

[2010.10]

(주)챔프정보

실루엣 팀

[help@champit.co.kr](mailto:help@champit.co.kr)

<http://www.champit.co.kr>

## [모듈병합 매뉴얼]

[모듈병합은 실루엣을 사용하여 병렬개발 지원 기능에서 복수의 모듈을 병합하는

기능 매뉴얼입니다]

변경이력

버전	일자	내용	수정자
1.0	2010-10-25	초안작성	박병준

## 목 차

<b>제1부: 모듈병합 소개</b>	<b>4</b>
<b>제1장: 병렬개발과 모듈병합의 이해</b>	<b>4</b>
병합에 필요한 모듈의 구성	4
<b>제2장: 상태의 유형과 해석</b>	<b>6</b>
<b>제2부: 모듈병합 기능</b>	<b>8</b>
<b>제3장: Module 정보</b>	<b>9</b>
<b>제4장: 옵션</b>	<b>10</b>
<b>제5장: 검색결과 필터</b>	<b>11</b>
<b>제6장: 병합작업</b>	<b>11</b>
올림 / 내림	12
열기(병합)	12

## 제1부: 모듈병합 소개

모듈병합에 필요한 기본적인 조건 및 상태에 대한 설명을 소개합니다.

### 제1장: 병렬개발과 모듈병합의 이해

병렬개발이라는 용어는 하나의 소스를 여러 개발자가 동시에 개발하는 경우를 말합니다.

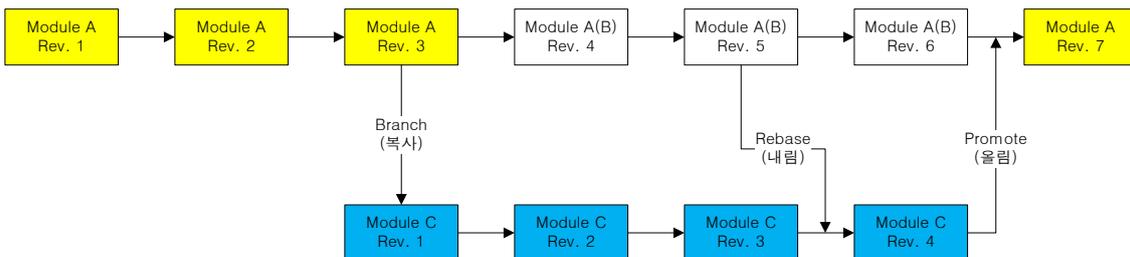
작게는 동일한 개발팀 내에서 일시적으로 하나의 소스를 2명의 개발자가 동시에 수정해야 하는 경우와 좀 크게는 운영중인 모듈(업무, 프로젝트) 전체를 별도의 개발팀을 조직해서 일정기간 동안 수정하는 경우입니다.

전자는 파일 하나의 Branch와 Merge의 관점이지만, 후자의 경우는 모듈 전체의 Branch와 Merge의 기능으로 사용됩니다.

본 매뉴얼에서 제공하는 모듈병합의 기능은 후자, 즉, 모듈 전체의 Branch와 Merge를 위한 기능입니다.

#### 병합에 필요한 모듈의 구성

모듈병합은 항상 3가지 모듈의 상태를 염두에 두고 개발되었습니다.



모듈상태 다이어그램

위 그림에서 하나의 모듈은 일정시점(Module A, Rev. 3)에 다른 모듈로 복사(Module C, Rev. 1)되어 병렬 개발되는 모습을 보여 줍니다.

이러한 상황은 웹 포탈 등에서 운영중인 서비스에 부가적인 하위 서비스를 SI 프로젝트로 추가하는 작업등에서 흔히 보여지고, 실루엣과 같은 Package S/W에서 하나의 중심버전(Mainline)을 기반으로 각 사이트마다 일정범위의 커스터마이징을 해서 납품하는 경우에도 발생하는 상황입니다.

#### A. 원본(Base)

원본은 정확하게 복제(Branch)를 생성하는 시점의 소스입니다. 위 다이어그램에 서는 Module A, Rev. 3이 되는 것입니다.

