



– Porphyre 2003 –

Serveur de Contenu

Version 1.1

Référence du document : DFS-SERVCONT-v1.1

Date de création : 12/02/2003

Dernière modification : 17/10/2003

Etat du document : en cours

Nombre de pages : 31

Rédacteur : Régine TRIBOLLET

Équipe : Informatique

Validation : Chef de Projet

Responsable thèse

SOMMAIRE

1.	POURQUOI UN SERVEUR DE CONTENU ?	3
2.	TERMINOLOGIE	4
3.	ARCHITECTURE	5
3.1	ARCHITECTURE PORPHYRE 2002 : VERSION SERVLET	5
3.2	PORPHYRE 2003 : ARCHITECTURE ET CHOIX TECHNOLOGIQUES	6
4.	LES TRAITEMENTS A LA DEMANDE	9
4.1	DOCUMENTS DE TYPE « TEXTE »	9
4.2	DOCUMENTS DE TYPE « IMAGE »	12
4.3	DOCUMENTS DE TYPE « EPURE »	19
5.	CONCLUSION	24
	ANNEXES	25
1.	SYNTAXES : RECAPITULATIF	25
2.	PACKAGES NECESSAIRES	26
3.	EVOLUTIONS POSSIBLES	27

1. **POURQUOI UN SERVEUR DE CONTENU ?**

La finalité du système informatique Porphyre est d'offrir à la communauté scientifique, en particulier dans le domaine de l'archéologie, un système de recherche d'information permettant d'assister le lecteur de publications savantes en le soulageant des aspects répétitifs de son activité et en le laissant se concentrer sur les aspects créatifs, intuitifs et à haut niveau d'abstraction.

Le serveur Porphyre 2002 permet la gestion des structures de chaque usager et un serveur Web étendu permet l'extraction des fragments de textes et images en fonction des différents modes de visualisation.

Le support applicatif de ce projet est un corpus hétérogène de documents archéologiques : textes, images ou épures ...

La version 2002, actuellement en exploitation, du serveur Web étendu ne donne pas pleinement satisfaction : les temps d'exécution sont trop longs, certaines fonctionnalités nécessitent des évolutions.

2. TERMINOLOGIE

Applet : programme qui est exécuté dans un browser : une applet Java est téléchargée du serveur Web à l'ordinateur de consultation. L'applet est ensuite exécutée sur l'ordinateur de consultation (poste client).

Document : source sur laquelle nous allons travailler à l'aide de Porphyre : Texte, Image ou Épure. Un document définit ce qui sera stocké physiquement sur le serveur.

EFA : École Française d'Athènes.

Épure : dessins vectoriels en 2D. Elles sont au format SVG.

Fragment : portion d'un document.

HTML (HyperText Markup Language) : langage de description des pages Web.

ImageMagick : librairie Open Source permettant d'effectuer des traitements sur des images.

INSA : Institut National des Sciences Appliquées.

JPEG : format de fichier de représentation d'image numérique permettant par un procédé de compression de conserver une qualité d'image suffisante sans pour autant nécessiter une grande capacité de stockage.

NetPbm : librairie Open Source permettant d'effectuer des traitements sur des images.

OS : Operating System= système d'exploitation

PHP (Pre-HyperTexte-Processor) : PHP est un langage utilisé dans des applications web pour écrire des scripts HTML.

Serveur : ordinateur dédié à fournir des services ou/et des données aux autres machines du réseau.

Servlet : applet côté serveur.

SVG (Scalable Vector Graphics) : recommandation W3C qui spécifie un vocabulaire XML utilisable seul ou inclus dans d'autres documents pour décrire des dessins vectoriels en 2D

XML (eXtensible Markup Language) : nouveau langage de balisage de documents.

W3C : consortium du World Wide Web.

3. ARCHITECTURE

3.1 ARCHITECTURE PORPHYRE 2002 : VERSION SERVLET

Dans la version de Porphyre 2002, les traitements sont effectués par le biais de Servlets.

Cette version du serveur Web étendu ne donne pas pleinement satisfaction :

- les temps d'exécution sont trop longs ;
- certaines fonctionnalités n'existent pas (par exemple les traitements sur les documents de type épures) et
- certaines fonctionnalités nécessitent des évolutions : seules les images au format JPEG peuvent être traitées par la version Porphyre 2002 (version "Servlet").

Pour cela, il faudrait donc, modifier les Servlets, les tester et les mettre en exploitation.

Mais, effectuer la maintenance de Servlets n'est pas aisée. C'est pourquoi, la décision a été prise des les remplacer par des scripts PHP qui sont plus facilement maintenables (création, modification, tests ...).

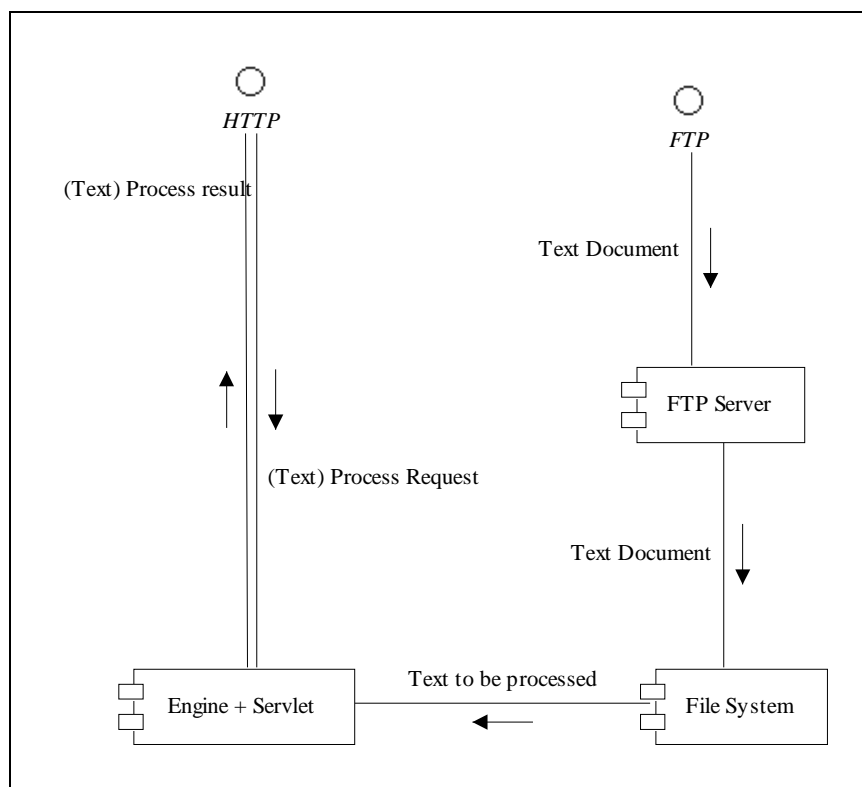


Figure 1 : Diagramme de déploiement de la version « Servlet »

3.2 PORPHYRE 2003 : ARCHITECTURE ET CHOIX TECHNOLOGIQUES

3.2.1 ARCHITECTURE

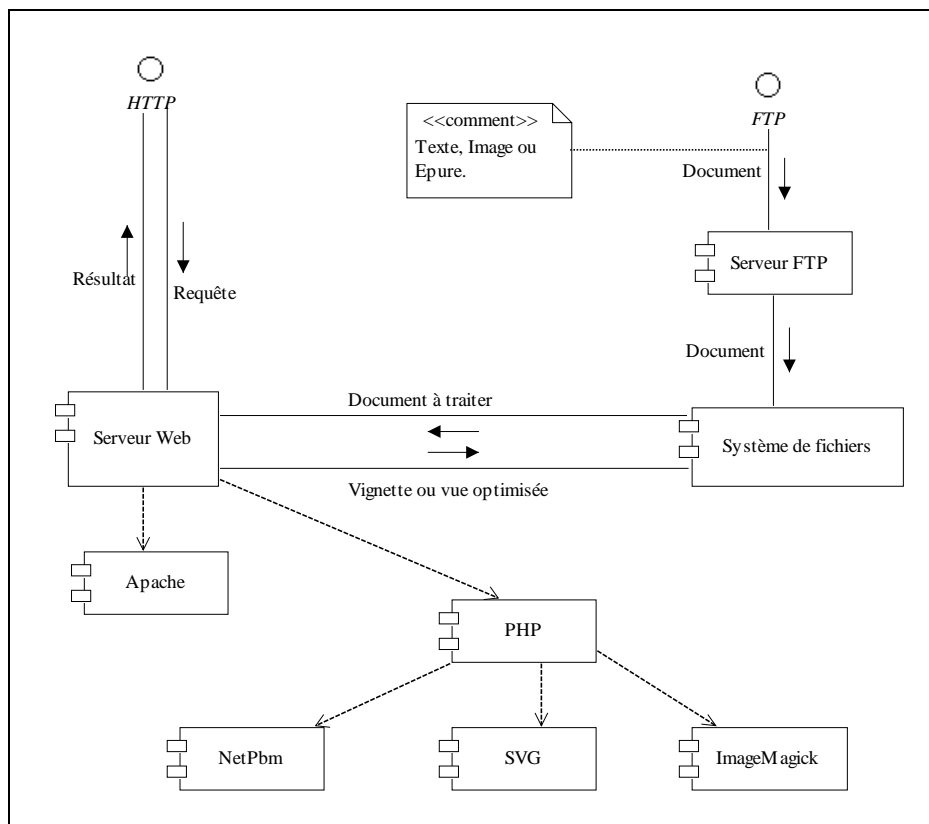


Figure 2 : Diagramme de déploiement de la version « PHP »

3.2.2 CHOIX TECHNOLOGIQUES : RAPPEL DES EXIGENCES

Les exigences du projet Porphyre sont les suivantes. Le système doit être :

- en « Open Source » ;
- portable;
- optimisé;
- modulaire;
- fiable.

3.2.3 POURQUOI PHP ?

L'optimisation et l'extension des traitements nécessitent des modifications importantes des Servlets. La maintenance (modification, tests, intégration) de Servlets n'est pas aisée. C'est pourquoi elles ont été remplacées au profit de PHP. Ceci afin de faciliter également les évolutions futures.

PHP est multi-plateforme : PHP fonctionne aussi bien sur UNIX, Linux, Windows 9x et NT, que sur Macintosh.

PHP fonctionne avec n'importe quel serveur Web : Même si son serveur Web préféré reste APACHE, en effet il permet d'exploiter certaines caractéristiques propres à ce dernier, il peut fonctionner avec n'importe quel autre serveur Web, y compris Internet Information Server de Microsoft.

PHP est portable : un script PHP pourra être porté sans aucune modification vers n'importe quelle plateforme Web ou OS.

