**MYKey**

**(Make Your Keyboard)**

**XML & Java  
Generator**

**Kim Tae-wan,**

**Dept. of Computer Engineering,**

**Hansung Univ**

1. 개요

1.1 XML & JAVA 생성기(XML & JAVA Generator) 란?

MYKey 프로젝트는 사용자가 UI 도구를 이용하여 안드로이드 소프트 키보드를 만드는 과정에서

전문적인 지식이 없이도 자신만의 소프트 키보드를 생성할 수 있음을 목표로 한다.

그렇기 때문에 MYKey는 사용자가 디자인 하는 것 외에 소프트 키보드의 레이아웃을 구성하는

XML과 한글 입력 오토마타와 연동되는 자바 파일 같은 기술적인 지식이 필요한 파일에 대하여

자동 생성 기능을 제공한다.

자동 생성은 사용자가 UI 도구를 이용하여 소프트 키보드 디자인을 완성한 경우 생성되는 파일,

즉, 레이아웃 테이블과 음소 테이블을 이용하여 진행되는데, 전문적으로 살펴보면 자료를 원하는

형태로 가공하는 파싱(Parsing)의 개념과 유사하나, 실체적으로 XML과 JAVA파일의 소스를

생성하는 코드 생성기(Code Generator)의 역할을 하기 때문에 이를 XML & JAVA 생성기로

정의하고 이어지는 각 절에서 MYKey프로젝트의 XML & JAVA 생성기에 대해서 소개한다.

2. XML & JAVA 생성기의 역할

MYKey에서 사용되는 생성 기술은 크게 2가지로 볼 수 있다.

첫 번째는, 소프트 키보드의 레이아웃을 구성하는 XML을 생성하는 것이며

두 번째는, 한글 오토마타에 문자 조합 정보를 알려 주는 자바 파일을 생성하는 것이다.

2.1 소프트 키보드의 레이아웃 구성

사용자가 UI 도구를 이용하여 자신만의 소프트 키보드를 구성하면 소프트 키보드의 레이아웃

정보를 담고 있는 문자 형식의 키 테이블 파일이 생성된다. 이를 안드로이드 프로젝트에

삽입되는 소프트 키보드 XML파일로 생성하기 위한 작업이 이루어진다.

2.2 한글 오토마타(Korean Automata)를 위한 정보제공

소프트 키보드를 이용한 영문 타이핑을 할 경우에는 해당 키의 코드 값(Code Value)만 있으면

되지만 한글의 경우에는 한글의 자음과 모음을 조합하는 기능을 하는 오토마타가 필요하다.

이 오토마타는 UI 도구에서 생성된 사용자가 정의한 문자 조합 정책 정보를 가진 음소 테이블의

정보를 이용해서 작동하기 때문에 음소 테이블의 정보를 가진 한글 오토마타와 결합할 수 있는

자바 파일(Java File)을 생성해야 한다.

3. XML & JAVA 생성기 구성

3.1 전체 구성

3.2.1 시스템 구조도

먼저, 사용자는 UI 도구를 이용하여 자신만의 소프트 키보드를 설계를 완료하면 UI도구에서

사용자가 디자인한 레이아웃 정보와 정의한 키 조합 정책을 테이블로 만들어 생성기로 전달한다.

다음으로, 생성기는 이 두 개의 테이블을 가지고 각각 XML과 자바 파일 생성을 진행하게 된다.