일반적인 경우 복사해서 생성한 Module C, Rev. 1의 경우에도 원본과 동일하겠지만, 필요에 따라서는 일부 파일만 복사 할 수도 있을 것입니다.

이 때 중요한 점은 복사본을 만든 시점, 즉, 원본(Base)은 시간상으로 그 의미를 가집니다.

#### B. 기준(Mainline)

기준은 복사본의 작업이 완료된 다음 다시 합쳐 저야 하는 소스입니다. 당연히 원본이 그 기준이 됩니다. 다만, 원본이 변화하지 않고 그 상태를 계속 유지한다면, (곰곰이 생각해 보면) 3가지 모듈의 상태가 성립하지 않습니다. ☺

그렇기 때문에, 복사본을 만든 이후에 작업되는 원본을 기준(Mainline)으로 부릅니다.

#### C. 작업(Branch)

작업은 복사된 작업본으로 작업이 완료된 다음 (일반적인 경우)없어져야 하는 소스입니다. 물론 Mainline과 Branch가 계속해서 지속되는 경우도 발생 합니다.

#### Tip 실루엣에서 Branch를 만드는 방법?

실루엣은 명시적으로 모듈의 Branch를 생성하는 메뉴를 제공하지 않습니다. 실루엣에서 가장 간단하게 모듈을 복제(Branch)하는 방법은 프로젝트 등록정보에서 프로젝트정보를 복사 한 다음, 작업영역에 있는 소스를 복사된 프로젝트에 추가하는 것입니다.

즉, 새로 프로젝트 하나 만들어서 최종 소스 전부 추가하는 것입니다.

## 제2장: 상태의 유형과 해석

상태는 3개지 모듈의 CRC32값을 비교한 값을 가지고 판단하는 것입니다

Flag는 3가지 유형의 값을 보여주는 것으로 0 = 없음, 1,2,3은 값이 동일 한지를 의미 합니다.

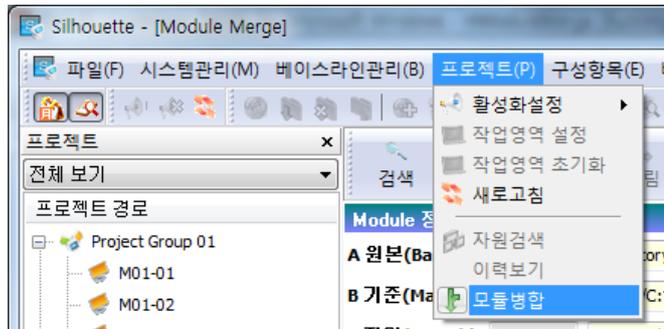
예를 들어서 1-1-2는 Base와 Mainline이 동일하고 Branch만 다르다는 것을 의미합니다.

상태	Flag	해석
동일	1-1-1	파일이 전혀 변경되지 않은 경우이다.
완료	0-1-1 1-0-0 1-2-2-	완료는 Base에 비교해서 병합작업이 완료된 경우이다. 실제 파일이 수정된 경우이기도 하지만, 파일이 동일하게 삭제되거나 추가된 경우도 포함된다.
올림	0-0-1 1-1-2	올림은 Branch에서 변경 혹은 추가된 것으로 필요하다면 Mainline에 반영 해야 하는 대상이다
내림	0-1-0 1-2-1	내림은 Mainline에서 변경 혹은 추가된 것으로 필요하다면 Branch에 반영 해야 하는 대상이다
검토		검토는 상황에 따라서 사용자의 판단이 필요한 경우이다.  소스의 내용을 판단하고, 파일의 생성시간을 참조 해서 처리방법을 판단 해야 한다.
	1-0-1	(일반적으로) Mainline에서 삭제된 파일이 Branch에 아직 남아 있는 경우일 가능성이 높다. Mainline의 삭제이력을 조사하자.
	1-0-2	Mainline에서는 삭제되었는데, Branch에서는 변경된 경우이다. Branch에서 계속 사용되고 있거나 그냥 변경 되었을 수 있다. Branch의 체크인 이력을 조사해서 누가 변경했는지 추적하자.
	1-1-0	(일반적으로) Branch를 생성 할 때 제외한 파일일 가능성이 높다. 물론 Branch에서 필요에 의해서 삭제 했을 수도 있다. Branch 생성시점의 이력을 조사하자.