PHP est ouvert : PHP dispose de nombreuses APIs lui permettant une connectivité vers les protocoles et les middlewares les plus prometteurs comme les plus anciens. Les APIs PHP sont codées en C. Il est donc possible de développer ses propres APIs et de les exploiter à partir de script PHP.

PHP, un langage facile : PHP est un langage de programmation structurée comme un autre, plutôt proche du C et de PERL. La prise en main pour des développeurs confirmés est très rapide, et débiter avec PHP n'est pas une aberration puisqu'il offrira une ouverture vers n'importe quel langage de programmation structuré proche du C (comme Java par exemple).

Le support de PHP est gratuit : Les questions adressées à la mailing list ne reste pas longtemps sans réponse. On peut également trouver sur le site Web de PHP, la liste de tous les bugs déclarés et l'évolution de leurs résolutions.

3.2.4 POURQUOI NETPBM ?

Il était nécessaire de remplacer les algorithmes « maison » de traitements des images par une librairie afin de permettre :

- l'optimisation des traitements et
- la prise en charge de différents formats d'images (et non pas seulement le JPEG).

Afin de réduire ces temps de traitements, le choix s'est porté sur une librairie installée en standard avec Linux Mandrake : NetPbm peut également fonctionner dans des environnements Windows.

NetPbm est **gratuit** et en « **Open Source** ».

Cette boîte à outils (anciennement PbmPlus) permet la conversion entre différents formats graphiques. Elle n'offre cependant pas d'outil de visualisation, mais cela n'est pas nécessaire dans le sens où, une fois traitées, les images sont affichées dans un butineur Web. Cette librairie offre une multitude de petites fonctions indépendantes et performantes qui peuvent être enchaînées.

Le traitement d'un document de type « Image » (Figure 3) peut être résumé de la manière suivante :

- Conversion de l'image dans un format spécifique NetPbm. Le format obtenu diffère en fonction des caractéristiques de l'image :
 - *pbm** : portable bitmap pour les images de départ en noir et blanc ;
 - *pgm** : portable graymap pour les images de départ à nuance de gris et
 - *ppm** : portable pixmap pour les images de départ en couleur).
- Traitements de l'image (extraction fragment, re-dimensionnement de l'image, encadrement du fragment ...) : certaines commandes ne sont disponibles que pour un format d'image NetPbm particulier (*pbm*, *pgm* ou *ppm*) alors que d'autres permettent des manipulations indépendamment des 3 formats internes à NetPbm (*pnm** : portable anymap).

- Conversion de l'image traitée au format souhaité pour le résultat, donc en une image au format JPEG dans le cas du serveur de contenu de Porphyre.

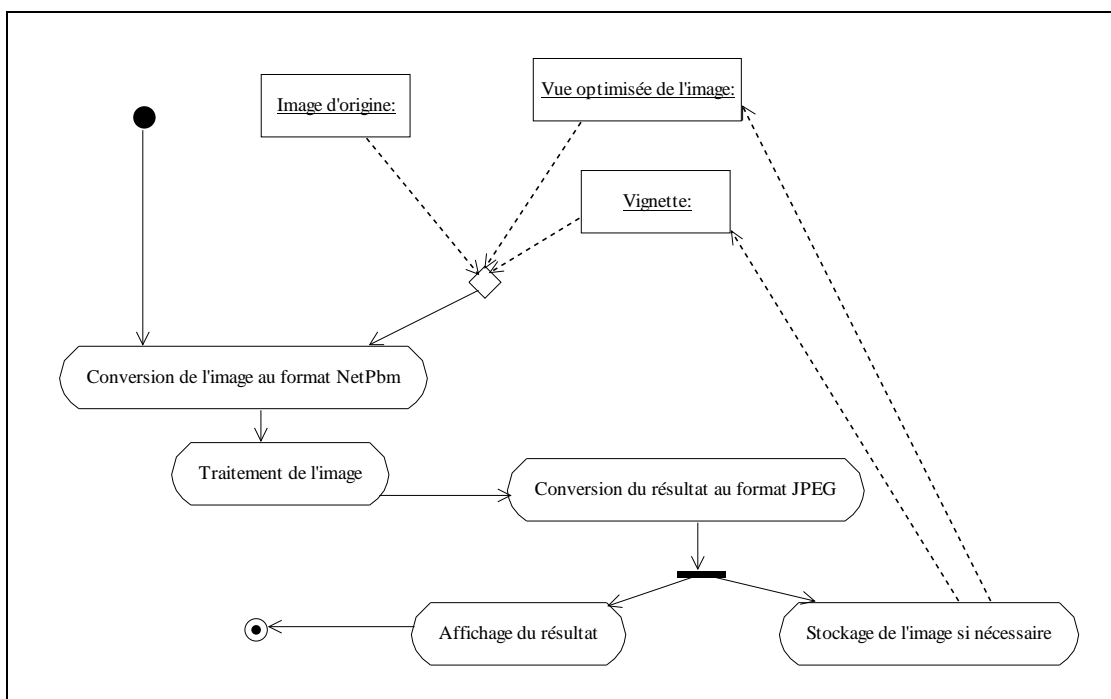


Figure 3 : Diagramme d'activité – traitements d'un document de type « Image »

3.2.5 POURQUOI IMAGEMAGICK ?

La bibliothèque ImageMagick a été étudiée en 2002 dans l'objectif de remplacer les Servlets et les algorithmes « maison » de traitement des images par des scripts PHP utilisant la librairie ImageMagick. Cependant, ce travail n'a pas été intégré au prototype Porphyre, car la bibliothèque ImageMagick ne permettait pas de résoudre les problèmes de temps de réponse. C'est pourquoi les Servlets ont été conservés dans le prototype Porphyre 2002. L'équipe projet à rechercher une solution optimum. En effet, l'étendu des fonctionnalités proposées par ImageMagick™ est vaste, mais Porphyre ne nécessite qu'un nombre limité de fonctions, mais qui doivent être performantes. En effet, la notion de temps d'attente est de manière générale importante, mais elle devient prépondérante lorsque l'utilisateur attend l'affichage d'une image dans son butineur.

Cette librairie est, néanmoins, présente dans l'architecture du serveur de contenu Porphyre 2003 car elle permet une conversion de documents au format « SVG » (de type « Épure ») en documents au format « JPEG ».

ImageMagick™ est une collection d'outils et de bibliothèques permettant de traiter des images dans différents formats (plus de 87 formats) comprenant des formats populaires : TIFF, JPEG, PNG, PDF, PhotoCD, GIF... ImageMagick™ permet l'exécution des traitements d'image par commande en ligne. ImageMagick est également **gratuite** et en « **Open Source** ». De plus, ses commandes peuvent être utilisées dans différents environnements, même si actuellement le choix a été fait d'utiliser un environnement Linux pour le serveur de contenu.

4. LES TRAITEMENTS A LA DEMANDE

4.1 DOCUMENTS DE TYPE « TEXTE »

Les traitements appliqués aux documents de type « Texte »

Les traitements à appliquer sur un document de type « Texte » sont :

- le soulignement de fragment(s) dans le texte d'origine
- l'extraction d'un fragment de texte à partir du fichier texte d'origine.

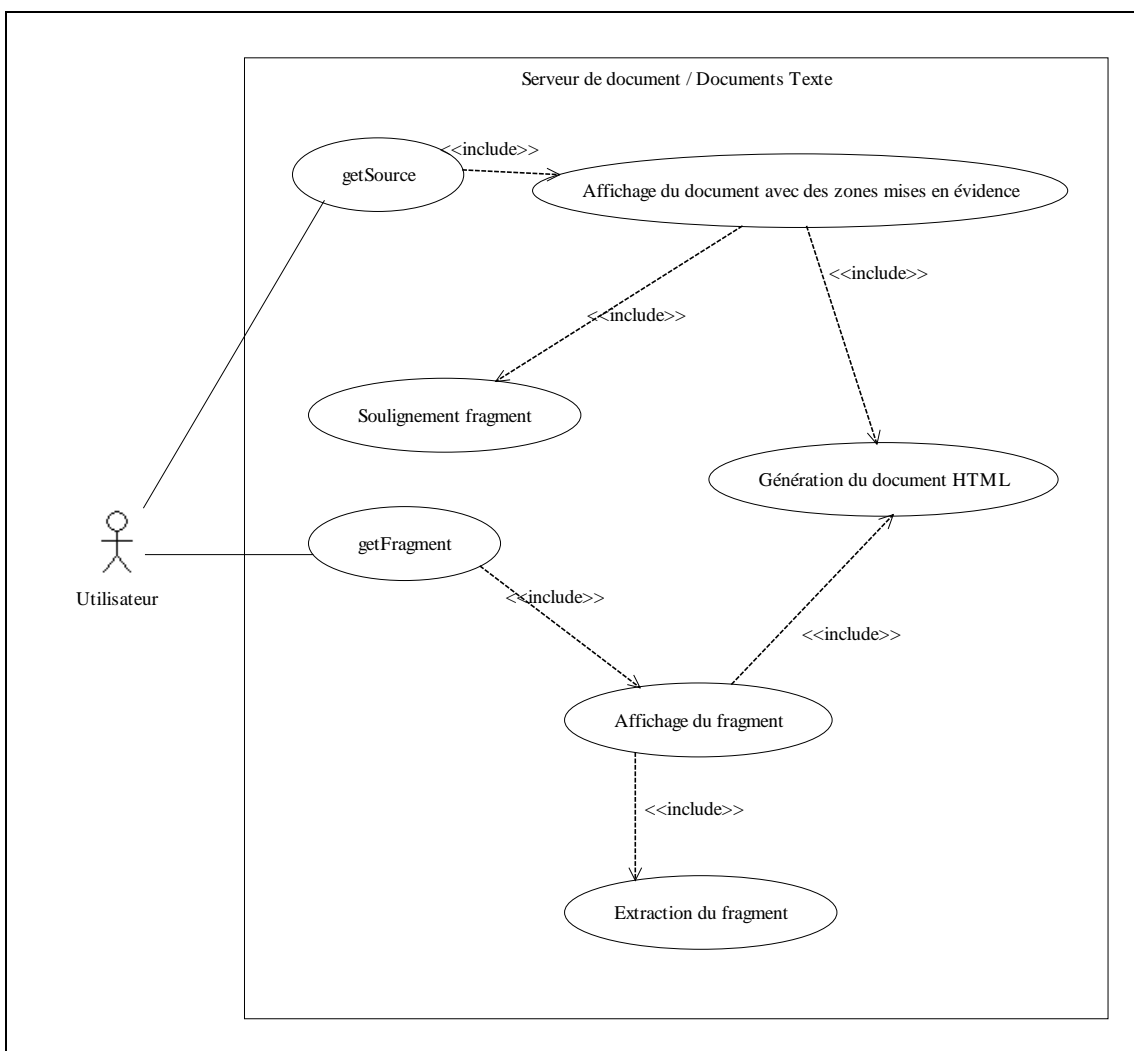


Figure 4 : Cas d'utilisation (traitement de documents de type « Texte »)

4.1.1 TEXT/GETFRAGMENT.PHP : EXTRACTION D'UN FRAGMENT

<http://localhost/Texte/getFragment?file=F&coord=X+Y>

(ex : <http://localhost/Texte/getFragment?file=1994&coord=50+300> permet d'extraire le texte situé entre le 50^{ème} et le 300^{ème} caractère à partir du fichier 1994.txt).

