

Mybatis 페이징 처리

Mybatis Rowbounds

Mybatis에서는 게시판 페이지 처리를 위한 객체인 Rowbounds 를 제공한다.

Rowbounds 객체 생성 방법

```
int offset = 100; // 뛰어 넘을 row 개수, 0 부터 시작  
int limit = 25;   // 읽어 올 row 개수  
  
RowBounds rowBounds = new RowBounds(offset, limit);  
  
session.selectList("Board.selectAll", null, rowBounds);
```

넘겨 줄 Parameter 객체가 없으므로
null을 기입하여 session을 작성한다.

게시판 페이지 계산 식 : Controller

```
int totalBoard = boardService.totalCount(); // 200 이라 가정한다

int pageLimit = 10; // 한 화면에 보여질 페이지 개수
int boardLimit = 10; // 한 페이지 내 최대 게시 글 수

int currentPage = 1; // 현재 페이지를 1이라 가정한다

int lastPage = (int)(Math.ceil((double)totalBoard / boardLimit));
// 마지막 페이지

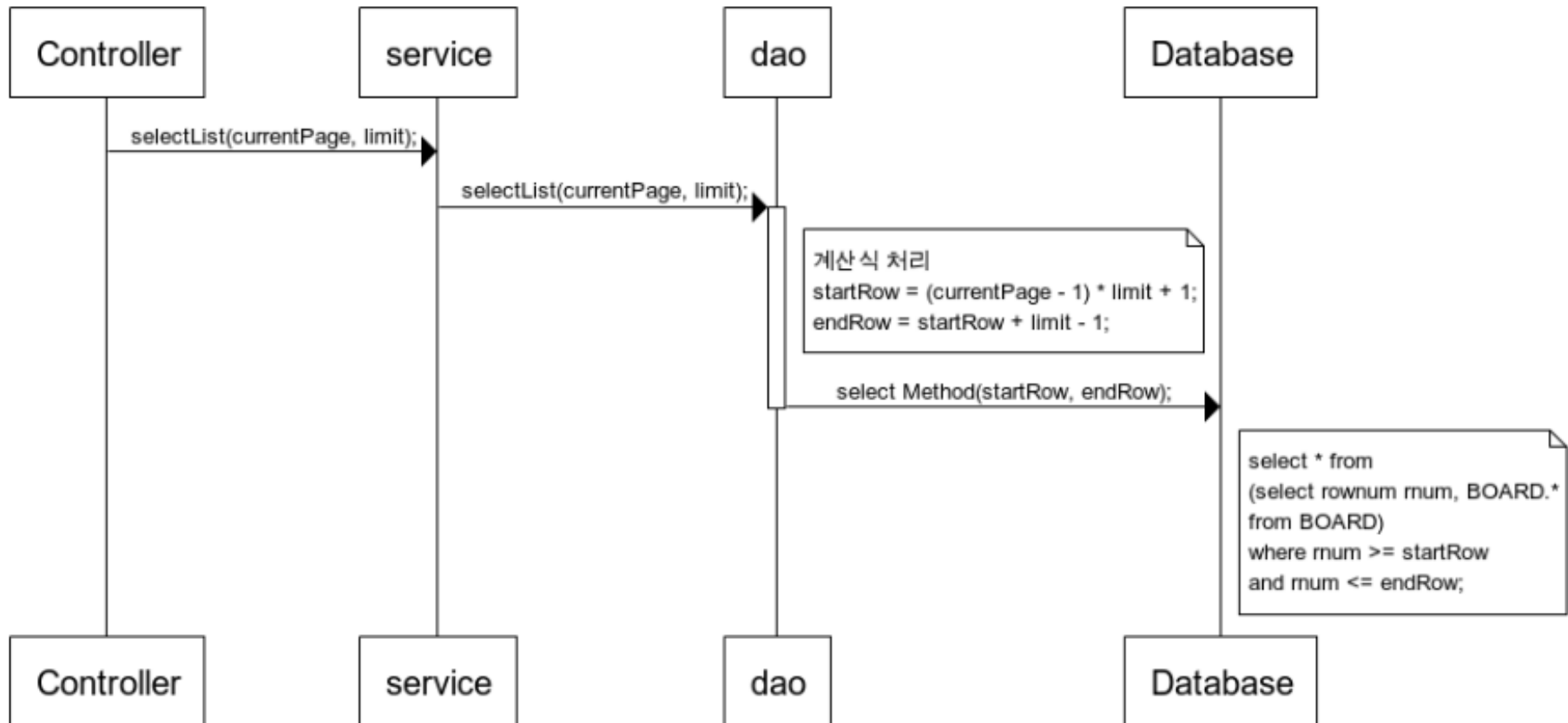
int startPage = ((currentPage-1)/pageLimit)*pageLimit + 1;
// 현재 보이는 처음 페이지

int endPage = startPage + pageLimit; // 현재 보이는 마지막 페이지

if(endPage > lastPage){
    endPage = lastPage;
}
```

