

فهرست

- .....سمتیک های سرویس وب WSDL-S
- .....۱- مقدمه
- .....۲- ملزومات ویژه سمتیکهای سرویس وب
- .....۳- یک مثال
- .....۴- استفاده از عناصر توسعه پذیر WSDL
- .....۵- پشتیبان ۱-۱ WSDL
- .....۶- مراجع
- .....۷- ضمیمه A: شرح طراحی الگو با استفاده از XSLT
- .....۸- ضمیمه B: شرح طراحی الگو با استفاده از XQUERY
- .....۹- ضمیمه C: آنتولوژی سفارش خرید
- .....۱۰- ضمیمه D: انتخابات طراحی

سمانتیک های سرویس وب - WSDL-S

یادداشت فنی

ورژن ۱/۰

آوریل، ۲۰۰۵

نویسندگان (به ترتیب حروف الفبا):

IBM ،تحقیق Rama Akkiraju

IBM ،گروه نرم افزار Joel Farrel

John Miller ، آزمایشگاه ISDIS ، دانشگاه جورجیا

Meenaksi Nagarajan ، آزمایشگاه ISDIS ، دانشگاه جورجیا

Marc - thomas schmiat ، گروه نرم افزاری IBM

Amit sheth ، آزمایشگاه ISIS ، دانشگاه جورجیا

Kunal verma ، آزمایشگاه ISBIS ، دانشگاه جورجیا

حق انحصاری مورد نظر

کپی رایت ۲۰۰۵ با همکاری مرکز کاربرد ماشینی داد و ستد (تجارت) جهانی و دانشگاه

ویرجینا است تمام حقوق محفوظ می باشد.

IBM و دانشگاه ویرجینیا به هر حال به شما اجازه کپی برداری و استفاده از خدمات سمانتیک های سرویس وب WSDL-S یادداشت فنی، حداقل بدون هزینه وامانت داری. فراهم آورده اند که شامل تمام کپی های حاصله از آنوتیشن های سمانتیک سرویس وب WSDL-S یادداشت فنی یا قسمت های دیگری می شود که شما به آن دست می یابید:

- ۱- ارتباط (لینک) یا URL به قسمت های خاص این مکان
- ۲- حق انحصاری مورد نظر همانگونه که در سمانتیک های سرویس وب WSDL-S یادداشت فنی نشان داده شده است.

به جز در مواردی که کپی داری در بالا تضمین شده است، نویسندگان به صورت درونی (مفهومی) یا بیرونی هیچ حقی در استفاده از موارد دیگر یا کنترل آنها را ندارد سمانتیک های سرویس وب WSDL-S در اطلاعات فنی آنچه که هست تهیه کرده و نویسندگان هیچ مجوز یا گواهی نامه ای، به صورتی داخلی یا خارجی که شامل مجوز و گواهی ارائه هدف خاص یا موضوعی باشد ندارند. این است که محتوای سمانتیک های سرویس وب WSDL-S در اطلاعات فنی مناسب برای دستیابی به هر هدفی است، اما چنین کاربردی برای چنین مفاهیمی به معنای استفاده از کپی رایب تجارت ها و حقوق دیگر نمی شود.

(مبتکران) نویسندگان هیچ حق مستقیم، غیر مستقیم، خاص، اتفاقی یا دائمی را در آسبهای حاصله یا مربوط به استفاده یا توزیع و گسترش سمانتیک های سرویس وب WSDL-S را در اطلاعات فنی ندارند.

نام و مارک تجاری مبتکران ممکن است در هر حالتی استفاده نشود که شامل تبلیغ یا نمایش عمومی موارد خاص یا محتوای آن بدون هیچ مجوز قبلی می شود. موضوع کپی رایت در سمانتیک های سرویس وب WSDL-S اطلاعات فنی در هر زمانی بر عهده مبتکران آن مورد است.

هیچ حقوق دیگری از طریق کاربرد و موارد دیگر وجود ندارد.

## چکیده

استاندارد WSDL متداول در سطح نحوی کار می کند و فاقد توصیف معنادار مورد نیاز برای نشان دادن ملزومات و امکانات سرویس های وب می باشد.

سمانتیک ها می توانند استفاده مجدد و کشف نرم افزاری را اصلاح کنند و به طور قابل توجهی ترکیب سرویس های وب را تسهیل بخشند و تکمیل برنامه های کاربردی میراث را که به عنوان بخشی از مجتمع سازی فرایند تجاری مقدور سازند اطلاعات فنی سمانتیک های سرویس وب مکانیزی برای مرتبط کردن تفاسیر معنادار با سرویس های وب مشخص می کند که با استفاده از زبان تصیف سرویس وب WSDL شرح داده می

شود این زبانی مبنی بر این مفهوم است اما با اصلاح قابل توجه به جزئیات در این طرح در این طرح ما فرض می کنیم که مدل های رمسی سمتیک مربوط به سرویسها اخیراً بوجود آمده اند. در روش ما، این مدل ها خارج از اسناد WSDL محافظت می شوند و از طریق عناصر توسعه پذیر WSDL از سند WSDL ارجاع می شوند. نوع اطلاعات سمتیکی که برای شرح سرویس وب مفید است پیرامون مفاهیم تعریف شده توسط

ارتباط وب سمتیک در OWL-S [owl-s] و سایر تلاش های [METEOR-SWSMO] است. اطلاعات سمتیک معین شده در این سند دربرگیرنده تعریف جاسازی پیش از پردازش، ورودی، خروجی و تأثیرات عملیات سرویس وب است. این روش مزایای

چندگانه ای را در مورد OWL-S پیشنهاد می دهد. ابتدا، کاربرد می تواند به روش سازگار و در حال پیشرفت جزئیات سطح عملیاتی و سمتیکی را در WSDL زبانی که یک توسعه دهنده ارتباط با آن آشنا است را شرح دهد. ثانیاً، با خارج کردن مدل های دمین سمیک ما یک روش آگنوستیک برای زبانهای بازنمایی آنتولوژی انتخاب کردیم. این کار به توسعه دهندگان (ارتقاء دهندگان)؟؟ وب اجازه می دهد تا سرویس های وب خود را با انتخاب زبان آنتولوژی (نظیر UML یا OWL) بر خلاف OWL-S، تفسیر کنند. این مسئله مهم است زیرا توانایی در استفاده مجدد از مدل های دمین موجود مطرح شده در زبان های مدل سازی نظیر زبان UML می تواند نیاز به سمتیک های مدل مجزا را به شدت کاهش دهد. در آخر، به روزرسانی تجهیزات موجود پیرامون WSDL ساده است و روش افزایشده ما را سازگار می کند.

