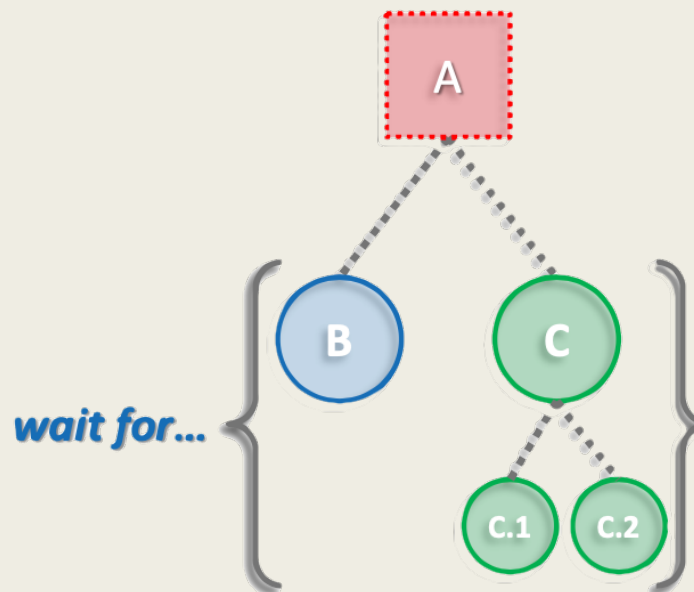


Расширение OpenMP 4.0 – директивы

#pragma omp taskgroup [reduction(id: items)]

- Определяет группу задач
- Обеспечивает синхронизацию завершения всех задач в группе

```
#pragma omp parallel
#pragma omp single
{
    #pragma omp taskgroup // A
    {
        #pragma omp task // B
        { ... }
        #pragma omp task // C
        { ... #C1; #C2 ... }
    }
}
```



Расширение OpenMP 4.0 – отмена выполнения

```
#pragma omp cancel [construct-type] [if(clause)]
```

- Запрос на отмену выполнения самого низшего уровня регионов
 - Исполняется «как можно скорее»
 - До OpenMP 3.1 – либо не выполняться совсем, либо выполняться до конца

```
#pragma omp cancellation point [construct-type]
```

- Проверка на «состояние отмены»
 - Добавляет возможность задать условия для возможности отмены
 - Необходима грамотная очистка данных при отмене

