

What is lightweighting of common component?

Background and Purpose

As eGov Standard Framework is used widely and its applications are increased, demand to increase the efficiency of common component has been increased. Since the existing common component is operated through the eGov Standard Framework portal, the sole utilization as a component is difficult due to component dependency and effort for further analysis has been requested. For this, the plan to reduce the weight of Common Components is suggested.

Due to the component dependency, other components should be installed together. Therefore, the error can be prevented through by removing unnecessary components and revise related codes. If the relationship between components is simplified through lightweighting, the codes required for the function or component can be reduced and the effort to analyze and revise the error can be minimized.

What's new?

The newly changed items are as following:

- **Provision of complete distribution version:** Distribution file itself is made a complete entity to operate. The developer can operate the function by just downloading the distribution file and importing it into the eGov Standard Framework.
- **Minimized common distribution file:** The common distribution file in former versions does not exist. Through this, the unnecessary files are minimized.
- **Definition of relationship with tables:** The tables required to operate by distribution file are defined and the table generation script are provided separately.
- **Naming the context of the web application:** The UI provided to Common Components are revised and improved to work normally in naming the contents of the web application. (limited to 95 types)

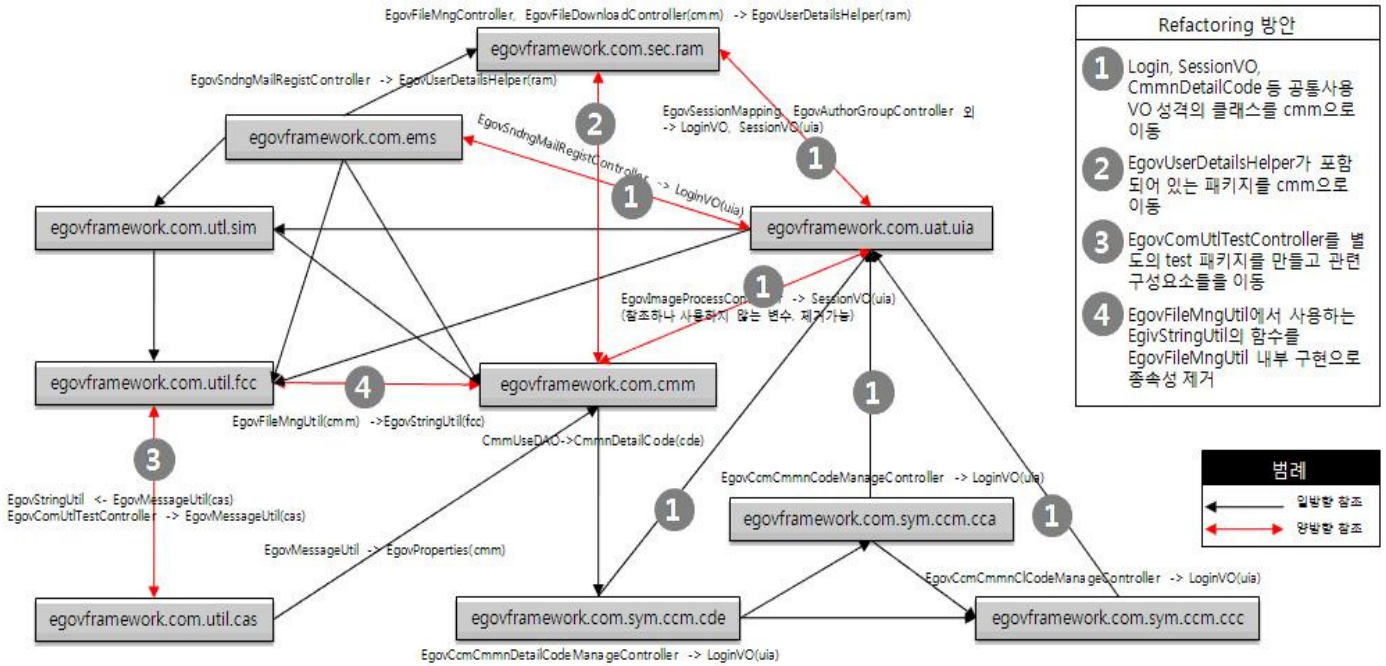
Method of lightweighting

Lightweighting of common distribution files and improving the certification related relationship

1) Lightweighting of common distribution files

The common distribution files of Common Component 1.0 were composed of 17 packages including the cmm. Since 10 packages of cmm, sec, ems, uat.uia, utl.fcc, utl.cas, utl.sim, sym.ccm.cde, sym.ccm.cca, and sym.ccm.ccc were inter-bounded complexly through functional dependency, all of 10 packages should be within the class pass when only one of them were required. Important causes of this are interpreted that the red lined packages were interrelated. Therefore, the interrelationship was re-factored to be single sided.

