های مشتری به قیمت نهایی فاکتورهایی که در نظر دارد، محدود می‌شود. یعنی قیمت نهایی باید بیشتر یا برابر با $P_{\min}$ و یا کمتر یا مساوری با $P_{\max}$ باشد.



شکل ۸- تطبیق دهنده SOA برای هماهنگی n واحدی سازمانی در محیط ابر

Archive of SID
www.SID.ir

پس از تعیین اهداف، در این تئوری باید به یک قانون جامع رسید، تا مفهوم SOA در قالب n واحد سازمانی در محیط ابری همانند شکل ۸ مشخص گردد (این قانون برای پیدا کردن n محیط سازمانی است).

بر اساس اصل استقرا ریاضی $(\forall P)\left[P(0)\wedge(\forall k\in \mathrm{N})(P(k)\Rightarrow P(k+1))\right]\Rightarrow(\forall n\in \mathrm{N})[P(n)]$، فرمول ۳ برقرار است:

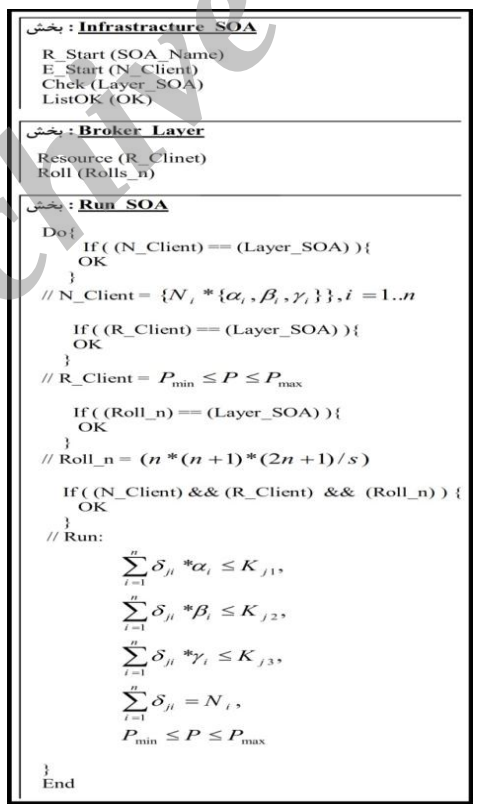
$$(n*(n+1)*(2n+1)/s) \quad (۳$$

با توجه به نیاز، هدف و قانون گفته شده، فروشنده در نهایت با فرمول ۴ به هدف نهایی خود می‌رسد.

$$\begin{aligned}
&\sum_{i=1}^{n}\delta_{ji}\ {}^{*}\alpha_i \le K_{j1},\\
&\sum_{i=1}^{n}\delta_{ji}\ {}^{*}\beta_i \le K_{j2},\\
&\sum_{i=1}^{n}\delta_{ji}\ {}^{*}\gamma_i \le K_{j3},\\
&\sum_{i=1}^{n}\delta_{ji} = N_i,\\
&P_{\min}\le P\le P_{\max}
\end{aligned} \quad (۴$$

نتیجه، رسیدن به حداقل قیمت P است که مشتری را راضی نگه می‌دارد.

کار در سه بخش انجام شده است: در بخش زیرساخت SOA نیاز مشتری گفته شده است، سپس در لایه Broker یا همان بخش Broker، هدف و قانون کار آماده می‌گردد. در نهایت با توجه به مسائل مطرح شده در اجرای کار به نتیجه اصلی دست می‌یابیم. مطالب بررسی شده در قالب شکل ۹ به نمایش درآمده است.



شکل ۹- الگوریتم پایه زمان‌بندی محاسبات ابری مبتنی بر معماری سرویس‌گرا

Archive of SID

www.SID.ir

حال با توجه به این الگو - مفهوم کامل یکپارچگی بین سازمان‌ها توسط لایه Broker (واسطه‌ای) که بیان شده است- می‌توانیم کلیه سازمان‌ها را در یک حیطه منحصر به فرد در راستای بهبود سطح فعالیت‌های بین سازمانی تعریف کنیم. در این حیطه، یک الگو برای همه سازمان‌ها توسط لایه واسطه‌ای ایجاد شده است و کلیه داده‌ها، فرآیندها، منابع در این حیطه یکی شده و سطح امنیت در این معماری جدید بهبود می‌یابد و همچنین رضایت مشتری با توجه به صرفه جویی از سایر پارامترهای تحت لایه واسطه‌ای بیشتر خواهد شد.