وضعیت

این یک یادداشت فنی است که به منظور بحث تهیه شده است و فیدبک موجود در روش را برای اضافه کردن سمتیکها به تعریفات سرویس های وب استنباط می کند.

## ۱- مقدمه

به عنوان مجموعه ای از سرویس های وب توسعه یافته، به صورت قابل توجهی داشتن وسایل اتوماتیکی برای آن مهم بوده برای اینکه کمکی به سرویس های مشخص منطبق بر نیازهای درخواست کننده ارائه دهد یافتن سرویس های وب مناسب بستگی به امکانات وجود ارائه دهندگان سرویس برای توصیف ویژگی های سرویس های آنها و درخواست کنندگان سرویس برای توصیف ملزومات خود به حالت ایده آل و واضح در فرم توصیف - ماشینی دارد. اضافه کردن سمانتیک ها برای ارائه ملزومات و درخواست ها از سرویس های وب برای این مورد آشکار و با تفسیر ماشینی ضروری است.

مزایای استفاده از سمانتیک ها چرخه وجود آنها را در سرویس های وب گسترده تر می سازد. در مدت توسعه، ارائه کننده سرویس می تواند سمانتیک های مورد نظر را با ارائه قسمتهای مناسب سرویس وب با مفاهیم موجود در مدل سمانتیک غنی تر مورد ارزیابی قرار دهد. از زمانی که مدل های سمانتیک توافقی را بر روی معنا و استفاده مورد نظر از شرایط انجام داده است و ممکن است تعاریف رسمی و غیر رسمی از ورودی ها انجام دهد، ابهام کمتری در سمانتیک های مورد نظر برای تهیه کننده آنها وجود دارد. در مدت کشف، درخواست کننده سرویس توصیفی را از ملزومات سرویس با استفاده از شرایط

در مدل سمانتیک بیان کرده است. تکنیک های منطقی را می توان برای یافتن تشابه سمانتیکی بین ارائه سرویس و درخواست به کار برد. در طول ارائه این مورد زمینه عملی تفاسیر برای افزایش کاربردی کردن سرویس های چندگانه و برای ارائه سرویس های مناسب انجام شده است. مهمتر اینکه، سمانتیک ها این امکان را فراهم می کنند که طراحی های خاص بین اطلاعات مبادله شده از طریق XML- اصلی با پیام های SOAP بوجود آید، که به طور قابل توجهی برای انجام دادن با ارائه نحوی پیشنهادی در استانداردهای اخیر مشکل می باشد. در طول ساخت، طراحی را می توان برای تبادل اطلاعات استفاده کرد. بنابراین، زمانی بیان شده، که سمانتیک های توانایی نفوذ را با استفاده از ابزارهایی برای خود کار کردن سرویس در کشف، محاسبه و الگوبرداری دارد. استانداردهای WS متداول در سطح نحوی کار کرده و فاقد توانایی های ارائه شده سمانتیکی هستند. این مورد مانعی برای توسعه ابزارهای کمک کننده به بشر و یا نگهداری و حفاظت از مرحله نیمه اتوماتیک و ترکیب کاربردی هستند. در این اطلاعات فنی، این مشکل را با بکار بردن موردی در گروه سرویس وب برای رساندن آن به استانداردهای سرویس وب مورد توجه قرار داده ایم.

کانسورتیوم وب گسترده جهانی (WTC) در معماری سرویس های وب (W3CWSA) دو جنبه از توصیف کامل یک سرویس وب را ارائه کرده است. اول اینکه توصیف



کاربردی نحوی به نحوی است که در WSDL بیان شد. مورد دوم به گونه‌ای که در سمانتیک های سرویس بیان شده، ارائه شده و با مورد خاصی پوشش داده نشده است. به طور عملی، توصیف سمانتیک به صورت غیر رسمی بیان شده یا از بین رفته است. با امتحان توصیف WSDL در یک سرویس نمی توانیم به صورت غیر مبهم و آشکاری آنچه را که سرویس انجام می دهد مشخص کنیم. می توانیم حالت نحوی را در ورودی ها و خروجی ها ببینیم اما در واقع آنچه که از آنها معنی می دهند یا تغییراتی را که در محیط سرویس انجام می شود نمی دانیم که معنی پارامترها چیست یا اینکه شرایط مربوط به اسناد ارائه شده را نمی دانیم. بی تردید هر دو سرویس می توانند دارای تعریف نحوی مشابهی باشند اما از نظر عملکردی لزوماً متفاوت عمل می کند، بدینگونه دو سرویس متفاوت از نظر نحوی با عمل مشابهی روبرو هستند.

ساخت سمانتیک در سرویس های وب به عنوان روشی برای موارد بالا است پیشنهادی نمونه شامل موارد پیش فرض (آغازگرها)، پروژه ها و زبانهایی همچون SWA/SWSL , METEOS-S , OWL-S می شوند. در حالیکه توسعه سمانتیک در OWL-S غنی و انعطاف پذیر بوده، بیشترین روش تحقیقی قابل رؤیت برای اطلاعات است. روش جدیدی برای توصیف سرویس های وب بوده و دارای محدودیت های قابل توجهی می باشد اول اینکه این مورد مغایرتی با استانداردهای سرویس موجود ندارد.

به عنوان مثال، در حالیکه مدل‌های زمينه در OWL-S در موارد کلی WSDL استفاده می شود مدل طولی OWL-S توصیفی از بقیه موارد WSDL می باشد. دوم اینکه این فرض وجود دارد که هر کسی از OWL برای ارائه مواردی استفاده می کند که همیشه مورد وجود ندارد برای غلبه بر چنین محدودیت های روش جدیدی را در این مورد ارائه داده ایم مشاهدات مشابهی برای بقیه پیشنهاد مشخص در بالا به کار می رود.

