

کاربرد زبان قید شی در افزایش دقت مدل های UML و مقایسه کارایی ابزارهای آن

الناز نوروزی^۱ و علیرضا خلیلیان^{۲*}

^۱ کارشناس نرم افزار، دانشگاه فنی و حرفه ای، دانشکده فنی دکتر شریعتی
elnaznoroizu1991@gmail.com

^۲ دانشجوی دکتری نرم افزار، دانشگاه اصفهان
khalilian@eng.ui.ac.ir

چکیده: مدل های UML، سال هاست که برای مدل سازی سیستم های نرم افزاری استفاده می شوند. یکی از مشکلات آن ها این است که نمی توان برخی قوانین معنایی محیط و قیود را در آن لحاظ کرد. زبان قید شی برای افزایش دقت مدل های UML معرفی شده است و هدف از آن تکمیل نمودار UML است و اگر به درستی مورد استفاده قرار گیرد، اضافه کردن عبارات زبان قید شی به نمودار UML، به طور قابل توجهی صحت مطالب و نهایتاً ارزش یک مدل UML را افزایش می دهد. در همین راستا، این زبان شامل ابزارهای متفاوتی می باشد که هر کدام دارای ویژگی هایی می باشد که کاربر بسته به نیاز خود، ابزار مناسبی را انتخاب کند. هدف از مقاله حاضر بررسی زبان قید شی، معرفی ابزارهای آن و مقایسه و ارزیابی آن ابزارها می باشد. بدین ترتیب پژوهشگران در این زمینه می توانند لیستی از ابزارهای زبان قید شی و مقایسه ی آن ها را داشته باشند تا بتوانند ابزار متناسب با نیازشان را برگزینند.

کلمات کلیدی: OCL، مدل های UML، محدودیت، پیش شرط، پس شرط، IDE4OCL

۱- مقدمه

در این مقاله یک توصیف اصولی از ابزارها داده می شود که معیار انتخاب آن ها داشتن دید کاربرپسند نسبت به سایر ابزارهای مرتبط می باشد. همچنین ابزارهای زبان قید شی از دیدگاه های مختلف با هم مقایسه می شوند.

ساختار ادامه مقاله به شرح زیر است: در بخش دوم کاربردهای زبان قید شی معرفی می شوند. بخش سوم مفاهیم بنیادی زبان قید شی را بیان می کند. بخش چهارم به دلایل ترکیب زبان قید شی و UML اختصاص دارد. در بخش پنجم موارد استفاده از زبان قید شی در UML مورد بحث قرار می گیرد. بخش ششم به تعریف محدودیت ها در زبان قید شی می پردازد. بخش هفتم ابزارهای رایج زبان قید شی را معرفی می نماید. در بخش هشتم هم نتایج مقایسه ابزارها گزارش می گردد. نهایتاً بخش نهم هم به نتیجه گیری و کارهای آتی اختصاص پیدا کرده است.

۲- کاربردهای OCL

زبان قید شی کاربردهای متعددی دارد [۴] از آن جمله می توان به موارد زیر اشاره کرد:

- به عنوان یک زبان پرس و جو

زبان قید شی یا OCL، زبان رسمی تعریف شده ای است که برای شرح توضیحات بر مدل های UML استفاده می شود [۱]. این توضیحات می تواند انواع شرط های مختلف برای سیستمی که قرار است مدل شود، باشد که با استفاده از این زبان نگه داشته می شود یا انواع پرس و جوهایی که بر روی اشیا تعریف می گردد. عبارات OCL می توانند برای مشخص کردن عملیات مختلف هنگام اجرا، استفاده شوند. سازنده های UML نیز می توانند از OCL برای مشخص کردن محدودیت های برنامه ی کاربردی مورد نظرشان و نیز برای تعریف کردن پرس و جوهایی بر روی مدل مربوط به آن استفاده کنند.

اگر با دقت به مدل های UML نگاه شود، ملاحظه می شود که همه نمودارهای آن به کمک ابزارهای CASE مربوط به UML به دقت طراحی و نمایش داده می شوند. اما متأسفانه معنای ایستای مدل های UML که با زبان OCL بیان می شوند، خطاهای متعددی از جنس دستوری، معنایی و مفهومی دارند [۲]. برای رفع این مشکل از ابزارهای پشتیبانی OCL استفاده می شود [۳]. در همه ی تحقیقات و پژوهش هایی که تاکنون صورت گرفته معیار انتخاب ابزارها، تلاش برای داشتن محیط توسعه یافته مجتمع IDE4OCL می باشد.

