

# 소프트웨어 공학개론

SA & SD

201011318 김슬기

201011334 박진성

© TemplateWise.com

# Original Statement of Purpose

- 위 프로그램은 C언어 소스코드를 받아들이며, 그 소스코드를 분석하고, 그것을 토대로 **Control Flow Graph**를 생성하는 알고리즘 (**CFG Construction Algorithm**)을 가동해 알맞은 **Control Flow Graph**를 그리는 것이 목표이다.
- **Control Flow Graph**를 생성하는 알고리즘(**CFG Construction Algorithm**)은 **Edge** 인식, 기본적인 **Block** 건설, 지연되는 것 해결, 데이터 충돌 해결로 총 4단계로 이루어지며 이는 블록과 **Edge**를 이용해 프로그램 제어구조를 그래프형태로 알맞게 그려내는 것을 목표로 한다.

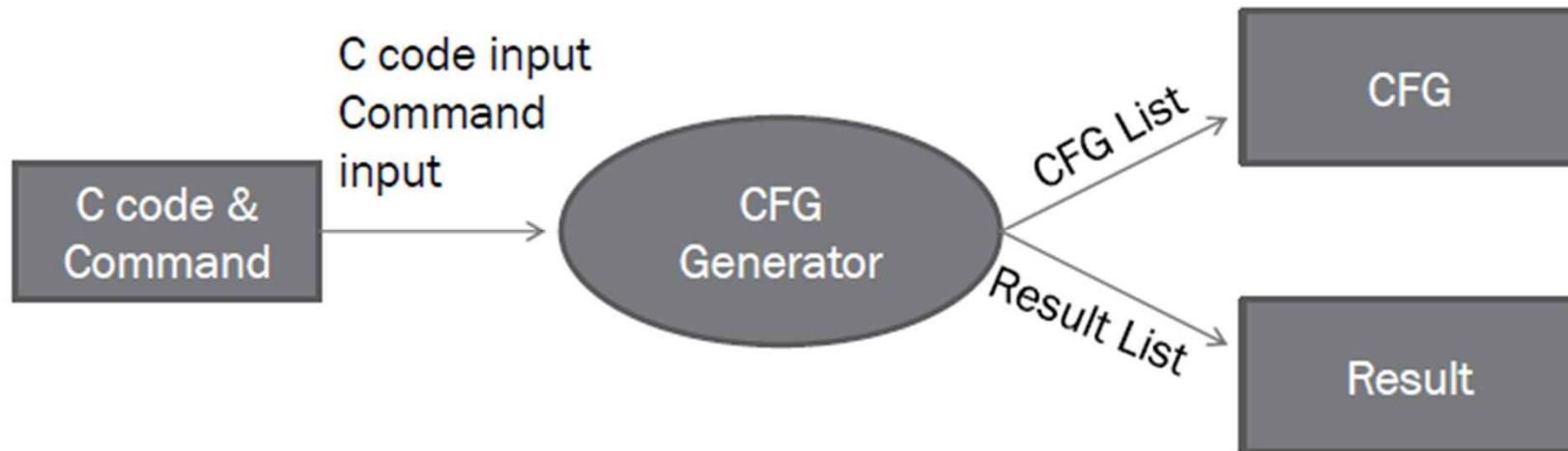
# Modified Statement of Purpose

- 이 프로그램은 C Source File을 받아들여, 그 Source Code를 분석하고, 그것을 토대로 Control Flow Graph를 생성하는 알고리즘(CFG Construction Algorithm)을 통해 알맞은 Control Flow Graph를 그리고 그에 따른 결과메시지를 출력하는 프로그램이다.
1. CFG가 만들어지기 시작하면, 화면에 시작한다는 Message를 출력해준다.
  2. CFG가 만들어지면, 화면에 Success Message만 보여주고, CFG는 File에 저장한다.
  3. CFG를 만드는데 실패하면 Error Message를 보여주고, 모든 Process를 종료한다.
  4. 무조건 C source code만을 다룬다.(\*.c)
  5. Main() Function 내부만 CFG로 만든다.

