Archive of SID

علوم و مهندسی کامپیوتر
نشریه علمی پژوهشی انجمن کامپیوتر ایران
مجلد ۱۰، شماره ۱ و ۳ (الف)، ۱۳۹۱
صفحات ۵۹-۶۷
مقاله کوتاه

ارزیابی معیارهای کارآیی در مدیریت پردازش شبکه‌ای بوسیله اشیای سیار

سیدمحمدرضا میرزابابائی محمدکاظم اکبری محمود صادقی

دانشکده مهندسی کامپیوتر و فناوری اطلاعات، دانشگاه صنعتی امیرکبیر، تهران، ایران

چکیده

رویکرد تمام کاربردهای مدیریت شبکه به سمت استفاده از عامل‌های هوشمند سیار می‌باشد تا بتوانند معماری‌های سنتی را به سمت محیط پردازش فراگیر ارتقا دهند. خصوصیاتی نظیر مقیاس‌پذیری، قابلیت اطمینان، قابلیت دسترسی، امنیت، کارآیی و ... نقش مهمی در مدیریت شبکه خصوصا زمانبندی پردازش شبکه‌ای دارند. مدیریت بسیاری از این خصوصیات را می‌توان با استفاده از عامل‌های سیار بهینه نمود. با توجه به فراگیری سیستم‌های مدیریت شبکه مبتنی بر دلال‌های درخواست شیء (ORB) و نیز تجاری بودن این میان‌افزارها مانند CORBA، می‌توان از آنها برای بدست آوردن معیارهای کارآیی و سپس زمانبندی با استفاده از آن اقدام نمود. در این پروژه نمونه بسیار کوچکی از قابلیت‌های (قابلیت تحرک براساس استاندارد MASIF) عامل‌های سیار هوشمند به این سیستم اضافه شده است که امکان استفاده از سرویس‌دهنده‌های نامگذاری محلی را فراهم می‌نماید. نتایج مشاهده شده، حاکی از بهبود کارآیی سیستم مدیریت زمانبندی شبکه پیاده‌سازی شده می‌باشد. منظور از کارآیی در این تحقیق، مدت زمان پاسخ و ترافیک عبوری در شبکه است.

کلمات کلیدی: محیط فراگیر، کوربا CORBA، مدیریت شبکه توزیع شده، عامل‌های سیار، ارزیابی کارآیی.

۱- مقدمه

اساساً مدیریت و زمانبندی پردازش در شبکه برای تحقق اهداف مدیریتی نیازمند نظارت و کنترل تجهیزات متصل به شبکه با استفاده از جمع‌آوری و تحلیل اطلاعات مختلف مدیریتی از تجهیزات شبکه می‌باشد [۱]. این ویژگی این امکان را برای مدیران شبکه فراهم می‌سازد تا بتوانند از یک نقطه‌ی شبکه، کل شبکه را تحت مدیریت خود داشته باشند. بسیاری از سیستم‌های مدیریتی از مدل مدیریت ایستا و متمرکز پیروی می‌کنند که موجب انتقال حجم عظیمی از اطلاعات خام از سیستم‌های راه دور به ایستگاه مرکزی مدیریت شبکه می‌گردد قبل از آنکه پردازش، خلاصه‌سازی و ساده‌سازی بر روی آنها انجام گیرد. در نتیجه این سیستمها، موجب ازدحام در شبکه می‌شوند که منجر به مشکلاتی نظیر تاخیر پاسخ‌ها، کاهش کارآیی، عدم واکنش سریع و به موقع در هنگام رخداد آسیب‌های شبکه‌ای و همچنین پهنای باند بسیار پایین می‌گردد.

بنابراین، با روند رشد نمایی که در اندازه، گستردگی و توزیع، پیچیدگی و نیز گوناگونی شبکه‌های ارتباطی و همچنین منابع و تجهیزات شبکه‌ای وجود دارد، اینگونه سیستم‌های مدیریتی محدودیت‌های جدی را از نظر کارآیی، مقیاس‌پذیری و نیز انعطاف‌پذیری دارا می‌باشند. برای مدیریت ابزارها و خدمات شبکه‌ای در چنین محیط‌های با پیچیدگی بالا، سازمان‌ها و موسسات مختلفی سکوهای مدیریتی گوناگونی (نظیر Internet Management Framework، OSI Management Framework، TMN [۲]) را با استفاده از انواع مختلف پروتکل‌های ارتباطی مدیریتی (همچون SNMP [۳]، CMIP [۴]) توسعه داده و آماده ساخته‌اند.

در حال حاضر، بسیاری از شبکه‌های داده توسط SNMP و بسیاری از شبکه‌های مخابراتی توسط CMIP مدیریت می‌شوند که هر دو از انواع پیاده‌سازی‌های مشتری/خدمتگزار می‌باشند. علاوه بر این دو روش فوق، CORBA [۵] که یک معماری توسعه‌یافته توسط OMG می‌باشد، چارچوبی را برای تعاملات بین اشیا در محیط‌های توزیع‌شده فراهم می‌سازد. این چارچوب نیز راه حل همه منظوره‌ای که مبتنی بر مدل مشتری/خدمتگزار می‌باشد را برای توسعه برنامه‌های کاربردی مدیریت شبکه فراهم می‌سازد که محدودیت خاصی را برای پروتکل ارتباطی تحمیل نمی‌کند. سیستم‌های مدیریت شبکه متمرکز[1] به نوبه خود

www.SID.ir

Archive of SID

سرویس‌گیرنده‌های عامل‌های مدیریتی هستند که در هریک از عناصر مدیریت‌شونده شبکه[۲]، مقیم می‌باشند. با این وجود و با مطالعه نمونه‌های بسیاری از برنامه‌های کاربردی مدیریت شبکه که عملا مورد استفاده قرار گرفته است، نقایص و محدودیت‌های مدل مشتری/خدمتگزار ـ به طور مثال، پردازش حجیم و محدودیت ترافیک در سیستم های مدیریت شبکه – سال‌هاست که مشخص و تعیین شده است.

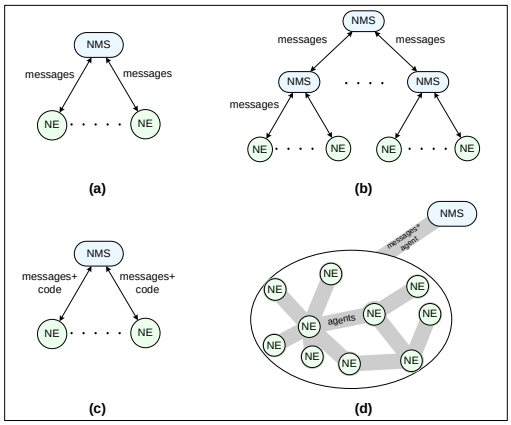
توزیع نمودن کارکردهای مدیریت شبکه ممکن است منافعی را به دنبال داشته باشد که از آن جمله می‌توان به موارد زیر اشاره نمود: تقلیل عمده حجم پردازش و ترافیک را در مدیریت شبکه به گونه‌ای که مقداری از پردازش بر عهده عناصر شبکه قرار بگیرد؛ افزایش قابلیت مقیاس‌پذیری در شبکه‌های بزرگ؛ استخراج اطلاعات در مکانهایی نزدیک به منابع داده جهت افزایش سرعت و کارآیی؛ همچنین پایداری بالاتر به دلیل مدیریت توزیع‌شده شبکه عدم نیاز به اتصال دائمی و مداوم سیستم مدیریت شبکه به عناصر شبکه. لذا با قطع ارتباط نیز می‌توان برخی از کارکردهای مدیریتی را پیاده‌سازی نمود.

