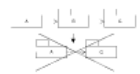
Modularization and Conceptualization of the Structure

The structure grows. How to maintain an overview?

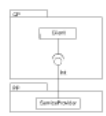
Is it a sufficient overview to just list the packages?

The dependencies between packages reflect the dependencies between their elements



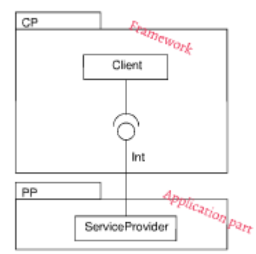
An interface should be placed together with the element that realizes (implements) it.

Or not?



```
package CP;
class Client { ... }
interface Int { ... }
\\ a client prescribes an interface

-----------------------------
package PP;
\\ the interface is implemented
\\ by a supplier from another package
class ServiceProvider implements Int { ... }
```



We have an overall class diagram

We have package class diagrams.

Do we need any other view?

fiit.sk/msoft/~vranic/dot

A comprehensive structure requires further modularization

Components and composite structure enable to reason over the structure conceptually

Let's forget about UML, classes, packages...

If we'd like to draw the overall system structure on a global level, how would we do that?

A formalization of a high-level structure in UML:

components

What's inside of components?

Component interaction can be indicated by a collaboration and it can be depicted in detail by a sequence diagram

By modeling behavior, we uncover structure.

By modeling structure, we uncover behavior.

Lecture 4:

