Seams in Compiler

Software Testing

Instructor: Dr. Sharbaf

Alireza Dastmalchi Saei

What are Seams?

1. A line where two piece of fabrics are stitched together.

2. A seam in software as a place where two parts of the code meet and where something else can be injected.

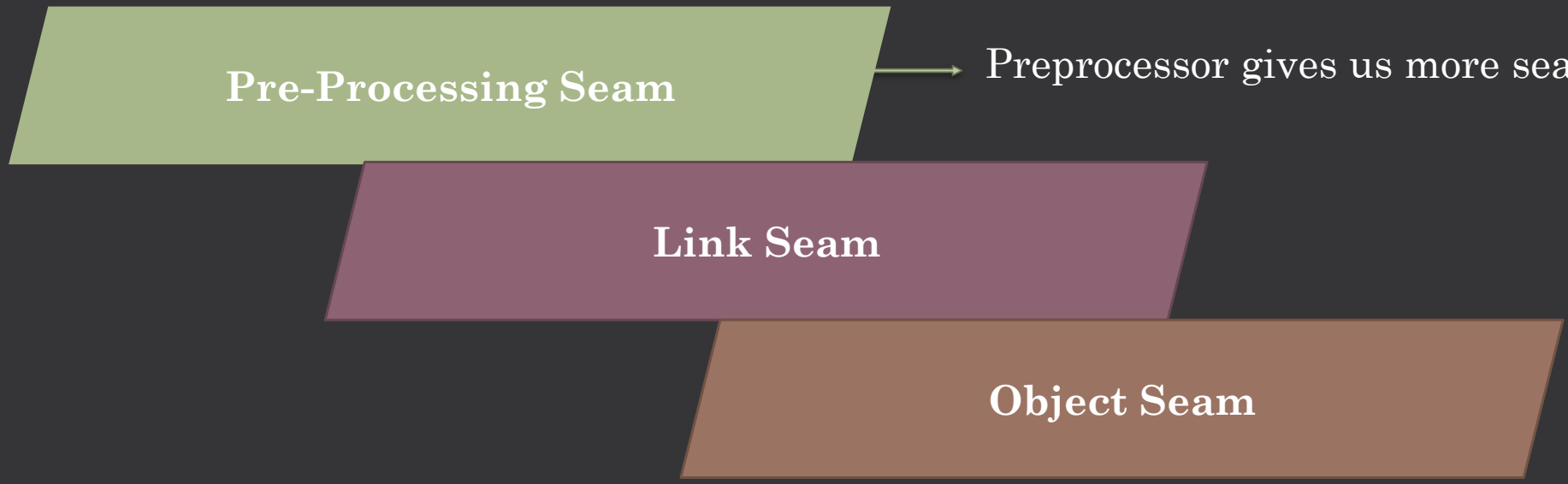
Seam Enabling Point

When you have a Seam:

- A place in code where the behavior can change.
- We can't go to that place and change the code to test it.

Every seam has an enabling point, a place where you can make the decision to use one behavior or another.

Types of Seams



Pre-Processing Seam

C and C++ provides with preprocessing tool.

A macro preprocessor runs before the compiler

Use the preprocessing seams to replace calls to another independent piece of code.

Conditional Compilation

Conditional Compilation Directives:

- #ifdef
- #ifndef
- #if
- #endif

Select which section of codes are included in the final compiled program.

```c
#include <stdio.h>

#define FEATURE_ENABLED

int main() {
#ifdef FEATURE_ENABLED
    printf("Feature is enabled.\n");
#else
    printf("Feature is disabled.\n");
#endif

    return 0;
}
```

Macro Replacement

Macros can be used for code replacement during the pre-processing phase.

Whenever the “MAX” is encountered in the code, it gets replaced by corresponding macro expansion.

```c
#include <stdio.h>

#define MAX(a, b) ((a) > (b) ? (a) : (b))

int main() {
    int x = 5;
    int y = 10;
    int max = MAX(x, y);

    printf("The maximum between %d and %d
is %d.\n", x, y, max);

    return 0;
}
```

Macro Creation

```
#include <DFHLItem.h>

extern int db_update(int, struct DFHLItem *);

void account_update(
   int account_no, struct DHLSRecord *record, int activated)
{
    if (activated) {
       if (record->dateStamped && record->quantity > MAX_ITEMS) {
            db_update(account_no, record->item);
       } else {
            db_update(account_no, record->backup_item);
       }
    }
  db_update(MASTER_ACCOUNT, record->item);
}
```

Macro Creation

```
#include <DFHLItem.h>

extern int db_update(int, struct DFHLItem *);

#include "localdefs.h"

void account_update(
   int account_no, struct DHLSRecord *record, int activated)
{
    if (activated) {
       if (record->dateStamped && record->quantity > MAX_ITEMS) {
            db_update(account_no, record->item);
       } else {
            db_update(account_no, record->backup_item);
       }
    }
  db_update(MASTER_ACCOUNT, record->item);
}
```

```
#ifdef TESTING
...
struct DFHLItem *last_item = NULL;
int last_account_no = -1;

#define db_update(account_no,item)\
{last_item = (item); last_account_no = (account_no);}
...
#endif
```


Pre-Processing Seam

The End

Thanks for your attention!