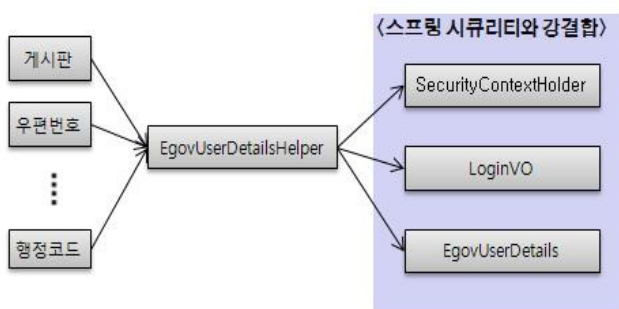
Most of packages have reference to the cmm package and it is written under common purpose of the Common Component. Some of classes are designed and realized as a structure to be allocated in each task package classified by the task or function. The problem is that some classes among these classes is referenced by almost all of classes in the common component. Therefore, when these classes are moved to the cmm package, the binding issue due to interrelation can be solved by moving these classes to the cmm package.



2) Improving the certification related relationship

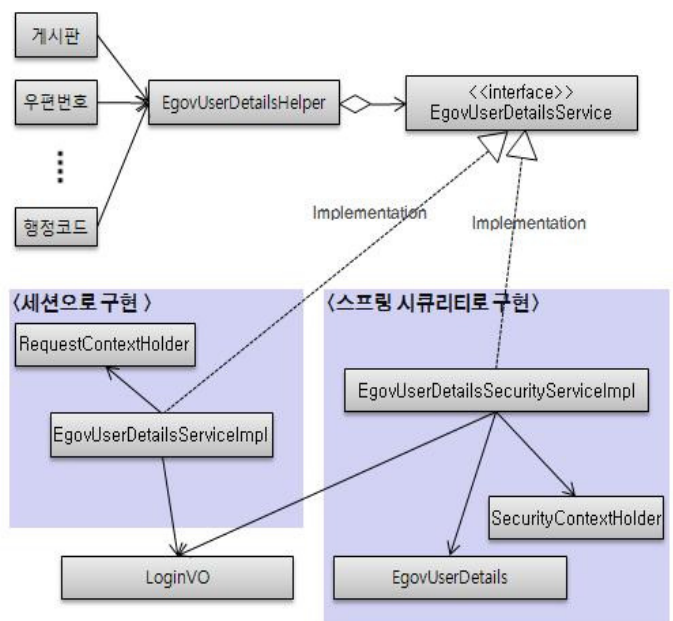
In 1.0, to use the Spring Security and confirm certification, the class called EgovUserDetailsHelper strongly integrated with SecurityContextHolder is used commonly. Although this class causes interrelation explained in 1), it is strongly integrated with the Spring Security in certification, the related packages should be used. In order to improve this, formerly designed EgovUserDetailsHelper class is changed to interface and implementation of this interface is improved to apply Spring Security or other methods and decoupled the relationship with the Spring Security. Developers using Common Component 2.0 can choose convenient methods which separated from unnecessary classes in the Spring Security. Moreover, by writing the class that implements the EgovUserDetailsHelper interface in integrating components from the Common Component 2.0 and certification of the sites and implementing the defined method, it can be integrated with the certification method without revising the Common Component.

인증 컴포넌트의 참조관계(1.0)



- 인증여부 확인 및 권한을 검사하는 Helper 객체가 Spring Security의 객체를 직접 호출하는 방식의 강결합 관계
- Spring Security는 권한관리 컴포넌트와 강결합되어 있어 단순한 인증에 의한 접근제어를 원하는 경우에 필요없는 권한관리 관련 컴포넌트도 함께 사용해야함

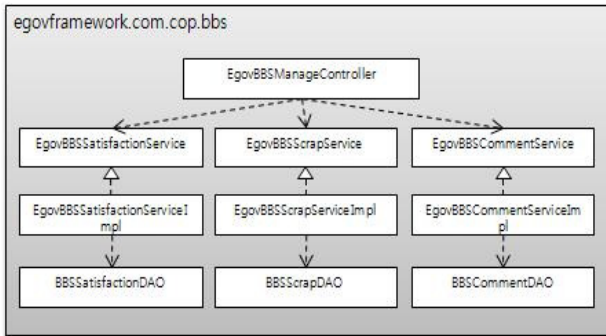
인증 컴포넌트 참조관계(2.0)



