

UML constraints

Multiple Generic Class

In C# syntax, we can declare classes with the same name but different generic type parameters in the same package.



Code example

```
namespace N {  
class A { .. }  
class A<T> { .. }  
class A<T, U> { .. } }
```

The user cannot create the model for these three classes in MagicDraw manually. They can only be created in C# reverse engineering.

When creating manually Class A in the model, it would show the error message when trying to create Class A.

Code Generation for Partial Class

Generate the code without the round trip feature. Separated partial classes and all its child elements will be generated into only one class code and one class file as the example below.



Case #1

Code used for reverse engineer

```
//The partial class is written into one class file.  
public partial class PartialA{  
    int a;  
    string actionA()  
    {  
    }  
}  
public partial class PartialA{  
    int b;  
    string actionB()  
    {  
    }  
}
```

Code after generation

```
public partial class PartialA{  
int a;  
int b;  
string actionA( )  
{return ;}  
string actionB( )  
{return ;}  
}
```



Case #2

Code used for reverse engineer

```
//The partial class is written into separate class file.  
//PartialA1.cs  
public partial class PartialA{  
    int a;  
    string actionA()  
    {  
    }  
}  
//PartialA2.cs  
public partial class PartialA{  
    int b;  
    string actionB()  
    {  
    }  
}
```

Code after generation

```
public partial class PartialA{  
    int a;  
    int b;  
    string actionA( )  
    { return ;}  
    string actionB( )  
    {return ;}  
}
```



Case #3

Code used for reverse engineer

```
//The partial class with inner class
public partial class PartialA{
    public class B
    {
        int b;
    }
}
public partial class PartialA{
    int a;
    string actionB( )
    {
    }
    public class C
    {
        int c;
    }
}
```

Code after generation

```
public partial class PartialA{
    int a;
    string actionB( )
    {return ;}
    public class B
    {int b;}
    public class C
    {int c;}
}
```