



# 전자정부 표준프레임워크 개요

－ 별첨 : 표준프레임워크 기반 개발방법소개



# 1. 표준프레임워크 기반 개발 - 구현시스템

Basic Board List

http://localhost:8080/sample/

List Sample

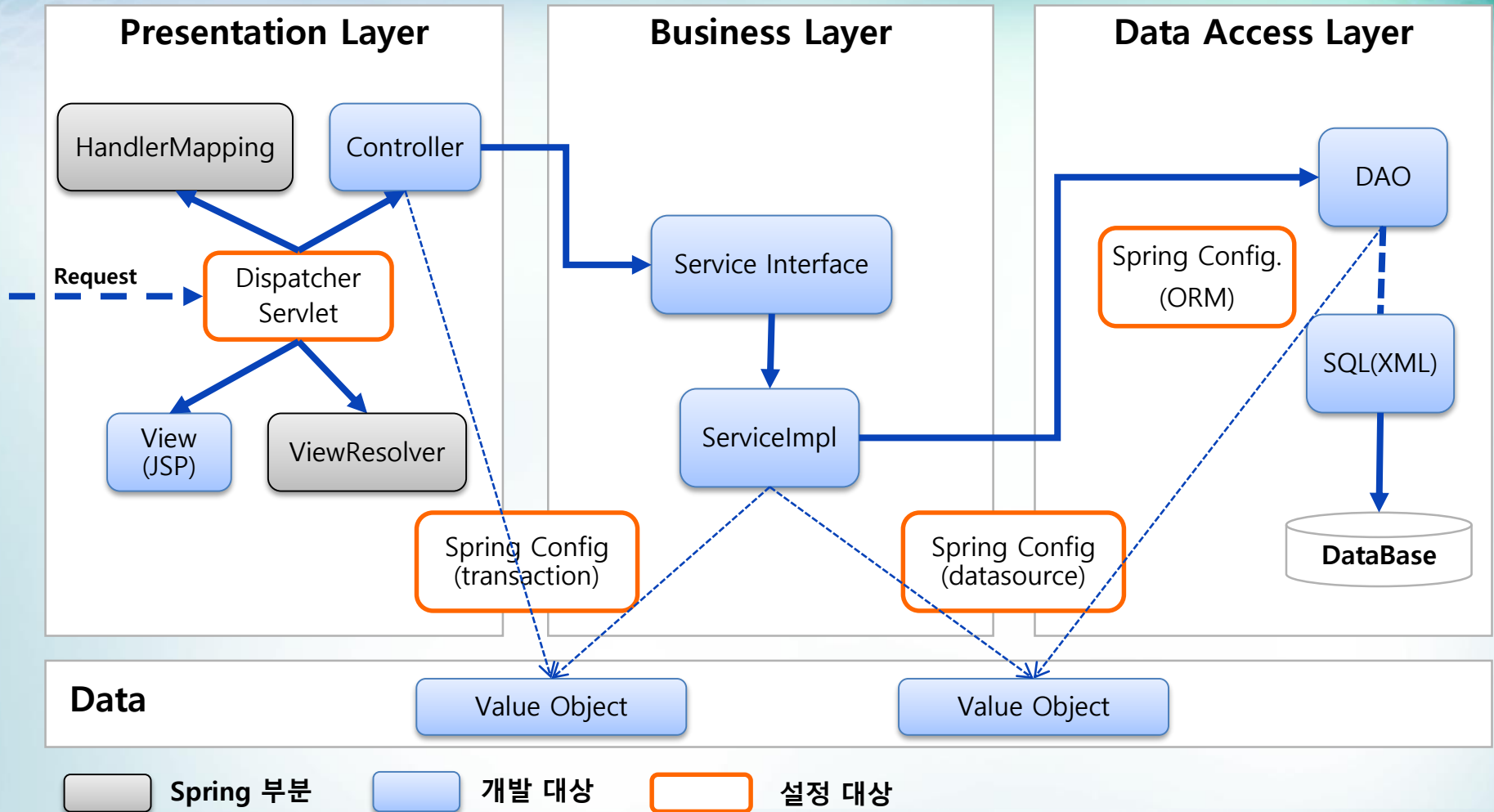
Name  검색

No	카테고리ID	카테고리명	사용여부	Description	등록자
1	SAMPLE-00004	Sample Test1	Y	This is initial test data,	test
2	SAMPLE-00003	test	Y	tset111222	qqqq
3	SAMPLE-00002	test Name	Y	test Desc	test
4	SAMPLE-00001	Sample Test	Y	This is initial test data,	test