3) Improvement of relations between board components

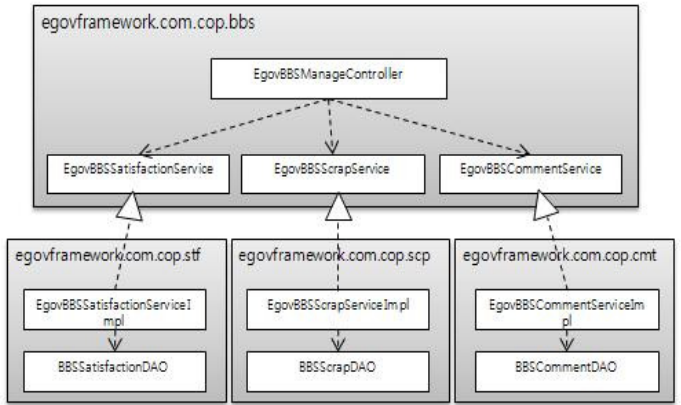
In case of board components, the reply, satisfaction research and scrap functions can be used selectively. However, in Common Component 1.0, implementation of these functions is interrelated which cannot be compiled only with the board package without the whole package. Therefore, the developers needed to remove related source files manually or include unnecessary functions. To improve on this, the classes related to reply, satisfaction research and scrap functions are separated in Common Component 2.0 and are defined as a separate package. The relations of these classes are improved to be used when it is necessary. (The reply, satisfaction research and scrap components should be installed with the board component due to the relations.)

게시판 관련 컴포넌트 (1.0)



- 업무 흐름이나 데이터 모델로는 댓글 등이 게시판을 참조하나 클래스 구조는 게시판(Controller)가 댓글 등(Service)을 참조하게 구성되어 있음
- 댓글, 만족도 조사, 스크랩과 같은 컴포넌트가 bbs 패키지 내에 함께 배치되어 있어 불용시 분리가 어려움
- Spring IoC에 의해 구현(Impl) 클래스가 인터페이스(Service)에 자동으로 "주입"되며 구현 클래스가 없는 경우 실행 오류가 발생하도록 되어 있음

게시판 관련 컴포넌트 (2.0)

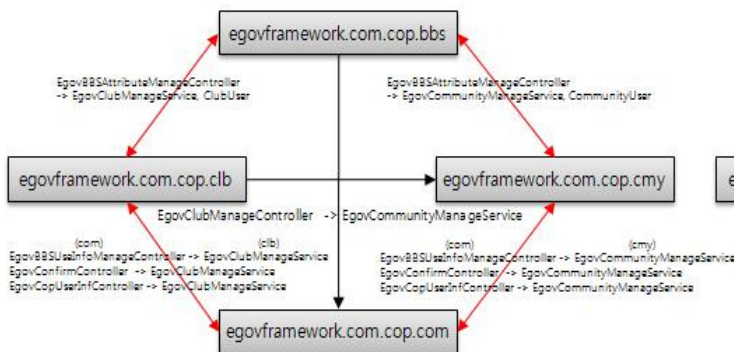


- 패키지를 분리함
- Autowiring을 통해 Impl이 없을 경우(댓글 등이 필요치 않을 경우) 객체가 생성되지 않으며 객체가 없으면 해당 로직을 수행하지 않도록 코드를 수정
- 결과로, 댓글, 만족도조사, 스크랩 컴포넌트가 분리 배포 가능

4) Improvement of relations of board and community components

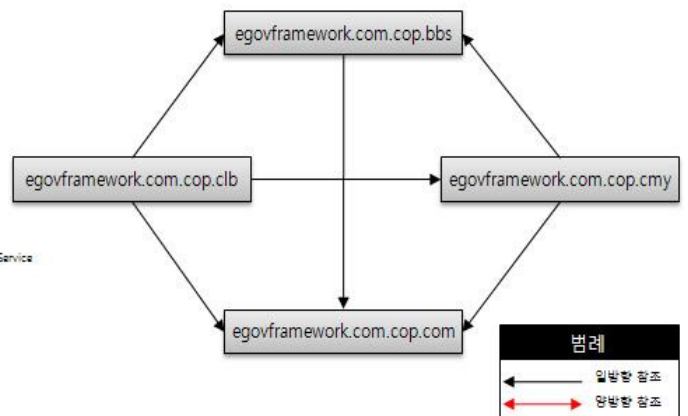
The board of Common Component 1.0 must have been installed with the community and society due to the interrelationship between board, community and society components. The relations are simplified by separating and moving the classes and the installation of the board component can be done separately.