Modularization and Conceptualization of the Structure

Valentino Vranić

Ústav informatiky, informačných systémov
a softvérového inžinierstva

STU
FIIT

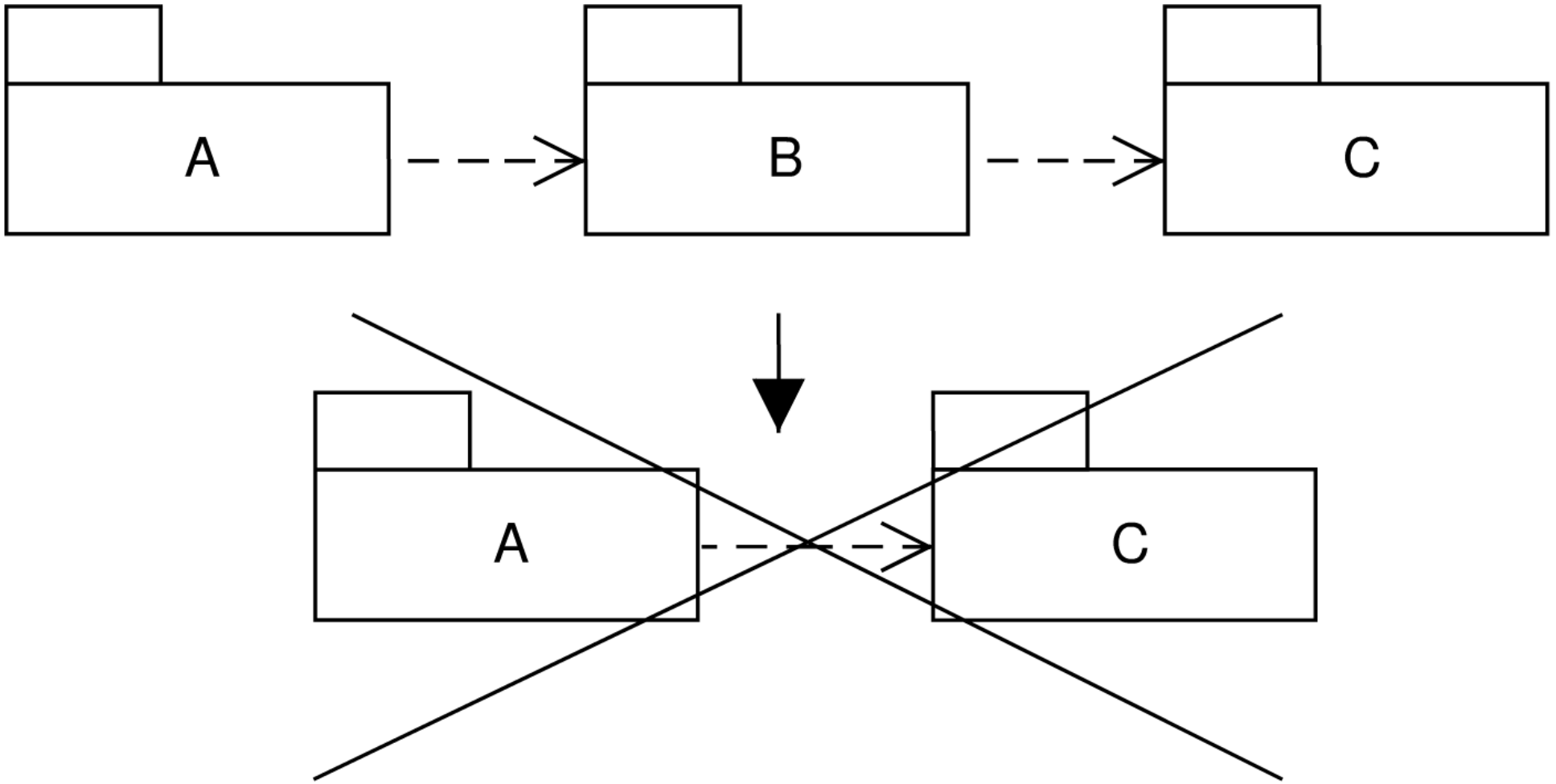
vranic@stuba.sk fiit.sk/~vranic

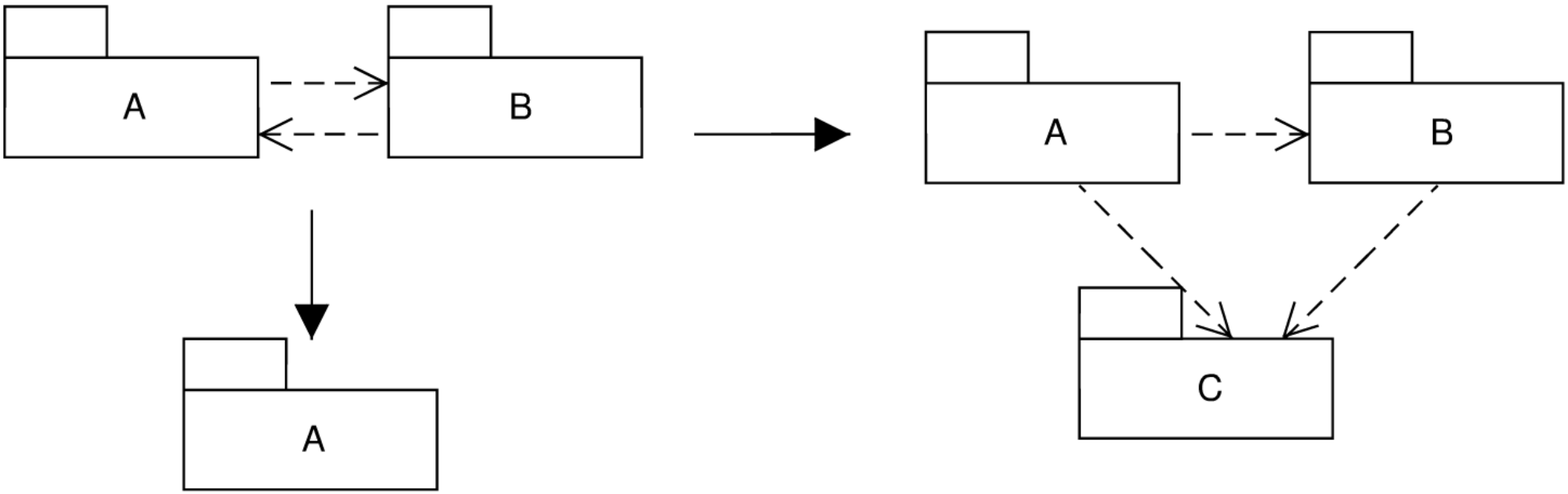
MSOFT 2019/20
15. 10. 2019

The structure grows.
How to maintain an
overview?

Is it a sufficient overview to just list the packages?