این روش توصیف شده در این اطلاعات فنی دارای حالت پیچیده و تکاملی در استانداردهای سرویس وب موجود و به صورت تخصصی تر در توصیف سرویس وب می باشند. این مرور مجددی بر پیشنهاد WSDL-S از گروه METEOR-S در دانشگاه جورجیا است. در این روش، به تأثیر WSDL با سمانتیک ها را به وسیله به کار بردن مفاهیم مشابه در OWL-S بهبود بخشیده ایم در حالیکه به سمانتیک زبان ارائه شده نا آشنا است. در این روش تنها به مدل طولی OWL-S (اجزای OWL-S که مشخص کننده کاربرد سرویس های وب است) و مدل OWL-S (اجزای OWL-S که توصیف کننده پروتکل متقابل سرویس های وب است) در مقایسه با BPE14WS پرداخته و در اینجا بحث نشده است. مزیت این روش توسعه یافته افزودن سمانتیک هایی به WSDL در حالت چند بعدی است اول اینکه روش هماهنگ فزاینده ای، هر دو مورد سمانتیک ها و سطح عملکرد جزئیات در WSDL را توصیف کرده، زبانی را ارائه داده که گروه

توسعه یافته با آن آشنا هستند. دوم اینکه با ارائه مدل های برجسته سمانتیک ما روش آگنوستیک را برای ارائه زبانهای موجود به دست آوردیم. این مورد به ارائه دهندگان سرویس وب برای ارائه خدماتشان با انتخاب زبانهایی (همچون UML و OWL) مجوز می دهد. این مورد به صورت برجسته ای وجود دارد از زمانیکه استفاده مجدد از مدل های برجسته موجود در زبانهای مدل همچون ULM به طور کلی حذف شده و نیاز به سمانتیک هایی در مدل جداگانه دارد. به هر حال، این روش نیاز به ارائه وجود چندین مورد را تشخیص می دهد که به صورت موارد مشابه یا متفاوت با سرویس وب واحد هستند و مکانیسمی را برای انجام آن ارائه می دهند. نهایتاً، مدرن کردن تجهیزات موجود در اطراف WSDL برای جای دادن روش واقعی ساده به نظر می رسد. در حالیکه بیان شده موارد فرضیه در OWL-S در توصیف منطقی زبان غنی تری را برای ارائه سمانتیک ها به کار می برند، معتقدیم که توسعه استانداردهای فنی همچون WSDL شامل سمانتیک هایی است که در روش عملی تر و کاربردی قابل پذیرش هستند. به هر حال با گسترش مدل های برجسته سمانتیک در پیشنهاد خود هنوز هم اجازه آماده سازی و ارائه مفاهیم برجسته و ارتباط زبانهایی همچون OWL و UML است که با یکدیگر هر دو مورد بهترین حالت را دارند.

## ۱-۱- لغات فنی

تعاریف اصلی را در قسمت لغات فنی که در اطلاعات فنی استفاده کرده ایم، در اینجا

ارائه می دهیم.

سمانتیک ها - سمانتیک ها در این مقاله به معانی اطلاعات و مفاهیم اطلاق می شود.

مدل سمانتیک ها - مدل سمانتیک شرایط و مفاهیمی برای توصیف و ارائه حوزه

اطلاعاتی یا چندین قسمت از پیرامون (آن) است که شامل سیستم نرم افزاری می شود

مدل سمانتیک معمولاً شامل مفاهیم برجسته ای از مارد ارائه شده، ارتباط بین آنها

ویژگیهایشان و مقادیرشان است. معمولاً این مورد بررسی کلی است که نشان دهنده

توافق است.

تفسیر سمانتیک - تفسیر سمانتیک اطلاعات اضافی در یک داکيومنت است که تعریفی از

سمانتیک های یک قسمت را در آن داکيومنت ارائه می دهد. در این اطلاعات فنی تفاسیر

سمانتیک به عنوان اطلاعات اضافی در داکيومنت WSDL است. آنها تعریف سمانتیک

ها را با رجوع به قسمتی از مدل سمانتیک تعریف می کنند که توصیف سمانتیک از یک

قسمت از داکيومنت تفسیر شده است.

سمانتیک های ورودی - سمانتیک های ورودی به معنای پارامترهای ورودی هستند که

با چند سمانتیک مدل تعریف شده اد.

سمانتیک هی خروجی - سمانتیک های خروجی به معنای پارامترهای خروجی است که با چند سمانتیک مدل تعریف شده است.

پیش پردازش - پیش پردازش مجموعه ای از چندین عبارت سمانتیک (با عبارتی است که در مفاهیم یک مدل سمانتیک استفاده شده اند) می باشند که نیاز به تصحیح دارند قبل از اینکه عملی به طور موفق بدست آید.

افکت مجموعه ای از عبارات سمانتیک است که باید پس از عمل مقدار خروجی را کامل کند. افکت های مختلف بستگی به این دارند که عملکرد موفقیت آمیز باشد یا نباشد.

## ۲-۱- مدل اطلاعات

شکل داکيومنت WSDL در نقطه اصلی برای توصیف سرویس های وب است. ساختن توانایی توصیفی WSDL، مکانیسمی را برای ارائه ویژگیها و ملزومات در سرویسهای وب با مفاهیم سمانتیک مربوط به مدل سمانتیک است. برای انجام این مورد مکانیسم هایی را برای مشخص کردن و ارائه پیش فرض ها و افکت ها در سرویسهای وب است. این پیش فرض ها و افکت ها با یکدیگر در ارائه سمانتیک خروجی و ورودی با حالت اتوماتیک در مرحله کشف سرویس است.

شکل ۱- ارائه بیرونی و ارتباط سمانتیک ها برای موارد WSDL

شکل ۱ نشان دهنده چگونگی تفسیر سمانتیک مربوط به عوامل گوناگون داکيومنت WSDL (شامل خروجی ها، ورودی ها و زمینه های کاربردی همانند عملکردها، پیش پردازش و افکت ها است) که مربوط به مفاهیم سمانتیک در مدل سمانتیک برجسته است. مدل برجسته شامل یکی از مفاهیم یا بیشتر است.

