

# Codestijl

ir. Jan-Hendrik Kuperus



# Wie ben ik?

- **ir. Jan-Hendrik Kuperus**
- **Software Engineer**
- **Java Coach**
- **Java Competence Network**
- **Main interests**
  - ◆ **Anything Java**
  - ◆ **Theoretical computing**
  - ◆ **Games!**



# Engineering World 2009



# Engineering World 2009



# Engineering World 2009

- **Back to the future**



# Engineering World 2009

- **Back to the future**
- **Codeestijl, irritatie of noodzaak?**



- **Back to the future**
- **Codeestijl, irritatie of noodzaak?**
- **Tijdreis!**



# Engineering World 2037





# Engineering World 2037

- **Welkom**



# Engineering World 2037

- **Welkom**
- **Columbus Missie**



# Engineering World 2037

- **Welkom**
- **Columbus Missie**
  - ◆ **Start Kolonisatie Mars**



# Engineering World 2037

- **Welkom**
- **Columbus Missie**
  - ◆ **Start Kolonisatie Mars**
- **Codestijl sleutel tot succes!**



# Codestijl is de sleutel



# Codestijl is de sleutel

- **Hoge kwaliteit code**



# Codestijl is de sleutel

- **Hoge kwaliteit code**
- **Gestandaardiseerde processen**



# Codestijl is de sleutel

- **Hoge kwaliteit code**
- **Gestandaardiseerde processen**
- **Verifieerbaarheid**





# Codestijl is de sleutel

- **Hoge kwaliteit code**
- **Gestandaardiseerde processen**
- **Verifieerbaarheid**
- **Security - PaSS**



# Het begin...



# Het begin...

- **JCN Styleguide**



# Het begin...

- **JCN Styleguide**
  - ◆ **Versie 3.0**



# Het begin...

- **JCN Styleguide**
  - ◆ **Versie 3.0**
- **Ontwikkelstraat**



# Het begin...

- **JCN Styleguide**
  - ◆ **Versie 3.0**
- **Ontwikkelstraat**
  - ◆ **Van intake...**
  - ◆ **...**
  - ◆ **... tot toproduct**



# Lage kwaliteit



# Lage kwaliteit

```
public void create() throws Exception
{
    String propName;
    propName = getGatewayId() + ".";
    setGateway(new org.smslib.modem.SerialModemGateway(getGatewayId
    if (getProperties().getProperty(propName + "protocol").equalsIgn
    else if (getProperties().getProperty(propName + "protocol").equ
    else throw new Exception("Incorrect parameter: " + propName + "
    getGateway()).setSimPin(getProperties().getProperty(propName +
    if (getProperties().getProperty(propName + "inbound").equalsIgn
    getGateway().setInbound(true);
}
```





# Hoge kwaliteit



# Hoge kwaliteit

- **JCN Styleguide**



# Hoge kwaliteit

- **JCN Styleguide**
  - ◆ **Opmaak**



# Hoge kwaliteit

- **JCN Styleguide**

- **✦ Opmaak**

```
public Gateway create(String gatewayId)
{
    return (this.gateway == null) ? new Gateway() : this.gateway;
}
```



# Hoge kwaliteit

- **JCN Styleguide**

- ◆ **Opmaak**

```
public Gateway create(String gatewayId)
{
    return (this.gateway == null) ? new Gateway() : this.gateway;
}
```

- ◆ **Of**



# Hoge kwaliteit

- **JCN Styleguide**

- ◆ **Opmaak**

```
public Gateway create(String gatewayId)
{
    return (this.gateway == null) ? new Gateway() : this.gateway;
}
```

- ◆ **Of**

```
public Gateway create (String gatewayId) {
    return ( this.gateway==null )
        ? new Gateway( )
        : this.gateway;
}
```



# Hoge kwaliteit



# Hoge kwaliteit

- **JCN Styleguide**





# Hoge kwaliteit

- **JCN Styleguide**
  - ◆ **Naamgeving**



# Hoge kwaliteit

- **JCN Styleguide**

  - ◆ **Naamgeving**

- **Slechte naamgeving:**



# Hoge kwaliteit

- **JCN Styleguide**

- ◆ **Naamgeving**

- **Slechte naamgeving:**

- ◆ **myString, abc, temp, test, pietje**



# Hoge kwaliteit

- **JCN Styleguide**
  - ✦ **Naamgeving**
  
- **Slechte naamgeving:**
  - ✦ **myString, abc, temp, test, pietje**
  
- **Goede naamgeving:**



# Hoge kwaliteit

- **JCN Styleguide**
  - ✦ **Naamgeving**
  
- **Slechte naamgeving:**
  - ✦ **myString, abc, temp, test, pietje**
  
- **Goede naamgeving:**
  - ✦ **id, phoneNr, subResult, searchName**



# Hoge kwaliteit



# Hoge kwaliteit

- **JCN Styleguide**



# Hoge kwaliteit

- **JCN Styleguide**
  - ◆ **Design**





# Hoge kwaliteit

- **JCN Styleguide**

  - ◆ **Design**

- **Constructors of Factories?**



# Hoge kwaliteit

- **JCN Styleguide**
  - ◆ **Design**
- **Constructors of Factories?**
- **Mutable of Immutable objects?**



