

Outils de génération de documentation

Ecole ENVOL 2010

F. Langrognat
Laboratoire de Mathématiques de Besançon



PLAN

- 1 Introduction
- 2 Comment ça marche ?
 - Sur quels types de fichiers ?
 - Pour quelles sorties ?
- 3 Un exemple détaillé : Doxygen
 - Fiche d'identité
 - Balises
 - Exemples
 - Diagrammes et graphes
 - Personnalisation des sorties
- 4 Pour aller plus loin ...

PLAN

- 1 Introduction
- 2 Comment ça marche ?
 - Sur quels types de fichiers ?
 - Pour quelles sorties ?
- 3 Un exemple détaillé : Doxygen
 - Fiche d'identité
 - Balises
 - Exemples
 - Diagrammes et graphes
 - Personnalisation des sorties
- 4 Pour aller plus loin ...

Introduction

- Objectif
Générer **automatiquement** (ou presque...) une documentation technique
- Pour qui ?
Tous les développeurs
Y compris moi !
- Historique
1995-1997 : 1^{ers} outils de génération :
 - ▶ **javadoc**, ROBODoc (1995)
 - ▶ Doxygen (1997)

The screenshot shows two windows from a Java IDE. The left window displays the 'All Classes' view for 'java.awt.datatransfer', listing various classes like Clipboard, ClipboardAction, ClipboardOwner, etc. The right window shows the source code for the 'Clipboard' class, which implements the 'ClipboardAction' interface. The code includes a 'contents' field, a 'owner' field, and methods like 'getContents()', 'getOwner()', and 'setContents()'. The right window also displays the Javadoc for the class, including a description: 'A class which implements a mechanism to transfer data using cut/copy/paste operations.' and a 'See Also' section pointing to 'Toolkit.getSystemClipboard()'.

Java SDK (javadoc)

Java™ 2 Platform, Standard Edition, v 1.4.2 API Specification

This document is the API specification for the Java 2 Platform, Standard Edition, version 1.4.2.

See: Description

Java 2 Platform Packages

Package	Description
java.applet	Provides the classes necessary to create applet and the classes an applet uses to communicate with its applet context.
java.awt	Contains all of the classes for creating UI interfaces and for painting graphics and images.
java.awt.color	Provides classes for color spaces.
java.awt.datatransfer	Provides interfaces and classes for transferring data between and within applications.
java.awt.dnd	Drag and Drop is a direct manipulation GUI found in many Graphical User Interface systems that provides a mechanism to transfer information between two entities (object) associated with presentation elements in GUI.
java.awt.event	Provides interfaces and classes for dealing with different types of events fired by AWT components.
java.awt.font	Provides classes and interface relating to text.
java.awt.geom	Provides the Java 2D classes for defining, performing operations on objects related to two-dimensional geometry.
java.awt.im	Provides classes and interfaces for the input method framework.
java.awt.im.spl	Provides interfaces that enable the development of input methods that can be used with any Java runtime environment.
java.awt.image	Provides classes for creating and modifying images.
java.awt.image.renderable	Provides classes and interfaces for providing rendering-independent images.
java.awt.print	Provides classes and interfaces for a general printing API.

Java™ 2 Platform, Standard Edition, v 1.4.2 API Specification

This document is the API specification for the Java 2 Platform, Standard Edition, version 1.4.2.

See: Description

Java 2 Platform Packages

Package	Description
java.applet	Provides the classes necessary to create applet and the classes an applet uses to communicate with its applet context.
java.awt	Contains all of the classes for creating UI interfaces and for painting graphics and images.
java.awt.color	Provides classes for color spaces.
java.awt.datatransfer	Provides interfaces and classes for transferring data between and within applications.
java.awt.dnd	Drag and Drop is a direct manipulation GUI found in many Graphical User Interface systems that provides a mechanism to transfer information between two entities (object) associated with presentation elements in GUI.
java.awt.event	Provides interfaces and classes for dealing with different types of events fired by AWT components.
java.awt.font	Provides classes and interface relating to text.
java.awt.geom	Provides the Java 2D classes for defining, performing operations on objects related to two-dimensional geometry.
java.awt.im	Provides classes and interfaces for the input method framework.
java.awt.im.spl	Provides interfaces that enable the development of input methods that can be used with any Java runtime environment.
java.awt.image	Provides classes for creating and modifying images.
java.awt.image.renderable	Provides classes and interfaces for providing rendering-independent images.
java.awt.print	Provides classes and interfaces for a general printing API.

NetBeans (javadoc)

Fichier Édition Affichage Aller Signets Configuration Fenêtre Aide

Overview AllClasses

Actions API (javadoc) Ant (javadoc) Ant-Based Project Support Update Service Bootstrap (javadoc) ClassFile Reader (javadoc) Classpath API (javadoc) Command Line Parser Common Annotation Common Palette (javadoc) Common Server (javadoc) Compatibility API (javadoc) Core - IDE (javadoc) Database Explorer (javadoc) Database Explorer (javadoc) Database Explorer (javadoc)

NetBeans API List

Current Development Version

26 Aug 2010

This document provides a list of *NetBeans APIs* with a short description of what they are used for, and a table describing different types of interfaces (see [What is an API](#) to understand why we list DTOs, file formats, etc.) and with a stability category (stable and official, under development, deprecated, friend or private; see [API life-cycle](#) for more info). The aim is to provide as detailed a definition of NetBeans module external interfaces as possible and give other developers a chance to decide whether they want to depend on a particular API or not.

Some of these APIs are part of the **NetBeans Platform** (for example lookup, loaders, utilities, nodes, explorer, window systems, multiview, etc.), some of them are specific to **NetBeans IDE** (projects, javacore, diff, etc.) and some are not included in the releases at all and are just provided for download (usually via autoupdate). Basically when building an application based on *NetBeans* one is free to choose the set of modules and their APIs to satisfy one's needs.

This is a list of APIs for development version, if you want to see a list of APIs for a particular version, you may want to go to:

- 6.9.1 - Javadoc as released for NetBeans IDE 6.9.1
- 6.9 - Javadoc as released for NetBeans IDE 6.9
- 6.8 - Javadoc as released for NetBeans IDE 6.8
- 6.7 - Javadoc as released for NetBeans IDE 6.7
- 6.5 - Javadoc as released for NetBeans IDE 6.5
- 6.1 - Javadoc as released for NetBeans IDE 6.1
- 6.0 - Javadoc as released for NetBeans IDE 6.0
- 5.5.1 - Javadoc as released for NetBeans IDE 5.5.1
- 5.5 - Javadoc as released for NetBeans IDE 5.5
- 5.0 - Javadoc as released for NetBeans IDE 5.0

To get the API of your module listed here, see the documentation for the Javadoc building infrastructure.

Additional Sources of Information

- Changes since previous release
- How to use certain NetBeans APIs
- Index of all NetBeans API classes
- Extracted List of Layer APIs
- Extracted List of Property APIs

FAQ and Mailing List

Can't find what you're looking for? Try the [FAQ](#) and then [dev@openide.netbeans.org](#).

Content

- Actions API** - Actions provides system of support and utility classes for 'actions' usage in NetBeans.
- Ant** - The Ant integration module recognizes Ant build scripts, facilitates editing them as text or structurally, lets the user run targets or create shortcuts to them, etc.
- Ant-Based Project Support** - Provides the basic infrastructure by which Ant-based projects can be created, read and write configuration parameters and properties from/to disk, modify custom queries and interfaces, etc.
- Auto Update Services** - `org.netbeans.api.autoupdate.org.netbeans.api.autoupdate`
- Bootstrap** - The NetBeans launcher starts the application.
- ClassFile Reader** - The ClassFile API is designed as a fast, low memory-cost way to access classfile information.
- Classpath APIs** - Models basic aspects of the metadata surrounding list of source roots, such as the classpath.
- Command Line Parsing API** - GetsOps like infrastructure to parse command line arguments with the cooperative participation of various modules.
- Common Annotations** - Provides common annotations serving as a documentation element

Page chargée.