چارچوب کوربا بر مدل مشتری/خدمتگزار بنا شده است به طوری که در روش‌های سنتی کاربردهای مدیریت شبکه، اشیا در عناصر شبکه به صورت دائمی مقیم هستند. کاربری و نیز مفید بودن استفاده از فناوری‌های عامل‌های سیار[۳] در مدیریت سیستم‌ها و شبکه‌های توزیع‌شده چندین سال است که به اثبات رسیده است [۶]. آنچه که در این مقاله به آن پرداخته شده است، افزودن ویژگی سیار بودن عامل‌های سیار، به اشیای ایستای کوربا در محیط‌هایی است که این چارچوب به عنوان زیرساخت مدیریتی استفاده می‌گردد، تا بتوان به کارآیی بهتری دست یافت. برای این منظور، یک محیط آزمایشی بر اساس چارچوب ارتباطی MAF برای اجرای اشیای سیار آماده گردید و اندازه‌گیری‌های لازم صورت پذیرفت. ادامه مقاله از بخش‌های زیر تشکیل یافته است: بخش ۲ یک مرور کلی است بر مدیریت توزیعی شبکه، چارچوب کوربا و قابلیت سیار بودن اشیا. بخش ۳ نمونه پیاده‌سازی شده را توضیح می‌دهد. بخش ۴ شبیه‌سازی و نتایج حاصله را بیان می‌دارد. و در نهایت نکات نتیجه‌گیری ارائه شده است.

۲- پیش‌زمینه

۲-۱- مروری بر مدیریت توزیعی شبکه

مدیریت متمرکز شبکه، مدیریتی با کارآیی بسیار پایین می‌باشد چراکه تمامی اعمال مدیریتی به یک مدیریت یکپارچه وابسته است. به وضوح می‌توان دریافت که توزیع کارکردهای مدیریت شبکه به سلسله مراتبی از مدیران میانی که مسئول مدیریت بخشی از شبکه می‌باشند، باعث افزایش کارآیی می‌گردد [۷].



شکل ۱- شیوه‌های مختلف مدیریت شبکه

شیوه‌های مدیریت شبکه که در اینجا مورد بحث قرار می‌گیرد به چهار دسته تقسیم می‌گردد (مطابق شکل ۱):

۱) **مدل مشتری/خدمتگزار (CS)**، که همان شیوه سنتی مدیریت شبکه می‌باشد که در آن برای جمع‌آوری اطلاعات و کنترل تجهیزات، یک شی سرویس‌دهنده مقیم در عنصر مدیریت شونده تحت عنوان شی مدیریت‌شونده قرار دارد و به درخواست‌های مدیر پاسخ می‌دهد (شکل ۱-a).

۲) **مدل مدیریت توزیعی شبکه – سلسله مراتبی ایستا (DNM-HS)**، که همانگونه که در شکل نیز مشاهده می‌شود، شامل سلسله‌مراتبی از مدیران است که زیرشبکه‌های خود را مدیریت می‌کنند و خود آنها نیز توسط مدیران سطح بالاتر مدیریت می‌گردند. هر کدام از زیرشبکه‌ها براساس مدل مشتری/خدمتگزار مدیریت می‌گردند (شکل ۱-b).

۳) **مدل مدیریت توزیعی شبکه– سیار ضعیف (DNM-WM)**، که از روش توزیع کد و انتقال آن به هر عنصر شبکه و اجرای آن توسط عنصر استفاده می‌کند. در این روش کد مدیریتی از مدیر به عنصر شبکه انتقال پیدا کرده و توسط وی اجرا می‌گردد و نتیجه به مدیر برگردانده می‌شود. نهایتا کد نیز در عنصر شبکه منقضی می‌شود. در این روش کد انتقال داده شده قابلیت مهاجرت به عناصر دیگر شبکه را ندارد (شکل ۱-c).

۴) **مدل مدیریت توزیعی شبکه – سیار قوی (DNM-SM)**، که با استفاده از گسیل تعدادی عامل سیار توسط سیستم مدیریت به سطح شبکه، وظیفه مدیریتی خود را به انجام می‌رساند. در این روش وظیفه مدیریتی با همکاری تعدادی عامل سیار انجام می‌گیرد به این ترتیب که آنها با سفر به عناصر مختلف شبکه براساس برنامه سفر از پیش تعیین شده یا مسیریابی پویا این عملیات را کامل نموده و وظیفه مدیریت را به انجام می‌رسانند (شکل ۱-d).

در کل، معیارهای کارآیی عبارتند از: مدت زمان تکمیل یک کار و همچنین بهره‌وری پهنای باند شبکه. در این مقاله نیز همین معیارها برای ارزیابی کارآیی مورد اندازه‌گیری قرار گرفته است.

۲-۲- CORBA و قابلیت سیار بودن

چارچوب کاری کوربا این امکان را برای برنامه‌نویسان سیستم‌های توزیعی فراهم می‌نماید تا آنها بتوانند بدون در نظر گرفتن زبان برنامه‌سازی و همچنین بدون درگیر شدن با لایه‌های پایین شبکه و نیز تنوع سیستم عامل، به فراخوانی متدهای اشیای چارچوب به شیوه‌ای شی‌گرا بپردازند. هسته این چارچوب یک دلال درخواست است که به عنوان یک میان‌افزار توزیع اطلاعات، ارتباط و همکاری بین اشیا را به صورتی یکپارچه فراهم می‌سازد. ارتباطات اشیای کوربا از مدل مشتری/خدمتگزار پیروی می‌کند که مبتنی بر مکانیزم درخواست-پاسخ[۴] می‌باشد. برای آنکه یک عملیات مشخص صورت گیرد، سرویس‌گیرنده باید یک متد مشخص شی راه دور را فراخوانی نموده و ارتباط خود را با سرویس‌دهنده حفظ نماید تا انتقال اطلاعات صورت گیرد.

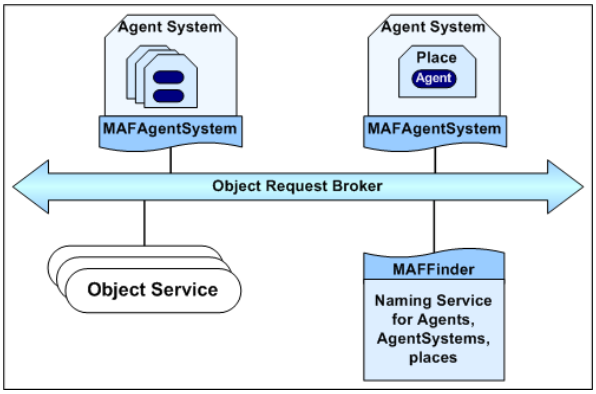
چارچوب کوربا مستقل از سکو است و لذا می‌تواند انواع مختلف شبکه‌ها، ماشین‌ها و سیستم‌عامل‌ها را پوشش دهد و این خود منجر به تولید سیستم‌ها و برنامه‌های کاربردی مبتنی بر اشیای توزیع‌شده با امنیت بالاتر و همچنین قابلیت استفاده دوباره نرم‌افزار می‌گردد. در چارچوب کوربا می‌توان سرویس‌های مختلفی را یافت که برای پیاده‌سازی سیستم‌های توزیعی موردنیاز است؛ سرویس‌هایی برای محلی‌سازی نظیر: سرویس نامگذاری و Trader و ارتباطات رخدادگرا نظیر سرویس رخداد از آن جمله‌اند. در این معماری پایداری، تراکنش‌ها و امنیت نیز تبیین شده است. تمامی این مباحث مورد علاقه سیستم‌های مبتنی بر عامل‌های سیار نیز

www.SID.ir

Archive of SID

می‌باشد، و بدون آنکه آنها را درگیر زبان‌های برنامه‌نویسی بکند (چرا که ارتباطات اشیا توسط زبان تعریف واسط یا همان IDL به زبان خاصی نگاشت می‌گردد)، در محیط‌های گسترده و بر اساس استانداردهای فراگیر قابل اجرا می‌سازد. پیاده‌سازی‌های انجام شده در کوربا، سیار بودن اشیا را پشتیبانی نمی‌کنند ولی می‌توانند در سیستم‌های مبتنی بر عامل‌های سیار برای موارد خاصی استفاده شوند نظیر: محیط‌های اجرایی میزبان عامل‌ها، زیرساخت ارتباطی عامل‌ها، سرویس فهرست‌گیری و ... .