1

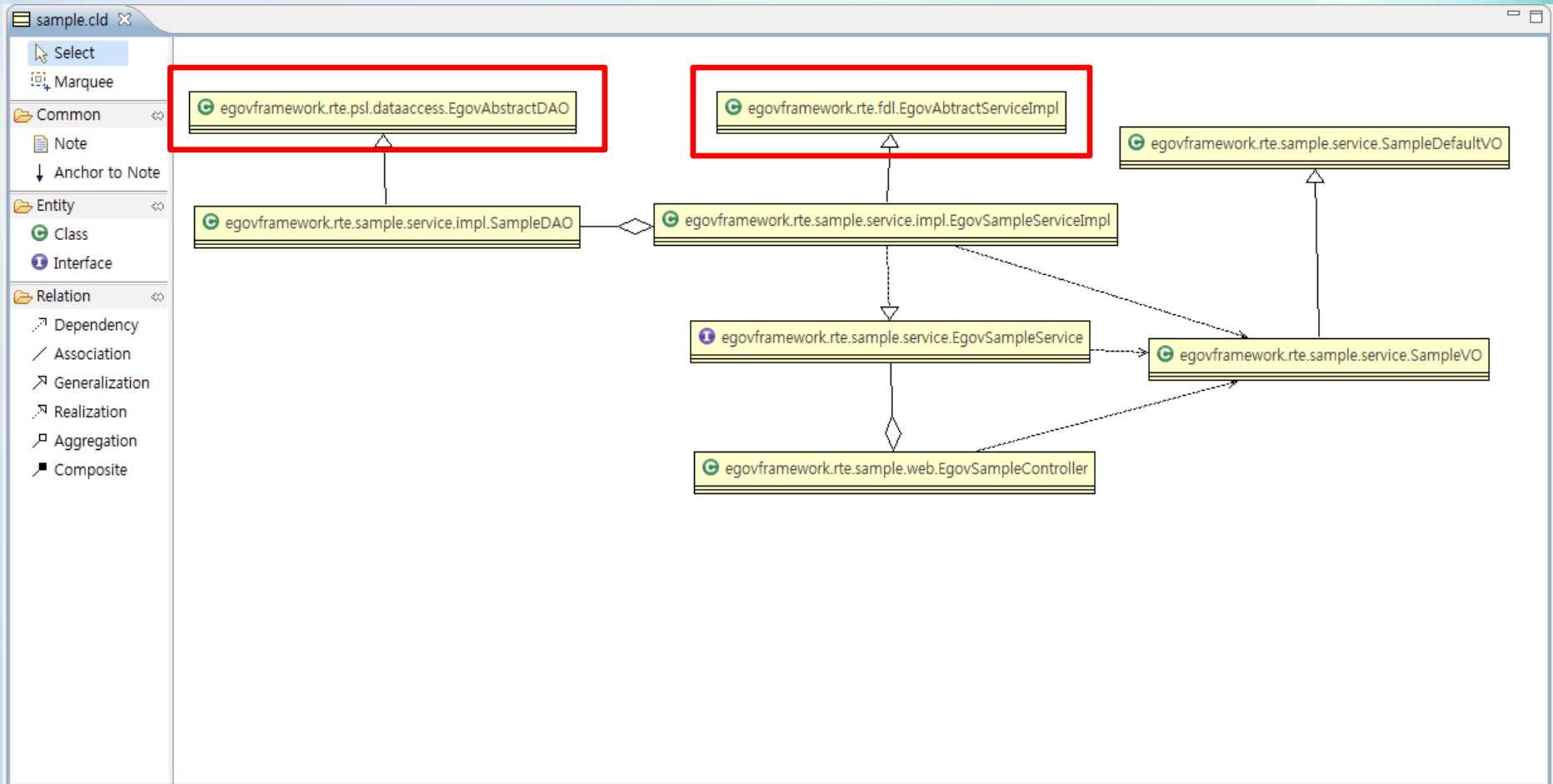
등록

# 1. 표준프레임워크 기반 개발 - 구현대상클래스



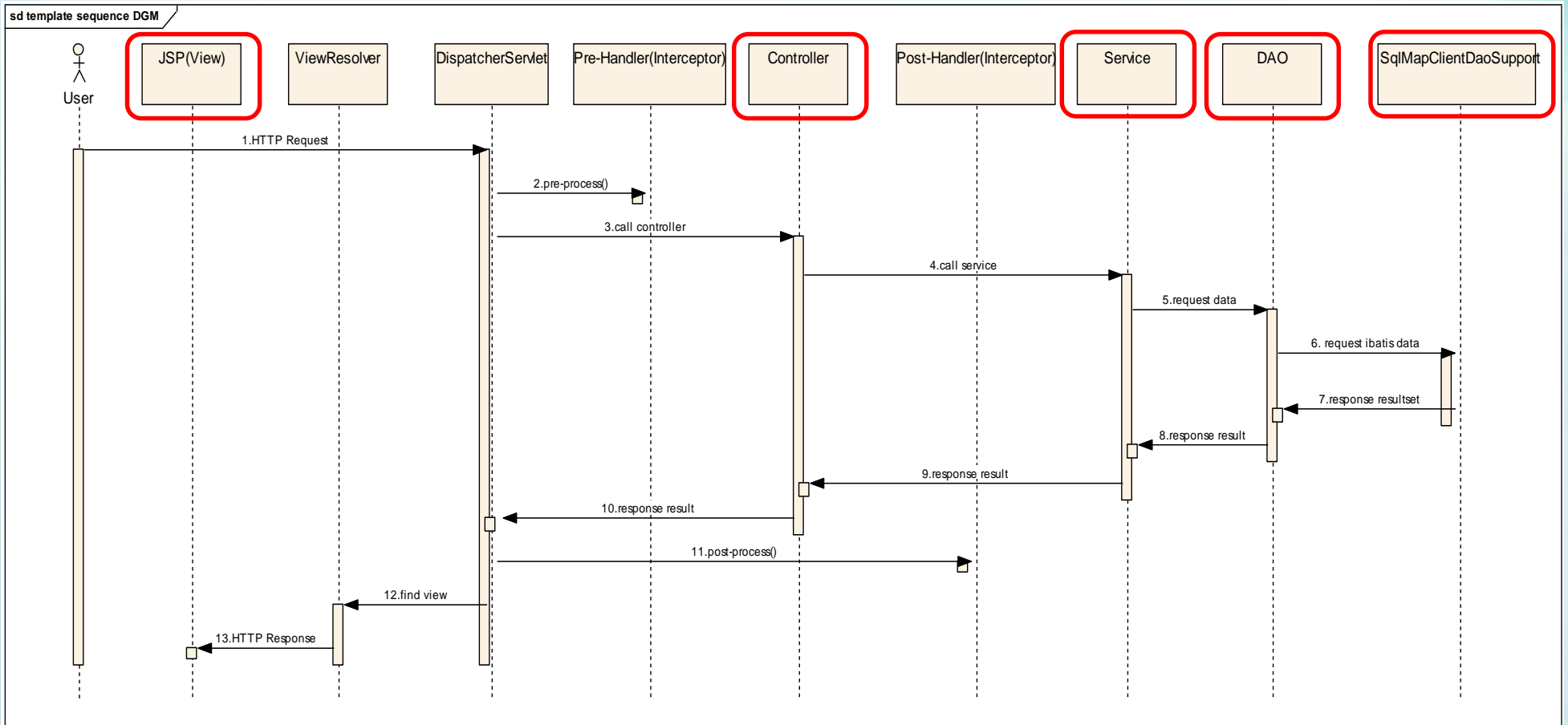
# 1. 표준프레임워크 기반 개발

## 실행환경에서 제공하는 아키텍처에 따른 클래스 설계



# 1. 표준프레임워크 기반 개발

실행환경에서 제공하는 어플리케이션 개발패턴에 따른 동적 설계 템플릿 및 개발 템플릿을 준수



# 1. 표준프레임워크 기반 개발

## Controller :

Request(요청)을 처리하고, 데이터 바인딩, 폼 처리 또는 멀티 액션 등의 기능 구현

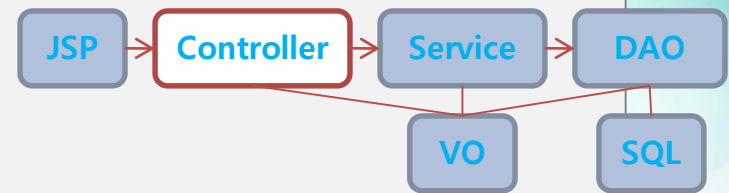
```
@Controller
@SessionAttributes(types=SampleVO.class)
public class EgovSampleController {

