



# *Методи на Транслация*

Транслация.

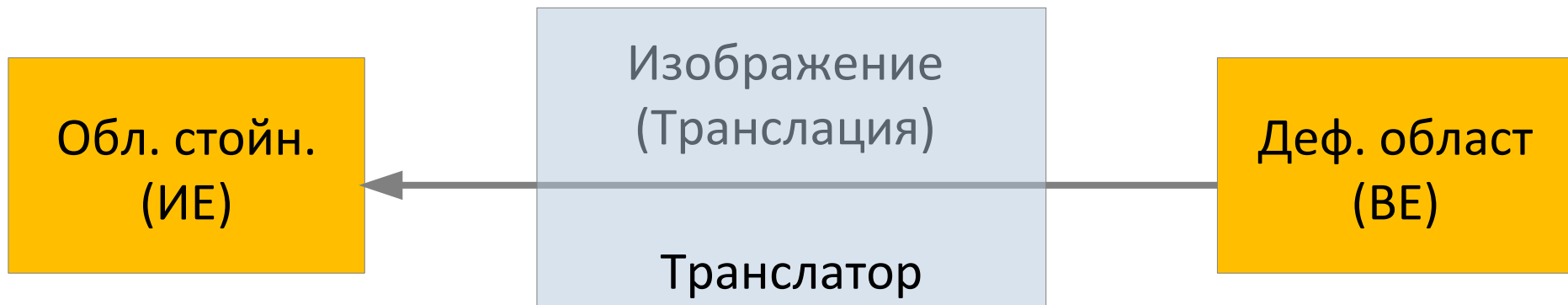
Методи и схеми на транслация.

Синтактично управляема транслация.

# *Транслация*

# Транслация

**Транслация** наричаме процесът на превод на програма, написана в термините на езика на една ВИМ (или входен език) в еквивалентна програма в термините на друга ВИМ (или изходен език).



# Цел на транслацията

Крайна цел на транслирането е преведената програма да се изпълни.

Следователно изпълнението на преведената програма може да се разгледа или като възможен етап в самия процес на транслация или като своеобразен превод от езика на входните данни в езика на изходните данни т.е. преведената програма е преводач.

Има две основни схеми на транслация съобразно това дали транслаторът е и изпълнител или не:

- ❖ Компиляция (Транслация в собствен смисъл);
- ❖ Интерпретация (Програмно моделиране);

# Видове транслятори

- ❖ Транслатор (Конвертор, Компилятор, Макро-асемблер, Асемблер, Свързващ редактор (Linker), Зареждач (Loader), Изпълнител, Крос-транслатор);
- ❖ Интерпретатор (Интерпретатор в собствен смисъл, Емулатор);
- ❖ Хибриден компилатор;

# Интерпретация vs Компиляция

Компиляция	Интерпретация
Превод	Превод и изпълнение
Всяка входна програма се обработва в съответствие с т.нар. правило за четене	Входната програма се обработва в съответствие с т.нар. логика на изпълнение
Всяка инструкция във входната програма се обработва фиксиран брой пъти (много често само веднъж), определен от избрания метод на трансляция	Всяка инструкция във входната програма се обработва толкова пъти колкото налагат данните
Входната програма се превежда веднъж	Входната програма се превежда при всяко изпълнение
Преведената програма заема доп. място	Не е необходимо допълнително място
Преведената програма се изпълнява много по-бързо от ев. интерпретираната	Изпълнението на входната програма е много по-бавно от евентуално компилир.

# Хибридни схеми за реализация

- ❖ Очевидно двете схеми имат почти противоположни предимства (по отношение на памет и скорост на изпълнение);
- ❖ Тези две чисти схеми се използват рядко в такъв вид;
- ❖ Съвременните хибридни компилатори действат по схемата: Компилатора превежда входния език до машинно-ориентиран език, а след това той се интерпретира (емулира)
- ❖ Примери за работещи по тази схема системи са Java и Java-bytecode, изпълняван от Java виртуалната машина; Аналогично .NET базираните езици и MSIL, изпълняван от .NET виртуалната машина;