۵- مکان‌های استفاده OCL در مدل UML

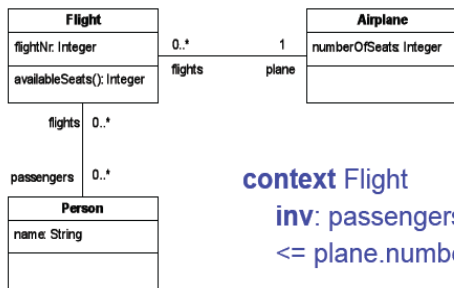
زبان قید شی را در بخش‌های مختلف نمودار UML می‌توان به‌کار گرفت [۷]:

۱. ویژگی‌ها
 - محدودیت روی یک کلاس و یا نوعی که همیشه باید نگه‌دارید
 - همچنین قابل استفاده برای انواع داده‌ها
۲. پیش‌شرط
 - محدودیتی است که باید قبل از اجرای یک عمل نگه داشته شود.
۳. پس‌شرط
 - محدودیتی است که باید پس از اجرای عملیات نگه داشته شود.
۴. گارد
 - محدودیت در انتقال ازحالتی به حالت دیگر

۶- تعریف محدودیت

یک محدودیت عبارتی است که بوسیله‌ی مقادیر ممکن یا جملات و یا معنای عناصر مدل‌ها، محدود شده است و این محدودیت باید همیشه برقرار باشد. محدودیت یک شرط یا یک قانون جامع را شرح می‌دهد. اساس زبان OCL مشخص کردن محدودیت‌های یک مدل (اضافه کردن محدودیت‌هایی به صفات و پیوند دادن بین اشیایی که نمی‌توانند در نمودار استفاده شوند) است. پیش‌شرط‌ها، پس‌شرط‌ها و ثابت‌ها را مشخص می‌کند. یک سفارش خاص را تعریف می‌کند و غیره محدودیت‌های ساده بر روی ویژگی‌های یک کلاس:

```
Customer.age >= 18 and age < 66
name: String title = if isMale then
title: String 'Mr.' else 'Ms.' endif
age: Integer name.size < 100
isMale: Boolean
```



context Flight
 inv: passengers -> size()
 <= plane.numberOfSeats

شکل (۱): نمودار کلاس پرواز

- برای مشخص کردن ثابت‌ها و انواع در کلاس‌ها
- برای مشخص کردن نوع ثابت‌ها در stereotype
- برای تعریف کردن پیش‌شرط‌ها یا پس‌شرط‌ها روی عملیات و متدها
- برای شرح حفاظ‌ها جهت حفاظت
- برای مشخص کردن اهداف پیغام‌ها در عملیات
- برای تعریف کردن محدودیت‌ها روی عملیات
- برای تعیین کردن قوانین اشتقاقی جهت مشخص کردن عبارات روی مدل‌های UML

۳- اساس زبان OCL

اساس زبان OCL مشخص کردن محدودیت‌های یک مدل (اضافه کردن محدودیت‌هایی به صفات) و پیوند دادن بین اشیایی که نمی‌توانند در نمودار استفاده شوند [۵].

۳-۱- تعریف

متن نوشته شده به سبک زیر، یک عبارت OCL را نشان می‌دهد.

```
'This is an OCL expression'
```

واژه زمینه، چارچوبی را برای عبارات معرفی می‌کند. کلمات کلیدی inv, pre و post، نشان دهنده‌ی stereotypes هستند.

```
context TypeName inv:
'this is an OCL expression with stereotype
<<invariant>> in the
context of TypeName' = 'another string'
```

در مثال بالا، کلمات کلیدی OCL به شکل برجسته هستند. این کلمات برجسته معنای رسمی ندارند اما قابلیت خواندن عبارات را بیشتر می‌کنند. این عبارات تنها با حروف اسکی نوشته شده اند.

۴- دلایل ترکیب OCL و UML

ترکیب OCL و UML بهترین نمودار و نمایش متنی را ارائه می‌دهد [۶]. نمایش متنی به تنهایی، به سهولت درک نمی‌شود. نمایش نموداری هم ممکن است نتواند همه‌ی مفاهیم را بطور دقیق نشان دهد. بدون عبارات OCL، مدل UML غیر قابل مشخص است. بدون نمودار UML هم عبارات OCL به عناصر مدلی اشاره می‌کنند که موجود نیست.

در یک قرارداد، همه‌ی رانندگان باید بالای ۱۸ سال سن داشته باشند.

