




Revisiones de código en el contexto del aseguramiento de calidad. Un caso práctico

*J.J. Gutiérrez, Universidad de Sevilla
M. Pérez, O. González, J. Ponce, J.M. Correa, A.I. Merino, Fundación Fideta
M.J. Escalona, Universidad de Sevilla*



Introducción

- La correcta aplicación de los métodos y procesos metodológicos en el desarrollo de un sistema software, son la mejor garantía de calidad para el producto final.
- A lo largo del ciclo de vida, los equipos que miden la calidad toman medidas y establecen puntos de control que van midiéndolo esa calidad poco a poco.
- El grupo de trabajo que ha desarrollado este artículo, focaliza su trabajo de investigación en la definición de protocolos de control de seguridad en el desarrollo, principalmente, de sistemas Web.

24-25 septiembre 2008 X Jornadas de Innovación y Calidad del Software 2008 XJICS - 2

X JICS
JORNADAS DE INNOVACIÓN Y CALIDAD DEL SOFTWARE
Conferencia Iberoamericana de Calidad del Software

Introducción: Metodología

```
graph TD; Start(( )) --> Requisitos[requisitos]; Requisitos --> Analisis[análisis]; Analisis --> Diseño[diseño]; Diseño --> Construcción[construcción]; Construcción --> Pruebas[pruebas]; Pruebas --> End(( ));
```

24-25 septiembre 2008 X Jornadas de Innovación y Calidad del Software 2008 XJICS - 3

X JICS
JORNADAS DE INNOVACIÓN Y CALIDAD DEL SOFTWARE
Conferencia Iberoamericana de Calidad del Software

La tecnología PMD

- El objetivo principal de una herramienta como PMD, es ofrecer un entorno automático que permite la obtención de métricas objetivas para medir la calidad de un código fuente.
- Sin embargo, en los desarrollos reales aparecen dos factores: tiempo y calidad, que en muchos casos resultan contradictorios.
- Para medir la calidad de un código hay dos opciones:
 - Utilizar listas de comprobación que puedan dar una valoración objetiva del mismo.
 - Utilizar entornos de herramientas que automaticen este proceso.

24-25 septiembre 2008 X Jornadas de Innovación y Calidad del Software 2008 XJICS - 4

X JICS
JORNADAS DE INNOVACIÓN Y CALIDAD DEL SOFTWARE
Conferencia Iberoamericana de Calidad del Software

La tecnología PMD

- **PMD se integra en ese último grupo de herramientas.**
- **PMD es integrable en entornos como NETBeans o Eclipse y, además, es compatible con otras herramientas similares.**
- **En su capacidad de análisis de código es capaz de detectar aspectos como:**
 - Posibles errores - try/catch/finally/switch incompletos.
 - Código que no se usa: variables locales, parámetros y métodos privados.
 - Código poco optimo - mejor uso de String/StringBuffer
 - Expresiones redundantes - if innecesarios, for loops que podrían ser while loops, etc.
 - Código duplicado - código copiado y pegado significa bugs copiados y pegados

24-25 septiembre 2008 X Jornadas de Innovación y Calidad del Software 2008 XJICS - 5

X JICS
JORNADAS DE INNOVACIÓN Y CALIDAD DEL SOFTWARE
Conferencia Iberoamericana de Calidad del Software

Package Explorer

- Commons-Lang
 - Java
 - New
 - Go Into
 - Open in New Window
 - Open Type Hierarchy
 - Cut
 - Copy
 - Paste
 - Delete
 - Source
 - Refactor
 - Import...
 - Export...
 - Refresh
 - Run
 - Team
 - Compare With
 - Restore from Local History...
 - PMD**
 - Clear violations reviews
 - Clear PMD Violations
 - Check Code With PMD
 - Properties

24-25 septiembre 2008

X JICS
JORNADAS DE INNOVACIÓN Y CALIDAD DEL SOFTWARE
Conferencia Iberoamericana de Calidad del Software

Adaptaciones de PMD

- **PMD es una herramienta que trae implementada un conjunto de reglas por defecto priorizadas de 5 (leve) a 1 (muy crítica).**
- **Sin embargo, se pueden configurar añadiendo nuevas, modificando las existentes o cambiando la prioridad.**
- **El proceso que hemos seguido en nuestra experiencia para implantar PMD se ha basado en una serie de pasos.**

24-25 septiembre 2008 X Jornadas de Innovación y Calidad del Software 2008 XJICS - 7

X JICS
JORNADAS DE INNOVACIÓN Y CALIDAD DEL SOFTWARE
Conferencia Iberoamericana de Calidad del Software