در این مقاله، ما از ویژگی‌ها و خصوصیات گفته شده در MASIF [۸] برای افزودن قابلیت سیار بودن عامل‌ها بهره گرفته‌ایم. این ویژگی‌ها که توسط OMG ارائه شده است، برای ایجاد قابلیت تعامل میان سکوهای عامل‌ها و خود عامل‌ها می‌باشد. چارچوب MASIF بر اساس مفاهیم زیر می‌باشد: مالکیت یک عامل، شخص یا سازمانی را که آن عامل از طرف وی عمل می‌کند، مشخص می‌کند و این مالکیت در صورت حرکت عامل باید مورد تایید قرار بگیرد؛ هنگامی که عاملی خود را انتقال می‌دهد، این عامل، بین محیط‌های اجرایی‌ای که مکان‌ها خوانده می‌شوند، سفر می‌کند. مکان، زمینه‌ای در سیستمِ عاملی می‌باشد که عامل در آن اجرا می‌شود. این زمینه می‌تواند توابعی مانند کنترل دسترسی، را فراهم نماید. مکان مبدا و مقصد، می‌تواند در یک سیستمِ عامل یا در دو سیستمِ عاملی که شناسنامه عامل یکسانی را پشتیبانی می‌کنند، وجود داشته باشد. عامل‌های سیار این توانایی را دارند که می‌توانند از یک مکان به مکانی دیگر، بین سیستم‌های عامل حرکت نمایند، و این در صورتی است که، نوع سیستم عامل آنها توسط سیستم عامل مقصد قابل شناسایی باشد. سیستم‌های عامل می‌توانند به یک مالک خاص محدود شوند و در این صورت در یک ناحیه قرار می‌گیرند. بر این اساس، عامل‌ها می‌توانند یک نام منحصر به فرد عمومی داشته باشند که از سه‌تایی (مالک، شناسه عامل، نوع سیستم عاملی) تشکیل می‌گردد. سیستمِ عامل خود سکویی است که می‌تواند عامل را ایجاد، اجرا، منتقل و ختم کند. سیستمِ عامل نیز، مانند عامل به مالکی که فرد یا سازمانی که برای آن کار می‌کند را مشخص می‌کند منصوب می‌شود. در اینجا سیستمِ عامل با نام و آدرسش بصورت یکتا مشخص می‌شود. ممکن است در یک میزبان، یک یا چند سیستمِ عامل داشته باشیم. نوع سیستمِ عامل، شناسنامه عامل را توضیح می‌دهد. برای مثال، اگر نوع سیستمِ عاملی Aglet باشد، سیستمِ عاملی، توسط IBM پیاده‌سازی شده، زبان Java را بعنوان زبان عامل پشتیبانی می‌کند، از Itinerary برای حرکت استفاده می‌کند و برای سریال‌سازی شی جاوا را بکار می‌برد.



شکل ۲- مدل مفهومی MASIF

این چارچوب توسط دو واسط کوربا مدیریت می‌گردد؛ واسط MAFAgentSystem باید توسط سیستم‌های عامل پیاده‌سازی شود تا بتوان عامل‌ها را مدیریت کرد (ایجاد، تعلیق، ازسرگیری و ختم) یا عامل‌های سیار مهاجر را دریافت نمود؛ همچنین کلاس‌های آنها را انتقال داد. واسط MAFFinder نیز برای ثبت‌نام و بازیابی عامل‌ها، مکان‌ها و سیستم‌های عامل اختصاص یافته است.

۳- پیاده‌سازی

براساس مطالب بیان شده در بخش‌های قبلی، در این پروژه، قابلیت حرکت عامل‌های سیار به اشیا به چارچوب کوربا اضافه شده و شی سیار کوربا (MCO) با استفاده از MASIF پیاده‌سازی شده است.

۳-۱- آژانس‌ها

آژانس، یک محیط اجرایی برای عامل‌های سیار فراهم می‌کند که در واژگان MASIF تحت عنوان رابط سیستم عامل شناخته می‌شود. هر آژانس به یک ناحیه تعلق دارد و نام آن در آن ناحیه منحصربه‌فرد است؛ همچنین این آژانس متعلق به یک مالک[۵] می‌باشد. در این سیستم، هر آژانس عبارت است از یک شی کوربا در نقش سرویس‌دهنده، که واسط MAFAgentSystem را پیاده‌سازی کرده است. عامل‌ها را می‌توان از طریق واسط MAFAgentSystem و نیز متدهای کلاس Agency مدیریت نمود. عملیاتی نظیر: ایجاد و خاتمه یک عامل، تعلیق و ازسرگیری فعالیت یک عامل، انتقال عامل، لیست‌گیری اسامی عامل‌های مقیم در یک رابط سیستم عامل، و دریافت اطلاعات عامل‌های محلی بر عهده یک آژانس می‌باشد.

۳-۲- اشیا (عامل‌های) سیار

آژانس‌ها دارای توابعی برای ایجاد و مدیریت اشیای کوربا هستند. این اشیا، واسط MobileObject را که به صورت واسط کوربا یا IDL نوشته شده است، پیاده‌سازی کرده‌اند. این واسط دارای متدهایی برای مدیریت دوره زندگی عامل‌های سیار می‌باشد. در شکل زیر بخشی از این واسط دیده می‌شود.

طراحی واسط MobileObject بسیار آسان بوده و از تحلیل چارچوب MASIF به دست می‌آید. بر اساس این استاندارد عامل‌ها می‌توانند ایجاد شوند، حرکت کنند، معلق شوند، ادامه فعالیت دهند و یا خاتمه پیدا کنند. پس یک آژانس باید تمامی این اعمال را پشتیبانی نماید. از طرف دیگر، MCO باید از رخدادهایی که برای آن اتفاق می‌افتد باخبر شود تا بتواند واکنش مناسب نسبت به آن رخداد را از خود نشان دهد. علاوه بر آن یک MCO باید قابلیت پذیرش درخواست‌های انتقال و خاتمه از راه دور را داشته باشد؛ با اینکه این اعمال در اصل توسط آژانس نگهدارنده MCO انجام می‌گیرد ولی MCO باید متدهای لازم برای ارتباط کلاینت‌ها را برای انجام چنین اعمالی پشتیبانی کند. MCO باید از رخدادهای مختلف دوره زندگی خود از شروع تا موفقیت یا شکست باخبر شود تا نه تنها بتواند به مناسبترین صورت عمل نماید بلکه، قادر باشد مجوزهای مختلف چنین عملی را بیازماید (جدول زیر را مشاهده نمایید).

پیاده‌سازی‌های انجام شده برای MCO، علاوه بر واسط MobileObject باید قابلیت رشته‌سازی[۶] را نیز پشتیبانی کنند. رشته‌سازی این امکان را فراهم می‌سازد تا بتوان حالت شی سیار را به صورت یک رشته تولید کرد. هدف این تحقیق، پیاده‌سازی یک چارچوب با قابلیت تحرک در جهت افزایش کارآیی عملیات مدیریتی مبتنی بر اشیا می‌باشد. بنابراین به جای استفاده از سرویس برونی‌سازی کوربا که فایل‌های stub زیادی تولید می‌کند و نیز دارای سرعت کمتری است از امکان سریال‌سازی خود جاوا استفاده شده است. لذا MCOها باید واسط java.io.Serializable را نیز پیاده‌سازی کنند.

www.SID.ir

Archive of SID


```
module mco{
  //....
  interface MobileObject {

    void afterBirth(in AgentSystem agnsys, in AgentInfo
    info, in Object arguments)
    raises(OperationException);

    void moveMe(in Location dstLocation, in Place
    dstPlace) raises(OperationException);

    void beforeMove(in Location dstLocation, in Place
    dstPlace) raises(OperationException);

    void afterMove(in AgentSystem agnsys, in Location
    dstLocation, in Place dstPlace)
    raises(OperationException);

    void afterMoveFailed(in Location dstLocation, in
    Place dstPlace, in Info message);

    void beforeDeath();

    void beforeSuspend() raises(OperationException);

    void beforeResume() raises(OperationException);

    void terminateMe() raises(OperationException);

    void beforeShutdown();
    //....
  };
};
```

شکل ۳- واسط MobileObject به زبان IDL

جدول ۱- مدیریت دوره زندگی MCO و متدهای واسط MobileObject

Agency operations	involved MCO's call-backs
createMCO	afterbirth
resumeMCO	Resume
suspendMCO	Suspend
moveMCO	moveMe beforeMove afterMove afterMoveFailed
terminateMCO	terminateMe beforeDeath

۳-۳- سناریوهای کنش MCO در محیط

توجه داشته باشید که در سناریوهای نشان داده شده، کنش‌های MAFFinder ارائه نشده است.