Fichier Édition Affichage Aller Signets Configuration Fenêtre Aide

Overview AllClasses

Actions API (javadoc) Ant (javadoc) Ant-Based Project Support Update Service Bootstrap (javadoc) ClassFile Reader (javadoc) Classpath API (javadoc) Command Line Parser Common Annotation Common Palette (javadoc) Common Server (javadoc) Compatibility API (javadoc) Core - IDE (javadoc) Database Explorer (javadoc) Database Explorer (javadoc) Database Explorer (javadoc)

NetBeans API List

Current Development Version

26 Aug 2010

This document provides a list of *NetBeans APIs* with a short description of what they are used for, and a table describing different types of interfaces (see [What is an API](#) to understand why we list DTOs, file formats, etc.) and with a stability category (stable and official, under development, deprecated, friend or private; see [API life-cycle](#) for more info). The aim is to provide as detailed a definition of NetBeans module external interfaces as possible and give other developers a chance to decide whether they want to depend on a particular API or not.

Some of these APIs are part of the **NetBeans Platform** (for example lookup, loaders, utilities, nodes, explorer, window systems, multiview, etc.), some of them are specific to **NetBeans IDE** (projects, javacore, diff, etc.) and some are not included in the releases at all and are just provided for download (usually via autoupdate). Basically when building an application based on *NetBeans* one is free to choose the set of modules and their APIs to satisfy one's needs.

This is a list of APIs for development version, if you want to see a list of APIs for a particular version, you may want to go to:

- 6.9.1 - Javadoc as released for NetBeans IDE 6.9.1
- 6.9 - Javadoc as released for NetBeans IDE 6.9
- 6.8 - Javadoc as released for NetBeans IDE 6.8
- 6.7 - Javadoc as released for NetBeans IDE 6.7
- 6.5 - Javadoc as released for NetBeans IDE 6.5
- 6.1 - Javadoc as released for NetBeans IDE 6.1
- 6.0 - Javadoc as released for NetBeans IDE 6.0
- 5.5.1 - Javadoc as released for NetBeans IDE 5.5.1
- 5.5 - Javadoc as released for NetBeans IDE 5.5
- 5.0 - Javadoc as released for NetBeans IDE 5.0

To get the API of your module listed here, see the documentation for the Javadoc building infrastructure.

Additional Sources of Information

- Changes since previous release
- How to use certain NetBeans APIs
- Index of all NetBeans API classes
- Extracted List of Layer APIs
- Extracted List of Property APIs

FAQ and Mailing List

Can't find what you're looking for? Try the [FAQ](#) and then [dev@openide.netbeans.org](#).

Content

- Actions API** - Actions provides system of support and utility classes for 'actions' usage in NetBeans.
- Ant** - The Ant integration module recognizes Ant build scripts, facilitates editing them as text or structurally, lets the user run targets or create shortcuts to them, etc.
- Ant-Based Project Support** - Provides the basic infrastructure by which Ant-based projects can be created, read and write configuration parameters and properties from/to disk, modify custom queries and interfaces, etc.
- Auto Update Services** - `org.netbeans.api.autoupdate.org.netbeans.api.autoupdate`
- Bootstrap** - The NetBeans launcher starts the application.
- ClassFile Reader** - The ClassFile API is designed as a fast, low memory-cost way to access classfile information.
- Classpath APIs** - Models basic aspects of the metadata surrounding list of source roots, such as the classpath.
- Command Line Parsing API** - GetsOps like infrastructure to parse command line arguments with the cooperative participation of various modules.
- Common Annotations** - Provides common annotations serving as a documentation element

Page chargée.

KDE (Doxygen)

Bichier Edition Affichage Ajout Signets Configuration Fenêtre Aide

KDE 3.5 API Reference

KDE API Reference API Reference Sitemap Contact Us

KMainWindowInterface Class Reference

DCOP interface to KMainWindow. More...

```
#include <KMainWindowInterface.h>
```

inheritance diagram for KMainWindowInterface:

```

graph TD
    DCOPObject --> KMainWindowInterface
    
```

[legend]

List of all members.

Public Member Functions

```

QStringList functionsDynamic () const KMainWindowInterface (KMainWindow *mainWindow)
bool processDynamic (const QString &fun, const QByteArray &data, QString &replyType, QByteArray &replyData)
~KMainWindowInterface ()
    
```

Detailed Description

DCOP interface to KMainWindow. This will provide a consistent DCOP interface to all KDE applications that use it.

Author: Tim Bernhart Gaiser <gaiser@yahoo.com>

Definition at line 40 of file kmainwindowinterface.h.

Constructor & Destructor Documentation

KMainWindowInterface::KMainWindowInterface(KMainWindow * mainWindow)

Construct a new Interface object.

Parameters: mainWindow - The parent KMainWindow object that will provide us with the KAction objects.

Definition at line 31 of file kmainwindowinterface.cpp.

KMainWindowInterface::~KMainWindowInterface()

Destructor Class Reference @ Doxygen, a free software project

Bichier Edition Affichage Ajout Signets Configuration Fenêtre Aide

KDE 3.5 API Reference

KDE API Reference API Reference Sitemap Contact Us

KHistoryCombo Class Reference

A combobox for offering a history and completion. More...

```
#include <khistorycombo.h>
```

inheritance diagram for KHistoryCombo:

```

graph TD
    QComboBox --> KHistoryCombo
    KCompletionBase --> KHistoryCombo
    KHistoryCombo --> KHistoryCombo
    
```

[legend]

List of all members.

Public Slots

```

void addHistory (const QString &item)
void clearHistory ()
    
```

Signals

```

void cleared ()
    
```

Public Member Functions

```

QStringList historyItems () const
KHistoryCombo (bool useCompletion, QWidget *parent=0L, const char *name=0L)
KHistoryCombo (QWidget *parent=0L, const char *name=0L)
KHistoryProvider * providerProvider () const
bool removeFromHistory (const QString &item)
void reset ()
void setHistoryItems (QStringList &items, bool setCompletionList)
void setHistoryItems (QStringList &items)
void setProviderProvider (KHistoryProvider *prov)
~KHistoryCombo ()
    
```

Protected Member Functions

```

void insertItems (const QStringList &items)
virtual void keyPressedEvent (QKeyEvent *)
bool useCompletion () const
    
```

Caractéristiques - Fonctionnalités

Caractéristiques

Formats d'entrée

- Code source (texte)
- Binaire

Formats de sortie

- HTML
- PDF
- Latex
- ps
- XML
- man pages
- RTF
- DocBook

Autres fonctionnalités

- Diagrammes (classes, appels, ...)
- Possibilité d'étendre à d'autres langages
- Personnalisation des sorties
- ...

PLAN

- 1 Introduction
- 2 Comment ça marche ?
 - Sur quels types de fichiers ?
 - Pour quelles sorties ?
- 3 Un exemple détaillé : Doxygen
 - Fiche d'identité
 - Balises
 - Exemples
 - Diagrammes et graphes
 - Personnalisation des sorties
- 4 Pour aller plus loin ...

PLAN

- 1 Introduction
- 2 **Comment ça marche ?**
 - **Sur quels types de fichiers ?**
 - Pour quelles sorties ?
- 3 Un exemple détaillé : Doxygen
 - Fiche d'identité
 - Balises
 - Exemples
 - Diagrammes et graphes
 - Personnalisation des sorties
- 4 Pour aller plus loin ...

Sur quels types de fichiers ?