Adaptaciones de PMD

- Elegir la plataforma
- Estudiar el rule set de PMD de manera empírica
- Definir las reglas en base al estudio anterior
Establecer las prioridades adecuadas
- Formar a los equipos e implantarlo

y se puede integrar tales IDEs:
Eclipse, NetBeans, IntelliJ IDEA, Textpad, NetBeans, Creator y Emacs

24-25 septiembre 2008 X Jornadas de Innovación y Calidad del Software 2008 XJICS - 8

X JICS
JORNADAS DE INNOVACIÓN Y CALIDAD DEL SOFTWARE
Conferencia Internacional de Calidad del Software

Adaptaciones de PMD

- Las reglas de PMD pueden definirse mediante JAVA o mediante Xpath.
- Para añadir nuestras propias reglas, hemos usado la herramienta de diseño de reglas que viene incluida con PMD.
- El proceso para crear una regla se resume en escribir un trozo de código que incumpla dicha regla, generar con la herramienta el árbol de sintaxis abstracta y se escribe la expresión XPath que encuentre sólo los nodos del árbol que incumplen la regla.
- La misma herramienta de diseño de reglas de PMD genera el código XML de la regla para poder incluirla dentro del Ruleset.

24-25 septiembre 2008 X Jornadas de Innovación y Calidad del Software 2008 XJICS - 9

The screenshot shows the PMD Rule Designer interface with four numbered callouts:

- 1**: The code editor showing a Java class `class A { List<Object> l; List<X> l; Set<Object> s; }`.
- 2**: The XPath query field containing the expression `parent/ReferenceType/ClassOrInterfaceType[@Image='Object']`.
- 3**: The Abstract Syntax Tree (AST) view showing the hierarchical structure of the code, including nodes like `TypeDeclaration`, `ClassOrInterfaceDeclaration`, `FieldDeclaration`, and `ReferenceType`.
- 4**: The AST node selection view showing the selected nodes: `ASTClassOrInterfaceType at line 3` and `ASTClassOrInterfaceType at line 7`.

X JICS
JORNADAS DE INNOVACIÓN Y CALIDAD DEL SOFTWARE
Conferencia Iberoamericana de Calidad del Software

Adaptaciones de PMD

- Las reglas de PMD pueden definirse mediante JAVA o mediante Xpath.
- Para añadir nuestras propias reglas, hemos usado la herramienta de diseño de reglas que viene incluida con PMD.
- El proceso para crear una regla se resume en escribir un trozo de código que incumple dicha regla, generar con la herramienta el árbol de sintaxis abstracta y se escribe la expresión XPath que encuentre sólo los nodos del árbol que incumplen la regla.
- La misma herramienta de diseño de reglas de PMD genera el código XML de la regla para poder incluirla dentro del Ruleset.


24-25 septiembre 2008 X Jornadas de Innovación y Calidad del Software 2008 XJICS - 11

X JICS
JORNADAS DE INNOVACIÓN Y CALIDAD DEL SOFTWARE
Conferencia Iberoamericana de Calidad del Software

Adaptaciones de PMD

- Además de redefinir y crear nuevas reglas, se han adecuado otros aspectos:
 - Parametrización de la herramienta CPD (Copy&Paste Detector). Ésta busca código duplicado. La parametrización de la herramienta se ha modificado a 50 caracteres (en su versión por defecto está en 25 caracteres), ya que es más que suficiente para detectar bloques de 5 o 6 líneas si éstas tienen muchos caracteres o unas 10 líneas si tienen pocos.
 - Adecuación de la salida de resultados de PMD: En un principio usamos el plugin de Eclipse de exportar el reporte de incidencias a CSV y posteriormente lo importamos desde una base de datos Access.
 - Posteriormente, decidimos hacer uso del paquete Renders, creando una clase que genera un documento RTF con la librería iText consistente en una tabla con todas las incidencias encontradas por PMD

24-25 septiembre 2008 X Jornadas de Innovación y Calidad del Software 2008 XJICS - 12



Adaptaciones de PMD

- **Implantar una herramienta como PMD no es sencilla y requiere el apoyo de los niveles directivos.**
- **En nuestra experiencia empírica, al inicio los equipos son muy reacios a la hora de sentirse evaluados por una herramienta como PMD.**
- **El nivel de criticidad de las reglas, la importación de la certificación del código, etc. no se ve de la misma manera desde una Oficina de Calidad que desde un equipo de desarrollo.**
- **En la implantación ha sido fundamental la ejecución de cursos de formación a los equipos.**

24-25 septiembre 2008
X Jornadas de Innovación y Calidad del Software 2008
XJICS - 13