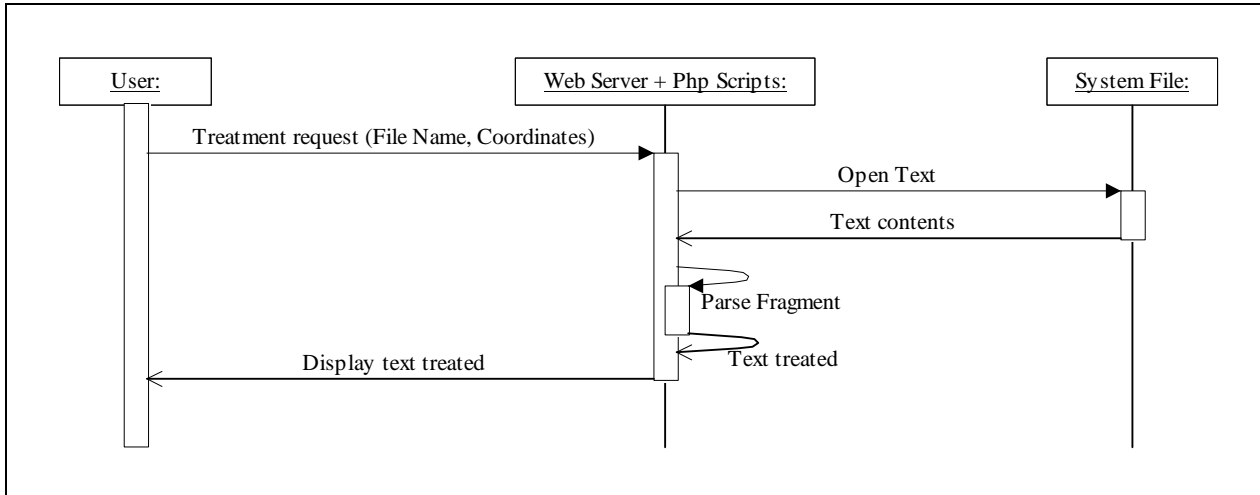


Figure 5 : Diagramme de séquence (extraction d'un fragment à partir d'un texte)

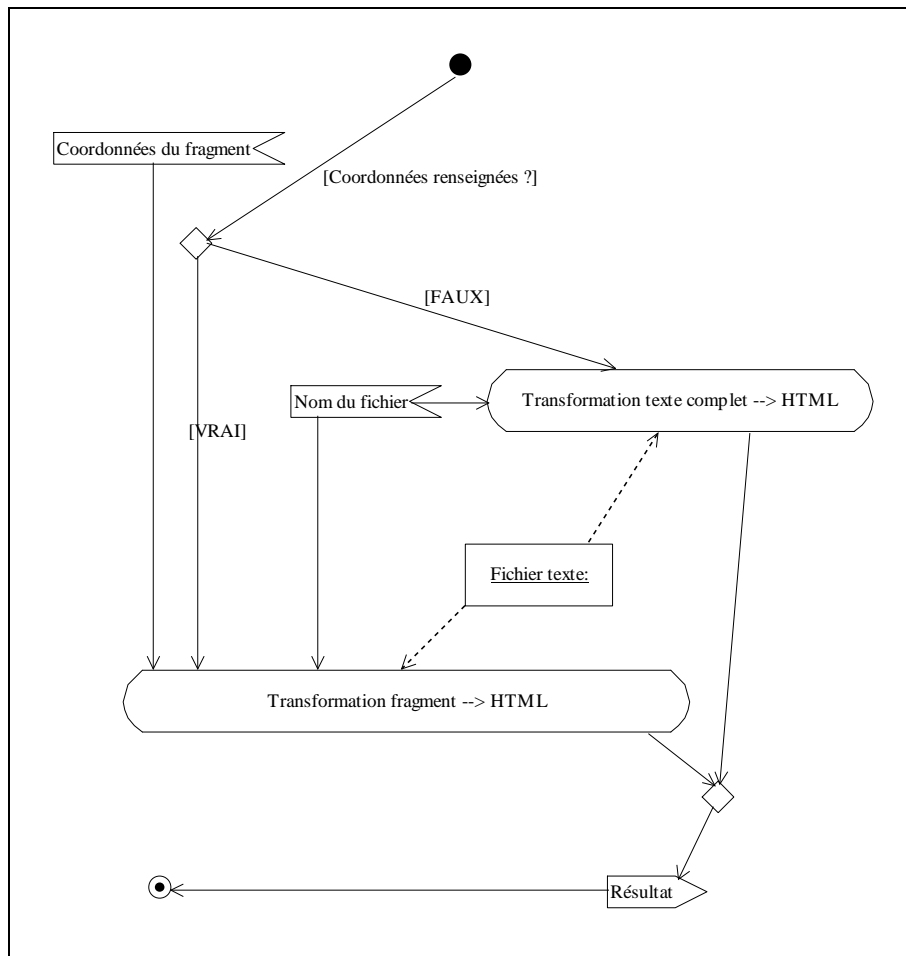


Figure 6 : Diagramme d'activité (extraction d'un fragment à partir d'un texte)

4.1.2 TEXT/GETSOURCE.PHP : SOULIGNEMENT DE FRAGMENT(S)

<http://localhost/Texte/getSource?file=F&coord=X1+Y1;X2+Y2>

(ex : <http://localhost/Texte/getFragment?file=1994&coord=50+300;10+20;400+500> permet d'extraire l'ensemble du fichier 1994.txt et de souligner 3 zones situées entre les 10^{ème} et 20^{ème}, les 50^{ème} et 300^{ème}, et les 400^{ème} et 500^{ème} caractères).

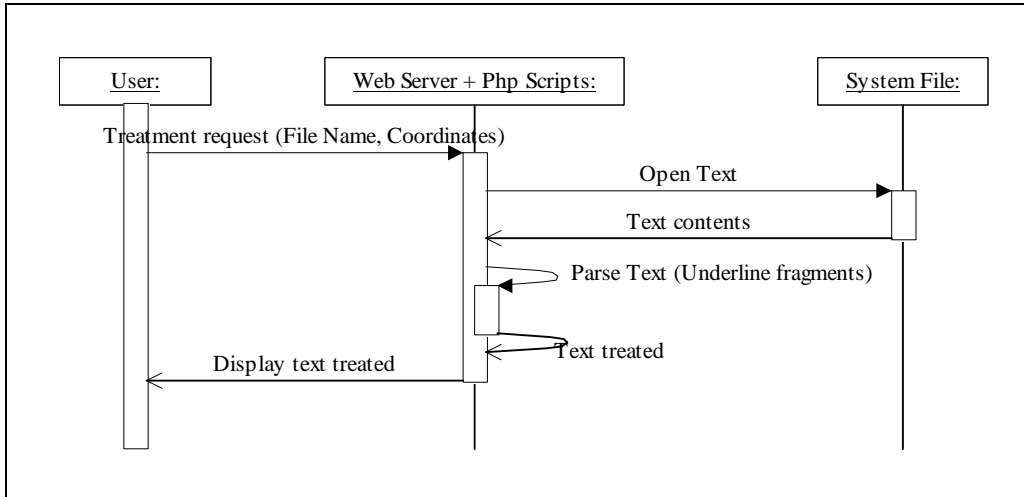


Figure 7 : Diagramme de séquence (soulignement de fragments dans un texte)

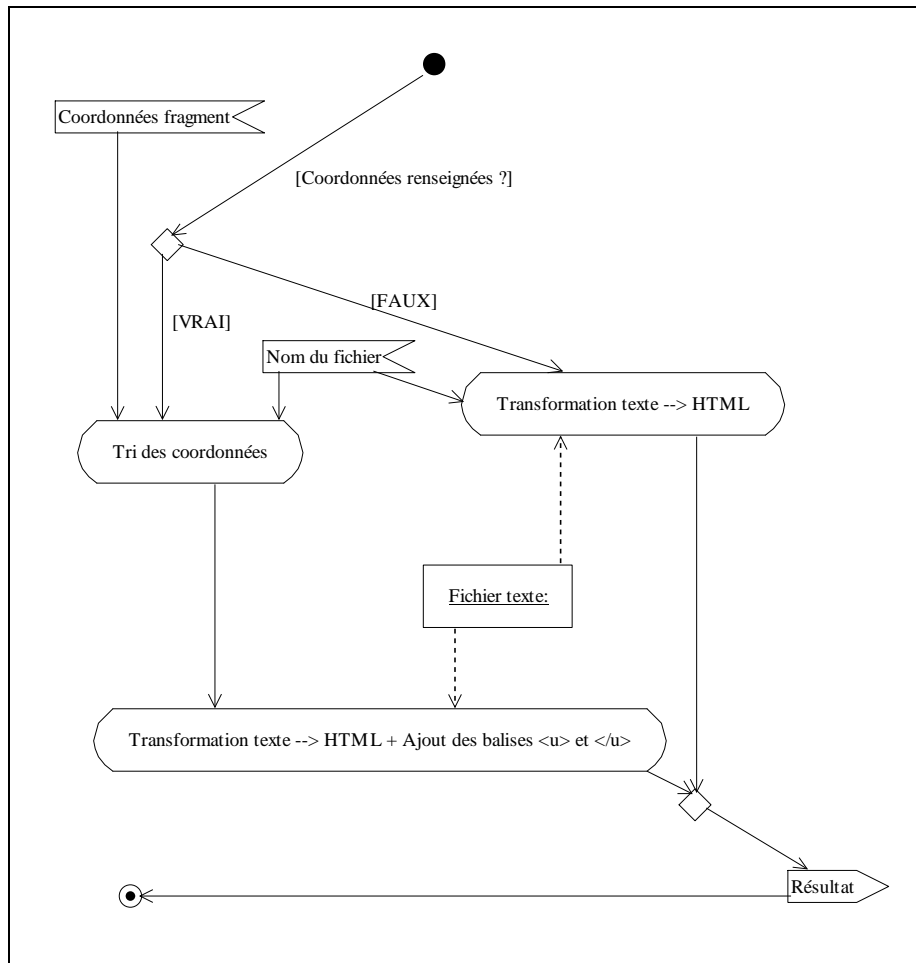
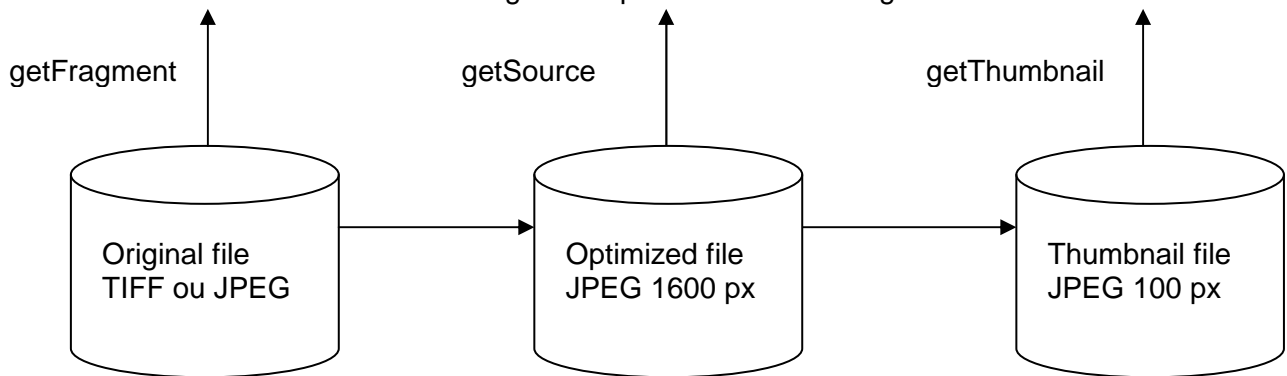