۷- ابزارهای OCL

۷-۱- Dresden OCL

ابزار Dresden OCL [۶] بطور گسترده‌ای در آموزش و پژوهش و تمرین استفاده می‌شود. این ابزار مشخصات و ارزیابی محدودیت‌های OCL را پشتیبانی می‌کند و می‌تواند برای چندین متامدل روی لایه‌های مختلف در فضاهاى فنى [۹، ۱۰، ۱۱] مورد استفاده قرار گیرد. در سال ۱۹۹۹، پیاده‌سازی کتابخانه‌ی استاندارد و یک پارسر برای OCL شروع شد که در ابتدا جعبه ابزار Dresden OCL نامیده شد و ایده‌ی آن ارائه‌ی کد منبع باز از کتابخانه‌ی ابزارهای OCL بود که می‌توانستند به آسانی با دیگر ابزارهای مدل‌سازی مجتمع شوند. توسعه‌ی Dresden OCL در بازه‌ی زمانی بیش از یک دهه بود و بطور عمده‌ای بوسیله‌ی پیشرفت در تحقیقات OCL، تکامل تدریجی استاندارد OCL و تحولات ابزار توسعه‌ی نرم‌افزار Model Driven، تحت تاثیر قرار گرفت.

۷-۲- Eclipse OCL

پروژه‌ی Eclipse OCL [۱۲] یک پیاده‌سازی از مشخصات OMG OCL را ارائه می‌دهد که با مدل‌های Ecore و UML روی پلتفرم Eclipse استفاده می‌شود. مشارکت کد اولیه از IBM، یک API جاوا برای پارسر و ارزیابی فرا مدل‌های Ecore، را فراهم می‌کند. نسخه‌ی کامل شده‌ی بعدی، UML را پشتیبانی می‌کند و کنسول ارزیابی تعاملی را به آن اضافه می‌کند. این ابزار در انواع پروژه‌های Eclipse [۱۳] مثل CDO استفاده می‌شود که برای پشتیبانی از پرس‌جوهای سمت سرور روی انواع مدل‌ها بکار می‌رود.

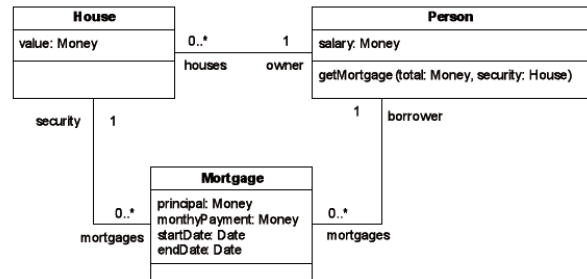
۷-۳- OCLE

هدف از ابزار OCLE [۱۴] پشتیبانی از MDE است و توسعه‌دهندگان را قادر می‌سازد تا کیفیت برنامه‌های کاربردی را بهبود بخشند. این ابزار ساخت و سازهای سازگار با UML (مطابق با WFRS) را پشتیبانی می‌کند و پلاگین‌های آن‌ها را با اضافه کردن ناظران و اظهارات OCL انجام می‌دهد.

۷-۴- SQUAM OCL

ابزار SQUAM OCL [۱۵] چارچوبی است که مدل‌سازی محیط‌های ناهمگن را یک‌پارچه می‌سازد و تجزیه و تحلیل کیفیت مدل‌ها و کتابخانه OCL تعریف شده توسط کاربر را پشتیبانی می‌کند. هدف نوشتن آسان عبارات OCL است، و در نتیجه کمک کردن به بهبود کاربرد و افزایش دامنه‌ی استفاده از این زبان. مثلاً هدف از

چگونگی این ویژگی در مثال زیر بیان شده است: در یک پرواز، تعداد مسافران به تعداد صندلی‌های هواپیما محدود می‌شود.
مثال ۲:

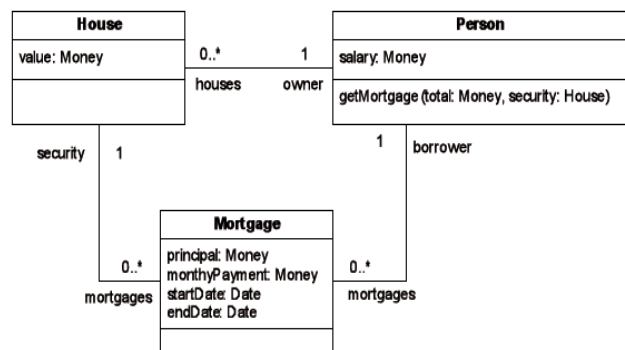


◆ context Mortgage

inv: startDate < endDate

شکل (۲): نمودار کلاس وام مسکن

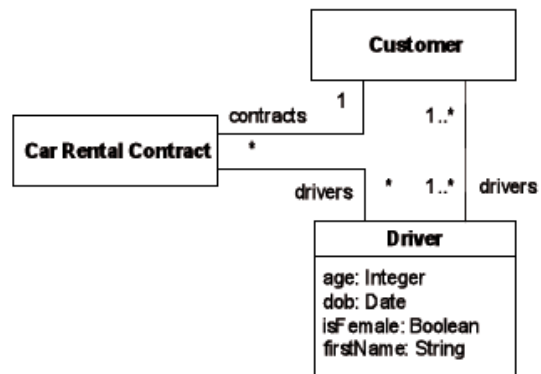
تاریخ شروع هر وام مسکن باید قبل از تاریخ پایان آن باشد.
مثال ۳:



Context Person::getMortgage(total:money,security:house)
Pre:self.mortgages.principal->sum()>=3*self.salary

شکل (۳): نمودار کلاس وام مسکن

وام مسکن جدید تنها زمانی داده خواهد شد که شخص درآمد کافی داشته باشد.
مثال ۴: شرکتی که اتومبیل اجاره می‌دهد.



شکل (۴): نمودار کلاس شرکت اجاره اتومبیل

۸- مقایسه ابزارها

هدف از جمع‌آوری داده‌ها، قادر ساختن کاربران برای انتخاب ابزار OCL متناسب با نیازهایشان است. این خصوصیت در مدل ویژگی متفاوت است و اهمیت آن وابسته به هدف استفاده از آن است. به همین دلیل ایجاد یک آمار اندازه‌گیری کمی یا کیفی از جزئیات مهم جلوگیری می‌کند و نباید به عنوان مقایسه‌ی ابزارها استفاده شود. معیار انتخاب اصلی برای ابزارهای موجود، توانایی آن‌ها برای خدمت‌رسانی به عنوان IDE4OCL بود. در نتیجه همه‌ی آن‌ها اجزای ارائه‌ی این قابلیت را دارند (جدول ۲). اهداف اصلی ابزارهای OCL متفاوت است و می‌تواند بوسیله‌ی معماری سطح بالا نشان داده شود. آن‌ها شامل اجزایی هستند که مربوط به بخش‌های مختلف از نظر OCL است: OCLE, SQUAM و USE قابلیت یک ابزار مدل‌سازی را فراهم می‌کنند. USE علاوه بر این قابلیت‌هایی را برای بررسی رسمی فراهم می‌کند، DRESDEN OCL و OCLE قابلیت ابزار مهندسی مبتنی بر مدل را ارائه می‌دهند.

جدول (۱): نمای کلی از ابزارهای OCL در چارچوب وب [۲۱]

نام ابزار	آدرس سایت
Dresden OCL	http://www.dresden-ocl.org/
مالک (owner)	Technische Universitaet Dresden, Germany
مجوز	LGPL
نسخه‌ی فعلی	3.1.0 (مربوط به 2011-01-17)
نسخه‌ی OCL پشتیبانی‌شده	OCL 2.2 / OCL 2.3
ارائه‌کننده‌ی اطلاعات	Birgit Demuth, Claas Wilke
Eclipse OCL	http://wiki.eclipse.org/MDT/OCL
مالک (owner)	Eclipse Foundation
مجوز	EPL v1.0
نسخه‌ی فعلی	3.1.0 (مربوط به 2011-11-06)
نسخه‌ی OCL پشتیبانی‌شده	OCL 2.3
ارائه‌کننده‌ی اطلاعات	Ed Willink
Oclarity	http://www.empowertec.de/products/oclarity/
مالک (owner)	EmPowerTec AG
مجوز	دارای مجوز تجاری اما قابل استفاده رایگان برای همه
نسخه‌ی فعلی	2.2 (مربوط به 2011-11-22)
نسخه‌ی OCL پشتیبانی‌شده	OCL 2.0
ارائه‌کننده‌ی اطلاعات	Andreas Awenius
OCLE	http://lci.cs.ubbcluj.ro/ocle/
مالک (owner)	Babes-Bolyai University of Cluj-Napoca
مجوز	LGPL

چارچوب وب، قادر ساختن خصوصیات کاربرپسند و ارزیابی عبارات OCL است. این با ایده‌ی IDE4OCL مطابق است.