    /** SampleService */
    @Resource(name = "sampleService")
    private EgovSampleService sampleService;

    /**
     * 글 목록을 조회한다. (paging)
     * @param searchVO - SampleDefaultVO
     * @param model
     * @return "/sample/egovSampleList"
     * @exception Exception
     */
    @RequestMapping(value="/sample/egovSampleList.do")
    public String selectSampleList(@ModelAttribute("searchVO") SampleDefaultVO searchVO,
                                   ModelMap model) throws Exception {

        List sampleList = sampleService.selectSampleList(searchVO);
        model.addAttribute("resultList", sampleList);

        return "/sample/egovSampleList";
    }
}
```





# 1. 표준프레임워크 기반 개발

*Service* : 기능적 operation에 대한 서비스를 인터페이스로 정의

```
public interface EgovSampleService {
```

```
/**
```

```
* 글 목록을 조회한다.
```

```
* @param searchVO - 조회할 정보가 담긴 VO
```

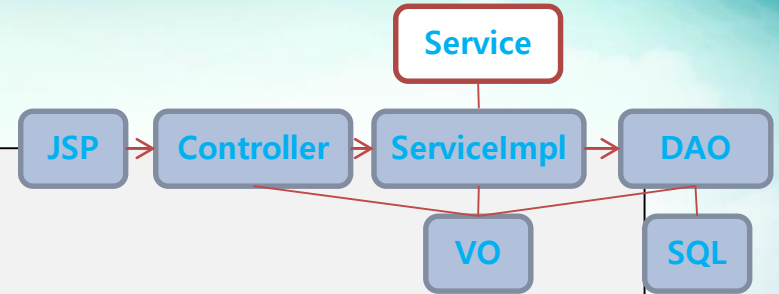
```
* @return 글 목록
```

```
* @exception Exception
```

```
*/
```