Figure 8 : Diagramme d'activité général (soulignement de fragments dans un texte)

4.2 DOCUMENTS DE TYPE « IMAGE »

Les traitements à appliquer sur un document de type « Image » sont :

- la génération de la vignette
- l'encadrement de fragment(s) à partir d'une vue optimisée
- l'extraction d'un fragment à partir du fichier d'origine



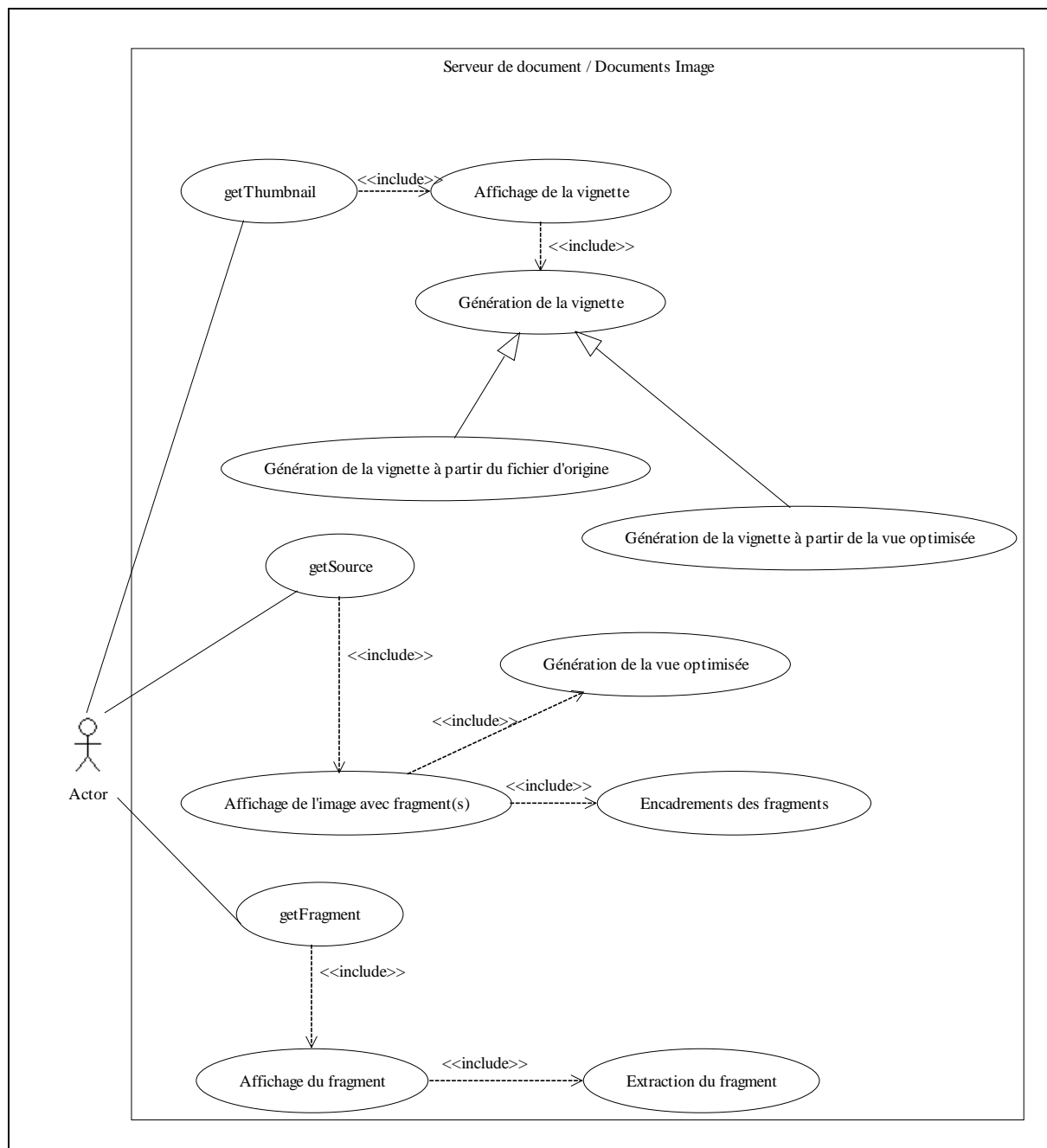


Figure 9 : Cas d'utilisation (traitement de documents de type « Image »)

4.2.1 TESTS DE PERFORMANCE

Pour chaque fonctionnalité, des tests de performance ont été effectués sous Linux.

- Tests avec la Servlet : seuls des fichiers originaux au format JPEG sont traités par la Servlet.

- Tests avec la librairie ImageMagick : Ils ont mis en évidence des temps de traitement variables en fonction de la taille de l'image traitée. En effet, les temps d'exécution sont acceptables lorsque la manipulation porte sur des images de taille raisonnable. Les images que l'on trouve dans CEFAEL sont des compressées avec l'algorithme CCITT groupe 4. Les résultats des traitements de ces images étaient catastrophiques. Il fallait en effet plus de 90 secondes pour seulement convertir le fichier tiff en png. C'est pourquoi ils ne figurent pas sur les graphiques.
- Tests avec la librairie NetPbm : les résultats de ces tests ne prenaient en compte que les temps de traitement. Ils ont donc été multiplié par 2 pour obtenir des temps de réponse effectifs réalistes, le processeur n'étant pas affecté à 100 % au traitement des tâches.

4.2.2 IMAGE/GETFRAGMENT.PHP : EXTRACTION D'UN FRAGMENT

<http://localhost/Image/getFragment?file=F&coord=X+Y+W+H&max=M>

(ex :

http://localhost/Image/getFragment?file=BCH_100_1_d1&coord=50+100+500+600&max=800 permet d'extraire une zone d'une largeur de 500 pixels et d'une hauteur de 600 pixels dont le coin haut-gauche se trouve au pixel d'abscisse 50 et d'ordonnée 100. Le résultat sera redimensionné de sorte que sa hauteur soit de 800 pixels

http://localhost/Image/getFragment?file=BCH_100_1_d1&coord=50+100+500+600 permet d'extraire une zone d'une largeur de 500 pixels et d'une hauteur de 600 pixels dont le coin haut-gauche se trouve au pixel d'abscisse 50 et d'ordonnée 100. Dans ce cas, le résultat ne sera pas re-dimensionné).

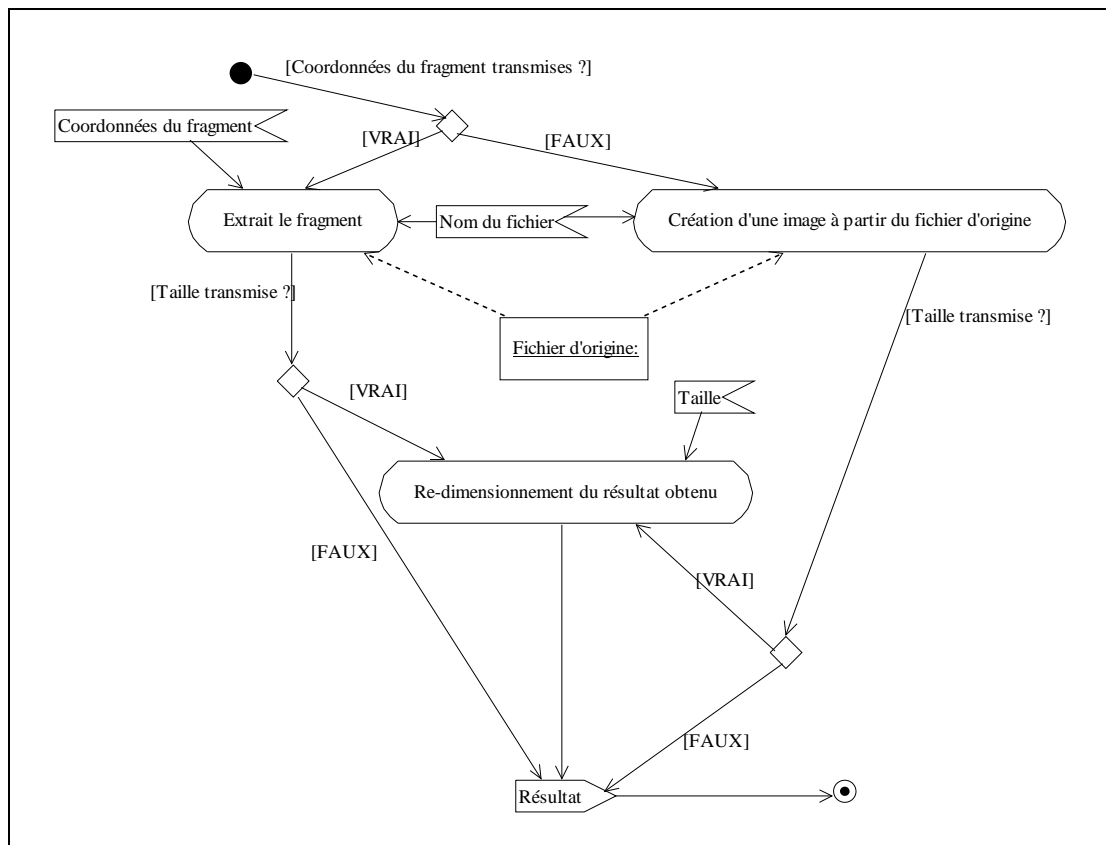


Figure 10 : Diagramme d'activité détaillé (extraction de fragment à partir d'une image)

