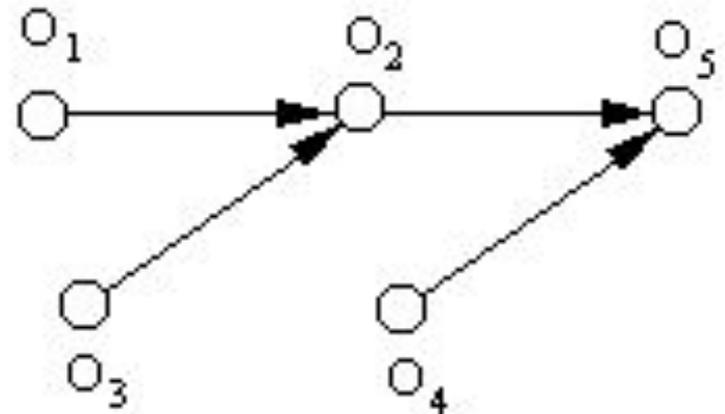


## Список дуг

Для этого способа хранения структуры составляется таблица, каждая строка в которой фиксирует дугу графа следования, причем в первом элементе строки записывается обозначение начальной вершины дуги, а во втором элементе - обозначение конечной вершины дуги. Если таблица будет представлена в памяти ЭВМ как массив размерностью  $2 \times k$ , где  $k$  - количество дуг, то массив будет занимать объем

$$V = 2k \text{ сл. ов}$$

Рассмотрим пример структуры сборочного технологического процесса.



## Список дуг для этой структуры

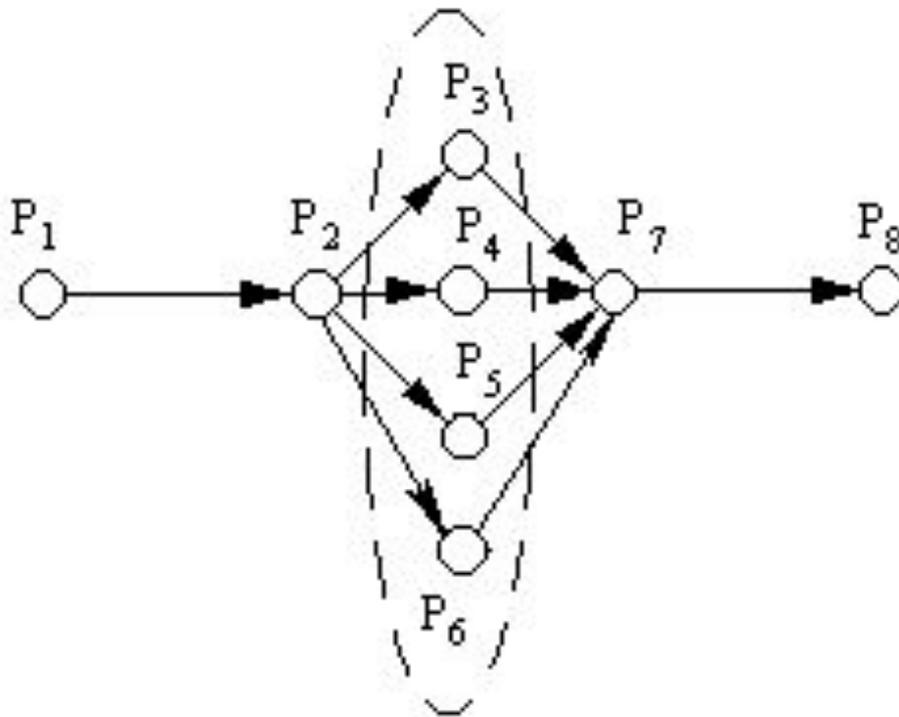
$O_i$	$O_j$
$O_1$	$O_2$
$O_3$	$O_2$
$O_2$	$O_5$
$O_4$	$O_5$

Для этого примера  $V=8$  слов. Этот способ всегда лучше, чем первые два способа в случае, когда структура представляет собой либо линейный граф, либо граф типа "дерево".