Fichier binaire

Documentation générée à partir des binaires

Exemple : **classDoc** pour java

Fichier source

Documentation générée à partir du code source

- Grammaire du langage
- Balises spécifiques

On peut donc générer des documentations sans balise spécifique

Exemple sans balise

Date.h

```
Fichier Édition Affichage Aller Signets Sessions Outils Configuration Aide
Nouveau Ouvrir Précédent Suivant Enregistrer
Explorateur de systèmes de fichiers Documents
class Date {
public:
private:
    int _jour;
    int _mois;
    int _annee;

public:
    // Constructeurs/destructeurs
    Date();
    Date(int iJour, int iMois, int iAnnee);
    Date(const Date & iDate);
    ~Date();

    // Selecteurs
    int getJour() const;
    int getMois() const;
    int getAnnee() const;
    void setJour(int iJour);
    void setMois(int iMois);
    void setAnnee(int iAnnee);

    // Methodes
    void saisir();
    void afficher() const;

    // Surcharges d'operateurs
    bool operator < (const Date& d) const;
    bool operator > (const Date& d) const;
    bool operator <= (const Date& d) const;
    bool operator >= (const Date& d) const;
    bool operator == (const Date& d) const;
    bool operator != (const Date& d) const;

    friend istream & operator >> (istream& is, Date& d);
    friend ostream & operator << (ostream& os, const Date& d);
};
Ligne : 49 Col : 9      INS LIGNE ISO-8859-1 date.h
Terminal Chercher dans les fichiers
```

Documentation générée (html)

Fichier Édition Affichage Aller Signets Configuration Fenêtre Aide

Main Page Classes Files Search

Class List

Date Class Reference

List of all members.

Public Member Functions


Date	(int iJour, int iMois, int iAnnee)
Date	(const Date &iDate)
int	getJour () const
int	getMois () const
int	getAnnee () const
void	setJour (int iJour)
void	setMois (int iMois)
void	setAnnee (int iAnnee)
void	saisir ()
void	afficher () const
bool	operator< (const Date &d) const
bool	operator> (const Date &d) const
bool	operator<= (const Date &d) const
bool	operator>= (const Date &d) const
bool	operator== (const Date &d) const
bool	operator!= (const Date &d) const

Friends

istream &	operator>> (istream &is, Date &d)
ostream &	operator<< (ostream &os, const Date &d)

The documentation for this class was generated from the following file:

- [date.h](#)

Generated on Thu Aug 26 15:29:20 2010 by  1.6.3

Page chargée.

PLAN

- 1 Introduction
- 2 **Comment ça marche ?**
 - Sur quels types de fichiers ?
 - **Pour quelles sorties ?**
- 3 Un exemple détaillé : Doxygen
 - Fiche d'identité
 - Balises
 - Exemples
 - Diagrammes et graphes
 - Personnalisation des sorties
- 4 Pour aller plus loin ...

Pour quelles sorties ?

Quelles informations ?

- Prototype de fonctions
- Classes
- Graphes d'appels
- Diagrammes (de classes, de collaboration, ...)
- Liens vers les fichiers sources
- ...

Format des fichiers de sortie

- Différents formats (en fonction des possibilités de l'outil) :
html, pdf, latex, ps, XML, ...
- Personnalisation possible des sorties

PLAN

- 1 Introduction
- 2 Comment ça marche ?
 - Sur quels types de fichiers ?
 - Pour quelles sorties ?
- 3 **Un exemple détaillé : Doxygen**
 - Fiche d'identité
 - Balises
 - Exemples
 - Diagrammes et graphes
 - Personnalisation des sorties
- 4 Pour aller plus loin ...

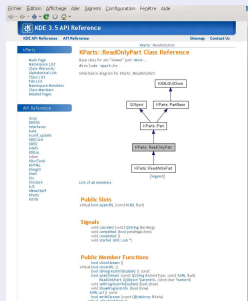
PLAN

- 1 Introduction
- 2 Comment ça marche ?
 - Sur quels types de fichiers ?
 - Pour quelles sorties ?
- 3 **Un exemple détaillé : Doxygen**
 - **Fiche d'identité**
 - Balises
 - Exemples
 - Diagrammes et graphes
 - Personnalisation des sorties
- 4 Pour aller plus loin ...

Doxygen - Fiche d'identité

- Auteur : Dimitri Van Heesch
- 1^{re} version : 1997
- OS : BSD, Linux, Mac OS, Unix, Windows
- Entrée : code source
- Sorties : (HTML, LATEX, Man Pages, RTF, XML, Qt Help Project, PDF, PS, ...)
- Nombreuses possibilités de personnalisation

Documentation KDE



Pour quels langages ?

Langages



+ Possibilités d'extension pour d'autres langages *non natifs*

Comment utilise t-on Doxygen (1) ?

En ligne de commande

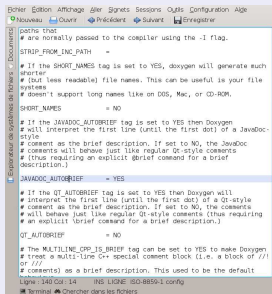
- Directement :

```
user$ doxygen *.h *.cpp
```

- Via un fichier de configuration :

- ▶ `user$ doxygen -g my_config.txt`

- ▶ `user$ kate my_config.txt`



```
Bohler Edition Affichage Aller Brevets Sessions Outils Configuration Aide
Nouveau Ouvrir Précédent Suivant Enregistrer
Documents
Explorateur de systèmes de fichiers
paths
# are normally passed to the compiler using the -I flag.
STRIP_FROM_INC_PATH =
# If the SHORT_NAMES tag is set to YES, doxygen will generate such
# (but less readable) file names. This can be useful in your file
# systems
# doesn't support long names like on DOS, Mac, or CD-ROM.
SHORT_NAMES = NO
# If the JAVADOC_AUTOBRIEF tag is set to YES then Doxygen
# will interpret the first line (until the first dot) of a JavaDoc
# comment as the brief description. If set to NO, the JavaDoc
# comments will behave just like regular Qt-style comments
# (thus requiring an explicit @brief command for a brief
# description.)
JAVADOC_AUTOBRIEF = YES
# If the QT_AUTOBRIEF tag is set to YES then Doxygen will
# interpret the first line (until the first dot) of a Qt-style
# comment as the brief description. If set to NO, the comments
# will behave just like regular Qt-style comments (thus requiring
# an explicit @brief command for a brief description.)
QT_AUTOBRIEF = NO
# The MULTILINE_CPP_IS_BRIEF tag can be set to YES to make Doxygen
# treat a multi-line C++ special comment block (i.e. a block of ///
# comments) as a brief description. This used to be the default
# behavior.
Ligne : 140 Col : 14 INS LIGNE ISO-8859-1 corrigé
Terminal Chercher dans les fichiers
```

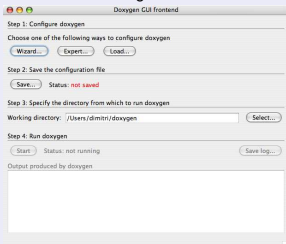
- ▶ `user$ doxygen my_config.txt`

Comment utilise t-on Doxygen (2) ?

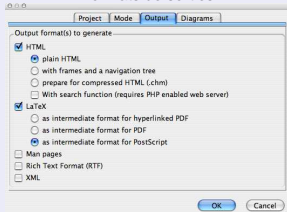
GUI

Exemple : **DoxyWizard** pour **configurer** et **lancer** doxygen

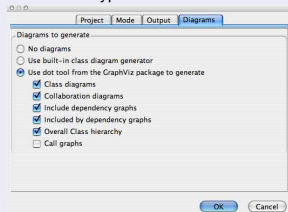
Configuration



Formats de sorties



Types de sorties



PLAN

- 1 Introduction
- 2 Comment ça marche ?
 - Sur quels types de fichiers ?
 - Pour quelles sorties ?
- 3 Un exemple détaillé : Doxygen
 - Fiche d'identité
 - **Balises**
 - Exemples
 - Diagrammes et graphes
 - Personnalisation des sorties
- 4 Pour aller plus loin ...

