



**HAL**  
open science

## Création d'un site 2D et 3D du Musée de Frontignan

Bruno Delcol

► **To cite this version:**

| Bruno Delcol. Création d'un site 2D et 3D du Musée de Frontignan. Web. 2010. dumas-00524545

**HAL Id: dumas-00524545**

**<https://dumas.ccsd.cnrs.fr/dumas-00524545v1>**

Submitted on 8 Oct 2010

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

**CENTRE REGIONAL ASSOCIE DE MONTPELLIER**

---

**MEMOIRE**

**présenté en vue d'obtenir  
le DIPLOME d'INGENIEUR CNAM**

**SPECIALITE : Réseaux et Systèmes Multimédia**

**OPTION : INFORMATIQUE**

**par**

**DELCOL Bruno**

---

**Création d'un site 2D et 3D du Musée de Frontignan**

**Soutenu le 6-7 juillet 2010**

---

**JURY**

**PRESIDENT :** M.Yves LALOUM

**MEMBRES :** M. Michel SALA (Responsable régional de la filière),  
M. Pierre POMPIDOR (Tuteur pédagogique),  
Mme. Simone TANT (Adjointe au Maire – Déléguée à la Culture),  
M. Robert FABRE (Directeur du service Culture, Fêtes et Patrimoine).

## Remerciements

Je tiens à remercier personnellement les différentes personnes qui m'ont consacré leur temps dans l'élaboration de ce mémoire :

M. Michel SALA, mon tuteur de stage, élu au patrimoine de la ville de Frontignan, pour m'avoir proposé un sujet aussi intéressant et formateur ;

M. Pierre POMPIDOR pour m'avoir dirigé et conseillé tout au long de mon projet ;

M. le Maire, Pierre BOULDOIRE pour m'avoir accueilli au sein de sa collectivité ;

Mme Simone TANT, M. Robert FABRE ainsi que tout le personnel du service et d'autres services à l'occasion de nos rencontres, pour m'avoir accueilli chaleureusement ;

Mlle Joanne MILLET, pour sa gentillesse et sa très grande participation à ce travail ;

Mme Carole BRIFFAUD, archiviste, qui m'a accueilli au sein de ses locaux durant toute la durée de mon stage.

## Sommaire

Remerciements.....	2
Sommaire .....	3
Acronyme.....	5
I. Introduction .....	6
II. Présentation de l'entreprise .....	8
2.1 La ville : .....	8
2.2 La Situation géographique:.....	9
2.3 La mairie :.....	9
2.4 Les services :.....	11
2.5 Le Service Culture/Fête/Patrimoine :.....	12
2.6 Education Enfance Jeunesse: .....	13
III. Etudes de l'existant et des besoins.....	13
3.1 Etude de l'existant .....	13
3.2 Etude des besoins .....	14
3.3 Charges et planification : .....	17
3.4 Bilan.....	19
IV. Etat de l'art des sites existants et des technologies .....	19
4.1 Etat de l'art des sites existants : .....	19
4.2 Etude des technologies existantes .....	26
4.2.1 Les bases de la 3D (la modélisation) :.....	26
4.2.2 Technologies hors navigateur (Open GL, DirectX,...) .....	27
4.2.3 Technologies intra-navigateur (VRML, 3DMLW) .....	29
4.2.4 Technologies supportées par le Flash Player.....	33
4.2.5 Comparatif des 4 principaux Moteurs 3D : .....	33
4.3 Bilan.....	41
V. Le musée virtuel en 2D.....	43
5.1. Charte graphique et charte éditoriale .....	43
5.2. Réalisation du Modèle Conceptuel de Données :.....	44
5.3 Conception : .....	47
5.3.1 Etude de l'enchaînement des pages :.....	48
5.3.2 Réalisation d'un musée virtuel en 2D : .....	49
5.3.3 La planification .....	49
VI. Les circuits touristiques .....	52
6.1 Google Earth : .....	52
6.2 Google Maps : .....	52
6.3 Procédure de création d'un circuit touristique :.....	53
VII. Le musée virtuel en 3D.....	57
7.1 Etude du musée en 3D.....	57
7.1.1 Introduction.....	57
7.1.2 Modélisation des fonctionnalités attendues (uses cases).....	58
7.1.3 Mise en application.....	60
7.2 Aperçu du Framework PV3D.....	69
7.2.1 Introduction.....	69

7.2.2	Création de l'environnement 3D .....	70
7.2.3	Fonctionnement d'un environnement Papervision3D .....	70
7.2.4	Espace global et espace local .....	71
7.2.5	Les Matériels : .....	72
7.2.6	Les caméras : .....	73
7.2.7	Optimisation de l'application PV3D : .....	74
VIII.	Conclusion.....	75
	Liste des figures.....	77
	Liste des tableaux.....	78
	Glossaire.....	79
	Webographie .....	81
	Bibliographie .....	83
	Annexes.....	84
	Cahier des charges.....	89
	Exemple de compte rendu de réunion.....	112
	Le Framework PV3D .....	112
	RESUME .....	136

## Acronyme

API	Application Programming Interface
AS 2.0	Action Script 2.0
AS 3.0	Action Script 3.0
CPU	Central Processing Unit
Do3D	Display object 3D
GC	Garbage Collector
GUI	Graphical User Interface
LOD	Level Of Detail
MCD	Modèle Conceptuel de Données
MLD	Modèle Logique de Données
MVF	Musée Virtuel de Frontignan
PV3D	PapperVision3D
SDK	Software Developement Kit
XML	eXtended Markup Language
VRML	Virtual Reality Modelling Language

## **I. Introduction**

Depuis plusieurs dizaines d'années, la ville de Frontignan a acquis divers objets qui constituent de véritables collections. Certains de ces objets sont exposés au Musée Municipal, hébergé à la chapelle des Pénitents Blancs. Ces collections s'enrichissent au fur et à mesure de découvertes ou autres moyens d'acquisitions (dons par exemple). En 2009, le nombre d'objets approximativement en possession de la ville est estimé à environ 3000. Une des contraintes de la ville est la taille du musée qui ne peut accueillir que seulement 10% des objets possédés.

La ville a décidé de mener différentes actions pour promouvoir le patrimoine de la ville à travers le musée municipal et le développement de circuits touristiques. Le service de Communication de la ville a mis en place plusieurs actions, plaquettes et autres documents pouvant servir de support et ainsi attirer les gens à venir découvrir les richesses de la ville. Les « frontignanais » pourront alors connaître à travers le patrimoine, l'histoire de cette ville de l'Hérault qui borde la mer méditerranée.

Une des solutions la plus intuitive retenue, est la création d'un site internet qui présenterait le musée municipal en 2D, en 3D ainsi que lesdits circuits.

La volonté de la municipalité est de mettre la culture à la portée de tous.

Ma lettre de mission consiste d'une part à concevoir et d'autre part à réaliser, un musée virtuel en 2D et un prototype de musée en 3D et la création de circuits touristiques. Une étude de faisabilité du musée en 3D sera réalisée et un prototype de ce dernier pourra être proposé en dernier lieu. Une des contraintes majeures et non négligeable reste un budget nul.

Pour mener à bien ce projet, nous devons faire une étude de l'existant et des besoins, de réaliser un état de l'art des sites existants et des technologies mises à disposition pour en choisir une qui sera retenue répondant aux besoins spécifiés.

C'est alors que nous pourrons concevoir et réaliser les deux musées nécessaires ainsi que les circuits touristiques pour répondre aux besoins de la commune de Frontignan.

En ce qui concerne la modélisation en 3 dimensions (ou 3D), elle restait jusqu'à nos jours réservées au domaine de la cinématographie ou des jeux vidéo. Aujourd'hui grâce à différentes technologies existantes la 3D devient accessible pour d'autres domaines et notamment celui d'Internet.

Depuis la création d'Internet, différentes technologies ont vu le jour pour agrémenter les sites internet, et notamment les rendre plus jolis, plus interactifs, plus animés voire plus attractifs en passant de page statique en page dynamique.

Il y a un peu plus de dix ans déjà que la technologie Flash de Adobe existe et a énormément contribué à rendre plus animés les sites marchands comme ceux pour les particuliers. Beaucoup de professionnels de la publicité ont tiré parti de cette technologie pour promouvoir différentes marques en créant et proposant des animations Flashes en deux dimensions au début, pour aujourd'hui intégrer des animations en trois dimensions.

Nous verrons les différentes technologies qui permettent ce genre de mise en œuvre. Ces technologies contribuent à créer des animations interactives en 3D et offrent une réelle interaction et participation des internautes. Elles permettent par exemple de simuler des pièces, de naviguer dans celles-ci et offrir un réalisme assez fort de ce que pourrait être la réalité.

L'idée de créer un Musée Virtuel en 3D, pour la ville de Frontignan va permettre d'offrir aux futurs internautes, à travers l'utilisation d'une de ces technologies novatrice en la matière, la possibilité de naviguer dans un environnement et de découvrir des œuvres qui ne sont malheureusement pas mises au jour pour des raisons de place actuellement restreinte et de mettre en œuvre une de ces technologies. La ville de Frontignan compte à terme intégrer cette création dans le site officiel de la ville ou créer un site indépendant pour mettre la culture à la portée de tous.



## II. Présentation de l'entreprise

Le cadre de ce projet est la Mairie de la commune de Frontignan, sise dans l'Hérault.

### 2.1 La ville :

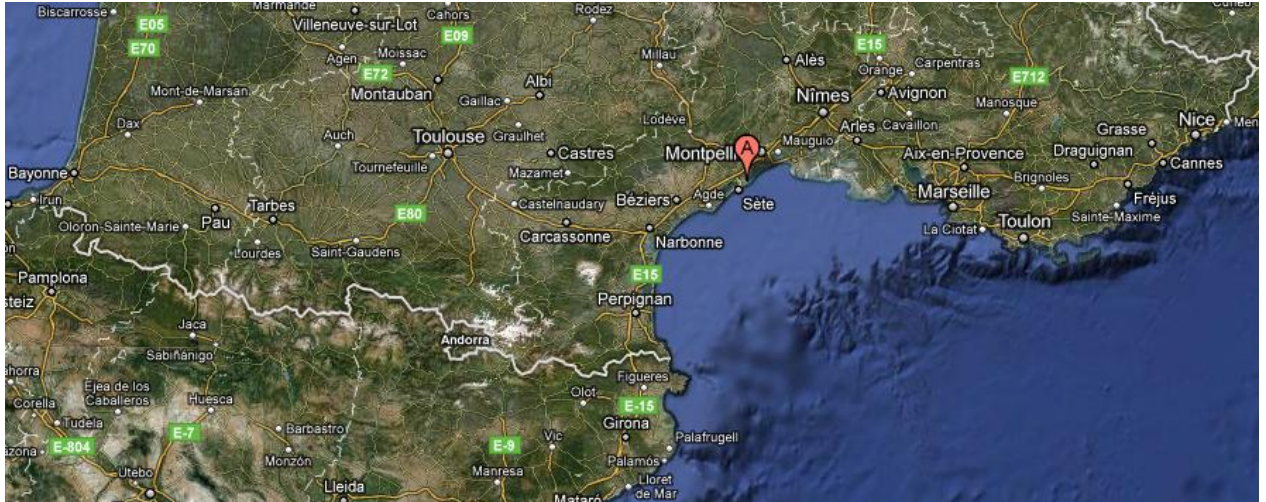
**Frontignan** est une commune française située dans le département de l'Hérault et la région Languedoc-Roussillon. Chef-lieu de canton, elle est la cinquième ville du département en termes de population, après Montpellier, Béziers, Sète et Lunel. Depuis le 31 décembre 2002, elle fait partie de la Communauté d'agglomération du Bassin de Thau.

Le quartier excentré de la Peyrade, revendique son indépendance de la ville de Frontignan, et les habitants de ce quartier, tiennent à s'appeler *Frontignan-La-Peyrade*. Pour autant, La Peyrade n'a jamais été une commune autonome, même si elle possédait jusqu'en 1988 un collège électoral séparé de Frontignan. La Peyrade, tout en étant effectivement le quartier situé le plus près de Sète, est encore aujourd'hui considéré par certains de ses habitants comme une entité qui ne partage avec Frontignan que la structure communale.

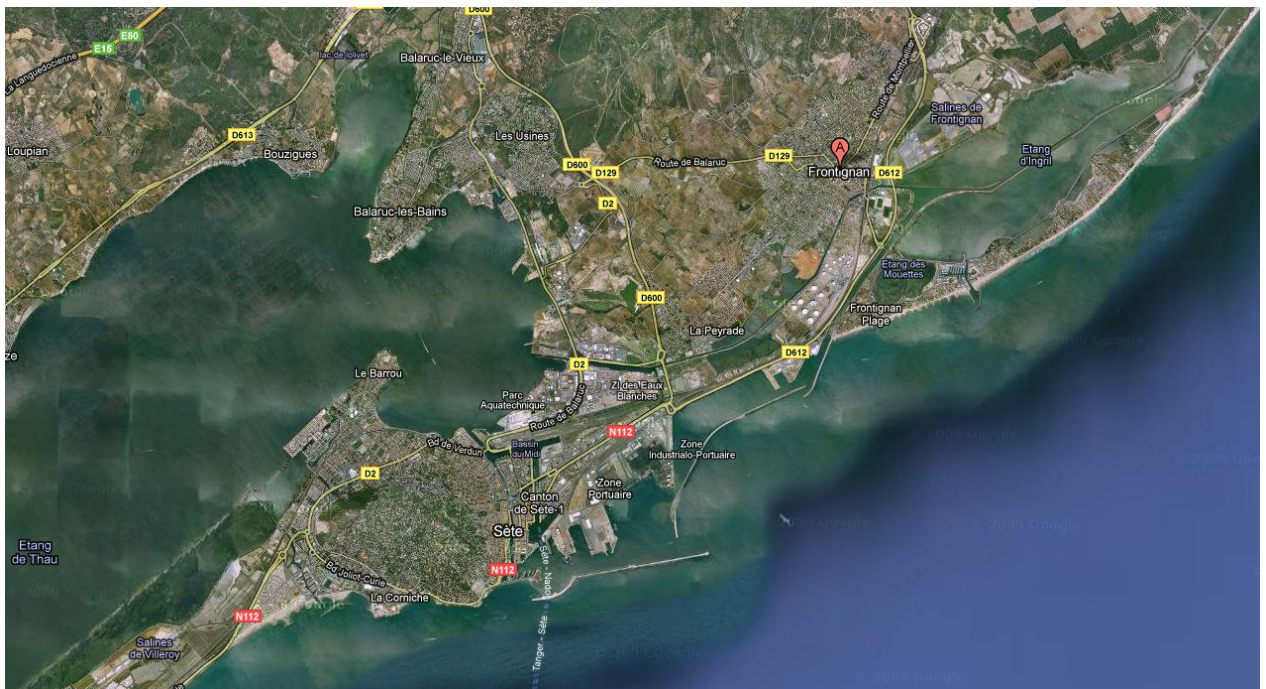
Ses habitants sont appelés les « Frontignais ». Leur surnom est les « Ventres-Bleus ».

Frontignan tire son nom du consul romain Sextus Julius Frontinus, dit « Frontin », qui venait régulièrement y séjourner. La ville est célèbre dans le monde entier pour son muscat. La légende raconte que, lors de son passage à Frontignan, Hercule tordsait la bouteille afin de la finir jusqu'à sa dernière goutte, d'où sa forme si particulière.

## 2.2 La Situation géographique:



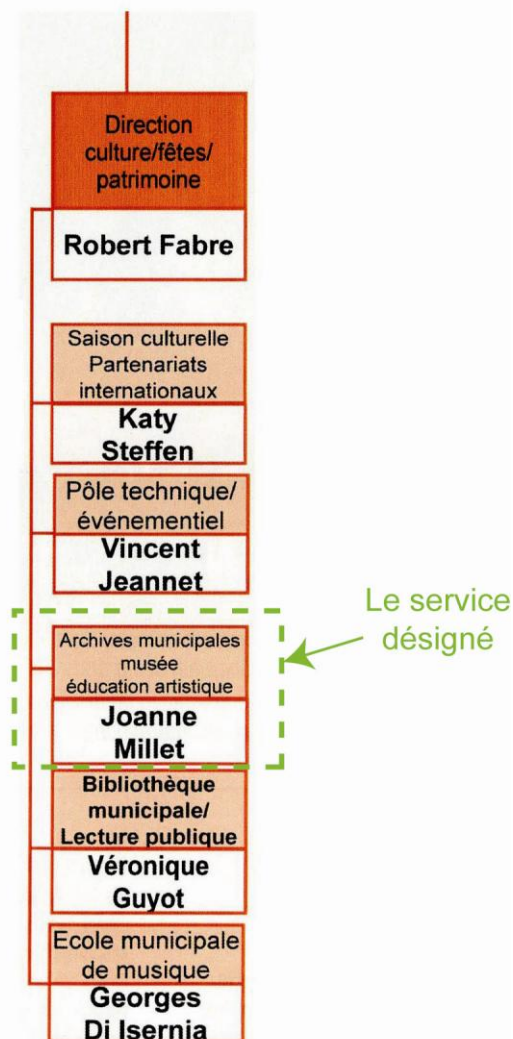
1 – Vision satellite de la région



2- Vue satellite de Frontignan et ses alentours

## 2.3 La mairie :

La mairie de Frontignan est constituée de différents services représentés dans l'organigramme général (voir Annexe Page 103).