Для структуры типа "сеть" хранение в виде списка дуг выгоднее, если  $k < 1.5n$ . Ниже приведена структура операции типа "сеть".

При хранении этой структуры списком дуг -  $V=20$  слов.

Список дуг



$p_i$	$p_j$
$p_1$	$p_2$
$p_2$	$p_3$
$p_2$	$p_4$
$p_2$	$p_5$
$p_2$	$p_6$
$p_3$	$p_7$
$p_4$	$p_7$
$p_5$	$p_7$
$p_6$	$p_7$
$p_7$	$p_8$

Изменение списка дуг производится путем добавления или вычеркивания строк.

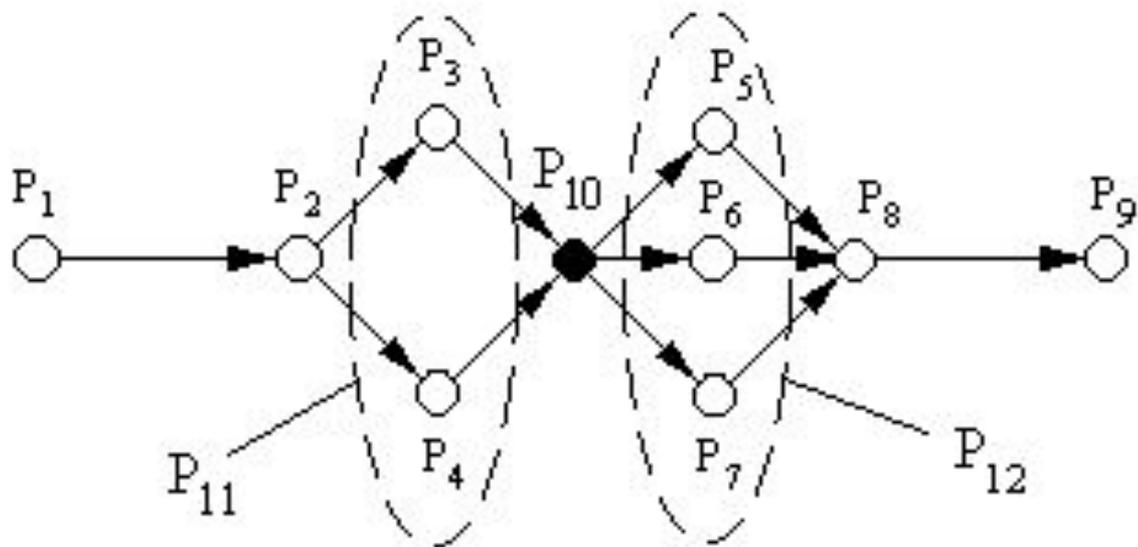
## Список вершин

Если имеется линейная структура, то ее можно записать в виде линейного списка вершин. Линейный список выражается одномерным массивом, каждый элемент которого хранит лишь наименование вершины (номер операции, номер перехода, номер перехода). Хранение списка вершин требует

**V=n** слов

Список вершин выражает только линейный граф, поэтому он всегда подходит для структуры, отражающей процесс механообработки так как структура такого процесса на уровне маршрута всегда линейна.

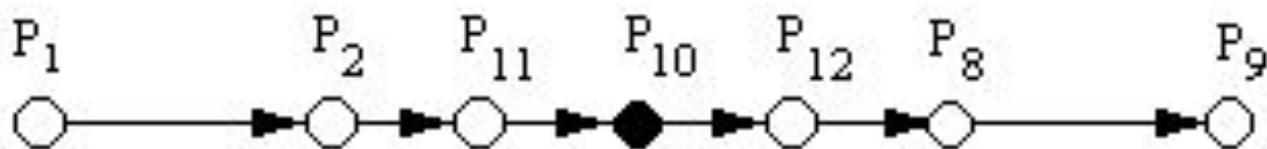
Если структура операции выражается графом типа "сеть", то, как было показано выше, линейному виду ее можно привести с помощью введения блочных переходов и псевдопереходов.