Balises à l'intérieur des commentaires

Les balises sont incluses dans les **commentaires interprétables par Doxygen**

En C/C++ :

- Style C avec avec deux *
`/**`
`* Documentation pour doxygen`
`*/`
- Style C++ avec avec trois /
`///`
`/// Documentation pour doxygen`
`///`
- Style C avec avec un !
`/*!`
`* Documentation pour doxygen`
`*/`
- Style C++ avec avec un !
`//!`
`//!Documentation pour doxygen`
`//!`

Balises

- **\file**
Description d'un fichier source ou d'en-tête
- **\brief**
Description courte (peut être complétée par un lien vers la description détaillée)
- **\author**
Auteur(s)
- **\version**
Version
- **\date**
Date
- **\param**
Description de paramètre(s) d'une fonction (/méthode)
- **\return**
Description de la valeur retournée
- **\bug**
Description d'un bug
- **\deprecated**
Description d'une partie de code obsolète
- **\class**
Description d'une classe

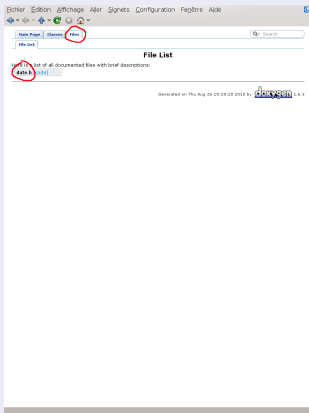
PLAN

- 1 Introduction
- 2 Comment ça marche ?
 - Sur quels types de fichiers ?
 - Pour quelles sorties ?
- 3 Un exemple détaillé : Doxygen
 - Fiche d'identité
 - Balises
 - **Exemples**
 - Diagrammes et graphes
 - Personnalisation des sorties
- 4 Pour aller plus loin ...

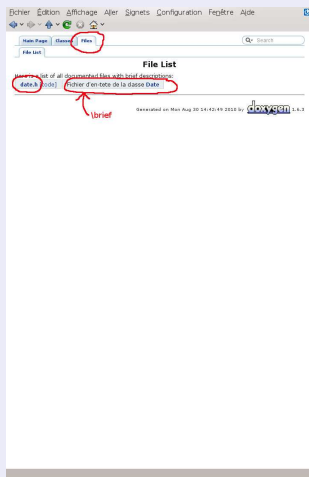
Informations sur les fichiers d'en-tête

Informations d'en-tête (1)

Sans balise doxygen



Avec balises d'en-tête doxygen



Informations d'en-tête (2)

Balises d'en-tête

```
Fichier Édition Affichage Aller Signets Sessions Outils Configuration Aide
Nouveau Ouvrir Précédent Suivant Enregistrer
explorateur de systèmes de fichiers / Documents
#ifndef DATE_H
#define DATE_H

/**
 * \file date.h
 * \author F. Langrognet
 * \version 1.1
 * \date 12 janvier 2009
 * \brief Fichier d'en-tete de la classe Date
 */

#include <iostream>
using namespace std;

class Date {
public:
private:
    int _jour;
    int _mois;
    int _annee;

public:
    //-----
    // Constructeurs/destructeurs
    //-----
    Date();
    Date(int iJour, int iMois, int iAnnee);
    Date(const Date & iDate);
    ~Date();

    //-----
    // Selecteurs
    //-----
    int getJour() const;
    int getMois() const;
    int getAnnee() const;
    void setJour(int iJour);
    void setMois(int iMois);
    void setAnnee(int iAnnee);

    //-----
    // Methodes
    //-----
    void saisir();
    void afficher() const;

    //-----
    // Surcharges d'operateurs
    //-----
    bool operator < (const Date& d) const;
    bool operator > (const Date& d) const;
};

Ligne : 1 Col : 1 INS LIGNE ISO-8859-1 date.h
Terminal Chercher dans les fichiers
```

```
Fichier Édition Affichage Aller Signets Configuration Fenêtre Aide
Main Page Classes Files
File List
date.h File Reference
Fichier d'en-tete de la classe Date. More...
#include <iostream>
Go to the source code of this file.
Classes
class Date
Detailed Description
Fichier d'en-tete de la classe Date.
Author:
F. Langrognet
Version:
1.1
Date:
12 janvier 2009
Generated on Mon Aug 30 15:16:46 2010 by doxygen 1.6.3
```

Description courte / détaillée

Description courte / détaillée (1)

Description courte / détaillée

On peut avoir :

- une description courte (**brief**)
- une description détaillée (**sans balise**)

```
#ifndef __DATE_H__
#define __DATE_H__

+ Fichier décrivant la classe Date \n
+ La classe Date est compose de 3 attributs, ... \n
... \n

+ \nfile date.h
+ \nauthor F. Langrognet
+ \nversion 1.1
+ \nrelease 12 janvier 2009

+ Brief Fichier d'en-tête de la classe Date
+ */

#include <iostream>
using namespace std;

class Date {
public:
// Constructeurs/destructeurs
Date();
Date(int iJour, int iMois, int iAnnee);
Date(const Date & iDate);
~Date();

// Selecteurs
int getJour() const;
int getMois() const;
int getAnnee() const;
void setJour(int iJour);
void setMois(int iMois);
void setAnnee(int iAnnee);

// Methodes
};
```

date.h File Reference

Fichier d'en-tête de la classe Date

More...

#include <iostream>

Go to the source code of this file.

Classes

class Date

Detailed Description

Fichier d'en-tête de la classe Date

Brief résumé de la classe Date
La classe Date est compose de 3 attributs, ...

Author: F. Langrognet

Version: 1.1

Date: 12 janvier 2009

Generated on Mon Aug 30 15:41:12 2010 by 1.6.3

Description courte / détaillée (2)

Description courte / détaillée

On peut aussi générer automatiquement la **description courte** à partir de la **description détaillée** (option JAVADOC_AUTOBRIEF à YES).

La description courte s'arrête au 1er point trouvé dans la description détaillée.

```
Fichier  Edition  Affichage  Aller  Signets  Sessions  Outils  Configuration  Aide
Nouveau  Ouvrir  Précédent  Suivant  Enregistrer

#ifndef DATE_H
#define DATE_H

/**
 * Fichier décrivant la classe Date.
 * La classe Date est composée de 3 attributs, ...\n
 * ...\\n
 *
 * \file date.h
 * \author F. Langrognet
 * \version 1.1
 * \date 12 janvier 2009
 */

#include <iostream>
using namespace std;

class Date {
public:
private:
    int _jour;
    int _mois;
    int _annee;

public:
    // Constructeurs/destructeurs
    Date();
    Date(int iJour, int iMois, int iAnnee);
    Date(const Date & iDate);
    ~Date();

    // Selecteurs
    int getJour() const;
    int getMois() const;
    int getAnnee() const;
    void setJour(int iJour);
    void setMois(int iMois);
    void setAnnee(int iAnnee);

    // Methodes
    void saisir();
    void afficher() const;
};
```

Description courte et détaillée

Il n'y a pas de \brief

```
Fichier  Edition  Affichage  Aller  Signets  Configuration  Fenêtre  Aide
Main Page  Classes  Fichiers
File List

date.h File Reference

Fichier décrivant la classe Date. More...
#include <iostream>
Go to the source code of this file.

Classes
class Date

Detailed Description
Fichier décrivant la classe Date.
La classe Date est composée de 3 attributs, ...
...

Author:
F. Langrognet

Version:
1.1

Date:
12 janvier 2009
```

Generated on Mon Aug 30 16:00:34 2010 by doxygen 1.6.3

Description d'une méthode / fonction

Description d'une méthode / fonction (1)

