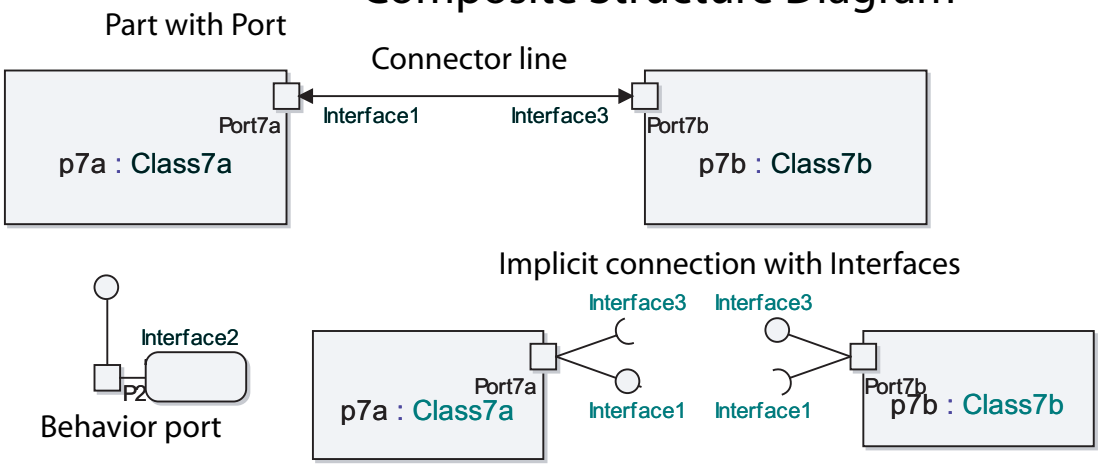
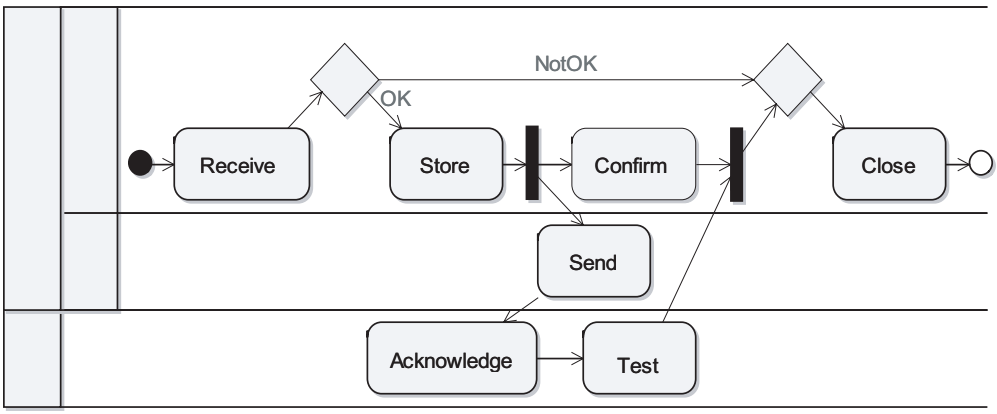
Telelogic Tau

