Розвиток стратегій проектування Александреску. Бібліотека Loki23

Виконав: Трохимчук Артем Андрійович[1]
Науковий керівник: Бублик Володимир Васильович[2]

[1]студент 4-го року навчання, спеціальності: 121 Інженерія програмного забезпечення

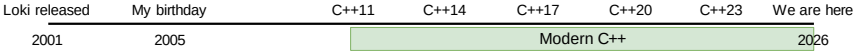
[2]доцент, кандидат фізико-математичних наук

Loki: дизайн на основі стратегій

```
struct CloneByValue {
  template<class T>
  static T clone(const T& value) {
    return value;
  }
};

template<class T, class ClonePolicy>
class Host : private ClonePolicy {
public:
  explicit Host(const T& value)
    : value_(ClonePolicy :clone(value)) {}

private:
  T value_;
};

using ValueHost =
  Host<int, CloneByValue>;
```

1. Host-клас не фіксує поведінку всередині реалізації
2. Політика є окремим типом з потрібною операцією
3. Конкретний тип утворюється композицією host + policy
4. Вибір поведінки відбувається на етапі компіляції
5. Гнучкість досягається без runtime-диспетчеризації

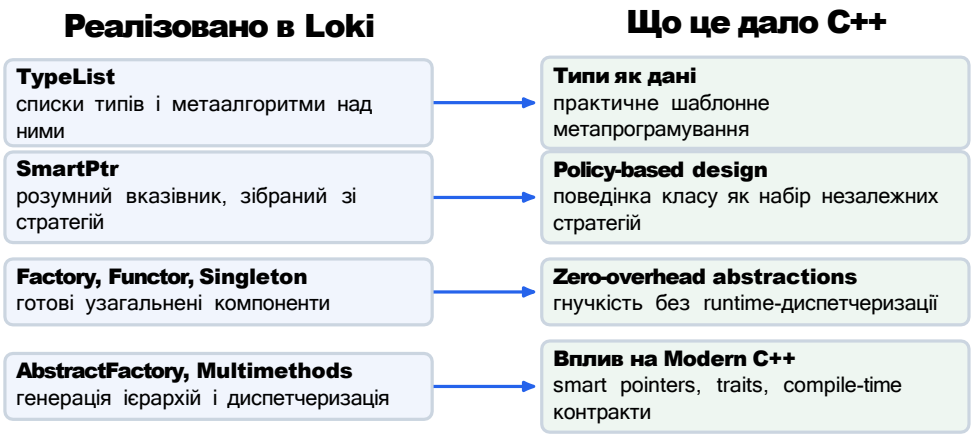
Modern C++: еволюція мови



C++98/03
макроси, рекурсивні шаблони, SFINAE

Modern C++
варіативні шаблони, концепти, модулі

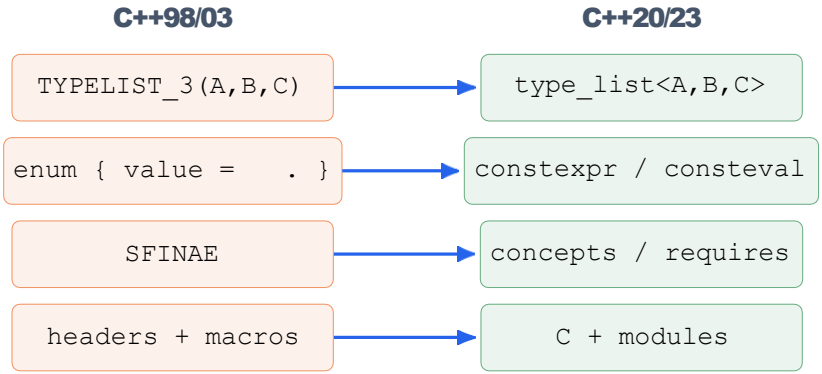
Вплив Loki на C++



Архітектура Loki23

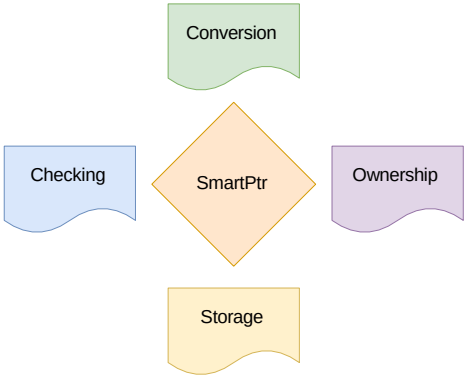
Модулі задають межі компонентів, а `type_list` з'єднує метапрограмні частини бібліотеки.

Modern C++ прибирає технічний борг



Ідея Loki збережена, але механізми стали явними для розробника

SmartPtr: вінець стратегій



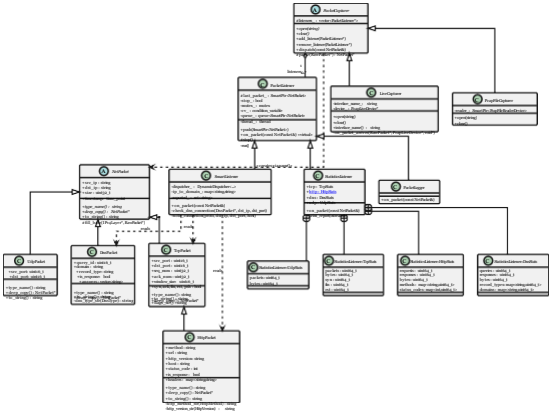
`SmartPtr<T, Own, Check, Store>`

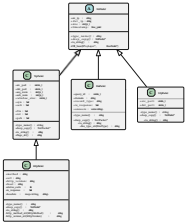
об’єкти пам’яті