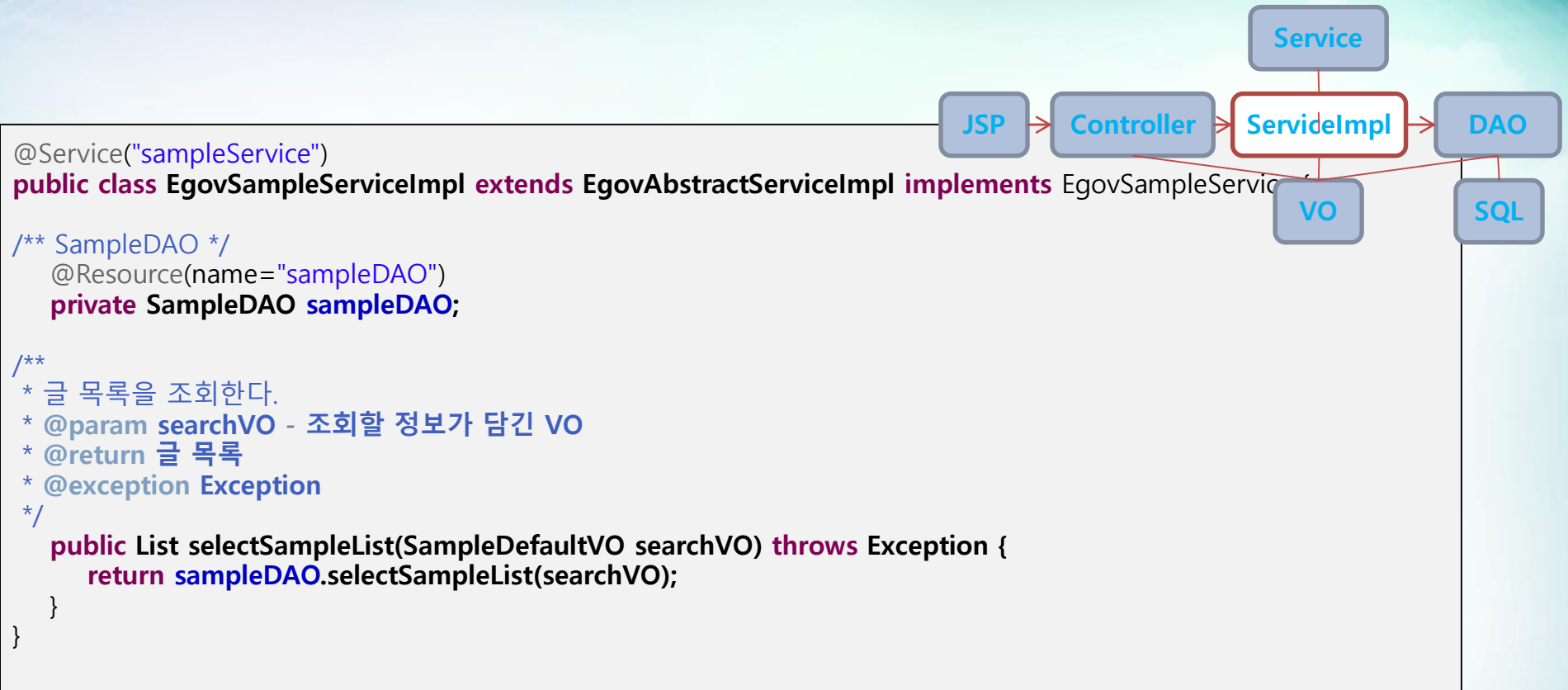
```
List selectSampleList(SampleDefaultVO searchVO) throws Exception;
```

```
}
```



# 1. 표준프레임워크 기반 개발

**ServiceImpl**: 서비스 인터페이스에서 정의한 operation의 상세 구현 클래스



※ Tip : ServiceImpl은 **EgovAbstractServiceImpl**(또는 **AbstractServiceImpl**)을 반드시 extends하여 작성한다.

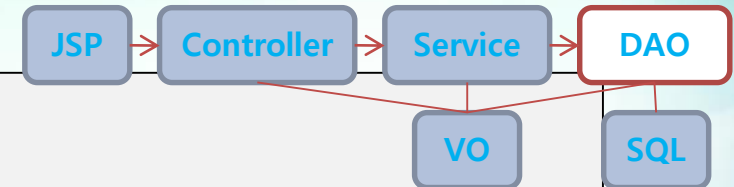


# 1. 표준프레임워크 기반 개발

## DAO : 데이터 처리 (iBatis 연동)

```
@Repository("sampleDAO")
public class SampleDAO extends EgovAbstractDAO {

    /**
     * 글을 조회한다.
     * @param vo - 조회할 정보가 담긴 SampleVO
     * @return 조회한 글
     * @exception Exception
     */
    public SampleVO selectSample(SampleVO vo) throws Exception {
        return (SampleVO) selectByPk("sampleDAO.selectSample_S", vo);
    }
}
```



※ Tip : DAO는 **EgovAbstractDAO**를 반드시 extends하여 작성한다.