۷. نتیجه گیری

محاسبات ابری، به عنوان یک شاهکار خارق العاده، همه ویژگی‌های عالی را داراست، هرچند هنوز محاسبات ابری دور از واقعیت ذات اصلی خود بوده و تحقیق و کنکاش بیشتر در این زمینه لازم است. معماری سرویس‌گرا، می‌تواند در پیشرفت فناوری ابری کمک شایانی نماید. فناوری قدرتمند محاسبات ابری، سبب ایجاد یک پایه و اساس ایده‌آل برای همگام شدن و مکمل شدن با معماری سرویس‌گرا شود که با توجه به موقعیت IT سال‌های اخیر در دنیای امروزی، می‌تواند به رشد و پیشرفت در این زمینه منجر شود. دراین مقاله الگوریتم‌هایی را مورد بحث قرار دادیم که برای محاسبات ابری بسیار ضروری می-باشند و محاسبات ابری زمان‌بندی منابع خود را بر اساس این الگوریتم‌ها انجام می‌دهد. از سوی دیگر، شناخت و بررسی لایه‌های SOCCA، شمار زیادی از مسئله‌ها مانند عملکرد مفید در الگوریتم زمان‌بندی محاسبات ابری را برای ما آشکار می‌سازد. همچنین، معماری لایه‌ای SOCCA به ما در فهم منطق میان دو مقوله SOA و محاسبات ابری کمک می‌کنند؛ زیرا بدون اطلاع از لایه‌ها در SOCCA، نمی‌توان به مدل ایده‌آل و ترکیبی در رابطه با الگوریتم زمان‌بندی در حیطه رضایت‌مندی مشتری دست پیدا کرد. با توجه به مساله زمان‌بندی در محاسبات ابری می‌توان از حیطه SOCCA بهره گرفت، همچنین می‌توانیم به کمک نیازهای مشتری در رابطه با بحث SOCCA یکسری از درخواست‌ها را با توجه به پارامترهای ریشه، حافظه و دیسک‌ها با توجه به محدودیت‌های مشتری بهبود بدهیم. این طرح ارائه شده همانند الگویی از پیش تعریف شده وارد عمل می‌شود، که با توجه به مساله یکپارچگی سازمان‌ها می‌تواند به بهترین نحو عمل کند و فاکتوری مهم برای رضایت‌مندی مشتریانی باشد که تحت محیط محاسبات ابری، سعی در اجرای اهداف خود دارند. در آینده می‌توان این ایده را با دیگر پارامترهایی نظیر سرعت و انرژی مصرفی در مراکز داده‌ای توزیع شده در محاسبات ابری در نظر گرفت که با توجه به آنتولوژی‌ها در SOCCA و لایه واسطه‌ای، می‌توان این بحث را کامل نمود.

منابع

1. Tsai W., Sun X., Balasooriya J,. 2010. Service-Oriented Cloud Computing Architecture, Seventh International Conference on Information Technology, Las Vegas, NV
2. Barry D.K., Dick D., 2013,Web Services,Service-Oriented Architectures, and Cloud Computing. Elsevier. Second Edition.Theusa
3. فریدون شمس و امیر مهجوریان، ۱۳۸۹، " معرفی اصول و روش‌های معماری سازمانی سرویس گرا "، مرکز چاپ و انتشارات دانشگاه شهید بهشتی،چاپ اول، تهران
4. National Institute of Standards and Technology(NIST).http://www.nist.gov/itl/csd/cloud-102511.cfm
5. Dillon T., Wu C., Chang E.,2010,Cloud Computing: Issues and Challenges, 24th IEEE International Conference on Advanced Information Networking and Applications, Perth, Australia
6. Rittinghouse W.J., Ransome F. J., 2010,Cloud Computing Implementation Management and Security. CRC PressTaylor & Francis Group.USA ,NY
7. Magoulès, F., Pan, J., & Teng, F. (2012). *Cloud computing: Data-intensive computing and scheduling*. CRC Press.
8. Waris, M., Khan, S. A., & Fakhar, M. Z. (2013, October). Factors effecting service oriented architecture implementation. In *Science and Information Conference (SAI), 2013* (pp. 1-8). IEEE
9. Williams, W. (2014). Security for Service Oriented Architectures. CRC Press
10. Zhang L., Zhou Q., 2009,CCOA: Cloud Computing Open Architecture,IEEE International Conference on Web Services, Los Angeles, CA
11. He, Z., & Liang, G. (2012, October). Research and evaluation of network virtualization in cloud computing environment. In *Networking and Distributed Computing (ICNDC), 2012 Third International Conference on* (pp. 40-44). IEEE.
12. Marks, E. A. (2008). *Service-oriented architecture (SOA) governance for the services driven enterprise*. John Wiley & Sons.
13. Yang, Z., Yin, C., & Liu, Y. (2011, October). A cost-based resource scheduling paradigm in cloud computing. In *Parallel and Distributed Computing, Applications and Technologies (PDCAT), 2011 12th International Conference on* (pp. 417-422). IEEE.

Archive of SID

www.SID.ir