файлові дескриптори

Unix-сокети

Практична демонстрація: netprobe





`pcap` або мережевий інтерфейс

`HierarchyFactory` створює компоненти

`SmartPtr<NetPacket>` між потоками

динамічна і статична диспетчеризація

Loki23 не просто оновлення

10
зіставних тестів

6
перемог Loki23

1.31x
швидше сумарно

Наукова новизна

1. Стратегії як іменовані концепти
неявні контракти Loki → `Checking`, `Ownership`, `Storage`, `Conversion`, `Creation`, …

2. SmartPtr: 4-та вісь + lock-free
додано `StoragePolicy` (FD, сокети) + `RefCountedMT` на `std :atomic acq_rel`

3. Singleton: коректний DCL
`atomic` + `acquire/release` + `CreateStatic` на placement-new у `alignas`-сховищі

4. Multimethods відкритого світу
fold-вирази + `type_index`-диспетчер; `FIRST_MATCH/ALL_MATCHES`, `NoOp/Exception/Log`

5. DI-контейнер — нова компонента
відсутня в Loki; `GenLinearHierarchy` + `LazyHolder` + маркери `From<T>`

Висновки

Наукова новизна

- Стратегії формалізовано як **C++20-концепти**
- `SmartPtr`: **4-та вісь** + lock-free керування лічильником
- Multimethods **відкритого світу**: статичний + динамічний диспетчери
- **DI-контейнер** – нова компонента (Loki її не мала)

Практичні результати

- **Loki23**: 7 компонентів на C++23-модулях (`type_list`, `SmartPtr`, Factory, Hierarchy, Singleton, Multimethods, DI)
- Перевірено в реальному сценарії `netprobe`: пакети, ресурси, потоки, диспетчеризація
- **6/10** перемог; сумарно **1.31x**; динамічні мультиметоди до **58.2%**

Дякую за увагу

Loki23

Ідеї Александреску перенесено з C++98 у типобезпечну й практичну модель Modern C++

a.trokhymchuk@ukma.edu.ua