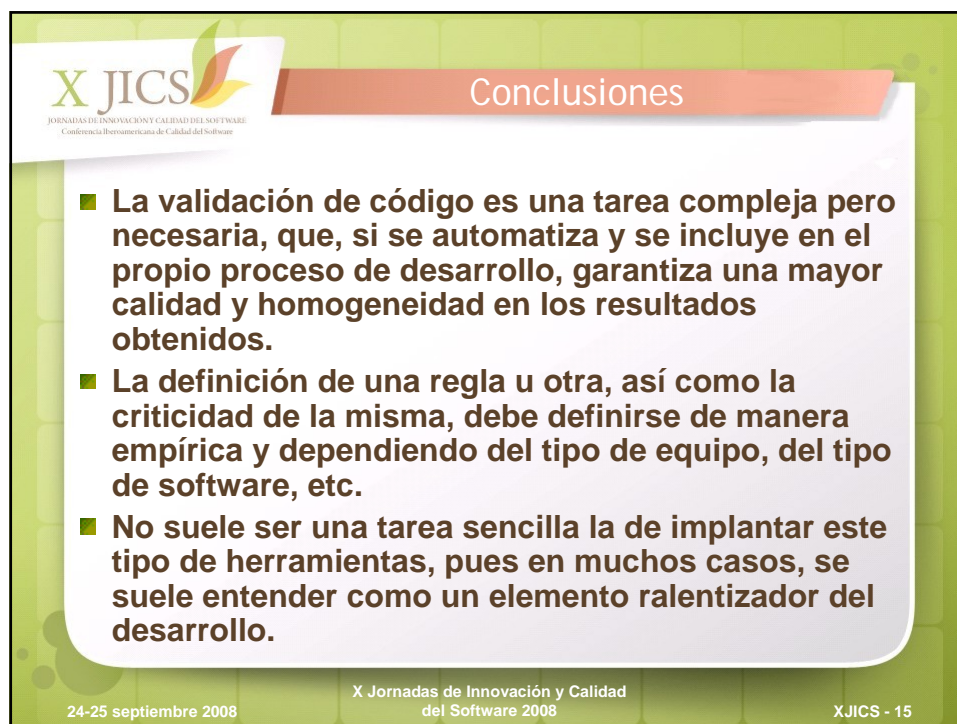


Trabajos relacionados

- **Existen otras herramientas similares a PMD:**

Herramientas gratuitas	Herramientas comerciales
Checkstyle	AppPerfect
DoctorJ	Assent
ECS/Java	Aubjex
FindBugs	AzoJavaChecker
Hammurapi	CodePro Studio
JCSC	Flaw Detector
Jikes	JStyle
JLint	JTest
JWiz	Lint4j

24-25 septiembre 2008
X Jornadas de Innovación y Calidad del Software 2008
XJICS - 14

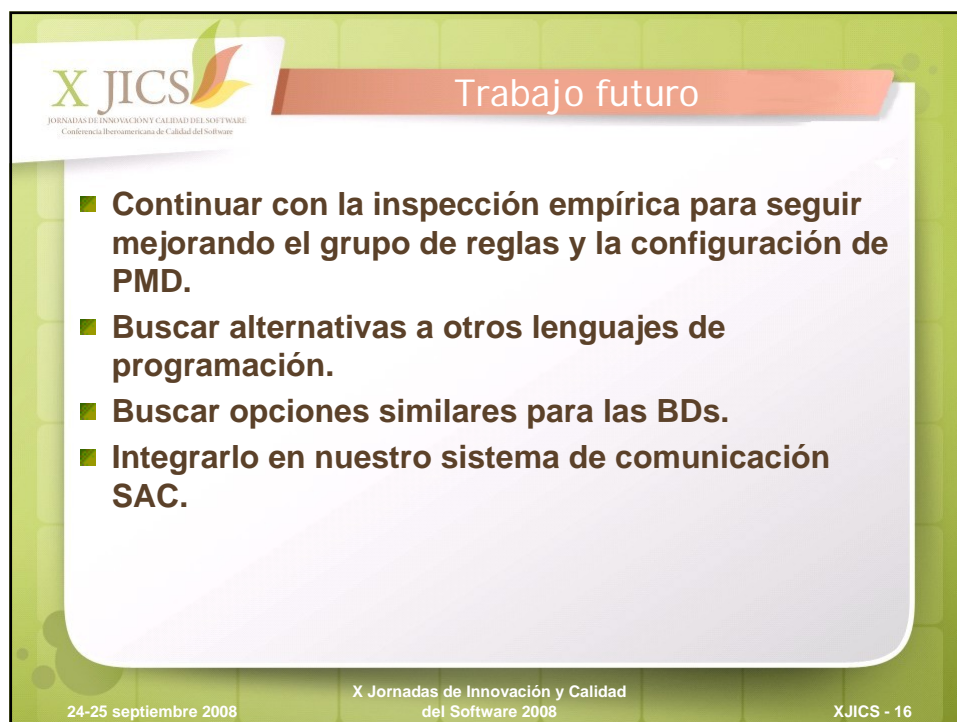


X JICS
JORNADAS DE INNOVACIÓN Y CALIDAD DEL SOFTWARE
Conferencia Iberoamericana de Calidad del Software