# Hoge kwaliteit

- **JCN Styleguide**
  - ◆ **Design**
- **Constructors of Factories?**
- **Mutable of Immutable objects?**
- **Best practices**



# Hoge kwaliteit



# Hoge kwaliteit

- **JCN Styleguide**



# Hoge kwaliteit

- **JCN Styleguide**
  - ◆ **Documentatie**



# Hoge kwaliteit

- **JCN Styleguide**

- ◆ **Documentatie**

```
/**  
 * Returns the name of this object.  
 */  
public String getName()
```



# Hoge kwaliteit

- **JCN Styleguide**

- **Documentatie**

```
/**  
 * Returns the name of this object.  
 */  
public String getName()
```

```
/**  
 * Calculates the age of this person, based on its birthdate.  
 * The age is returned as the number days since birth.  
 */  
public int getAgeInDays()
```





# Hoge kwaliteit



# Hoge kwaliteit

- **JCN Styleguide**



# Hoge kwaliteit

- **JCN Styleguide**
  - ◆ **Opmaak**



# Hoge kwaliteit

- **JCN Styleguide**

- ◆ **Opmaak**

- ◆ **Naamgeving**



# Hoge kwaliteit

- **JCN Styleguide**

- ◆ **Opmaak**

- ◆ **Naamgeving**

- ◆ **Design**



# Hoge kwaliteit

- **JCN Styleguide**

- ◆ **Opmaak**

- ◆ **Naamgeving**

- ◆ **Design**

- ◆ **Documentatie**



# Hoge kwaliteit

- **JCN Styleguide**

- ◆ **Opmaak**

- ◆ **Naamgeving**

- ◆ **Design**

- ◆ **Documentatie**

- ◆ **(Secure Programming Guidelines)**



# JCN Styleguide





# JCN Styleguide

- **Voorbeeld:**



# JCN Styleguide

- **Voorbeeld:**
- **Standard JCN-1.1-F**  
**“Variable names never begin or end with an underscore”**



# JCN Styleguide

- **Voorbeeld:**
- **Standard JCN-1.1-F**  
**“Variable names never begin or end with an underscore”**
- **Uitzonderingen gedocumenteerd**



# Lage kwaliteit



# Lage kwaliteit

```
public void create() throws Exception
{
    String propName;
    propName = getGatewayId() + ".";
    setGateway(new org.smslib.modem.SerialModemGateway(getGatewayId
    if (getProperties().getProperty(propName + "protocol").equalsIg
    else if (getProperties().getProperty(propName + "protocol").equ
    else throw new Exception("Incorrect parameter: " + propName + "
    getGateway()).setSimPin(getProperties().getProperty(propName +
    if (getProperties().getProperty(propName + "inbound").equalsIgn
    getGateway().setInbound(true);
}
```



# Hoge kwaliteit



# Hoge kwaliteit

```
/**
 * Creates a new modem Gateway, initializes it with properties
 * from the classpath. See class description for more information
 * on the properties.
 *
 * @param gatewayId the name of the Gateway from the config
 * @return the newly created Gateway
 */
public Gateway create(String gatewayId)
{
    // We need only properties which are prefixed with gatewayId
    String propertyPrefix = gatewayId + ".";

    // Create the new Gateway
    Gateway result = new SerialModemGateway(
        getProperty(propertyPrefix + PROP_MODEM_PORT,
        getProperty(propertyPrefix + PROP_MODEM_PROTOCOL));

    return result;
}
```



# Gestandaardiseerde processen





# Gestandaardiseerde processen

- **Versiebeheer**



# Gestandaardiseerde processen

- **Versiebeheer**
- **Continuous build**



# Gestandaardiseerde processen

- **Versiebeheer**
- **Continuous build**
- **Automated testing**



# Gestandaardiseerde processen

- **Versiebeheer**
- **Continuous build**
- **Automated testing**
- **Code review**



