

ÉCOLE FRANÇAISE D'ATHENES
SERVICE INFORMATIQUE

LA CHRONIQUE DES FOUILLES
ÉTUDE TECHNOLOGIQUE

Mai - Septembre 1999

La Chronique des fouilles : De la bibliothèque à l'Internet

Rapport de stage sur l'étude technologique

Title	Technological study
Author	Franck EYRAUD (Franck.Eyraud@bigfoot.com)
Project	The Excavations Chronicle : from Library to Internet
Organization	French School of Archaeology (Athens)
Created on	July 11, 1999
Modified on	October 17, 1999
Version	Primary draft
Language	French – standard

1 - PRESENTATION

1.1 - Contexte

Ce stage effectué à l'École Française d'Athènes s'inscrit dans le projet « Chronique des Fouilles ». On y cherche à rendre accessible les publications de ce périodique, édité par l'EFA, à une communauté de chercheurs, de la manière la plus pratique, accessible et adaptée possible. Lors d'études précédentes, il a été montré que les classifications et indexations traditionnelles de documents ne répondent pas suffisamment à ces attentes.

Une étude spécifique d'un nouveau type d'indexation et d'un « moteur de recherche » associé et adapté a donc été lancée, pour tenter de répondre à ces besoins exprimés. Le but du stage était de fournir un soutien technologique et une concrétisation pratique à l'étude théorique, menée par Aurélien Béné.

1.2 - Description du travail

Le travail a consisté en deux parties distinctes.

Dans un premier temps, une étude technologique a été entreprise sur les nouvelles technologies de description et représentation de documents, qui sont développées dans l'esprit d'une standardisation des informations documentaires partagées par les différents systèmes de part le monde. Les technologies qui ont retenu l'attention pour leur éventuelle utilisation dans le projet étaient :

- XML (eXtensible Markup Language), langage étendu qui offre une structure à la représentation de documents
- XSL (eXtensible Stylesheet Language), langage de transformation de structure de document
- RDF (Ressource Description Framework), langage de représentation de graphes

L'étude de ces technologies devait déterminer leur utilité éventuelle dans le projet, leur maturité effective et les outils existants qu'il serait possible d'utiliser pour les mettre en œuvre.

Dans un second temps, le développement d'un prototype de l'outil de recherche qui serait offert aux chercheurs devait être réalisé pour valider le modèle de l'étude théorique.

2 - ÉTUDE TECHNOLOGIQUE

2.1 - Démarche

Grâce à l'outil de recherche universel et puissant qu'est le « web », et plus généralement internet, il a été possible de trouver rapidement un grand nombre d'informations sur les technologies étudiées et sur les applications actuellement existantes ou en cours de développement.

Il est vite apparu que toutes ces technologies sont relativement jeunes et le plus souvent en rodage. Pour aucune d'elle n'a été trouvé d'application vraiment complète qui

utilisait pleinement leur possibilité. La plus grande majorité des exemples d'utilisation était surtout destinée à montrer et illustrer la puissance et les fonctionnalités de ces technologies.

2.2 - Résultats

2.2.1 - XML

2.2.1.1 - Principe

XML a été défini par le W3 Consortium (W3C). Son but avoué est de ne représenter d'un document électronique que sa structure logique. Un souci d'indépendance ressort de sa définition. Celle-ci ne comprend en effet qu'une syntaxe sommaire de structure de fichier, mais il est possible de définir soi-même une (ou plusieurs) syntaxe plus évoluée pour répondre à ses besoins propres. Une des aspects associés à ce standard est la possibilité de séparer le contenu d'un document de sa présentation. Historiquement, c'est une simplification du standard SGML utilisé comme standard reconnu dans l'industrie pour la gestion électronique des documents.

Le standard XML est une généralisation du standard HTML. En effet, il repose sur le même principe de Tags (balises), mais avec une plus grande rigueur dans leur construction, et aucune contrainte sur leur contenu.

