



## استخراج معماری کلاس ها از متن برنامه

گزارش درس کامپایلر پیشرفته

دانشجو:

محسن امیریان ۹۵۷۲۳۰۳۴

استاد:

دکتر سعید پارسا

آبان ماه ۱۳۹۵

## مقدمه

در این گزارش با استفاده از ابزارهای موجود سعی کرده ایم معماری کلاس های یک برنامه را از داخل متن برنامه آن بیرون بکشیم. در واقع هدف اصلی این گزارش، آشنایی با نحوه ی استفاده از ابزارهای Understand، Bunch و Rational Rose جهت استخراج معماری کلاس ها از متن برنامه است. خروجی نهایی پس از طی کردن تمام مراحل، مدل ارتباطی کلاس های برنامه و خوشه بندی آنها می باشد. هدف نرم افزار Bunch (که در ادامه بیشتر با آن آشنا می شویم) ارائه ی این مدل به شکلی بهینه است.

## ابزار Understand:

برنامه ای که قصد بررسی کد آن را داشتیم، به زبان C# بوده است. نحوه ی کار به این صورت است که متن برنامه را در ابتدا به نرم افزار Understand می دهیم. این نرم افزار پس از انجام آنالیز روی کد، یک فایل CSV به عنوان خروجی می دهد. این فایل که یکی از فایل های نرم افزار Excel است، میزان ارتباط کلاس ها را در قالب ماتریس وابستگی کلاس ها مشخص می کند. در این فایل، برای تمام کلاس های موجود در برنامه، تعداد دفعاتی را که یک کلاس، کلاس دیگر را فراخوانی کرده است، مشخص شده است.

نا گفته نماند نرم افزار Understand شامل گزارش های دیگری نیز می باشد. که از آن ها می توان به UML Class Diagram و Dependency Graphs اشاره کرد.

خروجی مد نظر ما ماتریس وابستگی کلاس ها است که در محیط نرم افزار از مسیر زیر قابل دسترسی است:

Reports-> Dependency -> Class Dependencies -> Export Matrix CSV

در شکل زیر بخشی از خروجی این نرم افزار را مشاهده می کنید:

Dependé	LumiSoft	LumiSoft	LumiSoft	LumiSoft	LumiSoft	LumiSoft	LumiSoft	LumiSoft	LumiSoft	LumiSoft	LumiSoft	LumiSoft	LumiSoft
LumiSoft.Net.SIP.Message.SIP_t_ContentCoding						3							2
LumiSoft.Net.SIP.Message.SIP_MVGroupHFCollection<>						2	14	3				14	
LumiSoft.Net.SIP.Message.SIP_t_ReferredBy						2			9	6	3		
LumiSoft.Net.SIP.Message.SIP_t_RValue						3							4
LumiSoft.Net.SIP.Message.SIP_t_RAck						3							10
LumiSoft.Net.SIP.Message.SIP_t_Challenge						3							4
LumiSoft.Net.SIP.Message.SIP_t_Timestamp						3							2
LumiSoft.Net.SIP.Message.SIP_t_Credentials						3							4
LumiSoft.Net.SIP.Message.SIP_ParameterCollection												12	
LumiSoft.Net.SIP.Message.SIP_t_SecMechanism						2			29		11		2
LumiSoft.Net.SIP.Message.SIP_t_IdentityInfo						2			9		3		4
LumiSoft.Net.SIP.Message.SIP_t_CallID						3							2
LumiSoft.Net.SIP.Message.SIP_t_AcceptRange						3					21		4
LumiSoft.Net.SIP.Message.SIP_t_LanguageTag						2			3				2
LumiSoft.Net.SIP.UA.SIP_UA_Call_EventArgs													
LumiSoft.Net.SIP.Message.SIP_SVGroupHFCollection<>						1	12	3					12
LumiSoft.Net.SIP.Message.SIP_t_RetryAfter						2			9		3		4
LumiSoft.Net.SIP.Message.SIP_t_AlertParam						2			3				2
LumiSoft.Net.SIP.Message.SIP_t_Replaces						2			21		6		2
LumiSoft.Net.SIP.Message.SIP_t_OptionTag						3							
LumiSoft.Net.SIP.Message.SIP_t_To						2			9	6	3		
LumiSoft.Net.SIP.Message.SIP_t_Info						2			9		3		2
LumiSoft.Net.SIP.Message.SIP_t_ViaParm						2			41		20		8

اعداد نمایش داده شده در بالا تعداد دفعاتی است که کلاس ها در درون خود کلاس دیگر را فراخوانی کرده اند.

## ابزار CSV2Bunch :

پس از اینکه ماتریس وابستگی ها را از مرحله قبلی به دست آوردیم، باید آن را به فرمت ورودی نرم افزار Bunch در بیاوریم. این امر به کمک ابزار CSV2Bunch صورت می گیرد.