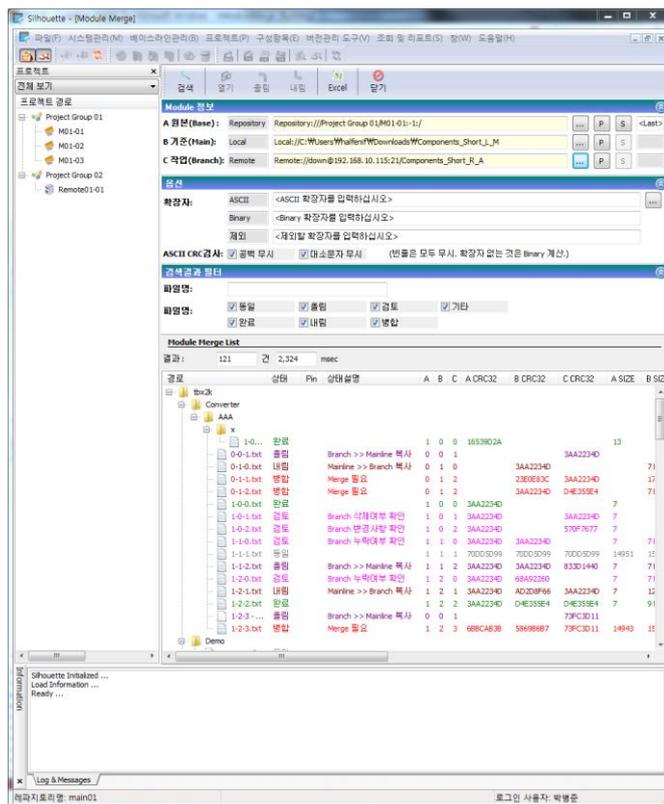
상태	Flag	해석
	1-2-0	Branch에는 포함되어 있지 않은데, Mainline에서는 수정되고 있는 경우이다. Branch 생성시점의 이력을 조사하자
병합	1-2-3	양쪽 모두에서 변경된 경우이다. 3WayDiff를 사용해서 어떤 내용이 변경되었는지 조사하자
기타	?-?-?	위 경우에 해당하는 않는 Case이다. 실루엣 개발팀에게 알려주자.

## 제2부: 모듈병합 기능

모듈병합 기능을 사용하기 위해서는 Workspace > 프로젝트 > 모듈병합을 실행한다.



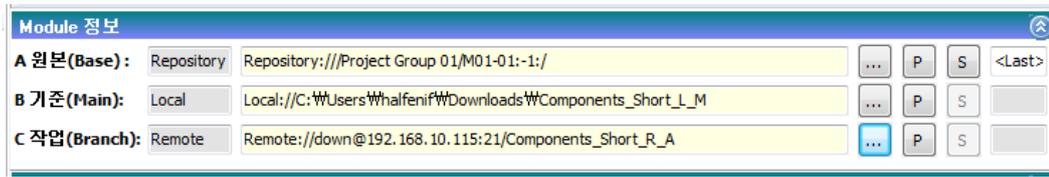
모듈병합 실행메뉴 (메뉴정보는 달라질 수 있음)



모듈병합 전체 화면

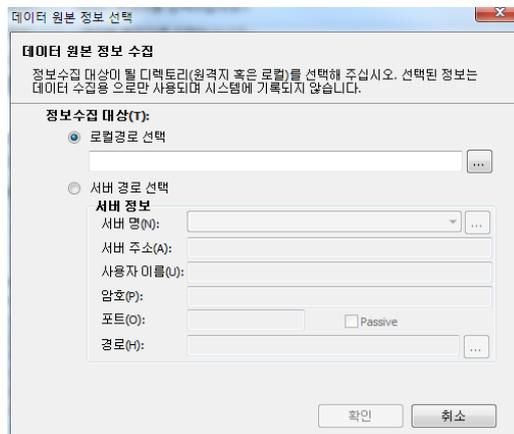
모듈병합은 3개의 조건과 1개의 결과 List로 구성됩니다. Module정보, 옵션은 검색을 하기 위한 조건이며, 검색결과 필터는 파일명 및 상태를 활용한 필터입니다.