### 3-Extrait de l'Organigramme fonctionnel des services de la ville de Frontignan

Simone TANT qui ne figure pas sur l'organigramme est l'élue au service de la direction « culture/fête/patrimoine ».

Mr Robert FABRE et Mlle Joanne MILLET ont été mes référents tout au long de mon stage concernant l'inventaire du musée et les musées virtuels afin de mener à bien ce projet.

## 2.4 Les services :

Nous allons expliquer brièvement leurs différentes missions.

Le « Cabinet du Maire » regroupe le service de proximité qui gère les relations avec les administrés comme par exemple les conflits, les réclamations.

Le service « Communication » s'occupe de la gestion de la communication interne et externe de la ville, avec la presse. La gestion du site Internet de la ville ainsi que le journal municipal sont aussi de leur ressort.

Les services « techniques » regroupent trois autres services.

Le service « Bâtiment/ sécurité civile » qui est ne charge de la réalisation des travaux neufs et en régie interne des bâtiments de la ville.

Le service « Logistique/matériel » qui s'occupe de la logistique matérielle des manifestations publiques, de gérer le stock du magasin, le parc automobile et les différents équipements.

Le service « Espaces publics et environnement » a la charge de l'aménagement et de la mise en œuvre des espaces publics comme l'éclairage, les ronds-points, la sécurité, la voirie, la propreté urbaine, les espaces verts et les arrêtés municipaux.

Le service « Urbanisme / aménagement » gère le patrimoine foncier de la ville, des planifications, des études urbaines, mais également le cadastre et les permis de construire.

La direction Générales des services est composée de 4 services.

Le service « Commerce » gère les relations du tissu économique de la ville, l'organisation des marchés à évènementiel comme le marché de Noël, les festivals mais également le parc informatique, la téléphonie fixe et mobile, l'achat de logiciels et des interventions nécessaires pour mener à bien leurs missions.

Le service des « Ressources Humaines » a en charge la gestion du personnel, du déroulement des carrières, des formations, de la paie, du pôle emploi et des formations pour les jeunes.

Le service « Finance » gère la facturation, la régie municipale et l'encaissement des prestations municipales comme la cantine, la garderie, le centre aéré, les activités péri et extra scolaires.

Le service « Sport et loisirs » gère les équipements sportifs, les animations péri et extra scolaires, les animations sportives de plein air, du port de plaisance ainsi que de l'office de tourisme.

La direction Administration Générales des services est composée de 5 services. Le service « Juridique » s'occupe de la gestion de l'aspect juridique des conventions passées entre la Mairie et les particuliers ou les associations.

Le service des « Achats Publics » gère les marchés publics et les achats de la ville soumis au code des marchés publics.

Le service Etat civil /Elections gère les enregistrements d'ordre civil comme les naissances, les mariages et les décès, de l'organisation des élections politiques.

Le service des « Moyens Généraux » gère l'accueil de l'hôtel de ville, la transmission du courrier interne, des polices d'assurances de la ville comme les assurances automobiles, des équipements, des sinistres divers et de la préparation administrative et l'organisation des conseils municipaux.

Le service de la « Police Municipale » s'occupe de gérer la sécurité sur la commune, les stationnements gênants, de faire respecter les arrêtés de stationnement, d'établir des procès-verbaux les cas échéants.

Le service du patrimoine, le musée et l'office du Tourisme ont une volonté commune de promouvoir les richesses patrimoniales et culturelles de la ville.

La création d'un Musée Virtuel permettra de répondre à leur besoin.

### **2.5 Le Service Culture/Fête/Patrimoine :**

Le service de la culture est en charge de la programmation culturelle de la ville, des ateliers de pratiques artistiques organisés dans les écoles, de la gestion du Musée et des collections, de la gestion des archives et des publications et autres expositions patrimoniales.

Quant au service fêtes, il s'occupe de la programmation de la saison culturelle de la ville, des échanges et partenariats internationaux (jumelage).

Le Pôle technique/Événementiel traite de la partie technique et logistique du petit matériel nécessaire à l'événementiel en cours (FIRN, Festival Muscat, journée des associations).

La bibliothèque gère l'organisation des lectures publiques mais aussi de l'apprentissage avec les scolaires. Elle propose des ateliers thématiques, et s'inscrit dans le cadre des opérations nationales (le temps des livres...)

L'école de musique organise le planning des intervenants et assure les programmations musicales intra et extra muros.

### **2.6 Education Enfance Jeunesse:**

En charge du bon fonctionnement des équipes municipales dans les primaires, maternelles, elle se charge de gérer les temps entre 12h et 14h mais aussi des accueils matin et soir.

La réussite éducative assure grâce à un encadrement des enfants en difficultés sociales et éducatives et aussi liées aux primo-arrivants. Soutenu par des référents qui suivent individuellement les enfants ainsi que les familles pour un accompagnement efficace.

La restauration et l'entretien est en charge du restaurant scolaire sur le canton (transports, conditionnement)

Le service jeunesse anime les après-midi des 15-25 ans à Frontignan-Lapeyrade. La gestion des évènementiels (Festival Pop, la carte Kifo..) qui permet aux jeunes de la ville de bénéficier d'avantages et de réduction chez les commerçants locaux.

## **III. Etudes de l'existant et des besoins**

### **3.1 Etude de l'existant**

Le musée municipal de la ville de Frontignan utilise un logiciel développé, par leur service informatique, sous ACCESS, nommé « GESPIMUS » pour gérer les objets

des différentes collections qu'il possède. Il est utilisé pour la gestion de l'inventaire des objets et des différentes caractéristiques qui les décrivent.

Hors l'inventaire n'est pas à jour pour différentes raisons (temps, personne...).

Il semblerait également que les utilisateurs potentiels ne trouvent pas ce dernier intuitif et facile d'utilisation.

### **3.2 Etude des besoins**

La ville de Frontignan a besoin de refondre le système d'informations pour la gestion de l'inventaire du Musée Municipal qui a l'appellation « Musée de France ». De ce fait, la base de données doit être compatible avec la base « Joconde » qui sera expliquée en premier lieu. Ensuite elle a aussi besoin de créer des circuits touristiques et la volonté de créer un musée virtuel en 2D et 3D.

Pour permettre d'établir les besoins, plusieurs entretiens avec Mlle Joanne MILLET ont été nécessaires ainsi qu'avec tout le personnel du service en complément d'informations. Des comptes rendus de réunion ont été établis pour améliorer après chacun de ces entretiens la définition des besoins. (Voir compte rendu en annexe P128).

#### **3.2.1 La base « Joconde » :**

Constitués au cours des siècles, les collections des musées de France sont d'une richesse remarquable. En raison de leur abondance et de leur répartition sur l'ensemble du territoire national, elles peuvent être parfois difficiles à appréhender par le public. Aussi le ministère de la culture et de la communication a entrepris de mettre en œuvre un catalogue collectif des collections significatives des musées de France.

Depuis le 10 mars 2004, ce catalogue, auparavant réparti en trois bases de données (collections des beaux-arts et d'arts décoratifs, archéologie et ethnologie) a été réuni en une base unique nommée « Joconde ».

Cette volonté de mise en ligne des collections, exposées ou en réserves atteste des quatre missions permanentes assignées aux musées de France qui sont

premièrement de conserver, restaurer, étudier et enrichir leur collections, deuxièmement de rendre leurs collections accessibles au public le plus large, troisièmement de concevoir et mettre en œuvre des actions d'éducation et de diffusion visant à assurer l'égal accès de tous à la culture et dernièrement de contribuer aux progrès de la connaissance et de la recherche ainsi qu'à leur diffusion.

Joconde est le fruit d'un partenariat étroit et quotidien avec les musées de France. C'est ainsi que la catalogue s'enrichit sans cesse. Chaque mois les nouvelles notices sont signalées, musée par musée. L'actualité du site est régulièrement valorisée.

Pour que tout le monde puisse adopter le « même langage » et puisse ainsi trouver les objets d'une même famille, le vocabulaire utilisé dans certains champs est contrôlé et harmonisé avant la mise en ligne de la notice.

En effet, en dehors de certains champs libres, rédigés en langage naturel, d'autres champs servant à la recherche utilisent un vocabulaire précis et répertorié dans les lexiques accessibles en ligne (dénominations, domaines, sujets représentés, matériaux-techniques, etc...). C'est pourquoi un certain formalisme est nécessaire pour la création de la base de données et ainsi faciliter l'exportation des fiches des objets détenus par le musée municipal de Frontignan.

### **3.2.2 Le système d'informations du Musée**

Le système d'informations en place lié avec le logiciel « GESPIMUS » et la base de données ACCESS, nécessite une refonte totale de ce dernier car il est obsolète et n'est pas compatible avec la base « Joconde » précédemment citée.

L'intérêt est d'avoir une application répondant aux différents critères d'utilisabilité et de compatibilité. Cette application devra être intuitive et facile d'utilisation.

Il s'agit de faire connaître l'existence du musée de la ville de Frontignan et d'inciter les internautes à venir visiter le musée pour y découvrir toutes ses richesses.

Effectivement ce dernier ne peut exposer que 10% des œuvres en raison d'un espace restreint, celui où est actuellement hébergé le Musée municipal, à savoir dans la chapelle des Pénitents Blancs.



Cette partie permettra de recenser les besoins et ainsi de déterminer les rubriques et sous-rubriques du site internet et le classement des œuvres dans leurs catégories et sous catégories respectives en se référant à l'existant.

### **3.2.3 Les circuits touristiques :**

La ville souhaite créer des circuits touristiques qui seront accessibles depuis internet (le site Internet de la ville). L'intérêt est de promouvoir le patrimoine de la ville et de le faire découvrir aux internautes. Différents circuits seront proposés selon leurs domaines (patrimoniaux et autres).

### **3.2.4 Les musées virtuels en 2D et 3D :**

Une autre des volontés de la Mairie est de créer un Musée virtuel en 2D et un autre en 3D pour promouvoir le musée municipal et mettre la culture à la portée de tous. Il s'agit de faire découvrir les différents trésors dont dispose la ville.

La ville de Frontignan souhaite avoir un logiciel mieux adapté que l'existant pour gérer l'inventaire de tous les objets possédés et permettre de le maintenir à jour. Etant labélisé « Musée de France », elle doit pouvoir échanger des informations de son inventaire à la DMF (Direction des Musées de France) et de ce fait être compatible avec la base de données nationale « JOCONDE ».

Le projet de « création de site Internet du Musée » a pour but de fournir à la ville de Frontignan un site Internet permettant de mettre en ligne les œuvres des différentes collections en possession du Musée Municipal.

L'intérêt est de pouvoir présenter toutes les œuvres en possession du musée triées par catégorie.

Deux méthodes seront utilisées pour offrir aux internautes une vue en 2D des œuvres et une vue en 3D du musée.

### ***3.3 Charges et planification :***

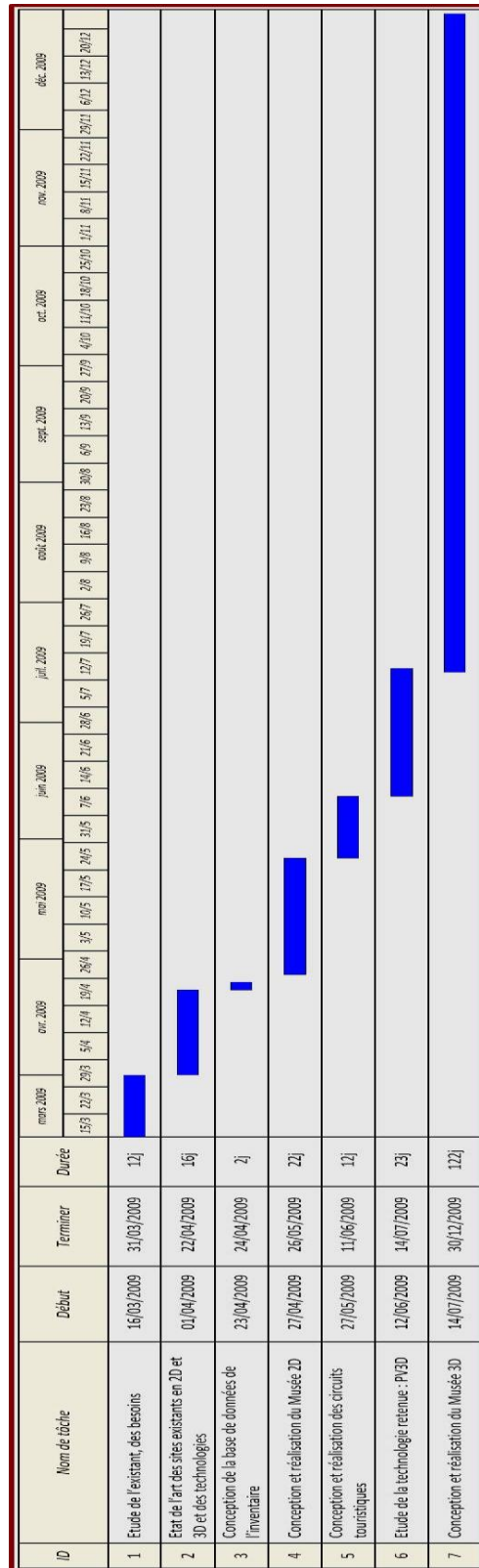
Après avoir analysé les besoins des utilisateurs, et après avoir réalisé l'étude de l'existant du système,

J'ai déterminé les différentes phases du projet (on en compte neuf). J'ai ensuite estimé et planifié les charges de chacune d'entre elles.

Le temps de la réalisation du cahier des charges est inclus dans ce diagramme. Ce diagramme représente l'organisation du travail avec une estimation des charges pour chacune des tâches comme l'analyse des besoins utilisateurs ou l'étude de l'existant. Ce dernier a été réalisé après plusieurs entretiens passés avec Mlle Joanne MILLET et les différentes personnes du service Festivité et Patrimoine.

Ce planning a été respecté.

Ce projet est composé de 7 grandes tâches qui sont ordonnées chronologiquement. La première consiste à faire une étude de l'existant et des besoins. La seconde a pour but de réaliser un état de l'art des sites existants en 2D et 3D ainsi que de rechercher et recenser les technologies respectivement utilisées pour la 2D et la 3D. La troisième a pour mission de créer la base de données de l'inventaire en fonction des spécifications et des besoins attendus. En se basant sur la base de données, le système d'informations ainsi défini, la quatrième tâche permet de concevoir et de réaliser le musée virtuel en 2D. La cinquième a pour objectif de concevoir et de réaliser les circuits touristiques. C'est dans la sixième tâche que la technologie 3D retenue pour ce projet est étudiée en profondeur. Ceci permettra alors dans la septième tâche de concevoir et de réaliser un prototype de musée virtuel en 3D.



4 – Diagramme de Gant

### 3.4 Bilan

En tenant compte de l'existant et des différents besoins recensés par la mairie, nous avons pu établir un cahier des charges pour répondre à ces derniers.

Les moyens mis en œuvre devront coûter zéro euro, ce qui est une des contraintes à respecter.

## IV. Etat de l'art des sites existants et des technologies

### 4.1 Etat de l'art des sites existants :

L'état de l'art permet d'analyser l'organisation des sites Internet de musées existants.

Un musée est un lieu ou un *établissement où est conservée, exposée, mise en valeur une collection d'œuvres d'art, d'objets d'intérêt culturel, scientifique ou technique.*

Le Conseil international des musées (ICOM) a élaboré une définition plus exigeante de la notion. Pour elle, « *Un musée est une institution permanente, sans but lucratif, au service de la société et de son développement, ouverte au public et qui fait des recherches concernant les témoins matériels de l'homme et de son environnement, acquiert ceux-là, les conserve, les communique et notamment les expose à des fins d'études, d'éducation et de délectation.* »

Les musées sont souvent spécialisés par catégories. Il existe 5 grandes catégories de musée, les musées d'Art, les musées d'Histoire, les musées de Sciences, les musées de la technique et les musées d'ethnologie.

Étymologiquement, le terme « musée » vient du grec « museion », temple des Muses, divinités des arts. C'est à la Renaissance, notamment en Italie, qu'on

nomma ainsi ces galeries renfermant des objets d'arts : des lieux habités par les Muses.

Le mot « musée », dans son acception moderne, apparaît en Italie dans la seconde moitié du XV<sup>e</sup> siècle, sous sa forme latine, « museum », à la suite de la Renaissance. En effet, les princes italiens sont les premiers à envisager l'idée d'une collection de tableaux et de sculptures, rassemblés, offerts aux regards des voyageurs et des artistes à l'intérieur des cours et des jardins, puis dans les galeries (large couloir reliant un bâtiment à l'autre). Ils associent les notions d'œuvre d'art, de collection et de public (celui-ci restant très restreint, ne concernant que des invités des princes, qui sont bien souvent d'autres princes), posant ainsi les éléments constitutifs du futur musée des arts. Ce n'est qu'à la fin du XVIII<sup>e</sup> siècle que le mot de muséum est délaissé en faveur de celui de musée, le terme de muséum ne désignant aujourd'hui que certains musées consacrés aux sciences naturelles.