Расширение OpenMP 4.0 – отмена выполнения

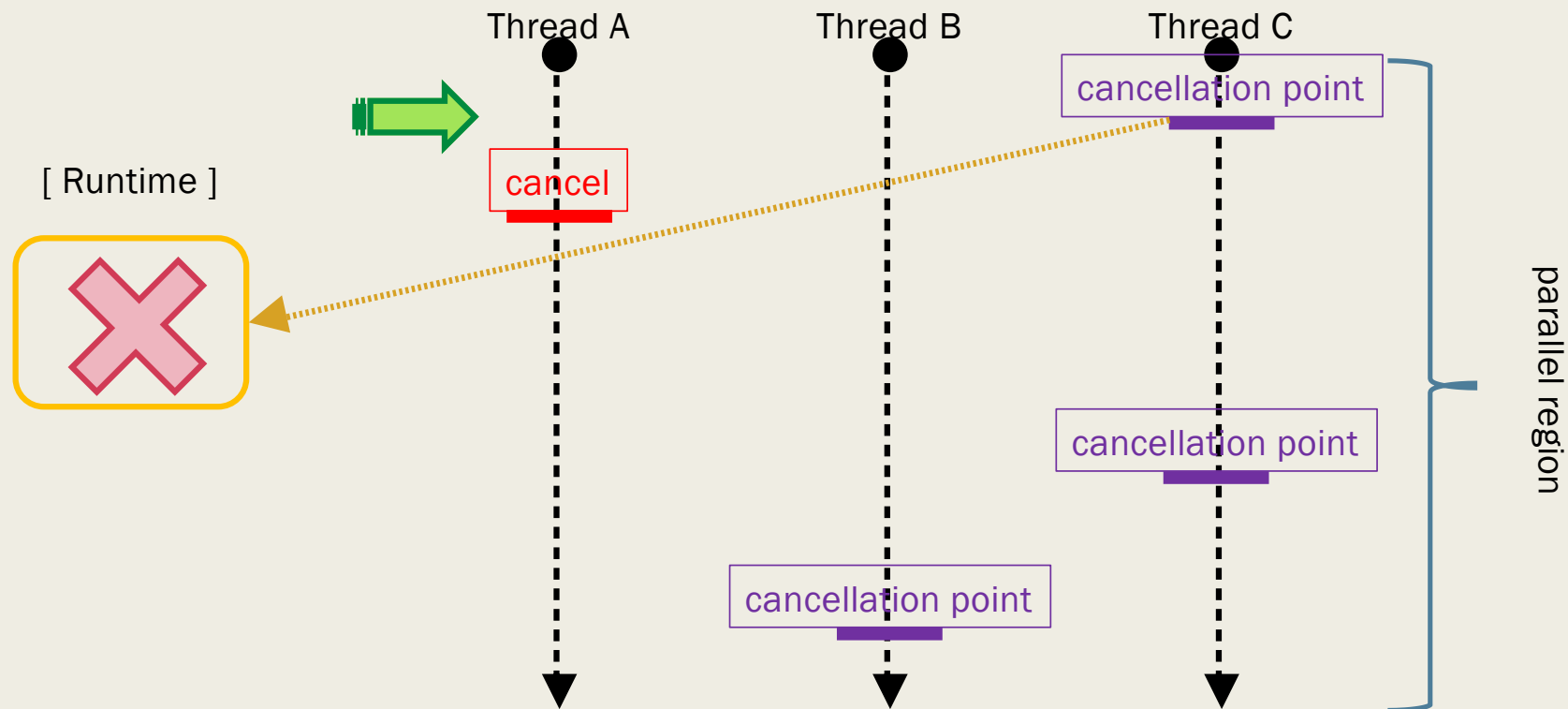
```
#pragma omp cancel [construct-type] [if(clause)]
```

```
#pragma omp cancellation point [construct-type]
```