Dans sa forme la plus simple, un tag est une suite :

Début de tag, contenu, Fin de tag
<tag>valeur</tag>

Le contenu peut être du texte brut, ou d'autres tags. Cela donne la structure d'arborescence au fichier, un tag appartenant toujours à un tag père. Il est donc interdit de faire se chevaucher les tags (on ferme toujours le dernier tag ouvert avant de fermer les autres)

On peut également ajouter des attributs à un tag :

<tag attribut1="valeur1" attribut2="valeur2">...</tag>

Dans la norme XML apparaît la notion d'« espace de nom » (namespace). En fait, chaque balise s'intègre dans un espace de nom qui définit une syntaxe particulière pour une utilisation particulière d'XML. Comme XML ne contient aucune syntaxe prédéfinie, il est possible d'en spécifier pour répondre à ses propres besoins. Ainsi, XSL et RDF sont des syntaxes s'appuyant sur la structure XML. Chaque balise doit donc déclarer à quelle syntaxe elle appartient, en prenant un préfixe (pre :) avant son nom. Ce préfixe est défini au début de la feuille XML et fait référence à une URI (Unique Ressource Identifier) qui représente la syntaxe utilisée.

2.2.1.2 - Utilisations

L'utilisation principale prévue pour le standard XML est la mise à disposition de documents par le web, en séparant le contenu de la présentation. L'auteur d'un document décide d'une structure descriptive de ses documents, écrit tous ses documents en XML en respectant sa structure, et propose une présentation en fonction de la structure qu'il a définie.

La définition de la présentation des documents se fait par l'intermédiaire d'une feuille de style, qui décrit comment transformer la structure logique du fichier XML de départ en une structure de fichier mis en page qui rend le document lisible à son destinataire prévu.

Une autre utilisation est la sérialisation de données. Il est en effet tout à fait envisageable d'enregistrer des données brutes au format XML. Cela a comme avantage de proposer un format standard, même si pas trivial, pour transmettre des données. Ces données peuvent ainsi facilement être lues par un lecteur (humain ou programme) qui ne sait pas a priori ce que représentent ces données. Et puis, en étendant l'utilisation précédente, on est à même de pouvoir facilement procéder à la présentation des données grâce à une feuille de style.

2.2.1.3 - Outils

XML ne représentant qu'une structure logique d'un document ou de données, il est inutile tel quel. Des outils (sous forme d'APIs, nommés « parser ») permettent de lire un fichier XML et d'en extraire la structure arborescente. En revanche, ils sont incapables d'interpréter son contenu, XML ne proposant aucune indication sur la signification de son contenu. Ces outils doivent donc être utilisés en plus d'un outil capable d'associer une signification ou des actions à compléter aux informations extraites. Par exemple, un outil qui permet d'appliquer une feuille de style.

Plusieurs formats de feuille de styles ont été entrepris d'être définis par le W3C. Le DSSSL était surtout destiné au format SGML, trop théorique pour être mis en œuvre. Le CSS a été défini au départ pour insérer des styles dans les fichiers HTML. Mais sa dernière version est tout à fait capable de jouer le rôle de feuille de style pour des documents XML. De plus, elle complètement définie et est déjà mise en œuvre dans certains butineurs, tel IE5.

Le dernier format, destiné spécifiquement à XML est le XSL. Sa définition n'est pas pour le moment (06/99) définitive et les quelques implémentations existantes ne peuvent garantir leur compatibilité avec une définition qui évolue constamment. Cependant, beaucoup de monde semble prévoir qu'à terme, ce format devrait se présenter comme standard, d'autant plus que son écriture repose sur le format XML. Il est certain qu'une fois la définition d'XSL approuvée par le W3C, tous les outils vont la mettre en œuvre, notamment les butineurs principaux, IE5 et Netscape (Mozilla).

