A Syntax Directed Environment for Tabular Form Processing

○仲川　俊一　(日本大学)
有田　友和　(日本大学)
冨山　聖宣　(日本大学)
夜久　竹夫　(日本大学)
宮寺　庸造　(東京学芸大学)
土田　賢省　(東洋大学)

電子情報通信学会　２００１年総合大会
To Appear in ICSE 2001(Poster)

Targets of Our Works

Project Code:	A 5
Program Name:	**Program Specification-1** p
Library Code:	Version:
Author:	Original Release:
Approver:	Current Release:
Problem Description:	
Problem Supplementary Information (Theoretical Principles, Methods and References):	
Problem Solution: 1.Conventions and Terminology 2.Principles and Algorithms	

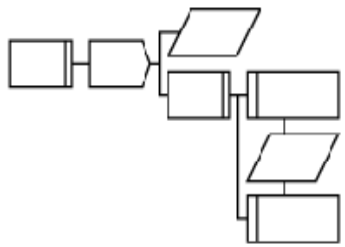
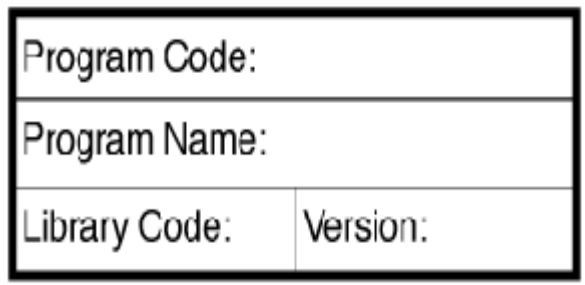
Program Specification Form

Graph Grammar

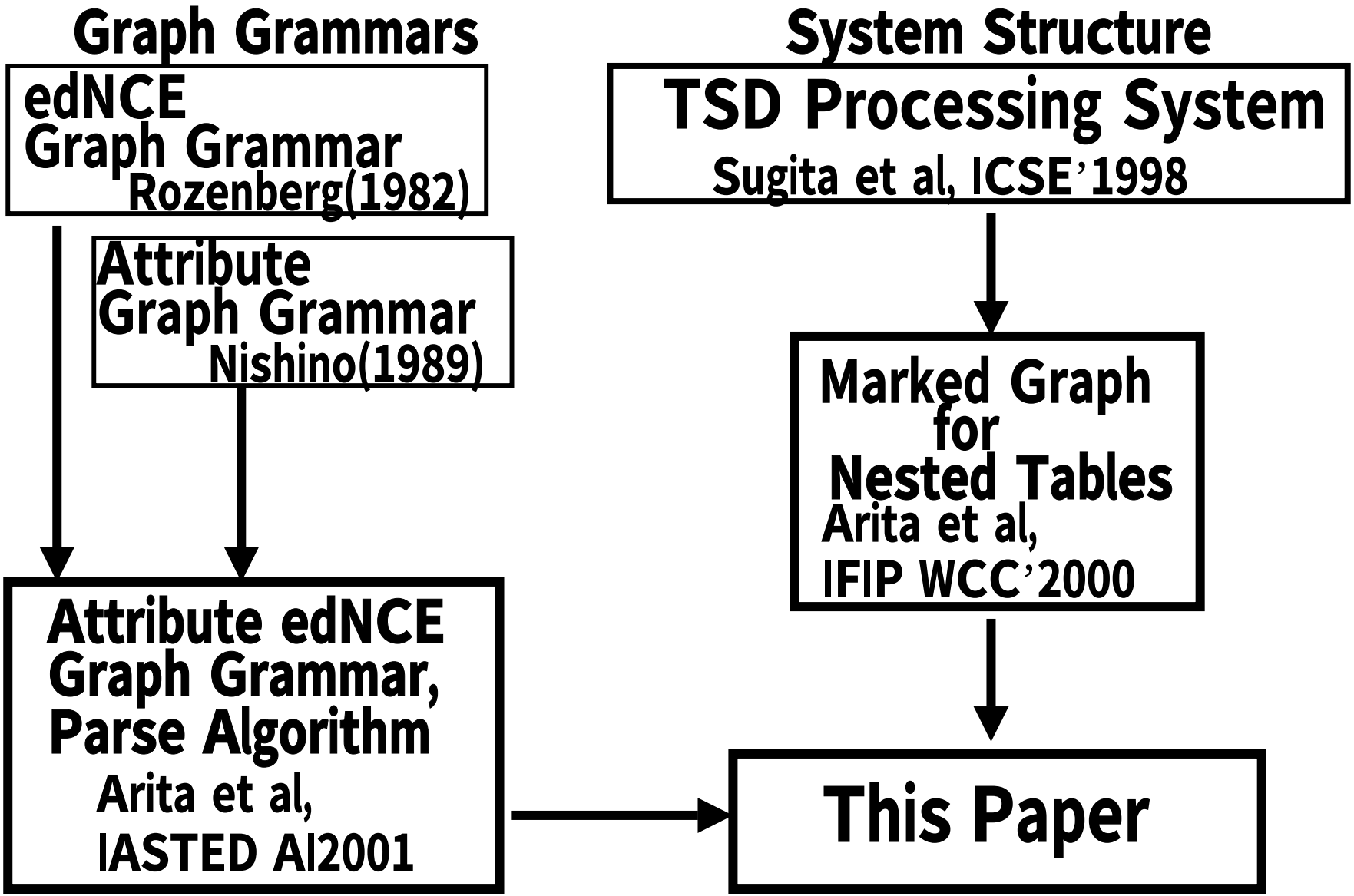
- **Syntax Analysis**
- **Mechanical Drawing**