XML 생성은 사용자가 직접 UI 도구에서 디자인한 소프트 키보드의 레이아웃 정보를 갖고 있는

키 테이블을 이용하여 실제 안드로이드 디바이스에서 사용되는 소프트 키보드의 레이아웃을 위한

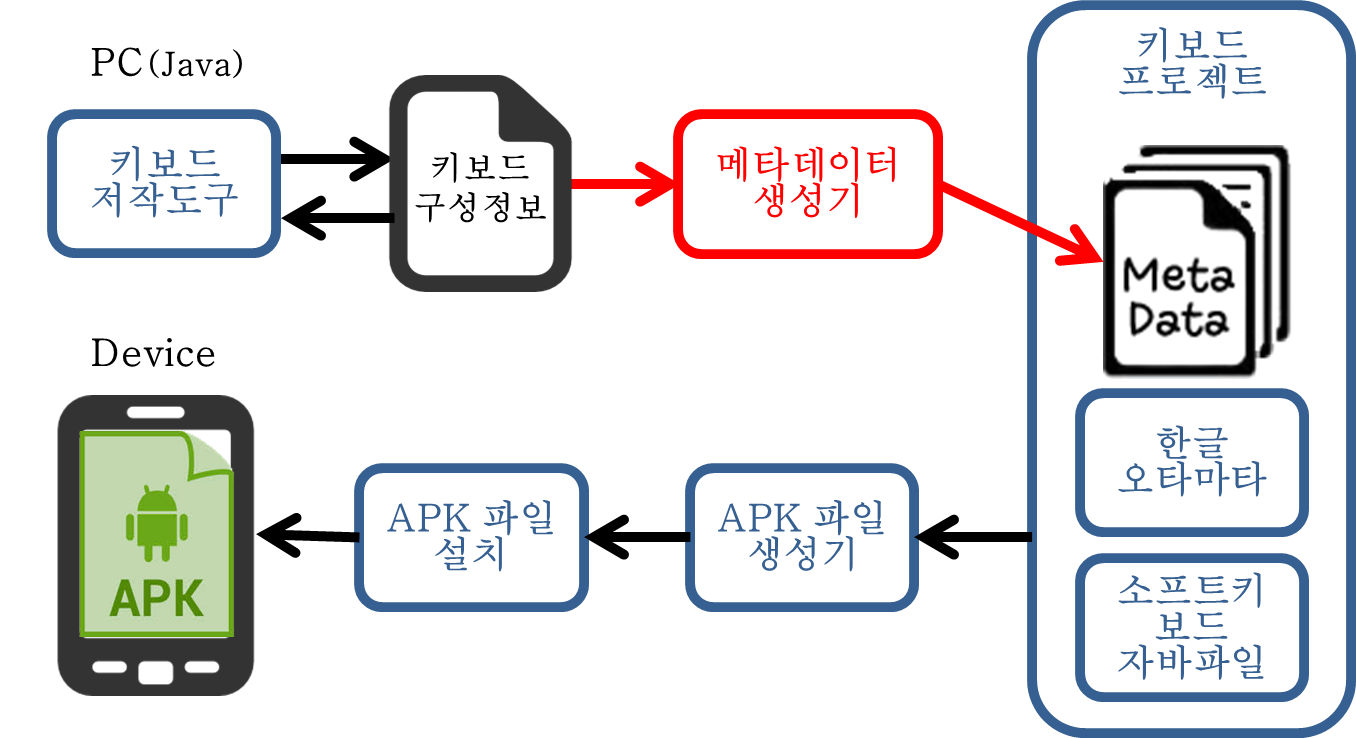
XML파일을 생성한다. JAVA파일 생성의 경우 사용자가 UI 도구에서 결정한 문자 조합 정책

정보를 가지는 음소 테이블을 이용하여 오토마타에게 조합 정보를 알려줄 수 있는 자바 파일을

생성한다. 이 두 개의 생성된 파일은 타겟 프로젝트에서 소프트 키보드의 레이아웃과 한글 문자

조합을 위한 오토마타의 작동을 위해 사용 된다.

MYKey에서의 생성기에 대한 구조도는 [ 그림 3-1 ]과 같다.