### 제3장: Module 정보



병합의 대상이 되는 모듈정보는 3개의 대상(Base, Mainline, Branch)에 3개의 Source(Local, Remote, Repository)를 선택 할 수 있습니다.

[...] 버튼은 Local 및 Remote의 작업영역을 선택 할 수 있습니다.



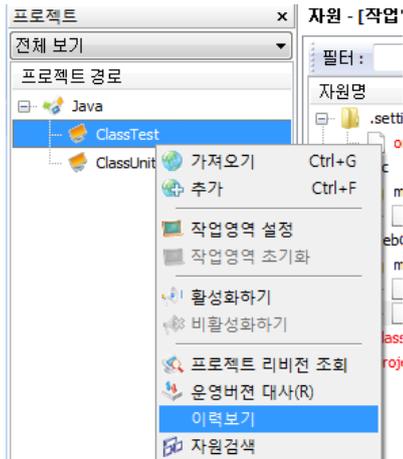
#### Local 및 Remote 선택

[P] 버튼은 자신이 멤버 권한이 있는 Repository의 프로젝트를 선택 할 수 있도록 합니다. 기본적으로 해당 모듈의 Root(/)와 최종 리비전이 선택됩니다.

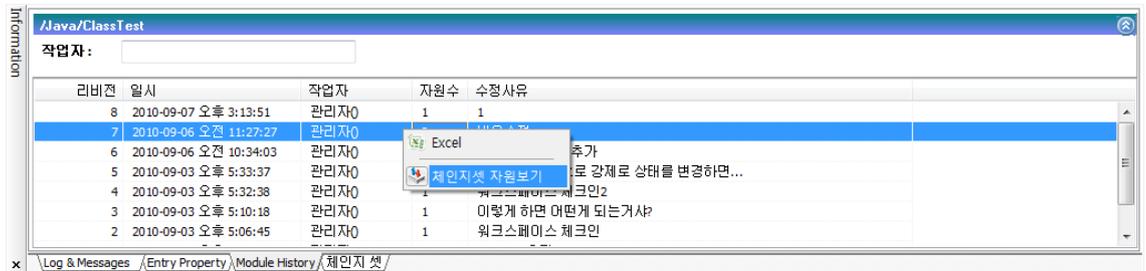
혹시 모듈 중에서 일부 하위 폴더에 대한 병합만을 조사하기 위해서는 [S]버튼을 눌러 하위 폴더를 지정 할 수 있습니다.

또한 마지막 [<Last>]라고 표시된 리비전 필드에 모듈 리비전을 입력 할 수 있습니다. 주로 Base Repository를 설정 하기 위해서는 Workspace에서 프로젝트 이력을 조사하여 모듈 리비전을 입력 해야 합니다.

모듈(프로젝트)의 이력을 조사하기 위해서는 아래 그림과 같이 조사 하고자 하는 프로젝트의 이력을 조사하여 해당 리비전을 알 수 있습니다.



프로젝트 이력보기 메뉴



프로젝트 이력 목록

**Tip Main과 Branch는 작업영역을 선택 하십시오!**

모듈병합의 3WayDiff, 올림 및 내림명령은 실제 작업영역의 파일을 변경 변경할 수 있도록 하지만, Repository에 직접 반영하는 기능을 제공하지는 않습니다.

즉, 작업영역에 병합(Merge)한 다음 실루엣 Repository에 체크인 해야 합니다. 그러기 위해서는 가급적 (다른 사정이 있는 것이 아니라면) Main과 Branch는 작업영역을 선택하는 것이 바람직합니다.

**제4장: 옵션**



옵션은 포함하거나 제외 할 확장자와 CRC검사를 위한 옵션을 설정 할 수 있습니다.

일반적으로 Binary에 대한 병합은 필요가 없기 때문에 ASCII 확장자에 자신이 검색하고자 하는 소스의 확장자를 입력해서 검색할 파일의 목록을 축소 할 수 있습니다.