۳-۳-۱- ایجاد یک شی سیار

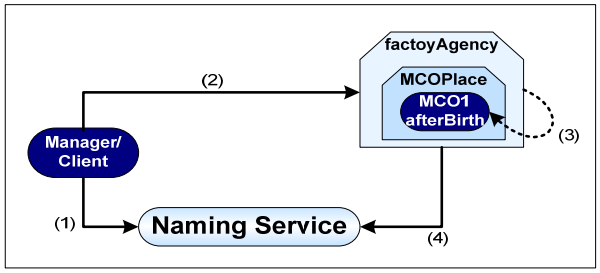
سناریوی کنش:

۱) Resolve: کلاینت شی factoryAgency را با استفاده از سرویس نامگذاری resolve می‌کند. این شی آژانس تولیدکننده MCOهای خاصی می‌باشد که کلاینت، مد نظر دارد.

۲) ایجاد: کلاینت متد createMCO مربوط به Factory مناسب را صدا می‌زند. این فراخوانی از راه دور بوده و بر اساس پروتکل IIOP صورت می‌گیرد.

۳) آژانس تولیدکننده، MCO درخواست‌شده را از روی کلاس‌های جاوایی که در اختیار دارد، در سیستم‌عاملی خود و در مکان تخصیص داده شده ایجاد می‌کند.

۴) Bind: شی ایجاد شده جدید، زمینه اسمی جدیدی در Naming Service ایجاد کرده و نام خود را به آن منسوب می‌کند و سپس کلاینت‌ها می‌توانند با یافتن مولفه نام، کنش‌های لازم را با آن انجام دهند.



شکل ۴- ایجاد یک شی سیار

۳-۳-۲- حذف شی سیار

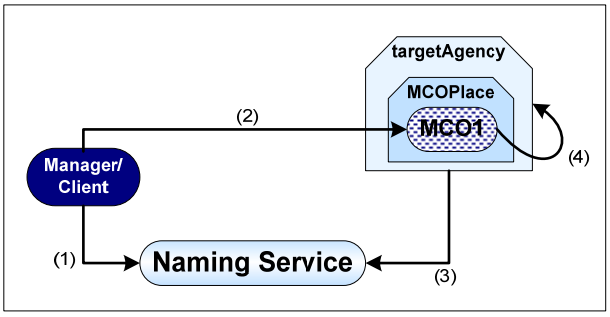
سناریوی کنش:

۱) Resolve: کلاینت با اتصال به Naming Service، شیی که باید حذف شود را جستجو می‌کند تا بتواند به شی مورد نظر دسترسی داشته باشد.

۲) حذف: کلاینت متد terminateME مربوط به MCO را صدا می‌زند.

۳) Unbind: MCO با فراخوانی تابع terminateMCO آژانسی که در آن مقیم است، عملیات حذف را می‌آغازد. در ابتدا، آژانس، مدخل نام MCO را از انتساب Naming Service خارج می‌کند.

۴) حذف نهایی: پس از حذف انتساب نام MCO از سرویس نامگذاری، آژانس مذکور خود شی MCO را حذف می‌کند.



شکل ۵- حذف شی سیار

۳-۳-۳- انتقال یک شی سیار

سناریوی کنش:

۱) Resolve: کلاینت، نام MCO موردنظر را به سرویس نامگذاری ارسال کرده و اشاره‌گر MCO را برمی‌گرداند.

۲) آغاز انتقال: کلاینت متد moveMe شی MCO را فراخوانی می‌کند تا انتقال صورت گیرد. برای مثال در شکل فوق انتقال MCO ایجاد شده در factoryAgency را به آژانس مقصد (NEAgency) بیان می‌کند.

۳) حال MCO متد moveMCO مربوط به آژانس دربرگیرنده خود را فراخوانی می‌کند تا انتقال به آژانس مقصد را آغاز کند.

۴) Resolve: آژانس مبدا، اشاره‌گر آژانس مقصد را با ارسال درخواست به سرویس نامگذاری دریافت می‌کند.

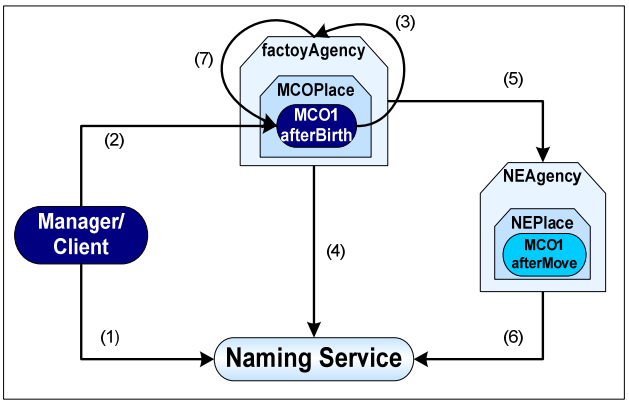
www.SID.ir

Archive of SID

۵) انتقال: آژانس مبدا MCO موردنظر را به آژانس مقصد انتقال می‌دهد که این عمل شامل سریال‌سازی شی MCO و ارسال آن به آژانس مقصد می‌باشد. البته، قابل ذکر است که در اینجا، انتقال حالت اجرایی MCO مدنظر نبوده و فقط حالت داده آن برونی شده و ارسال می‌شود.

۶) Rebind: آژانس مقصد فرم برونی شده MCO را دریافت کرده و بر اساس کلاس‌هایی که به آنها دسترسی دارد، شی مورد نظر را در زمینه خود ساخت و با استفاده از سرویس نامگذاری، نام آن را Rebind می‌کند.

۷) حذف: پس از آنکه شی MCO با موفقیت و بدون خطا به آژانس مقصد انتقال یافت، آژانس مبدا MCO موجود در محیط اجرایی خود را حذف می‌کند.



شکل ۶- انتقال یک شی سیار

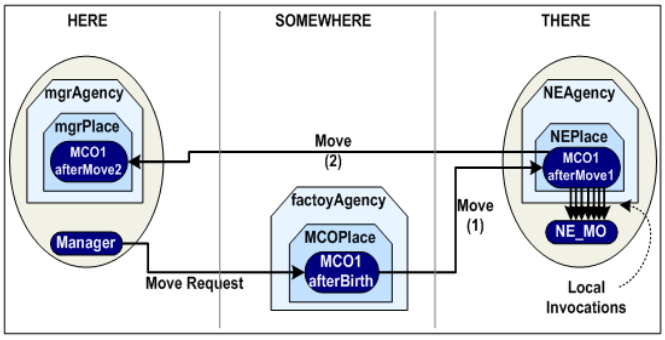
در اینجا نحوه استفاده از این چارچوب برای اهداف مدیریت شبکه بیان می‌شود. سه نوع آژانس در شبکه موجود است:

۱) آژانس مدیر، که در نزدیکی اشیا مدیر ایجاد می‌شود.

۲) آژانس مولد، که در هر جایی از شبکه می‌تواند قرار گیرد.

۳) آژانس شی مدیریت‌شونده، که در نزدیکی شی مدیریت شوند استقرار دارد.