Description d'une méthode

Utilisation de `\param` et `\return`.

```
Echier Edition Affichage Aller Signets Sessions Outils Configuration Aide
Nouveau Ouvrir Précédent Suivant Enregistrer
Explorateur de systèmes de fichiers Documents
Date(int iJour, int iMois, int iAnnee);
/** Constructeur par copie.
 */
Date(const Date & iDate);
/** Destructeur
 */
~Date();
/**
 * Selecteur (get) de l'attribut _jour.
 */
return la valeur du jour
/**
 * Int getJour() const;
 * Selecteur (get) de l'attribut _mois.
 */
return la valeur du mois
/**
 * Int getMois() const;
 * Selecteur (get) de l'attribut _annee.
 */
return la valeur de l'annee
Int getAnnee() const;
/**
 * Selecteur (set) de l'attribut _jour.
 */
param la valeur du jour
void setJour(int iJour);
/**
 * Selecteur (set) de l'attribut _mois.
 */
param la valeur du mois
void setMois(int iMois);
/**
 * Selecteur (set) de l'attribut _annee.
 */
param la valeur de l'annee
void setAnnee(int iAnnee);
//-----
// Methodes
//-----
void saisir();
Ligne : 1 Col : 1 INS LIGNE ISO-8859-1 date.h
Terminal Chercher dans les fichiers
```

```
Echier Edition Affichage Aller Signets Configuration Fenêtre Aide
create new file (Date/Mois/Annee).
Parameters:
    Jour entier representant le jour
    Mois entier representant le mois
    Annee entier representant l'annee
Member Function Documentation
int Date::getAnnee ( ) const
Selecteur (get) de l'attribut _annee.
Returns:
    la valeur de l'annee
int Date::getJour ( ) const
Selecteur (get) de l'attribut _jour.
Returns:
    la valeur du jour
int Date::getMois ( ) const
Selecteur (get) de l'attribut _mois.
Returns:
    la valeur du mois
void Date::setAnnee ( int iAnnee )
Selecteur (set) de l'attribut _annee.
Parameters:
    la valeur de l'annee
void Date::setJour ( int iJour )
Selecteur (set) de l'attribut _jour.
Parameters:
    la valeur du jour
void Date::setMois ( int iMois )
Selecteur (set) de l'attribut _mois.
Parameters:
    la valeur du mois
The documentation for this class was generated from the following file:
    • date.h
Generated on Mon Aug 30 16:48:30 2010 by doxygen 1.6.3
```


Description d'une méthode / fonction (2)

Balises pour une méthode / fonction

```
Richier Edition Affichage Aller Signets Sessions Outils Configuration Aide
Nouveau Ouvrir Précédent Suivant Enregistrer Enregistrer sous
Explorateur de systèmes de fichiers Documents
#include <ostream>
using namespace std;

class Date {
private:
    int _jour;
    int _mois;
    int _annee;

public:
    /** Constructeur par défaut.
     * crée une nouvelle date (01/01/0000). */
    Date();

    /** Constructeur par initialisation.
     * crée une nouvelle date (iJour/Mois/iAnnee).
     * param iJour entier représentant le jour
     * param iMois entier représentant le mois
     * param iAnnee entier représentant l'annee
     */
    Date(int iJour, int iMois, int iAnnee);

    /** Constructeur par copie.
     */
    Date(const Date & iDate);

    /** Destructeur
     */
    ~Date();

    // Selecteurs
    // ...
    int getJour() const;
    int getMois() const;
    int getAnnee() const;
    void setJour(int iJour);
    void setMois(int iMois);
    void setAnnee(int iAnnee);

    // Methodes
    // ...
    void saisir();
    void afficher() const;
    // ...
};
```

Description courte et détaillée

Description courte uniquement

Ligne : 31 Col : 6 INS LIGNE ISO-8859-1 date.h
Terminal Chercher dans les fichiers

```
Richier Edition Affichage Aller Signets Configuration Fenêtre Aide
Public Member Functions
Date()
    Constructeur par défaut
Date (int iJour, int iMois, int iAnnee)
    Constructeur par initialisation.
Date (const Date &iDate)
    Constructeur par copie.
~Date()
    Destructeur
int getJour () const
int getMois () const
int getAnnee () const
void setJour (int iJour)
void setMois (int iMois)
void setAnnee (int iAnnee)
void saisir ()
void afficher () const
bool operator= (const Date &d) const
bool operator< (const Date &d) const
bool operator>= (const Date &d) const
bool operator== (const Date &d) const
bool operator!= (const Date &d) const

Friends
ostream & operator<> (ostream &os, const Date &d)
ostream & operator<< (ostream &os, const Date &d)

Constructor & Destructor Documentation
Date::Date ( )
    Constructeur par défaut.
    crée une nouvelle date (01/01/0000).

Date::Date ( int iJour,
             int iMois,
             int iAnnee
             )
    Constructeur par initialisation.
    crée une nouvelle date (iJour/Mois/iAnnee).

Parameters:
    iJour    entier représentant le jour
    iMois    entier représentant le mois
    iAnnee   entier représentant l'annee

The documentation for this class was generated from the following file:
    • date.h

Generated on Mon Aug 30 16:26:22 2010 by doxygen 1.8.4
```

Description courte

Description détaillée

PLAN

- 1 Introduction
- 2 Comment ça marche ?
 - Sur quels types de fichiers ?
 - Pour quelles sorties ?
- 3 **Un exemple détaillé : Doxygen**
 - Fiche d'identité
 - Balises
 - Exemples
 - **Diagrammes et graphes**
 - Personnalisation des sorties
- 4 Pour aller plus loin ...

Diagrammes de classes (1)

Fonctionnement par défaut

Hiérarchie de classes

Fichier Édition Affichage Aller Signets Configuration

Main Page Classes Files Search

Class List Class Hierarchy Class Members

Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

- Date
- **Personne**
 - **Adulte**
 - **Enfant**
- Population

Generated on Wed Sep 8 10:26:04 2010 by **doxygen** 1.6.3

Fichier Édition Affichage Aller Signets Configuration

Main Page Classes Files Search

Class List Class Hierarchy Class Members

Adulte Class Reference

Classe représentant un adulte (caractérisé par sa profession).
More...

```
#include <adulte.h>
```

Inheritance diagram for Adulte:

```
graph BT;
    Adulte --> Personne;
```

List of all members.

Public Member Functions

Adulte ()
Constructeur par défaut.
Adulte (const char *strNom, const char *strPrenom, const Date &date, const char *strProfession)
Constructeur par initialisation.

Diagrammes de classes (2)

Personnalisation

Avec le logiciel **graphviz** (et l'option HAVE_DOT=YES)

- Hiérarchie de classes
- Diagramme de collaboration

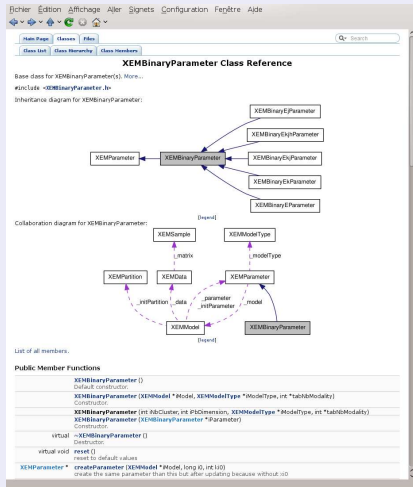
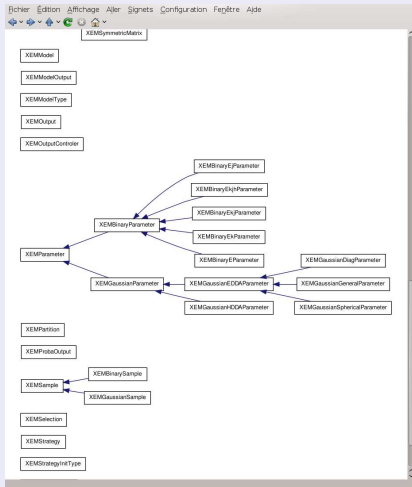
The screenshot shows the Doxygen web interface. The top navigation bar includes 'Fichier', 'Édition', 'Affichage', 'Aller', 'Signets', 'Configuration', 'Fenêtre', and 'Aide'. Below the navigation bar are tabs for 'Main Page', 'Classes', and 'Files'. The 'Classes' tab is active, and sub-tabs for 'Class List', 'Class Hierarchy', and 'Class Members' are visible. The main content area is titled 'Graphical Class Hierarchy' and contains a diagram showing the relationships between classes: 'Date' is a separate class; 'Personne' is the base class, with 'Adulte' and 'Enfant' as subclasses; and 'Population' is another class. At the bottom, it says 'Generated on Wed Sep 8 10:23:42 2010 by doxygen 1.4.3'.