UML Quick Reference

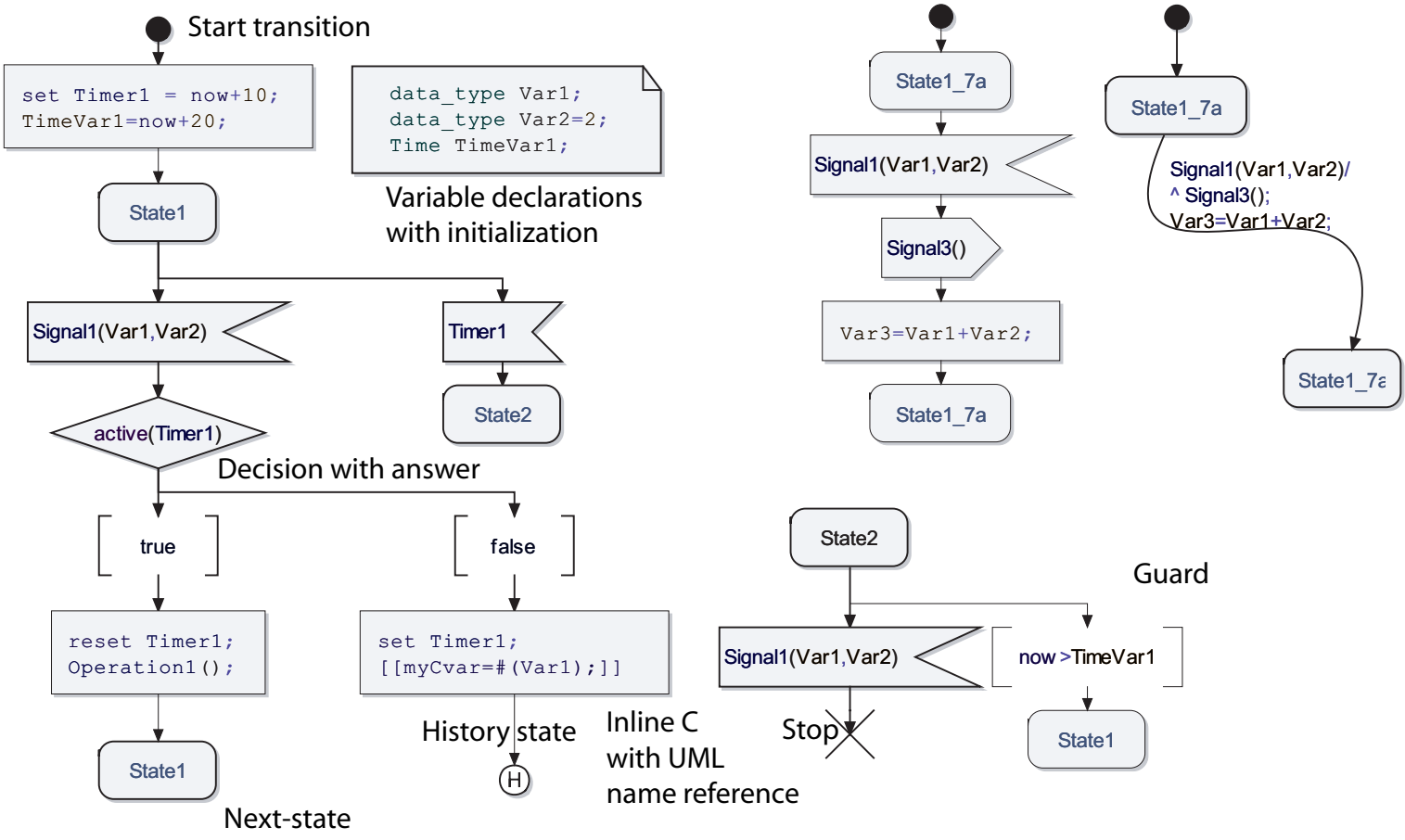
Composite Structure Diagram



Activity Diagram



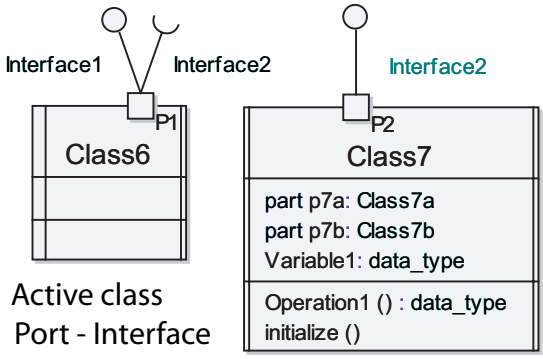
State Machine Diagram



Package Diagram

Package with import dependency

Component Diagram



Active class

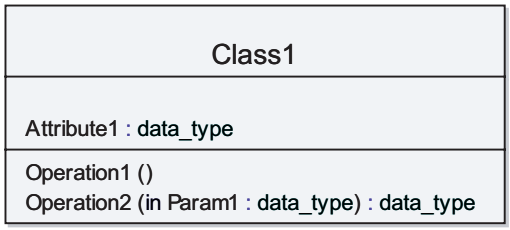
Port - Interface

Deployment Diagram

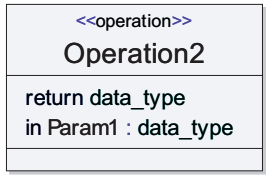
Configuration build

Class Diagram

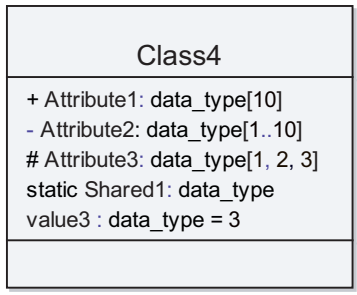
Passive class with attributes and operations