۷-۵- TOPCASED VF

ابزار TOPCASED [۱۶] نرم‌افزاری متن باز مبتنی بر Eclipse است و متدها و ابزارهایی را برای توسعه‌ی سیستم‌های تعبیه شده حساس، ارائه می‌دهد. این ابزار مبتنی بر پروژه‌های مدل‌سازی Eclipse است که برای آدرس دهی به مواردی از جمله مدل‌سازی، انتقال مدل‌ها، ردیابی نیازمندی‌ها، تولید کد، تولید سند و تصدیق مدل‌ها، بکار می‌رود [۱۷]. در میان تمام این مسائل، OCL بطور گسترده‌ای توسط اجزای TOPCASED VF [۱۸] استفاده می‌شود که به منظور بررسی سازگاری مدل‌های صنعتی در هر زمانی در طول دوره‌ی مدل‌سازی، استفاده می‌شود.

۷-۶- USE

ابزار USE [۱۹] توسعه‌دهندگان را در مراحل اولیه‌ی طراحی برای مدل‌های معتبر مشخص شده در زیرمجموعه‌ای از UML، پشتیبانی می‌کند [۲۰]. این ابزار تعریف کردن، اعتبارسنجی و بررسی زمان اجرای ثابت‌ها، پیش شرط و پس شرط‌ها را پشتیبانی می‌کند. از یک زبان ساده‌ی متنی برای تعریف مدل‌ها استفاده می‌کند. ابزار USE با موفقیت در دوره‌های مدل‌سازی دانشگاه برمن و چندین دانشگاه ملی و بین‌المللی مورد استفاده قرار گرفت و به عنوان یک کتابخانه با محصول نرم‌افزاری دولت فدرال آلمان یک‌پارچه شد که نامش XGenerator بود و به عنوان موتور پرس‌وجو و اعتبارسنج مورد استفاده قرار گرفت.

۷-۷- OCLARITY

ابزار OCLARITY [۲۰] ابزار مستقلی است که برای ارائه‌ی یک محیط نوشتن یکپارچه جهت آموزش OCL در نظر گرفته شده است و تمرکز بر سهولت نصب و استفاده دارد نه بر مجموعه ویژگی‌های قوی. این ابزار فقط معتبر بودن عبارات OCL را بررسی می‌کند و هیچ پشتیبانی در زمان اجرا ندارد. در حال حاضر ENTERPRISE ARCHITECT و MAGICDRAW را پشتیبانی می‌کند. ابزار OCLARITY با هیچ نرم‌افزار دیگری نمی‌تواند مجتمع شود حتی با نرم‌افزارهای OCL. با OCLARITY تلاش برای ارائه‌ی یک پیاده‌سازی مطمئن از استاندارد OCL2.0 انجام می‌شود. هدف آن اصلاح همه‌ی خطاهای گزارش شده در مورد استاندارد زبان یا قابلیت‌های موجود دیگر است. مهاجرت به استاندارد OCL 2.3 ممکن است در آینده رخ می‌دهد که بسته به علاقه و تقاضا از کاربران است. در غیر این صورت، هیچ برنامه‌ای برای اضافه کردن قابلیت‌های جدید به OCLARITY وجود ندارد.

قابلیت IDE4OCL ، امکان پذیر است (جدول ۳). تقریباً تمام ابزارها یک پارسر OCL، OCL evaluator، و ویراستار OCL دارند، همه ابزارها جز SQUAM و TOPCASED-VF، خودشان پارسر OCL دارند. ابزارهای SQUAM و TOPCASED-VF مبتنی بر OCL ECLIPSE هستند. همه ابزارها جز OCLARITY، OCL evaluator دارند و همه ابزارها جز USE خودشان ویراستار OCL دارند. ابزارهای OCL در بهبود قطعات و توسعه OCL، متفاوت هستند. ابزار OCL ECLIPSE تحلیلی جهت بررسی تاثیر بر بهینه‌سازی ارزیابی مجدد عبارات OCL، را ارائه می‌دهد. ابزار SQUAM از OCL Unit، OCL Lib، OCL Doc و OCL پشتیبانی می‌کند. ابزار TOPCASED-VF از OCL Checker و OCL Reporting برای ذخیره‌ی نتایج ارزیابی عبارات OCL درون یک مدل نتیجه و تولید یک گزارش HTML، پشتیبانی می‌کند. USE از OCL Unit و OCL Extensions در Ruby پشتیبانی می‌کند. هرچند که در توصیف ابزارها اشاره نشده بود اما جالب توجه است که ابزارهای OCL شباهت‌های فنی بهم دارند. تقریباً همه ابزارها در جاوا نوشته شده‌اند، فقط OCLARITY از چارچوب شبکه استفاده می‌کند. برای پیاده‌سازی های OCL ECLIPSE DRESDEN OCL استفاده می‌کنند که ادغام ابزارهای Eclipse مختلف مربوط به مدل‌سازی را امکان پذیر می‌کند.