1. Introduction

Position of This Paper

	DiaGen	HichartED	HiformED
Diagram	NS chart, Trees for hierarchical structure and so on	 Hierarchical Diagram	 Program Specification
Theoretical Model	Attribute Hypergraph Grammar	Attribute CFGG	Attribute NCE CFGG
System	○	○	This Paper

Background


Graph Grammars
edNCE
Graph Grammar
Rozenberg(1982)
Attribute
Graph Grammar
Nishino(1989)
Attribute edNCE
Graph Grammar,
Parse Algorithm
Arita et al,
IASTED AI2001
System Structure
TSD Processing System
Sugita et al, ICSE'1998
Marked Graph
for
Nested Tables
Arita et al,
IFIP WCC'2000
This Paper

Motivation

To construct an editing system of tabular forms which can perform exact drawing based on graph grammars

Purpose

- Constructing the parsing engine for the tabular form editor
- Constructing the system include the parsing engine and the syntactic editor

Results

- We decided the system structure and the file structure of this tabular form editor
- We developed the parsing engine based on the structures
- We considered syntax editing mechanisms of tabular forms

2. Tabular Forms and Their Syntax

Hiform : Program Specification Forms

- **Hiform includes all items in ISO6592**
- **Hiform consists of 17 types of forms**

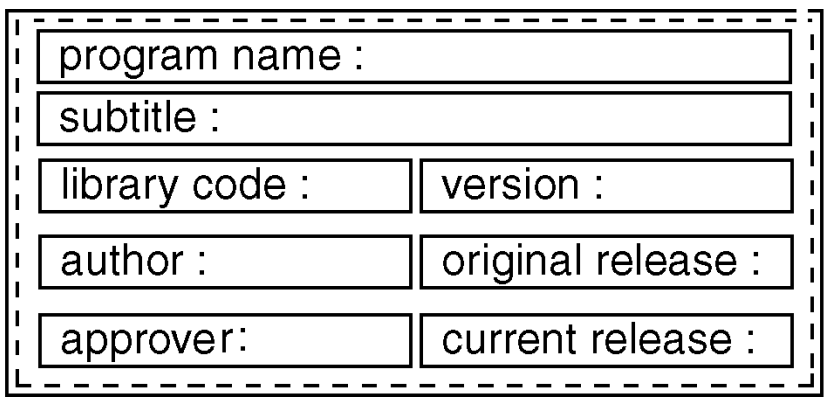
Project Code:	A 5
Program Name:	**Program Specification-1** p
Library Code:	Version:
Author:	Original Release:
Approver:	Current Release:
Problem Description:	
Problem Supplementary Information (Theoretical Principles, Methods and References):	
Problem Solution: 1.Conventions and Terminology 2.Principles and Algorithms	