2.2.2 - XSL

2.2.2.1 - Principe

XSL est un des formalismes définis pour jouer le rôle de feuille de style. XSL a été défini pour servir spécifiquement aux documents XML. Une feuille de style XSL se présente sous la forme d'un fichier au format XML, utilisant une syntaxe particulière XSL, définie par le W3C.

Le « langage » XSL est affublé de mots clefs, qui représentent les noms des tags et leurs attributs. Cette syntaxe s'ajoute à la structure de base d'un fichier XML. Ces mots clefs permettent de donner des ordres de mise en page.

XSL est un langage semi-déclaratif (sa nature est sujette à plusieurs débats) qui indique comment afficher les données relatives à tels tags du document XML source.

Le principe de la transformation XSL est de transformer l'arbre initial de la feuille XML en un autre arbre, aux nœuds duquel on applique ensuite des styles de formatage. Les déclarations XSL sont une série de « Captures » de « Motifs », parmi toutes les balises de la

feuille XML, qui définissent le style ou les transformations à appliquer aux balises correspondant au motif.

Ainsi, on peut capturer toute les balises de nom `prix` et ajouter le signe \$ à la valeur attribuée :

```
<xsl:template match="prix">
  $<xsl:apply-templates/>
</xsl:template>
```

Pour appliquer un format à une balise, l'ancienne version, seule mise en œuvre pour le moment utilisait les balises prédéfinies du langage HTML. La nouvelle feuille de travail du W3C utilise des balises particulières, derrière le préfixe `fo`, qui définissent le style de formatage (taille de police, couleur...) Ces données s'apparentent beaucoup à celles de CSS, ancien standard de feuille de style pour XML, qui a l'avantage d'être définitif.

2.2.2.2 - Utilisation

XSL est avant tout destiné à être utilisé comme feuille de style à des documents XML disponibles sur internet pour les afficher dans un navigateur. Un langage de mise en page, donc, pour lequel un grand nombre de mots-clefs/macros sont défini pour décrire des styles de présentation (italique, encadré, cadres flottants...)

Cependant, XSL peut-être considéré comme un langage de transformation d'un document XML en un autre document. Ce dernier peut-être, dans le cas le plus simple, au format XML (ou HTML, qui est compatible), mais on peut également obtenir un format « texte brut » auquel on peut faire prendre le format désiré (cela avec néanmoins moins de facilité)

2.2.2.3 - Outils

Les outils permettant de mettre en œuvre XSL se nomment « XSL Processors ». Leur rôle est de lire un fichier XML, une feuille de style XSL et de produire le résultat mis en page et formaté. Ils s'appuient sur les « parsers » XML pour lire le fichier XML source et la feuille de style XSL, écrite également au format XML.

2.2.2.4 - Remarques

XSL n'est encore actuellement qu'un « working draft » (brouillon) du W3C. Sa spécification n'est donc pas dans sa version définitive, ce qui entraîne un comportement et une efficacité pour l'instant incertains de la part des outils XSL. Ils ne mettent pas tous en œuvre la même version de spécification du standard, et pas toujours de façon complète ; de plus, étant, à l'instar des spécifications, toujours inachevés, ils ne sont pas encore optimisés et leur efficacité est à démontrer.

Il est raisonnable de penser que d'ici quelques mois sa définition sera validée et que peu de temps après apparaîtront des outils valables. Cependant, XSL n'en reste pas moins un langage assez complet, donc complexe, pour lequel un grand investissement doit être fourni pour maîtriser toutes les subtilités. Du moins, en se basant sur le texte des spécifications, pour l'instant le seul document complet sur le langage ; mais, encore une fois, il est fort probable que dans l'avenir, lorsque le standard sera adopté, des documents de vulgarisation apparaissent.

De toutes les façons, si XSL ne peut pour le moment pas être utilisé à cause de son inachèvement, il est fort possible de se retourner vers CSS qui, s'il est moins complet, est plus « stable » qu'XSL.