### Method Summary

Object insert(String queryId, Object parameterObject);	Object selectByPk(queryId, Object parameterObject);
int update(String queryId, Object parameterObject);	List list(String queryId, Object parameterObject);
int delete(String queryId, Object parameterObject);	List listWithPaging(String queryId, Object parameterObject);

# 1. 표준프레임워크 기반 개발

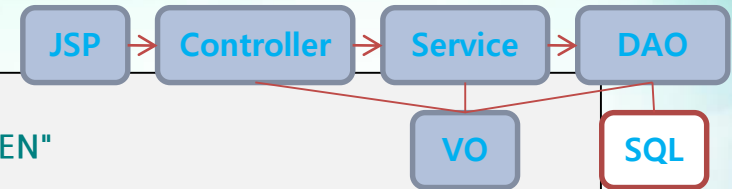
*iBatis* SQL Map Config : *iBatis* 연동을 위한 기본 메일 설정 파일로서 transaction 관리, sql mapping xml 파일 경로 설정 등을 포함

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE sqlMapConfig PUBLIC "-//iBATIS.com//DTD SQL Map Config 2.0//EN"
"http://www.ibatis.com/dtd/sql-map-config-2.dtd">

<sqlMapConfig>
  <settings useStatementNamespaces="false" cacheModelsEnabled="true"/>

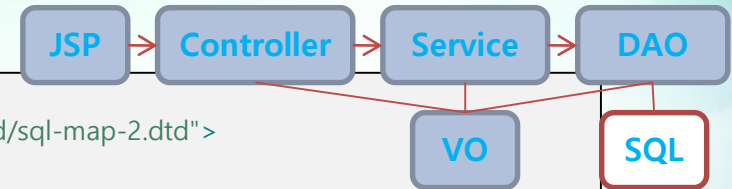
  <sqlMap resource="egovframework/sqlmap/rte/sample/EgovSample_Sample_SQL.xml"/>

  <!-- sqlmap 추가 -->
</sqlMapConfig>
```



# 1. 표준프레임워크 기반 개발

## *iBatis* SQL Map : 실행된 SQL문과 결과 mapping 정보 정의



```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE sqlMap PUBLIC "-//iBATIS.com//DTD SQL Map 2.0//EN" "http://www.ibatis.com/dtd/sql-map-2.dtd">

<sqlMap namespace="Sample">

  <typeAlias alias="egovMap" type="egovframework.rte.psl.dataaccess.util.EgovMap"/>
  <typeAlias alias="searchVO" type="egovframework.rte.sample.service.SampleDefaultVO"/>

  <resultMap id="sample" class="egovframework.rte.sample.service.SampleVO">
    <result property="id" column="id"/>
    <result property="name" column="name"/>
    <result property="description" column="description"/>
    <result property="useYn" column="use_yn"/>
    <result property="regUser" column="reg_user"/>
  </resultMap>

  <select id="sampleDAO.selectSampleList_D" parameterClass="searchVO" resultClass="egovMap">
    SELECT
      ID, NAME, DESCRIPTION, USE_YN, REG_USER
    FROM SAMPLE
    WHERE 1=1
    <isEqual prepend="AND" property="searchCondition" compareValue="0">
      ID = #searchKeyword#
    </isEqual>
    <isEqual prepend="AND" property="searchCondition" compareValue="1">
      NAME LIKE '%' || #searchKeyword# || '%'
    </isEqual>
    ORDER BY ID DESC
    LIMIT #recordCountPerPage# OFFSET #firstIndex#
  </select>

</sqlMap>
  
```

# 1. 표준프레임워크 기반 개발

VO(Value Object) : 클래스간 데이터를 전달한 객체

```
public class SampleVO extends SampleDefaultVO {
    private static final long serialVersionUID = 1753729060514530707L;

    /** 아이디 */
    private String id;

    /** 이름 */
    private String name;

    public String getId() {
        return id;
    }

    public void setId(String id) {
        this.id = id;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }
}
```