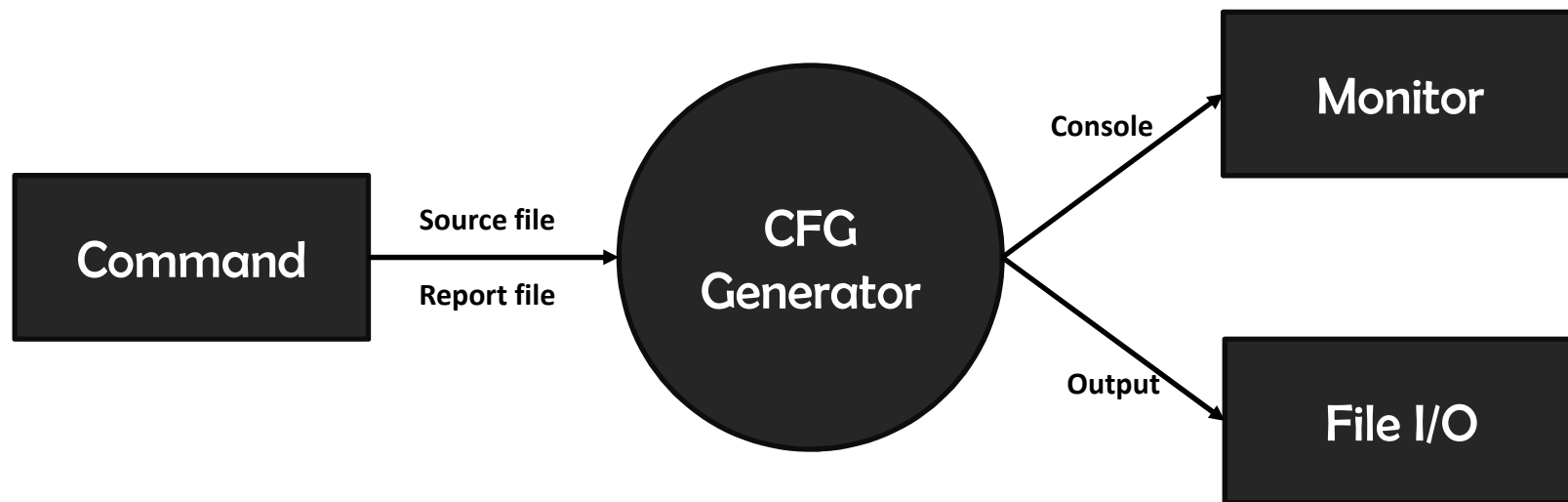
# Modified Statement of Purpose

6. '{}'와 같은 Block이 없거나, 입력된 File 경로가 올바르지 않은 경우, CFG를 만들어 없으므로, CFG Generator가 에러로 처리되어 종료된다.
7. 전체 Compile Error는 고려하지 않고, CFG를 만드는데 필요한 경우의 Error만 고려한다.('{}'와 같은 Block이 하나 모자라거나, Main()이 없는 경우)
8. CFG가 만들어지면, File에 CFG의 BB List와 Edge List를 한 줄씩 저장한다.
9. Help Message는 C Source Code의 File 경로가 틀렸을 경우 출력된다.
  - Help Message는 해당 형태로 출력된다.  
`$/cg <source code filepa(*.c)> <report filepath(*.txt)>`

# Original System Context Diagram



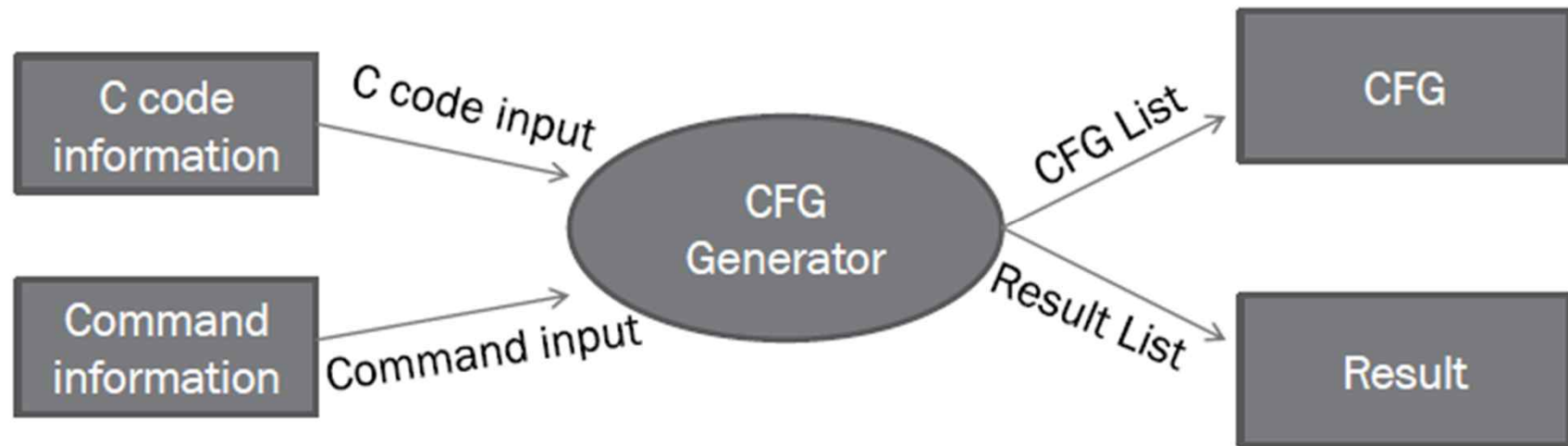
# Modified System Context Diagram



# Event List

Input/Output Event	Description	Format/Type
Source File	C Source Code의 File 경로	문자열(*.c)
Report File	완성된 CFG가 출력되는 File 경로	문자열(*.txt)
Console	콘솔에 출력될 Success/Start/Error 문자열	콘솔에 출력 될 문자열
Output	CFG의 Basic Block들과 Edge들을 출력하는 문자열	File에 출력되는 문자열