# Versiebebeheer



- **Single change commit**



# Versieebeheer

- **Single change commit**
- **Revisie documentatie**



# Versiebeheer

- **Single change commit**
- **Revisie documentatie**

◆ **Fout:**





# Versiebeheer

- **Single change commit**
- **Revisie documentatie**

◆ **Fout:**

Changed some files



# Versiebeheer

- **Single change commit**
- **Revisie documentatie**

- ◆ **Fout:**

Changed some files

- ◆ **Goed:**



- **Single change commit**
- **Revisie documentatie**

◆ **Fout:**

Changed some files

◆ **Goed:**

Bugfix for issue #1337  
Turned off lazy loading



# Continuous build



# Continuous build

- **Snel opsporen compiler fouten**



# Continuous build

- **Snel opsporen compiler fouten**
- **Opstap naar automatisch testen**



# Continuous build

- **Snel opsporen compiler fouten**
- **Opstap naar automatisch testen**
- **Nightly builds**



# Continuous build

- **Snel opsporen compiler fouten**
- **Opstap naar automatisch testen**
- **Nightly builds**





# Automated testing



# Automated testing

- **Vroeg opsporen fouten**



# Automated testing

- **Vroeg opsporen fouten**
  - ✦ **Unit testing**



# Automated testing

- **Vroeg opsporen fouten**
  - ✦ **Unit testing**
  - ✦ **Integration testing**



# Automated testing

- **Vroeg opsporen fouten**
  - ◆ **Unit testing**
  - ◆ **Integration testing**
- **Regressietesten**



# Automated testing

- **Vroeg opsporen fouten**
  - ◆ **Unit testing**
  - ◆ **Integration testing**
- **Regressietesten**
- **Zichtbare voortgang**



# Code review



# Code review

- **Collega's reviewen code**





# Code review

- **Collega's reviewen code**
- **Reviews kosten wel tijd**



# Code review

- **Collega's reviewen code**
- **Reviews kosten wel tijd**
- **Biedt kansen om van elkaar te leren**



# Security



- **Security Aware Engineers**



- **Security Aware Engineers**
  - ◆ **OWASP**



- **Security Aware Engineers**
  - ◆ **OWASP**
  - ◆ **PaSS**



- **Security Aware Engineers**
  - ◆ **OWASP**
  - ◆ **PaSS**
  
- **Security analyse**



# Security

- **Security Aware Engineers**
  - ◆ **OWASP**
  - ◆ **PaSS**
  
- **Security analyse**
  - ◆ **Fortify**





# Security

- **Security Aware Engineers**
  - ◆ **OWASP**
  - ◆ **PaSS**
  
- **Security analyse**
  - ◆ **Fortify**
  - ◆ **Assessments**



# Security

- **Security Aware Engineers**
  - ◆ **OWASP**
  - ◆ **PaSS**
- **Security analyse**
  - ◆ **Fortify**
  - ◆ **Assessments**
- **Security Officer**



# Verifieerbaarheid



# Verifieerbaarheid

- **Formele documentatie**



# Verifieerbaarheid

- **Formele documentatie**
  - ✦ **Pre- en postcondities**



# Verifieerbaarheid

- **Formele documentatie**
  - ✦ **Pre- en postcondities**
- **Bytecode analysis**



# Verifieerbaarheid

- **Formele documentatie**
  - ✦ **Pre- en postcondities**
- **Bytecode analysis**
- **Model checking**



# Formele documentatie





# Formele documentatie

- **Pre-condities**



# Formele documentatie

- **Pre-conditions**
  - ◆ **JSR-303 Bean Validation**



# Formele documentatie

- **Pre-conditions**
  - ✦ **JSR-303 Bean Validation**
- **Post-conditions**



# Formele documentatie

- **Pre-conditions**
  - ✦ **JSR-303 Bean Validation**
- **Post-conditions**
- **Lus invarianten**



# Formele documentatie

```
/**
 * Calculates the earned interest on this account and adds it
 * to the balance.
 *
 * @param the interest rate to apply to the account
 */
@Require(interestRate >= 0 && interestRate < 100)
@Ensure(NEW.getBalance() == OLD.getBalance() +
        (OLD.getBalance() * (interestRate / 100.0)))
public void calculateInterest(int interestRate);
```



# Bytecode analyse



# Bytecode analyse

- **FindBugs**



# Bytecode analyse

- **FindBugs**

- ✦ **Spoort honderden soorten bugs op**





# Bytecode analyse

- **FindBugs**

- ✦ **Spoort honderden soorten bugs op**
- ✦ **Detecteert 'bad practices'**



# Bytecode analyse

- **FindBugs**

- ✦ Spoort honderden soorten bugs op
- ✦ Detecteert 'bad practices'
  
- ✦ 2000 bugs in Sun JDK gevonden



# Bytecode analyse

- **FindBugs**

- ✦ Spoort honderden soorten bugs op
- ✦ Detecteert 'bad practices'
  