Program Specification Form

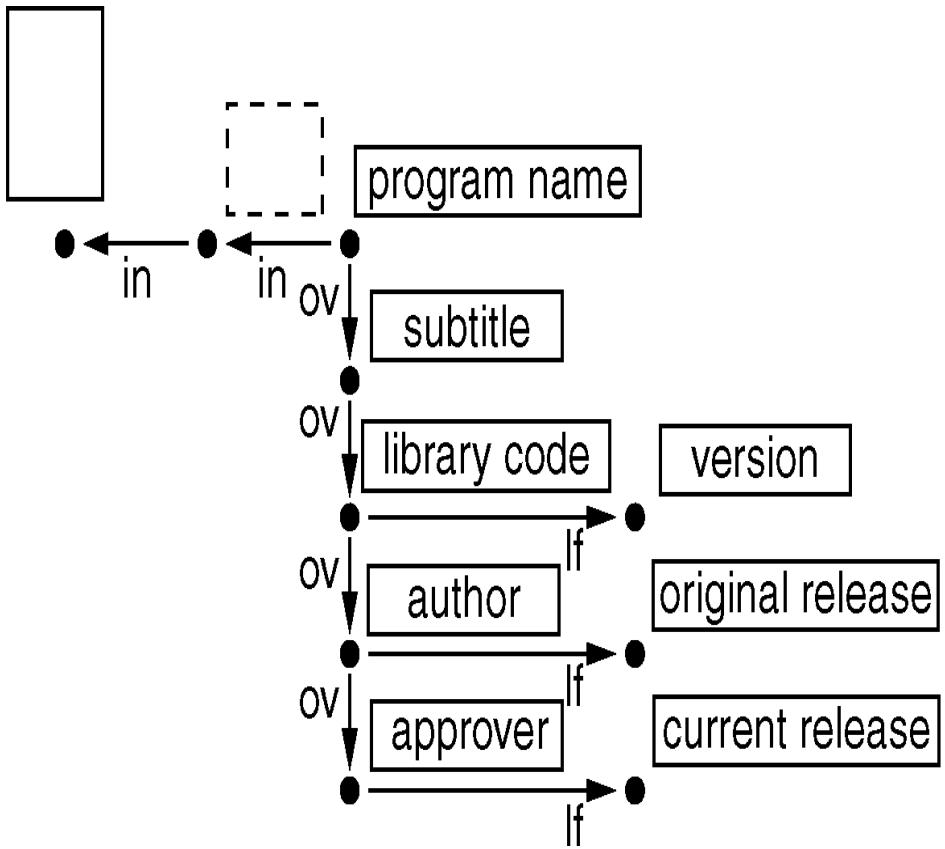
Tabular Forms and Marked Graphs

program name :	
subtitle :	
library code :	version :
author :	original release :
approver:	current release :

Tabular form



Nested Diagram



Marked Graph

edNCE Graph Grammars

Definition

An edNCE graph grammar ：

$G=(\Sigma, \Delta, \Gamma, \Omega, P, S)$

- Σ：the alphabet of node labels
- $\Delta \subseteq \Sigma$：the alphabet of terminal node labels
- Γ：the alphabet of edge labels
- $\Omega \subseteq \Gamma$：the alphabet of final edge labels
- P：the finite set of productions

  production $p : X \rightarrow (D,C)$
- $S \in \Sigma - \Delta$：the initial nonterminal □

Attribute edNCE Graph Grammars

Definition

An attribute edNCE Graph Grammar :

AGG = <G, Att, F>

G= (Σ , Δ , Γ , Ω, P, S) :

an underlying graph grammar of AGG

Att = $\bigcup_{Y \in \Sigma}$ Att(Y), (Att(Y) = Inh(Y) ∪ Syn(Y)) :

the set of attribute

F = $\bigcup_{p \in P}$ Fp : the set of semantic rules of AGG □

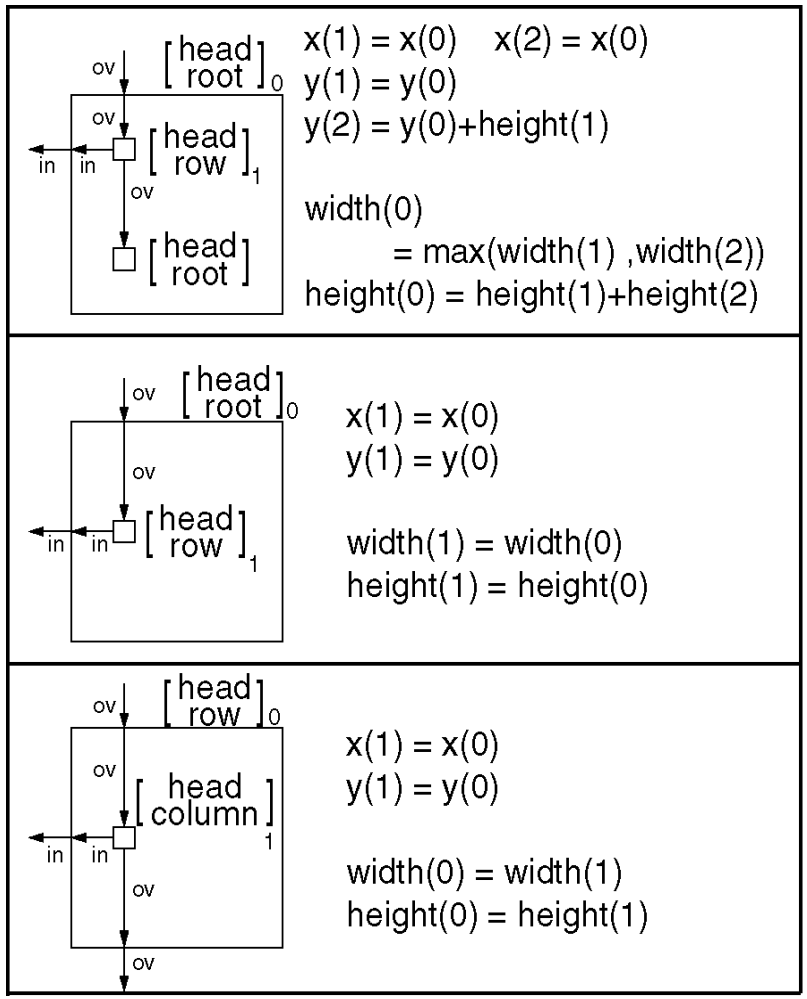
An Attribute edNCE Graph Grammar for Hiform