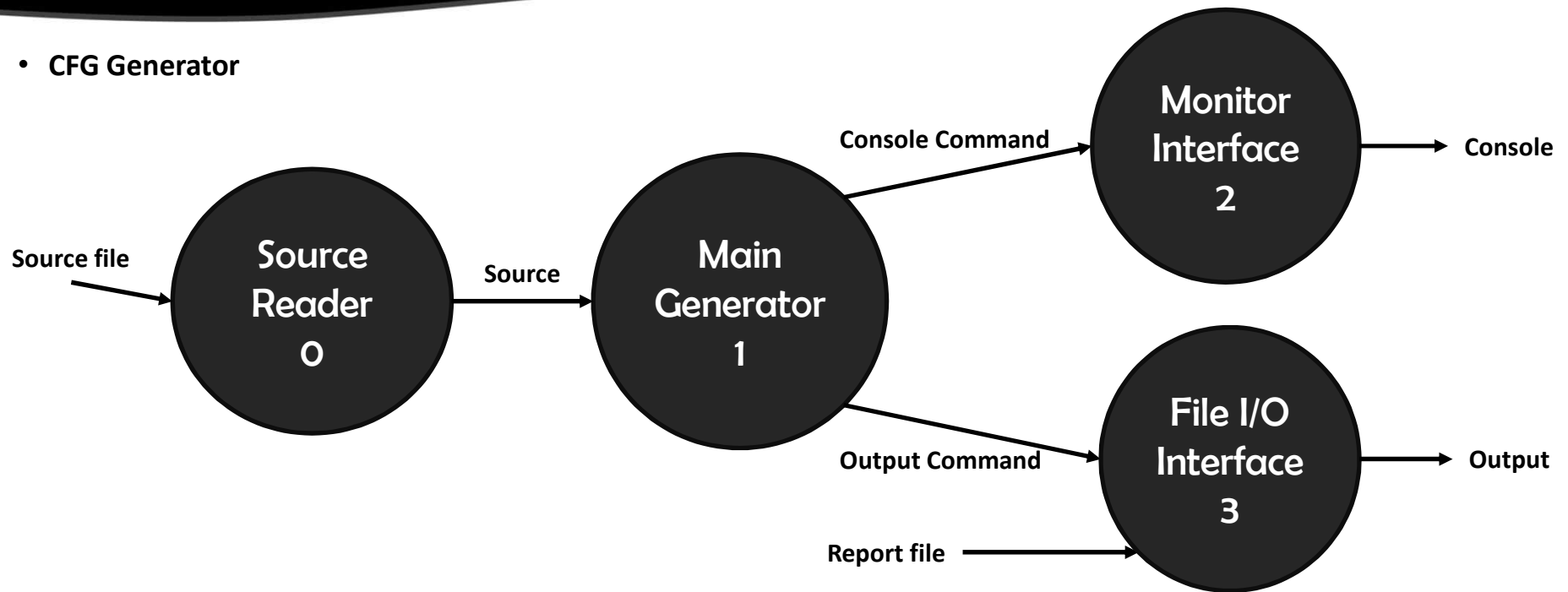
# Original Data Flow Diagram – Level 0





# Modified Data Flow Diagram – Level 0

- CFG Generator



# Data Dictionary – Level 0

Input/Output Event	Description	Format/Type
Source	Source file을 문자열로 변환	문자열/char *
Console Command	Console에 출력하기 위한 문자열	문자열/char *
Output Command	File에 써주기 위한 문자열	문자열/char *

# Process Specification- Level 0

Name	Source Reader
Reference Number	0
Input	Source File
Output	Source
Description	File I/O를 사용해서 Source File을 읽어 들여, 전부 Source로 출력해준다. Source File이 존재하지 않는 경로 일 경우, Source에 NULL을 저장한다.

# Process Specification- Level 0

Name	Main Generator
Reference Number	1
Input	Source
Output	Console Command, Output Command
Description	Source를 받아서, 이것을 파싱 하여 Console에 에러/성공에 해당하는 메시지를 출력해주고, Source File에 CFG를 쓴다.

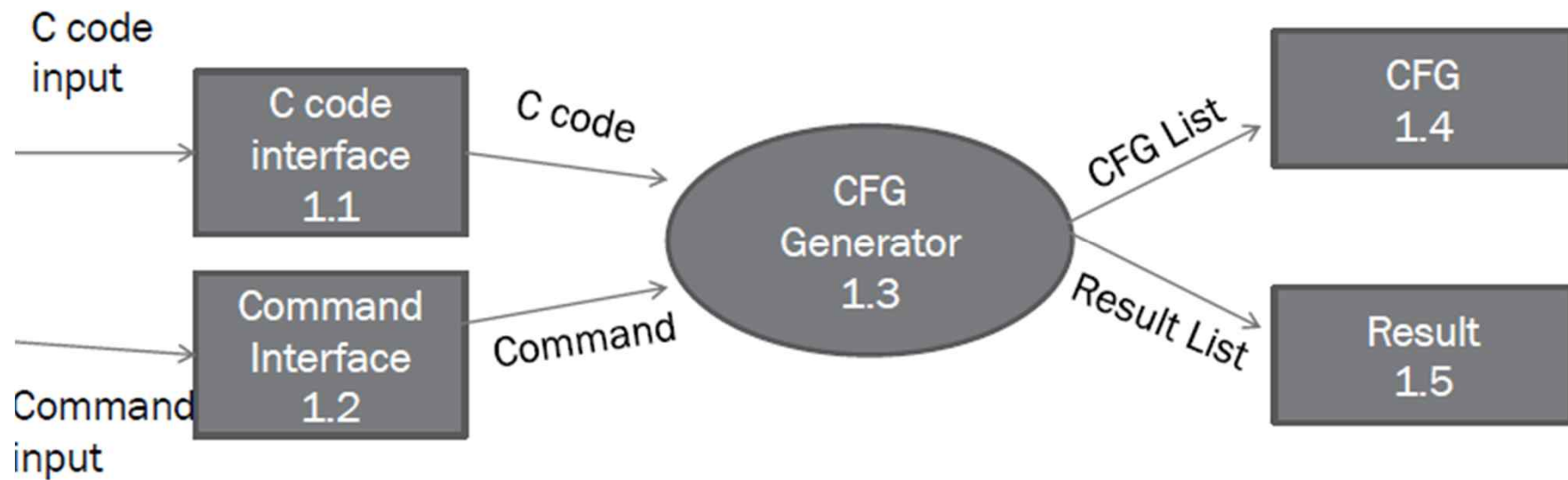
# Process Specification- Level 0

Name	Monitor Interface
Reference Number	2
Input	Console Command
Output	Console
Description	Console Command를 받아서, Console로 모니터에 출력해준다.

# Process Specification– Level 0

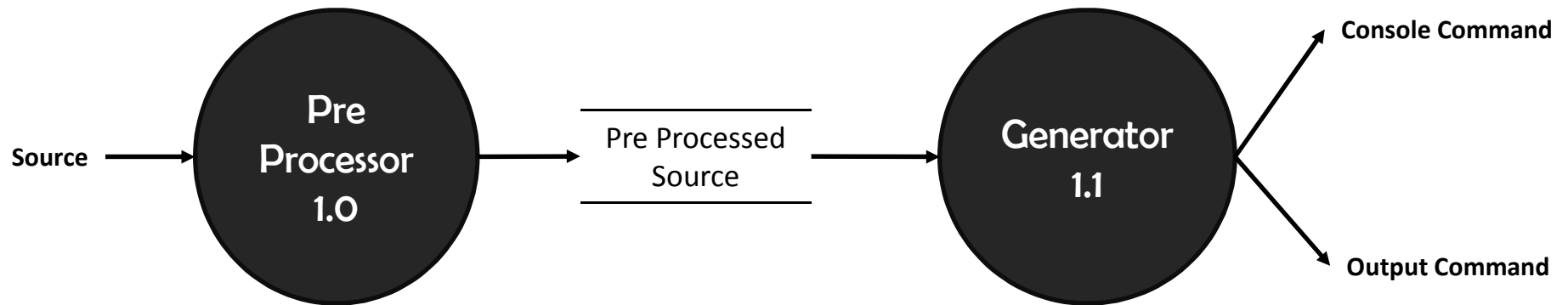
<b>Name</b>	<b>File I/O Interface</b>
Reference Number	3
Input	Output Command
Output	Output
Description	Output Command를 입력 받아서, Report File에 Output을 출력해준다.