(물론 대상 파일이 100여 개 내외의 작은 경우라면 그렇게 까지 설정 할 필요를 느끼지 못할 것입니다.)

확장자를 입력하는 포맷은 실루엣 유형정보에 등록하는 포맷과 동일하게 입력해야 합니다.

각 확장자는 '\*확장자' 형태로 입력되어야 하면 반드시 구분자로 ';'가 포함되어야 합니다.

예) \*.txt;\*.c;

[...] 버튼을 눌러서 관리자가 미리 설정한 확장자 정보를 사용 할 수도 있습니다.

ASCII CRC 검사 옵션은 파일의 CRC를 검사하기 위한 옵션을 제공합니다.

사용하는 개발환경이 공백 혹은 대소문자에 의한 비교가 필요하지 않은 경우라면 해당 옵션을 적용 할 수 있습니다.

**단, 이 옵션을 적용하기 위해서는 반드시 ASCII 확장자에 값이 등록된 경우만 적용됩니다.** ASCII 확장자에 아무것도 입력하지 않은 상태에서 해당 옵션만 체크되어 있다고 해서 적용되는 것은 아닙니다.

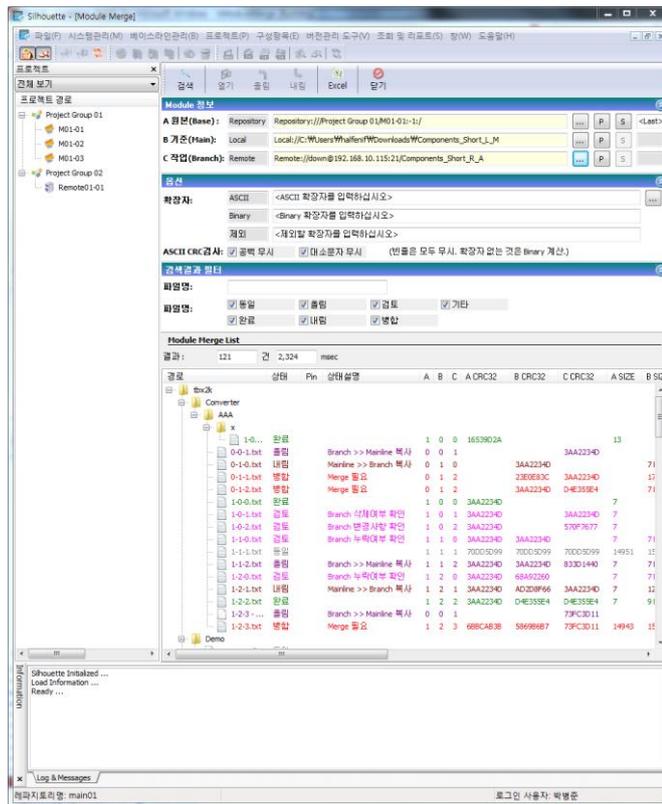
## 제5장: 검색결과 필터



검색결과 필터는 파일명과 비교상태를 사용하여 하단 출력목록을 필터링하는 기능입니다.

## 제6장: 병합작업

병합작업은 실제 작업영역에 있는 소스파일을 병합하거나 Mainline vs. Branch간의 파일을 복사하여 동기화 하는 올림 혹은 내림 기능입니다.



모듈병합 작업화면

### 올림 / 내림

올림 / 내림은 대상(Target)이 반드시 작업영역이어야 한다.