[ 그림 3-1 ] 프로젝트 내의 생성기의 구조도

위 [ 그림 3-1 ] 에서는 메타데이터 생성기로 표현되어 있으며, 저작도구와 분리되어

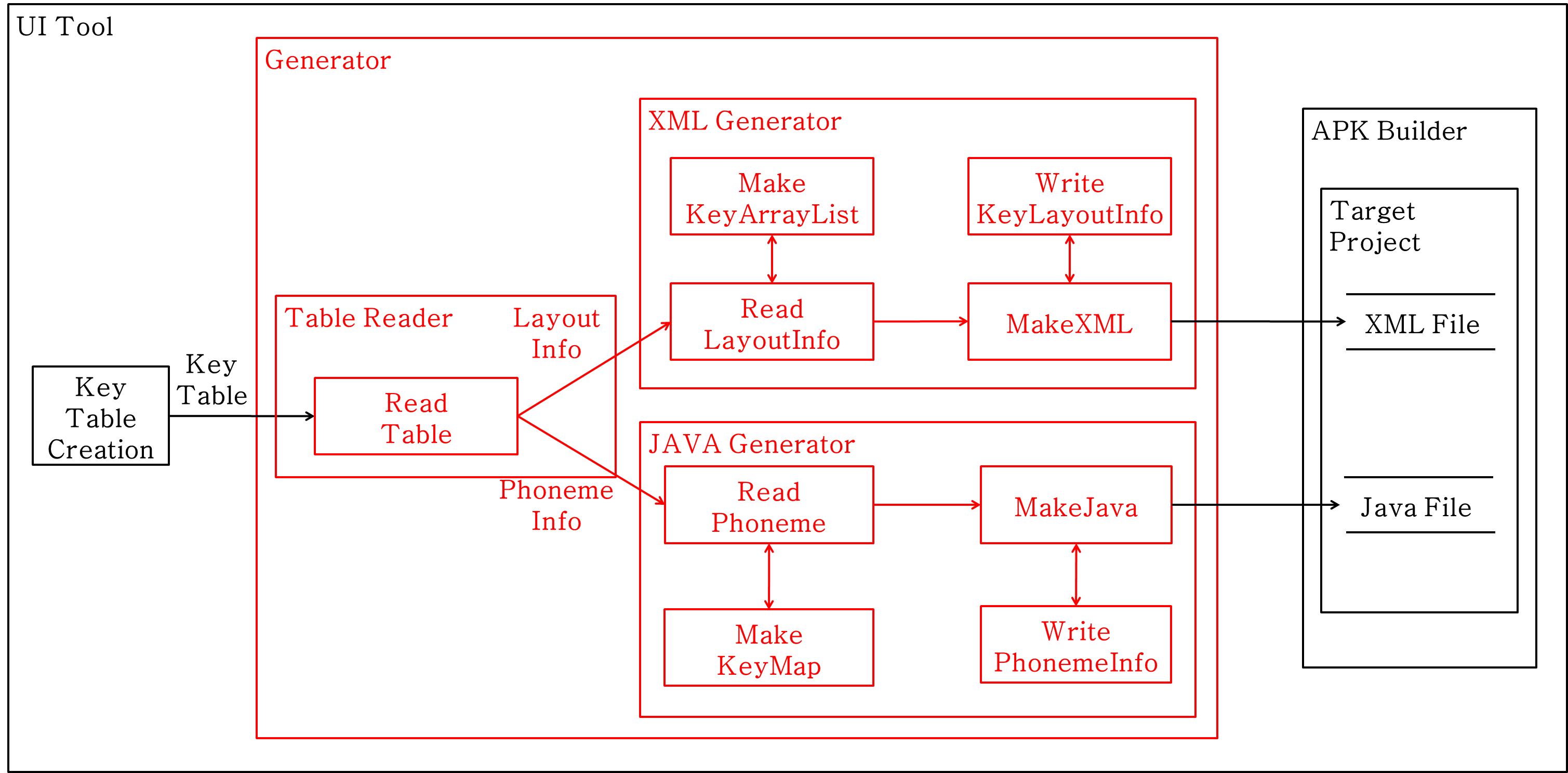
표기되어 있으나 실제로는 도구에 포함되어 있으며, 역할을 분담하기 위하여 분리하였다.

사용자가 디자인한 키보드 구성정보를 이용하여 메타데이터, 즉 자바 파일과 XML파일을

생성하는 것이다.

3.2.2 모듈 구조도

XML & JAVA Generator의 모듈 구조도는 [ 그림 3-2 ]와 같다.



[ 그림 3-2 ] 모듈 구조도

먼저, 저작 도구에서 키보드 생성을 마치면 레이아웃 정보와 음소 조합 정보를 가진 키 테이블

(Key Table)을 생성한다. 테이블을 읽어(Read Table) 레이아웃 정보와 음소 조합 정보를

각각 XML Generator와 JAVA Generator에 전달한다.

XML Generator는 레이아웃 정보를 받아 각 요소가 레이아웃 정보를 갖는 벡터를 생성하게

되는데, 여기서 벡터 자료구조를 사용한 이유는 각 요소를 인덱스로 접근할 수 있으며

단일 스레드 환경에서 최적의 효율을 내기 위해서 사용하였다. 모든 레이아웃 정보를 읽어

벡터 구성을 완료하면, XML파일을 생성하고, 모든 정보를 XML파일에 출력한다.

JAVA Generator는 음소 조합 정보를 받아 해시 맵을 생성하는데, Key 값은 소프트 키보드의

코드 값이 되며, 리턴되는 Value 값은 해당 음소의 음소 값이 된다.

(음소 값의 경우 오토마타 설계자가 미리 정의하였으며, 문서화되어 있음)