فایل خروجی نرم افزار CSV2Bunch به شکل زیر است:

A B 3

C D 7

...

در مثال بالا کلاس A ۳ بار درون خود کلاس B را فراخوانی کرده، و کلاس C ۷ بار کلاس D را صدا زده است.

در شکل زیر بخشی از خروجی نرم افزار Bunch را می بینیم:

```
test_one_folder_2_ClassDependencyMatrix - Notepad
File Edit Format View Help
Message.SIP_t_ContentCoding Message.SIP_ParseException 2
Message.SIP_t_ContentCoding Message.SIP_t_Value 3
Message.SIP_MVGroupHFCollection<> Message.SIP_MultiValueHF<> 19
Message.SIP_MVGroupHFCollection<> Message.SIP_HeaderFieldCollection 5
Message.SIP_MVGroupHFCollection<> Message.SIP_Message 14
Message.SIP_MVGroupHFCollection<> Message.SIP_t_Value 2
Message.SIP_MVGroupHFCollection<> Message.SIP_Message 14
Message.SIP_MVGroupHFCollection<> Message.SIP_HeaderField 3
Message.SIP_t_ReferredBy Message.SIP_ParameterCollection 3
Message.SIP_t_ReferredBy Message.SIP_t_ValueWithParams 9
Message.SIP_t_ReferredBy Message.SIP_t_Value 2
Message.SIP_t_ReferredBy Message.SIP_t_ValueWithParams 9
Message.SIP_t_ReferredBy Message.SIP_t_NameAddress 6
Message.SIP_t_ReferredBy Message.SIP_Parameter 3
Message.SIP_t_RValue Message.SIP_ParseException 4
Message.SIP_t_RValue Message.SIP_t_Value 3
Message.SIP_t_RAck Message.SIP_ParseException 10
Message.SIP_t_RAck Message.SIP_t_Value 3
Message.SIP_t_Challenge Message.SIP_ParseException 4
Message.SIP_t_Challenge Message.SIP_t_Value 3
Message.SIP_t_Timestamp Message.SIP_ParseException 2
Message.SIP_t_Timestamp Message.SIP_t_Value 3
Message.SIP_t_Credentials Message.SIP_ParseException 4
Message.SIP_t_Credentials Message.SIP_t_Value 3
Message.SIP_ParameterCollection Message.SIP_Parameter 12
Message.SIP_t_SecMechanism Message.SIP_ParameterCollection 13
Message.SIP_t_SecMechanism Message.SIP_t_ValueWithParams 29
Message.SIP_t_SecMechanism Message.SIP_ParseException 2
```

پسوند این فایل `.txt` می باشد که همانطور که گفته شد، ورودی نرم افزار `Bunch` است. که در بخش بعدی آن را بررسی خواهیم کرد.

البته ابزاری با نام `Graphviz` نیز وجود دارد که می توان به کمک آن فایل نمایش داده شده در بالا را در قالب مفهوم گراف مشاهده کرد و درک خوبی از ارتباط کلاس ها با یکدیگر به دست آورد.

## ابزار Bunch:

حال که ورودی برنامه Bunch آماده شده است، از این برنامه برای خوشه بندی و استخراج ارتباطات بهینه استفاده خواهد شد. می توان گفت اصلی ترین مرحله در تحلیل مدل ارتباطی کلاس ها را این ابزار انجام می دهد.

در این ابزار می توان عمل خوشه بندی (Clustering) را به انتخاب خود با یکی از الگوریتم های ژنتیک، تپه نوردی و یا الگوریتم جامع انجام داد. همچنین میتوان رابطه ی (فرمول) مورد استفاده را نیز انتخاب کرد. در مثال انجام شده، از الگوریتم تپه نوردی (Hill Climbing) و رابطه Basic MQ استفاده کرده ایم. خروجی این نرم افزار، یک فایل متنی با پسوند dot. می باشد. در زیر بخشی از خروجی را می بینید:

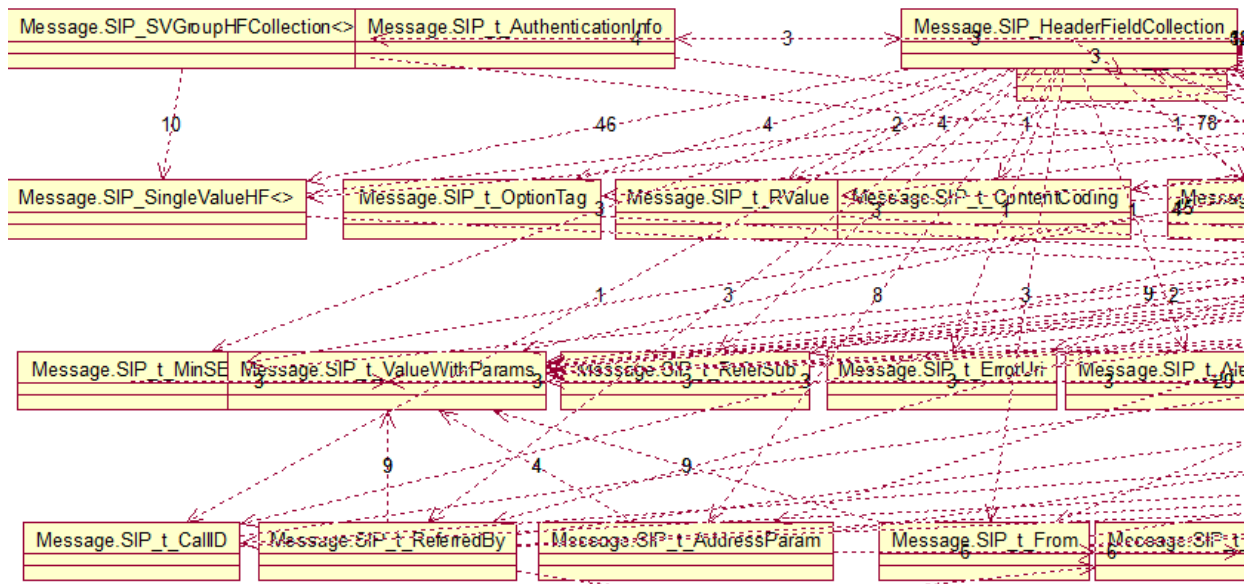
```
/* ----- */
/* created with bunch v3 */
/* Objective Function value = 1.0*/
/* ----- */

digraph G {
size= "10,10";
rotate = 90;
subgraph cluster0 {
label = "(SS-L1):Message.SIP_Message";
color = black;
style = bold;

"Message.SIP_SVGGroupHFCollection<>"[label="Message.SIP_SVGGroupHFCollection<>", shape=ellipse, color=lightblue, fontcolor=black, style=filled];
"Message.SIP_SingleValueHF<>"[label="Message.SIP_SingleValueHF<>", shape=ellipse, color=lightblue, fontcolor=black, style=filled];
"Message.SIP_t_OptionTag"[label="Message.SIP_t_OptionTag", shape=ellipse, color=lightblue, fontcolor=black, style=filled];
"Message.SIP_t_AuthenticationInfo"[label="Message.SIP_t_AuthenticationInfo", shape=ellipse, color=lightblue, fontcolor=black, style=filled];
"Message.SIP_t_RValue"[label="Message.SIP_t_RValue", shape=ellipse, color=lightblue, fontcolor=black, style=filled];
"Message.SIP_t_CSeq"[label="Message.SIP_t_CSeq", shape=ellipse, color=lightblue, fontcolor=black, style=filled];
"Message.SIP_t_ContentCoding"[label="Message.SIP_t_ContentCoding", shape=ellipse, color=lightblue, fontcolor=black, style=filled];
"Message.SIP_HeaderFieldCollection"[label="Message.SIP_HeaderFieldCollection", shape=ellipse, color=lightblue, fontcolor=black, style=filled];
"Message.SIP_t_Method"[label="Message.SIP_t_Method", shape=ellipse, color=lightblue, fontcolor=black, style=filled];
```

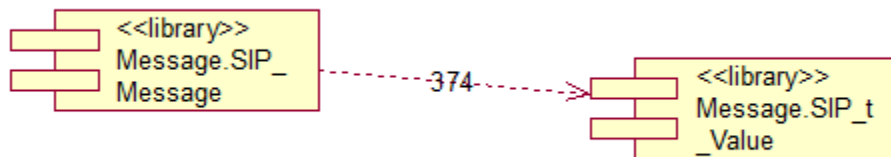
## ابزار Package View و Rational Rose

مرحله آخر، استفاده از ابزار Package View است. این ابزار فایل های آماده شده دو مرحله قبل (خروجی CSV2Bunch و Bunch) را به عنوان ورودی دریافت کرده و به عنوان خروجی، مدل استخراج شده را در محیط نرم افزار Rational Rose به نمایش می گذارد. در شکل های زیر خروجی های این ابزار را مشاهده می کنید:



تصویر بالا بخشی از مدل ارتباطی کلاس ها را نمایش می دهد. (Component Diagram Matrix)

تصویر زیر نحوه ی خوشه بندی کلاس ها را به نمایش می گذارد. (Component View Diagram)



همانطور که می بینیم خروجی نهایی این ابزارها به مدلی منجر شد که تمام کلاس های موجود در برنامه را بر اساس میزان وابستگی آن ها به یکدیگر، به دو خوشه تفکیک کرده است.