Граф приведен к линейному виду путем ввода блочных переходов  $p_{11}$  и  $p_{12}$ . После этого мы можем записать этот граф в линейном виде.

$p_1$	$p_2$	$p_{11}$	$p_{10}$	$p_{12}$	$p_8$	$p_9$
-------	-------	----------	----------	----------	-------	-------

Список вершин займет всего  $V=7$  слов, вместо 28 слов. Однако появляется недостаток - потеряна информация о переходах  $p_3, p_4, p_5, p_6$  и  $p_7$ , входящих в блочные переходы  $p_{11}$  и  $p_{12}$ .



# Процесс принятия решений в САПР ТП

Процесс принятия решения в САПР ТП обычно не является функциональным, так как при решении отдельных задач для заданных исходных данных на выходе может получиться несколько решений. В этих случаях принятие решений вместо функций можно выразить с помощью аппарата соответствий. Обобщенно, каждое соответствие  $\Gamma$  можно выразить следующим образом:

$$\Gamma = \langle \mathbf{G}, \mathbf{V}, \mathbf{R} \rangle;$$

где  $G$  - график соответствия ;

$V$  - множество входных элементов;

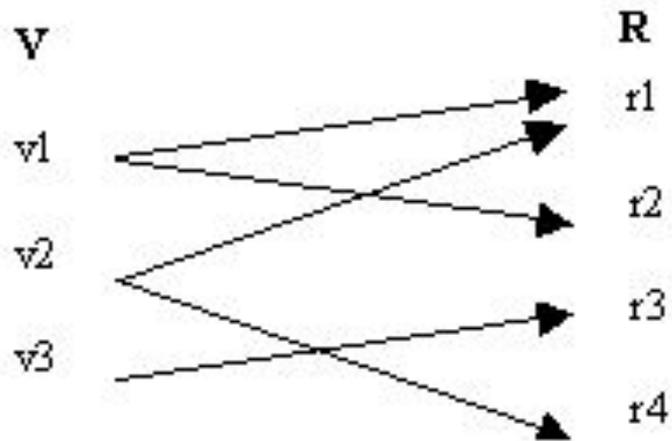
$R$  - множество выходных элементов;

$V = \{ v_i \}; i=1, nv; nv$ - количество входных элементов;

$R = \{ r_j \}; j=1, nr; nr$ -количество выходных элементов ;

$G = \{ \langle v_i, r_j \rangle \}; G$  принадлежит  $V \times R$  .

*Пример графика соответствия показан ниже*



$$G = \{ \langle v1, r1 \rangle, \langle v1, r2 \rangle, \langle v2, r1 \rangle, \langle v2, r4 \rangle, \langle v3, r3 \rangle \}$$

График соответствия может быть представлен 2 способами:

**Процедурным** - при котором алгоритм, отражающий график соответствия G, выражен с помощью операторов какого-либо языка программирования. Полученный программный модуль позволяет принимать решения в соответствии с заданным графиком.

В САПР ТП различного рода соответствий может быть достаточно много и процедурное их представление приводит к большому объему программного обеспечения. 8

Достоинство процедурного представления графика соответствия: быстрая работа модуля, принимающего решение.

Недостаток процедурного представления графика соответствия: большой объем программного обеспечения.

**Декларативным** - при котором алгоритм, отражающий график соответствия  $G$ , выражен с помощью какого-либо непроцедурного языка. График соответствия может быть представлен в виде таблицы решений, предикатной таблицы, информационной таблицы и так далее. Такая таблица и записывается в базу данных или знаний. Для работы с такого рода таблицами используется универсальный модуль, интерпретирующего типа. Этот модуль вызывает заданную таблицу из базы, проводит анализ таблицы, на основании которого выполняет определенные действия по нахождению решения.