2.2.3 - RDF

2.2.3.1 - Principe

RDF est un formalisme défini par le W3C dont le but est de représenter des graphes. Le principe en est simplifié au maximum, et consiste à associer à un nœud du graphe (ressource) une valeur pour une propriété donnée. Par exemple, comme RDF est avant tout destiné à la description de ressources informatiques sur le Web, On peut ainsi dire que la valeur de la propriété « Auteur » pour la ressource « Document d'étude technologique, projet Chronique des Fouilles » est « Franck Eyraud ».

La formalisation RDF peut se faire sous trois formes :

- + graphique, avec nœuds, arcs orientés et valués.
- + sous forme de triplets (ressource, propriété, valeur)
- + au format XML, avec une syntaxe définie, ainsi que des variantes plus ou moins concises, et des règles pour générer une liste de triplets à partir d'un « fichier RDF ».

2.2.3.2 - Utilisations

Actuellement, aucune réelle utilisation n'existe. Les utilisations envisagées dans le futur sont toutes basées sur des recherches améliorées de document par l'utilisation de graphes conceptuels dans leur description, représentés en RDF.

2.2.3.3 - Outils

Les outils utilisant RDF sont pour l'instant très peu nombreux. À terme, il est raisonnable de penser que les butineurs prendront en compte ce standard, mais uniquement pour l'utilisation de la description des documents. Les outils actuellement existants proposent uniquement l'interprétation d'un fichier RDF pour en extraire les triplets, et des représentations graphiques du graphe RDF correspondant. Mais leur application n'est pour l'instant que démonstrative.

Il existe cependant des API java réutilisables pour l'interprétation de RDF. Ces derniers, tels que SiRPAC, mettent à la disposition de programmeurs une interprétation du fichier XML/RDF et fournissent les triplets générés. C'est ensuite à l'application de définir les traitements sur les triplets extraits.

Il n'existe pas pour le moment d'outil de sérialisation de RDF ; c'est-à-dire l'opération inverse de génération d'un document RDF/XML à partir d'une structure en triplets. Cet état de fait empêche pour l'instant d'utiliser RDF comme « format de stockage ». Les fichiers RDF actuellement existants sont tous produits manuellement. Cependant, plusieurs outils répondant à ce besoin sont en développement, sous le regard de la communauté spécialiste de RDF.

2.3 - Conclusions pour le projet

Au vu de cette étude, et après confrontation avec les besoins du projet, l'orientation de l'utilisation de ces différentes technologies a été légèrement revue dans les directions suivantes :

- utilisation du couple XML-XSL limitée dans la semi-structuration des documents consultés, pour faciliter leur mise en page et son évolution en utilisant le système de transformation selon le modèle d'une feuille de style. La semi-structuration dans un format standard qu'est XML permettra entre autres de faciliter l'automatisation de la saisie informatique de l'ensemble des articles de la Chronique des Fouilles, à l'aide d'outils existants (ou qui existeront) ou bien d'un outil spécifique, s'appuyant sur les modules de sérialisation XML, Java ou autres.
- l'utilisation de RDF serait quant à elle écartée, ne semblant pas répondre aux besoins du projet et du modèle. En effet, le modèle d'indexation sous forme de graphe s'est relativement simplifié durant la durée de cette étude, et la mise en œuvre de RDF ne ferait que compliquer inutilement le produit. D'autant plus que l'orientation de l'application prend plutôt une direction « propriétaire » qui ne nécessite pas un échange important de ces données avec l'extérieur, et donc une contrainte de standardisation de leur représentation.