해시 맵을 완성하면, 자바 파일을 생성하고, 음소 조합 정보를 갖는 해시 맵의 코드를 파일에

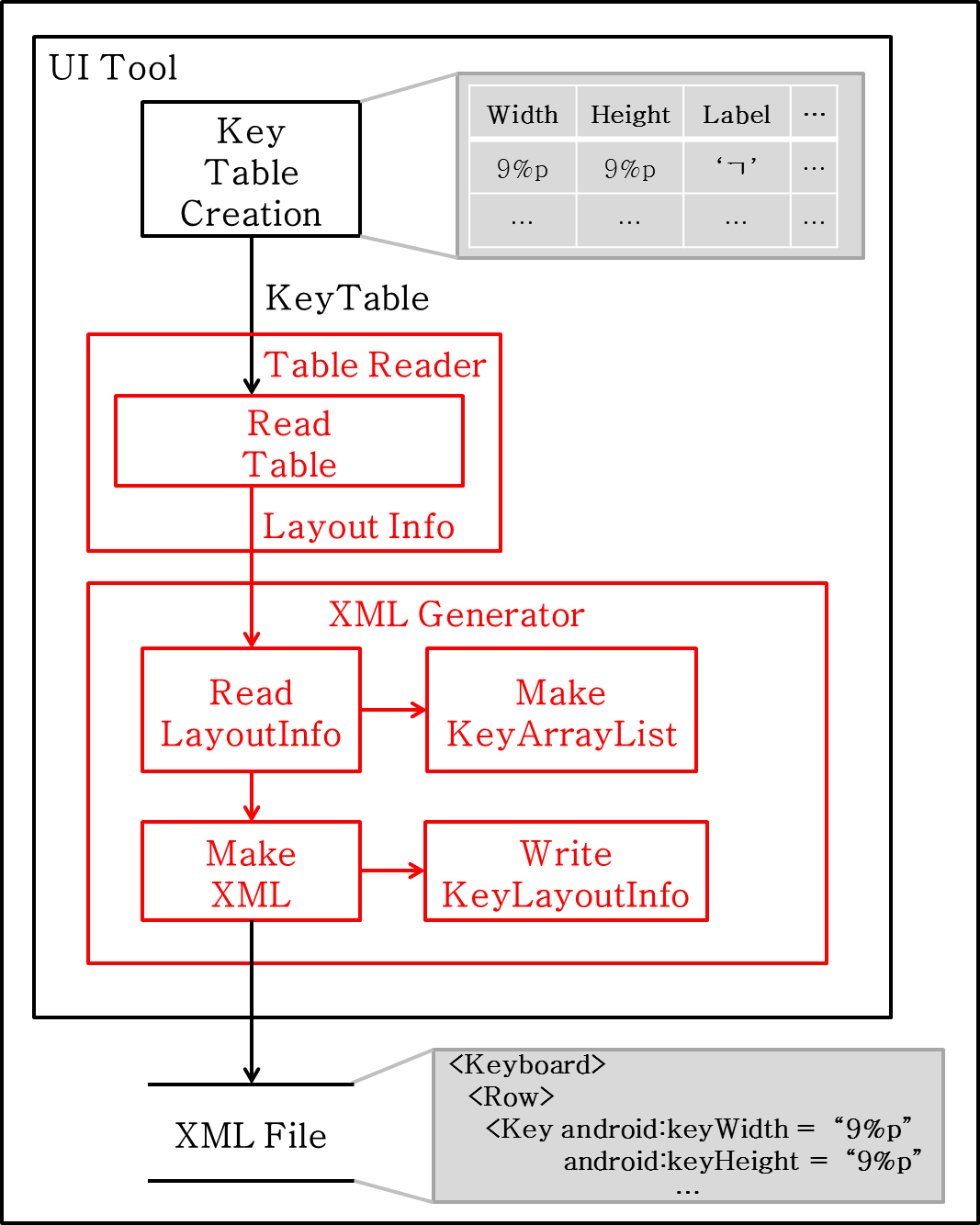
출력한다.

위 두 가지의 역할을 종료하면 생성기의 역할은 종료된다.

3.2 XML 생성기

3.2.1 전체 구조도

키보드의 레이아웃을 구성하는 XML 생성기의 구조도는 [ 그림 3-3 ] 와 같다.



[ 그림 3-3 ] XML 생성 구조도

XML 생성에 대한 설명은 앞서 전체 구조도에서 진행되었으므로 생략합니다.

3.2.2 키 클래스(Key Class)

사용자가 저작한 키보드의 각 키는 Key클래스의 객체화되어 관리가 된다. Key클래스는 키의

너비, 높이뿐만 아니라 아이콘, 특수 키 기능을 설정할 수 있는 멤버 변수를 가지고 있기 때문에

사용자가 직접 저작하는 키보드와 실제 안드로이드 디바이스에서 작동하는 소프트 키보드의

기능을 동일하게 구현할 수 있도록 해준다. Key클래스의 형태는 [ 그림 3-4] 와 같다.