مدیر شبکه به عنوان یک شی کوربا، در نزدیکی آژانس مدیر است. و شی مدیریت شونده نیز خود یک شی است که در نزدیکی آژانس عنصر شبکه قرار دارد. شی MCO هم توسط آژانس مولد ایجاد می‌شود. منظور از عبارت "نزدیک"، حضور در یک شبکه محلی یا روی یک میزبان می‌باشد. مدیر می‌تواند با درخواست از آژانس مولد، شی MCO خاص موردنظر خود را ایجاد کرده و آن را به نزدیک شی مدیریت‌شونده ارسال نموده و سپس آن را به مقصد دیگر بفرستد و یا اینکه برنامه سفر آن را از ابتدا مشخص نماید.



شکل ۷- الگوی مدیریت شبکه با استفاده از MCO

برای مثال مدیر می‌تواند، برای جمع‌آوری اطلاعات از یک (یا چند) شی، در خواست ساخت یک MCO جمع‌کننده داده را به factoryAgency ارسال کند و سپس آن را با دادن برنامه سفر به سوی اشیای مختلف ارسال کند تا، به جمع‌آوری اطلاعات بپردازد.

همانگونه که در شکل فوق دیده می‌شود، مدیر درخواست تولید شی MCO1 را با فراخوانی راه دور createMCO از آژانس مولد تقاضا می‌کند. در این عملیات، می‌توان پارامترهای لازم برای ایجاد این شی را به آژانس مولد داد (مانند اینجا که برنامه سفر به او داده می‌شود). شی MCO ایجاد شده، بلافاصله شروع به حرکت بر روی شبکه کرده و به آژانس نزدیک شی مدیریت‌شونده گسیل می‌شود. حال بر اساس کد نوشته شده در پیاده‌سازی آن، شروع به انجام فراخوانی‌های محلی کرده و اطلاعات لازم را از شی مدیریت‌شونده دریافت می‌کند.

منظور از فراخوانی‌های محلی به معنای فراخوانی‌های با تاخیر کم می‌باشد چرا که، آژانس عنصر شبکه و شی مدیریت‌شونده هر دو در یک شبکه محلی و یا حتی بر روی یک میزبان می‌توانند وجود داشته باشند. شی MCO پس از انجام جمع‌آوری داده، به سوی مقصد حرکت می‌کند که می‌تواند همان مدیر آغازکننده انتقال باشد و یا اینکه مدیر دیگری که وظیفه بررسی اطلاعات را دارد، به عنوان مقصد نهایی این MCO مشخص شده است.

۴- ارزیابی

در این بخش به بررسی مدل‌ها و معیارهای مورد ارزیابی پرداخته می‌شود. این مدل‌ها و معیارها بر اساس استفاده از چارچوب کوربا به عنوان زیرساخت ارتباطی مدیریت شبکه بوده و با یکدیگر مقایسه می‌شود. مدل‌ها به دو بخش ایستا و سیار تقسیم‌بندی می شود. البته این مقایسه در پارامترهای کارآیی انجام می‌گیرد که عمده‌ترین این پارامترها را می‌توان مدت زمان پاسخ به اعمال مدیریتی و نیز ترافیک ایجاد شده بر روی شبکه در نظر گرفت.

در اینجا ترافیک شبکه، هم برای مدیر و هم برای سرویس‌دهنده نامگذاری اندازه‌گیری و مقایسه شده است. آنچه که واضح است، نقش غیر قابل انکار سرویس دهنده های نامگذاری[۷] می‌باشد که می‌تواند در برخی مواقع گلوگاه سیستم شود و لذا، از اهمیت بالایی برخوردار است. پس ترافیک اندازه‌گیری شده هم برای مدیر و هم برای سرویس‌دهنده نامگذاری می‌باشد.

برای این ارزیابی‌ها، آژانس و شی بر روی یک ماشین قرار داده شده‌اند. همچنین، اندازه‌گیری‌های انجام شده برای هر مورد، ۱۰۰ بار تکرار شده و میانگین آنها ذکر گردیده است. اجرای این پروژه بر روی ماشین‌های مشابه با مشخصات زیر بوده است:

پردازش‌گر: AMD Sempron 2.4

حافظه: ۵۱۲ مگابایت

سیستم عامل: Microsoft windows xp SP2

ORB: JacORB 2.2.3

۴-۱- مدل‌های مورد ارزیابی

مدل‌هایی که مورد ارزیابی قرار گرفته است، مبتنی بر ویژگی قابلیت تحرک و همچنین نحوه استفاده از سرویس نامگذاری می‌باشند. بدین ترتیب، یک عملیات مدیریتی می‌تواند به طرق مختلف پیاده‌سازی و اعمال شود. مدل‌ها را می‌توان به سه بخش زیر تقسیم نمود:

www.SID.ir

Archive of SID

۴-۱-۱- مدل ایستا

این مدل، همان مدل عمومی و سنتی استفاده از کوربا برای پردازش توزیعی توسط اشیای توزیع‌شده می‌باشد که در آن اشیا به صورت توزیع‌شده بر روی میزبان‌های مختلف در شبکه قرار گرفته و با فراخوانی‌های از راه دور رویه‌های یکدیگر به کنش متقابل می‌پردازند. در این مدل، اشیا به هیچ وجه دارای قابلیت تحرک نبوده و بر روی میزبانی که اجرا شده و مقیم شده‌اند باقی می‌مانند. مدیر شبکه به عنوان یک شی ایستا بر روی یک میزبان خاص قرار داشته و اشیای مربوط به تجهیزات مدیریت‌شونده در میزبان‌های دیگر استقرار یافته‌اند. مدیر برای انجام اعمال مدیریتی، درخواست‌های فراخوانی را بر روی شبکه و به صورت مستقیم به شی مدیریت‌شونده ارسال می‌دارد. در این مدل، نقش سرویس‌دهنده بسیار کلیدی است چرا که باید تمامی اشیای مدیریت‌شونده در آن ثبت شوند تا مدیر بتواند مستقیما متدهای آن را فراخوانی نماید. با افزایش تعداد اشیای مدیریت شونده، سرویس نامگذاری ممکن است دچار اختلال شود.

۴-۱-۲- مدل سیار

در این مدل، اشیا می‌توانند بر روی میزبان‌های مختلف حرکت نمایند. این مدل، از نظر برنامه‌نویسی و پیاده‌سازی سیاست مدیریتی، شبیه مدل قبل می‌باشد و مکان اشیا نیز کاملا مشخص می‌باشد. بر اساس مکانیزم گفته شده در بخش‌های قبل، اشیا می‌توانند در شبکه حرکت نموده و دوباره خود را در سرویس نامگذاری ثبت نمایند، لذا این امر تقریبا تاثیری در پیاده‌سازی‌ها ندارد. در این مدل، مدیر برای انجام وظایف مدیریتی از اشیای سیار بهره می‌گیرد. اشیا سیار می‌توانند ساختار بسیار ساده و یا پیچیده داشته باشند و این بستگی به میزان هوشمندی و منطقی دارد که مدیر لازم می‌داند تا در آنها پیاده‌سازی شود. در مدل‌هایی که ارزیابی شده است، از اشیا سیار برای جمع‌آوری اطلاعات استفاده شده است به این ترتیب که آنها بتوانند به نزدیک اشیای مدیریت‌شونده مهاجرت نموده و فراخوانی‌های راه دور را از نزدیک انجام داده و نتایج را در خود نگهداشته و به مدیر انتقال دهند. پر واضح است که این اشیا می‌توانند اعمال مختلفی نظیر خلاصه‌سازی، فشرده‌سازی و یا یک سری پیش‌پردازش‌های دیگر را در میزبان انجام دهند.

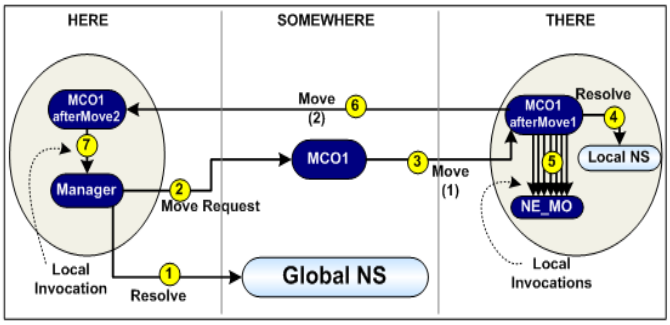
در این مدل نیز همانند مدل پیشین، تنها یک سرویس‌نامگذاری موجود بوده و تمامی آژانس‌ها، اشیای سیار، اشیای مدیریت‌شونده و مدیر در آن ثبت شده‌اند لذا، در این طرح نیز نقش سرویس‌دهنده نام بسیار مهم بوده و ممکن است سیستم مدیریت شبکه را دچار اختلال نماید.