# Original Data Flow Diagram – Level 1



# Modified Data Flow Diagram – Level 1

- Main Generator 1





# Data Dictionary – Level 1

Input/Output Event	Description	Format/Type
Pre Processed Source	Source에서 main함수 내부만 뽑아낸 문자열	문자열/char *

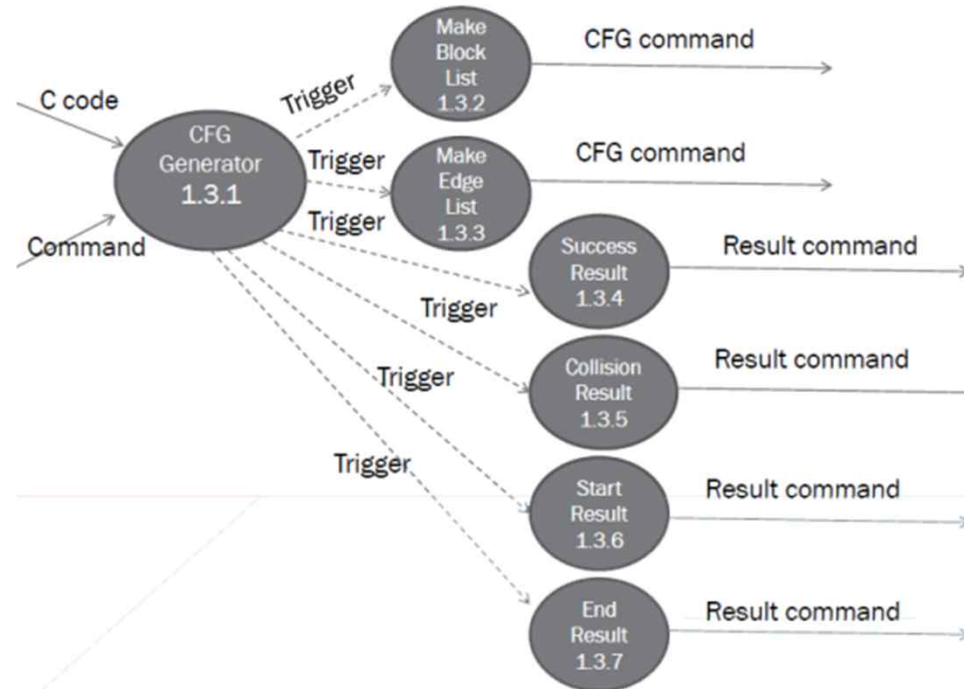
# Process Specification– Level 1

Name	Pre Processor
Reference Number	1.0
Input	Source
Output	Pre Processed Source
Description	Source에서 main함수의 내부를 뽑아 내어 Pre Processed Source에 저장한다. Source가 NULL일 경우, Pre Processed Source에 NULL을 저장한다.

# Process Specification– Level 1

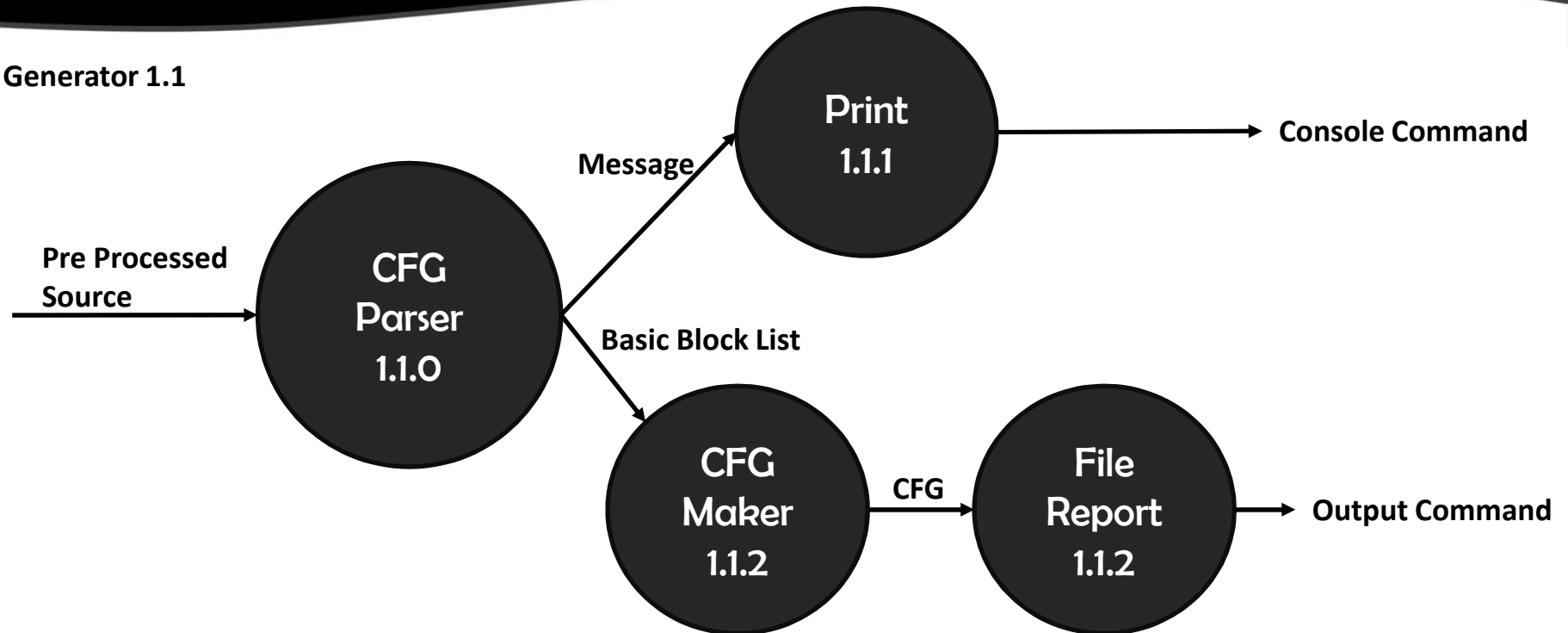
Name	Generator
Reference Number	1.1
Input	Pre Processed Source
Output	Output Command, Console Command
Description	Pre Processed Source를 CFG형태로 변환 한 후, 적절한 Output Command와 Console Command를 출력해준다.