HNGG = $<G_N, A_N, F_N>$ formulates Hiform

Underlying graph grammar $G_N = (\Sigma_N, \Delta_N, \Gamma_N, \Omega_N, P_N, S_N)$ (edNCE context-free graph grammar)

Productions : 280 rules

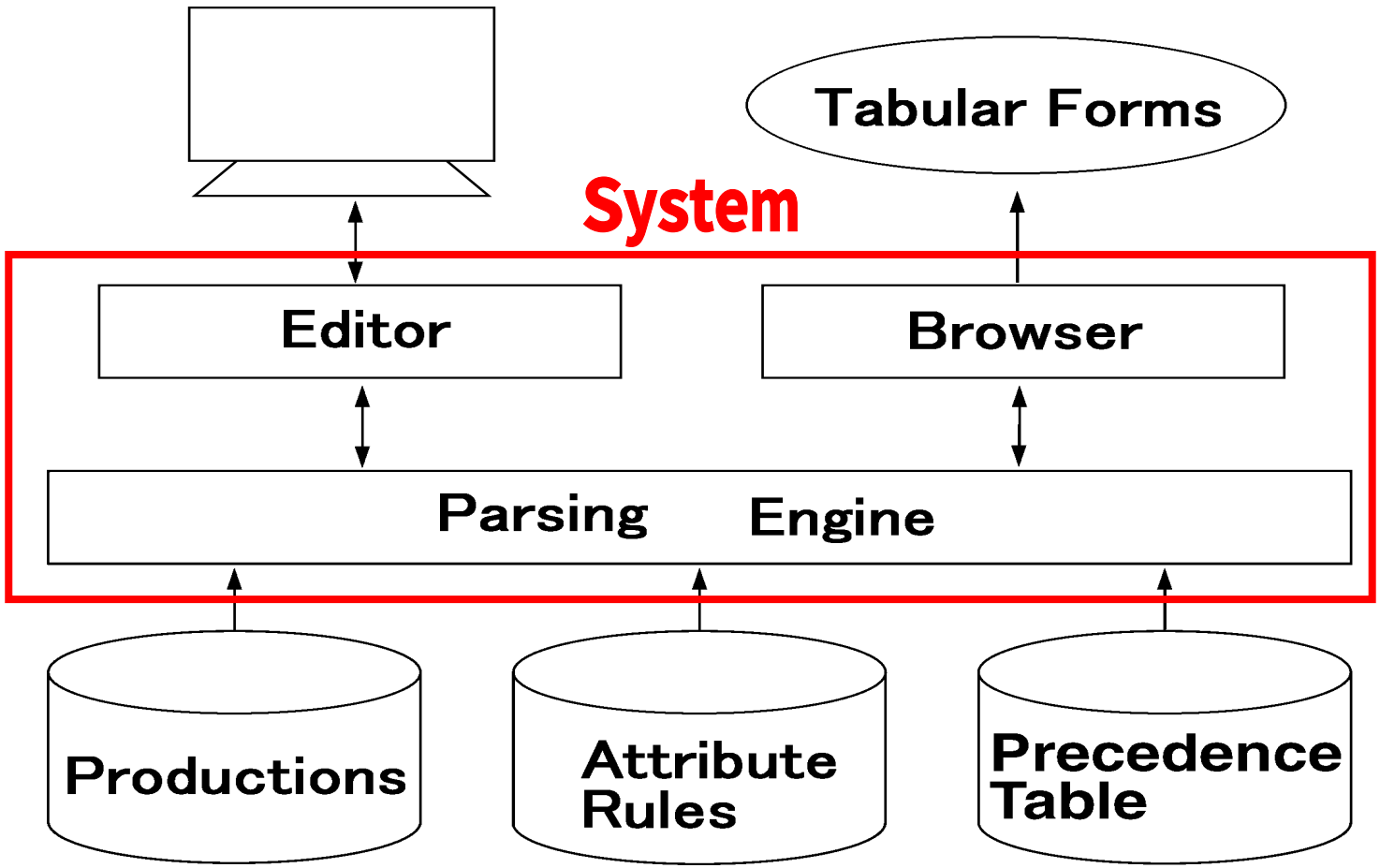
Attribute Rules : 1248 rules



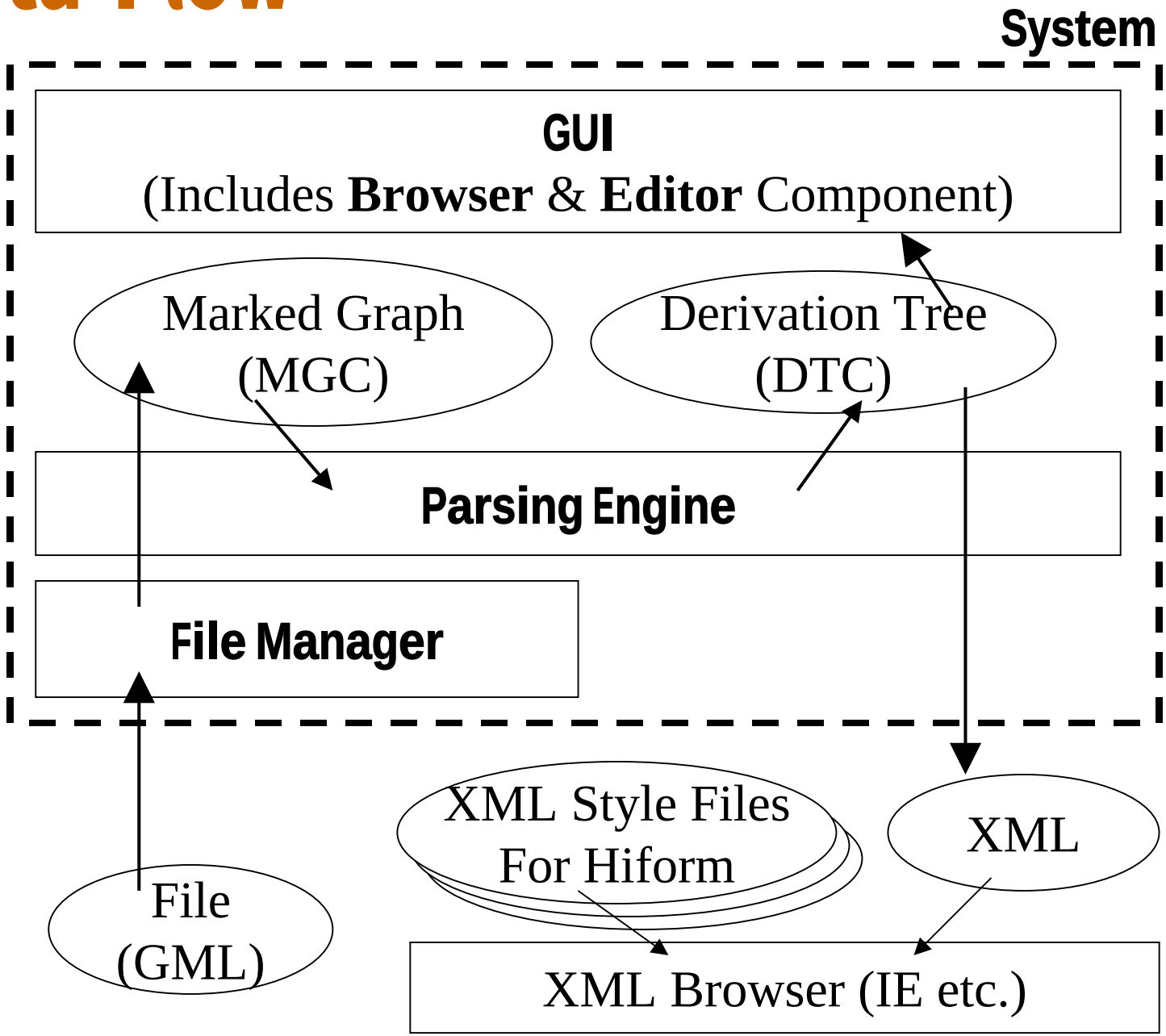
A part of productions in HNGG

3. System

3.1 System Overview



Data Flow

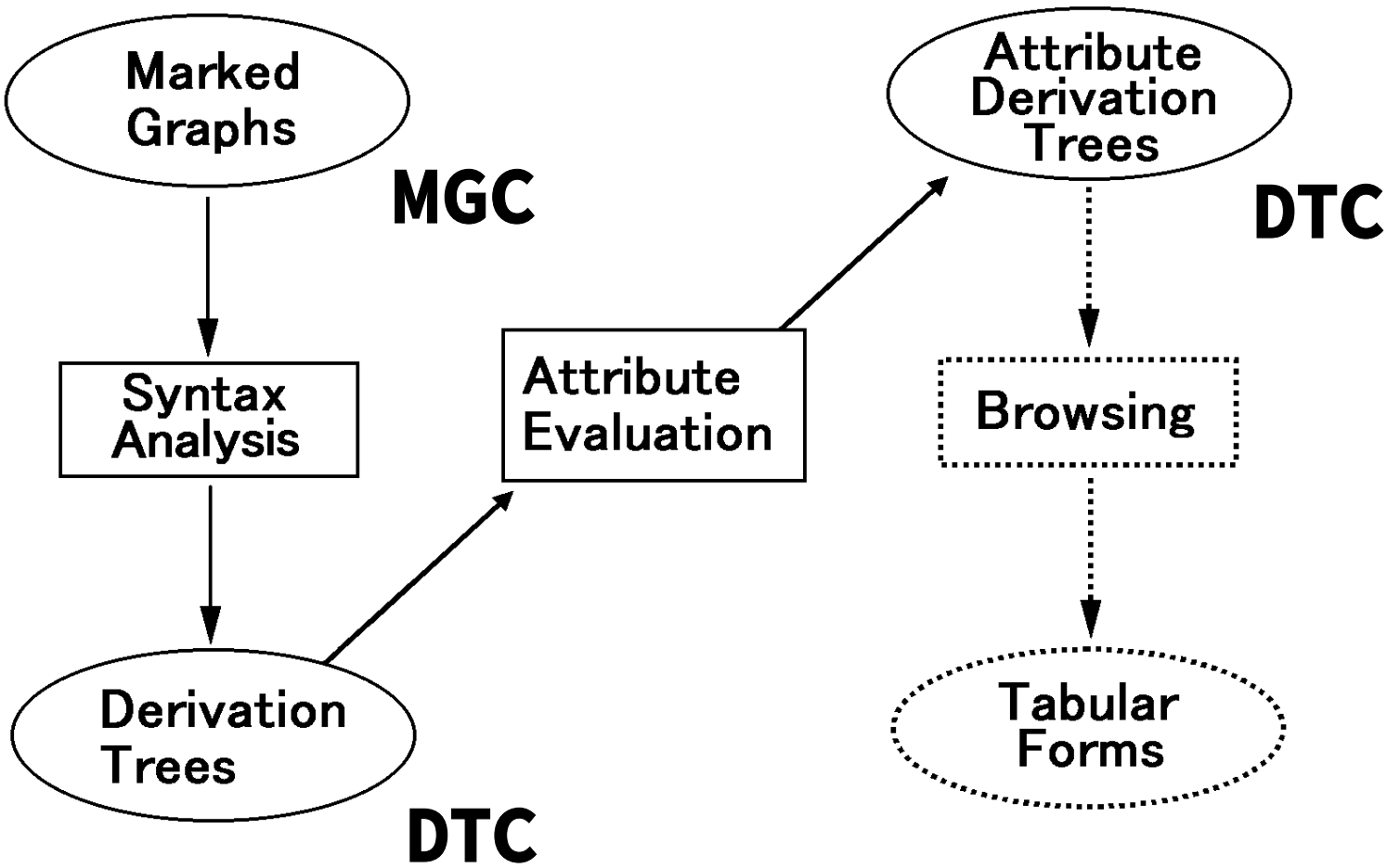

System
GUI
(Includes Browser & Editor Component)
Marked Graph
(MGC)
Derivation Tree
(DTC)
Parsing Engine
File Manager
XML Style Files
For Hiform
XML
File
(GML)
XML Browser (IE etc.)