**class Key {**

**int keyCode; 코드 값**

**int xPos; X좌표  
 int yPos; Y좌표**

**int keyWidth; 너비**

**int keyHeight; 높이**

**String keyLabel; 레이블**

**String keyIcon; 아이콘**

**int keyEdgeFlags; 엣지 설정**

**int isRepeatable; 반복 입력 여부**

**}**

[ 그림 3-4 ] Key클래스

3.2.3 키 테이블(Key Table)

사용자가 UI 도구를 이용하여 구성한 키보드는 키 테이블(Key Table)이라는 키보드 디자인

정보를 가지는 파일로 변환된다. 이 파일을 이용해서 실질적으로 디바이스에서 보여지는

소프트 키보드의 레이아웃을 설계하게 되는데, 이 키 테이블의 구성은 [ 표 3-1 ]과 같다.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **코드 값** | **너비** | **높이** | **좌표 x** | **좌표 y** | **레이블** | **아이콘** | **경계** | **반복여부** |
|  |  |  |  |  |  |  |  |  |

[ 표 3-1 ] Key Table

키 테이블에는 각 키의 설정 정보들이 모두 포함되어 있기 때문에 이를 바탕으로 사용자가

설계한 키보드의 레이아웃이 안드로이드 디바이스에서 동일하게 보여진다.

코드 값 : 안드로이드 소프트 키보드에서 적용될 각 키의 코드 값을 말한다.

이는 오토마타 설계자가 명시한 정의에 따라서 코드 값이 할당되어졌다.

너비 : 해당 키의 너비를 뜻한다. 실제 디바이스에서 적용되며, 전체 너비는 100%p이고

한 줄(Row)에 100%p를 초과할 수 없다.

높이 : 해당 키의 높이를 뜻한다. 실제 디바이스에서 적용되며, 전체 너비는 100%p이지만

대중화된 쿼티, 천지인 키보드의 전체 높이는 40%p ~ 50%p이다.

한 줄(Row)마다 높이를 정할 수 있으며, 모든 줄의 높이 합이 전체의 높이가 된다.

좌표 x : 각 키 사이에 공백을 넣기 위하여, 좌표 개념이 도입되었으며, 인접한 키의 x좌표와

너비를 비교하여 공백을 계산한다.

좌표 y : 각 줄(Row)의 공백을 넣기 위함이며, 인접한 줄(Row)과 높이를 비교하여 공백을

계산한다.

레이블 : 실제 디바이스에서 키에 표기되는 문자를 뜻한다.

아이콘 : 레이블 대신, 그림을 넣을 수 있다. 레이블과 아이콘 중 선택해야 한다.

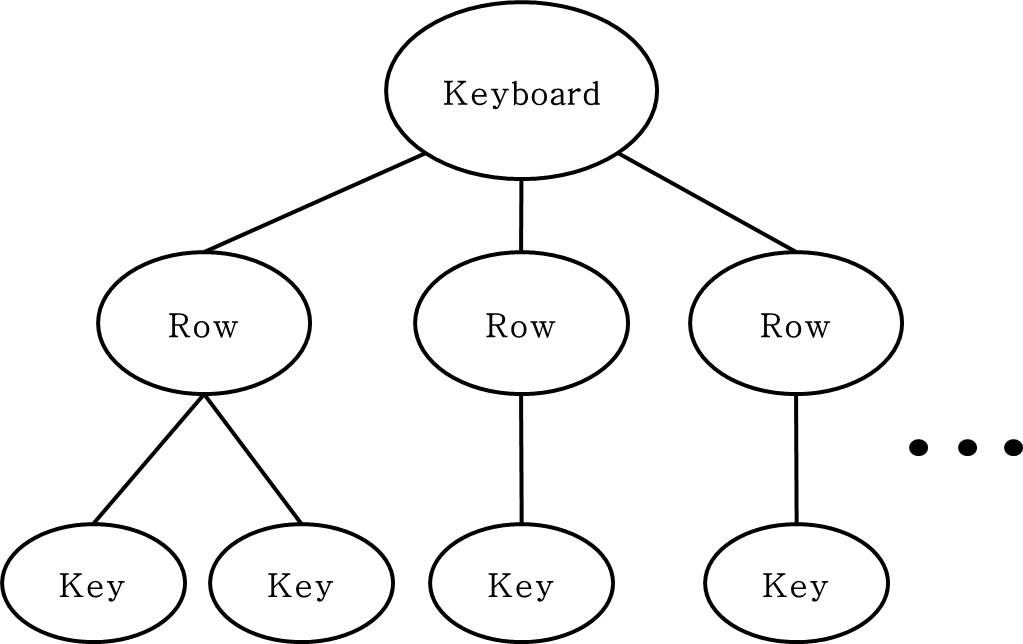
경계 : 좌측 또는 우측의 마지막 변임을 뜻한다.

3.2.4 키보드 클래스

키보드 클래스(Keyboard Class)의 경우 자바 코드가 아닌 XML을 이용하여 각 키의 모습을

디자인할 수 있는데 이 XML은 [ 그림 3-5 ] 처럼 최 상단 Keyboard에서 하위 노드 Row와

Key를 갖는 트리 형태의 구조로 되어있다.