۴-۱-۳- مدل سیار با سرویس‌دهنده محلی نامگذاری

یکی از مشکلات اساسی که در دو مدل قبل دیده می‌شود، استفاده از "یک" سرویس‌دهنده نام برای تمامی اشیای موجود در سیستم مدیریت شبکه می‌باشد. یکی از مشکلات اساسی مدل ایستا، اجبار در ثبت اشیا در یک سرویس‌دهنده قابل دسترسی توسط مدیر و اشیای مدیریت‌شونده می‌باشد. در مدل اول برای رفع مشکل گلوگاه بودن سرویس‌دهنده نام، از خوشه‌بندی سلسله‌مراتبی استفاده می‌کنند. به این ترتیب که اشیایی را که در حوزه دسترسی یکدیگر قرار دارند، بر روی یک سرویس‌دهنده نام ثبت می‌کنند و این اشیا با اشیای حوزه سرویس‌دهنده نامگذاری دیگر نمی‌توانند ارتباط برقرار کنند. به عنوان مثال فرض می‌کنیم برای هر شبکه محلی از اشیا، یک سرویس‌دهنده نام درنظر گرفته می‌شود و اشیای ثبت شده در آن فقط می‌توانند به همدیگر دسترسی پیدا کنند و اشیایی که از بیرون از این شبکه خواستار دسترسی به آنها می‌باشند، بایستی به این سرویس‌دهنده دسترسی داشته باشند. فرض کنید مدیر شبکه در یک سرویس‌دهنده نام عمومی خود را ثبت کرده باشد و اشیای مدیریت شونده نیز در سرویس دهنده های محلی خود ثبت شده باشند. در این صورت مدیر نمی‌تواند از طریق سرویس‌دهنده عمومی به آنها دسترسی پیدا کند. برای رفع این مشکل دو راه‌حل وجود دارد: یکی اینکه، مدیر باید نگاشتی از اشیای مدیریت‌شونده و سرویس دهنده های آنها داشته باشد تا بتواند با درخواست از سرویس‌دهنده مناسب، اشاره‌گر شی را به دست آورد که این خود باعث افزایش بار پردازشی مدیر شده و در بسیاری از موارد که سرویس‌دهنده‌های محلی دور از مدیر هستند، باعث افزایش تاخیر می‌شود. همچنین بار ترافیکی مدیر را افزایش می‌دهد. روش دیگر این است که شی مدیریت‌شودنه علاوه بر ثبت در سرویس دهنده های محلی در سرویس دهنده های عمومی نیز ثبت گردد که این روش مشکل خاصی را حل ننموده و همان مدل اول را تداعی می‌کند.

تمامی این مشکلات به دلیل عدم توانایی در انتقال اشیا به حوزه‌های نامگذاری مختلف است. مدلی که پیشنهادی ما برای سیستم مدیریت شبکه طرح به این ترتیب است که هر آژانس یک جفت سرویس دهنده محلی و عمومی دارد. اسامی آژانس‌ها و MCOها در سرویس‌دهنده نام عمومی ثبت می‌شود و از سرویس‌دهنده‌های محلی برای ثبت اسامی اشیای مدیریت‌شونده و مدیران استفاده می‌شود.

روال کار به این ترتیب است که: مدیر، از سرویس‌دهنده عمومی درخواست یافتن شی MCO مورد نظر خود را می‌کند و سپس آژانس دربرگیرنده این شی نیز از NS عمومی درخواست اشاره‌گر آژانس مقصد نموده و MCO به مقصد ارسال می‌شود. در آژانس مقصد، MCO، نام شی مدیریت‌شونده را از سرویس دهنده محلی درخواست نموده و سپس فراخوانی‌ها را انجام داده و با ارسال درخواست به سرویس دهنده عمومی برای یافتن آژانس شی مدیر، به آژانس مبدا برمی‌گردد (همانگونه که در شکل زیر توضیح داده شده است).



شکل ۸- مدیریت شبکه با استفاده از MCO و سرویس دهنده محلی

۴-۲- معیارهای مورد ارزیابی

اندازه‌گیری‌های انجام شده در پارامترهای کارآیی می‌باشد که مهم‌ترین این معیارها را می‌توان مدت زمان پاسخ به اعمال مدیریتی و نیز ترافیک ایجاد شده بر روی شبکه در نظر گرفت. در ادامه به بررسی بیشتر آنها و نتایج مربوطه پرداخته می‌شود.

۴-۲-۱- اندازه‌گیری زمان پاسخ

زمان پاسخ برای هر یک از مدل‌های ایستا و سیار چنین تعریف می‌شود:

www.SID.ir

Archive of SID

- زمان پاسخ برای مدل ایستا عبارت است از مدت زمان میان ارسال درخواست به سرویس‌دهنده نام برای یافتن شی مدیریت‌شونده و دریافت کل نتایج فراخوانی‌ها.
- زمان پاسخ برای مدل سیار عبارت است از مدت زمان میان ارسال درخواست به سرویس‌دهنده نام برای یافتن شی MCO و ارسال آن به سمت اشیا مدیریت‌شونده و بازگرداندن نتایج فراخوانی‌ها به مدیر.

اندازه‌گیری‌های انجام شده برای حالت‌های مختلف در موارد زیر آورده شده است. در هر یک از آنها شرایط آزمایش بیان شده است.

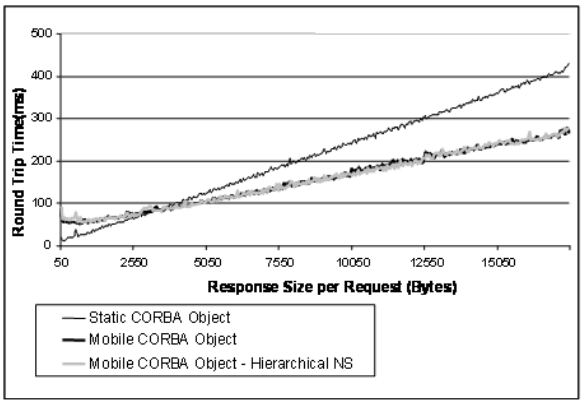
این شرایط عبارتند از:

۱) پهنای باند شبکه

۲) اندازه پاسخ برگشتی برای هر ویژگی (که معادل با یک فراخوانی راه دور می‌باشد)

- مورد ۱

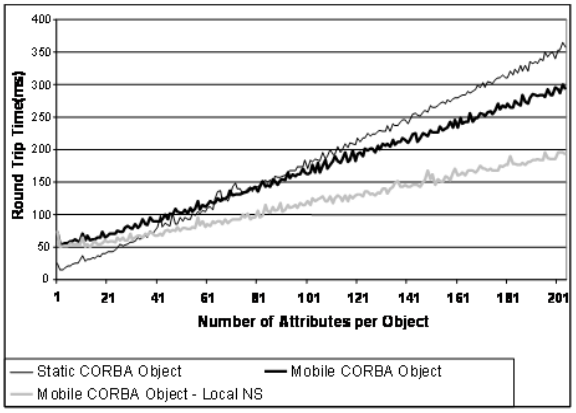
پهنای بند	10Mbps Ethernet
اندازه پاسخ برگشتی برای هر ویژگی	Variable
تعداد ویژگی‌ها برای هر شی مدیریت‌شونده	50
تعداد اشیای مدیریت‌شونده برای هر آژانس	1



شکل ۹- نمودار زمان پاسخ بر حسب اندازه پاسخ

- مورد ۲

پهنای بند	10Mbps Ethernet
اندازه پاسخ برگشتی برای هر ویژگی	100Bytes
تعداد ویژگی‌ها برای هر شی مدیریت‌شونده	Variable
تعداد اشیای مدیریت‌شونده برای هر آژانس	1