# Original Data Flow Diagram – Level 2



# Modified Data Flow Diagram – Level 2

- Generator 1.1



# Data Dictionary – Level 2

Input/Output Event	Description	Format/Type
Message	에러/성공/시작 등을 담고 있는 문자열이다.	문자열/char *
Basic Block List	각 단계에서 만들어진 Basic Block들이다. Basic Block들은 서로 연결되어 있다.	이것과 연결되어진 List와 Source/BasicBlock{BasicBlockListNode *, char *}
CFG	Basic Block들을 연결시킨 최종적인 결과물CFG	Basic Block들이 서로 연결되어있음 / Graph

# Process Specification- Level 2

Name	CFG Parser
Reference Number	1.1.0
Input	Pre Processed Source
Output	Message, Basic Block List(+ Datastructure)
Description	만약, 실패 할 경우 Message에 실패 메시지를 담아 출력해주고, 성공 할 경우, 각 단계에서 만들어진 Basic Block 들을 출력한다.

# Process Specification- Level 2

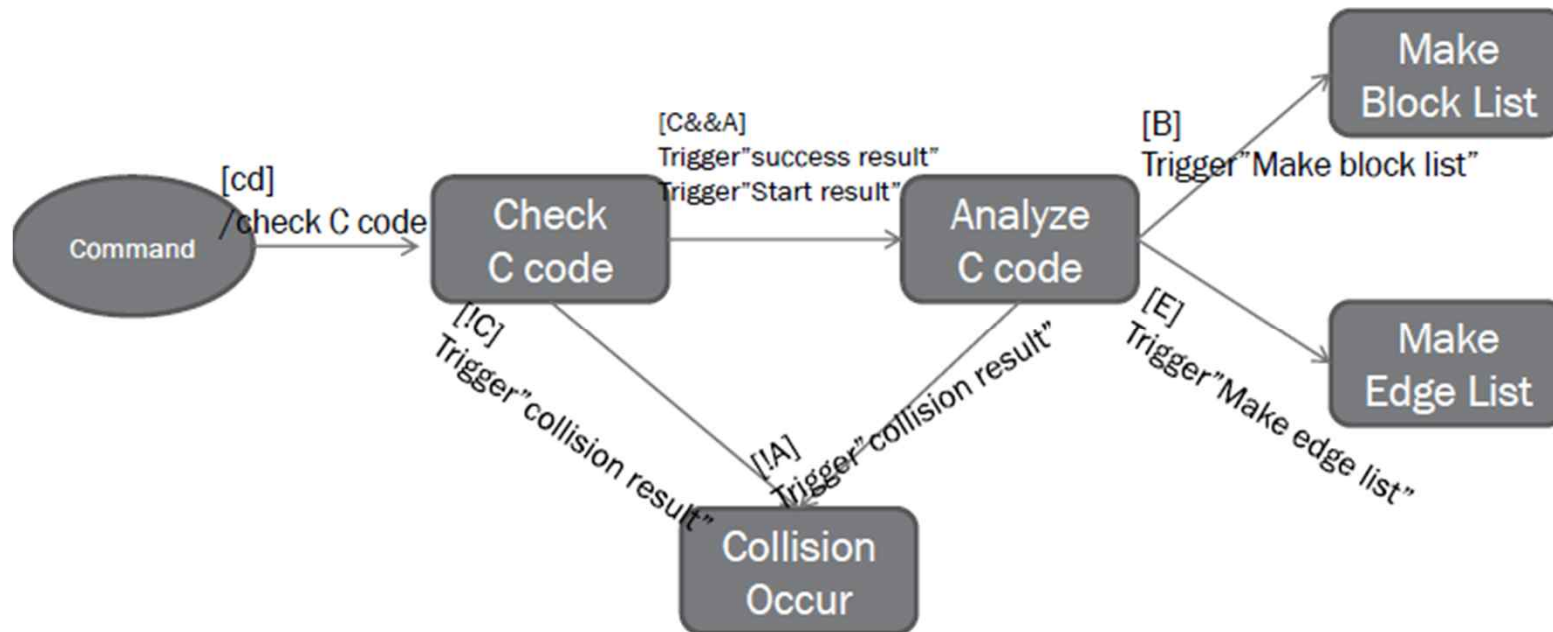
<b>Name</b>	<b>Print</b>
Reference Number	1.1.1
Input	Message
Output	Console Command
Description	Message를 입력 받아서, Console Command로 변환하여 출력한다.