Operation with parameters



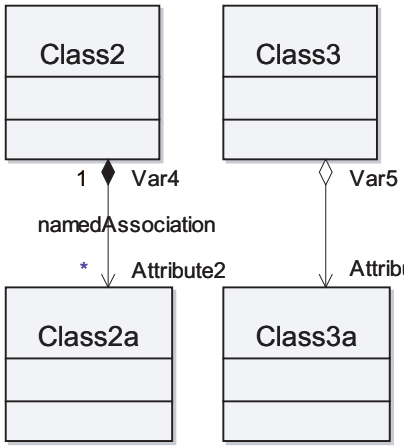
Attribute Visibility, Multiplicity



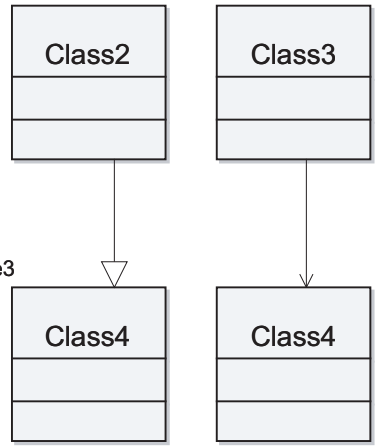

```
/* Local type declarations */
syntype data_type = Integer;
const   Integer   value1=1;
const   data_type value2=2;
class Class1 {
    data_type Attribute1;
    void      Operation1();
    data_type Operation2 ( in data_type Param1);
}
```

```
class Class4 {
    public    data_type [10]      Attribute1;
    private   data_type [1..10]   Attribute2;
    protected data_type [1, 2, 3] Attribute3;
    static    data_type Shared1;
    const     data_type value3=3;
}
```

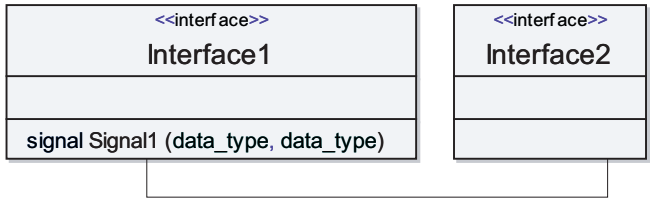
Association - Composition



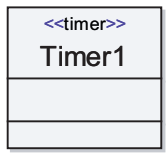
Inheritance - Dependency



Interface with signal - Associated interfaces



Timer and default duration




```
class Class2 {
    part Class2a [*] Attribute2;
}
class Class3 {
    shared Class3a Attribute3;
}
```

```
class Class2 : Class4 {

}
class Class3 dependency to Class4 {

}
```

```
timer Timer1=5;
```

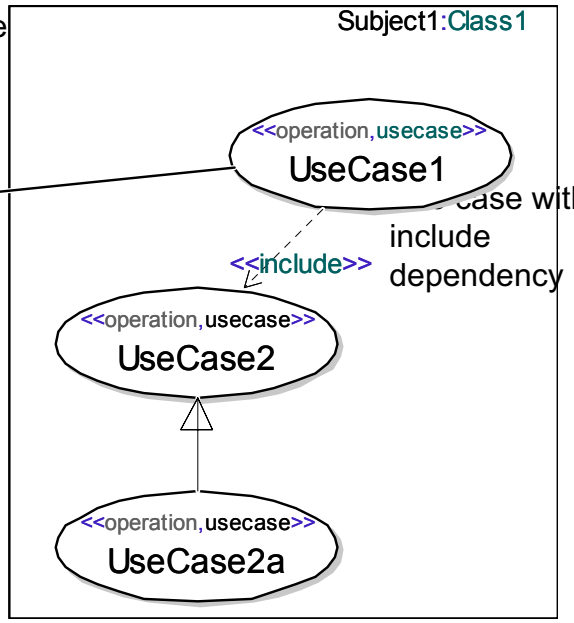
Text Diagram

data_type Operation1()

```
{
Integer i;
Integer j=10;
for(i=0;i<j;i=i+1)
    j=j-1;
return j;
}
```