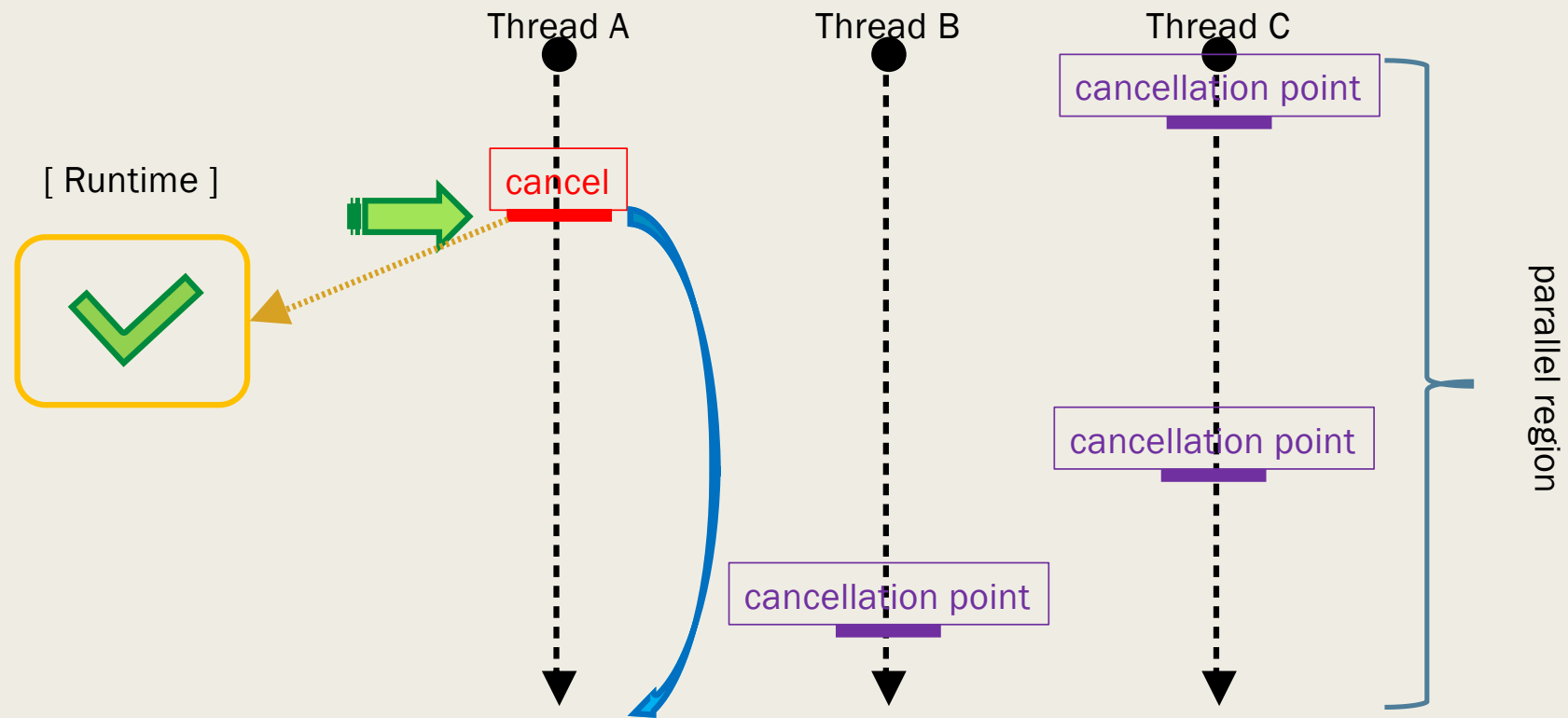
■ Типы:

- *parallel*
- *sections*
- *for*
- *taskgroup*

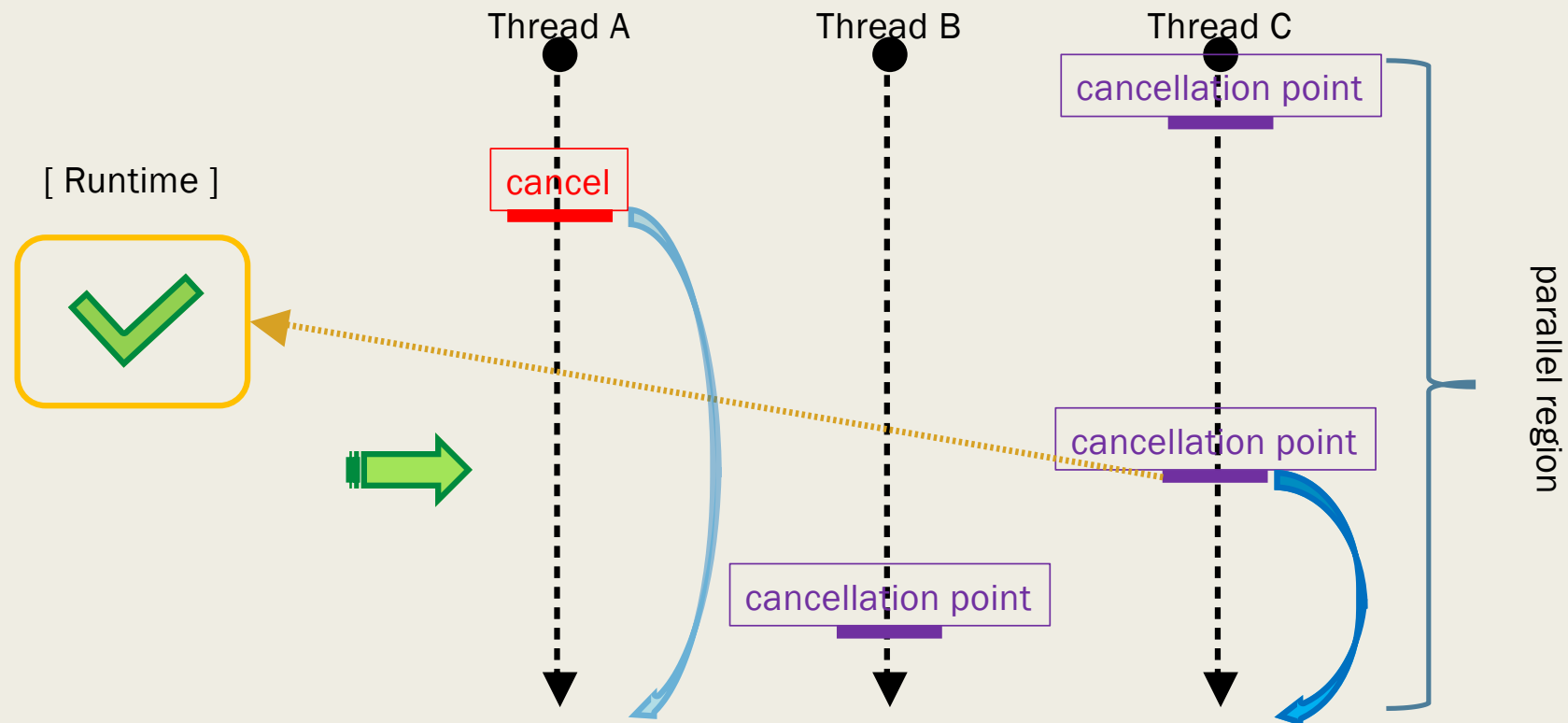
Расширение OpenMP 4.0 – отмена выполнения



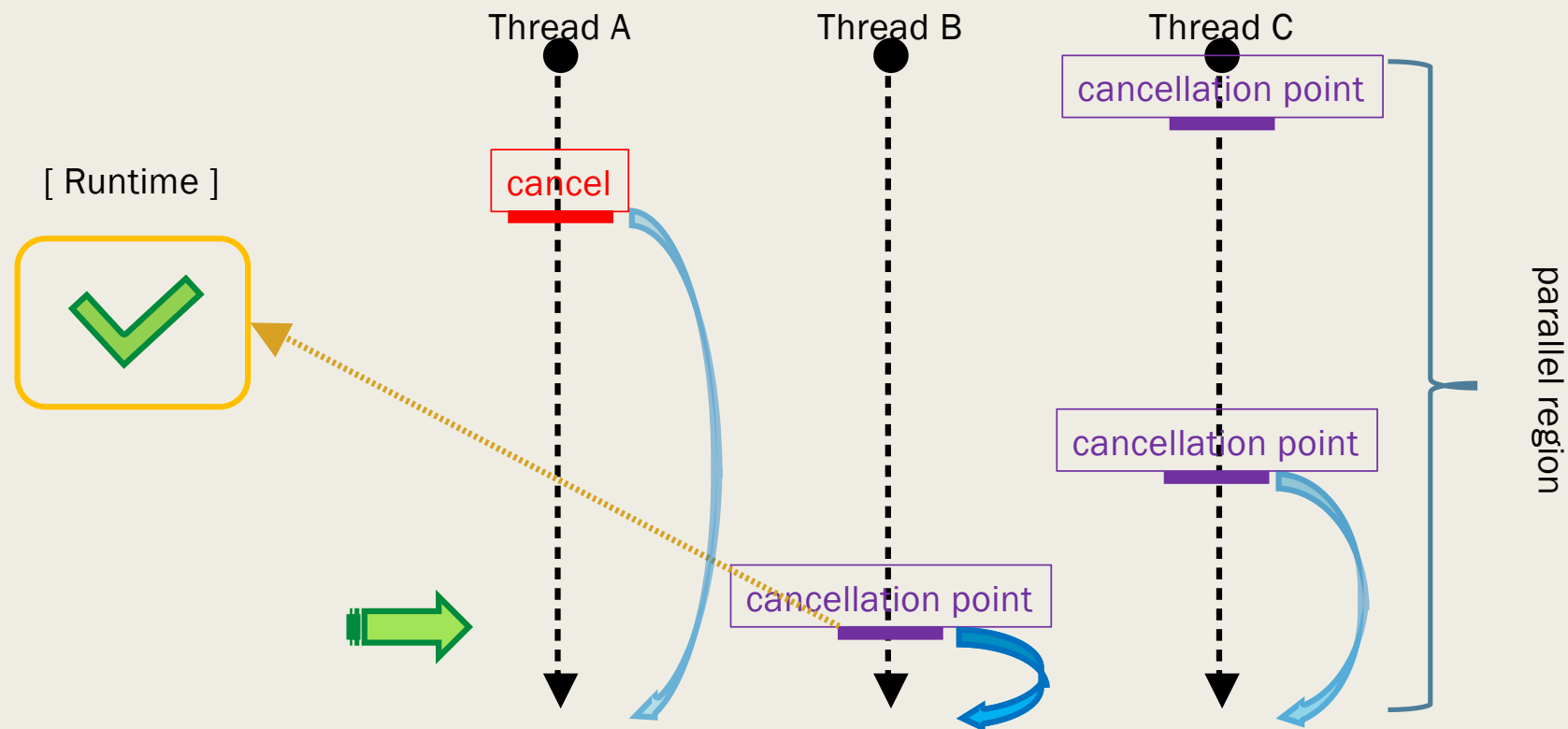
Расширение OpenMP 4.0 – отмена выполнения