# Process Specification- Level 2

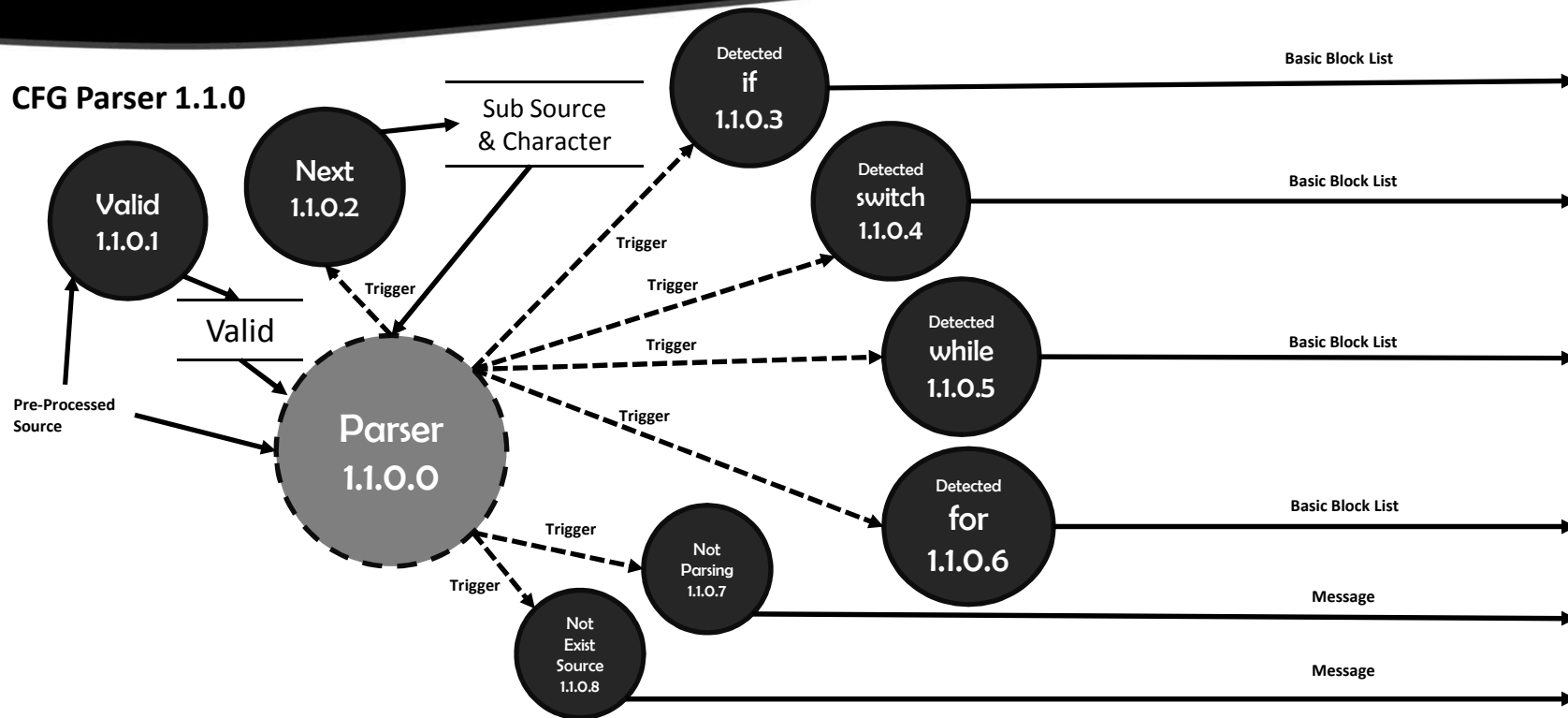
Name	File Report
Reference Number	1.1.2
Input	CFG
Output	Output Command
Description	CFG를 계층구조 형태로 쪽 돌면서, Basic Block과 Edge 들을 뽑아낸 후, 그것들을 Output Command로 출력해 준다.

# Original Data Flow Diagram – Level 3



# Modified Data Flow Diagram – Level 3

- CFG Parser 1.1.0



# Data Dictionary – Level 3

Input/Output Event	Description	Format/Type
Valid	Pre Processed Source 가 제대로 되어있는지 검사하여 결과가 저장되었다.	문자열 / char *
Sub Source	Clear되고 난 후, Next 로 부터 한글자씩 붙여진 소스의 조각이다.	문자열 / char *
Character	Next로부터 나온 문자이다.	문자 / char

# Process Specification– Level 3

Name	Parser
Reference Number	1.1.0.0
Input	Pre Processed Source(PPS), Valid(V), Sub Source(SS), Character(C)
Output	Trigger
Description	PPS가 NULL이라면, Not Exist Source를 Trigger한다. 그리고 매번 Next를 호출하여, C와 SS를 가져와서, 해당 조건에 맞게 Detected들과 Statement를 Trigger해준다.

# Process Specification– Level 3

Name	Valid
Reference Number	1.1.0.1
Input	Pre Processed Source
Output	Valid
Description	Pre Processed Source를 돌면서, 괄호( ( ), { }, [ ] )들이 제대로 되어 있는 지만 검사한다.

# Process Specification- Level 3

Name	Next
Reference Number	1.1.0.2
Input	Trigger
Output	Sub Source(SS), Character(C)
Description	Pre Processed Source를 한글자씩 C에 저장하고, SS는 Clear되고 나서, 지금까지의 C를 붙여만들어진 문자열로 저장한다.

# Process Specification- Level 3

Name	Detected if
Reference Number	1.1.0.3
Input	Trigger
Output	Basic Block List
Description	Next를 호출하여, if문 전체를 파싱 해서, Basic Block들로 만들고 그것들을 Basic Block List에 저장해준다.



# Process Specification– Level 3

Name	Detected switch
Reference Number	1.1.0.4
Input	Trigger
Output	Basic Block List
Description	Next를 호출하여, switch문 전체를 파싱 해서, Basic Block들로 만들고 그것들을 Basic Block List에 저장해준다.

# Process Specification– Level 3

Name	Detected while
Reference Number	1.1.0.5
Input	Trigger
Output	Basic Block List
Description	Next를 호출하여, while문 전체를 파싱 해서, Basic Block들로 만들고 그것들을 Basic Block List에 저장해준다.

# Process Specification– Level 3

Name	Detected for
Reference Number	1.1.0.6
Input	Trigger
Output	Basic Block List
Description	Next를 호출하여, for문 전체를 파싱 해서, Basic Block들로 만들고 그것들을 Basic Block List에 저장해준다.