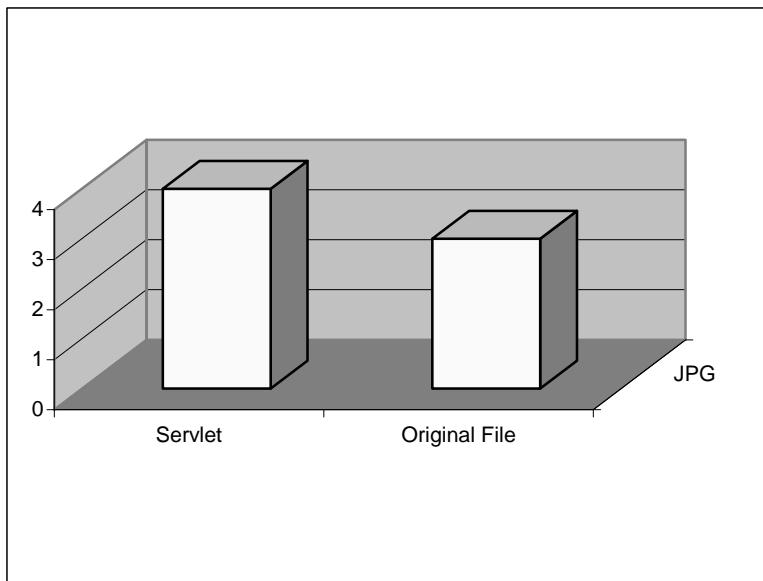


Figure 11 : Tests de performance comparatifs sur des images naturelles (extraction de fragment à partir d'une image)

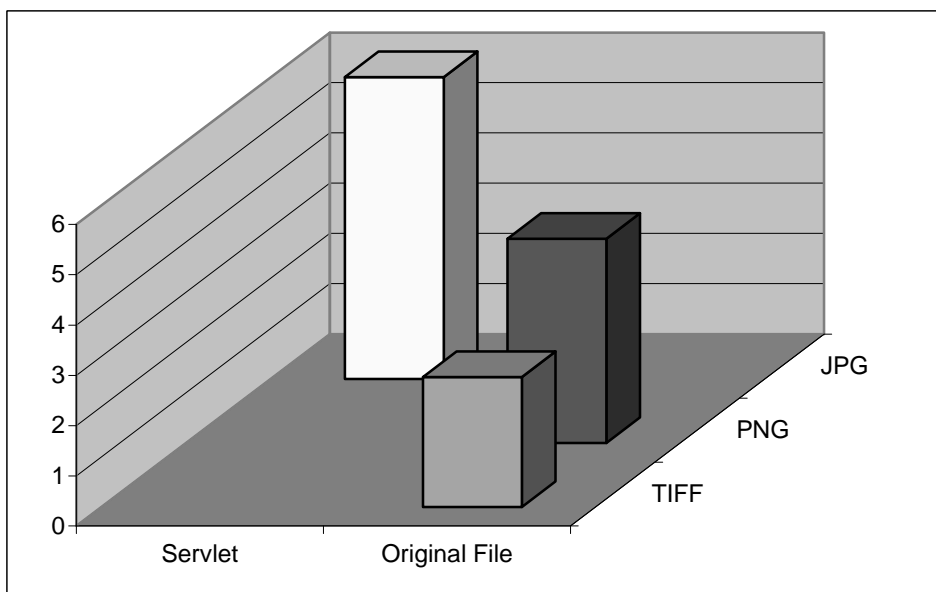


Figure 12 : Tests de performance comparatifs sur des images artificielles (extraction de fragment à partir d'une image)

4.2.3 IMAGE/GETSOURCE.PHP : ENCADREMENT DE ZONE(S)

<http://localhost/Image/getSource?file=F&coord=X1+Y1+W1+H1;X2+Y2+W2+H2&max=M>

(ex :
http://localhost/Image/getFragment?file=BCH_100_1_d1&coord=50+100+500+600;600+750+700+800&max=800 permet d'encadrer deux zones dans l'image : la première aura une largeur de 500 pixels et d'une hauteur de 600 pixels et le coin haut-gauche se trouve au pixel d'abscisse 50 et d'ordonnée 100 ; la seconde aura une largeur de 700 pixels et une hauteur de 800 pixels et débutera au pixel d'abscisse 600 et d'ordonnée 750. Le résultat sera redimensionné de sorte que sa hauteur soit de 800 pixels
http://localhost/Image/getFragment?file=BCH_100_1_d1&50+100+500+600;600+750+700+800 permet d'encadrer deux zones dans l'image : la première aura une largeur de 500 pixels et d'une hauteur de 600 pixels et le coin haut-gauche se trouve au pixel d'abscisse 50 et d'ordonnée 100 ; la seconde aura une largeur de 700 pixels et une hauteur de 800 pixels et débutera au pixel d'abscisse 600 et d'ordonnée 750.. Dans ce cas, le résultat ne sera pas re-dimensionné).

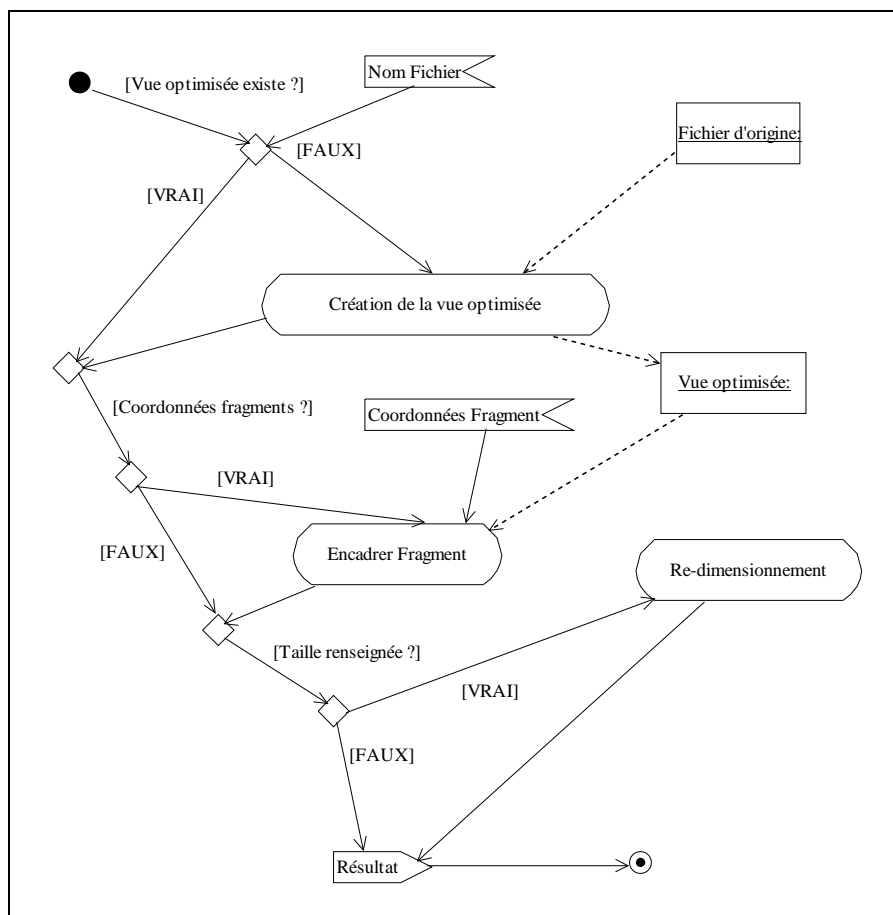


Figure 13 : Diagramme d'activité (encadrement de fragments dans une image)

4.2.3.1 TESTS DE PERFORMANCE (GETSOURCE)

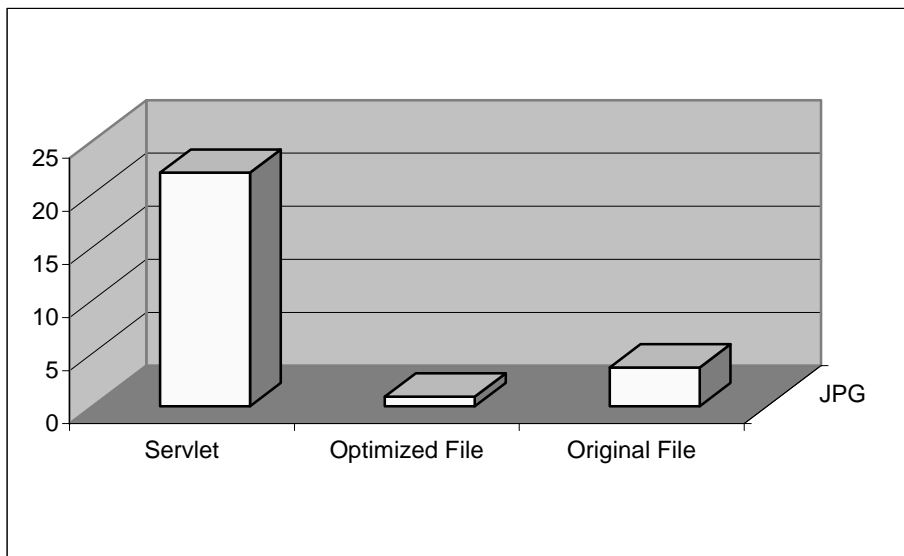


Figure 14 : Tests de performance sur des images naturelles (encadrement de fragments dans une image)

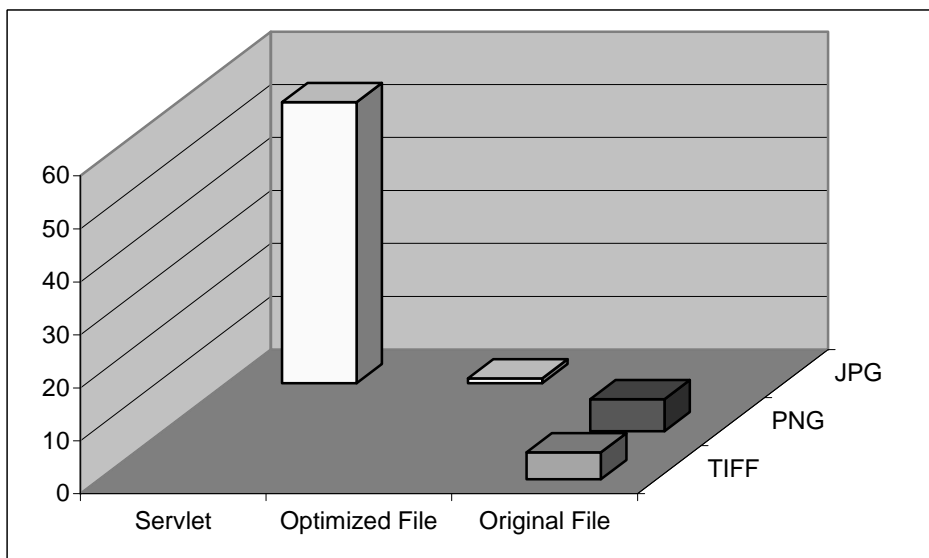


Figure 15 : Tests de performance sur des images artificielles (encadrement de fragments dans une image)

4.2.4 IMAGE/GETTHUMBNAIL.PHP : GENERATION DE VIGNETTE

<http://localhost/Image/getThumbnail?file=F>

Les vignettes générées ont une longueur standard de 100 pixels.