Достоинства декларативного представления графика соответствия:

- небольшой объем программного обеспечения;
- простота введения изменений в алгоритмы путем корректировки данных, хранимых в базе данных (знаний);
- повышение адаптивных свойств САПР ТП.

Недостаток декларативного представления графика соответствия: пониженное быстродействие из-за необходимости работы с базой данных (знаний) и интерпретирующего характера универсального модуля, обрабатывающего графики соответствия.

Сам процесс принятия решения можно записать следующим образом:

$$\mathbf{MR} = \Gamma(\{v_i\});$$

MR-массив решений

Массив решений - это образ множества задаваемых входных элементов  $\{v_i\}$  относительно соответствия  $\Gamma$ .

Применительно к нашему случаю:

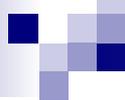
$$\begin{aligned} \mathbf{MR} &= \Gamma(\{v_2\}) \text{ или } \mathbf{MR} = \{r_1; r_4\}; \\ \mathbf{MR} &= \Gamma(\{v_1, v_2\}) \text{ или } \mathbf{MR} = \{r_1, r_2, r_4\}. \end{aligned}$$

Аппарат соответствий является одним из основных математических аппаратов применяемых в САПР. Более подробно со свойствами соответствий можно познакомиться по литературе [1, Д2].

## Уровни автоматизации

Современные САПР являются сложными системами, принципиальными особенностями которых является необходимость их адаптации к условиям предприятий, на которых они внедряются. Адаптация к условиям предприятий заключается в следующем:

- Формирование базы данных для технологического оснащения;
- Формирование базы данных для заготовок, припусков и режимов резания;
- Формирование базы данных для типовых и групповых технологических процессов;
- Формирование базы знаний для фиксации особенностей технологических процессов данного предприятия;
- Настройка САПР на те программные модули, которые будут использоваться на данном предприятии.
- Настройка САПР на те технологические документы, которые применяются на данном предприятии.



Объем информации о технологическом оснащении, заготовках, припусках весьма велик. Формирование баз данных и знаний является весьма трудоемким процессом, который может затянуться на многие месяцы и отодвинуть внедрение САПР на предприятии. Поэтому при внедрении САПР целесообразно ориентироваться на концепцию последовательной автоматизации.

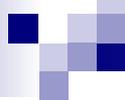
**Концепция последовательной автоматизации** заключается в следующем:

- Внедрение на первых этапах САПР с низким уровнем автоматизации;
- Формирование баз данных параллельно с эксплуатацией САПР;
- Настройка на технологические особенности предприятия так же вести параллельно с эксплуатацией САПР.

Будем различать три уровня автоматизации, как показано в нижеприведенной таблице:

<b>Уровень</b>	<b>Назначение</b>
Первый	Автоматизация оформления технологической документации (маршрутные, операционные карты и другие документы).
Второй	Автоматизация поиска и расчетных задач.
Третий	Автоматизация принятия сложных логических решений

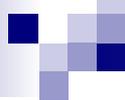
**Первый уровень автоматизации** - автоматизация низкого уровня, при которой автоматизировано только оформление технологической документации (маршрутные, операционные карты и другие документы). Бланк документа выводится на экран монитора и технолог в режиме диалога заполняет этот документ. На первых этапах внедрения параллельно с эксплуатацией САПР с низким уровнем проектирования формируется база данных с технологическим оснащением.



Второй уровень автоматизации - автоматизация среднего уровня, который достигается, когда базы данных частично сформированы и начинают работать поисковые и расчетные модули. Чем больше заполнена база данных, тем эффективнее начинает работать САПР.

Работа поисковых модулей основана на использовании информационно-поисковой системы (ИПС), при этом условие поиска технолог вводит в режиме диалога. Условия поиска, которые являются стабильными, хранятся в базе знаний.