۳-۱- قراردادهای برجسته

کلیدهای "OPTIONAL" "MAY" "RECOMMENED" "SHOULDNOT" "SHOULD" "SHALL NOT" "SHALL" "REQUIRED" "MUST NOT" در

این داکيومنت همانگونه که در TEC2119 توصیف شده اند.

۴-۱- نیم سپیسز

در اسامی نیم سپیسز URIS با این اطلاعات فنی به گونه زیر تعریف شده است:

۲- ملزوماتی برای سمانتیک های سرویس وب

پیشنهاد کرده ایم که اصول خاصی هر کاری را برای تعریف سمانتیک های سرویس وب به کار می برد. کار ما با اصول زیر به دنبال است.

؟ ساخت استانداردهای سرویس های وب موجود: استانداردهای سرویس های وب به

سرعت تکنولوژی برای ترکیب کاربرد می سازد به دلیل اینکه کاربرد درونی آنها وجود

دارد شرکت ها سرمایه گذاری را در پروژه های ترکیبی براساس سرویسهای وب

ارائه داده اند بنابراین معتقدیم که هر روشی برای اضافه کردن سمانتیک ها به سرویس های وب باید در حالت مشابهی مشخص شود به گونه ای که هیچ مشکلی را در سرویس های وب نصب شده اصلی ایجاد نکند.

؟ مکانیسم برای ارائه سرویس های وب با سمانتیک ها باید مستقل از زبان کنونی سمانتیک باشد در اینجا تعدادی از زبانهای ممکن برای ارائه سمانتیک ها همچون OWL و WSMO و UML می باشند. هر زبانی سطوح مختلفی از ارائه سمانتیک و حفاظت توسعه دهنده را ارائه می دهد. موقعیت ما به گونه ای که لازم نیست استانداردهای سرویس های وب به زبان کنونی سمانتیک خاصی پیوند بخورد. این مورد در هماهنگی با روش ارائه شده توسط SIVASHANMUGAM و همکارانشان بوده است. با نگهداری از مکانیسم تفسیر سمانتیک مجزای از ارائه توصیف های سمانتیک، این روش ارائه دهنده انعطاف پذیری در گروه توسعه دهنده برای انتخاب زبان ارائه شده سمانتیک است.

در قسمت بعدی روش موفقیت چنین استقلالی را توضیح می دهیم.

؟ مکانیسم برای تفسیر سرویس های وب با سمانتیک ها باید اجازه ارتباط تفاسیر چندگانه در سمانتیک های مختلف از زبانهای خاطر را ارائه دهد:

همانگونه که قبلاً بیان شد، بسیاری از سمانتیک های ممکن در زبانهای کنونی وجود دارد. ارائه دهندگان سرویس ممکن است برای تفسیر این سرویس ها در سمانتیک های چندگانه با زبانهای ارائه شده چند مورد مختلف را کشف کنند. بنابراین معتقدیم که مکانیسم برای تفسیر سرویس های وب با سمانتیک ها باید اجازه تفاسیر چندگانه مربوط به سرویس های وب را بدهد.

نگهداری تفسیر سمانتیک از سرویس های وب که انواع اطلاعات را در طرح xml ارائه داده است، تمرین معمول در ترکیب سرویس های اصلی وب استفاده مجدد از رابطه ای مشترکی است که در xml بیان شده است، تعریف داکومننت های تجاری استفاده شده در نمونه xml یک مورد موفقیت آمیز و موفق می باشد نمونه های xml به عنوان فرمت تعریف اطلاعاتی برای موارد قابل پیش بینی در آینده است.

معتقدیم که تفاسیر سمانتیک در ورودی ها و خروجی های سرویس باید از نمونه های xml در تفاسیر نگهداری کند. WSDL ۲/۵ استفاده از انواع سیستم های دیگر را در مجموع نمونه های xml حفاظت می کند، بنابراین ساختارهای موجود در مدل های سمانتیک مثل طبقاتی از موارد owl برای تعریف سرویس وب در انواع اطلاعات ورودی و خروجی قابل تعریف است. اما روشی که نمونه xml را مخاطب قرار نمی دهد قادر به



استفاده از موارد موجود نبوده و یا مجوز توسعه دائمی از داکيومنت های WSDL را که شامل سمانتیک ها می شود، ندارد.

ارائه حفاظت و نگهداری از مکانیسم نقشه برداری (ارائه طرح) پیشرفته بین انواع سرویس وب و موارد پیشرفته در ارائه وضعیت و شرایط اهمیت تفسیر نمونه های xml در توصیف های سرویس وب، توجه باید معطوف به چگونگی طراحی نمونه xml در انواع پیچیده برای موارد ارائه شده باشد. به عنوان مثال، چنانچه مدل مهمی در owl ارائه شود، طراحی بین موارد WSDLXSD و مفاهیم owl را می توان با هر زبان انتخابی استفاده کننده بیان کرد مثال: Xquery, XSLT, OWL, RDF و هر زبان دلخواه دیگری همانند زبان های انتخابی که کاملاً با محیط خودش مطابقت داشته باشد.

۳- یک نمونه

در اول نمونه ای از داکيومنت WSDL ارائه دادیم که با اطلاعات سمانتیک به خواننده مروری بر آنچه که در بقیه مطلب است ارائه می دهد. تفاسیر سمانتیک در بخش ۴ با نمونه هایی خاص گرفته شده از این نمونه بیان شده اند. ص ۷ از ۴۲

در این نمونه یک خرید ساده از سرویس پیشنهادی را ارائه می دهیم ورودی ها و خروجی هایی از سرویس Process Purchase Order با سمانتیک ها تفسیر شده اند، دو مورد جدید اساساً پیش پردازش ها و افکت هایی هستند که در عناصر موجود برای

ایجاد عملکرد در WSDL می باشد و مورد ارائه شده به عنوان طبقه بندی اضافه شده به ساختار رابط است مفاهیم سمانتیک و روابط آنها در قسمت های owl نمونه گذاری

شده اند. Purchase. owl

در داکيومنت WSDL، ورودی Process Purchase Order شامل موارد پیچیده ای است.