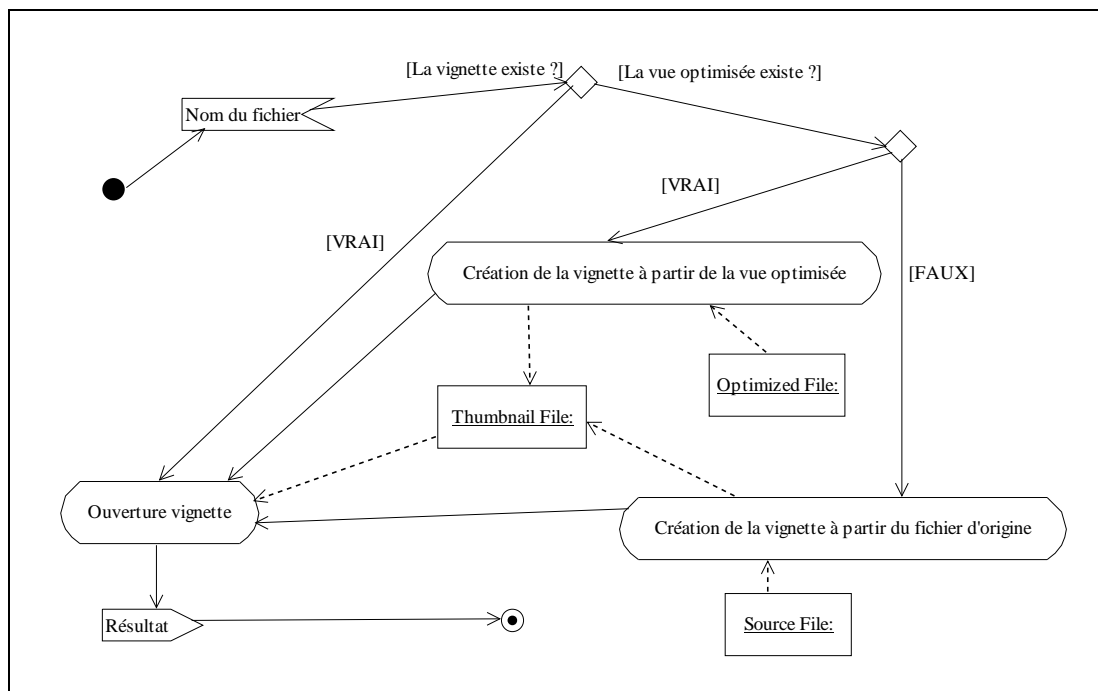


Figure 16 : Diagramme d'activité général (génération d'une vignette à partir d'une image)

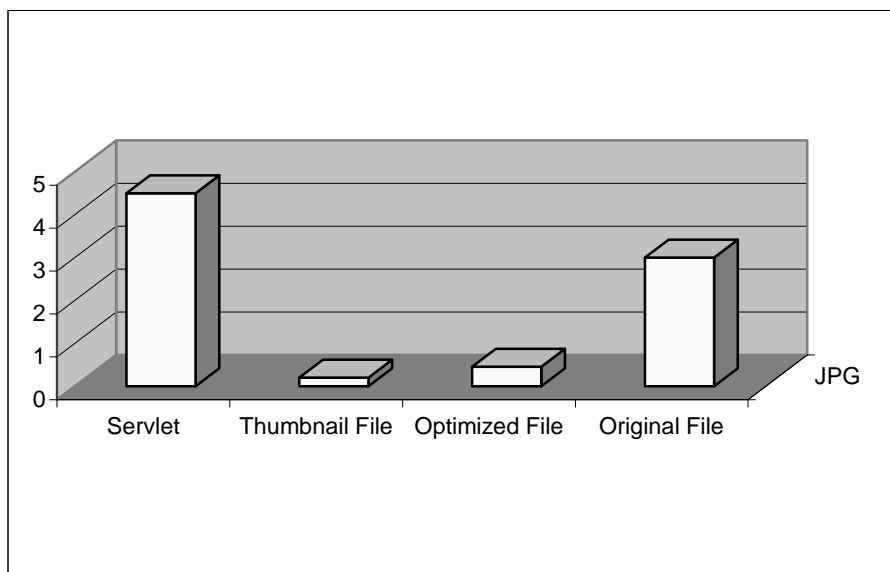


Figure 17 : Tests de performance sur des images naturelles (génération d'une vignette à partir d'une image)

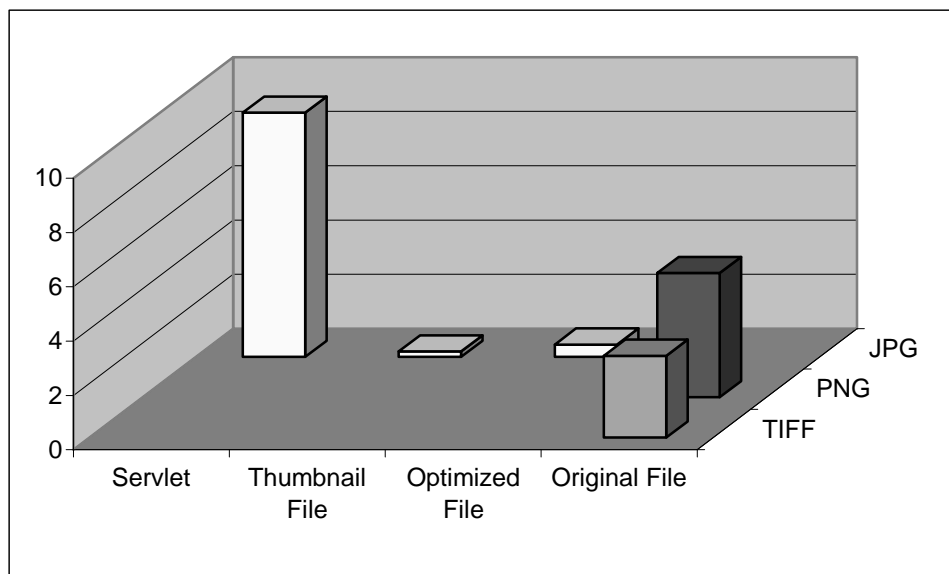


Figure 18 : Tests de performance sur des images artificielles (génération d'une vignette à partir d'une image)

4.3 DOCUMENTS DE TYPE « EPURE »

Les traitements à appliquer sur un document de type « Image » sont :

- la génération de la vignette
- l'encadrement de fragment(s) à partir d'une vue optimisée
- l'extraction d'un fragment à partir du fichier d'origine.

A l'heure actuelle, tous les navigateurs ne prennent pas en charge le format SVG (ou nécessitent l'installation d'un plug-in)> C'est la raison pour laquelle le fichier SVG résultat doit être converti en une image au format JPEG.

Cette transformation est effectuée par l'utilisation de la librairie Image Magick.

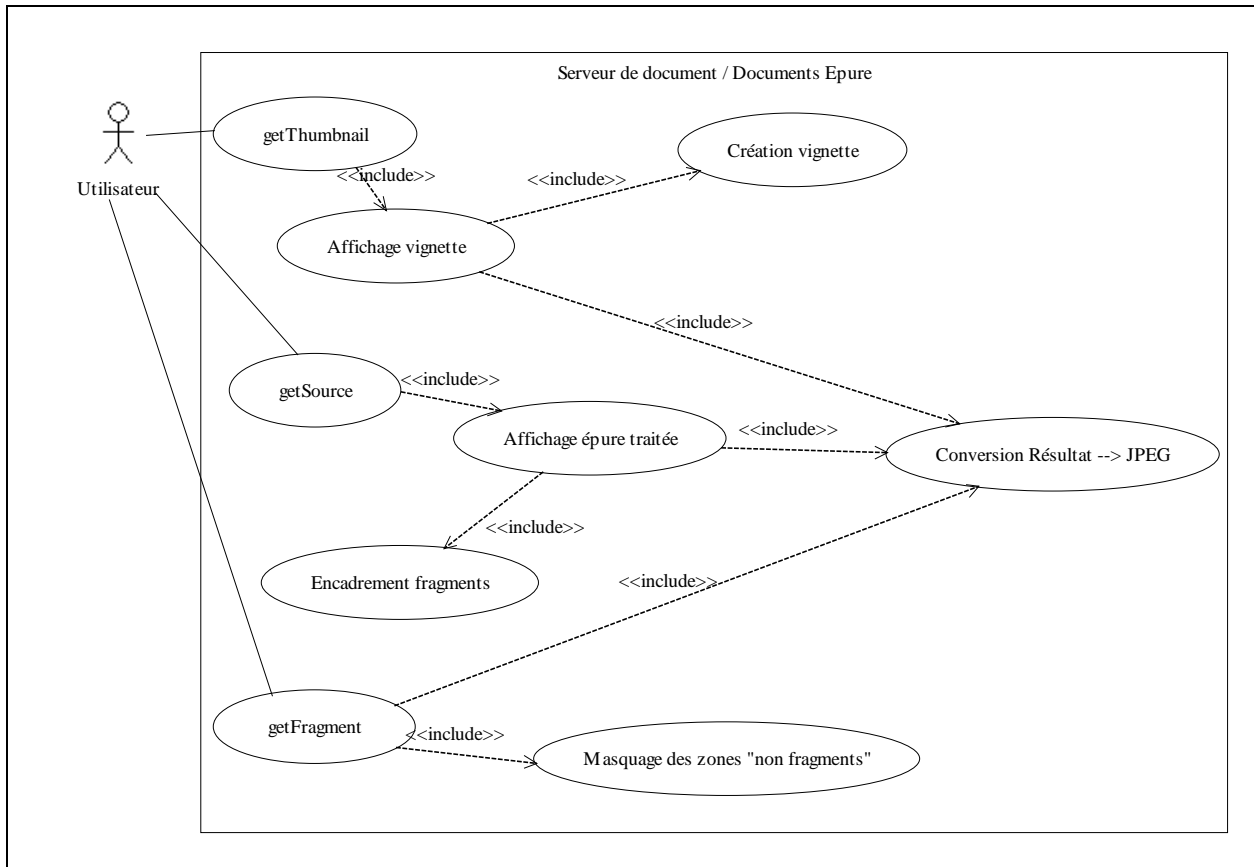


Figure 19 : Cas d'utilisation (traitement de documents de type « Epure »)

4.3.1 EPURE/GETFRAGMENT.PHP : EXTRACTION D'UN FRAGMENT

Dans le cas d'un document de type « Epure », l'extraction d'un fragment consiste en fait à masquer les zones indésirables du document. C'est la raison pour laquelle il est possible de laisser apparents plusieurs zones simultanément.