# *Етапи на Транслация*



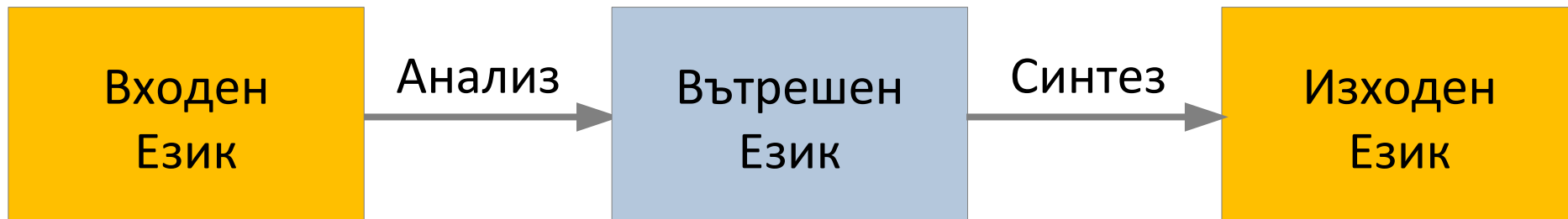
# Етапи на трансляция

- ❖ **Анализ** на входната програма:
  - ❖ Лексикален анализ – разпознава лексемите от входния поток;
  - ❖ Синтактичен анализ – разпознава синтактичните структури;
  - ❖ Семантичен анализ – придаване на смисъл на елементите (лексика и синтаксис), поддържане на необходимите даннови структури (таблица на символите, абстрактно синтактично дърво), проверка на семантични и други ограничения;
- ❖ **Синтез** на изходните програми:
  - ❖ Оптимизация – машинно независима и машинно зависима;
  - ❖ Генерация на код – цели произвеждане на изходната програма на базата на резултатите събрани от анализа;



# Етапи на трансляция

**Транслация** наричаме процесът на превод на програма, написана в термините на езика на една ВИМ (или входен език) в еквивалентна програма в термините на друга ВИМ (или изходен език).



# *Методи и Схеми на Транслация*

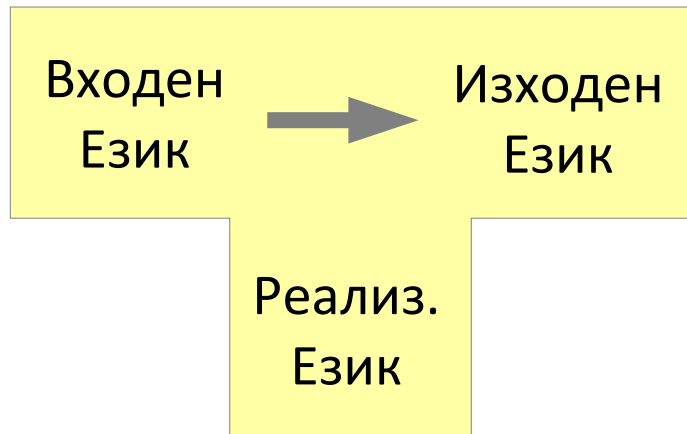
# Методи на транслация

- ❖ Метод на транслация наричаме:
  - ❖ Отговор на въпроса как да извършим превода;
  - ❖ Алгоритъм, чиято реализация извършва превода;

За да се овладее методите на транслация е необходимо да се знаят отговорите на следните въпроси:

- ❖ Що е компютър (изчислителна машина)?
- ❖ Как функционира компютъра?
- ❖ Що е език (Език за програмиране – ЕП)?
- ❖ Какви видове ЕП има?
- ❖ Какви са структурата и съставът на всеки ЕП?
- ❖ Как се описва един ЕП?