[ 그림 3-5 ] Keyboard의 구조

또한 소프트 키보드의 레이아웃을 구성하는 Keyboard와 Row 그리고 Key에서 설정할 수 있는

속성들은 이어지는 [ 표 3-2 ]와 같다.

|  |  |  |
| --- | --- | --- |
| **구 분** | **속 성** | **기 능** |
| **Keyboard** | horizontalGap | 키 사이의 수평 간격 조정 |
| keyHeight | 키의 높이 조정 |
| keyWidth | 키의 너비 조정 |
| verticalGap | 키 사이의 수직 간격 조정 |
| **Row** | horizontalGap | 키의 수평 간격 조정 |
| keyHeight | 키의 높이 조정 |
| keyWidth | 키의 너비 조정 |
| keyboardMode | 키보드의 모드 설정 |
| rowEdgeFlags | 가장 끝을 표기 |
| verticalGap | 키의 수직 간격 조정 |
| **Key** | isRepeatable | 길게 누를 경우 반복 입력 설정 |
| isSticky | 토글 키 설정 |
| keyEdgeFlags | 가장 끝을 표기 |
| keyHeight | 키의 높이 조정 |
| keyIcon | 키에 보여지는 그림 설정 |
| keyLabel | 키에 보여지는 문자 설정 |
| keyOutputText | 키가 눌렸을 때 보이는 문자 설정 |
| keyWidth | 키의 너비 설정 |
| popupCharacters | 팝업 키보드에서 문자 표기 |
| popupKeyboard | 팝업키보드의 XML키보드 레이아웃 |

[ 표 3-2 ] keyboard의 xml속성 정의

실제로 사용되는 소프트 키보드에서 위의 모든 속성이 필요한 것이 아니기 때문에

XML 생성기의 경우 SoftKeyboard의 필수적인 속성을 이용하여 키보드를 구성하게 되지만

UI 도구에서 사용자가 설정한 방식대로 속성에 대한 정보가 주어지기 때문에 사용자가 어떻게

구성하느냐에 따라서 XML에서 사용되는 속성은 달라지게 된다.

3.2.5 키보드 뷰 클래스

키보드 뷰 클래스(KeyboardView Class)는 가상 키보드를 보여주는 역할을 하며 키의 렌더링과

키 입력 그리고 터치 행위를 감지하고 처리한다. 이 키보드 뷰 클래스를 통해서 보여지는 소프트

키보드의 키 색상, 레이블의 크기 등을 변경할 수 있으며, 키보드 뷰 클래스도 키보드 클래스와

마찬가지로 XML을 이용하여 디자인할 수 있지만 키보드 클래스와 다르게 키보드 뷰 클래스의

경우 [ 그림 3-3 ] 과 같이 inflate메소드(method)를 이용하여 res디렉터리의 하위 디렉터리인

layout에 존재하는 XML정보를 읽고 그 정보에 맞추어 객체를 생성한다.

***KeyboardView* = (KeyboardView) getLayoutInflater().inflate( *XML*, *ViewGroup*);**

[ 그림 3-6 ] KeyboardView객체 생성

키보드 뷰의 속성을 갖고 있는 *XML*을 이용하여 KeyboardView클래스를 상속 받은 클래스의

객체를 생성하는 것이다. 실제로 키보드 뷰의 setKeyboard메소드를 통해서 키보드의 한글-영문

또는 숫자 전환 시에 키보드의 모습을 바꾸어 준다.

위 [그림 3-3]의 inflate되는 XML에서 수정이 가능한 키보드 뷰의 속성은 [ 표 3-3 ]과 같다.

|  |  |  |
| --- | --- | --- |
| **KeyboardView** | keyBackground | 키의 이미지 설정 |
| keyPreviewLayout | 키 입력 피드백을 위한 레이아웃 리소스 |
| keyPreviewOffset | 키에서 키 입력 피드백의 수직 오프셋 |
| keyTextColor | 키의 문자 색상 |
| keyTextSize | 문자 키의 글자 크기 |
| labelTextSize | 문자열 키의 글자 크기 |
| popupLayout | 팝업 키보드를 위한 레이아웃 리소스 |
| verticalCorrection | 편파성 보정을 위한 Y좌표의 오프셋 |

[ 표 3-3 ] KeyboardView의 xml속성 정의

기본적으로 안드로이드에서 제공하는 Sample Keyboard의 경우에는 KeyboardView가

input.xml에 정의되어있다. 이 파일을 통하여 키 레이블의 크기 등을 수정할 수 있다.