Mybatis 페이징 처리

일반적인 페이지 시퀀스 다이어그램



일반적인 페이지 처리

SQL Query 구문 생성

```
select * from  
    (select rownum rnum, BOARD.*  
    from BOARD order by BOARD_DATE desc)  
where rnum >= ? and rnum <= ? ;
```

표시할 게시 글 순서 계산

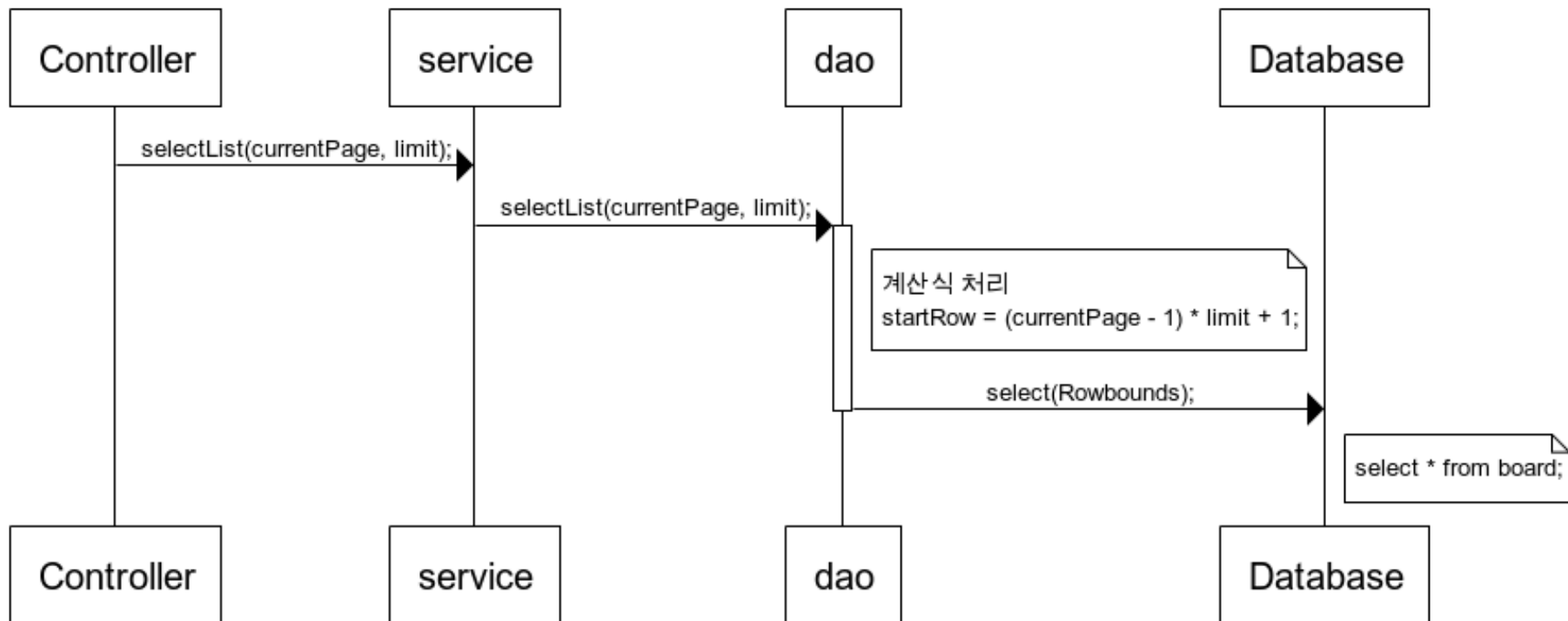
```
int startRow = (currentPage - 1) * limit + 1;  
int endRow = startRow + limit - 1;
```

구문 실행

```
pstmt = con.prepareStatement(sql);  
pstmt.setInt(1, startRow);  
pstmt.setInt(2, endRow);  
rset = pstmt.executeQuery();
```

Mybatis 페이징 처리

Mybatis 페이지 시퀀스 다이어그램



Mybatis 페이지 처리

mapper에 SQL Query 구문 생성

```
<select id="selectAll">  
  select * from BOARD;  
</select>
```

Dao 에서 Rowbounds 객체 생성

```
RowBounds rows = new RowBounds(((currentPage-1)*limit), limit);  
return new ArrayList<Board>  
    (sqlSession.selectList("Board.selectAll", null, rows));
```

ROWNUM VS Rowbounds

| | ROWNUM / ROW_NUMBER | Rowbounds |
|----|----------------------------------|--------------------------|
| 장점 | 대량의 데이터도 빠르게 페이징 처리를 하여 가져올 수 있다 | 구현이 쉽고, 코드의 유지보수가 간편하다 |
| 단점 | 페이징 처리를 구현하기 위한 코드가 복잡하다 | 대량의 데이터를 사용할 경우 수행속도가 느다 |