جدول ۳: ابزارهای OCL با قابلیت IDE4OCL [۲۱]

قابلیت	نام ابزار						
	USE	TOPCASED-VF	SQUAM	OCLE	Oclarity	Eclipse	Dresden
OCL Parser	+			+	+	+	+
OCL Evaluator	+	+	+	+		+	+
OCL Editor		+	+	+	+	+	+
Impact Analyser							
OCL Libraries			+				
OCL Doc			+				
OCL Extensions	+		+				
OCL Checker		+					
OCL Reporting		+					

چهار ویژگی وجود دارد که در همه ابزارهای ارائه شده، پیاده‌سازی شده است: ویرایش اولیه، رابط سند، برجسته کردن گرامر، تطابق گرامر با نسخه‌های مختلف مشخصات. اینها تقریباً در تمامی نیازهای کاربران OCL مشاهده شده است. ویرایش اولیه و برجسته کردن گرامر به عنوان مهم‌ترین ویژگی شناخته شده‌اند. ویژگی‌های بیشتری که برای باقی ماندن درخواست شدند عبارتند از: تکمیل خودکار، اشکال زدایی و پشتیبانی از بازسازی کد. ویژگی تکمیل خودکار

نسخه‌ی فعلی	2.04 (مربوط به 2005-07-15)
نسخه OCL	OCL 2.0
پشتیبانی شده	
ارائه کننده‌ی اطلاعات	Dan Chiorean
	http://squam.info/
مالک (owner)	University of Innsbruck, arctis GmbH
مجوز	آموزشی و تجاری
نسخه‌ی فعلی	0.8.0 (مربوط به 2010-10-18)
نسخه OCL	OCL 2.0 (MDT OCL Eclipse Galileo)
پشتیبانی شده	
ارائه کننده‌ی اطلاعات	Joanna Chimiak-Opoka, Hannes Moesl
	http://gforge.enseeiht.fr/projects/topcased-vf/
مالک (owner)	TOPCASED consortium
مجوز	EPL v1.0
نسخه‌ی فعلی	4.3.0 (مربوط به 2011-02-07)
نسخه OCL	OCL 2.3 (ابزارهای مبتنی بر Eclipse MDT OCL)
پشتیبانی شده	
ارائه کننده‌ی اطلاعات	Sebastien Gabel
	https://sourceforge.net/projects/useocl/
مالک (owner)	University of Bremen, Database Systems Group
مجوز	GNU General Public License (GPL)
نسخه‌ی فعلی	2.6.2 (مربوط به 2010-11-02)
نسخه OCL	OCL 2.2
پشتیبانی شده	
ارائه کننده‌ی اطلاعات	Lars Hamann

جدول (۲): بررسی اجمالی ابزارهای OCL. کاربرد قسمت های خاکستری رنگ توسط ابزارهای خاص ارائه شده است [۲۱].

نام ابزار	هسته	اجزای اضافی			
Dresden OCL	IDE4OCL	ابزار مدل‌سازی	مخزن	ابزار MDE	ابزار FV آزمون
Eclipse OCL	IDE4OCL	ابزار مدل‌سازی	مخزن	ابزار MDE	ابزار FV آزمون
Oclarity	IDE4OCL	ابزار مدل‌سازی	مخزن	ابزار MDE	ابزار FV آزمون
OCLE	IDE4OCL	ابزار مدل‌سازی	مخزن	ابزار MDE	ابزار FV آزمون
SQUAM	IDE4OCL	ابزار مدل‌سازی	مخزن	ابزار MDE	ابزار FV آزمون
TOPCASED-VF	IDE4OCL	ابزار مدل‌سازی	مخزن	ابزار MDE	ابزار FV آزمون

نگاه دقیق‌تر به معماری این اجرا نشان می‌دهد که مشاهده‌ی شباهت‌ها و تفاوت‌های بین ابزارهای OCL و راه آن‌ها در پشتیبانی