Расширение OpenMP 4.0 – отмена выполнения



Расширение OpenMP 4.0 – отмена выполнения



Расширение OpenMP 4.0 – отмена заданий

- Отмена выполнения действует только на задания, объединенные в ***taskgroup***
 - Уже выполняемые задания продолжают до завершения (если не объявлены *cancellation points*)
 - Задания, не начавшие выполнение, будут отменены (и обозначены как успешно завершённые)
- Отмена заданий также происходит при отмене родительского параллельного региона (кроме заданий с разделением работы)

Расширение OpenMP 4.0 – отмена заданий

```
tree_t * search_parallel(tree_t *tree, int value) {  
    tree_t * found = NULL;  
    #pragma omp parallel shared(found,tree,value)  
        #pragma omp master  
            #pragma omp taskgroup  
                found = search_sequent(tree, value);  
    return found;  
}
```

Расширение OpenMP 4.0 – отмена заданий

```
tree_t * search_sequent(tree_t *tree, int value) {
    tree_t * found = NULL;
    if(tree) {
        if(tree->value == value) {
            found = tree;
        } else {
            #pragma omp task shared(found)
            {
                tree_t * found_left;
                found_left = search_sequent(tree->left, value);
                if(found_left) {
                    #pragma omp atomic write
                    found = found_left;
                    #pragma omp cancel taskgroup
                }
            }
            #pragma omp task shared(found)
            {
                tree_t * found_right;
                found_right = search_sequent(tree->right, value);
                if(found_right) {
                    #pragma omp atomic write
                    found = found_right;
                    #pragma omp cancel taskgroup
                }
            }
            #pragma omp taskwait
        }
    }
    return found;
}
```

Расширение OpenMP 5.0 – расширения

#pragma omp task depend(dependency-type: lvalue)

- Lvalue-expressions в списке зависимостей:
 - *#pragma omp task depend(inout: a.x, v.operator[](*i*))*

#pragma omp taskwait (dependency-type: list)

- Добавление зависимостей в конструкцию *taskwait*:
 - *#pragma omp taskwait (in: x)*