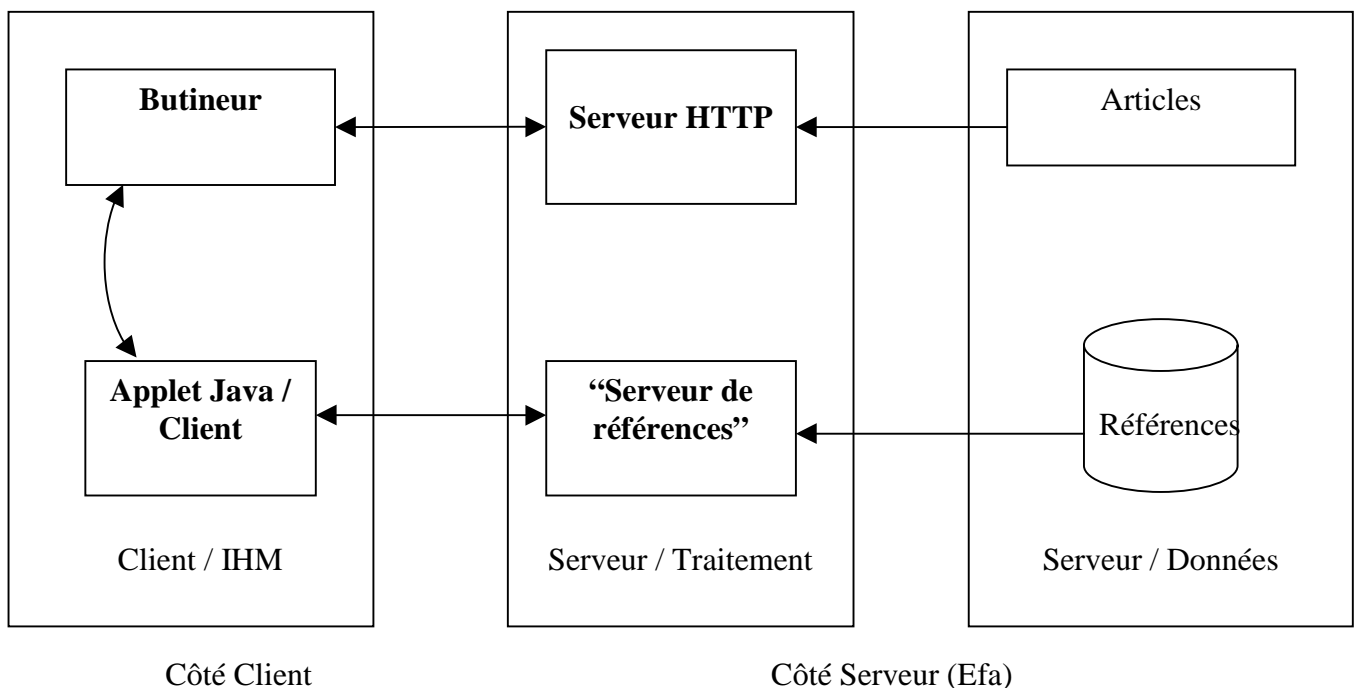
3 - CONCEPTION DU PROTOTYPE

3.1 - Étude de l'architecture

Une partie de la durée du stage a été dédiée à l'étude de l'architecture de l'application finale, les différentes parties applicatives (clients, serveurs) et la répartition des données.

La première idée d'architecture, qui partait de l'idée d'une diffusion la plus large possible des informations que l'application tient à disposition des lecteurs, voulait au maximum tirer profit des standardisations offertes par internet, et notamment l'utilisation des technologies du web.

Architecture proposée :



Cependant, cette solution s'avérerait trop limitée au niveau des fonctionnalités supplémentaires qui pourraient être proposées aux chercheurs car son implémentation et sa gestion seraient lourdes et décentralisées, et desservies par les contraintes qu'impose une grande quantité d'utilisateurs, inconnus et de profils totalement différents.

Le projet est en effet plus ambitieux et, derrière la volonté de fournir un moteur de recherche amélioré existe le désir de créer un outil destiné à une communauté entière de chercheur, qui enregistrerait toutes leurs recherches, leurs descriptions de documents pour qu'ils puissent s'y référer dans le futur, et s'échanger des informations entre eux.