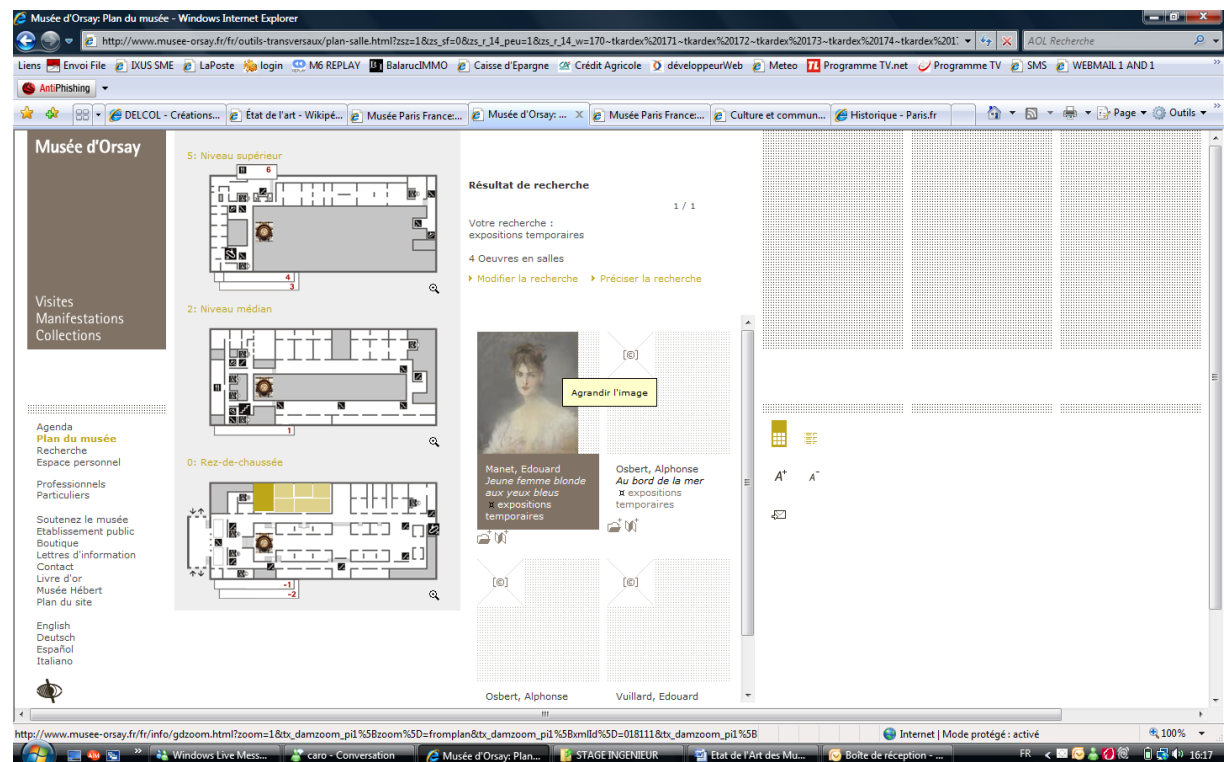
Le musée et la collection publique, tels que nous les connaissons aujourd'hui, sont une invention relativement récente, puisqu'elle date du XVIII<sup>e</sup> siècle, et peut être considérée comme le fruit de la Philosophie des Lumières. En France, c'est la Révolution qui met en place les premiers musées, pour mettre à la disposition des citoyens les œuvres d'art des collections royales ou celles qui ont été confisquées aux nobles et aux congrégations religieuses. Le musée, lieu officiel de l'exposition de l'art, occupe dès lors une place centrale dans la vie de la cité. À Paris, c'est le palais du Louvre qui est choisi pour devenir un musée.

Le but du musée, institution publique au départ, est de rendre accessible à tous le patrimoine collectif de la Nation, l'idée du beau et du savoir à travers une sélection d'objets. Le musée montre l'art, mais aussi la science, la technique, l'histoire, toutes les nouvelles disciplines porteuses de progrès et de modernité. (Extrait de l'encyclopédie).

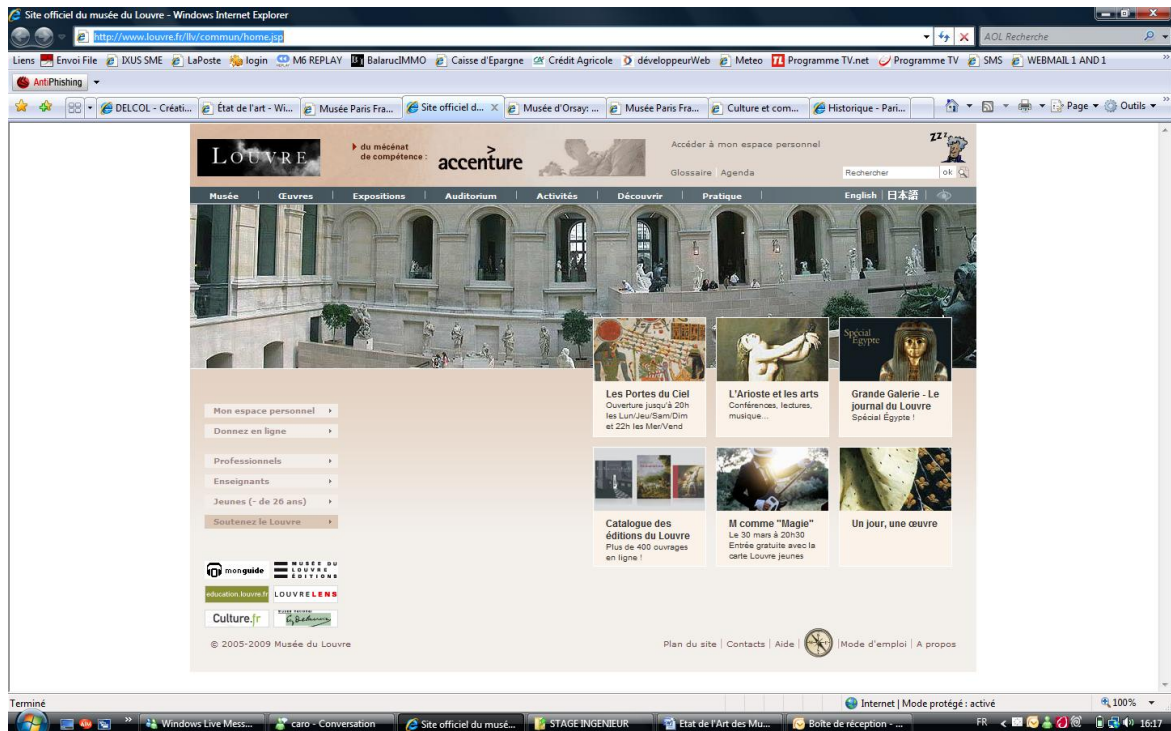
La technologie internet a permis de mettre les richesses des musées à la portée de tous enlevant ainsi les limites liées à la distance. Depuis quelques années déjà, les grands musées offrent au public la possibilité de voir leurs principales œuvres à travers des sites Internet.

Ces œuvres sont présentées en 2D (communément appelé catalogue) ou en 3D ce qui permet de représenter les pièces où elles sont stockées comme dans la réalité. Après avoir visité un grand nombre de sites de musées, voici les sites Internet référents pour cet état de l'art qui sont, le musée d'Orsay, le musée du Louvre et Organismuseum (un musée virtuel en 3D).

Il s'agit ici de lister et de visualiser les principaux sites Internet sur les musées dont certains proposent une vision 3D.



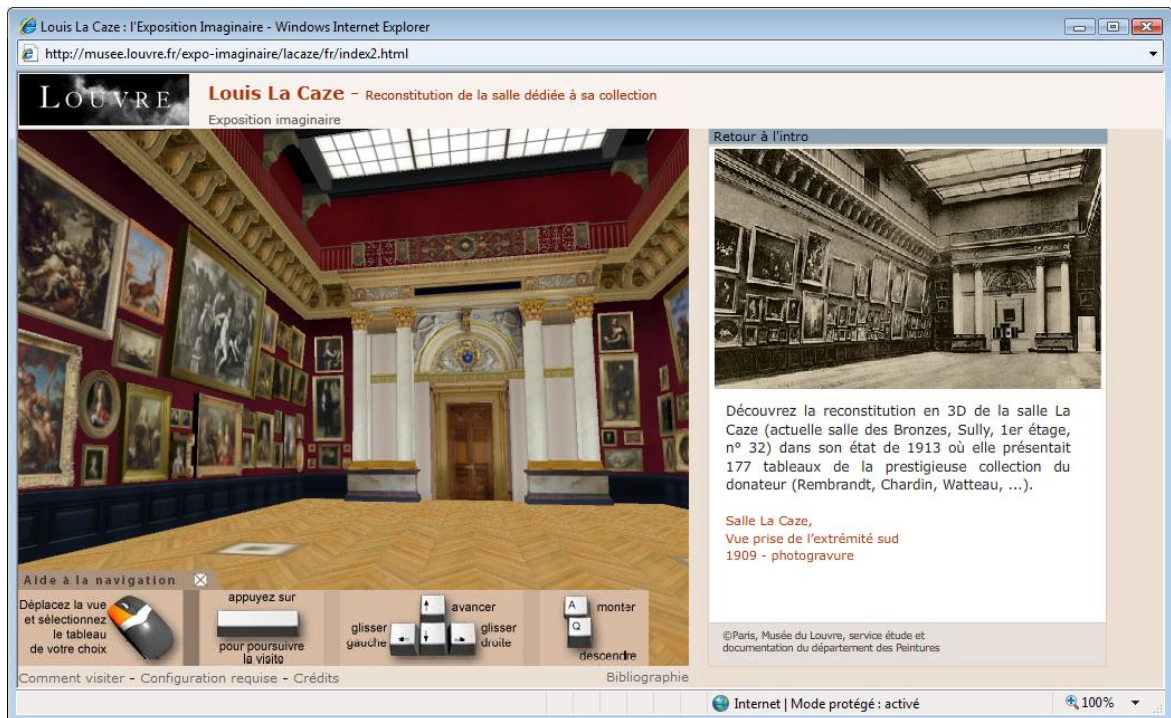
5- visualisation du site du Musée d'orsay



6- visualisation du site du Musée du Louvre

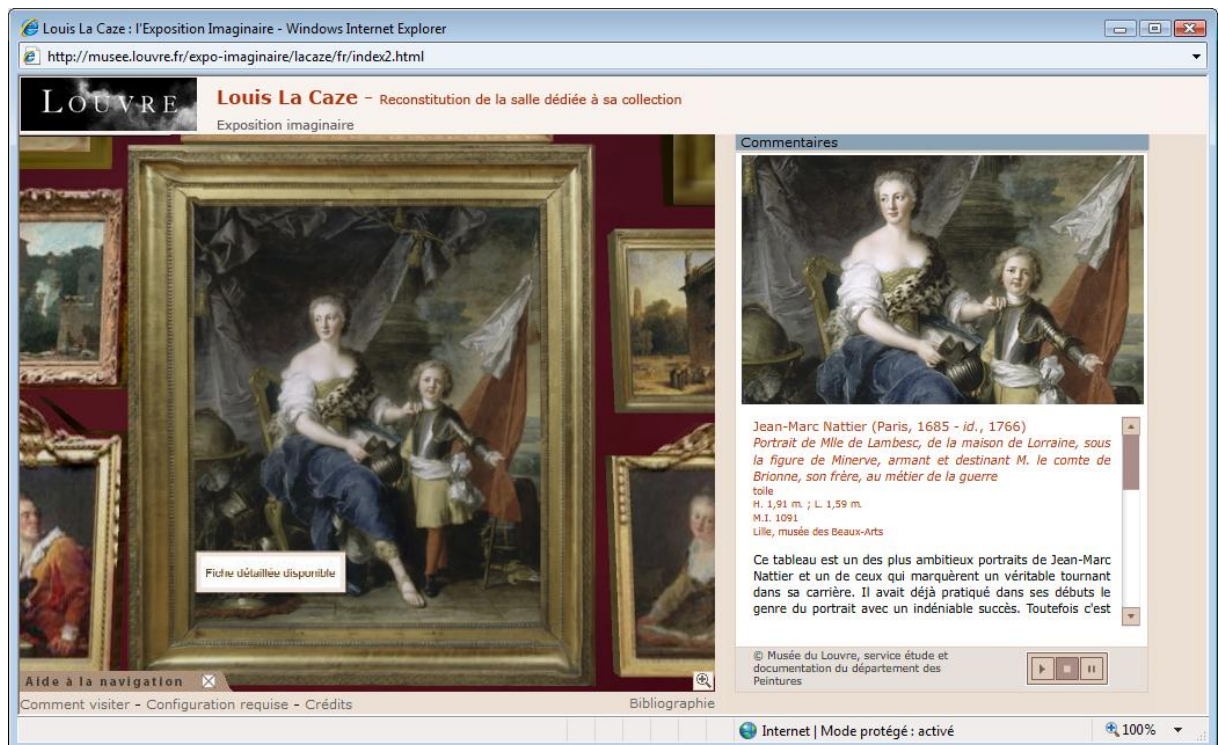
La photographie ci-dessous montre une vision de la visite virtuelle du Louvre (deux pièces et une église au choix), nécessite l'installation d'un composant supplémentaire pour le navigateur (un contrôle ActiveX). Le site du Louvre est un des sites les plus aboutis en termes de modélisation 3D. Toute l'interface du site est conviviale, fonctionnelle. La charte graphique donne du plaisir à découvrir ce musée et les œuvres dont il regorge.

Malgré ses qualités, il existe un autre site qui dépasse celui-ci technologiquement, Il s'agit d'OrganisMuséum que nous allons découvrir ensuite, un exemple de fiche détaillée d'une œuvre sélectionnée. Le principe est simple, lorsque l'on clique sur une œuvre sa fiche respective de l'œuvre apparait et nous permet d'obtenir un grand nombre d'informations avec un visuel plus précis que dans l'environnement 3D. En effet ce dernier sert à naviguer dans la salle et y découvrir les œuvres qui tapissent les différents murs.



7- visualisation du Musée virtuel du Musée du Louvre

Ci-dessous, une vision d'une fiche détaillée après avoir cliqué sur un tableau au hasard dans la salle :



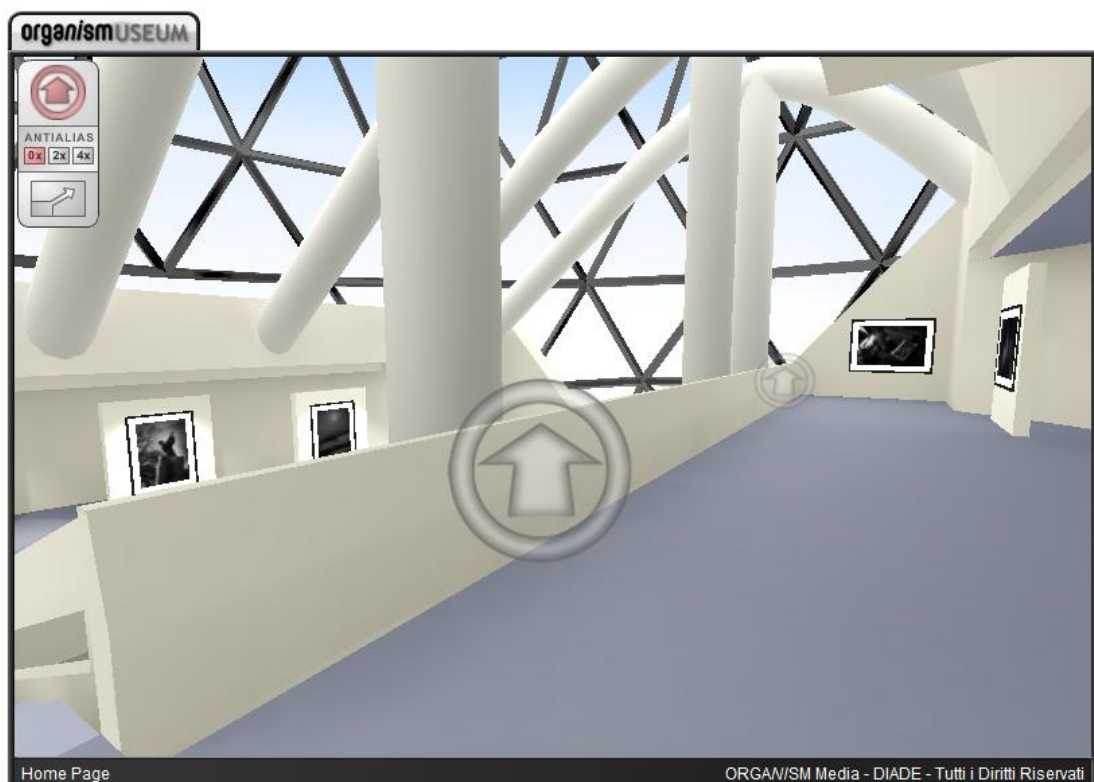
8- visualisation d'une fiche détaillée d'une oeuvre



Le Musée virtuel d'OrganisMuséum est une application qui permet de virtualiser un musée en créant un environnement et une navigation assez plaisante et à la fois réaliste.

Il s'agit d'une projection d'un musée qui se veut réaliste dans un environnement en trois dimensions. Chacune des pièces est modélisée et les œuvres sont projetées dans la scène. C'est ainsi que l'internaute ou le visiteur réalise une visite virtuelle. Il est possible de consulter les détails d'une œuvre (auteur, époque, type, ...). Aucune classification n'est disponible, il s'agit d'une visite libre comme dans bien des musées. Il est donc impossible de trier les œuvres par auteur, date ou autre données, c'est la principale limite de ce musée virtuel.