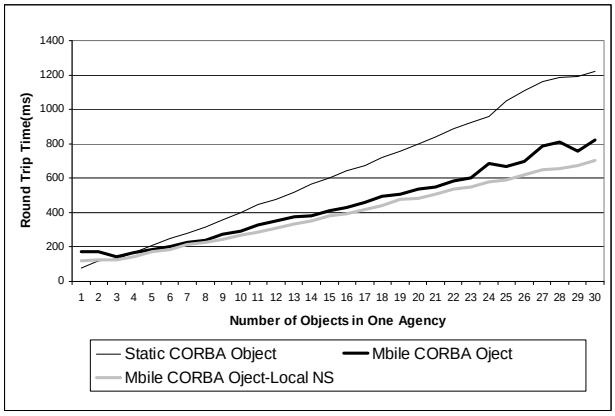
شکل ۱۰- نمودار زمان پاسخ بر حسب تعداد ویژگی‌ها

۳) تعداد ویژگی‌ها برای هر شی مدیریت‌شونده

۴) تعداد اشیای مدیریت‌شونده برای هر آژانس

- مورد ۳

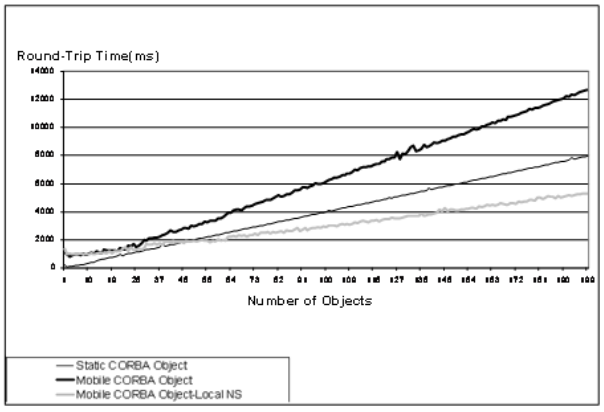
پهنای بند	10Mbps Ethernet
اندازه پاسخ برگشتی برای هر ویژگی	50Bytes
تعداد ویژگی‌ها برای هر شی مدیریت‌شونده	30
تعداد اشیای مدیریت‌شونده برای هر آژانس	Variable



شکل ۱۱- نمودار زمان پاسخ بر حسب تعداد اشیا

- مورد ۴

پهنای بند	64Kbps
اندازه پاسخ برگشتی برای هر ویژگی	100Bytes
تعداد ویژگی‌ها برای هر شی مدیریت‌شونده	Variable
تعداد اشیای مدیریت‌شونده برای هر آژانس	1



شکل ۱۲- نمودار زمان پاسخ بر حسب تعداد ویژگی با پهنای باند کم

۴-۲-۲- تفسیر نتایج اندازه‌گیری زمان پاسخ

همانگونه که در اکثر نمودارها دیده می‌شود، زمان پاسخ شی سیار با سرویس محلی بهتر از مدل‌های دیگر است.

موارد ۱ نتیجه تاثیر اندازه پاسخ برگشتی بر روی زمان پاسخ را نشان می‌دهد. در این موارد تعداد ویژگی‌ها، متغیر است. به علت اینکه پاسخ برگشتی برای هر فراخوانی، اندازه کمی دارد لذا زمان ارسال و دریافت پاسخ به همراه سرآیند، در برابر مدل سیار قابل ملاحظه می‌باشد. در مورد ۲ تاثیر تعداد فراخوانی‌ها با حجم

www.SID.ir

Archive of SID

داده‌های متفاوت بر روی زمان پاسخ نشان داده می‌شود. در این حالت نیز، معمولا نتایج مدل‌های سیار بهتر از ایستا می‌باشد. در موارد ۳ و ۴ نیز تاثیر تعداد اشیا بر روی زمان پاسخ بررسی شده است و در آنها هم تعداد ویژگی و هم پهنای باند شبکه را تغییر داده و نتایج ثبت شده است. نتایج نشان می‌دهد که در تمامی موارد، مدت زمان پاسخ مدل‌های سیار بهتر از مدل ایستا می‌باشد.

در کل باید این موضوع در نظر گرفته می‌شود که مدت زمان پاسخ به پارامترهای زیادی وابسته است و در مواردی که در بالا ذکر شده است، سعی شده است شرایط یکی در نظر گرفته شود ولی آنچه که واضح است این است که در واقعیت، شبکه می‌تواند بسیار متفاوت بوده و تاخیرهای متفاوت داشته باشد همان طور که قدرت پردازشی میزبان‌ها نیز متغیر است که در آن حالات، مدل‌های سیار از پاسخ بهتری برخوردار است.

همانگونه که در نمودارهای فوق نمایش داده می‌شود، تفاوت میان مدل‌های سیار با سرویس دهنده عمومی و محلی هنگامی اتفاق می‌افتد که میزان رجوع به سرویس دهنده قابل ملاحظه می‌باشد. همچنین تاخیر مدت زمان ارسال و دریافت داده در مقابل تاخیر درخواست از سرویس دهنده نباید قابل توجه باشد.

۴-۲-۳- اندازه‌گیری ترافیک

ترافیک اندازه‌گیری شده هم برای حالت ایستا و هم برای سیار عبارت است از ترافیک ارسال، ترافیک دریافت و ترافیک مجموع برای مدیر و سرویس‌دهنده نام عمومی.

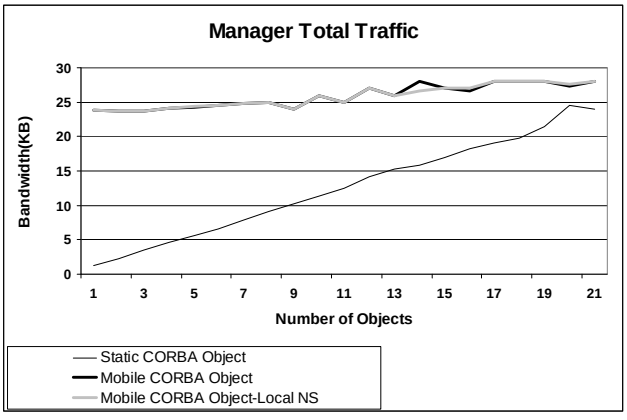
اندازه‌گیری ترافیک، برای سه حالت انجام شده است.

- یکی آنکه پاسخ فراخوانی‌ها متغیر باشد.
- دیگری برای حالتی که تعداد اشیای مدیریت‌شونده داخل آژانس‌ها متغیر باشد.
- و در پایان برای حالتی که تعداد آژانس‌ها متغیر باشد.

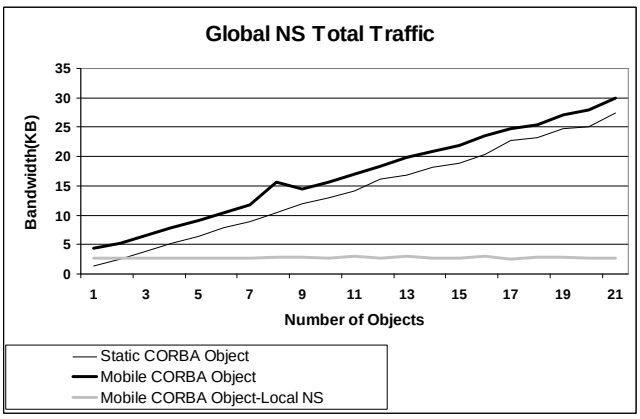
همانگونه که در شکل‌ها نیز مشاهده می‌شود، در کل استفاده از MCO باعث کاهش ترافیک بر روی مدیر و نیز سرویس دهنده عمومی می‌شود. علاوه بر آن باعث از میان رفت خاصیت گلوگاهی آنها نیز می‌شود.