게시판 관련 컴포넌트 (1.0)



- 컴포넌트 간 상호참조로 인하여 필요없는 컴포넌트도 함께 설치되어야 함
 - 게시판만 사용하고자 하는 경우에도 커뮤니티와 동호회를 모두 설치해야 함

게시판 관련 컴포넌트 (2.0)



- 필요한 컴포넌트를 최소화하여 사용할 수 있도록 참조 관계를 단순화
 - 게시판과 커뮤니티, 게시판과 동호회의 상호참조를 제거하여 게시판을 독립적으로 사용할 수 있도록 개선함

Improvement of package structure

The result of analyzing the relations between the package structure and component of the Common Component 1.0 showed that many number of components existed in one package (refer to the table below). In developing using these components, unnecessary classes should be used or those should be removed manually leaving the required classes in the package. To improve on these issues, the package was separated when many components are mapped within one package.

Component	No.	Package (Common Component 1.0)	Separated Package (Common Component 2.0)	Type
Certificate login General login Login policies	3	egovframework.com.uat.uia	egovframework.com.uat.uia egovframework.com.uat.uap (Login policy management)	Login's are in identical class, and login policy management is in separate class
Name verification Public I-PIN Name verification	2	egovframework.com.sec.rnc	Identical	Identical class
Authority management Authority group management Role management Department authority management	5	egovframework.com.sec.ram	egovframework.com.sec.ram (Authority management) egovframework.com.sec.rgm (Authority group management) egovframework.com.sec.gmt (Group management) egovframework.com.sec.rmt (Role management) egovframework.com.sec.drm (Department authority management)	Separate class
Board statistics User statistics Connection statistics Screen statistics Report statistics	5	egovframework.com.sts	egovframework.com.sts.bst (Board statistics) egovframework.com.sts.ust (User statistics) egovframework.com.sts.cst (Connection statistics) egovframework.com.sts.sst (Screen statistics) egovframework.com.sts.rst (Report statistics)	Separate class
Board Board generation management Notice management Design template Effective board Anonymity board Library management Reply management Scrap Satisfaction research	10	egovframework.com.cop.bbs	egovframework.com.cop.bbs(Board) egovframework.com.cop.tpl(Design template) egovframework.com.cop.cmt(Reply management) egovframework.com.cop.scp(Scrap) egovframework.com.cop.stf(Satisfaction research)	Partially identical or separate
Community management Community guest book Community album	3	egovframework.com.cop.cmy	Identical	Guess book and album are identical class as the board

Corporate member management User management Member management Site Map	4	egovframework.com.uss.umt	Identical	Separate class
Search postal code Postal code management	2	egovframework.com.sym.ccm.zip	Identical	Identical class
Log management User log management Send / receive log management System history management Web log management Connection log management	6	egovframework.com.sym.log	egovframework.com.sym.log.lgm (Log management) egovframework.com.sym.log.ulg (User log management) egovframework.com.sym.log.tlg (Send / receive log management) egovframework.com.sym.log.slg (System history management) egovframework.com.sym.log.wlg (Web log management) egovframework.com.sym.log.clg (Connection log management)	Identical class
Menu management Menu generation management Shortcut menu management Program management	4	egovframework.com.sym.mpm	egovframework.com.sym.mnu.mpm (Menu management) egovframework.com.sym.mnu.mcm (Menu generation management) egovframework.com.sym.mnu.bmm (Shortcut menu management) egovframework.com.sym.prm (Program management)	Separate class

Some packages are separated to operate separately through package separation process. And in some packages are separated but should be distributed together since they are related. Statistics and log related packages are the former; and user management, menu management and authority management related components are the latter. Login related components maintain one package structure for certification login and general login; whereas, the login policy management is separated. For board related components, although notice, effective board, anonymity board and library management are expressed to be functionally separated, the implementation is identical. Since it is separated by the property, the structural separation was not possible. Thus, with an exception for the templates, the board related components are remained as one package. For community related components, the community guest book and album is another name for board components and it does not exist as a code. Therefore, the community related package is remained as original. Since the community is implemented to include the board, it is designed to be distributed with the board.

Composition of lightweighted distribution file

The lightweighted component is provided to the user as a distribution file bound with the source code. These provided components must be operated without error in the development environment of eGov framework. For this, the relations between components are analyzed for the lightweighted components and the components are bound to compose the distribution file.

The reason for composing the distribution file is as following. The relations between the common components can be classified into functional dependency, UI or task relations and these relations cannot be removed completely by lightweighted. Therefore, without the components in functional dependency, error may occur in compiling and without the components in UI or task relations, error may occur in operation. Thus, to prevent the error, components in relations should be bound for distribution.

Additionally, the script to generate related tables and establish initial data is provided together to improve the degree of completion of the distribution file.

- [Refer to composition plan of the distribution file](#)