The screenshot shows the Doxygen web interface for the 'Personne' class reference. The top navigation bar is the same as in the previous screenshot. The 'Classes' tab is active, and the 'Class Hierarchy' sub-tab is selected. The main content area is titled 'Personne Class Reference' and contains the following information:

- A brief description: 'Classe représentant une personne: nom, prenom et date de naissance. More...'
- The include directive: '#include <personne.h>'
- The text 'Inheritance diagram for Personne:' followed by an inheritance diagram showing 'Personne' at the top, with 'Adulte' and 'Enfant' below it, both inheriting from 'Personne'.
- The text 'Collaboration diagram for Personne:' followed by a collaboration diagram showing 'Date' and 'Personne' boxes. An arrow points from 'Personne' to 'Date' with the label 'dateDeNaissance' and a multiplicity of '[1]' at the 'Date' end.
- The text 'List of all members.'
- The section 'Public Member Functions' with three entries:
 - Personne ()**: Constructeur par défaut.
 - Personne (const char *Prenom, const char *Nom, const Date &IDate)**: Constructeur par initialisation.
 - Personne (const Personne &)**: Constructeur par copie.

Diagrammes de classes (3)

Autre exemple



Graphes d'appels

- Graphes d'appels (et l'option CALL_GRAPH=YES)
- Graphes d'appelants (et l'option CALLER_GRAPH=YES)

The screenshot shows a software interface with a menu bar (Fichier, Edition, Affichage, Aller, Signets, Configuration, Fenêtre, Aide) and a toolbar. The main content area displays call graphs for several functions:

- Personne::Personne () const [Personne.h: 6]**: A constructor call graph showing 'Personne::Personne' calling 'Personne::getDateDeNaissance()', 'Personne::getSexe()', and 'Personne::getNom()'. Below it is a 'Distributeur' section.
- Membre Function Documentation**: A section for 'void Personne::afficher () const [id.h:11]'. It includes a description in French and a call graph showing 'Personne::afficher' calling 'Date::afficher'.
- const char * Personne::getSexe () const**: A function returning a string. It includes a description in French and a call graph showing 'Personne::getSexe' calling 'Personne::operation()' multiple times.
- const char * Personne::getNom () const**: A function returning a string. It includes a search bar and an 'Options...' button.

The screenshot shows a software interface with a menu bar (Fichier, Edition, Affichage, Aller, Signets, Configuration, Fenêtre, Aide) and a toolbar. The main content area displays call graphs for XEMGaussian functions:

- Table of inverse of sept of determinant of covariance matrix for each cluster**: A call graph for 'XEMGaussianDDAParameter::getTabInSigDeSigma' showing it calls 'XEMGaussianDDAParameter::recup' and 'XEMGaussianDDAParameter::XEMGaussianDDAParameter'.
- XEMMatrix ** XEMGaussianDDAParameter::getTabSigma () const [id.h:16]**: A selector function. Its call graph shows 'XEMGaussianDDAParameter::getTabSigma' calling 'XEMGaussianDDAParameter::rUBER', 'XEMGaussianDDAParameter::recup', 'XEMGaussianDiagParameter::XEMGaussianDiagParameter', 'XEMGaussianGeneralParameter::XEMGaussianGeneralParameter', and 'XEMGaussianSphericalParameter::XEMGaussianSphericalParameter'.
- void XEMGaussianDDAParameter::initForUser_PARTITION (int & nbInitializedCluster, int & tabNbInitializedCluster, virtual)**: A function with virtual attributes. Its call graph shows 'XEMGaussianDDAParameter::initForUser_PARTITION' calling 'XEMGaussianParameter::computeGlobalDataBalance', 'XEMGaussianParameter::computeTabMeanUser_PARTITION', 'XEMGaussianParameter::computeMeanOne', 'XEMDiagMatrix::add', 'XEMModel::getSample', 'XEMModel::getData', and 'XEMModel::getInPartition'.

PLAN

- 1 Introduction
- 2 Comment ça marche ?
 - Sur quels types de fichiers ?
 - Pour quelles sorties ?
- 3 Un exemple détaillé : Doxygen
 - Fiche d'identité
 - Balises
 - Exemples
 - Diagrammes et graphes
 - **Personnalisation des sorties**
- 4 Pour aller plus loin ...

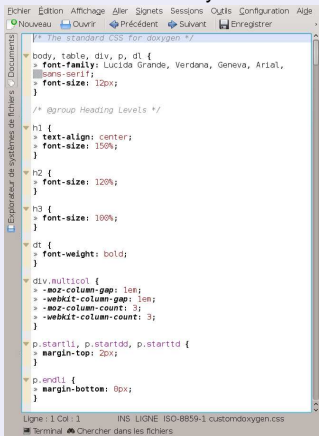
Personnalisation des sorties (1)

Sortie HTML

Exemple de personnalisation :

```
user$ doxygen -w html header.html footer.html customdoxygen.css
```

Feuille de style



```
/* The standard CSS for doxygen */
body, table, div, p, dl {
  > font-family: Lucida Grande, Verdana, Geneva, Arial,
  > sans-serif;
  > font-size: 12px;
}

/* @group Heading Levels */

h1 {
  > text-align: center;
  > font-size: 150%;
}

h2 {
  > font-size: 120%;
}

h3 {
  > font-size: 100%;
}

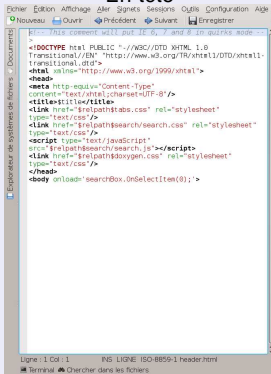
dt {
  > font-weight: bold;
}

div.multicol {
  > -moz-column-gap: 1em;
  > -webkit-column-gap: 1em;
  > -moz-column-count: 3;
  > -webkit-column-count: 3;
}

p.startli, p.starttd, p.starttd {
  > margin-top: 2px;
}

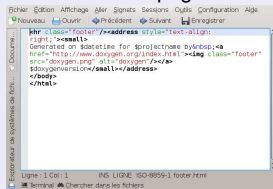
p.endli {
  > margin-bottom: 0px;
}
```

En-tête



```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type"
content="text/xhtml;charset=UTF-8"/>
<title>Title/Title</title>
<link href="$relpath$tabs.css" rel="stylesheet"
type="text/css"/>
<link href="$relpath$search/search.css" rel="stylesheet"
type="text/css"/>
<script type="text/javascript"
src="$relpath$search/search.js"></script>
<link href="$relpath$doxygen.css" rel="stylesheet"
type="text/css"/>
</head>
<body onload="searchBox.onSelect(0);">
```

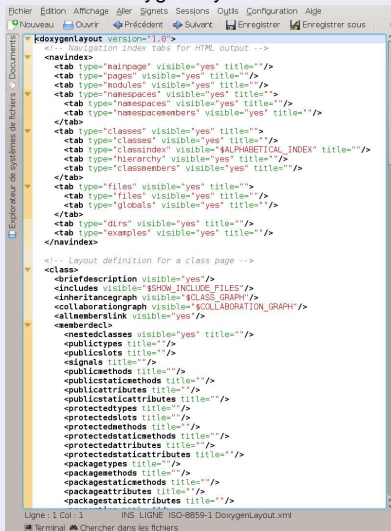
Pied de page



```
<hr class="footer"><address style="text-align:
right;><b>doxall</b>
Generated on <date> for <projectname> by <br/><b>doxall</b>
href="http://www.doxygen.org/index.html"><img alt="doxygen logo" data-bbox="910 560 950 580" class="logo" style="float: right; margin-left: 10px; height: 1em; vertical-align: middle;"/>
doxygen version <small> </small> </address>
</body>
</html>
```