<http://localhost/Epure/getFragment.php?file=F&ext=E&MaxScale=M&coord=A1+B1,A2+B2,An+Bn&circle=C1+D1+E1+F1;Cn+Dn+En+Fn&points=Xa1+Ya1,Xb1+Yb1,Xn1+Yn1;Xa2+Ya2,Xb2+Yb2,Xn2+Yn2>

<http://localhost/Epure/getFragment.php?file=799-800&ext=JPG&MaxScale=0.4&coord=400+400,600+400,600+600,400+600;800+900,1500+1500,1600+1600,1000+1800,100+600&circle=1500+1500+800+800;1000+500+300+300&points=0+0,300+0,300+300,0+300;0+1500,300+1500,300+1800,0+1800>

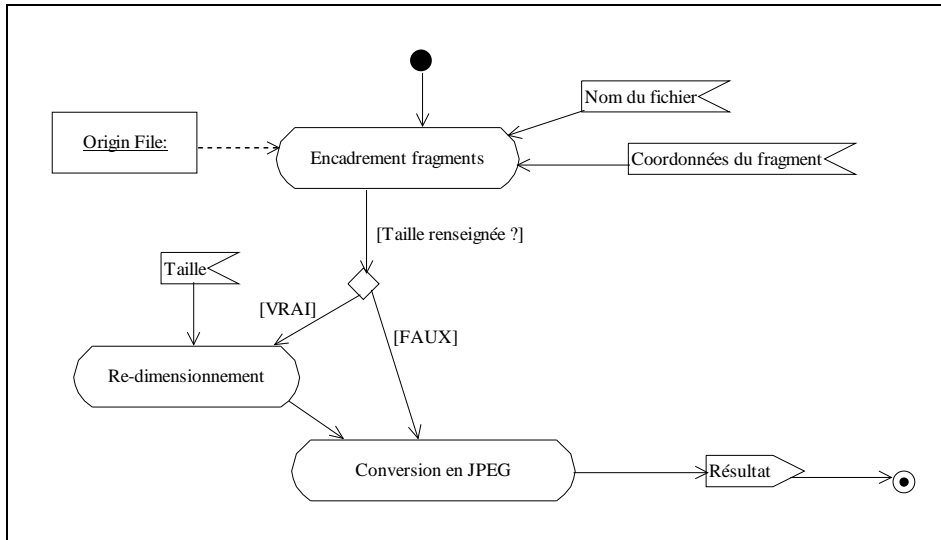


Figure 20 : Diagramme d'activité général (extraction d'un fragment à partir d'une épure)

4.3.2 EPURE/GETSOURCE.PHP : ENCADREMENT DE ZONE(S)

<http://localhost/Epure/getSource.php?file=F&ext=E&MaxScale=M&coord=A1+B1,A2+B2,An+Bn&circle=C1+D1+E1+F1;Cn+Dn+En+Fn&points=Xa1+Ya1,Xb1+Yb1,Xn1+Yn1;Xa2+Ya2,Xb2+Yb2,Xn2+Yn2>

<http://localhost/Epure/getSource.php?file=799-800&ext=JPG&MaxScale=0.4&coord=400+400,600+400,600+600,400+600;800+900,1500+1500,1600+1600,1000+1800,100+600&circle=1500+1500+800+800;1000+500+300+300&points=0+0,300+0,300+300,0+300;0+1500,300+1500,300+1800,0+1800>

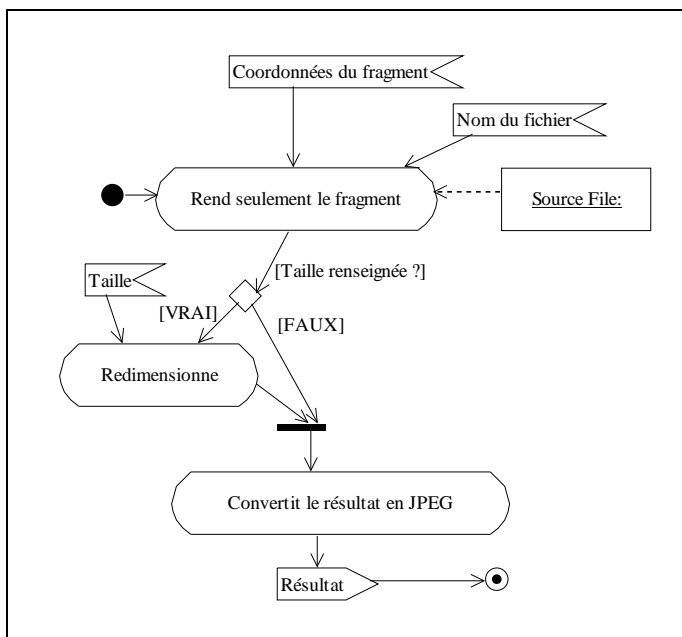


Figure 21 : Diagramme d'activité général (encadrement de zone(s) à partir d'une épure)

4.3.3 EPURE/GETTHUMBNAIL.PHP : GENERATION DE VIGNETTE

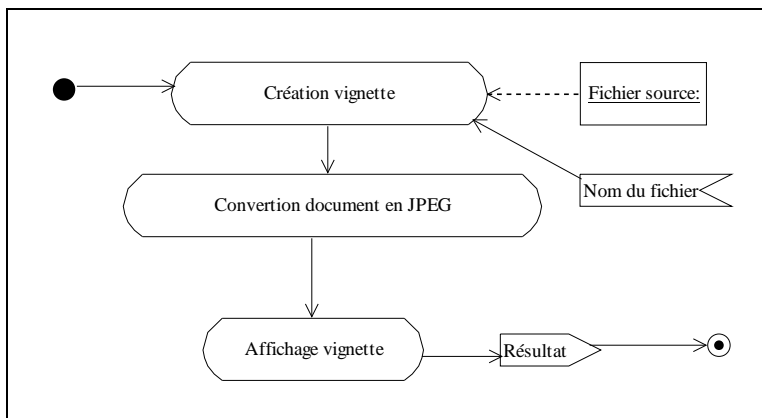


Figure 22 : Diagramme d'activité général (génération d'une vignette à partir d'une épure)

4.3.4 LE STANDARD SVG

4.3.4.1 DESCRIPTION

XML (Extensible Markup Language ou Langage Extensible de Balisage) est le langage destiné à succéder à HTML sur le World Wide Web. Comme HTML (Hypertext Markup Language), il s'agit d'un langage de balisage, c'est-à-dire un langage qui présente de l'information encadrée par des balises. Mais contrairement à HTML, qui présente un jeu limité de balises orientées présentation (titre, paragraphe, image, lien hypertexte, etc.), XML est un métalangage, qui va permettre d'inventer à volonté de nouvelles balises pour isoler toutes les informations élémentaires (titre d'ouvrage, prix d'article, numéro de sécurité sociale, référence de pièce...), ou agrégats d'informations élémentaires, que peut contenir une page Web. Cet environnement est pour la plus grande part encore en cours d'élaboration par le W3C (World Wide Web Consortium). Il peut donc encore subir des modifications.

SVG est un langage de description de graphiques bi-dimensionnels en XML. Il est recommandé par le W3C. La spécification W3C en définit les fonctions et la syntaxe. SVG admet trois types d'objets graphiques : des contours graphiques vectoriels (par exemple, des tracés consistant en lignes droites et courbes), des images et du texte. Les objets graphiques peuvent être regroupés, stylés, transformés et composés dans des objets précédemment rendus. L'ensemble de fonctions comprend des transformations imbriquées, des tracés de rognage, des masques basés sur la couche alpha et des objets de gabarit.

Ce format permet de faire tout ce que Flash proposait dans le domaine du vectoriel mais il ne s'agit en aucun cas d'un format propriétaire. Il est donc ouvert et accessible à tous (comme le HTML). Comme il se base sur XML, il est donc extensible. Les dessins SVG peuvent être interactifs et dynamiques. On peut définir et déclencher des animations, soit déclarativement (en incorporant les éléments d'animation SVG dans un contenu SVG), soit via un script.

4.3.4.2 ÉLÉMENTS UTILISÉS

Il est possible d'encadrer ou « extraire » plusieurs fragments en utilisant les éléments suivants :

- ellipse

L'élément 'ellipse' définit une ellipse, alignée sur les axes du système de coordonnées utilisateur et basée sur un point central et deux rayons.

- polygon

L'élément 'polygon' définit une forme fermée consistant en un jeu de segments de droite reliés.

- path

On définit un tracé avec un élément '**path**' ayant un attribut **d = "<données-de-tracé>"**.

Actuellement seules les commandes suivantes sont utilisées, mais d'autres sont disponibles, s'il s'avère nécessaire de faire évoluer cette fonctionnalité :

Commande	Nom	Paramètres	Description
Z ou z	closepath	(aucun)	Achève le sous-tracé courant par le dessin d'une ligne droite, du point courant jusqu'au point initial du sous-tracé courant.
L (absolue) l (relative)	lineto	(x y)+	Dessine une ligne du point courant jusqu'au point de coordonnées (x,y) donné, qui devient le nouveau point courant. Un L (majuscule) annonce que des coordonnées absolues vont suivre, un l (minuscule) des coordonnées relatives. On peut spécifier un certain nombre de paires de coordonnées pour le dessin d'un polygone. Une fois la commande exécutée, le nouveau point courant s'établit au dernier jeu de coordonnées fourni.