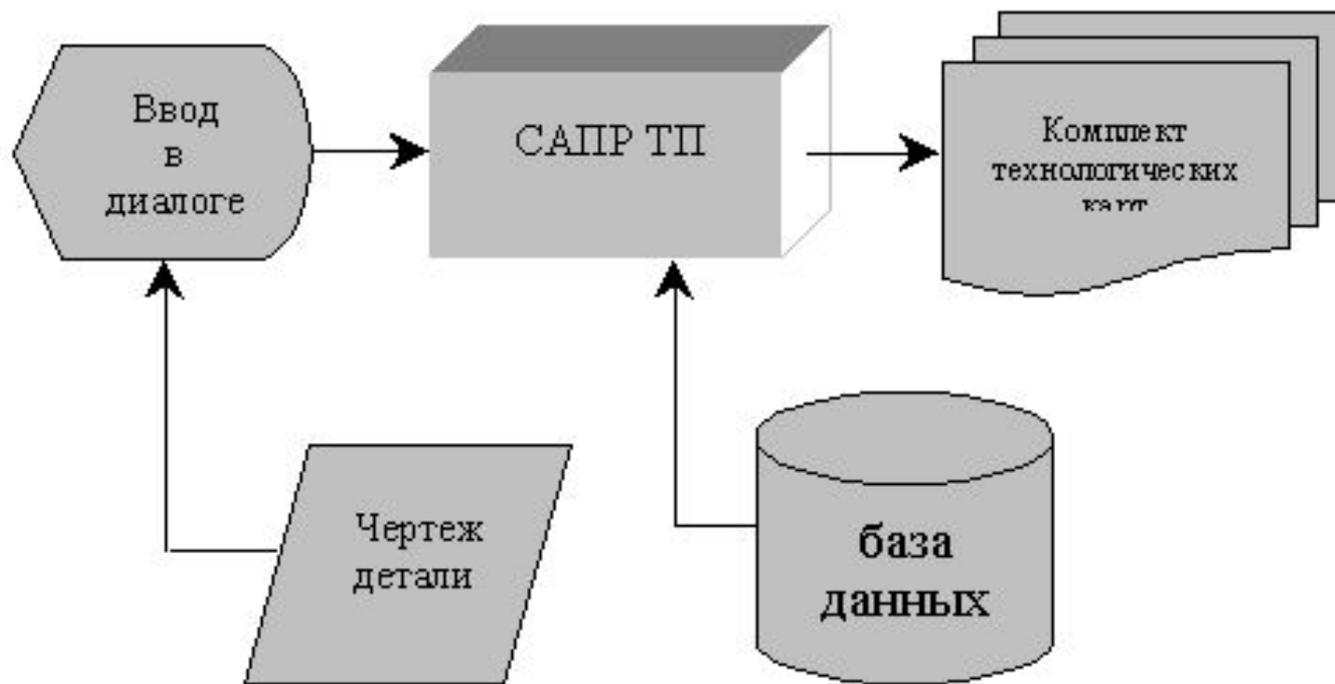
Расчетные модули, например модули расчета припусков, расчета режимов резания и норм времени, начинают работать, когда сформированы базы данных с нормативно-справочной информацией.



Третий уровень автоматизации - автоматизация высокого уровня, который достигается, при заполнении базы знаний. В этом случае становится возможным автоматизированное принятие сложных логических решений, связанных, например, с выбором структуры процесса и операций, назначением технологических баз и другие подобные задачи.