Pour le visiter, nous sommes obligés d'installer comme pour le Louvre un plugin ou composant.(nommé « 3DVIA Player »).



9- visualisation du site d'Organismuseum (vue 1)



10- visualisation du site d'Organismuseum (vue 2)

Après différentes recherches, le nombre de site Internet qui propose des visites virtuelles en 3D est faible contrairement à ceux en 2D. Seul certains grands musées comme le Louvre offre ce genre de découverte en 3D mais limitée aux pièces principales.

D'autres sites utilisent des plans dynamiques comme le Musée d'Orsay qui affiche pièce par pièce les œuvres respectives.

Les musées virtuels ne sont pas encore très présents sur la toile. Cependant l'Organismuseum donne un véritable aperçu de ce qu'il est possible de faire en termes de réalité virtuelle.

## ***4.2 Etude des technologies existantes***

En ce qui concerne les technologies utilisées pour le musée en 2D, elles sont traditionnelles. Effectivement, les technologies les plus couramment utilisées sont Flash (avec Action Script), MySql pour la gestion des bases de données et enfin la technologie « ajax » pour le confort de l'utilisateur et son interaction avec le site.

### **4.2.1 Les bases de la 3D (la modélisation) :**

En infographie, la modélisation 3D est la création, la manipulation et le stockage des primitives géométriques pour représenter des objets qui sont autour de nous ou des objets virtuels. Le processus de modélisation 3D utilise des logiciels de modélisation 3D. Le produit résultant est appelé le modèle 3D. Les modèles peuvent être créés automatiquement (procéduralement) ou manuellement.

De nos jours, les modèles 3D sont utilisés dans plusieurs domaines différents (médical, industriel,...).

Le cinéma par exemple a besoin de modèles 3D pour modéliser des personnages et des objets afin de permettre leur animation. C'est dans l'industrie du jeu vidéo que ces modèles sont primordiaux tout comme dans l'industrie architecturale pour modéliser des bâtiments, des arbres, des voitures et bien d'autres éléments.

La génération procédurale est la possibilité de créer des contenus médias à la volée, les fractales en sont des exemples classiques.

La complexité des modèles 3D se mesure par le nombre de polygones qui les définissent. La génération et le rendu progressif d'un objet dépend du niveau de détail ou LOD (level of détail). Le principe est simple, plus un objet est petit ou loin, plus sa définition est petite, donc moins de polygones sont utilisés. Plus on se rapproche de ce dernier, plus la résolution augmente et implique l'utilisation d'un nombre de polygones plus important. Plus un objet est détaillé, cela demande plus de ressource pour l'afficher.

## 4.2.2 Technologies hors navigateur (Open GL, DirectX,...)

### 4.2.2.1 La technologie « Open GL »

OpenGL (Open Graphics Library) est une spécification qui définit une API multiplateforme pour la conception d'applications générant des images 3D (mais également 2D). Elle utilise en interne les représentations de la géométrie projective afin d'éviter toute situation faisant intervenir des infinis.

L'interface regroupe environ 250 fonctions différentes qui peuvent être utilisées pour afficher des scènes tridimensionnelles complexes à partir de simples primitives géométriques. Du fait de son ouverture, de sa souplesse d'utilisation et de sa disponibilité sur toutes les plates-formes, elle est utilisée par la majorité des applications scientifiques, industrielles ou artistiques 3D et certaines applications 2D vectorielles. Cette bibliothèque est également utilisée dans l'industrie du jeu vidéo où elle est souvent en rivalité avec la bibliothèque de Microsoft : Direct3D. Une version nommée OpenGL ES a été conçue spécifiquement pour les applications embarquées (téléphones portables, agenda de poche, consoles de jeux...).

OpenGL est une évolution d'IrisGL, API 3D développé par SGI. Cette dernière étant difficile à faire évoluer et à étendre. Les spécifications et développements de base ont été réalisés par une équipe de SGI. Le projet Fahrenheit, initiative de Microsoft et de SGI, tenta d'unifier les interfaces OpenGL et Direct3D. Celui-ci apporta au début l'espoir de mettre de l'ordre dans le monde des API 3D, mais pour des contraintes financières de la part de SGI, le projet dut être abandonné.

La spécification OpenGL est actuellement surveillée par l'Architecture Review Board (ARB), formée en 1992. L'ARB se compose d'entreprises ayant un profond intérêt pour la création d'une API cohérente et largement disponible. Selon le site officiel d'OpenGL, 3DLabs, Apple, ATI, Dell, Evans & Sutherland, Hewlett-Packard, IBM,

Intel, Matrox, nVidia, SGI et Sun Microsystems font partie des membres votants (juin 2002). Microsoft, l'un des membres fondateurs, s'est retiré en mars 2003.

Les spécifications officielles d'OpenGL 3.0 ont été présentées le 11 août 2008. À l'origine, OpenGL 3 devait constituer un changement majeur, avec une refonte totale de l'API pour la rendre plus concurrentielle face à Direct3D. Il était notamment question d'abandonner les fonctionnalités obsolètes qui n'étaient, jusque-là, conservées que par souci de compatibilité avec les anciennes versions, l'utilisation d'un nouveau modèle objet plus cohérent qui encapsule notamment les états. Cependant, après un an de retard et une absence totale de communication de la part du groupe « Khronos », cette refonte totale a été abandonnée au profit d'une mise à jour incrémentale permettant simplement de supporter les dernières technologies graphiques<sup>3</sup>. Cette décision serait due à la volonté de rassurer certains marchés, par exemple en CAO, qui ne souhaitent pas voir disparaître ces anciennes fonctionnalités<sup>4</sup>. Malgré tout, les fonctions obsolètes seront marquées « *deprecated* » dans les nouvelles spécifications, et pourraient être supprimées dans les versions futures.

#### 4.2.2.2 La technologie « DirectX »

Microsoft DirectX est une collection de bibliothèques destinées à la programmation d'applications multimédia et le plus souvent pour les jeux et programmes nécessitant de la vidéo. À l'origine le nom de chacune de ces bibliothèques commençait par Direct, comme par exemple : Direct3D, DirectDraw, DirectMusic, DirectPlay, DirectSound. DirectX étant le nom générique faisant référence à l'ensemble de ces technologies. Avec le temps cette convention de nommage est un peu tombée en désuétude, le X prenant l'ascendant des technologies intitulées Xact, Xinput ont rejoint la grande famille des technologies DirectX. Ainsi lorsque Microsoft s'est lancé dans le développement d'une console faisant un usage intensif de ces technologies, le X était prédominant ce qui a pu conduire au nom Xbox (et par la suite Xbox 360).

Direct3D (la bibliothèque permettant de faire de la 3D temps réel) est largement utilisée dans le développement de jeux pour la plate-forme Microsoft Windows, pour Xbox et Xbox 360. Direct3D est aussi utilisé pour d'autres type d'applications s'appuyant sur des graphismes en 3D de haute qualité pour visualiser des données complexes, comme par exemple dans le secteur du CAD/CAM bien que son concurrent OpenGL soit dans ce domaine mieux implanté car existante sur bien plus de plates-formes. Direct3D étant peut-être une des technologies DirectX les plus reconnues, il n'est pas rare de voir le nom DirectX utilisé en lieu et place de Direct3D.

La multitude des composants DirectX existe sous deux formes. L'une uniquement destinée à faire tourner les applications développées en utilisant ces technologies (les *runtime components*), l'autre étant un kit de développement à l'usage des développeurs. Initialement les composants destinés à l'exécution des applications étaient redistribués avec les jeux qui en faisaient usage, aujourd'hui, ils se trouvent aussi préinstallés avec Windows. Le SDK est disponible en téléchargement gratuit à partir du site MSDN. La partie « runtime » est propriétaire et les sources ne sont pas accessibles.

Les dernières versions de Direct3D, à savoir Direct3D 10 et Direct3D 9Ex, sont l'apanage de Windows Vista. Ces deux versions font usage de fonctionnalités propres au nouveau modèle de pilote *Windows Display Driver Model* apparu avec Windows Vista. La nouvelle infrastructure graphique de Windows supporte la virtualisation du matériel graphique vis-à-vis de multiples applications et services. Avec Windows XP, les applications avaient un accès exclusif à la carte graphique et pouvaient s'accaparer les ressources de celle-ci.

## 4.2.3 Technologies intra-navigateur (VRML, 3DMLW)

### 4.2.3.1 La technologie « VRML »

Le Virtual Reality Modeling Language (ou VRML en abrégé) ou Virtual Reality Markup Language est un langage de description d'univers virtuels en 3 dimensions.

Ce langage interprété est une norme internationale ISO et les fichiers VRML ont habituellement pour extension « .wrl ».

En fait, c'est à proprement parler un langage de présentation et non de programmation, puisque comme pour le langage HTML par exemple, un fichier VRML ne contient généralement pas une suite d'instructions mais plutôt les informations permettant au visionneur d'afficher ensuite les éléments (formes, senseurs, lumières, etc.).

Présenté lors de la World Wide Web Conference de 1994, VRML n'est pas l'œuvre d'un unique programmeur, mais plutôt le résultat de la collaboration de plusieurs professionnels de la 3D, dont entre autres Mark Pesce, Tony Parisi, Gavin Bell (Silicon Graphics) et Paul Strauss (Silicon Graphics).

Le principal objectif de ce langage est de permettre la représentation d'univers interactifs 3D virtuels. Les fichiers « .wrl » sont des fichiers texte décrivant les scènes virtuelles à l'aide du langage VRML. Ces fichiers qui peuvent être stockés localement sur un ordinateur ou téléchargés depuis un serveur web, sont visualisés à l'aide d'un visionneur, qui est soit un plugin ajouté au navigateur web ou encore un logiciel autonome indépendant du navigateur web, qui est installé sur l'ordinateur de l'utilisateur.

Les programmes VRML peuvent décrire des formes simples (points, lignes, polygônes) ou complexes (sphères, cubes, cônes, cylindres...), du texte, des images, des animations, des éclairages, des sons, des hyperliens, ainsi que leur agencement dans l'espace, leur texture, leur couleur, leur matériaux...

Une fois le programme interprété par le visionneur, le monde virtuel s'affiche à l'écran, en 3D, la caméra (c'est-à-dire le point de vue) se positionne à l'endroit prévu de la scène, et l'utilisateur est alors libre de se déplacer dans ce monde virtuel (généralement à l'aide du clavier ou de la souris) et d'interagir avec les différents objets présents. Les "sensors" de proximité permettent de lancer une action lors du

passage à proximité d'un objet, les "sensors" de touché permettent de déclencher, par exemple, avec un clique sur l'objet l'ouverture d'un autre monde virtuel. En 1996, une nouvelle version du langage fut présentée : VRML 2.0 (rebaptisée ensuite VRML97). Parmi les améliorations par rapport à la version 1.0, on peut citer :

- L'animation des objets: les mondes créés ne sont plus condamnés à demeurer statiques,
- L'interaction avec les objets, en effet chaque objet du monde peut réagir à des signaux, ces signaux pouvant être générés par l'utilisateur (clavier, souris) ou par d'autres objets du même monde,
- La création de scripts d'animation (en javascript/ecmascript) inclus aux fichiers « .wrl »,
- La gestion des sons en 3D,
- L'extrusion d'objets,
- Les effets de brouillard,
- L'utilisation d'une séquence vidéo en tant que texture d'un objet ,
- création de prototypes d'objets.

En complément des informations de base concernant les différents objets de la scène 3D, le VRML v2 possède un véritable langage de programmation interne appelé VmrlScript, dont la syntaxe est similaire au JavaScript. Il permet notamment de manipuler les objets (nœuds) de la scène VRML (de type SFNode), ou d'autres types de données propres au VRML comme les SFTIME, SFCOLOR, ou encore SFRotation. Un script écrit en VmrlScript est exécuté à l'intérieur d'un objet (ou nœud) de type Script.

Ce langage, bien documenté sur le web, est ouvert et accessible à tous Il est souvent utilisé pour représenter des objets réels numérisés, créer des jeux vidéo 2D/3D, représenter des objets dans des documentations techniques (la NASA avait par exemple publié un fichier VRML qui permettait d'observer le module Mars Pathfinder dans ses moindres détails), des sites Internet de rencontre et de discussion, où chaque visiteur choisit un avatar pour le représenter, avant de



s'aventurer dans le monde virtuel à la rencontre des avatars des autres visiteurs, des simulations d'intérieurs (pour l'étude de l'agencement et de l'éclairage des pièces), ou bien encore des graphiques statistiques en 3D (cartographie, histogrammes 3D).

#### **4.2.3.2 La technologie 3DMLW**

3DMLW est une technologie Open Source pour la création et l'affichage de contenu en 3D et 2D sur internet à travers les navigateurs internet les plus communs.

Son domaine d'utilisation est la représentation de ville en 3D comme pour la technologie « CityGML ».

Ces technologies permettent néanmoins de mettre la 3D à la portée des téléphones mobiles.

Cette technologie se compose de cinq parties.

3DMLW est un langage de spécification pour la création de documents 3DMLW. 3DMLW ressemble à du XHTML et reste facile de compréhension,

Cette technologie permet le support de contenu dynamique et interactif,

Des feuilles de style peuvent être utilisées pour un design soigné et facile à réaliser,

Un complément est nécessaire pour les navigateurs internet qui pourront alors montrer les animations à partir de fichier au format 3DMLW,

Elle est également compatible avec les logiciels courant de création d'objet 3D comme 3DStudio, Blender, et autres.

Voici une liste non exhaustive d'extensions supportées, 3ds, obj, blend et an8 files.

#### 4.2.4 Technologies supportées par le Flash Player

Les technologies qui sont supportées par le lecteur Flash sont PV3D, Alternativa 3D, Sandra, Away 3D, ...

L'avantage de ces technologies est de n'avoir aucun plugin à installer dans le navigateur internet afin de visionner l'application réalisée. D'autant plus que tous les navigateurs actuels intègrent nativement le lecteur Flash comme Internet Explorer, Firefox, Opéra et autres...

#### 4.2.5 Comparatif des 4 principaux Moteurs 3D :

Il s'agit de montrer un comparatif de performance concernant les moteurs 3D (Alternativa3D, Away3D, Papervision3D et Sandy3D) à partir d'une étude réalisée. Il s'agit de comparer les temps de création d'objet (Cube, Sphère, Cône et Plane) en utilisant indépendamment chacun des moteurs.

Cependant, suite aux dernières mises à jour du Framework de Papervision3D, les problèmes de mémoire évoqués lors d'un autre comparatif n'ont plus lieu d'être.

#### Les sources :

- Alternativa3D\_ (le Framework n'est pas en open-source)
- Away3D
- Papervision3D
- Sandy3D

La méthode utilisée est l'implémentation au plus simple et identique pour chacun des moteurs, la même mécanique identique de création des objets, instance unique, une seule instance des « materials » (filare et texture). Ensuite on utilise les calculs asynchrones afin de ne pas avoir de valeurs faussés ( la création de l'objet est faite quand le processeur le permet .

Les tests sont effectués sur la création de 1000 objets, avec différentes variables (filaire ou bitmap, lissage de l'image ou non), avec un stockage des données dans un tableau. Les résultats utilisés sont les moyennes de temps de création et de mémoire utilisés par objet. En dernier lieu nous vérifions l'utilisation de la mémoire initiale, du pique de mémoire, ainsi que de la mémoire utilisée après les tests.

Les version des moteurs pour les tests sont respectivement Papervision3D ( Public Beta 2.0 - Great White (revision 804)), Away3D : 2.2.0, Sandy3D 3.0.2, Alternativa3D : 5.5.0. Comme on pourra le remarquer les résultats sont très disparates suivant les moteurs, et même si les objets n'ont pas les mêmes caractéristiques (le nombre de segments d'un cube ne peut pas être augmenté dans Away3D & Sandy3D, contrairement à Alternativa3D & Papervision3D), les différences sont sans équivoques.

#### 4.2.5.1 La taille des fichiers SWF

L'exportation du fichier « swf » et le poids de celui-ci (non négligeable au moment de l'accès au site, notamment pour les petites connexions) montre les premières différences entre les moteurs. Deux cas de figure sont testés, le premier cas importe les package de camera, vue et scène (d'initialisation) et le second cas importe en plus les packages qui regroupent les cubes, les plans, les cônes et les sphères).

I- Tableau comparatif de la taille de l'animation SWF créé pour les 4 moteurs 3D

	<i>Cas 1 : Implémentation du moteur</i>	<i>Cas 2 : Implémentation primitives</i>
1. Sandy3D	60ko	87ko
2. Alternativa3D	70ko	94ko
3. Papervision3D	83ko	95ko
4. Away3D	105ko	132ko

#### 4.2.5.2 L'Espace mémoire à l'initialisation du moteur :

Voyons maintenant l'espace mémoire pris lors de l'implémentation du moteur 3D. On teste la mémoire sans lancer aucune création d'objet, et les moteurs montrent déjà que certains sont plus optimisés et d'autres, beaucoup moins, voici les valeurs retenues pour chacun des Framework.

1. *Papervision3D* - 167 ko
2. *Away3D* - 199 ko
3. *Alternativa3D* - 202 ko
4. *Sandy3D* - 1 136 ko

Sandy3D possède un premier handicap important qui est l'espace mémoire important qu'il requière. La suite du comparatif montrera que Sandy3D a d'autres lacunes.