تعریف تفاسیر سمانتیک برای این انواع پیچیده در سطحی از موارد و انواع پیچیده در

این نمونه بیان شده اند. نقشه گذاری سطح اصلاحی جدید بعداً با جزئیات بیان می شود.

در این اطلاعات فنی، روش انتخابی برای تفسیر انواع پیچیده نیز پیشنهاد شده است

نمونه طراحی عملکردها در XSLT, Query با جزئیات در ضمیمه های A, B به ترتیب

بیان شده است. نگهداری کلی از نمونه های owl با شکل کارما ممکن است جزئیاتی

برای نوع owl وجود دارد که در ورژن قدیمی تر این کار موجود است [WSDL-s].

تعاریف XSD از تمام موارد گسترده در این داکيومنت در بخش بعدی است.

این نمونه بر اساس تعریف XMLNS است. تعریف نیم سپس wssem سه مورد بعدی

xml در داکيومنت های نمونه تعریفی از داکيومنت های ورودی و خروجی برای

سرویس هستند.

POIM.xsd که نشان دهنده ویژگی های این آیتم در مرحله خرید است و در زیر بیان

شده است.

POBilling.xsd نشان دهنده اطلاعات فاکتور در خرید است که در زیر بیان شده است.

در POBilling.xsd موارد پیچیده POBilling اساساً در bill to Address, Ship to

Address وجود دارند که در نمونه POAddress به عنوان نمونه پیچیده ای در خودش

است.

بنابراین، ویژگی تفاسیر سمانتیک برای این نمونه های پیچیده در مقایسه با موارد xsd- در

نمونه POAddress می باشند.

POAddress.xsd که به عنوان آدرسی در شرایط خرید است که در زیر داده شده است.

۴- استفاده از عناصر گسترده در WSDL

در این قسمت چگونگی تفاسیر سمانتیک به قسمت های داکيومنت WSDL را بررسی

می کنیم.

اتفاقاً ۲/۰ WSDL دارای ساختارهای بعدی برای آماده کردن توصیفات سرویس است.

رابط پردازشگر، عملکرد، پیام، سرویس و موارد پایانی (نهایی) را شامل می شود. از این

موارد ۳ مورد اول، ساختارهای پیام، عمل، رابط مربوط به تعریف انتزاعی سرویس است

در حالیکه سه مورد باقیمانده مربوط به عملکرد سرویس می شوند. در این اطلاعات

فنی، تمرکز اصلی ما بر روی تفسیر معنایی از تعریف انتزاعی یک سرویس برای کشف

پویا و ترکیب سرویس ها است (این مورد مهم است که تفاسیر سمانتیک باید در

کاربردهای سرویس به کار گرفته شوند. به عنوان مثال، چنانچه پیامی در پروتکل a مشابه نوع دیگری در پروتکل B باشد چنین اطلاعاتی را می توان در مدل‌های مشخص نشان داد و از آن در طول عملکرد استفاده کرد. به هر حال، ما تفسیر کاربرد سیستم منظومان نیست. تفاسیر سطح سرویس در قسمتی مورد توجه WS-POLYIY قرار گرفته است) مکانیسم های مرجع URL را از طریق عناصر گسترده برای رابط پردازشگر عملکرد و ساختارهای پیام تهیه کرده ایم برای اینکه به تفاسیر سمانتیک تعریف شده در مدل های مشخص سرویس ها اشاره شود.

خلاصه کلی از عناصر گسترده در این اطلاعات فنی به شرح زیر است:

؟ عنصر گسترده، اصطلاحاً مدل مرجع، برای طراحی یک به یک عناصر نمونه به مفاهیم در مدل سمانتیک

؟ کار (عملکرد) گسترده، اصطلاحاً طراحی نمونه، به عناصر و نمونه های پیچیده XSD افزوده شده برای اینکه ارتباطی با عناصر نمونه در سرویس وب با مدل‌های سمانتیک بیاید چنین مواردی برای چندین مورد در یکی از موارد است.

؟ دو عنصر جدید اصطلاحاً پیش پردازش و افکت، که به عنوان عناصر اولیه و عناصر عملکرد هستند. پیش فرض ها و افکت ها اساساً در یافتن سرویس ها به کار می روند و لزوماً برای ارائه یک سرویس مورد نیاز نیستند (در این اطلاعات فنی به ارائه جزئیات

پیش فرض ها و افکتهایی پرداخته ایم که شامل ترکیبی از موارد پیچیده تا مدل ها و اطلاعات موجود سمانتیک می باشند.

؟ عمل گسترده عنصر رابط پردازشگر که در اصطلاح طبقه خوانده می شود شامل اطلاعات طبقه بندی شده ای می باشد که می توان آن را در زمان ارائه یک سرویس در سرویس های وب مانند UDDI به ثبت رساند. طبقه بندی سمانتیک از UDDI از آنتولوژیهای پیشنهادی در [NWSDI , SUSO4] استفاده می کند.

تفسیر عناصر ورودی و خروجی

در این قسمت به توصیف چگونگی تفسیر عناصر خروجی و ورودی در سند WSDL می پردازیم در نمونه سفارش خرید با عملیات Processpurchase order یک مورد داده و یک مورد خروجی داشتیم. مورد ورودی (داده) با عنصر Process purchase order Request می باشد که بوسیله نوع پیچیده processpurchase order Request داده شده است. مورد خروجی با عنصر process purchase order Response است. عملیات WSDL در زیر داده شده است.

در این نمونه، داده (ورودی) نوع پیچیده ای است در حالیکه خروجی یک نوع ساده است برای تفسیر انواع ساده از مورد گسترده عنصر xsd استفاده می کنیم. یک نمونه از طرح xml برای عنصر xsd در زیر نشان داده شده است این مورد نشان می دهد که هر

عنصری می تواند "بوسیله عملکرد در هر فضای بدون طرحی" گسترش یابد ما از فضای wssem با عملکرد مدل مرجع برای تفاسیر مربوط به عنصر استفاده کردیم.

تفسیر انواع پیچیده

انواع پیچیده را می توان با روشهای پیچیده تفسیر کرد. ما دو طرح جداگانه برای تفسیر این انواع پیچیده پیشنهاد کردیم.