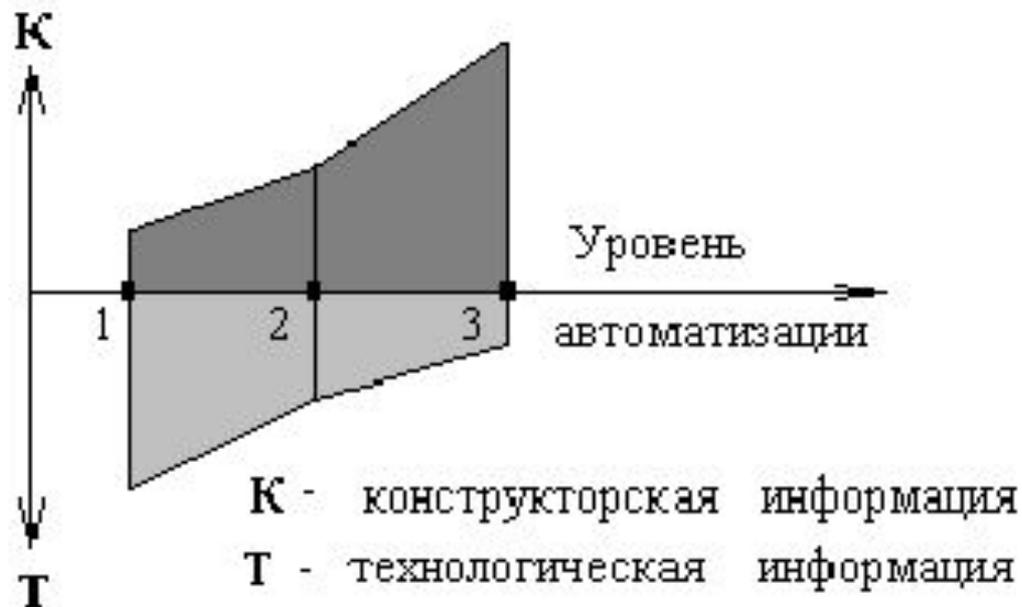
Методику принятия таких решений полностью автоматизировать не удастся, поэтому режим диалога остается и на третьем уровне автоматизации.

Процесс проектирования в САПР ТП представляет собой сложный процесс переработки конструкторской информации, заданной в чертеже детали, в технологическую информацию, которая затем фиксируется в технологической документации.



Чертеж детали не обязательно хранится на твердой копии (бумаге). Если на предприятии функционирует конструкторский САПР, то чертеж детали может храниться в электронном архиве.

Соотношение вводимой технологической и конструкторской информации зависит от уровня автоматизации.



При низком уровне автоматизации, когда технолог заполняет бланк технологической карты на экране монитора, он сам проектирует технологический процесс, т. е. сам выполняет переработку конструкторской информации в технологическую и заносит технологическую информацию в технологическую карту. Объем вводимой конструкторской информации незначителен.

На среднем уровне автоматизации проектирования объем вводимой конструкторской информации резко увеличивается. Например, для поиска технологического оснащения необходимо иметь информацию о конструктивных элементах детали. Поэтому на этом уровне автоматизации проектирования ТП осуществляется кодирование чертежа детали и формирование параметрической модели детали (ПМД)\*.