3.2 Parsing Engine

- Syntax Analysis
- Attribute Evaluation

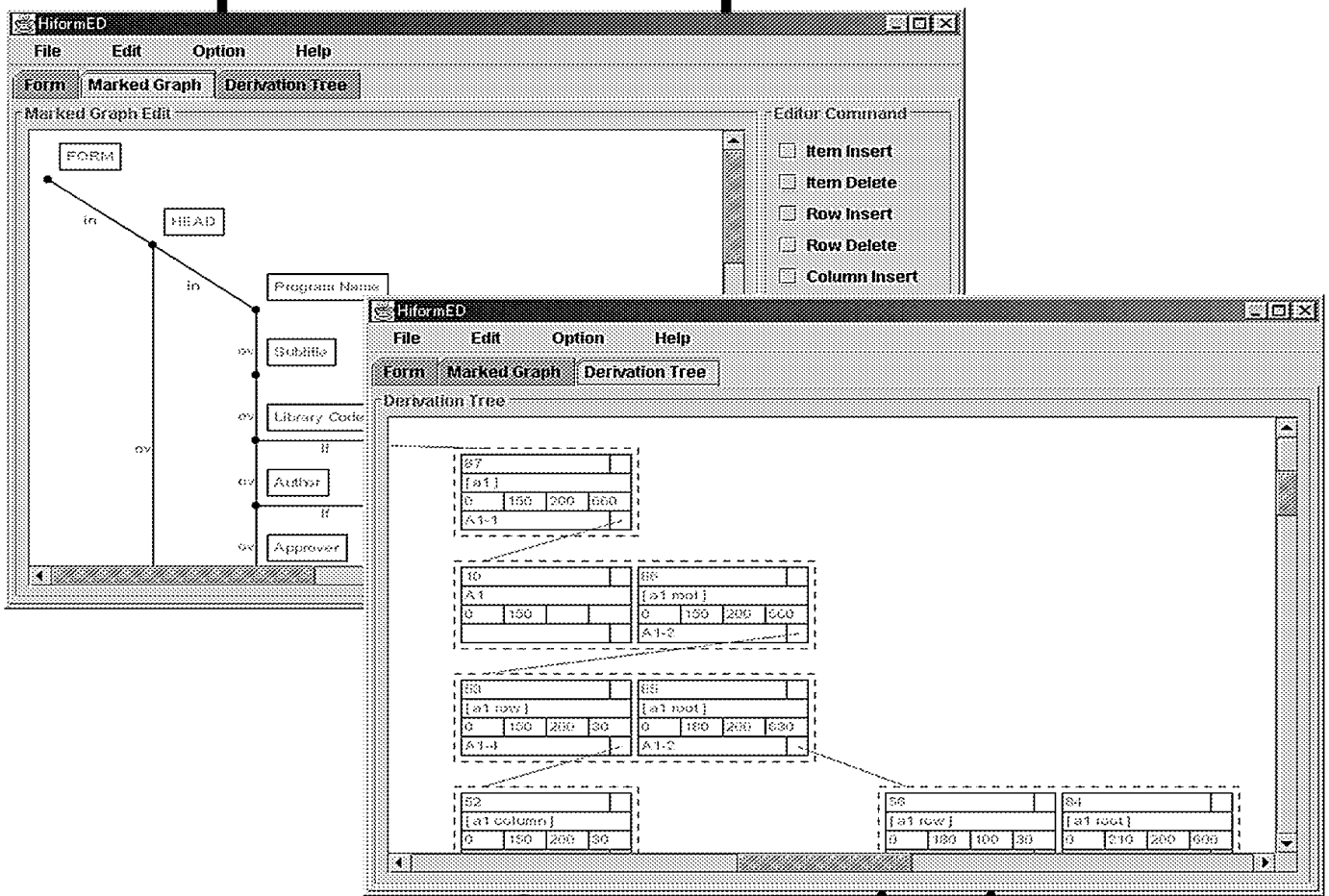
- Input : Marked Graph with Attribute (Marked Graph Class)
- Output : Attribute Derivation Tree (Derivation Tree Class)