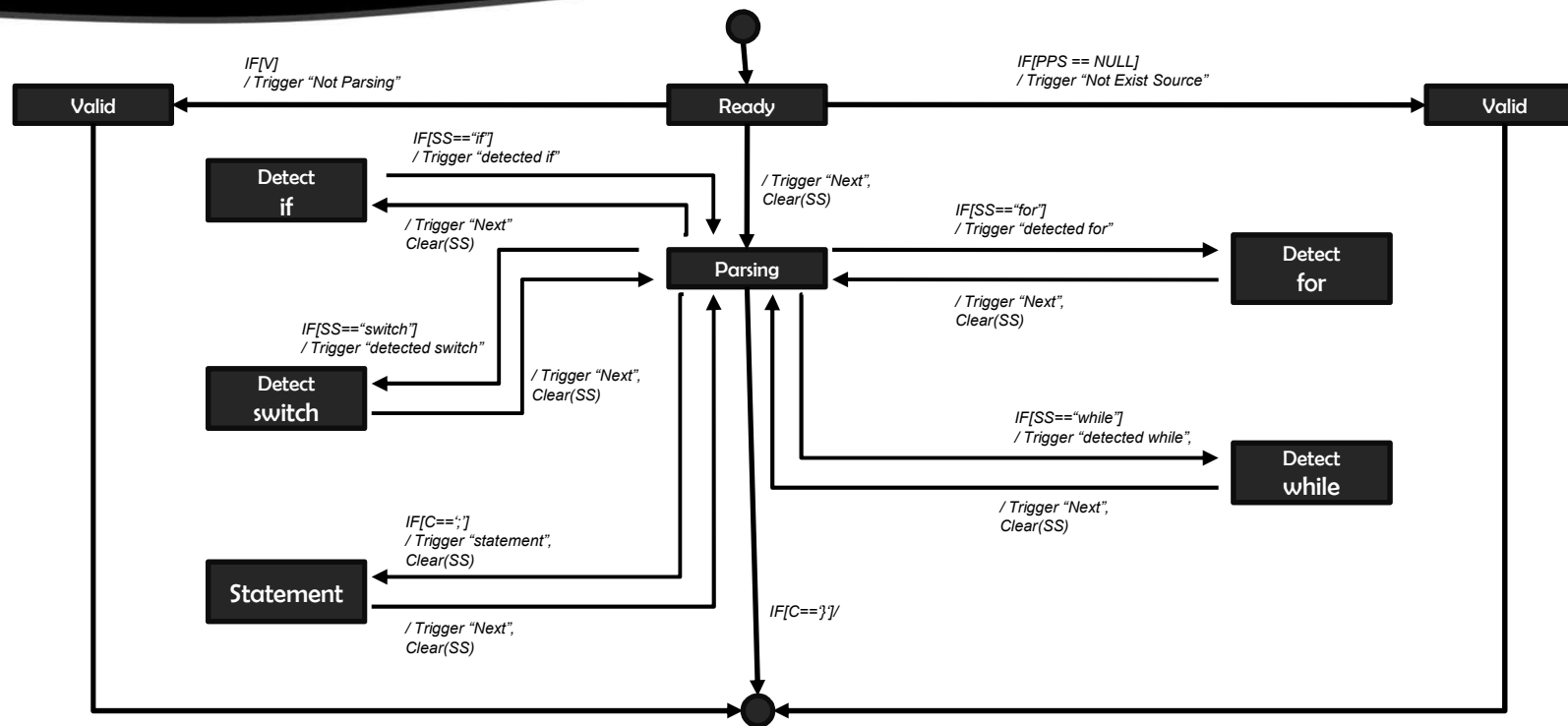
# Process Specification– Level 3

<b>Name</b>	<b>Not Parsing</b>
Reference Number	1.1.0.7
Input	Trigger
Output	Message
Description	Message에 에러를 저장한다.

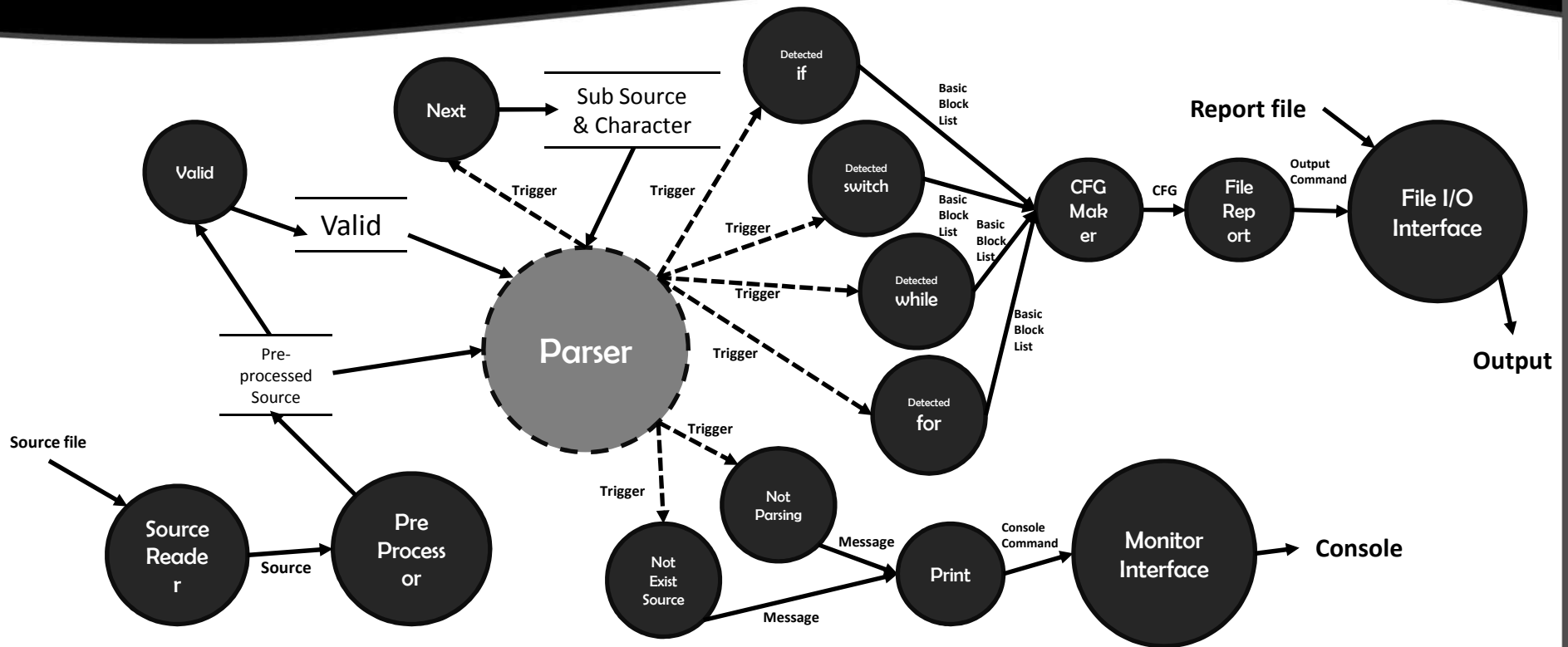
# Process Specification- Level 3

<b>Name</b>	<b>Not Exist Source</b>
Reference Number	1.1.0.8
Input	Trigger
Output	Message
Description	Message에 Usage에 해당하는 문자열을 저장한다.