- مورد ۱

اندازه پاسخ برگشتی برای هر ویژگی	100 Bytes
تعداد ویژگی‌ها برای هر شی مدیریت‌شونده	1
تعداد اشیای مدیریت‌شونده برای هر آژانس	Variable



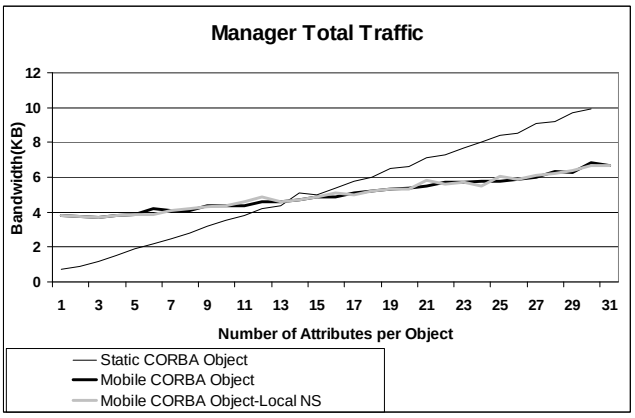
شکل ۱۳- نمودار ترافیک مدیر بر حسب تعداد اشیا



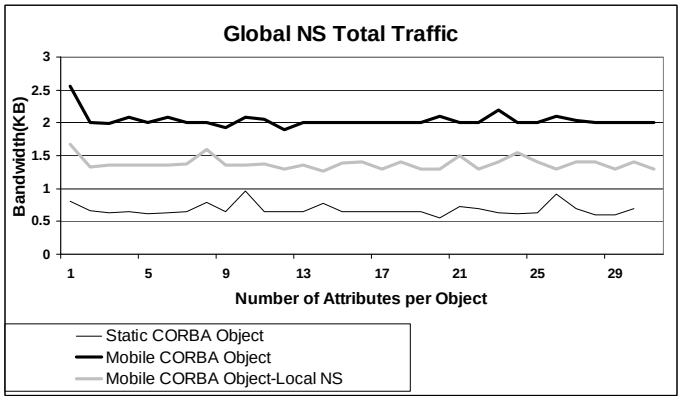
شکل ۱۴- نمودار ترافیک سرویس‌دهنده نام بر حسب تعداد اشیا

- مورد ۲

اندازه پاسخ برگشتی برای هر ویژگی	100 Bytes
تعداد ویژگی‌ها برای هر شی مدیریت‌شونده	Variable
تعداد اشیای مدیریت‌شونده برای هر آژانس	1



شکل ۱۵- نمودار ترافیک مدیر بر حسب تعداد ویژگی‌ها



شکل ۱۶- نمودار ترافیک سرویس‌دهنده نام بر حسب تعداد ویژگی‌ها

۵- نتیجه‌گیری

رویکرد کاربردهای مدیریت شبکه به سمت استفاده از عاملهای هوشمند سیار می‌باشد تا بتوانند مشکلات استفاده از معماری‌های سنتی و چارچوبهای قدیمی را

www.SID.ir

Archive of SID

رفع کند. مشکلاتی نظیر مقیاس‌پذیری، قابلیت اطمینان، قابلیت دسترسی، امنیت، کارآیی و ... همچنان در کاربردهای مدیریت شبکه مطرح هستند. این مسایل را تا حد زیادی می‌توان با استفاده از عامل‌های سیار و یا ترکیب آنها با سیستم‌های مدیریت شبکه موجود حل نمود. این تحقیق برای ارزیابی و سنجش پویای معیارهای کارآیی انجام شده است. با یافتن ضرائب مناسب برای معیارهای مختلف قابل اندازه گیری می توان مدیریت بسیار بهتری بر انجام پردازش در شبکه خصوصا شبکه های پردازش ابری و نسل آینده محیط فراگیر انجام داد.

در این پروژه نمونه بسیار کوچکی از قابلیت‌های (قابلیت تحرک) عامل‌های سیار هوشمند به کوربا اضافه شده است. نتایج مشاهده شده، حاکی از بهبود کارآیی سیستم مدیریت شبکه پیاده‌سازی شده است.

تشکر و قدردانی

این پروژه با حمایت مالی پژوهشگاه تحقیقات ارتباطات و فناوری اطلاعات انجام گرفته است.

اختصارات

CMIP	Common Management Information protocol
CORBA	Common Object Request Broker Architecture
DNM	Distributed Network Management
HS	Hierarchical static
IIOP	Internet Inter-ORB Protocol
LAN	Local Area Network
MA	Mobile Agent
MASIF	Mobile Agent System Interpretability Facility
MCO	Mobile CORBA Object
NE	Network Element
NMS	Network Management System
NS	Name Server
OMG	Object Management Group
ORB	Object Request Broker
SM	Strong Mobility
SNMP	Simple Network Management Protocol
WM	Weak Mobility

مراجع

[1] W. Stallings, *SNMP, SNMPv2 and RMON: Practical Network Management*. Addison-Wesley 1999.

[2] ITU-T Recommendation M.3010, *Principles for a Telecommunication Management Network,* May 1996.

[3] J. Case, M. Fedor, M. Schoffstall, and C. Davin, "The Simple Network Management Protocol (SNMP)," *RFC 1157,* May 1990.

[4] ITU-T Recommendation X.711, *Common Management Information Protocol (CMIP)-Part 1: Specification,* 1991.

[5] OMG, *The Common Object Request Broker: Architecture and Specification,* Revision 2.4, Oct 2000.

[6] E. Reuter, and F. Baude, "*System and Network Management Itineraries for Mobile Agents*," MATA, 2002.

[7] T. M. Chen, S. S. Liu, "A Model and Evaluation of Distributed Network Management Approaches," *IEEE Journal on Selected Areas in Communications,* vol. 20, no. 4, pp. 850-857, May 2002.

[8] IBM, *Mobile Agent System Interoperability Facilities Specification*. Joint submission: GMD Fokus & IBM Corp., supported by Crystaliz Inc., General Magic Inc., The Open Group. OMG TC document orbos/97-10-05 (1997).

سیدمحمدرضا میرزابابائی فارغ‌التحصیل کارشناسی مهندسی کامپیوتر از دانشگاه صنعتی امیرکبیر در سال ۱۳۷۴ و کارشناسی ارشد هوش ماشین و رباتیک از دانشکده فنی دانشگاه تهران در سال ۱۳۷۷ و هم‌اکنون دانشجوی دکترای رشته معماری کامپیوتر در دانشگاه صنعتی امیرکبیر. علاقمند به موضوعات هوش مصنوعی و پردازش موازی.

آدرس پست‌الکترونیکی ایشان عبارت است از:

mirzababaei@aut.ac.ir

محمدکاظم اکبری فارغ‌التحصیل کارشناسی دانشگاه شهید بهشتی در سال ۱۳۶۲ و کارشناسی ارشد و دکترا در مهندسی کامپیوتر از دانشگاه کیس وسترن رزرو در سال‌های ۱۹۹۱ و ۱۹۹۵. علاقمند به موضوعات شبکه‌های عصبی، طراحی سیستمی و پردازش موازی، پردازش توری و معماری کامپیوتر.

آدرس پست‌الکترونیکی ایشان عبارت است از:

akbari@aut.ac.ir

محمود صادقی فارغ‌التحصیل کارشناسی و کارشناسی ارشد رشته معماری کامپیوتر دانشگاه صنعتی امیرکبیر در سال‌های ۱۳۸۷ و ۱۳۹۰. علاقمند به موضوعات پردازش متحرک، پردازش ابری و موازی.

آدرس پست‌الکترونیکی ایشان عبارت است از:

sadeghi@aut.ac.ir

اطلاعات بررسی مقاله:

تاریخ ارسال: ۸۵/۱۱/۳

تاریخ اصلاح: ۹۱/۵/۲

تاریخ قبول شدن: ۹۱/۸/۱۶

نویسنده مرتبط: دکتر محمدکاظم اکبری، دانشکده مهندسی کامپیوتر و فناوری اطلاعات، دانشگاه صنعتی امیرکبیر، تهران، ایران.

[1] Centralized NMSs
[2] Network Element (NE)
[3] Mobile Agents
[4] Request-Reply
[5] Authority
[6] Streamable
[7] Name Server

www.SID.ir