3.2 Parsing Engine (continued)



An Execution Screen of Parsing Engine

Input : Marked Graph



Output : Derivation Tree

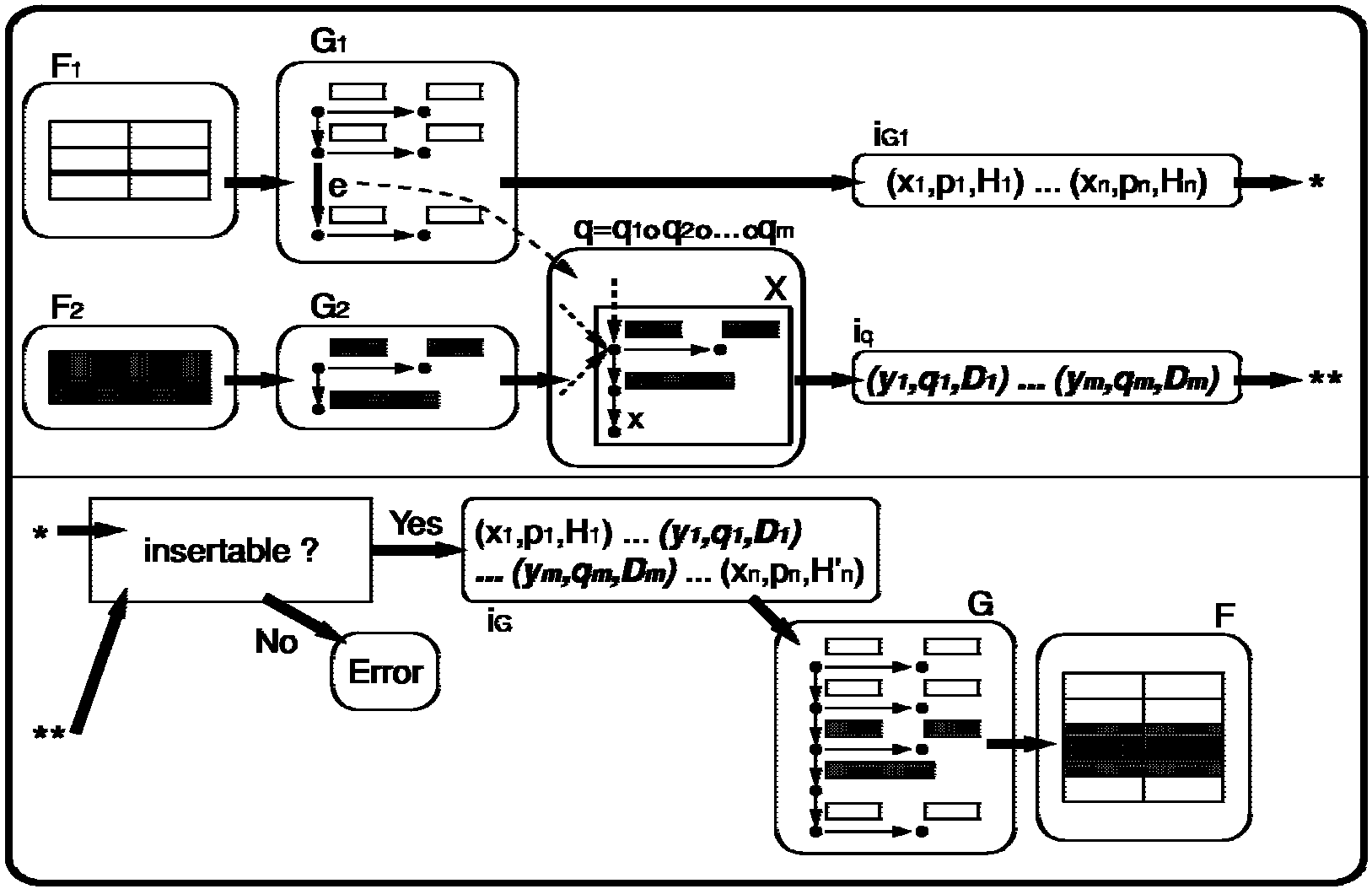
Feature of Editor

Editing mechanisms

- Formalized mathematically
- Do not cause syntax error
- Include Insertion, Deletion and so on

A Flow of an Insertion Process

Insertion of F_2 into F_1 at e



4. Conclusions

- We decided the system structure and the file structure of this tabular form editor
- We developed the parsing engine based on the structures
- We considered syntax editing mechanisms of tabular forms

4. Conclusions(continued)

Feature of Our System

Editor	Under development
Browser	3k Java lines

Parsing Engine	2k Java lines

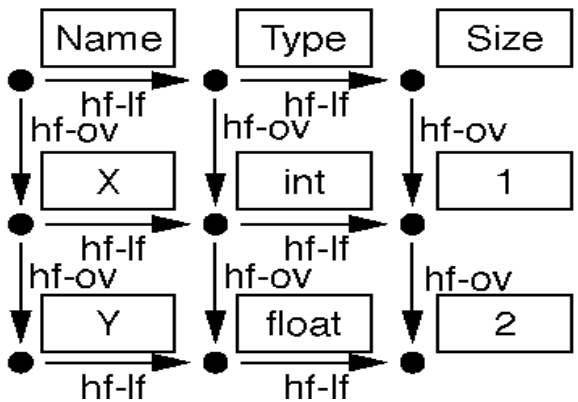
Production	280 Rules
Attribute Rules	1248 Rules
Precedence Table	5376 Relations

Marked Graph (Inner Code)	Marked Graph Class
Marked Graph (File Format)	Graph Modeling Language
Derivation Tree (Inner Code)	Derivation Tree Class

Future Works

- We are now investigating environment for tessellation forms based on HTGG.

Name	Type	Size
X	int	1
Y	float	2



Tessellation tabular form and marked graph