- Création d'un fichier layout :
user\$ doxygen -l
- Edition du fichier DoxygenLayout :
user\$ kate DoxygenLayout.xml
- Edition du fichier de configuration
(*LAYOUT_FILE = DoxygenLayout.xml*) :
user\$ kate config.xml
- Execution de doxygen :
user\$ doxygen config.txt

DoxygenLayout.xml



```
kdoxygenlayout version="1.0">
  <!-- Navigation index tabs for HTML output -->
  <navindex>
    <tab type="mainpage" visible="yes" title="" />
    <tab type="pages" visible="yes" title="" />
    <tab type="modules" visible="yes" title="" />
    <tab type="namespaces" visible="yes" title="" />
    <tab type="namespaces" visible="yes" title="" />
    <tab type="namespacemembers" visible="yes" title="" />
  </tab>
  <tab type="classes" visible="yes" title="" />
  <tab type="classes" visible="yes" title="" />
  <tab type="classindex" visible="$ALPHABETICAL_INDEX" title="" />
  <tab type="hierarchy" visible="yes" title="" />
  <tab type="classmembers" visible="yes" title="" />
  </tab>
  <tab type="files" visible="yes" title="" />
  <tab type="files" visible="yes" title="" />
  <tab type="globals" visible="yes" title="" />
  </tab>
  <tab type="dirs" visible="yes" title="" />
  <tab type="examples" visible="yes" title="" />
</navindex>

  <!-- layout definition for a class page -->
  <class>
    <brieffdescription visible="yes" />
    <includes visible="$SHOW_INCLUDE_FILES" />
    <inheritancegraph visible="$CLASS_GRAPH" />
    <collaborationgraph visible="$COLLABORATION_GRAPH" />
    <allmemberlink visible="yes" />
    <memberdecl>
      <nestedclasses visible="yes" title="" />
      <publictypes title="" />
      <publicslots title="" />
      <signals title="" />
      <publicmethods title="" />
      <publicstaticmethods title="" />
      <publicattributes title="" />
      <publicstaticattributes title="" />
      <protectedtypes title="" />
      <protectedslots title="" />
      <protectedmethods title="" />
      <protectedstaticmethods title="" />
      <protectedattributes title="" />
      <protectedstaticattributes title="" />
      <packagetypes title="" />
      <packagemethods title="" />
      <packagestaticmethods title="" />
      <packageattributes title="" />
      <packagestaticattributes title="" />
    </memberdecl>
  </class>

```

PLAN

- 1 Introduction
- 2 Comment ça marche ?
 - Sur quels types de fichiers ?
 - Pour quelles sorties ?
- 3 Un exemple détaillé : Doxygen
 - Fiche d'identité
 - Balises
 - Exemples
 - Diagrammes et graphes
 - Personnalisation des sorties
- 4 Pour aller plus loin ...

Quel outil de génération de documentation choisir ?

Quel langage ?

Language support

The programming languages the generators recognize.

[edit]

	C/C++	Java	C#	VB / VBScript	Delphi / Pascal	Ada	D	IDL	.NET	Access
Autoduck	Yes	No	No	Yes	No	No	No	No	No	No
BI Documenter	No	No	No	No	No	No	No	No	Yes	No
classdoc	No	Yes	No	No	No	No	No	No	No	No
CppDoc	Yes	No	No	No	No	No	No	No	No	No
DB Manual	No	No	No	No	No	No	No	No	Yes	No
DBDesc	No	No	No	No	No	No	No	No	Yes	No
DBScribe	No	No	No	No	No	No	No	No	No	No
Ddoc	No	No	No	No	No	No	Yes	No	No	No
devscribe	No	No	Yes	No	No	No	No	No	No	No
Doc-O-Matic	Yes	Yes	No	Yes	Yes	No	No	Yes	Yes	No
DOC++	Yes	Yes	No	No	No	No	No	Yes	No	No
Document X	C++/C1 only	No	Yes	Yes	No	No	No	Yes	Yes	Yes
Doxygen	Yes	Yes	Yes	No	No	No	Partial	Yes	No	No
Epydoc	No	No	No	No	No	No	No	No	No	No
fpdoc	No	No	No	No	Yes	No	No	No	No	No
GenHelp	Yes	No	Yes	No	Yes	No	No	No	Yes	No
HeaderDoc	No	No	No	No	No	No	No	No	No	No
HeaderDoc	Yes	Yes	No	No	Yes	No	No	Yes	No	No
Help Generator	Yes	No	Yes	Yes	No	No	No	No	Yes	Yes
HyperSQL	No	No	No	No	No	No	No	No	No	No
Image 4D	Yes	Yes	No	No	No	No	No	No	No	No
Javadoc	No	Yes	No	No	No	No	No	No	No	No
JGrouseDoc	No	No	No	No	No	No	No	No	No	No
JSdoc	No	No	No	No	No	No	No	No	No	No
JsDoc Toolkit	No	No	No	No	No	No	No	No	No	No
KDOC	Yes	No	No	No	No	No	No	Yes	No	No
Natural Docs	Partial	Partial	Yes	Partial	Partial	Partial	No	No	No	No
NDoc	No	No	Yes	No	No	No	No	No	Yes	No
phpDocumenter	No	No	No	No	No	No	No	No	No	No
phpSimpleDoc	No	No	No	No	No	No	No	No	No	No
Project Analyzer	No	No	No	Yes	No	No	No	No	Yes	No
RDoc	Partial	No	No	No	No	No	No	No	No	No
ROBODoc	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	No
TwinText	Yes	Yes	Yes	Yes	Yes	Yes	Partial	Yes	Yes	No
Universal Report	Yes	Yes	Yes	Yes	Yes	Yes	No	No	Yes	Yes
VBDoc	No	No	No	Yes	No	No	No	No	No	No
VSdocman	No	No	Yes	Yes	No	No	No	No	No	No
YARD	No	No	No	No	No	No	No	No	Yes	No
	C/C++	Java	C#	VB / VBScript	Delphi / Pascal	Ada	D	IDL	.NET	Access

Quel outil de génération de documentation choisir ?

Quel langage ? ... suite

	PHP	Perl	Python	Ruby	JavaScript	ActionScript	PL/SQL	Tcl	Haskell	Any With Comments ²
Autodock	No	No	No	No	No	No	No	No	No	No
BI Documenter	No	No	No	No	No	No	No	No	No	No
classdoc	No	No	No	No	No	No	No	No	No	No
CppDoc	No	No	No	No	No	No	No	No	No	No
DB Manual	No	No	No	No	No	No	No	No	No	No
DBScribe	No	No	No	No	No	No	Yes	No	No	No
Ddoc	No	No	No	No	No	No	No	No	No	No
devscribe	No	No	No	No	No	No	No	No	No	No
Doc-O-Matic	Yes	No	No	No	Yes	No	No	No	No	Yes
DOC++	No	No	No	No	No	No	No	No	No	No
Document! X	No	No	No	No	No	No	Yes	No	No	No
Doxygen	Yes	No	Yes	No	No	No	No	No	No	No
Epydoc	No	No	Yes	No	No	No	No	No	No	No
fpdoc	No	No	No	No	No	No	No	No	No	No
GenHelp	No	No	No	No	No	No	No	No	No	No
HeaderDoc	No	No	No	No	No	No	No	No	Yes	No
HeaderDoc	Yes	Yes	No	No	Yes	No	No	No	No	No
HyperSQL	No	No	No	No	No	No	Yes	No	No	No
Image4D	No	No	No	No	No	No	No	No	No	No
javadoc	No	No	No	No	No	No	No	No	No	No
jGreuseDoc	No	No	No	No	Yes	No	No	No	No	No
jsDoc	No	No	No	No	Yes	No	No	No	No	No
jsDoc Toolkit	No	No	No	No	Yes	No	No	No	No	No
KDOC	No	No	No	No	No	No	No	No	No	No
Natural Docs	Partial	Yes	Partial	Partial	Partial	Yes	Partial	Partial	No	Yes
nDoc	No	No	No	No	No	No	No	No	No	No
phpDocumentor	Yes	No	No	No	No	No	No	No	No	No
phpSimpleDoc	Yes	No	No	No	No	No	No	No	No	No
Project Analyzer	No	No	No	No	No	No	No	No	No	No
RDoc	No	No	Yes	Yes	No	No	No	No	No	No
ROBODoc	Yes	Yes	No	Yes	Yes	Partial	Yes	Yes	No	Yes
ThimbleDoc	Yes	No	No	No	No	No	No	No	No	No
TwinText	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes
Universal Report	Yes	Yes	Yes	No	Yes	Yes	Yes	No	No	No
VBDOC	No	No	No	No	No	No	No	No	No	No
VSDocman	No	No	No	No	No	No	No	No	No	No
YARD	No	No	No	Yes	No	No	No	No	No	No
	PHP	Perl	Python	Ruby	JavaScript	ActionScript	PL/SQL	Tcl	Haskell	Any With Comments ²

Quel outil de génération de documentation choisir ?