؟ تفسیر سطح پائین - تفسیر سطح عنصر Leaf (گره ساختار درختی)

؟ تفسیر سطح بالا: تفسیر در سطح نوع پیچیده

در تفسیر سطح پائین. تمام عناصر Leaf در نوع پیچیده بررسی می شوند. مزیت این روش ساده بودن آن است. این فرضیه وجود دارد که مقایسه مفهومی در مدل برجسته وجود دارد که طراحی هر عنصر Leaf را انجام می دهد. در این مورد جایی که هیچ مفهوم قیاسی وجود نداشته باشد تفسیر سمتیک به صورت نامشخصی باقی می ماند. عیب این مرحله بر این فرض است که مقایسه یک - به یک بین عناصر طرح و مفاهیم مدل نمونه وجود دارند زمانیکه ارتباطات یک - به - همه و همه به یک وجود دارد مشخص می شود که ارتباط در هر عنصر Leaf ممکن است وجود نداشته باشد. در تفسیر سطح بالا انواع پیچیده خودشان با مفهوم سمتیک تفسیر می شوند. مزیت این روش این است که اجازه می دهد موارد خاص طراحی پیچیده بین عناصر موجود در

انواع پیچیده و مفاهیم مشخص در مدل های مشخص به وجود آید عیب این روش است که پیچیدگی آن محرز است. در این اطلاعات فنی از هر دو روش در تفاسیر استفاده می کنیم. در زیر به توصیف و مکانیسم هر دو مورد می پردازیم.

تفسیر سطح پائین: تفسیر عناصر Leaf در نوع پیچیده

در مواردی، عناصر نوع پیچیده به مقایسه روش یک به یک با مفاهیم در مدل برجسته می پردازند. برای جاسازی این مورد روش مستقیم و ساده ای وجود دارد این مورد را با اضافه کردن wssem نگهداری می کنیم: مدل مرجع با عنصر طرح مربوطه با تعریف عملکرد مشارکت دارد. تفاسیر پیچیده ای با در ارتباط با این عنصر انجام می دهیم. نمونه برای ارتباط یک مدل مرجع و مشارکت زیر به صورت زیر است:

تفسیر سطح بالا: تفسیر نوع پیچیده با استفاده از نمونه طراحی کار نوع پیچیده می تواند از طریق wssem دارای یک تفسیر سمتیک باشد: مشارکت مدل مرجع این مشارکت را می توان در مقایسه با سطح بالای موارد ارائه شده قرار داد. این سطح بالای تفسیر سمتیک در نوع پیچیده آن را می توان در طول کشف تشخیص مقدماتی این دو ساختار به دست آورد که با حالت سمتیک مربوطه مطابقت دارد به عنوان مثال یک نوع پیچیده chip در نمونه xsd دارای تفسیر سمتیکی در ریزپردازنده های چیپ است. این مورد برای عنصر نمونه در طول کشف سرویس ها به کار می رود. به عنوان مثال کشف

سرویس ممکن است به اشتباه chip را به معنی تراشه ای در موارد مشخص بگیرد اما دارای تفسیر سمتیکی باشد که کمکی برای مشخص کردن متن است.

تفسیر ریزپردازنده تراشه را می توان به عنوان قسمتی از موارد الکترونیکی مشخص یافت و ویژگیهای مربوط در آن مانند ویژگیهای الکتریکی در ریزپردازنده کمکی به مجموعه محتوا برای درک آن می کند. در این مورد اطلاعات فنی wseem یا مشارکت مدل مرجع در نوع پیچیده آن تفاسیر سمتیک خاص را در عناصر نوع پیچیده مشخص نمی کند تفاسیر سمتیک برای عناصر شامل نوع پیچیده است که از نوع wseem می باشد مشارکت طراحی نمونه را دارد. نمونه برای ارتباط یک مدل مرجع به صورت زیر است.

نمونه ای از طرح xml برای نوع پیچیده در زیر نشان داده شده است. این نشان می دهد که انواع پیچیده توانایی توسعه یافتن را مشارکت موارد بدون طرح را دارند. با استفاده از wseem در Names pace با schema Mapping تفاسیری در انواع پیچیده انجام می گیرد.

درست مانند این روش، انتخاب زبان مدلسازی دمین را درک می کنیم، همچنین انتخاب زبان طراحی الگو را درک می کنیم.



روش ما مکانیزمی را برای آمیختن صفر با تابع های طراحی الگو با نوع پیچیده را بدون هیچ محدودیت انتخاب زبان طراحی الگو فراهم می کند. موارد گلچین شده زیر از سفارش خرید نحوه استفاده از XSLT را به عنوان زبان طراحی الگو برای مشخص کردن روابط میان عناصر XSD و مفاهیم OWL نشان می دهد.

نمونه های تفصیلی طراحی را با استفاده از XSLT و XQuery نشان می دهد که به ترتیب در ضمیمه های B,A آمده اند.

برای طراحی موضوعات از XML به واسطه مفاهیم OWL ما هر دو پیشنهادات زیر را پیشنهاد می کنیم:

۱- انواع پیچیده XML در کلاس های OWL طراحی می شوند.

۲- عناصر xml در خصوصیات OWL طراحی می شوند.

در POBilling.xsd، نوع پیچیده POBilling (Complestype) مستقیماً به مفهوم OWL

Billing (ضمیمه) مربوط هستند. از این رو مشخصه Model Reference برای مربوط

کردن آن به مفهوم OWL Billing به کار می رود. دستور b ص ۱۵

POAddress. xsd نوع پیچیده POAddress به طور مستقیم در مفهوم OWL موجود در

آنتولوژی طراحی نمی شود.

از این رو ویژگی طراحی الگو برای اشاره بر تبدیل یک XSLT به کار می رود.

طراحی الگو مورد استفاده توسط POAddress. Xsd, POAddress. Xsl در زیر نشان داده شده است.

POAddress. xsl

استفاده از Wssem: modelReference در سطح برگ و Wssem: schemaMapping

در سطح نوع پیچیده با نوع پیچیده الگوی XML بسیار منطبق است. به طور ذاتی اگر

یک طراحی الگو برای نوع پیچیده مشابه ارائه شود، مرجع مدل منفرد معین شده در

عناصر برگ (گره آخر در ساختار درختی Leaf) را باطل می کند.