تلاش براین بود تا مجموعه‌ای از بهترین ابزارهای OCL مناسب با IDE4OCL انتخاب کنیم اما ادعا نمی‌شود که این لیست کامل‌ترین می‌باشد، لیست طولانی‌تری که انواع مختلف ابزارها را پوشش می‌دهد در پورتال OCL در دسترس است. در این مقاله، تجزیه و تحلیل نیازمندی‌های کاربران OCL برای یک محیط مجتمع مناسب و ابزارهای موجود در آن، را بررسی کردیم. مدل ویژگی IDE4OCL در چارچوب وب پالایش و ثبت می‌شود که بصورت آنلاین برای عموم قابل دسترس است و برای ابزارهای OCL بیشتر قابل توسعه است. از چارچوب وب برای توصیف داده‌های جمع آوری شده از ابزارهای هفت گانه‌ی OCL استفاده می‌شود (DRESDEN OCL, ECLIPSE OCL, OCLARITY, OCLE, SQUAM OCL, TOPCASED VF USE) که ویژگی‌های سازگار با IDE4OCL را پشتیبانی می‌کنند. هر کدام از این ابزارها بطور خلاصه در این مقاله شرح داده شد.

مراجع

- [1] Warmer, J. B., & Kleppe, A. G. *The object constraint language: getting your models ready for MDA*. Addison-Wesley Professional, 2003.
- [2] Akehurst, D. H., & Bordbar, B. "On querying UML data models with OCL," In *<< UML >> 2001—The Unified Modeling Language. Modeling Languages, Concepts, and Tools*, pp. 91-103, 2001.
- [3] Chiorean, D., Bortes, M., Corutiu, D., & Sparleanu, R. "UML/OCL tools-objectives, requirements, state of the art-the OCLE experience," In *Proceedings of the 11th Nordic Workshop on Programming and Software Development Tools and Techniques NWPER*, Vol. 34, pp. 163-180, 2004.
- [4] Chimiak-Opoka, J. D., Demuth, B., Silingas, D., & Rouquette, N. F. "Requirements analysis for an integrated OCL development environment," *Electronic Communications of the EASST*, 24, 2010.
- [5] Demuth, B., Hußmann, H., & Loecher, S. "OCL as a specification language for business rules in database applications," In *<< UML >> 2001—The Unified Modeling Language. Modeling Languages, Concepts, and Tools*, pp. 104-117, 2001.
- [6] Gogolla, M., Büttner, F., & Richters, M. "USE: A UML-based specification environment for validating UML and OCL," *Science of Computer Programming*, Vol. 69, No. 1, pp. 27-34, 2007.
- [7] Heidenreich, F., Johannes, J., Karol, S., Seifert, M., Thiele, M., Wende, C., & Wilke, C. "Integrating OCL and textual modeling languages," In *Models in Software Engineering*, pp. 349-363, 2011.
- [8] Dresden (2014, May). [Online], Available: <http://www.dresden-ocl.org/>, 14
- [9] Demuth, B., & Wilke, C. "Model and object verification by using Dresden OCL," In *Proceedings of the Russian-German Workshop Innovation Information Technologies: Theory and Practice, Ufa, Russia*, pp. 687-690, 2009.
- [10] Hussmann, H., Demuth, B., & Finger, F. "Modular architecture for a toolset supporting OCL," *Science of Computer Programming*, Vol. 44, No. 1, 51-69, 2002.
- [11] Wilke, C., Thiele, M., & Wende, C. "Extending variability for OCL interpretation," In *Model Driven Engineering Languages and Systems*, pp. 361-375, 2010.