#### 4.2.5.3 La Mémoire Globale:

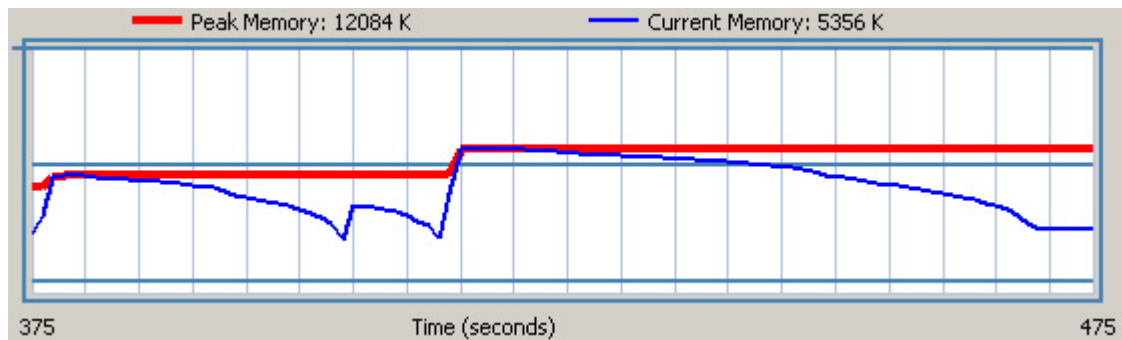
L'écart se creuse encore, Papervision3D et Sandy3D ont une très mauvaise gestion de la mémoire. En effet malgré la suppression de l'objet sur la scène et de son instance, il est toujours en mémoire. L'application monte donc en mémoire, sans jamais redescendre, ce qui peut s'avérer extrêmement dangereux... On peut noter qu'Away3D et Alternativa3D gèrent par contre très bien la mémoire. La suppression des éléments est bien gérée. Après la création de plusieurs milliers d'instance de plusieurs objets, on se retrouve avec une mémoire qui varie de manière non homogène selon les moteurs, nous pouvons remarquer que les différences sont parfois conséquentes.

1. *Away3D* - 700 ko
2. *Alternativa3D* - 876 ko
3. *Papervision3D* - 50 784 ko
4. *Sandy3D* - 175 552 ko

Les graphes ci-dessous montrent bien la gestion de la mémoire par les 4 moteurs 3D. On remarque que sur Alternativa3D, Away3D et PV3D, la mémoire évolue

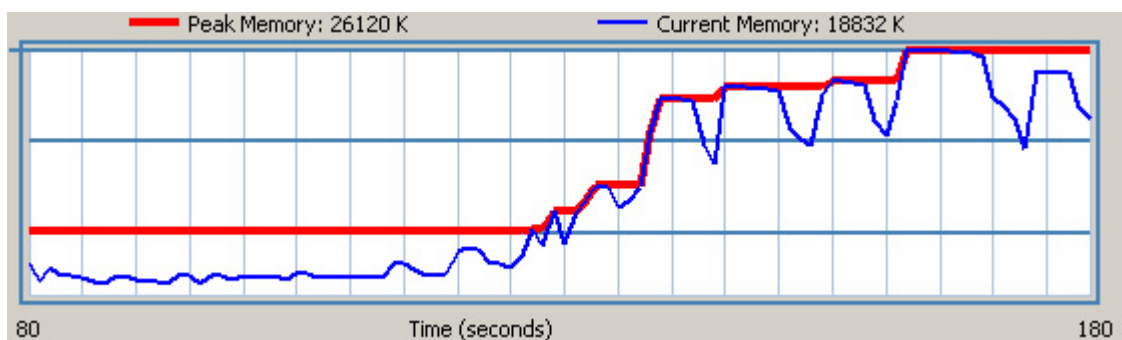
suivant les suppressions d'éléments et le lancement du « Garbage Collector », alors que pour et Sandy3D la mémoire ne cesse de monter.

### Alternativa3D



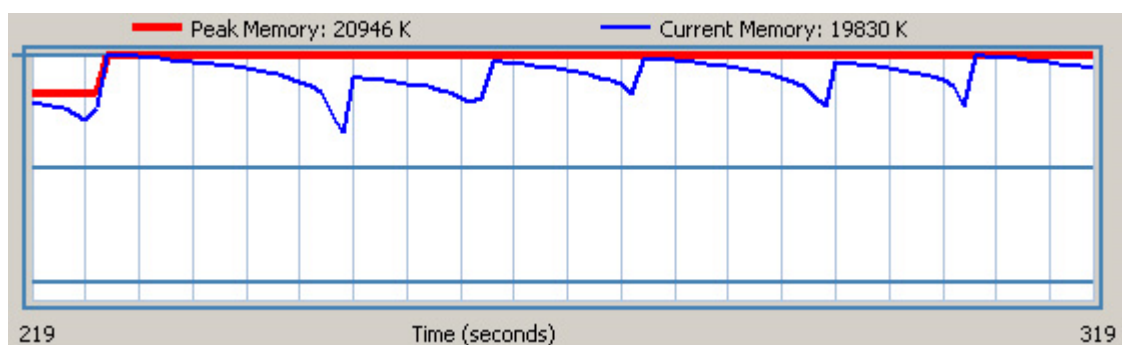
11- Visualisation de l'utilisation de la mémoire par Alternativa 3D

### Away3D



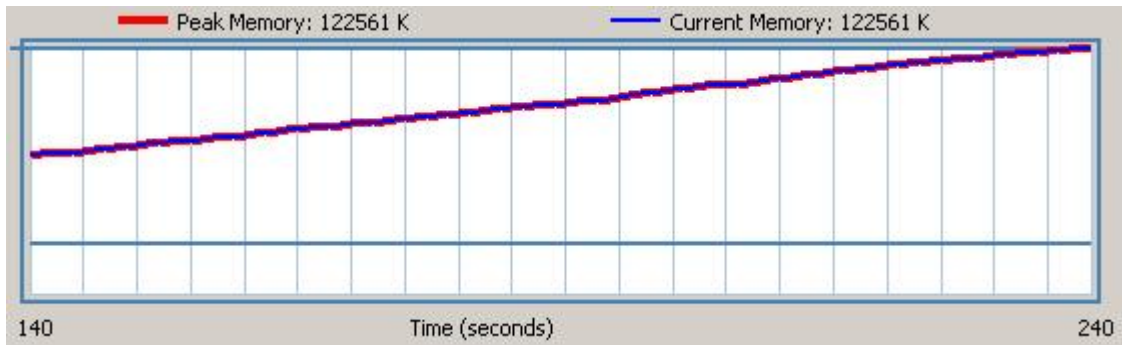
12- Visualisation de l'utilisation de la mémoire par Away 3D

### Papervision3D



13- Visualisation de l'utilisation de la mémoire par PV3D

### Sandy3D



14- Visualisation de l'utilisation de la mémoire par Sandy 3D

#### 4.2.5.4 La Création d'objet :

De par la logique, plus l'objet à créer est compliqué, plus il faudra de temps pour le créer, et plus la mémoire utilisée par cet objet sera importante. Comparer les temps de création et d'espace mémoire entre chacun des moteurs objets est assez compliqué. En effet, malgré des valeurs identiques, les moteurs créent des objets plus ou moins complexes.

II- Tableau comparatif de temps et de taille pour la création d'objet pour les 4 moteurs 3D :

Création d'un objet filaire				
	Box	Plane	Cône	Sphère
<b>Away3D</b>	0.814ms - 23ko	0.321ms - 9ko	1.265ms - 37ko	4.884ms - 138ko
<b>Alternativa3D</b>	5.865ms -	1.913ms -	5.723ms -	9.434ms -

	95ko	20ko	102ko	179ko
<b>Papervision3D</b>	4.961ms - 126ko	0.872ms - 23ko	2.996ms - 76ko	4.126ms - 84ko
<b>Sandy3D</b>	1.645ms - 16ko	0.267ms - 12ko	10.848ms - 222ko	15.299ms - 314ko
<u>Création d'un objet avec bitmap non lissé</u>				
	Box	Plane	Cône	Sphère
<b>Away3D</b>	1.292ms - 35ko	0.817ms - 23ko	1.67ms - 49ko	5.159ms - 153ko
<b>Alternativa3D</b>	5.909ms - 96ko	1.966ms - 21ko	5.733ms - 102ko	9.478ms - 178ko
<b>Papervision3D</b>	6.486ms - 169ko	1.801ms - 70ko	3.607ms - 121ko	5.31ms - 130ko
<b>Sandy3D</b>	1.856ms - 18ko	0.254ms - 12ko	11.389ms - 229ko	16.295ms - 330ko
<u>Création d'un objet avec bitmap lissé</u>				
	Box	Plane	Cône	Sphère
<b>Away3D</b>	1.319ms - 35ko	0.843ms - 23ko	1.722ms - 51ko	5.195ms - 152ko
<b>Alternativa3D</b>	5.811ms - 97ko	1.954ms - 21ko	5.764ms - 102ko	9.475ms - 182ko
<b>Papervision3D</b>	6.391ms - 169ko	1.837ms - 70ko	3.556ms - 121ko	5.259ms - 128ko
<b>Sandy3D</b>	1.747ms - 18ko	0.221ms - 12ko	11.16ms - 228ko	15.657ms - 330ko

Nous allons aborder la notion de délais pour comparer les Framework en termes de rapidité. De manière générale, le moteur Away3D est le plus rapide. Sandy3D a des performances équivalentes voire meilleures pour la création de Box ou de Plan, mais est très loin d'être aussi performant dans la création de Cône ou de Sphère. Une fois de plus celui-ci se démarque par sa singularité. Quel que soit le type de texture utilisé, Alternativa3D a des temps de création quasi similaires, il est moins performant lors de création d'objet filaire, mais cet handicap se réduit lors de la création d'objet "texturé". Les temps de création d'objet par Papervision3D se

situent dans la moyenne de celle d'Away3D et Alternativa3D, excepté pour les Box, ceci étant due aux nombres de segments créés par PV3D.

1. Away3D
2. Papervision3D
3. Alternativa3D
4. Sandy3D

#### 4.2.5.5 La Mémoire :

Comme pour les temps de création, Away3D est le moins gourmand en général. En effet, Sandy3D consomme moins de ressources lors de la création de Box ou de Plane, mais explose littéralement lors de la création de Cône et de Sphère. PV3D consomme beaucoup moins de ressource lors de la création de Sphère, mais utilise presque 6 fois plus de mémoire qu'Away3D lors de la création de Box (dépend du nombre de segments créés par Papervision3D).

Alternativa3D est quant à lui bien placé au vu des performances des autres moteurs. Voici le classement obtenu en fonction de la taille de la mémoire utilisée.

1. Away3D,
2. Papervision3D / Alternativa3D,
3. Sandy3D.

III- Comparatif des temps de création d'objet filaire pour les 4 moteurs 3D :

	Box	Plane	Cône	Sphère
Away3D	+/- 0.5ms	+/- 0.5ms	+/- 0.5ms	+/- 0.3ms
Alternativa3D	+/- 0.1ms	+/- 0.1ms	+/- 0.1ms	+/- 0.1ms
Papervision3D	+/- 1.5ms	+/- 1.0ms	+/- 0.7ms	+/- 1.2ms
Sandy3D	+/- 0.1ms	+/- 0.1ms	+/- 0.5ms	+/- 1.5ms

Ces écarts de temps de création sont à corréliser avec la mémoire utilisée par la création de cet objet.



IV- Comparatif détaillé de la mémoire utilisée lors de la création d'objets 3D pour les 4 moteurs 3D

	Box	Plane	Cône	Sphère
Away3D	+/- 12ko	+/- 14ko	+/- 13ko	+/- 14ko
Alternativa3D	+/- 1ko	+/- 1ko	+/- 1ko	+/- 3ko
Papervision3D	+/- 43ko	+/- 47ko	+/- 45ko	+/- 46ko
Sandy3D	+/- 2ko	+/- 2ko	+/- 6ko	+/- 15ko

On remarque que ces écarts sont quasi identiques pour chacun des objets créés par chaque moteurs 3D, Alternativa3D ayant une différence proche de zéro. On peut conclure que la gestion de texture filaire ou bitmap se fait de la même façon. Les différences de delta pour Sandy3D entre les objets Box/Plane et Cône/Sphère sont remarquables.

Les tests effectués ne sont peut-être pas représentatifs de la création d'une application, ou d'un site internet, mais ils nous montrent comment réagissent ces moteurs 3D dans des cas extrêmes, et si certains cas semblent dangereux (notamment concernant la gestion de la mémoire), il faut garder en mémoire que tout le monde n'a pas de machine surpuissante, et qu'il faut toujours penser à qui est destiné le développement que l'on fait.

Il faut donc choisir le plus judicieusement possible le moteur en fonction de ses possibilités ainsi que de ses performances.

Le 1<sup>er</sup> Alternativa3D est le dernier venu dans le monde des moteurs 3D, gère extrêmement bien (ou plutôt normalement) la mémoire, il est cependant plus gourmand en temps de création (entre 2 et 5 fois plus) que les autres moteurs. Malgré son coup de licence qui peut freiner. Les performances en font un des meilleurs moteurs 3D actuels.

1<sup>er</sup> ex aequo, Away3D possède des temps de création les plus bas du comparatif avec une utilisation de la mémoire suffisante à la création la plus basse excepté sur

quelques points (Box et Sphère), mais qui reste excellents. La gestion de la mémoire globale est excellente. Ce moteur 3D est à recommander en termes de performance sans pour autant avoir de bonnes possibilités de rendu graphique.

1<sup>er</sup> ex aequo également, PV3D possède des temps de création qui se situent dans la moyenne. C'est au niveau de la mémoire que la dernière version du Framework apporte une meilleure gestion de la mémoire. Il reste bien moins gourmand lors de la création de Box et de Plane qu'avec l'ancienne version. PV3D semblait avoir des problèmes pour supprimer ces instances, tout comme Sandy3D, mais qui ont été corrigés avec l'évolution de la dernière version du Framework.

En 2<sup>ème</sup> position, on retrouve Sandy3D qui n'est pas un modèle d'efficacité en termes d'utilisation de la mémoire, et cela, dès l'initialisation du moteur, et même si les performances lors de la création de Box et de Plan sont parmi les meilleures. La gestion de la mémoire (pas de suppressions des instances), les performances "catastrophiques" pour la création de cône et de sphère (donc d'éléments plutôt complexes), montrent les lacunes de la part de Sandy3D. Au vu des résultats, c'est le seul moteur qui ne réagit pas de façon homogène. Si on veut l'utiliser, il faut choisir des formes simples et en petite quantité.

Il est donc difficile d'attribuer une première place voire impossible. On peut dire qu'ils sont en moyenne aussi performants les uns des autres.

### **4.3 Bilan**

Après avoir étudié les sites existants en 2D et en 3D, nous avons pu nous rendre compte des possibilités offertes pour visualiser les œuvres, obtenir leurs caractéristiques pour chacune d'elles.

Contrairement aux musées en 2D, nous ne pouvons pas dans les musées 3D trier par genre ou catégorie les œuvres.

Parmi les différentes technologies utilisées en 2D nous choisirons le PHP pour tout ce qui concerne le dynamisme du site pour afficher les œuvres par catégories et sous-catégories ainsi que pour effectuer des recherches particulières.

En ce qui concerne les technologies 3D utilisées pour modéliser un musée, une semble sortir du lot et de ce fait être plus intéressante à choisir. Effectivement, le Framework « Papervision 3D » semble être tant au niveau performance qu'au nombre de support sur internet, le choix le plus judicieux à faire parmi les autres Framework.

## V. Le musée virtuel en 2D

Le Musée de Frontignan est classé Musée de France depuis l'an 2000 et de ce fait a des obligations quant aux détails de chacune des fiches descriptives de chacun des objets inventoriés.

En effet après s'être mis en relation avec la Direction des Musées de France (la DMF), nous avons pu établir une base commune de champs qui permettra d'être compatible avec la base JOCONDE qui recense tous les objets inventoriés de tous les musées classés « Musée de France ».

Ceci explique donc cette nécessité de refonte.

Après une étude des besoins pour améliorer l'outil informatique (« Gespimus » sur base Access) existant qui gère l'inventaire des objets des différentes collections du Musée de Frontignan. Un cahier des charges a été réalisé répondant aux nouvelles exigences.

Cette analyse a permis de créer un MCD qui va permettre d'aboutir à la création de la nouvelle base de données. Cette dernière permet la gestion des œuvres (objets) du musée pour l'inventaire d'une part et d'autre part, de changer les œuvres à afficher dans le Musée.

### 5.1. Charte graphique et charte éditoriale

La charte graphique sera établie en collaboration avec le service Communication et le webmaster lors de la mise en ligne réelle. Ce prototype se base sur une recherche graphique et iconographique. La mise en page a été conduite par les remarques et besoins établis lors des derniers rendez-vous avec la personne référent de ce projet. Les menus et sous menus sont le fruit d'une recherche et d'une logique d'accessibilité pour les internautes.

Ce prototype a pour but de montrer la navigation à la fois simple et ludique avec l'utilisation de menus et sous-menus.

Le graphisme devra mettre en valeur les éléments clés de chaque page et l'ensemble du site sera basé sur des feuilles de style CSS, de manière à pouvoir changer rapidement de style et pour pérenniser sa fraîcheur.