ما روش خود را برای تفسیر انواع پیچیده (Complex Types) در زیر خلاصه می کنیم.

۱- یک تایپ پیچیده می تواند دارای تفسیر سمانتیک انتخابی از طریق Wssem:

modelReference باشد. منظور اولیه این روش توانایی تضمین مربع و کامل در طول

کشف بدون مراجعه به جزئیات ساختار است.

۲- عناصر درون تایپ پیچیده می تواند دارای تفاسیر سمانتیک معین شده به یک یا دو

روش باشد.

a- از طریق Wssem: modelReference در هر عنصر گره ساختار درختیمحتوای تایپ

پیچیده و یا b- از طریق: مشخصه Wssem: SchemMapping در نوع پیچیده که

طراحی های پیچیده را بین عناصر دارای نوع پیچیده و مفاهیم موجود در مدل یا آنولوژی دمین خارجی مشخص می کند.

ما قواعد زیر را برای حل هر تعارض بالقوه پیشنهاد می کنیم.

۱- چنانچه Wssem: SchemaMapping برای نوع پیچیده تعیین شود و از Wssem:

modelReference در هر عنصر گره ساختار درختی (Leaf) در نوع پیچیده به کار رود،

سپس SchemaMapping، مدل ریفرنس های (model/References) سطح

عنصر Leaf را باطل می کند.

۲- تفسیر معنایی (سمتیک) ارائه شده در نوع پیچیده با استفاده از Wssem:

modelReference خود تنها در نوع پیچیده به کار می رود. درباره تفسیر معنایی

(سمتیک) برای عناصر درون نوع پیچیده دخالتی صورت نمی گیرد، حتی اگر مفهوم

سمتیک در نوع پیچیده معین شود و دارای مفاهیم ظاهراً مربوط به عناصر موجود در

نوع پیچیده باشد. برای شرح تفاسیر برای عناصر داخل یک نوع پیچیده یا از Wssem:

modelReference در عناصر Leaf یک نوع پیچیده استفاده می شود یا اینکه از

Wssem: Schema Mapping در شرح تفسیر معنایی برای عناصر چندگانه استفاده

می شود.

۳- چنانچه یک نوع پیچیده (Complex type) داخل یک Complex type باشد، پس می توان از ترکیب Wssem: modelReference, Wssem: Schema Mapping استفاده کرد.

به عنوان مثال POBilling.xcd دارای دو عنصر پیچیده Ship to Address و bill to Address می باشد که هر یک از نوع POAddress و عنصر accountID از نوع سری است. در هر حالت، عنصر account ID می تواند دارای یک Wssem: modelReference باشد تا مستقیماً به مفهوم سمتیک در آنتولوژی اشاره کند در حالی که تفاسیر معنایی (سمتیک) برای bill to Address, Ship to Address را می توان یک مرتبه در سطح نوع پیچیده POAddress معین کرد.

در این بخش، مکانیزمی را برای یکپارچه کردن تفاسیر با عناصر Wsdl: Output, Wsdl: input در سند WSDL ارائه کردیم. در بخش بعد، نحوه بسط یافتن WSDL: Operation را برای تطبیق دادن و جای دهی داده پیش از پردازش و تأثیرات شرح می دهیم.

#### ۴-۲- پیش شرط (Perconditions)

پیش شرط مجموعه ای از قاعده ها را تعیین می کند که قبل از تقاضای عملیات سرویس وب باید بازبینی شود. آنها می توانند ملزوماتی را که باید بررسی شوند مشخص کنند،

نظیر: برای این کمپانی باید دارای حساب موجودی باشد یا محدودیت ها نظیر: "تنها مشتریان ایالات متحده می توانند سرویس دهی شوند". پیش شرط به عنوان عناصر بچه عملیات معین می شدند زیرا پیش شرط تعیین شده است.

الگوی جای دهی داده پیش از پردازش در زیر نشان داده شده است.

پیش شرط در زیر تعریف شده است:

`/Precondition`

این عنصر، تفسیر معنایی را برای عملیات والدین معین می کند.

`/Precondition/@name`

مشخصه نام، یک معین کننده منحصر به فرد را در مجموعه ای از پیش شرط های

موجود در سند WSDL شرح می دهد.

`/Precondition/@mode/ Reference`

مشخصه `URI, mode/ Reference` بخشی از مدل سمتیگی را شرح می دهد که خود

پیش شرط را توضیح می دهد. مشخصه `mode/ Reference` و مشخصه عبارت متقابلاً

انحصاری هستند.

`/Precondition/@expression`

این عبارتی است که پیش شرط را تعریف می کند. فرمت عبارت توسط زبان بازنمایی سمتیک معین می شود و از آن برای بیان مدل سمتیک به کار می رود. مشخصه /mode

Reference و مشخصه عبارت متقابلاً انحصاری هستند. دستور (G) ص ۱۸

پیش شرط ها و تأثیرات به عنوان عناصر بچه یک عنصر عملیاتی معین می شدند. هر عمل ممکن است حداکثر یک پیش شرط و حتی الامکان تعدادی تأثیرات داشته باشد. ما حداکثر برای یک پیش شرط مجاز هستیم تا سادگی تعریف را حفظ کنیم. ما بر این باوریم که پیش شرط های پیچیده یا شرطی باید در مدل دمین یا سمتیک بیان شوند. به عنوان مثال، در OWL مجموعه ای از پیش شرط ها (تأثیرات) را می توان از طریق عبارات منطقی تعریف کرد تا عملیات هایی نظیر 'and', 'or', 'xor' و غیره در ارزیابی این عبارات را تسهیل کرد.

در این یادداشت فنی، تصور می کنیم که زبان برجسته بازنمایی (سمتیک) دستیابی به چنین پیش شرط های چندگانه در داخل یک پیش شرط منفرد و سطح بالا را پشتیبانی می کند که می تواند در سطح عنصر عملیات در حین تعیین WSDL ارجاع داده شود.