در چهار ابزار پیاده‌سازی شده است. اشکال زدایی پیچیده است و چندین زیرگزینه دارد و در زمینه‌ی OCL نسبتاً جدید است. هم‌چنین ویژگی‌های منحصر به فرد ابزارهای خاص جالب توجه است. ابزار OCL DRESDEN تنها ابزاری است که با MDE می‌آید و از ویژگی انتقال مدل پشتیبانی می‌کند. ابزار OCL2SQL نه تنها کد SQL که عبارات OCL را ارزیابی می‌کند، تولید می‌کند بلکه تبدیل مدل‌های UML برای دید رابطه‌ای را هم پشتیبانی می‌کند. ابزار OCL ECLIPSE یک دید ترکیبی OCL/MOF را ارائه می‌دهد. ابزار OCL SQUAM تنها ابزاری است که اسناد را خارج از نظرات OCL Doc تولید می‌کند. این ویژگی در ECLIPSE OCL توسعه یافته است. مسئله خطایابی چندگانه فقط در USE پیاده‌سازی شده است (دیدن متغیرها، درج مقادیر و آزمون خودکار). اما آن‌ها در OCL ECLIPSE, DRESDEN OCL و OCLE برنامه‌ریزی شده اند. USE منحصراً، زبان ماژولار را عرضه می‌کند اما احتمالاً به زودی تغییر می‌کند، در حال حاضر در DRESDEN OCL و ECLIPSE OCL توسعه یافته است. برای به رسمیت شناختن روند رو به رشد، جالب است که به ویژگی‌هایی توجه شود که هنوز اجرا نشده اند اما در حال توسعه هستند: فنون بازسازی کد گسترده‌تر و بیشتر در DRESDEN OCL در حال توسعه است. ویژگی‌های استاندارد مرتبط، دید و حوزه‌ی لغوی و انطباق با XML در ECLIPSE OCL در حال توسعه است و USE با ابزارهای آزمون در حال گسترش است.

۹- نتیجه‌گیری و کارهای آتی

مقایسه‌ی کیفی ابزارها یک مسئله‌ی حیاتی است و می‌تواند موضوعی برای تحقیقات بیشتر باشد. باید این را ذکر کرد که در این مقاله، داده‌های جمع آوری شده توسط خود توسعه‌دهندگان ابزارها ارائه شده است و این بینش عمیقی به ابزارها را تضمین می‌کند اما ممکن است از نحوه‌ی موثر بودن ابزارها جلوگیری کند. علاوه بر این در جمع آوری داده‌ها معلوم می‌شود که تعداد کمی از ویژگی‌های کاربردی (مثل ویژگی‌های اجرا شده، ویژگی‌های در حال توسعه، ویژگی‌های برنامه ریزی شده و ویژگی‌های پشتیبانی نشده) اعمال کرده است که ناکافی است و می‌تواند منجر به اشتباه شود. در گام بعدی باید ارزشیابی مقادیر بوسیله‌ی اجرا و تست کامل، گسترش داده شود. با این حال، نیاز است که در مورد معانی این ارزش‌ها مفصلاً بحث شود. باید ادغام کردن OCL درون زبان‌های دیگر آزمایش می‌شود و خطایابی عبارات OCL انجام گیرد. هم‌چنین افزایش مقیاس پذیری در ارزیابی OCL بررسی شود، چراکه مطالعات موردی حاکی از آن است که در ارزیابی بسته‌های OCL حجیم روی مدل‌های بزرگ و یا مجموعه‌ای از اشیا، مشکلات عملکردی وجود دارد.

باید تاکید کرد که هنوز هم هدف ترویج و بحث در مورد توسعه‌ی ابزارهای OCL است که برای هر مدل و زبانی و نیز برای برنامه‌های کاربردی پیشرفته‌ی صنعتی، مناسب است.

- [12] ECLIPSE MDT/OCL(2014, January) [Online], Available:
<http://wiki.eclipse.org/MDT/OCL>
- [13] eclipse (2014, May) [Online], Available:
<http://www.eclipse.org/projects/Object>
- [14] lci (2014, Apr) [Online] Available:
<http://lci.cs.ubbcluj.ro/ocle>.
- [15] squam(2010,March) [Online], Available:
<http://squam.info/>
- [16] topcased(2013, November) [Online], Available:
<http://www.topcased.org/>
- [17] Farail, Patrick, Pierre GOUTILLET, Agusti Canals, Christophe Le Camus, David Sciamma, Pierre Michel, Xavier Crégut, and Marc Pantel. "The TOPCASED project: a toolkit in open source for critical aeronautic systems design," *Ingenieurs de l'Automobile* 781, pp. 54-59, 2006.
- [18] Canals, S. Gabel, G. P. The SAM and OCL components of the TOPCASED project. In *Neptune'2009*. Paris, France, (2009, May). [Online], Available:
<http://gforge.enseiht.fr/projects/topcased-vf/>
- [19] sourceforge(2013), [Online], Available:
<http://sourceforge.net/projects/useocl/>
- [20] empowertec(2010), [Online], Available:
<http://www.empowertec.de/products/oclarity/>
Chimiak-Opoka, Joanna Dobroslawa, Birgit Demuth, Andreas Awenius, Dan Chiorean, Sebastien Gabel, Lars Hamann, and Edward Willink. "OCL tools report based on the ide4OCL feature model." *Electronic Communications of the EASST* 44, 2011.