

Fragment kodu z klasy Foreman

```
protected virtual void OnExplodedShaft(List<Dwarf> _killedDwarfs)
{
    if (ExplodedShaft != null)
        ExplodedShaft(this, new ExplodedEventArgs() {killedDwarfs = _killedDwarfs});
}

public void CallToGravedigger(List<Dwarf> _killedDwarfs)
{
    OnExplodedShaft(_killedDwarfs);
}
```

W klasie Foreman został utworzony event który jest publikowany w momencie gdy został zburzony szyb. Event odwołuje się do klasy Gravedigger, którego odpowiedzialnością jest zabranie nieżywych krasnali. Wykorzystanie eventów w tym miejscu pozwala na lepsze przydzielenie odpowiedzialności klas, aby na przykład klasa Shaft nie musiała wiedzieć o swoim zburzeniu ani klasa Mine, tylko żeby klasa Foreman mogła wywołać metodę która publikuje event nasłuchujący w klasie Gravedigger. Dzięki temu nie musimy tworzyć w klasie Mine nowej instancji klasy Gravedigger, tylko użycie publisher'a z klasy Foreman, który poinformuje nasłuchujące obiekty.

Fragment kodu z klasy Mine

```
private void EndWorkOnTheShaft(Shaft shaft, List<Dwarf> dwarfsThatWasWorked)
{
    if (shaft.isExist == false)
    {
        City.newsPaper.Add("Mine: Shaft exploded!");
        foreman.CallToGravedigger(shaft.dwarfs);
    }
    else
    {
        City.newsPaper.Add("Mine: dwarfs digged some raw materials.");
        //Foreman let go dwarfs out from first shaft
        dwarfsThatWasWorked.AddRange(foreman.LetGoDwarfs(shaft));
    }
}
```