Cette charte graphique permettra de créer un modèle de présentation élégante déclinée à toutes les pages du site avec l'utilisation de couleur thématique pour une meilleure lisibilité et de respecter les règles fondamentales d'ergonomie.

La couleur F0A232 sera la couleur principale pour ce prototype à titre d'exemple.



15- Récapitulatif des couleurs du site internet officiel de la ville

## 5.2. Réalisation du Modèle Conceptuel de Données :

Après l'analyse effectuée au préalable, nous avons pu aboutir afin de modéliser les données pour la gestion des œuvres du musée. Cette base de données permettra notamment l'édition de l'inventaire du Musée, compatible avec la base Joconde, et de faciliter les recherches internes ainsi que sa gestion.

Un logiciel créé sous ACCESS appelé « GESPIMUS » servait jusqu'alors pour la gestion de l'inventaire. Hors ce dernier n'était pas si intuitif comme l'aurait voulu les utilisateurs. C'est pour pallier à ses défauts, que cette refonte est devenue nécessaire.

Après avoir établi les besoins, avec l'aide de ma référente, concernant l'inventaire des œuvres du Musée Municipal de la ville de Frontignan et en se basant sur l'existant, nous avons pu améliorer ce dernier.

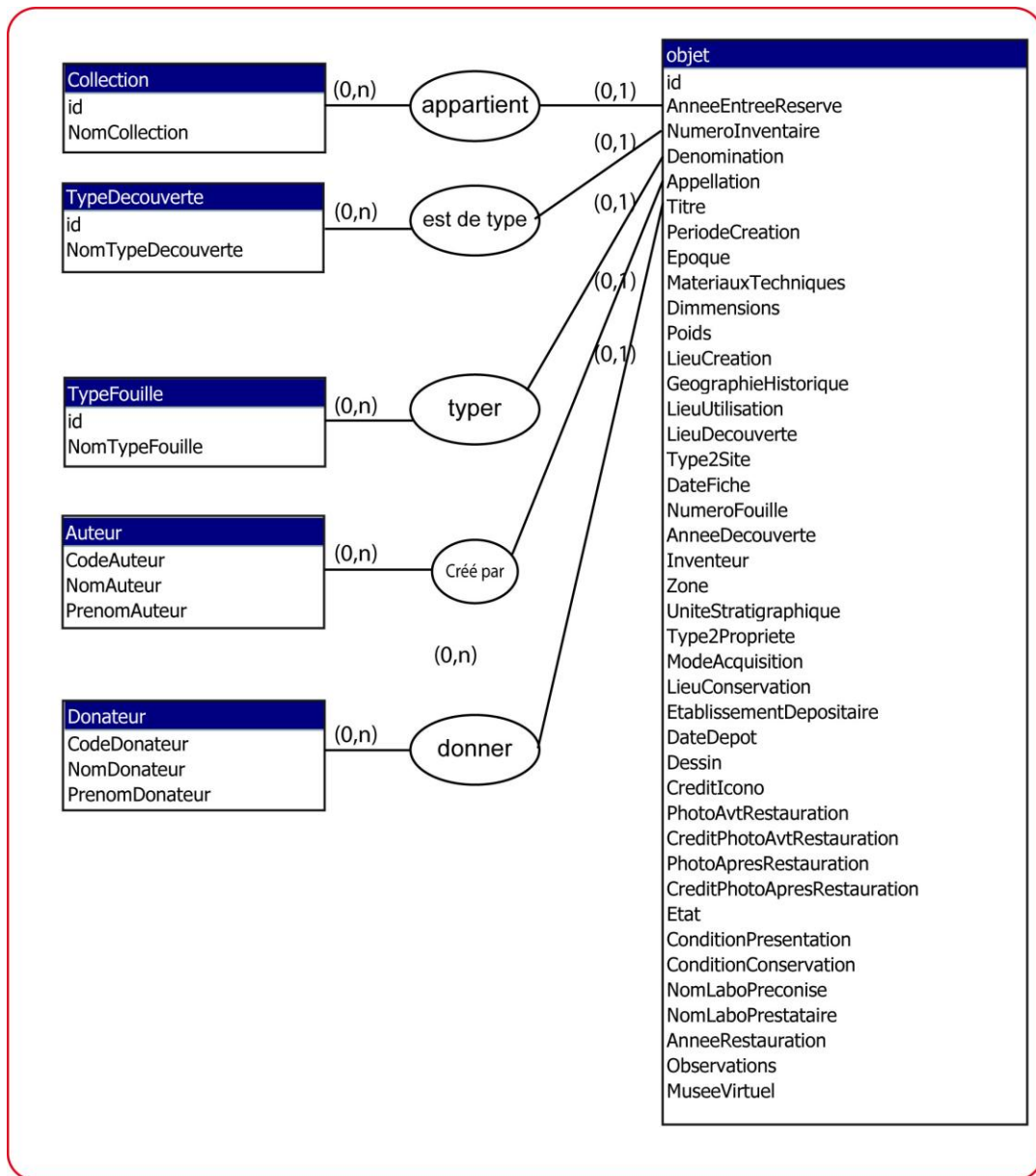
En utilisant Merise, nous avons abouti à un Modèle Conceptuel de Données (ou MCD) qui servira à implémenter la base de données définitive du musée pour la gestion de l'inventaire.

(Voir figure ci-contre.)

Afin d'optimiser le modèle physique et en tenant compte du fait que les informations concernant les « auteurs » et les « donateurs » peuvent être réduits à la première lettre du prénom, suivi d'un point, d'un espace, puis du NOM (ex : B. DUPONT), nous avons pu optimiser ce dernier en insérant les champs « Auteur » et « Donateur » dans la table « Objet ».

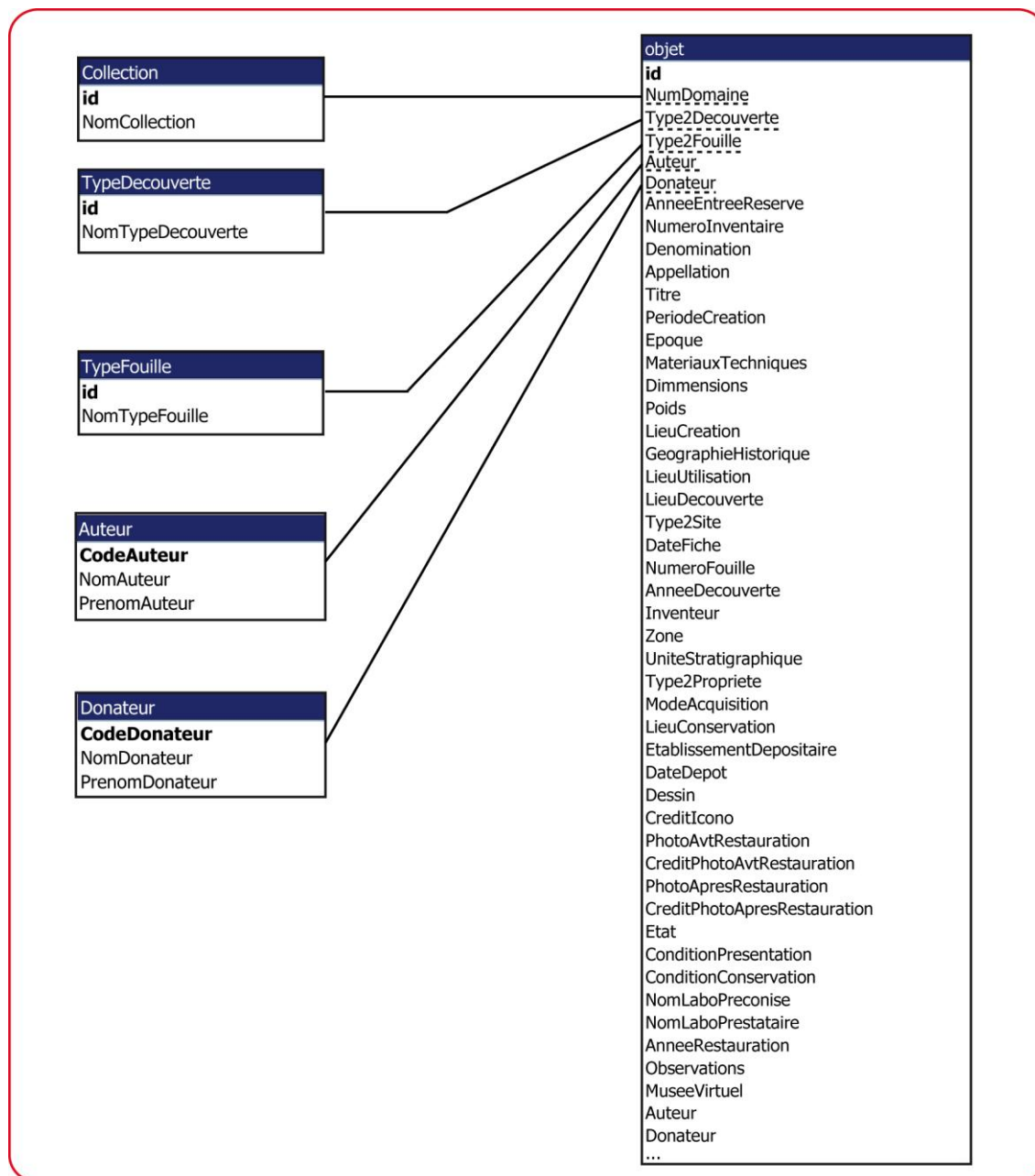
Au départ de l'identification des besoins, il a été convenu que nous devions récupérer davantage d'informations sur ces personnes, comme l'adresse, la ville, le numéro de téléphone, l'adresse du courrier électronique. Après réflexion de la part de Joanne MILLET, nous avons redéfini ces spécifications qui ont été appliquées au modèle physique de données.

La suppression des 2 tables précédemment citées a pour but d'optimiser les temps de requêtes sur la base en supprimant ces 2 jointures prévues initialement, d'autant qu'il n'y aura pas de recherche sur ces deux champs précités.



16-Modèle Conceptuel de données de l'inventaire (MCD)

Après dérivation du MCD, nous obtenons le MLD ci-dessous qui deviendra le MPD (Modèle Physique de Données), en tenant compte des informations concernant les tables « Auteur » et Donateur » désigné ci-dessus, qui disparaissent du modèle initial.



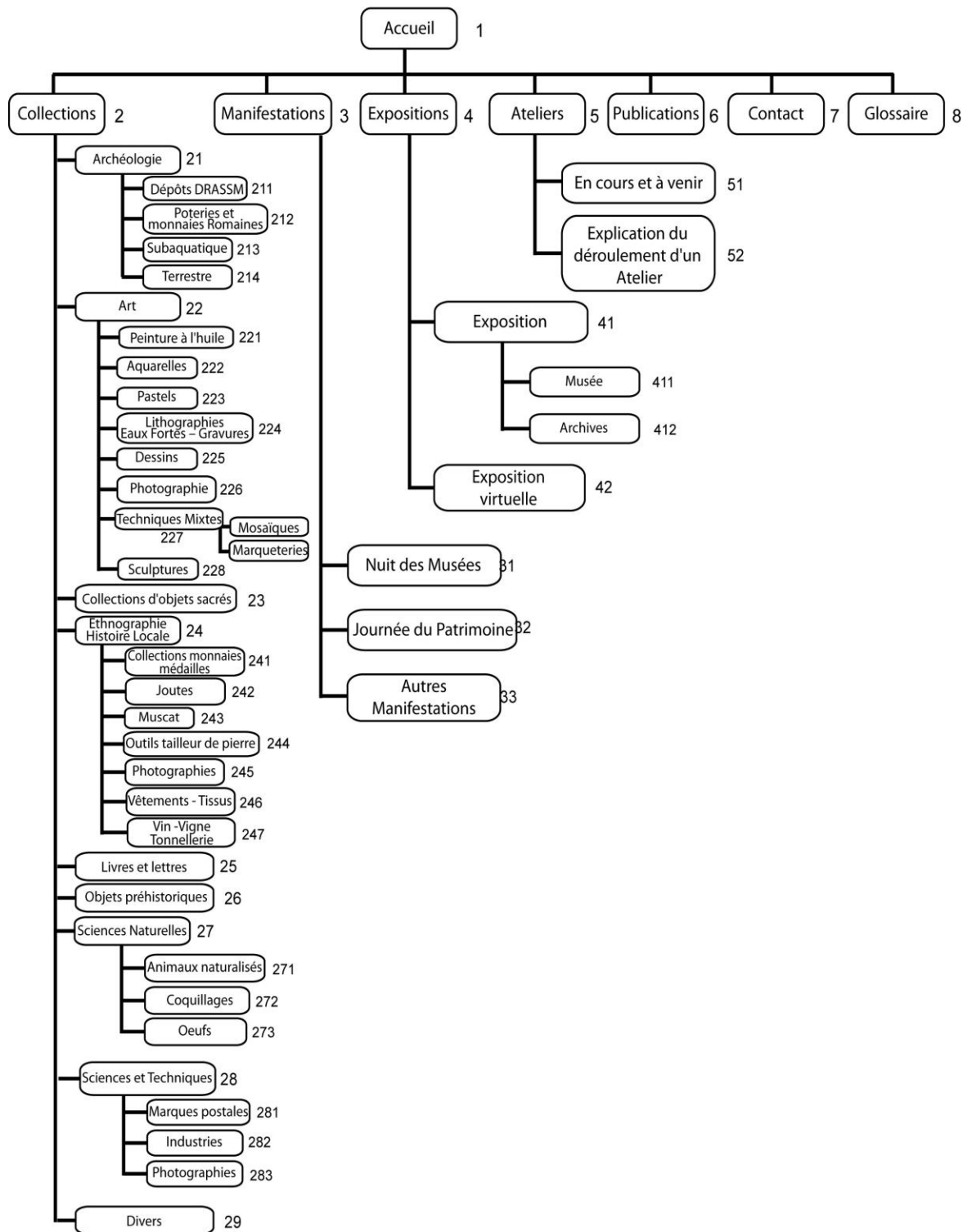
17-Modèle Logique de données de l'inventaire (MLD)

### 5.3 Conception :

Il s'agit de la réalisation d'un site internet traditionnel basé sur le modèle MVC de S. Tahé pour la constitution du prototype. Cependant, le choix technique final lors de la mise en ligne réelle sur le site internet de la ville de Frontignan dépend du choix réalisé, à savoir une intégration au site actuel ou un nouveau site totalement indépendant.



### 5.3.1 Etude de l'arborescence des pages du site :



### **5.3.2 Réalisation d'un musée virtuel en 2D :**

Réussir un projet de création de site Internet nécessite de fixer des objectifs, de cibler les besoins à partir d'un recueil des besoins et d'un état de l'art qui aura été réalisé au préalable.

Cette étape d'analyse a permis la réalisation d'un recueil des besoins avec l'aide de Mlle Joanne MILLET et de tout le personnel du service Patrimoine pour d'autres compléments d'informations.

Une fois le cahier des charges réalisé, il est nécessaire de répondre techniquement aux exigences qui y seront établies. Après validation de celui-ci, une maquette est établie comme base de travail. Les différentes modifications sont apportées pour permettre une intégration aisée dans le site de la ville actuel.

Cette réalisation s'effectue en deux temps, un premier pour réaliser la partie 2D (le catalogue) et un deuxième temps pour la modélisation du musée municipal en 3D, le musée virtuel.

La société « Pyxis » qui est la société qui gère le site de la ville sera en charge de me documenter techniquement pour permettre une intégration optimale au sein de l'organisation du site existant.

En ce qui concerne la conduite de projet, un comité de pilotage a été mis en place avec M.SALA et M. POMPIDOR qui seront mes tuteurs de stage et me suivront durant tout le projet.

### **5.3.3 La planification**

Comme évoqué dans un précédent chapitre, la planification respecte le découpage du projet en sept tâches principales visible sur le diagramme de Gant.

Il s'agit d'un cycle de vie en « V » pour la méthodologie utilisée pour ce projet.

Dans un premier temps, une analyse de l'existant et du recueil des besoins a permis d'établir le cahier des charges pour permettre la réalisation d'une maquette du site.

Il faut éventuellement penser à son éventuelle intégration dans le site actuel de la ville. Une interface d'administration permettra la gestion des œuvres. Cette étape durera trois mois environ.

Dans un deuxième temps, une étude des techniques existantes de modélisations 3D permettra de choisir celle qui répondra le mieux aux besoins qui seront établis au préalable et répondre aux besoins fonctionnels attendus pour la modélisation du musée virtuel.

Une fois le choix technologique déterminé, une phase d'initiation sera nécessaire pour appréhender ce domaine. Ensuite la modélisation sera mise en place étape par étape pour arriver au réalisme voulu et final.

Après avoir établi les fonctionnalités attendues avec de Joanne Millet, nous avons pu réaliser une maquette du site du musée en 2D. Lors d'une réunion inter élus, cette maquette a été appréciée et validée fonctionnellement. La réalisation concernant ce projet de création du site du musée municipal en mesurant tous les impacts financiers et humains que cela engendrerait, fût soumis aux personnes compétentes pour sa validation.

La maquette répond à tous les besoins que devait offrir le site en termes de fonctionnalités (attendues et définies).

L'aspect graphique n'est ici qu'une illustration, la charte graphique finale est à déterminer avec le service « Communication » de la ville le cas échéant.

Pour l'heure, ce projet reste en attente pour les raisons précédemment évoquées.

