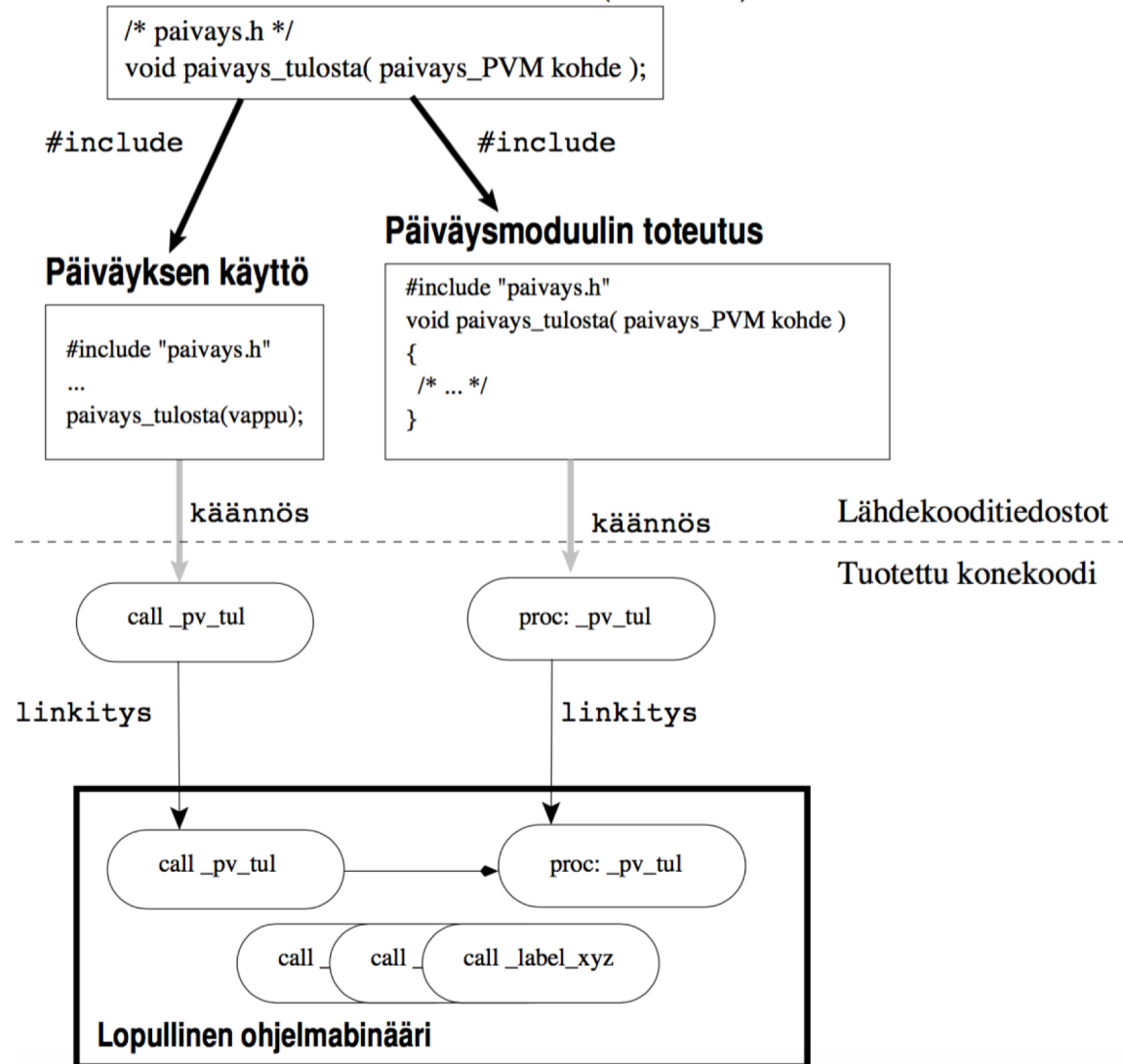


Modulaarisuus Nimiavaruudet

18.9.2018

Modulaarisuus

Kuva 1.6 s. 35 (1.6 s. 43):



Nimikonfliktit

- Ongelma:
 - Sama nimi käytössä eri merkityksissä ⇒
nimiavaruuden roskaantuminen ja
“nimitörmäykset”
- Ratkaisu: nimiavaruus

```
Paivays::tulosta();  
Kirjastonkirja::tulosta();
```

Rajapinta nimiavaruudella

```
#ifndef PAIVAYS_HH
#define PAIVAYS_HH

namespace Paivays {
    struct Pvm { int p_, k_, v_; };
    Pvm luo( int paiva, int kuukausi, int vuosi );
    void tulosta( Pvm kohde );
    ...
}
#endif
```

Nimiavaruuksien edut

- Nimikonfliktit vähenevät hierarkisuuden seurauksena
- Hierarkia näkyvässä ohjelman syntaksissa (:: - operaattori)
- Moduulin (nimiavaruuden) sisällä silti käytössä lyhyet nimet

Nimiavaruus `std`

- Standardi määrittelee yhden nimiavaruuden: `std`, jonka sisällä ovat kaikki ISO C++ -nimet

`std::vector`

`std::cin`

...

Lyhyet nimet: `using`

- `using`-lause "nostaa" nimiavaruuden sisällä olevan nimen käytettäväksi nykyisessä näkyvyysalueessa

using-lause

```
#include "paivays.hh"
#include <iostream>
void kerroPaivays( Paivays::Pvm p ) {
    using std::cout;
    using std::endl;
    using namespace Paivays;
    cout << "Tänään on: ";
    tulosta( p ); // Kutsuu Paivays::Tulosta
    cout << endl;
}
```


Nimeämätön nimiavaruus

- Käännösyksikön näkyvyysalue

```
namespace {  
    unsigned long int viiteLaskuri;  
    void lisaaViiteLaskuria() {  
        ++viiteLaskuri;  
  
        ...  
    }  
}
```