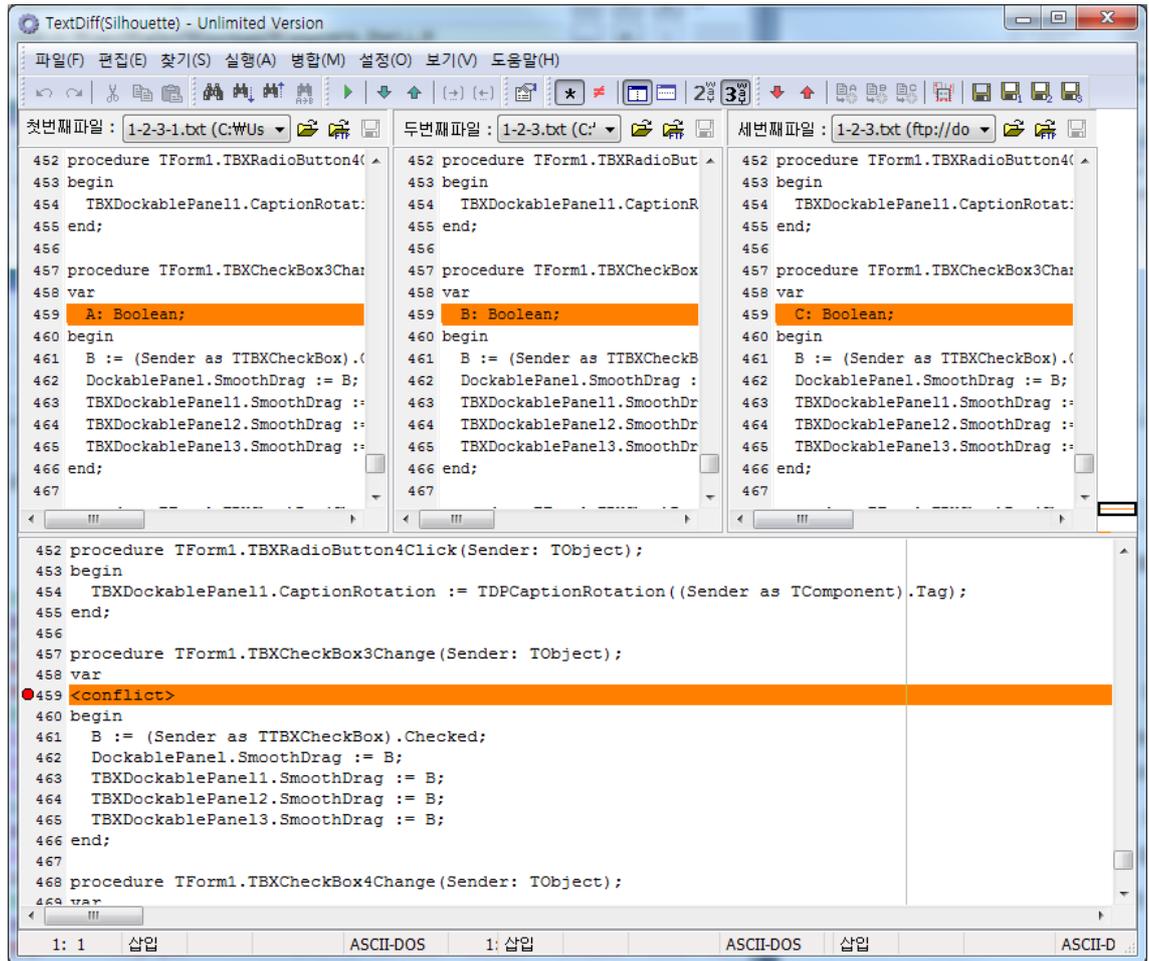
즉, 올림(Branch > Mainline)의 경우에 Target이 되는 Mainline은 반드시 Local 혹은 Remote인 경우에만 가능하다.

한번에 하나의 파일만 작업이 가능하며, 이러한 제약은 병합의 편리성과 안전성 측면에서 안전성을 강조하는 제약이다.

실제 올림 혹은 내림 명령을 실행하게 되면, Source의 파일을 Target 영역으로 복사한다.

### 열기(병합)

열기 기능은 소스를 병합하기 위한 목적으로 문자열 비교기를 사용하여 연다.



병합을 위해서 문자열 비교기가 실행 된 모습

파일이 2개만 있는 경우나 심지어 1개만 있는 경우에도 문자열 비교기를 사용하여  
 오픈 합니다.

문자열비교기는 원격지 열기 및 저장 기능을 제공하기 때문에, 모듈의 작업영역을  
 원격지로 설정 한 경우에 문자열비교기에서 수정 한 후 직접 저장 할 수 있습니다.

주) 문자열 비교기의 자세한 기능은 해당 매뉴얼을 참조하시기 바랍니다.