3.3 JAVA 생성기

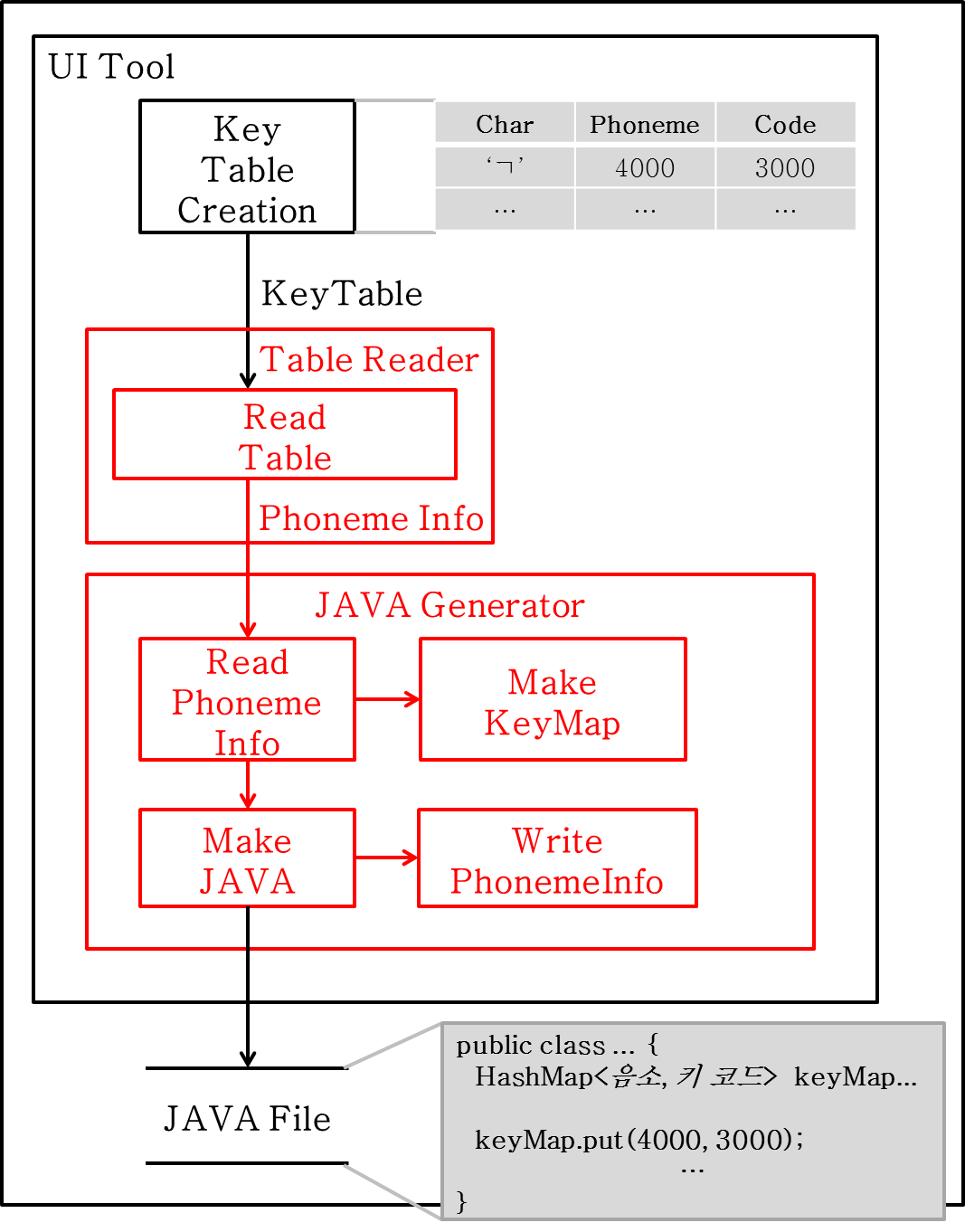
MYKey의 생성기는 앞선 [ 그림 3-2 ] 에서 소개한 것처럼 키보드의 레이아웃을 구성하는

XML뿐만 아니라 한글 오토마타에게 문자가 조합되는 정보를 전달하기 위해 필요한 Java파일을

생성하는 기능도 가지고 있다.

3.3.1 전체 구조도

문자의 조합 정보를 가지는 Java파일 생성의 구조도는 [ 그림 3-3 ] 과 같다.



[ 그림 3-7 ] JAVA 생성 구조도

먼저 UI도구의 Key Table Creation 과정에서 사용자가 지정한 문자의 조합 정책을 키 테이블에

저장하고 Read Table 과정에서 테이블을 읽어 음소 조합 정보를 Java Generator에 전달한다.

Java Generator는 음소 조합 정보를 이용하여 키 Key값을 키 코드 값, 리턴되는 Value값을

음소 코드 값으로 하는 해시맵을 생성한다. 여기서 해시맵을 사용한 이유는 입력되는 키의

코드 값에 대한 음소 값을 반환을 복잡도 O(1)으로 진행하기 위함이다.