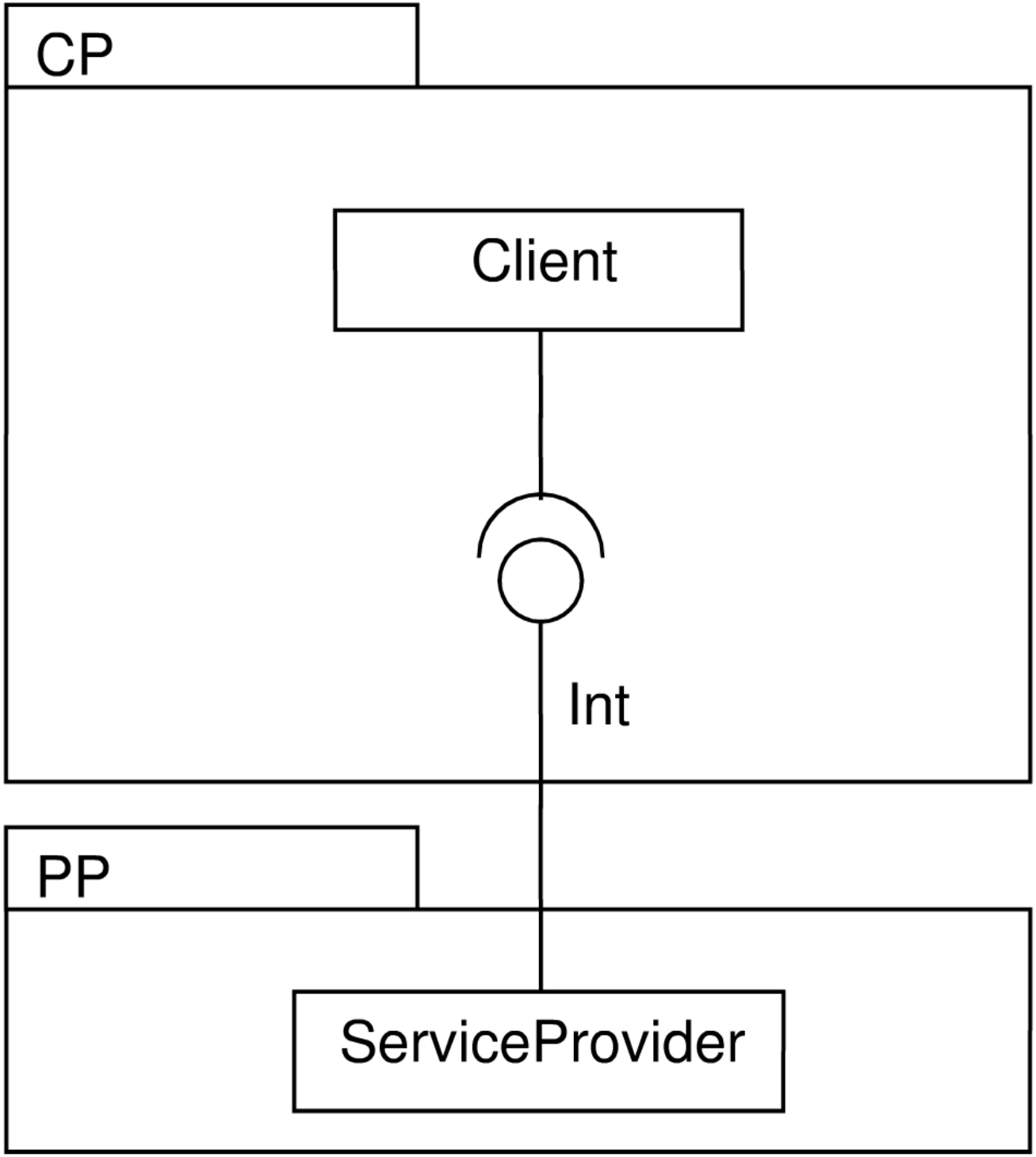
The dependencies
between packages reflect
the dependencies
between their elements

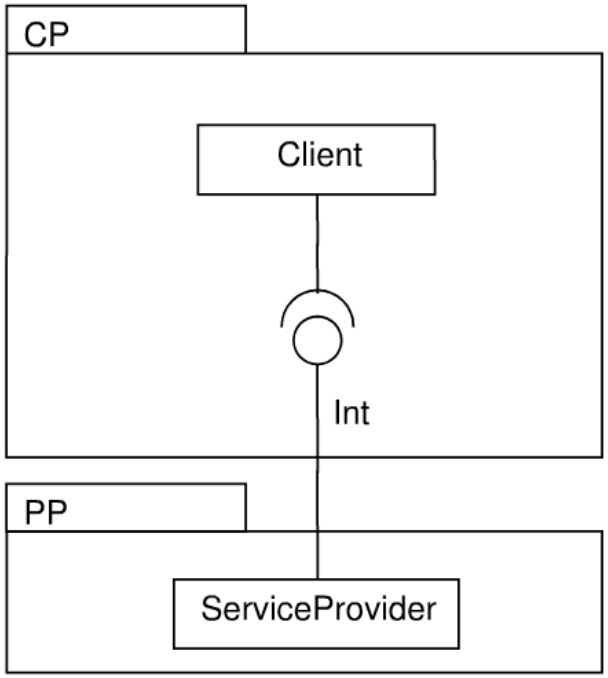

A
B
C
A
C


A
B
A
A
B
C

An interface should be placed together with the element that realizes (implements) it.

Or not?


