

ПОСТРОЕНИЕ ОРГРАФА ПО ИЕРАРХИЧЕСКОЙ IDEF0-МОДЕЛИ С ЦЕЛЮ ПОСЛЕДУЮЩЕГО СТРУКТУРНОГО АНАЛИЗА

С. А. Огрызков

Владимирский государственный университет

Как уже отмечалось в [1], IDEF0-модели (равно как и DFD-модели) по природе своей являются иерархическими, то есть содержат несколько уровней абстракции (минимум – два), на каждом из которых – своя декомпозиция функций (потоков данных) системы.

Там же, в [1], был проведён структурный анализ (с использованием теории графов и множеств) декомпозиции одного (первого) уровня, что для решения проблемы выбора вариантов моделей было вполне достаточно. Для выполнения же анализа такой модели **в целом**, в частности, для преобразования всей модели в один орграф требуется провести ей линеаризацию.

Линеаризация иерархической модели

Линеаризация иерархической модели (для примера – IDEF0-модели), то есть её приведение от иерархического вида к линейному, одноуровневому (в данном случае – к орграфу) выполняется по следующему алгоритму:

1. Отобрать из дерева IDEF0-диаграмм все диаграммы, на которых присутствуют недекомпозированные блоки («листья»).
2. Составить список всех недекомпозированных блоков отобранных (в п. 1 данного алгоритма) диаграмм, дав им номера вида $Ax.n$ (где Ax – это идентификатор диаграммы, а n – номер блока на ней) и отсортировав список по ним.
3. Добавить в составленный (в п. 2 данного алгоритма) список первым элементом блок № 0 «Внешняя среда».
4. Построить квадратную матрицу размером $N \times N$ (где N – число элементов в списке блоков после п. 3 данного алгоритма), поименовав столбцы и строки отсортированными номерами блоков.

5. Заполнить элементы главной диагонали матрицы (п. 4 данного алгоритма) нулевыми значениями (0), что означает отсутствие переходов блоков в самих себя.

6. Элементы 1-й строки матрицы (определяющие дуги, исходящие от блока № 0 «Внешняя среда» к другим блокам) заполнить значениями по следующему принципу: последовательно перебирая все входные стрелки, стрелки ограничений и механизмов, существующие на контекстной диаграмме А-0, проследить их прямое вхождение во все блоки из списка п. 2 данного алгоритма, и поместить в соответствующую ячейку матрицы значение 1, если такое вхождение имеет место быть, и 0 в противном случае.

7. Элементы 2-й и далее (до N-й) строки матрицы заполнить значениями по следующему принципу: последовательно перебирая все выходные стрелки блока, номер которого подписан у заполняемой строки, проследить их прямое вхождение во все блоки из списка п. 3 данного алгоритма, кроме этого же самого блока (см. п. 5 данного алгоритма), и поместить в соответствующую ячейку матрицы значение 1, если такое вхождение имеет место быть, и 0 в противном случае.

Полученная матрица будет являться матрицей смежности для орграфа, соответствующего исходной IDEF0-модели. Графическое представление такого графа может оказаться весьма перегруженным и ненаглядным, но матрицы смежности и так вполне достаточно для его структурного анализа.

Автоматизация процесса линеаризации

Автоматизация процесса линеаризации возможна с использованием не графического представления модели, а в виде IDL-файла. Формат IDL-файлов является стандартным текстовым форматом обмена информацией о моделях процессов в нотации IDEF0 между CASE-средствами различных производителей [2].

Структура IDL-файла такова:

KIT (начало файла)

IDL VERSION . . . (версия IDL)
TITLE . . . (название модели)
AUTHOR . . . (автор модели)
CREATION DATE . . . (дата создания)
PROJECT NAME . . . (название проекта)
MODEL (начало модели)
 AUTHOR . . . (автор модели)
 PROJECT NAME . . . (название проекта)
 DIAGRAM GRAPHIC . . . (начало одной из диаграмм)
 CREATION DATE . . . (дата создания)
 REVISION DATE . . . (дата пересмотра)
 TITLE . . . (название диаграммы)
 STATUS . . . (статус диаграммы)
 BOX . . . (начало одного из блоков)
 NAME . . . (название блока)
 BOX COORDINATES . . . (координаты блока)
 (DETAIL REFERENCE N . . .) (ссылка на декомпозицию)
 ENDBOX (конец одного из блоков)
 . . .
 ARROWSEG . . . (начало одной из стрелок)
 SOURCE (TUNNEL) BORDER/BOX/BRANCH/JOIN . . . (нач.)
 PATH . . . (путь прохождения)
 LABEL . . . (название стрелки)
 LABEL COORDINATES . . . (координаты стрелки)
 SINK (TUNNEL) BORDER/BOX/BRANCH/JOIN . . . (конец)
 ENDSEG (конец одной из стрелок)
 . . .
ENDDIAGRAM (конец одной из диаграмм)

...
DIAGRAM (начало глоссариев диаграмм)
GLOSSARY A0G (начало глоссария блоков)
TERM ... (один из терминов)
 ' ... ' (определение термина)
N ... (примечание к термину)
A ... (автор термина)
U ... (источник термина)
S ... (статус термина)
C ... (цвет термина)
F 1.000000 (неизвестно)

...
ENDGLOSS (конец глоссария блоков)
GLOSSARY (начало глоссария стрелок)
TERM ... (один из терминов)
 ' ... ' (определение термина)
N ... (примечание к термину)
A ... (автор термина)
S ... (статус термина)
C ... (цвет термина)

...
ENDGLOSS (конец глоссария стрелок)
ENDDIAGRAM (конец глоссариев диаграмм)
ENDMODEL (конец модели)
ENDKIT (конец файла)

Файл такой структуры может быть легко проанализирован программно. В частности, описанный выше алгоритм в его программном приложении будет выглядеть следующим образом:

1. Найти в IDL-файле все узлы DIAGRAM GRAPHIC (диаграммы), в которых есть подузлы BOX (блоки) без указания параметра DETAIL REFERENCE N (без декомпозиции).

2. Составить список всех подузлов BOX отобранных (в п. 1 данного алгоритма) узлов DIAGRAM GRAPHIC, дав им номера вида Ax.n (где Ax – это идентификатор узла DIAGRAM GRAPHIC, а n – номер его подузла BOX) и отсортировав список по ним.

3. Добавить в составленный (в п. 2 данного алгоритма) список первым элементом узел BOX 0 «Внешняя среда».

4. Построить квадратную матрицу размером N x N (где N – число элементов в списке узлов BOX после п. 3 данного алгоритма), поименовав столбцы и строки отсортированными номерами узлов BOX.

Ну и так далее.

Данный алгоритм наверняка можно было бы представить более наглядно в виде блок-схемы, однако это не меняет его сути – он пригоден для программной реализации и автоматического построения матрицы смежности для орграфа.

Программную реализацию алгоритма при желании можно дополнить функциями построения графического представления полученного орграфа, однако это нужно лишь для человека, и при этом может оказаться ненаглядным, а на возможность дальнейшего структурного анализа орграфа IDEF0-модели вообще никак не влияет.

Источники информации

1. Огрызков С. А. Применение структурного анализа для выбора варианта построения интегрированной системы управления. / Алгоритмы, методы и системы обработки данных: Сб. статей. – Муром: МИ ВлГУ, 2006.

2. *IDL Escapes*. – Platinum Technology, 1993-99.

Ключевые слова: IDEF0, иерархическая модель, линеаризация, орграф, ориентированный граф, автоматизация, структурный анализ.