최종적으로 해시맵의 정보를 이용하여 실제적으로 소프트 키보드에서 한글 조합 역할을 하는

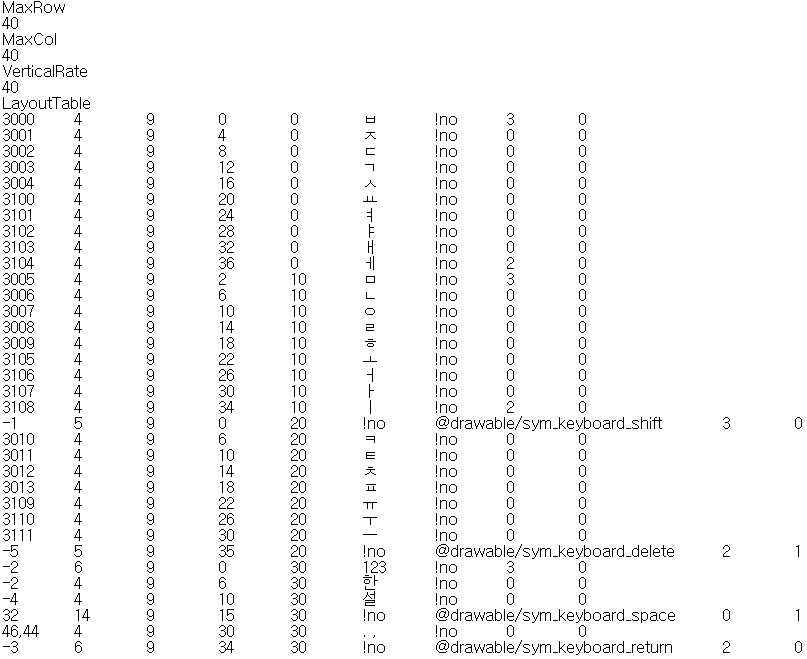
오토마타와 결합되는 자바 파일을 생성하게 된다.

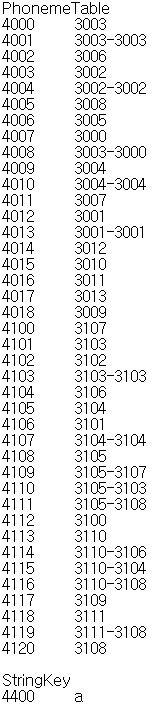
4. XML & JAVA 생성기의 개선 방향

4.1 Table Reader

UI 도구에서 디자인 된 키보드의 정보를 갖고 있는 키 테이블(Key Table)은 현재 3.2.3절처럼

정의되어 있는데, 자세히 살펴보면 [ 그림 4-1 ]과 같다.





[ 그림 4-1 ] Key Table의 실제 모습

MaxRow와 MaxCol은 키보드 저작 도구에서 드래그하여 키 버튼을 만드는 작업공간의 칸 수를

말하고, VerticalRate는 키보드의 최대 높이를 뜻한다. LayoutTable은 다음에 이어지는 내용이

키보드의 레이아웃 정보임을 알려주는 키워드고, PhonemeTable은 음소 조합 정보가 시작됨을

알려주는 키워드다. 마지막으로 StringKey같은 경우는 문자열 키의 정보를 뜻한다.

이처럼 현재 키 테이블은 임의로 정의되어 있다. 이러한 구성으로 소프트 키보드의 레이아웃을

구성하는 데 어려움이 없으나, 키보드의 색상, 글씨 크기 변경 등에 대한 내용 추가가 앞으로

키 테이블을 개선시켜야 하는 부분이다. 이는 Key Class의 정의와도 연관이 있다.

4.2 JAVA Generator

MYKey에서 Java Generator는 음소 조합 정보를 나타내는 자바 파일을 생성하는 역할이다.

오토마타가 이 파일을 참조하여 음소를 조합하게 되는데, 결과적으로 이 파일은 오토마타에

종속되어 있는 것이다. 오토마타가 수정되는 경우 이 파일의 정보도 수정되어야 한다.

4.3 XML Generator

현재 XML 생성기는 2개의 XML 생성을 목표로 하고 있다. 초기에는 1개 파일이었으나,

급박하게 하나의 파일을 더 생성하는 루프를 추가하게 되었다. 이 부분에 대해서는 수정하거나

또는 수정하지 않아도 전체적인 성능에 영향을 끼치지 않으나, 차후에 3개 이상의 XML파일을

생성하게 되는 경우(Shift키 등을 눌러 키보드 전환을 3번 하고 싶은 경우) XML Generator의

수정이 필요하게 된다.