# State Transition Diagram – Level 4

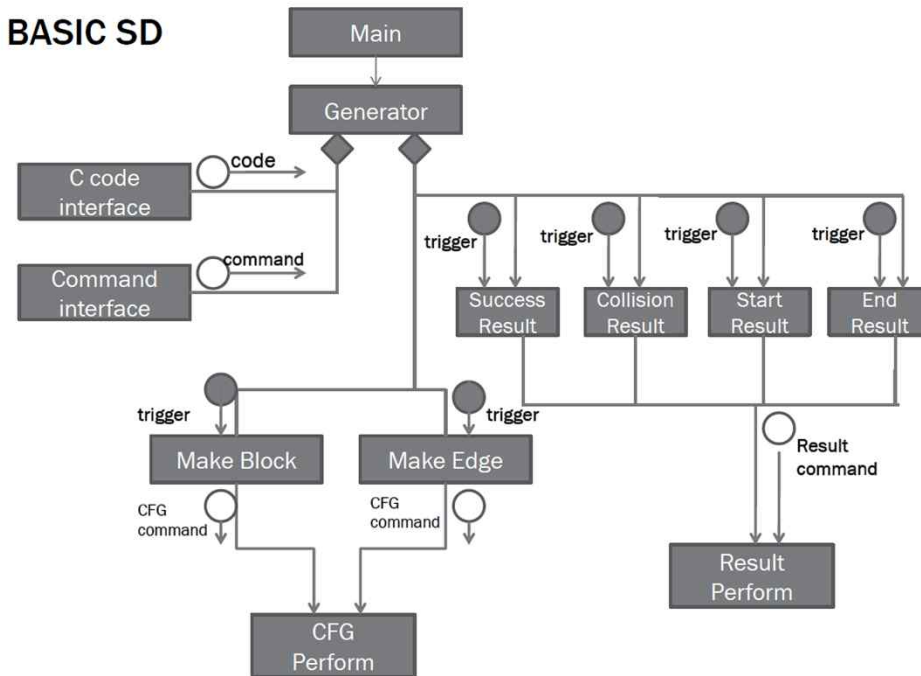


# Modified Total Data Flow Diagram



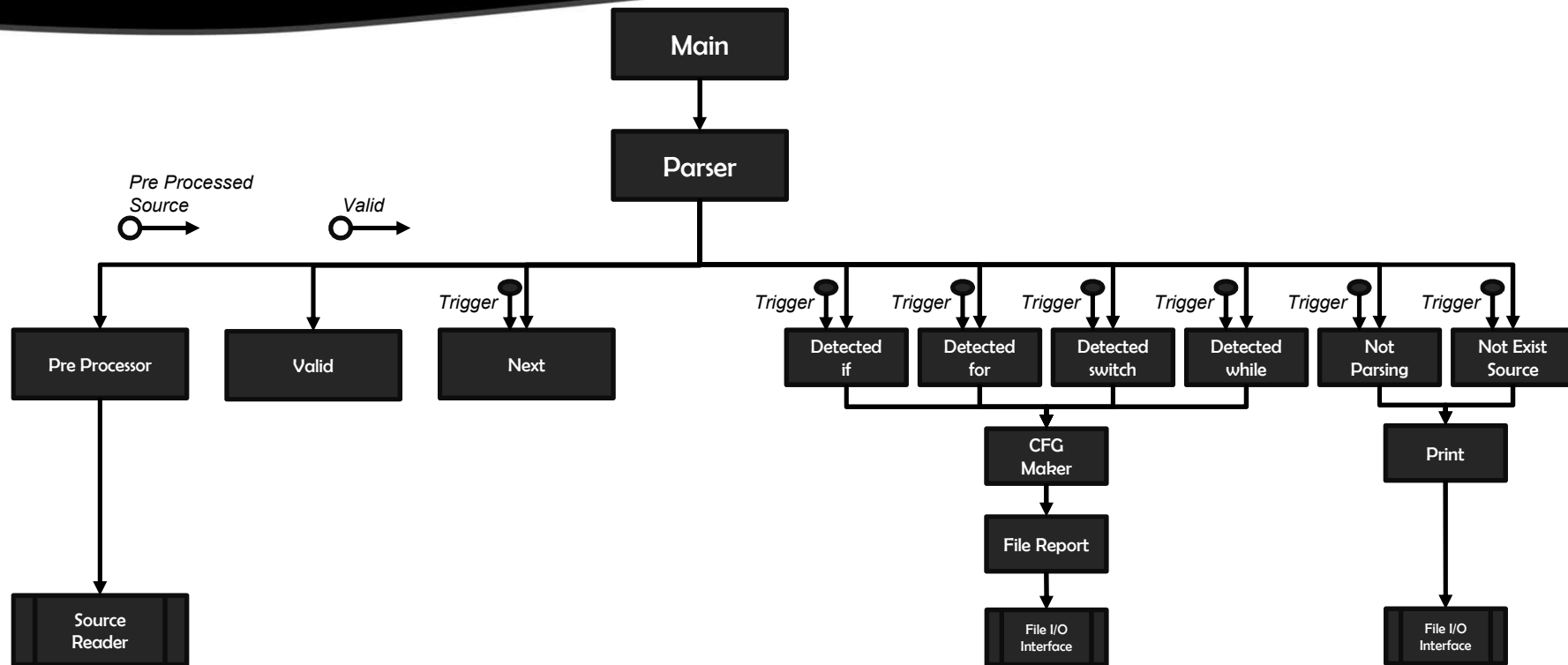
# Original Structured Charts

BASIC SD





# Modified Structured Charts



THANK YOU!!!