- ✦ 2000 bugs in Sun JDK gevonden



# Model checking



# Model checking

- **Model van code**



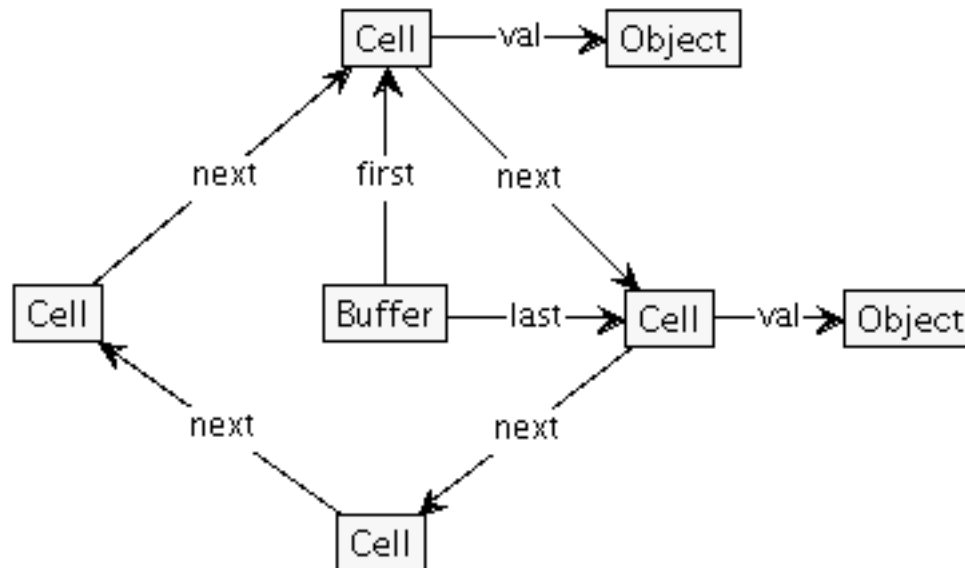
# Model checking

- **Model van code**
- **Circular buffer:**



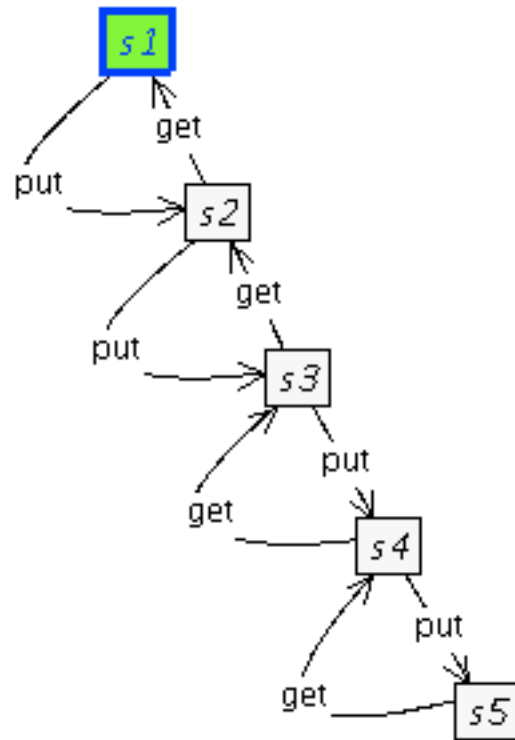
# Model checking

- **Model van code**
- **Circular buffer:**



# Model checking (2)

- **Doorrekenen alle toestanden**





# Model checking (3)



# Model checking (3)

- **GROOVE**



# Model checking (3)

- **GROOVE**
  - ✦ **Universiteit Twente**



# Model checking (3)

- **GROOVE**
  - ✦ **Universiteit Twente**
  - ✦ **Java analyse**



# Model checking (3)

- **GROOVE**
  - ◆ **Universiteit Twente**
  - ◆ **Java analyse**
- **Spin**



# Model checking (3)

- **GROOVE**
  - ✦ **Universiteit Twente**
  - ✦ **Java analyse**
- **Spin**
  - ✦ **LTL model analyse**



# Codestijl



- **Afspraken in ruil voor kwaliteit**





- **Afspraken in ruil voor kwaliteit**
- **Tools ter ondersteuning**



- **Afspraken in ruil voor kwaliteit**
- **Tools ter ondersteuning**
- **Teamverloop is geen issue**



- **Afspraken in ruil voor kwaliteit**
- **Tools ter ondersteuning**
- **Teamverloop is geen issue**
- **Code wordt kunst**





**Staat voor resultaat**

 Java Competence Network  
De Javapartij van Nederland