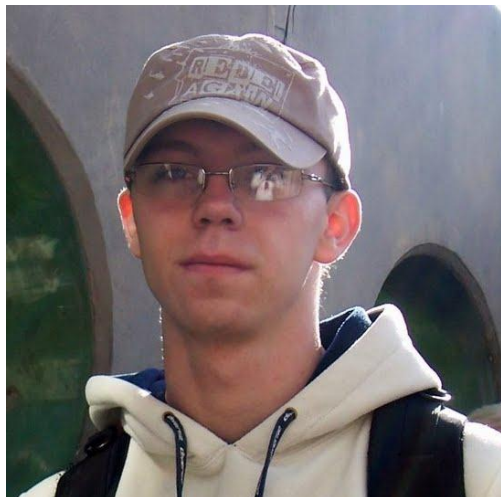


# SYSTEMY BUDOWANIA APLIKACJI

ANT

Gdańsk, 22.01.2015

# O MNIE



- Absolwent wydziału Fizyki Technicznej i Matematyki Stosowanej Politechniki Gdańskiej
- Programista w firmie Solwit
- Wykładowca na Uniwersytecie Gdańskim
  
- Blogger: [blog.dragonia.org.pl](http://blog.dragonia.org.pl)
- Twitter: [@Smoczysko](https://twitter.com/Smoczysko)
- LinkedIn: [pl.linkedin.com/in/Smoczysko](https://pl.linkedin.com/in/Smoczysko)

# PODSTAWOWE USTALENIA

✓ nie jestem alfą ani omega

# PODSTAWOWE USTALENIA

- ✓ nie jestem alfą ani omegą
- ✓ nie jestem sprzedawcą, a zwykłym developerem

# PODSTAWOWE USTALENIA

- ✓ nie jestem alfą ani omegą
- ✓ nie jestem sprzedawcą, a zwykłym developerem
- ✓ pytania mile widziane, szczególnie w trakcie

# PODSTAWOWE USTALENIA

- ✓ nie jestem alfą ani omegą
- ✓ nie jestem sprzedawcą, a zwykłym developerem
- ✓ pytania mile widziane, szczególnie w trakcie
- ✓ 2 spotkania - **2 narzędzia do budowania**

<https://github.com/Smoczysko/introduction-to-build-tools>

***maven***



# SŁÓW KILKA O LOKALNYCH PLIKACH MAVEN'A

- ✓ wszystkie pliki wykorzystywane w trakcie pracy znajdują się w katalogu M2\_HOME (zwyczajowo .m2)
  - ✓ repository - folder, w którym zapisywane są wszystkie ściągnięte biblioteki
  - ✓ settings.xml - pozwala na konfigurację m.in. repozytoriów
- ✓ biblioteki wykorzystywane w projekcie nie są do niego kopiowane

***maven***



# APACHE ANT

- ✓ pierwsze narzędzie do budowania aplikacji Javowych

# APACHE ANT

- ✓ pierwsze narzędzie do budowania aplikacji Javowych
- ✓ cała konfiguracja znajduje się w pliku XML (build.xml)

# APACHE ANT

- ✓ pierwsze narzędzie do budowania aplikacji Javowych
- ✓ cała konfiguracja znajduje się w pliku XML (build.xml)
- ✓ wszystkie czynności muszą być jawnie podane ("zakodowane")

# APACHE ANT

- ✓ pierwsze narzędzie do budowania aplikacji Javowych
- ✓ cała konfiguracja znajduje się w pliku XML (build.xml)
- ✓ wszystkie czynności muszą być jawnie podane ("zakodowane")
- ✓ konfiguracja bardziej przypomina skryptowanie

# APACHE ANT

- ✓ pierwsze narzędzie do budowania aplikacji Javowych
- ✓ cała konfiguracja znajduje się w pliku XML (build.xml)
- ✓ wszystkie czynności muszą być jawnie podane ("zakodowane")
- ✓ konfiguracja bardziej przypomina skryptowanie
- ✓ nie dostarcza mechanizmu zarządzania zależnościami

***maven***





# APACHE IVY

- ✓ dodaje możliwość zarządzania zależnościami

# APACHE IVY

- ✓ dodaje możliwość zarządzania zależnościami
- ✓ nowsze wersje kompatybilne z repozytoriami Maven 2

# APACHE IVY

- ✓ dodaje możliwość zarządzania zależnościami
- ✓ nowsze wersje kompatybilne z repozytoriami Maven 2
- ✓ konwencja zależności analogiczna do Maven'owej:
  - ✓ org -> groupId
  - ✓ name -> artifactId
  - ✓ rev -> version

# APACHE IVY

- ✓ dodaje możliwość zarządzania zależnościami
- ✓ nowsze wersje kompatybilne z repozytoriami Maven 2
- ✓ konwencja zależności analogiczna do Maven'owej:
  - ✓ org -> groupId
  - ✓ name -> artifactId
  - ✓ rev -> version
- ✓ rozwiązywanie zależności projektu odbywa się w dwóch fazach:
  - ✓ weryfikacja zależności (i ich zależności rekurencyjnie)
  - ✓ faktycznie ściąganie zależności

# APACHE IVY

- ✓ dodaje możliwość zarządzania zależnościami
- ✓ nowsze wersje kompatybilne z repozytoriami Maven 2
- ✓ konwencja zależności analogiczna do Maven'owej:
  - ✓ org -> groupId
  - ✓ name -> artifactId
  - ✓ rev -> version
- ✓ rozwiązywanie zależności projektu odbywa się w dwóch fazach:
  - ✓ weryfikacja zależności (i ich zależności rekurencyjnie)
  - ✓ faktycznie ściąganie zależności
- ✓ wykorzystywane biblioteki są bezpośrednio kopiowane do projektu

# APACHE IVY

- ✓ dodaje możliwość zarządzania zależnościami
- ✓ nowsze wersje kompatybilne z repozytoriami Maven 2
- ✓ konwencja zależności analogiczna do Maven'owej:
  - ✓ org -> groupId
  - ✓ name -> artifactId
  - ✓ rev -> version
- ✓ rozwiązywanie zależności projektu odbywa się w dwóch fazach:
  - ✓ weryfikacja zależności (i ich zależności rekurencyjnie)
  - ✓ faktycznie ściąganie zależności
- ✓ wykorzystywane biblioteki są bezpośrednio kopiowane do projektu
- ✓ wymaga to ręcznego czyszczenia

# APACHE IVY - KONFIGURACJA

- ✓ w pliku build.xml dodajemy przestrzeń nazw XML:  
xmlns:ivy="antlib:org.apache.ivy.ant"
- ✓ plik ivy.xml:

```
<ivy-module version="2.0">  
  <info organisation="pl.org.dragonia" module="buildtools"/>  
  
  <dependencies>  
    <dependency org="javax" name="javaee-web-api" rev="7.0" />  
  </dependencies>  
</ivy-module>
```

# APACHE IVY – IVYSETTINGS.XML

```
<ivysettings>  
  <settings defaultResolver="mainchain" />  
  <resolvers>  
    <chain name="mainchain">  
      <ibiblio name="main" m2compatible="true" />  
    </chain>  
  </resolvers>  
</ivysettings>
```



# APACHE IVY - WYKORZYSTANIE

```
<target name="clearcache">  
  <ivy:cleancache />  
</target>
```

```
<target name="ivySettings">  
  <ivy:settings file="ivysettings.xml" />  
</target>
```

```
<target name="resolve" depends="ivySettings">  
  <ivy:retrieve />  
</target>
```

PYTANIA?

Q&A