De plus, comme tous ces chercheurs sont potentiellement installés dans le monde entier, il est prévu de distribuer l'application pour installer des serveurs au sein d'une forte concentration d'une partie de la communauté, et de répartir les données pour qu'elles soient situées de préférences au plus proche des personnes qui les utilisent.

L'architecture simplifiée proposée ne constitue alors plus qu'une ébauche de l'architecture globale. Il ne semble néanmoins pas évident, devant cette optique d'application que l'utilisation de l'outil web s'avère nécessaire. Il semble en effet envisageable de ne distribuer l'application qu'en certains endroits bien définis et connus, ce qui permet en plus avec un développement spécifique de mettre en œuvre plus facilement les fonctionnalités désirées. Quitte à éventuellement proposer aux lecteurs anonymes un accès limité à l'application via le web et une version bridée sous forme d'applet Java.

3.2 - Réalisation d'un prototype (Porphyry)

La réalisation d'un prototype pour valider le modèle d'indexation spécifié dans l'étude théorique et estimer l'ergonomie de ce dernier dans un cas d'exemple a semblé être le choix suivant à effectuer. Abstraction sera donc faite de toute la partie architecture pour se concentrer sur l'« interface homme-machine » et le calcul des requêtes émises par l'utilisateur.

Le prototype sera donc un produit mono-poste, pour lequel l'agencement en modules fonctionnels devra respecter l'esprit de l'architecture globale future. Ainsi, puisque à terme le « serveur de références » qui calcule le parcours dans le graphe en fonction des choix de l'utilisateur et le client qui assure le dialogue avec l'utilisateur et la présentation graphique des résultats seront présents sur au moins deux systèmes séparés, il est important de les développer de façon indépendante dans le prototype. Et, entre ces deux parties, une troisième, que l'on nommera « contrôleur », se chargera de la coordination entre elles, symbolisant le rôle de la partie communication de l'architecture finale.

Nous avons donc, pour le prototype, le schéma suivant :

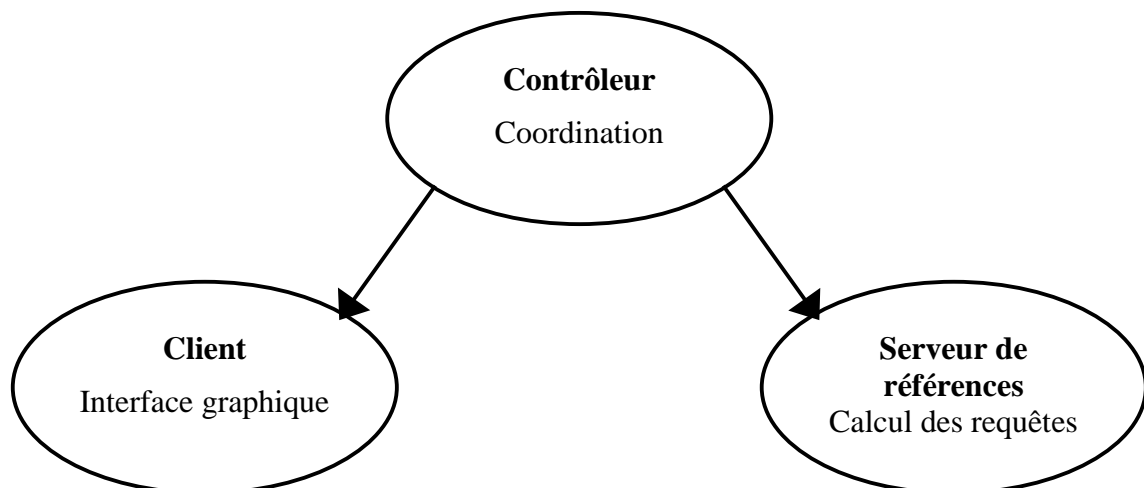


Schéma d'architecture du prototype Porphyry

Il a été décidé de développer ce prototype en utilisant le langage Java. Tout d'abord, sa simplicité et sa grande modularité permettent un développement simple et efficace. De plus, ce langage pourra favoriser, à terme, la distribution multi-plateforme de l'application, grâce à son indépendance du matériel.