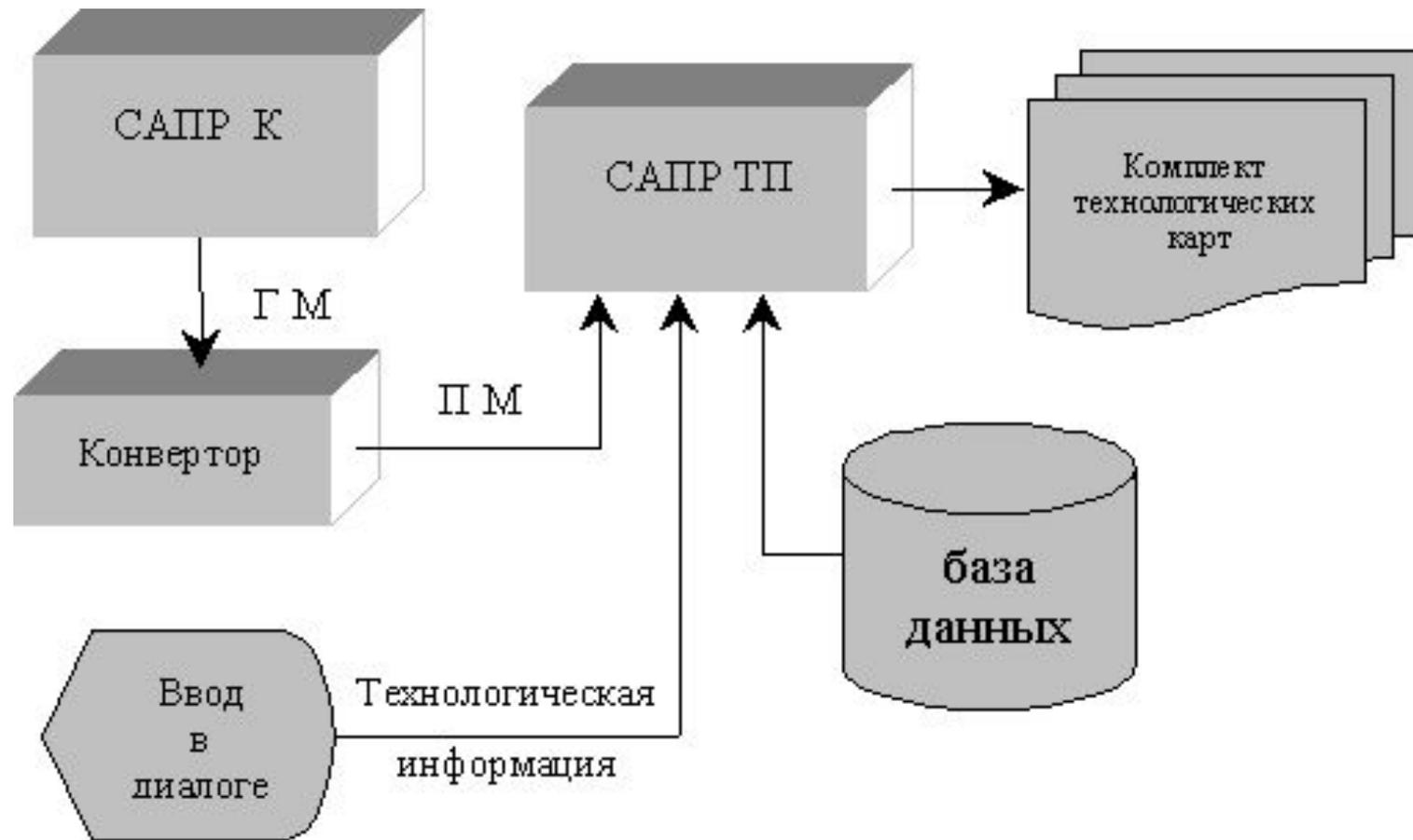
Расчетные задачи так же требуют информацию и детали и операционных заготовках. Объем вводимой технологической информации на этом уровне уменьшается, так как часть конструкторской информации перерабатывается в технологическую.

На высоком уровне автоматизации для проектирования ТП требуется еще более подробная информация о детали, т.е. нужно вводить параметрическую модель детали с наибольшей степенью детализации, в тоже время объем вводимой технологической информации на этом уровне еще больше уменьшается.

Кодирование чертежа детали и создание параметрической модели с высоким уровнем детализации описания детали является достаточно трудоемким процессом, снижающим эффективность САПР ТП.

Коренное решение задачи получения параметрической модели детали заключается в интеграции конструкторской и технологической САПР. Интеграция заключается в создании конвертора, преобразующего графическую модель (ГМ) детали в параметрическую, как будет показано ниже.

Если автоматизированное конструирование изделия на предприятии не ведется, то графическая модель детали (в формате CAD - системы или в виде нейтрального формата передачи данных IGES или STEP) выбирается из электронного архива данных



Преобразование графической модели (ГМ) детали  
в параметрическую (ПМ)

## **ВЫВОД**

Использование концепции последовательной автоматизации позволяет:

- осуществить быстрое внедрение САПР ТП в технологическую подготовку производства на предприятии;
- выполнять параллельно адаптацию САПР ТП к условиям предприятия путем формирования баз данных и знаний;
- последовательно повышать уровень автоматизации проектирования технологических процессов;
- по мере повышения уровня автоматизации уменьшать трудоемкость создания параметрической модели путем интеграции с САПР конструирования изделий.