# T-диаграми

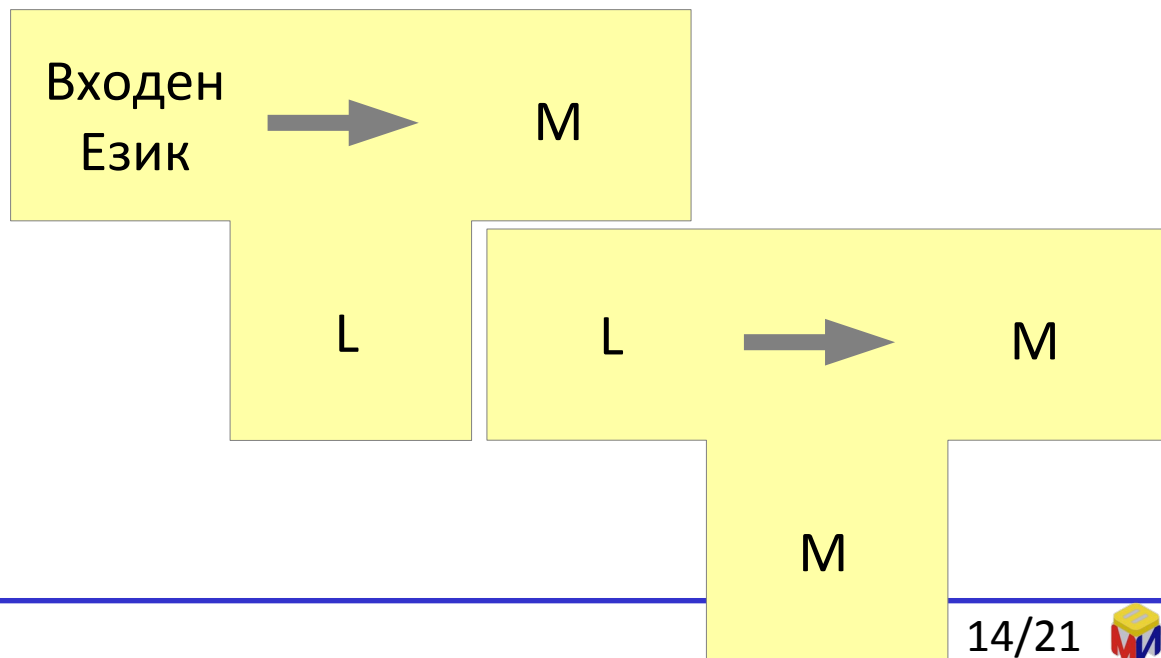


- ❖ Всеки разработчик на транслятор използва 3 езика: входен (ВЕ), изходен (ИЕ) и реализационен (РЕ);
- ❖ За РЕ трябва да съществува изпълнител т.е. ВИМ на който транслятора да се изпълни. Такива езици ще отбелязваме с L;
- ❖ Често ВЕ е ЕП>3 ниво, а ИЕ е машинен (ще отбелязваме с М);
- ❖ Рядко ВЕ = ИЕ;
- ❖ Може РЕ = ВЕ;

# Транслатор

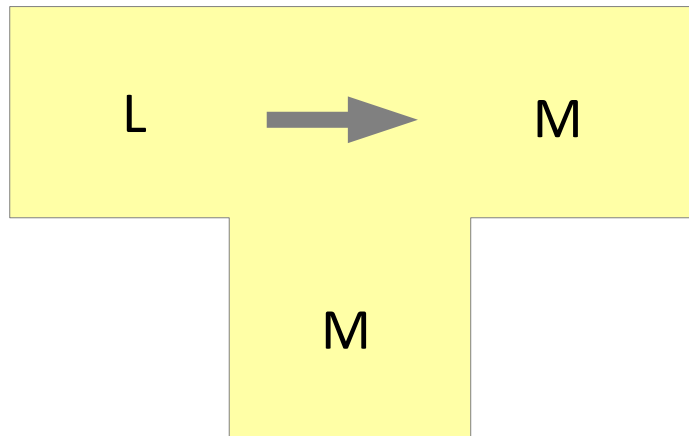
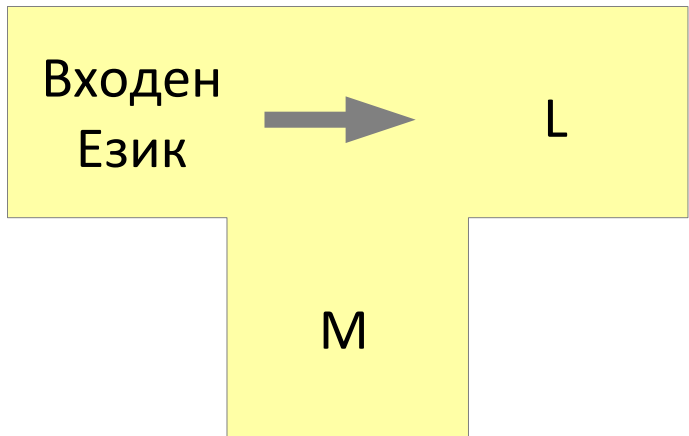
Обикновено терминът транслатор се употребява, когато:

- ❖ PE = ИЕ;
- ❖ от PE съществува транслатор до ИЕ. Втората схема показва, че в този случай PE може да се смята за еквивалентен на машинния;



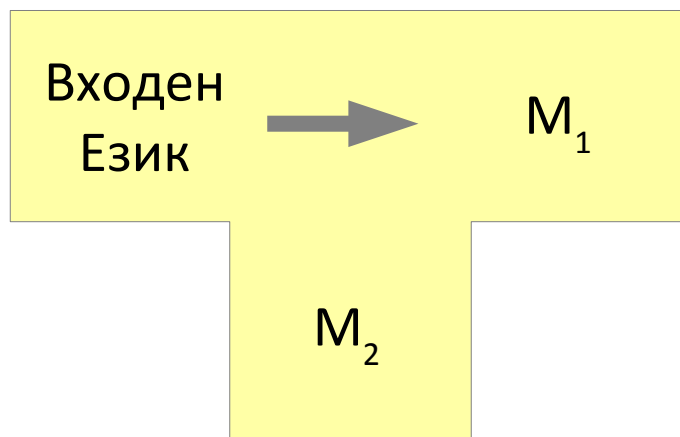
# Схеми

- ❖ Ако изходния език е от високо ниво, но за него има транслятор;
- ❖ Терминът транслятор употребяваме, когато  $PE = IE = M$ .



# Схеми

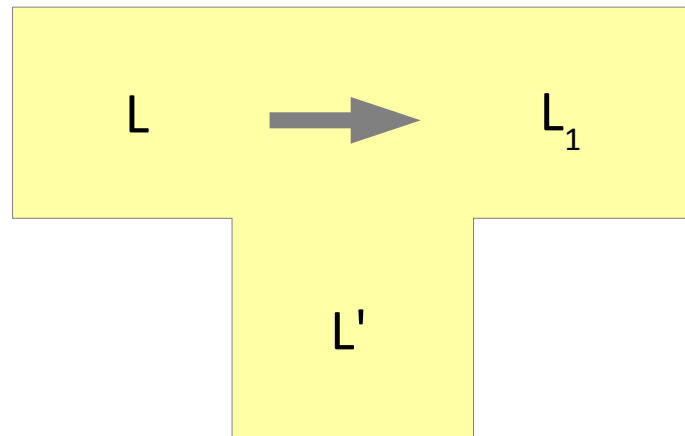
- ❖ Когато РЕ и ИЕ са машинно-ориентирани, но не съвпадат, се употребява терминът крос-транслатор;





## Схеми

- ❖ Разработката на транслятор на машинен език е трудна задача, затова обикновено РЕ е език от високо ниво;
- ❖ Възможно е РЕ да е подмножество на ВЕ, като тогава казваме че ВЕ е диалект на РЕ;
- ❖ Това е често срещана идея и е начин за нарастване на изразителната сила на РЕ със „собствени сили“;



$$L' \subseteq L$$

# *Синтактично Управляема Транслация*

# Синтактично управляема трансляция

Голяма част от съвременните транслатори са така наречените „**Синтактично управляеми**“. Това означава, че:

- ❖ Водещ е синтактичният анализатор;
- ❖ Той се обръща към лексикалния при необходимост от нова лексема за продължаване на анализа на синтаксиса;
- ❖ Разпознаването на дадено синтактично правило води до задействане на семантичния анализ;
- ❖ След успешно преминаване през семантичните проверки и построяване на таблици се извиква генерираща процедура свързана с правилото, която генерира изходния или междинен код, ако ще има етапи на оптимизация;
- ❖ Крайния резултат се записва или изпълнява;





*Въпроси?*  
*apenev@uni-plovdiv.bg*