Use Case Diagram

Subject frame with type



Actor - Use case
(Performance relation)

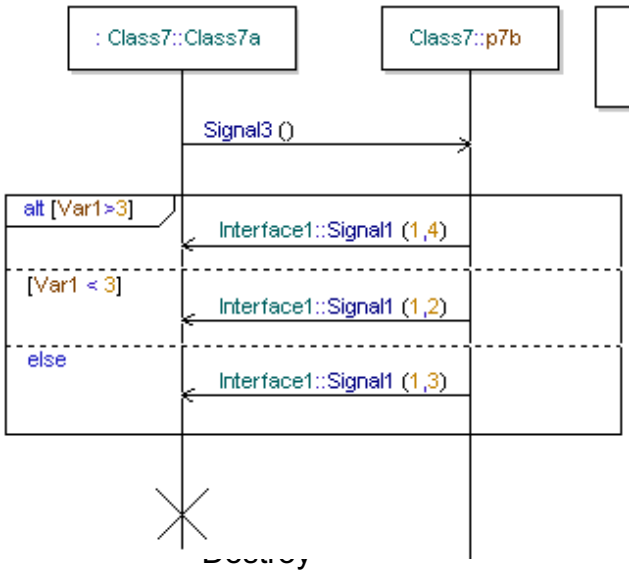
Generalization
between Use cases

case with
include
dependency

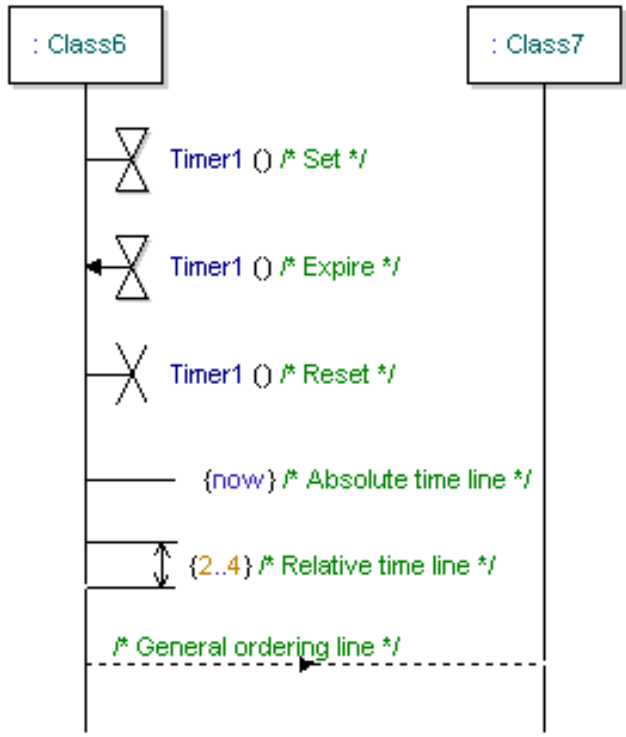
Sequence Diagram

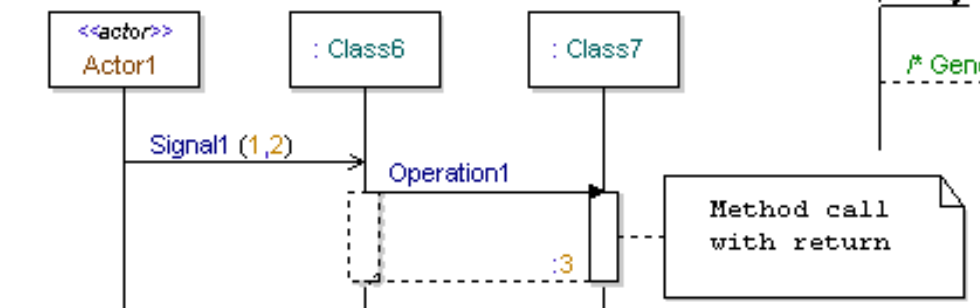
Lifeline,
typed

Lifeline,
named



Timer and time line





Referenced
Use case