Voyons maintenant des exemples du prototype réalisé.



19- La page d'accueil du site du musée municipal



20- Exemple de menus permettant le tri des œuvres par collection  
(mode catalogue)

## VI. Les circuits touristiques

L'intérêt de ce projet est la création d'une solution de publication de circuits touristiques sur le site internet de la ville de Frontignan.

On peut citer par exemple « la balade piétonne » qui est actuellement sur support papier. On peut très bien imaginer une version numérique mise à disposition sur le site internet de la ville.

Il existe 2 méthodes de travail, l'une utilisant Google Earth comme outil pour créer les points clés du parcours, l'autre en utilisant les coordonnées connues de chacun des points clés. En association avec l'outil Google Maps, ces données sont ensuite représentées sur une carte qui identifie par des marques l'ensemble des points clés du parcours.

### 6.1 Google Earth :

Le choix de la version de l'outil Google Earth se portera sur la version gratuite. Cet outil fonctionne sous différents systèmes comme Windows (ses différentes versions), Linux ou Macintosh. Cet outil permet de créer facilement des parcours en accédant à n'importe quel point du monde, en visualisant des images satellites qui fournissent une réelle précision. On peut en outre naviguer dans différentes villes.

### 6.2 Google Maps :

Comme Google Earth, Google Maps est un outil gratuit qui permet d'afficher et de manipuler des cartes géographiques à travers un navigateur internet comme Internet explorer ou Firefox et bien d'autres....

Ce service est également accessible depuis les téléphones mobiles et offres de nombreuses fonctionnalités comme la localisation d'une adresse, le calcul d'itinéraire entre 2 lieux, de gérer ces propres cartes (ou parcours), le partage avec d'autres internautes de ses propres cartes et vice versa.

Ces 2 outils sont compatibles et interagissent entre eux de façon aisée.

### **6.3 Procédure de création d'un circuit touristique :**

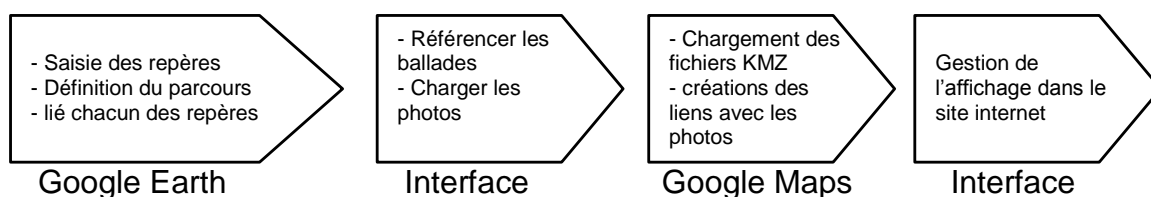
Pour permettre la création de circuit nous allons aborder deux manières de faire, l'une utilisant Google Earth pour créer les points, l'autre en utilisant une base de données (on supposera ici de connaître les coordonnées géodésique).

#### **1<sup>ère</sup> solution :**

Nous utiliserons l'application Google Earth dans un premier temps pour créer les repères de la ballade qui définiront le parcours, une fois tous liés entre eux. Ce parcours est ensuite enregistré dans un fichier d'export, un fichier avec une extension « .KMZ ».

Dans un second temps nous utiliserons l'application Google Maps qui utilisera ce fichier pour placer les repères sur une carte et ainsi y représenter le parcours du circuit touristique. Si plusieurs parcours sont nécessaires, il y a aura autant de fichier KMZ qu'il existe de parcours.

Google Maps se charge d'archiver le parcours et un lien permettra d'accéder à ce dernier depuis le site internet de la ville.



21- Schéma des étapes pour réaliser un parcours (solution N°1):

Avantages :

Cette solution ne nécessite pas un développement informatique particulier et conséquent. Il en est de même sur le plan financier et celui des ressources et des délais.

Une journée suffit pour mettre en place un parcours.

Inconvénients :

Cette phase de saisie n'est pas forcément agréable, elle reste fastidieuse, c'est un des inconvénients de cette solution. Cela nécessite d'utiliser deux applications Google Earth et Google Maps.

### **2<sup>ème</sup> solution :**

Nous partons du principe que nous connaissons les coordonnées géographiques des points clés du parcours à réaliser. Nous utiliserons une base de données qui stockera les infos de chacun des points pour générer automatiquement le parcours sur la carte de Google Maps. Un script PHP s'occupera de traiter chacun des enregistrements de la base pour représenter chacun d'eux sur une carte à l'aide de « marqueurs » qui permettront également d'afficher plus de détails sur le lieu dit.

En effet lors du clic sur un marqueur, une photo apparaît sur un premier onglet, et un texte descriptif sur un second onglet.

L'intérêt de cette technique permet de ne pas surcharger la carte en informations et laisse l'internaute le choix de cliquer sur les marqueurs souhaités pour obtenir plus d'informations.

Avantages :

Cette solution ne nécessite pas un développement informatique particulier tant au niveau financier, des délais et des ressources.

Une journée suffit pour mettre en place un parcours.

Un des avantages est la numérotation des repères possibles.

L'utilisation d'une base de données combinée à l'outil Google Maps apporte une meilleure personnalisation des informations à afficher.

Inconvénients :

Cette solution nécessite de connaître les coordonnées géographiques des points à déterminer. On peut très bien utiliser une application mobile pour permettre de

connaître les dites coordonnées, pour les intégrer par la suite dans une base de données.



## 22- Schéma des étapes pour réaliser un parcours (solution N°2) :

Nous pouvons découvrir maintenant un exemple d'une représentation du parcours avec Google Maps et ainsi visualiser l'ordre défini de la visite.



## 23- Exemple de visualisation d'un parcours touristique

L'inertation des liens de droite permet de mettre le focus sur la sélection est d'accéder aux informations disponibles.

L'API Google Maps met à disposition d'autres fonctionnalités comme le zoom avant arrière, de revenir à la vision originale, d'afficher le plan seulement, une vue aérienne, ou bien un version mixte (plan + aérien), d'autre fonctionnalités peuvent être ajoutées à cette carte.



L'exemple suivant affiche la carte après avoir cliqué sur le lien « Mairie » situé sur la droite, voir la photo ci-dessous :



24- Visualisation des informations d'un marqueur (onglet titre)



25- Visualisation des informations d'un marqueur (onglet descriptif)

## VII. Le musée virtuel en 3D

### 7.1 *Etude du musée en 3D*

#### 7.1.1 Introduction

L'intérêt de ce projet de création d'un Musée Virtuel pour la ville de Frontignan va permettre de créer de manière générique ce dernier à partir d'une spécification XML. Ce travail permettra de l'étendre à d'autres projets similaires pour la représentation de salles en 3D. La réutilisation du code permettra la création d'autre musée ou autre environnement en 3D.

Après avoir établi les fonctionnalités attendues avec la responsable, nous avons pu réaliser une maquette du site du musée en 3D en y intégrant les fonctions demandées (navigation, affichage de la fiche d'une œuvre).

Avec l'utilisation d'un module en amont (en collaboration avec M.POMPIDOR), qui permettra de positionner les murs, d'affecter les œuvres pour chacun d'eux et en dernier lieu de générer un fichier XML de cette description, on pourra représenter le musée, associé à ce fichier, en trois dimensions et ainsi visionner le musée virtuel. Nous avons la possibilité de créer en parallèle ce fichier manuellement pour avoir une même finalité. Le fichier XML est le point clé pour la création du musée virtuel. C'est à partir de ce dernier que l'application va générer les murs, y placer les œuvres respectives (aux positions X et Y) présentes sur ce fichier en précisant également leur largeur et hauteur respectives. Ainsi les salles dessinées seront représentées sur la scène.

### **7.1.2 Modélisation des fonctionnalités attendues (uses cases)**

Les fonctionnalités attendues sont de :

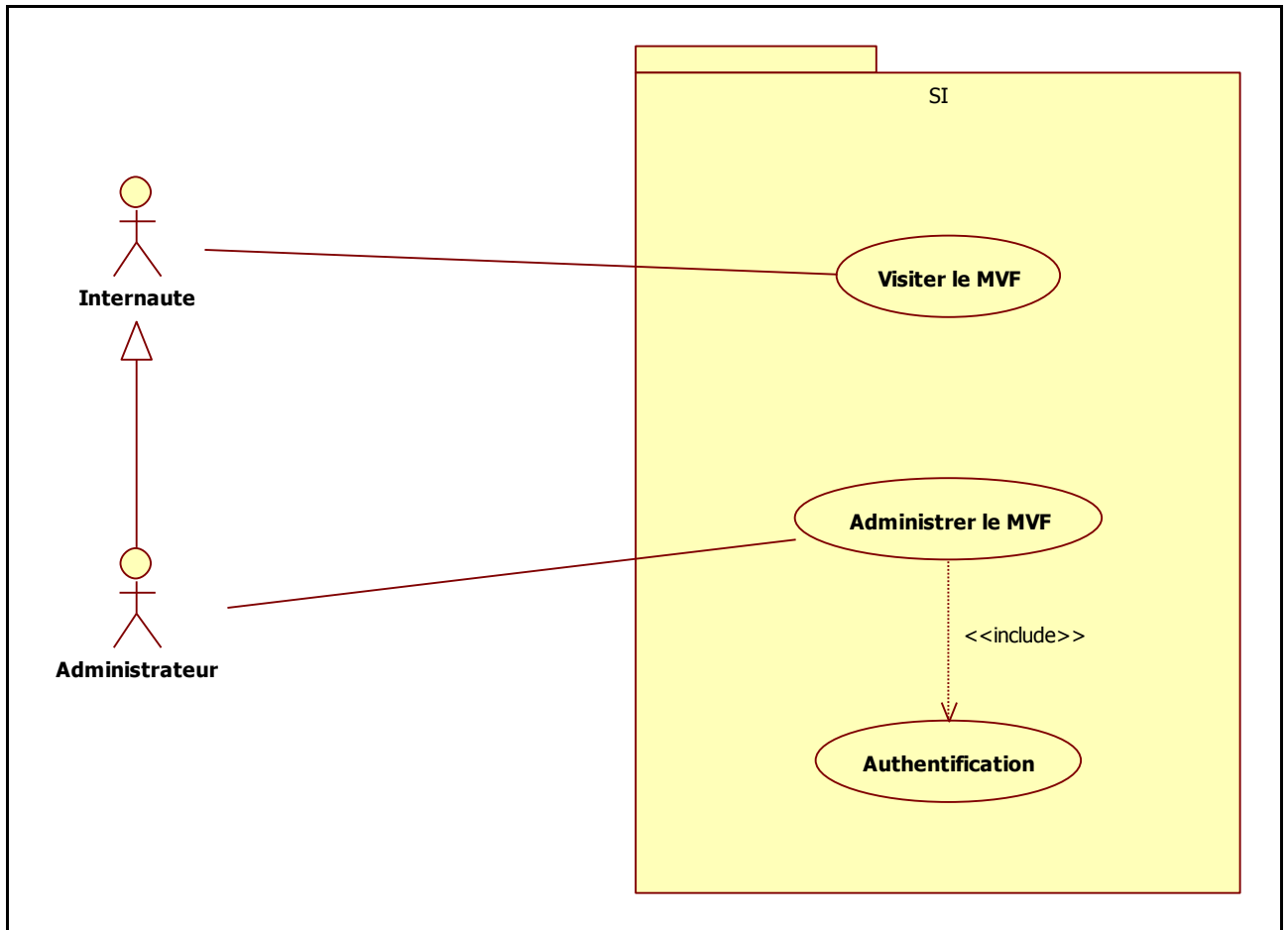
#### **Visiter le MVF : (Use Case 1)**

- Visualiser la scène,
- Positionner le sol dans la scène,
- Positionner les murs dans la scène,
- Placer les œuvres sur leurs murs respectifs,
- Permettre la navigation à l'intérieur de(s) salle(s) en utilisant les flèches du clavier (avant, arrière, gauche, droite) et tourner avec la souris depuis sa position,
- Afficher le descriptif de chacune des œuvres sélectionnées individuellement avec toutes leurs spécificités affichées.

#### **Administrer le MVF : (Use Case 2)**

- Xml → MySQL
- MySQL → Xml
- Création de l'inventaire (fichier XML)
- Modifier les tableaux affichés dans le MVF

**Use Case Général :**



26-Cas général des Uses Cases

V- Use Case pour la visite du musée virtuel :

Use Case 1 : Visiter le MVF

Cas d'Utilisation	<b>Visiter le Musée Virtuel</b>	
Acteur primaire	Internaute	
Système	SI	
Opérations		
	1	Naviguer dans le MVF (avant, arrière, gauche, droite, tourner)
	2	Afficher le détail d'une œuvre par un clic sur l'objet voulu
Extensions		
	1A	Impossible d'accéder au MVF
	1A.1	Erreur

XI- Use Case pour administrer le musée virtuel :

### Use Case 2: Administrer le MVF

Cas d'Utilisation		<b>Administrer le Musée Virtuel</b>
Acteur primaire		administrateur
Système		SI
Opérations		
	1	XML → MySQL (copie en base de données)
	2	MySQL → XML (export XML, MAJ fichier XML)
	3	Créer l'inventaire en fichier XML
	4	Modifier les œuvres affichées
	5	Créer un nouveau MVF (option avec un module supplémentaire)
Extensions		
	1A	Authentification impossible
	1A.1	Echec
	2A	Connexion à MySQL impossible
	2A.1	Erreur de connexion

## 7.1.3 Mise en application

### 7.1.3.1 Description du projet

Les choix techniques utilisés pour la réalisation de ce projet sont PHP pour les parties de présentation et d'administration, puis AS 3.0, Flash et PV3D pour la conception et MySQL comme base de données.

Pour des raisons de vitesse, cette application sera conçue pour le Flash Player 10 qui contribue de façon plus efficace que les versions antérieures, à améliorer le rendu 3D et notamment la fluidité de navigation. En effet ce dernier gère plus efficacement l'accélération graphique.

Les fichiers AS 3.0 une fois compilés sont transformés en un fichier Flash (.swf) qui contient toute l'application.

L'application du musée est de type générique. Elle se base sur un fichier XML qui permet à partir de ce dernier d'afficher le rendu désiré, à savoir le nombre de salle

défini, le nombre de mur par salle, le nombre d'œuvre respectivement associé à chacun des murs.

Les noms des balises et leurs attributs ont été spécifiés comme modèle de base.

Voici la structure du fichier XML :

```
- <MONDE>
- <salle numero="0">
- <murs hauteur="50">
- <mur numero="1" texture="" couleur="">
  <segment x1="5.00" y1="4.00" x2="9.00" y2="2.00" />
- <oeuvres>
  <oeuvre id="101" x="8.00" y="13.00" largeur="100" hauteur="100" />
  <oeuvre id="102" x="20.00" y="13.00" largeur="50" hauteur="100" />
  </oeuvres>
  </mur>
- <mur numero="2">
  <segment x1="9.00" y1="2.00" x2="15.00" y2="2.00" />
- <oeuvres>
  <oeuvre id="103" x="8.00" y="13.00" largeur="100" hauteur="100" />
  <oeuvre id="104" x="20.00" y="13.00" largeur="50" hauteur="100" />
  </oeuvres>
  </mur>
- <mur numero="3">
  <segment x1="15.00" y1="2.00" x2="16.00" y2="6.00" />
- <oeuvres>
  <oeuvre id="105" x="8.00" y="13.00" largeur="100" hauteur="100" />
  <oeuvre id="106" x="20.00" y="13.00" largeur="50" hauteur="100" />
  </oeuvres>
  </mur>
- <mur numero="4">
  <segment x1="16.00" y1="6.00" x2="16.00" y2="11.00" />
- <oeuvres>
  <oeuvre id="107" x="8.00" y="13.00" largeur="100" hauteur="100" />
  <oeuvre id="108" x="20.00" y="13.00" largeur="50" hauteur="100" />
  </oeuvres>
  </mur>
- <mur numero="5">
  <segment x1="16.00" y1="11.00" x2="12.00" y2="13.00" />
- <oeuvres>
  <oeuvre id="109" x="8.00" y="13.00" largeur="100" hauteur="100" />
  <oeuvre id="110" x="20.00" y="13.00" largeur="50" hauteur="100" />
  </oeuvres>
  </mur>
- <mur numero="6">
  <segment x1="12.00" y1="13.00" x2="8.00" y2="13.00" />
- <oeuvres>
```

```
<oeuvre id="111" x="8.00" y="13.00" largeur="100" hauteur="100" />
<oeuvre id="112" x="20.00" y="13.00" largeur="50" hauteur="100" />
</oeuvres>
</mur>
- <mur numero="7">
  <segment x1="8.00" y1="13.00" x2="5.00" y2="11.00" />
- <oeuvres>
  <oeuvre id="113" x="8.00" y="13.00" largeur="100" hauteur="100" />
  <oeuvre id="114" x="20.00" y="13.00" largeur="50" hauteur="100" />
</oeuvres>
</mur>
- <mur numero="8">
  <segment x1="5.00" y1="11.00" x2="5.00" y2="4.00" />
- <oeuvres>
  <oeuvre id="115" x="8.00" y="13.00" largeur="100" hauteur="100" />
  <oeuvre id="116" x="20.00" y="13.00" largeur="50" hauteur="100" />
</oeuvres>
</mur>
</murs>
</salle>
</MONDE>
```

On peut très bien envisager d'ajouter d'autres attributs aux balises pour apporter encore plus de précisions en ce qui concerne le projet. Il s'agit ici de montrer les possibilités de Papervision3D pour la virtualisation de salles.