توسعه زبان های markup برای نمایش دادن پیش شرط ها و تأثیرات در سطح تحقیق فعالانه می باشد. انجمن های مختلف در حال کار بر روی این زبان هستند. به عنوان مثال، انجمن وب سمتیک در حال کار بر روی زبان (SWRL)[SWRL] است و انجمن

مدل سازی در حال کار بر روی زبان Object Constrain (OCL) [OSL] و غیره است. در حالی که ما اهمیت پیش شرط ها و تأثیرات را در این سند تشخیص می دهیم و برای تطبیق آنها در حین شرح سرویس های وب قلاب هایی را فراهم می کنیم، بر این باوریم که استفاده از آنها در سیستم های تولیدی هنوز خیلی زود است. تحقیقات و نمونه های عینی بیشتری برای کمک به درک بهتر تطبیق و ارزیابی پیش شرط ها و تأثیرات در حین تطبیق دهی اتوماتیک تعریف های سرویس های وب مورد نیاز است. ما چنین مبحثی را تا ورژن بعدی این مقاله به تعویق می اندازیم.

الگو برای عملیات تقاضا و پاسخ به واسطه پیش شرط های تأثیرات اختیاری در زیر نشان داده شده است:

دستور عملیات تقاضا و پاسخ به واسطه پیش شرط ها و تأثیرات اختیاری در زیر نشان داده شده است:

#### ۳-۴ تأثیرات (Effects)

یک تأثیر نتیجه تقاضای یک عملیات را مشخص می کند.

آن می توان به آسانی اظهار کند که خروجی بازگردانده شده یا می تواند گزارشاتی را درباره نوع تغییرات در وضعیتی که انتظار می رود به محض درخواست سرویس رخ

دهد ارائه دهد. به عنوان مثال، "یک تراز حساب جدید در دسترس خواهد بود" یا "حساب کارت اعتباری بدهی خواهد داشت". دستور (J) ص ۱۹ و ۲۰

عنصر تأثیر بچه عنصر عملیات یا عنصر خطا است. عنصر تأثیر مطابق ذیل تعریف می شود:

/effect

این عنصر، تفسیر سمتیک تأثیری را شرح می دهد که برای عملیات والدین به کار می رود.

/Precondition/@name

مشخصه این نام (name) یک تعیین کننده منحصر به فرد را در داخل مجموعه ای از تأثیرات در سند WSDL شرح می دهد.

/effect/@mode/ Reference

مشخصه URL mode/ Reference بخشی از مدل سمتیکی را شرح می دهد که تأثیر را مشخص می کند. مشخصه mode/ Reference و مشخصه عبارت (expression) متقابلاً

انحصاری هستند.

effect/@ expression

این عبارتی است که تأثیر را معین می کند.



فرمت عبارت توسط زبان بازنمایی سمتیک به کار رفته برای بیان مدل سمتیک، تعیین می شود. مشخصه mode/ Reference و مشخصه عبارت متقابلاً انحصاری هستند.

یک تأثیر می تواند با خط در رابطه باشد. برای مثال، تأثیر "کارت اعتباری پردازش نمی شود" زمانی که یک خطای 'Invalid credit card Error' رخ می دهد ممکن است واقع شود. تأثیر خطا به عنصر خطای بچه عملیات ضمیمه می شود که این کار از طریق مشخصه mode/ Reference انجام می شود.

تذکر: لطفاً به تذکر موجود در بخش (۳-۳) پیش شرط مراجعه شود تا نحوه استفاده از عنصر تفسیر در عملیات مشاهده شود و اگر شما از یک ورژن سفارش الگوی WSDL در جایی استفاده کنید عملیات عناصر توسعه پذیر را پشتیبانی نمی کنند، می توانید تأثیرات را به دست آورید.

#### ۴-۴ طبقه بندی سرویس Servis Categorization

هدف از سرویس های تفسیری، قادر ساختن جستجوی پویای سرویس ها است. این امر زمانی امکان پذیر است که سرویس ها به واسطه سمتیک ها انتشار، فهرست بندی و تفسیر می شوند. در این یادداشت فنی، تا اینجا ما بر روی نحوه تفسیر سرویس ها تمرکز کرده ایم. اکنون مکانیزمی را برای اضافه کردن اطلاعات طبقه بندی به سرویس های

ارائه می دهیم که می توانند در حین انتشار سرویس هادر رجیسترهایی نظیر UDDI مورد استفاده قرار گیرند.

این کار با کاهش دامنه سرویس های داوطلب به جستجوی سرویس کمک می کند. از طبقه بندی می توان به عنوان ورودی زمانی که سرویس در رجیستری UDDI منتشر می شود به کار برد یا آن می تواند طبقه بندی مؤثری را زمانی که سرویس توسط زبان جستجوی سرویس وب [WSIL] یا سایر ابزار راه حل خاص در دسترس باشد، ایجاد کند.

طبقه بندی سرویس نیز به منظور پشتیبانی طبقه بندی های مخصوص میان افزار یا سرویس های کاربردی نظیر سرویس های واسطه است.

در این یادداشت فنی هدف ما اطمینان از این است که اطلاعات طبقه بندی اساسی و سطح بالایی درباره سرویس وجود دارد و جزئیاتی را درباره سیستم طبقه بندی واقعی و نگهداری دسته بندی، طبقه بندی و غیره ارائه می دهد تا رجیستری های سرویس را برجسته کند. همان طور که ذکر شده، مفهوم اطلاعات طبقه بندی سرویس ارتباطاتی از OWL-S عاریه گرفته شده است، اما در اینجا آن برای عمل در داخل پرامترهای تعریف WSDL تطبیق یافته است.

ما با استفاده از عناصر توسعه پذیر در عنصر واسط WSDL یک دسته سرویس را مدل سازی کردیم. اینطور فرض می شود که کل عملیات در رابطه یک سند معین WSDL به دسته ای مشابه تعلق دارند.

[www.kandoo.cn.com](http://www.kandoo.cn.com)

[www.kandoo.cn.com](http://www.kandoo.cn.com)

[www.kandoo.cn.com](http://www.kandoo.cn.com)

[www.kandoo.cn.com](http://www.kandoo.cn.com)