Quel OS ?

Operating system support

The operating systems the generators can run on.

[edit]

	Windows	Mac OS X	Linux	BSD	Unix
Autodock	Yes	No	No	No	No
BI Documenter	Yes	No	No	No	No
classdoc	Yes	Yes	Yes	Yes	Yes
CppDoc	Yes	No	Yes	No	No
DB Manual	Yes	No	No	No	No
DBDesc	Yes	No	No	No	No
DBScribe	Yes	No	No	No	No
Odac	Yes	Yes?	Yes	Yes?	No
docscribe	Yes	No	No	No	No
Doc-G-Matic	Yes	No	No	No	No
DOC++	Yes	Yes?	Yes	Yes	Yes
Document! X	Yes	No	No	No	No
Doxygen	Yes	Yes	Yes	Yes	Yes
Epydoc	Yes	Yes	Yes	Yes	Yes
fpdoc	Yes	Yes	Yes	Yes	Yes
GenHelp	Yes	No	No	No	No
Harddoc	Yes	Yes	Yes	Yes	Yes
HeaderDoc	No	Yes	Yes	Yes	Yes
Help Generator	Yes	No	No	No	No
HyperSQL	Yes	Yes	Yes	Yes	Yes
Imagix 4D	Yes	No	Yes	No	Yes
Javadoc	Yes	Yes	Yes	Yes	Yes
JGrouseDoc	Yes	Yes	Yes	Yes	Yes
JSdoc	Yes	Yes	Yes	Yes	Yes
JDoc Toolkit	Yes	Yes	Yes	Yes	Yes
KDOC	Partial	Yes	Yes	Yes	Yes
Natural Docs	Yes	Yes	Yes	Yes	Yes
NDoc	Yes	No	No	No	No
phpDocumentor	Yes	Yes	Yes	Yes	Yes
phpSimpleDoc	Yes	Yes	Yes	Yes	Yes
Project Analyzer	Yes	No	No	No	No
RDoc	Yes	Yes	Yes	Yes	Yes
RDocDoc	Yes	Yes	Yes	Yes	Yes
Sandcastle	Yes	No	No	No	No
ThimbleDoc	Yes	Yes	Yes	Yes	Yes
TwinText	Yes	No	No	No	No
Universal Report	Yes	No	No	No	No
VBDOX	Yes	No	No	No	No
V5docman	Yes	No	No	No	No
YARD	Yes	Yes	Yes	Yes	Yes
	Windows	Mac OS X	Linux	BSD	Unix

Note (4): GNU D Compiler.

Quel outil de génération de documentation choisir ?

Format de sortie

Output formats

[\[edit\]](#)

The output formats the generators can write:

	HTML ¹	CHM ²	RTF ³	PDF ⁴	LaTeX ⁵	PostScript ⁶	man pages ⁷	DocBook ⁸	XML ⁹
Autobuck	Yes	No	Yes	No	No	No	No	No	No
Bit Documenter	Yes	Yes	No	No	No	No	No	No	No
classdec	Yes	No	No	No	No	No	No	No	No
CppDec	Yes	No	No	No	No	No	No	No	No
DB Manual	Yes	Yes	No	No	No	No	No	No	No
DBDesc	Yes	No	Yes	Yes	No	No	No	No	Yes
DBScribe	Yes	Yes	Yes	No	No	No	No	No	No
Docc	Yes	Yes ⁴	No	Yes ⁴	Yes ⁴	Yes ⁴	Yes ⁴	No	Yes ⁴
devscribe	Yes	No	No	No	No	No	No	No	Yes
Doc-O-Matic	Yes	Yes	Yes	Yes	No	No	No	No	Yes
DOCx	Yes	No	No	Yes ⁷	Yes	Yes ⁷	No	No	No
Document! X	Yes	No	No	No	No	No	No	No	No
Dosygen	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes
Epydoc	Yes	No	No	Yes	indirectly ⁵	indirectly ⁵	No	No	No
fpdoc	Yes	Yes	Yes	No	Yes	No	Yes	No	Yes
Genhelp	Yes	Yes	Yes	No	No	No	No	No	No
Haddock	Yes	Yes	No	No	No	No	No	Partial	No
HeaderDoc	Yes	No	No	No	No	No	Yes	No	Yes
Help Generator	Yes	Yes	Yes	No	No	No	No	No	Yes
HyperSQL	Yes	No	No	No	No	No	No	No	No
Imagix 4D	Yes	No	Yes	No	No	No	No	No	No
Javadoc	Yes	indirectly ⁵	indirectly ⁵	indirectly ⁵	indirectly ⁵	indirectly ⁵	indirectly ⁵	indirectly ⁵	indirectly ⁵
JGrouseDoc	No	No	No	No	No	No	No	No	Yes
JSDec	Yes	No	No	No	No	No	No	No	No
JsDoc Toolkit	Yes	No	No	No	No	No	No	No	Yes+JSON
KDOC	Yes	No	No	No	No	No	No	No	No
Natural Docs	Yes	No	Natural	No	No	No	No	No	No
RDdoc	Yes	Yes	No	No	No	No	No	No	No
phpDocumentor	Yes	Yes	No	Yes	No	No	Yes	Yes	Yes
phpSimpleDoc	Yes	No	No	No	No	No	No	No	No
Project Analyzer	Yes	Yes	Yes	Yes	No	No	No	No	No
Sandcastle	Yes	Yes	No	No	No	No	No	No	No
RDoc	Yes	No	No	No	No	No	No	No	No
ROBODoc	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
ThimbleDoc	Yes	No	No	No	No	No	No	No	No
TwineText	Yes	Yes	No	No	No	No	No	No	No
Universal Report	Yes	No	Yes	Yes	Yes	Yes	No	No	No
VBDOM	Yes	Yes	Yes	No	No	No	No	No	No
VSdocman	Yes	Yes	Yes	No	No	No	No	No	Yes
YARD	Yes	No	No	No	No	No	No	No	No
	HTML	CHM	RTF	PDF	LaTeX	Postscript	man pages	DocBook	XML

Quelques références

- Comparaison des outils de génération de documentation :
en.wikipedia.org/wiki/Comparison_of_documentation_generators
- Doxygen :
 - ▶ Doxygen (officiel) :
www.stack.nl/~dimitri/doxygen/
 - ▶ Manuel d'utilisation Doxygen :
www.stack.nl/~dimitri/doxygen/manual.html
 - ▶ Wikipedia :
fr.wikipedia.org/wiki/Doxygen
 - ▶ Initiation à Doxygen pour le C++ :
franckh.developpez.com/tutoriels/outils/doxygen/
- javadoc
 - ▶ officiel :
download.oracle.com/javase/1.5.0/docs/tooldocs/windows/javadoc.html
 - ▶ Wikipedia :
fr.wikipedia.org/wiki/Javadoc

FIN

Merci de votre attention

Florent Langrognat