Conclusiones

- **La validación de código es una tarea compleja pero necesaria, que, si se automatiza y se incluye en el propio proceso de desarrollo, garantiza una mayor calidad y homogeneidad en los resultados obtenidos.**
- **La definición de una regla u otra, así como la criticidad de la misma, debe definirse de manera empírica y dependiendo del tipo de equipo, del tipo de software, etc.**
- **No suele ser una tarea sencilla la de implantar este tipo de herramientas, pues en muchos casos, se suele entender como un elemento ralentizador del desarrollo.**

24-25 septiembre 2008 X Jornadas de Innovación y Calidad del Software 2008 XJICS - 15

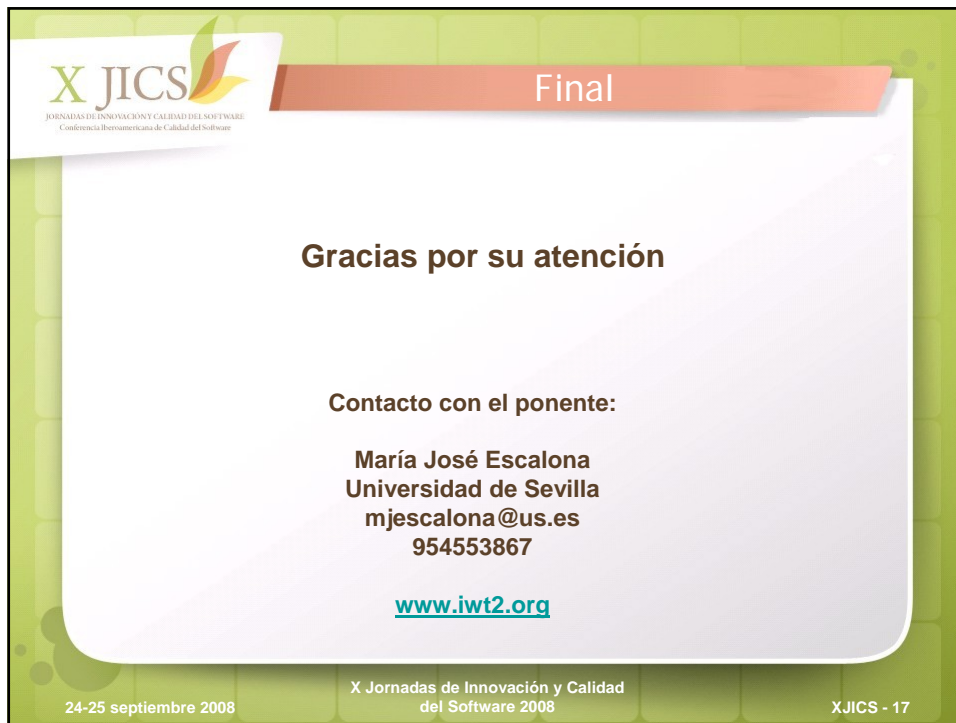


X JICS
JORNADAS DE INNOVACIÓN Y CALIDAD DEL SOFTWARE
Conferencia Iberoamericana de Calidad del Software

Trabajo futuro

- **Continuar con la inspección empírica para seguir mejorando el grupo de reglas y la configuración de PMD.**
- **Buscar alternativas a otros lenguajes de programación.**
- **Buscar opciones similares para las BDs.**
- **Integrarlo en nuestro sistema de comunicación SAC.**

24-25 septiembre 2008 X Jornadas de Innovación y Calidad del Software 2008 XJICS - 16



X JICS
JORNADAS DE INNOVACIÓN Y CALIDAD DEL SOFTWARE
Conferencia Iberoamericana de Calidad del Software

Final

Gracias por su atención

Contacto con el ponente:

María José Escalona
Universidad de Sevilla
mjescalona@us.es
954553867

www.iwt2.org

24-25 septiembre 2008 X Jornadas de Innovación y Calidad del Software 2008 XJICS - 17