CP
Client
Int
PP
ServiceProvider



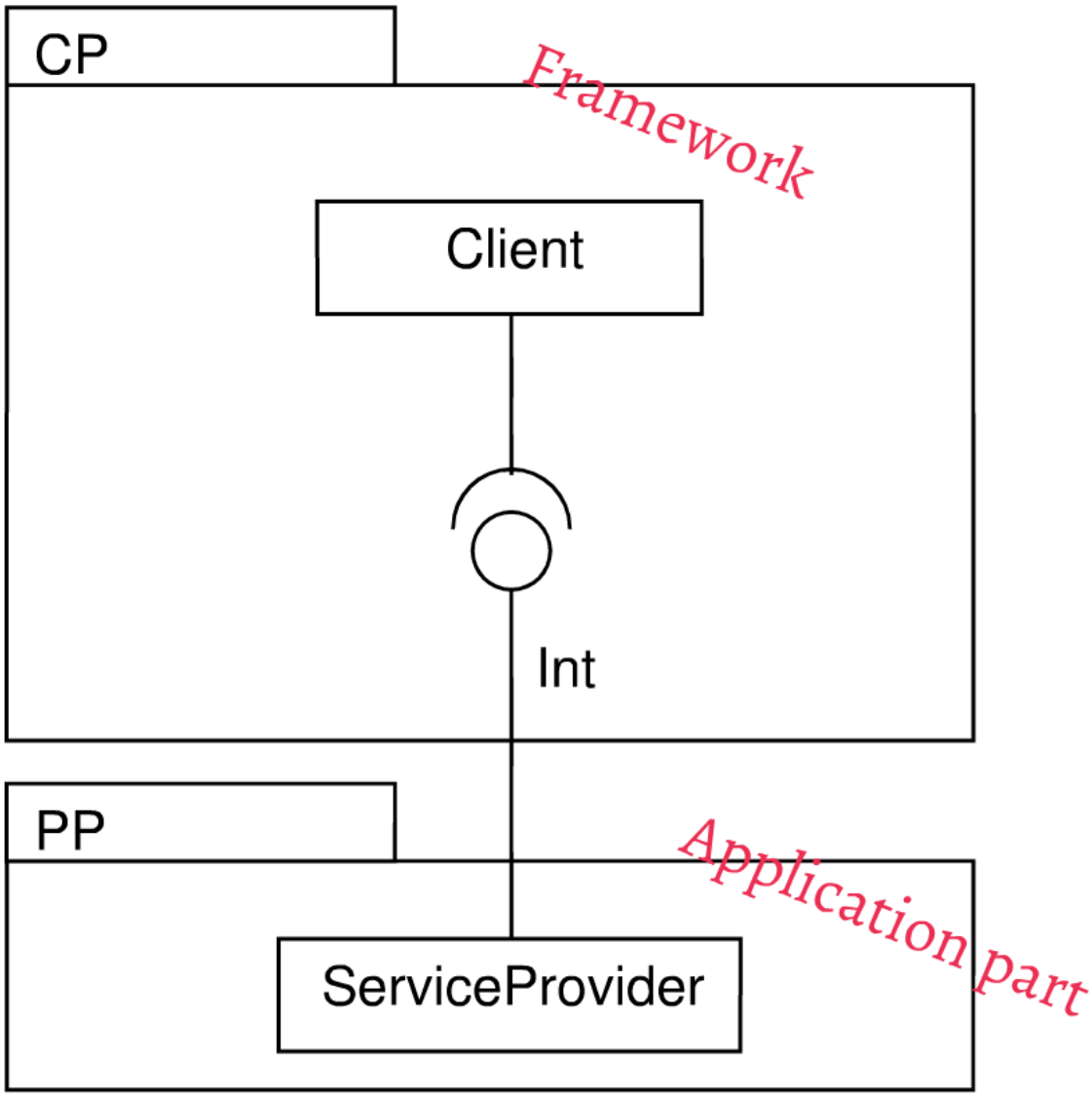

```
package CP;
class Client { ... }
interface Int { ... }
\\ a client prescribes an interface

-------------------------

package PP;
\\ the interface is implemented
\\ by a supplier from another package
class ServiceProvider implements Int { ... }
```

```
package CP;
class Client { ... }
interface Int { ... }
\\ a client prescribes an interface

------------------------
package PP;
\\ the interface is implemented
\\ by a supplier from another package
class ServiceProvider implements Int { ... }
```



We have an overall class diagram

We have package class diagrams.

Do we need any other view?

A comprehensive structure requires further modularization

Let's forget about UML, classes, packages...

If we'd like to draw the overall system structure on a global level, how would we do that?

A formalization of a high-level structure in UML:

components

What's inside of components?

Component interaction can be indicated by a collaboration and it can be depicted in detail by a sequence diagram

By modeling behavior, we uncover structure.

By modeling structure, we uncover behavior.

Components and composite structure enable to reason over the structure conceptually

fiit.sk/msoft/~vranic/dot

A comprehensive structure requires further modularization

Components and composite structure enable to reason over the structure conceptually