Remarque : Quand un sous-tracé s'achève par une commande "closepath," son comportement diffère d'un sous-tracé qui est clos « manuellement », via une commande "lineto. Avec une commande "closepath", la fin du dernier sous-tracé est « jointe » avec le début du segment initial du sous-tracé. Par contre, si on ferme « manuellement » le sous-tracé via une commande "lineto", le début du premier segment et la fin du dernier segment ne sont pas joints, mais chacun se superposant. Une fois la commande exécutée, le nouveau point courant s'établit au point initial du sous-tracé courant.

5. CONCLUSION

Les principaux objectifs sont :

- l'amélioration des temps de réponse
- l'élargissement des fonctionnalités
- la possibilité de traiter différents formats de fichier
- une maintenance facilitée

Le stockage de la vue optimisée et de la vignette d'une image permet de diminuer fortement les temps d'exécution après le premier traitement appliqué à une image. De même, par rapport à la version « PHP / ImageMagick + GD », la version « PHP / NetPbm » permet des traitements des fichiers au format TIFF plus rapides.

La possibilité de mettre en évidence plusieurs zones dans une image est le principal élargissement des fonctionnalités.

De plus, la librairie d'outils NetPbm permet d'utiliser une grande diversité de format de stockage pour les fichiers originaux (JPG, TIF, PNG, photo CD ...). Il sera donc possible choisir le format de stockage des images indépendamment de la manière de les traiter. Dans la version « Servlet », seuls des fichiers au format JPEG étaient exploitables.

Les apports de la version « PHP » sont donc :

- plus grande modularité (fonctionnalités, tests, installation)
- plus grande capacité de traitement

Annexes

1. Syntaxes : Récapitulatif

1.1 Documents de type « Texte »

Paramètre	getFragment	GetSource	getThumbnail
file	nom du fichier à traiter sans extension		X
coord	<p>X+Y avec X représentant la position du premier caractère de la zone à extraire ou à encadrer et Y représentant la position du dernier caractère de la zone à extraire ou à encadrer.</p> <p><u>Nota</u> : Si les coordonnées ne sont pas mentionnées, le texte sera entièrement affiché, sans aucune partie mise en évidence.</p> <p><u>Compléments</u> : Il est possible de souligner plusieurs fragments dans un texte, il suffit de délimiter les différentes coordonnées avec le symbole ' ; '. Dans ce cas, l'ordre de saisie des coordonnées des différents fragments est indifférent. (par exemple, on obtiendra le même résultat en saisissant : coord=5+15;80+90 ou coord=80+90;5+15).</p>		

1.2 Documents de type « Image »

Paramètre	GetFragment	getSource	getThumbnail
file	nom du fichier à traiter avec extension		X
coord	<p>X+Y+W+H X et Y représentent les abscisse et ordonnée du point de départ du fragment à extraire ou à encadrer. W représente l'abscisse du point « bas-droit » de la zone. H représente l'ordonnée du point « bas-droit » de la zone.</p> <p><u>Nota</u> : Si les coordonnées ne sont pas mentionnées, l'image sera entièrement affichée, sans aucune zone mise en évidence.</p> <p><u>Compléments</u> : Il est possible d'encadrer plusieurs fragments dans une image, il suffit de délimiter les différentes coordonnées avec le symbole ' ; '. (par exemple : coord=50+75+400+300;80+90+300+600).</p>		

max	longueur ou largeur maximale d'affichage du résultat. <u>Nota</u> : Si cette information n'est pas précisée, le résultat du traitement ne sera pas re-dimensionné.	
-----	---	--

1.3 Documents de type « Épure »

Paramètre	getFragment	getSource	getThumbnail
file	nom du fichier à traiter avec extension		
MaxScale	Pourcentage d'affichage du document		
circle	X et Y représentent les abscisse et ordonnée du point central de l'ellipse) à extraire ou à encadrer. W représente le rayon sur l'axe-x de l'ellipse. H représente le rayon sur l'axe-y de l'ellipse.		
points	Permet de définir une forme fermée consistant en un jeu de segments de droite reliés (polygone). Chaque couple de coordonnées de point X+Y est délimité par le symbole ','.		
coord	Un tracé représente le contour d'une forme (path). Chaque couple de coordonnées de point X+Y est délimité par le symbole ','.		
type	'R' \mathbb{E} indique que les paramètres de coord sont à utiliser en relatifs pour effectuer le tracer. (Par exemple 40+70,50+100 : en coordonnées absolues, le curseur débute au point ayant pour abscisse 40 et pour ordonnée 70 et se déplace au point ayant pour abscisse 50 et ordonnée 100. En coordonnées relatives, le curseur débute au point ayant pour abscisse 40 et pour ordonnée 70 et se place au point situé à l'abscisse +50 et à l'ordonnée + 100 par rapport au point de départ. <u>Nota</u> : Si le type n'est pas renseigné, les coordonnées sont utilisées en absolu.		

2. Packages nécessaires

2.1 PHP

L'installation de PHP peut s'effectuer lors de l'installation de Mandrake 9.1.

2.2 php-ini-4.3.0-1mdk

L'installation (sous Mandrake) du package php-ini-4.3.0-1mdk est obligatoire. Si ce package n'est pas installé, le mode de sécurité de PHP (safe-mode) est activé, ce qui désactive certaines fonctionnalités nécessaires au traitement des images.

2.3 apache

L'installation d'Apache s'est effectuée lors de l'installation de Mandrake 9.1.

2.4 ImageMagick

L'installation d'ImageMagick est nécessaire au traitement des documents de type « Épure ». Il faut installer les packages suivants :

- o imageMagick-5.5.4.4-7mdk¹
- o libMagick5.5.4-devel-5.5.4.4-7mdk¹
- o php-imagick-4.3.1_0.9.4-1mdk

2.5 NetPbm

L'installation de la librairie NetPbm est nécessaire au traitement des documents de type « Image ». Il faut installer les packages suivants :

- o netpbm-9.24-4mdk¹
- o libnetpbm9-9.24-4mdk¹

3. Évolutions possibles

L'élément path permet d'effectuer des tracés plus variés et complexes (ligne horizontale, ligne verticale, courbe de Bézier cubique², courbe de Bézier quadratique³ ou arc elliptique⁴) grâce à l'utilisation des commandes suivantes :

Commande	Nom	Paramètres	Description
H (absolue) h (relative)	lineto horizontal	x+	Dessine une ligne horizontale du point courant (cpx, cpy) jusqu'au point (x, cpy). Un H (majuscule) annonce que des coordonnées absolues vont suivre, un h (minuscule) des coordonnées relatives. On peut fournir plusieurs valeurs x (bien que cela n'ait généralement aucun sens). Une fois la commande exécutée, le nouveau point courant devient (x, cpy) pour la valeur finale de x.

¹ Livré avec la distribution Mandrake 9.1.

² Un segment de Bézier cubique est défini par un point de départ, un point de fin et deux points de contrôle

³ Un segment de Bézier quadratique est défini par un point de départ, un point de fin et un point de contrôle

⁴ Un segment d'arc elliptique dessine un segment d'ellipse.

Commande	Nom	Paramètres	Description
V (absolue) v (relative)	lineto vertical	y+	Dessine une ligne verticale du point courant (cpx, cpy) jusqu'au point (cpx, y). Un V (majuscule) annonce que des coordonnées absolues vont suivre, un v (minuscule) des coordonnées relatives. On peut fournir plusieurs valeurs y (bien que cela n'ait généralement aucun sens). Une fois la commande exécutée, le nouveau point courant devient (cpx, y) pour la valeur finale de y.
C (absolue) c (relative)	curveto	(x1 y1 x2 y2 x y)+	Dessine une courbe de Bézier cubique du point courant jusqu'au point (x,y) en utilisant (x1,y1) comme point de contrôle en début de courbe et (x2,y2) en fin de courbe. Un C (majuscule) annonce que des coordonnées absolues vont suivre, un c (minuscule) des coordonnées relatives. On peut spécifier plusieurs jeux de coordonnées pour le dessin d'un polybézier. Une fois la commande exécutée, le nouveau point courant devient la dernière paire de coordonnées utilisée dans le polybézier.
S (absolue) s (relative)	curveto raccourci / lissé	(x2 y2 x y)+	Dessine une courbe de Bézier cubique du point courant jusqu'au point (x,y). Le premier point de contrôle est sensé être le reflet du second point de contrôle de la commande précédente par rapport au point courant. (S'il n'y a pas de commande précédente ou si celle-ci n'était pas une commande C, c, S ou s, le premier point de contrôle est supposé coïncider avec le point courant). Le point ((x2,y2) est le second point de contrôle (i.e., le point de contrôle en fin de courbe). Un S (majuscule) annonce que des coordonnées absolues vont suivre, un s (minuscule) des coordonnées relatives. On peut spécifier plusieurs jeux de coordonnées pour le dessin d'un polybézier. Une fois la commande exécutée, le nouveau point courant devient la paire de coordonnées (x,y) utilisée dans le polybézier.

Commande	Nom	Paramètres	Description
Q (absolue) q (relative)	curveto Bézier quadratique	(x1 y1 x y)+	Dessine une courbe de Bézier quadratique, du point courant jusqu'au point (x,y), en utilisant le point (x1,y1) comme point de contrôle. Un Q (majuscule) annonce que des coordonnées absolues vont suivre, un q (minuscule) des coordonnées relatives. On peut spécifier plusieurs jeux de coordonnées pour le dessin d'un polybézier. Une fois la commande exécutée, le nouveau point courant devient la paire de coordonnées finale (x,y) utilisée dans le polybézier.
T (absolue) t (relative)	curveto Bézier quadratique raccourci / lissé	(x y)+	Dessine une courbe de Bézier quadratique, du point courant jusqu'au point (x,y). Le point de contrôle est sensé être le reflet du point de contrôle de la commande précédente par rapport au point de contrôle. (S'il n'y a pas de commande précédente ou si celle-ci n'était pas une Q, q, T ou t, le point de contrôle est sensé coïncider avec le point courant). Un T (majuscule) annonce que des coordonnées absolues vont suivre, un t (minuscule) des coordonnées relatives. Une fois la commande exécutée, le nouveau point courant devient la paire de coordonnées finale (x,y) utilisée dans le polybézier.
A (absolue) a (relative)	arc elliptique	(rx ry rotation-axe- x drapeau- arc-large drapeau- balayage x y)+	Dessine un arc elliptique, du point courant jusqu'au point (x, y). La taille et l'orientation de l'ellipse sont définies par deux rayons (rx , ry) et une valeur de rotation sur l'axe-x rotation-axe-x , qui indique la rotation de l'ellipse dans son ensemble par rapport au système de coordonnées courant. Le centre de l'ellipse (cx , cy) est calculé automatiquement pour satisfaire aux contraintes imposées par les autres paramètres. Les valeurs drapeau-arc-large et drapeau-balayage contribuent à ce calcul automatique et aident à déterminer le dessin de l'arc.