3.3 - Le prototype

3.3.1 - Présentation

Le prototype « Porphyry », à son état à la fin de ce stage, met donc en œuvre les possibilités suivantes :

- navigation graphique dans un arbre d'indexation (sélection de nœuds, résultats sur le graphe)
- affichage simultané de plusieurs recherches dans des fenêtres indépendantes
- possibilité (pour l'instant sommaire) de modification de l'arbre d'indexation (ajout de nœuds, ajout et suppression d'arcs)

Fonctionnalités non encore incluses dans ce développement :

- identification de l'utilisateur et séparation des différentes classes d'utilisateurs
- stockage des informations de l'arbre d'indexation (travaille actuellement uniquement en mémoire)

3.3.2 - Fonctionnement

Dans cette version limitée de l'application, le fonctionnement en est tout aussi simple.

→ Navigation

Pour démarrer une recherche, il est nécessaire d'ouvrir au moins une nouvelle fenêtre. Dans une fenêtre nouvellement créée est automatiquement sélectionné le descripteur « Universel », qui représente l'ensemble des documents. Ensuite, la sélection d'un nœud fournit le résultat de la restriction, en terme d'arcs connus, possibles ou impossibles (distingués par la couleur). De même, un descripteur peut être désélectionné pour élargir le domaine de la recherche.

La sélection d'un descripteur impossible (au bout d'un arc impossible) aurait comme résultat un corpus vide, ce qui est absurde. Par conséquent, cette action a comme effet la désélection de tous les autres descripteurs déjà sélectionnés.

Lors de la navigation, il est possible de réarranger la disposition du graphe représenté en faisant glisser les descripteurs à sa guise (les calculs de disposition des nœuds du graphe sont éventuellement à améliorer pour gagner en ergonomie)

→ Ajout de descripteur

L'ajout d'un descripteur se fait en choisissant l'option correspondante dans le menu de la fenêtre principale. Cette action ne demande que l'entrée d'un nom pour ce nouveau descripteur (les doublons sont autorisés).

Le descripteur ne figure alors pas encore sur le graphe, car il n'est lié à aucun autre descripteur. Une liste des descripteurs isolés existe, et est automatiquement montrée lors de la création d'un nouvel arc pour pouvoir y piocher un de ces nœuds.

→ Création d'un arc

Pour ajouter un arc, et après avoir choisi l'option dans le menu, il est nécessaire de désigner, dans l'ordre le nœud père et le nœud fils de l'arc. La sélection peut se faire dans une des fenêtres de navigation en cliquant sur le nœud désiré. Le père doit obligatoirement être un nœud non isolé ; en revanche, la sélection du nœud fils peut se faire dans la fenêtre des nœuds isolés.

On peut annuler la création d'un arc en cours en choisissant l'item de menu correspondant, dans la fenêtre principale. Pendant la création d'un arc, plus aucune navigation n'est possible et ce jusqu'à l'annulation, la création effective de l'arc ou l'apparition d'un message d'erreur en cas de création illicite, détectée par le moteur d'indexation.

4 - CONCLUSION

Ce stage a donc permis, pour le projet « Chronique des fouilles » de donner une petite note technique et pratique aux recherches théoriques menées sur la mise à disposition aux chercheurs des articles de cette revue de l'École Française d'Athènes. La concrétisation en a été la production d'un produit permettant de visualiser une base de l'application future.

La veille technologique sur les technologies nouvelles de l'information a également apporté certaines réponses, ou des éléments de réponses, sur les choix à effectuer par la suite. De plus, elle permettra certainement de se faire une idée plus précise de l'opportunité d'accorder un intérêt à ces technologies, pour d'autres parties de ce projet, ou même pour de futurs autres projets.