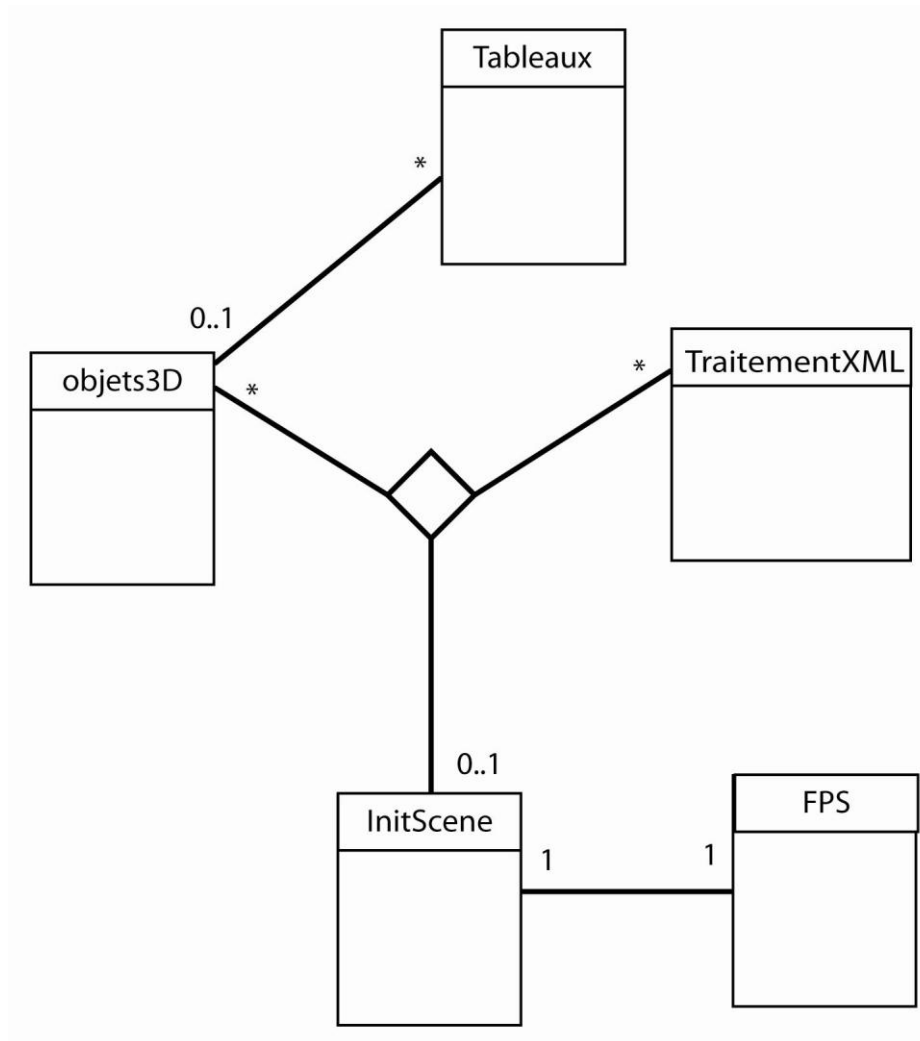
On peut notamment penser pouvoir ajouter des objets modélisés à partir de logiciels spécifiques tel que Studio 3D ou Blender. On pourrait tout à fait intégrer au musée un statue ou atout autre objets modélisé (un pavois, etc..).

### 7.1.3.2 Partie création

Après avoir étudié les rudiments pour utiliser le Framework PV3D, voici l'étape d'adaptation à notre projet de musée virtuel. Nous avons choisi d'utiliser un modèle MVC (Modèle Vues Contrôleur) traditionnel. Le projet sera construit autour de différentes classes qui inter agiront entre elles et notamment celle nommée « Client.as » qui représente la classe communément appelée la « classe contrôleur ».

Plusieurs classes sont nécessaires pour la réalisation du projet, « Client.as », « Objets3D.as », « TraitementXML.as », « InitScene.as », « Tableaux.as », « FPS.as ».

### 7.1.3.3 Diagramme de classes :



35- Diagramme de classes

La classe « InitScene » sert à créer l'environnement minimal en utilisant les classes de base pour mettre en œuvre le Framework PV3D. Une fois cet environnement créé nous pouvons à travers cette agrégation, parser le fichier XML à l'aide de la classe « TraitementXML » et créer les objets en utilisant la classe Objets3D, qui servent à mettre en place les éléments du musée virtuel en 3D (un prototype). Une



fois les éléments placés comme le sol, nous pouvons positionner les murs qui eux même utiliseront la classe « Tableau » pour y placer les leurs œuvres respectives.

La classe « FPS » sert pour afficher sur la scène le nombre d'image par seconde qui indique la fréquence des images instantanées et ainsi se rendre compte de la fluidité de l'application qui reste un des points clés pour réussir une application en 3D.

#### **7.1.3.4 Positionnement des tableaux sur les murs :**

La technique choisie pour positionner les tableaux sur les murs, est d'affecter les tableaux au mur lors de leur création.

Lorsque nous ajoutons un objet à un autre, son positionnement se fait automatiquement au milieu de l'objet parent. C'est alors que les coordonnées contenues dans le fichier XML seront utilisées pour les positionnées comme spécifié dans ce fichier.

Pour donner une idée de relief, nous avons utilisé des cubes en affectant les photos des œuvres respectives sur les deux faces verticales. Ceci permet de mettre les photos des œuvres en avant des murs porteurs dans le but de donner une dimension.

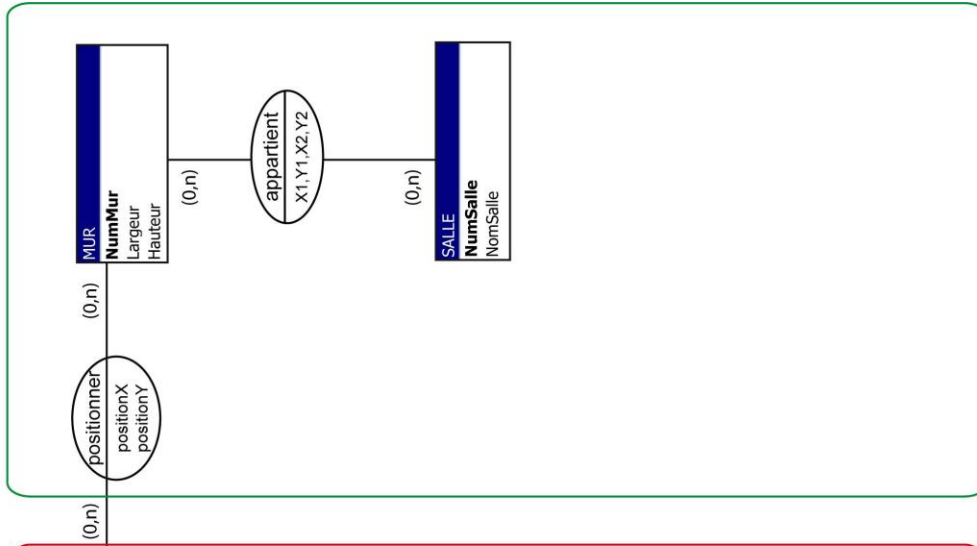
#### **7.1.3.5 Partie Administration**

Une partie administration est prévue pour permettre de modifier les œuvres présentes sur les murs et ainsi contribuer à faciliter l'actualisation du musée et de ses œuvres.

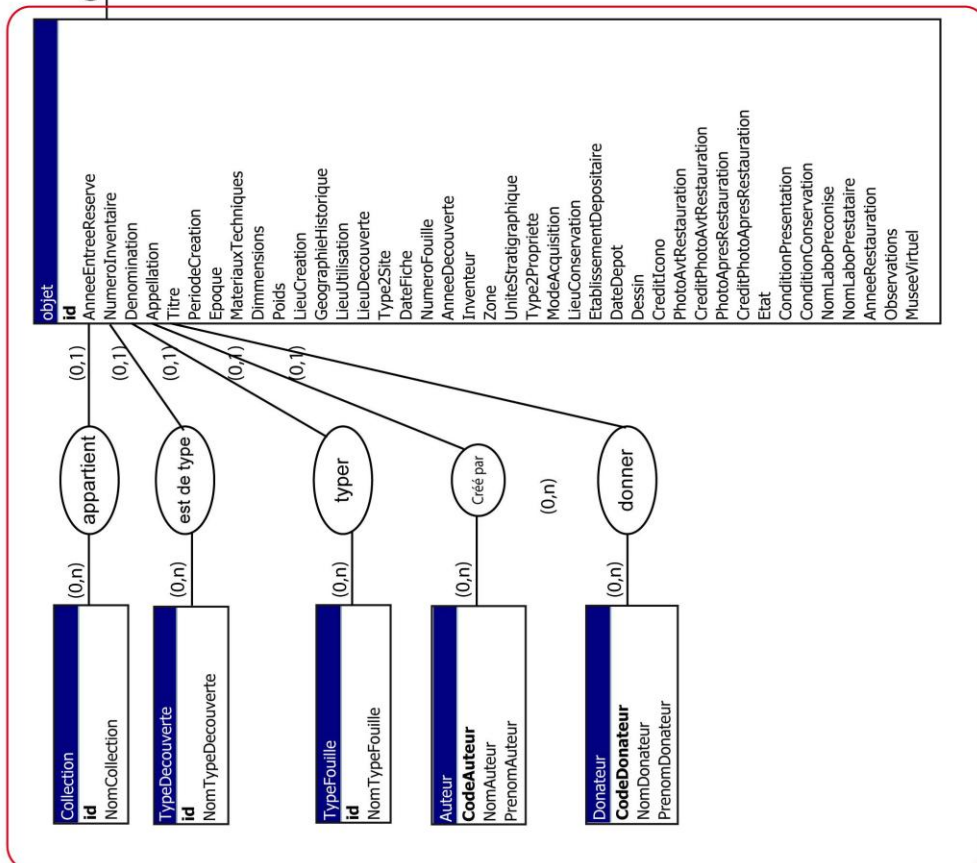
Il est donc nécessaire de modifier le fichier XML qui sert de base pour la construction du musée et maintenir une corrélation entre la base de données qui enregistre les modifications éventuelles à apporter et ce fichier clé.

Voici le modèle Conceptuel de Données qui permet de gérer l'administration du musée virtuel en corrélation avec la base de données de l'inventaire du Musée.

Pour le Musée Virtuel



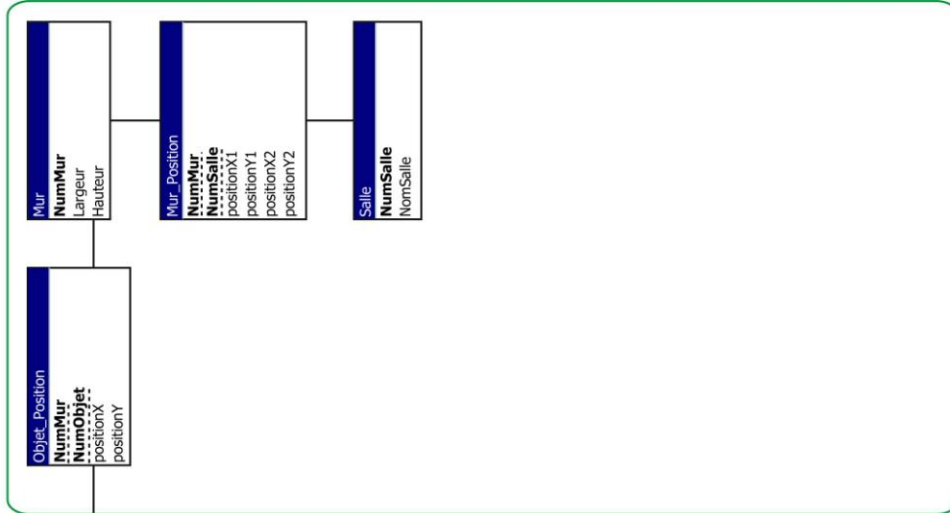
Pour l'inventaire



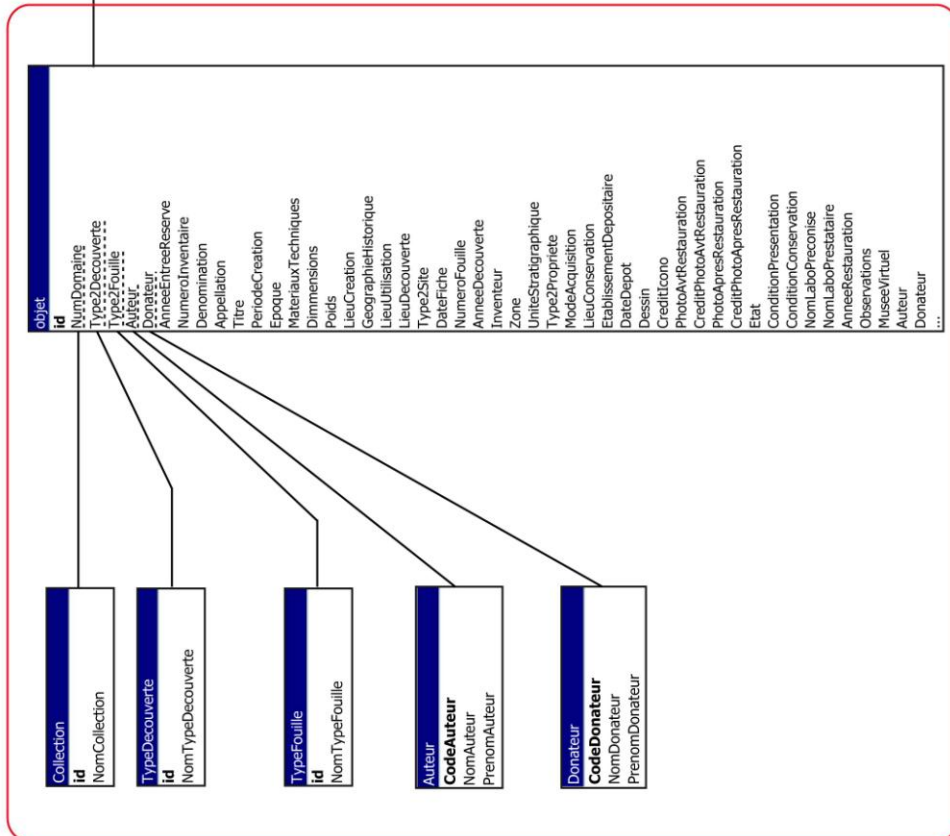
36- Modèle Conceptuel de Données pour le musée virtuel (MCD)

Du MCD découle le modèle logique de données qui permet également de créer le modèle physique de la base de données, une fois optimisée.

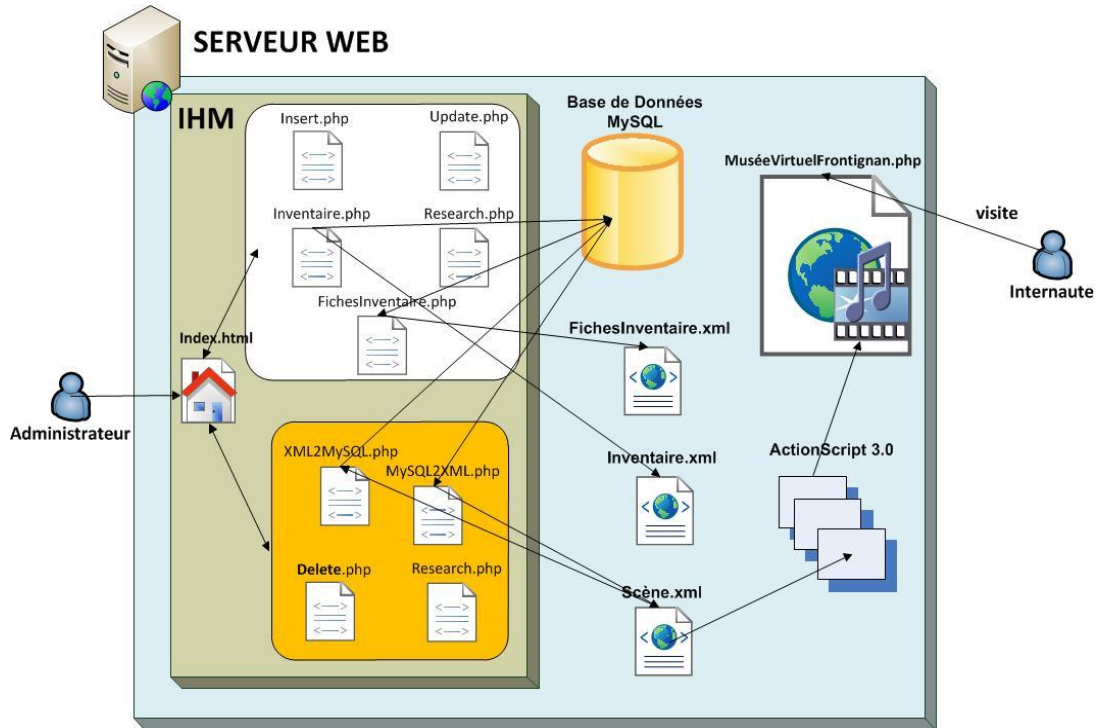
Pour le Musée Virtuel



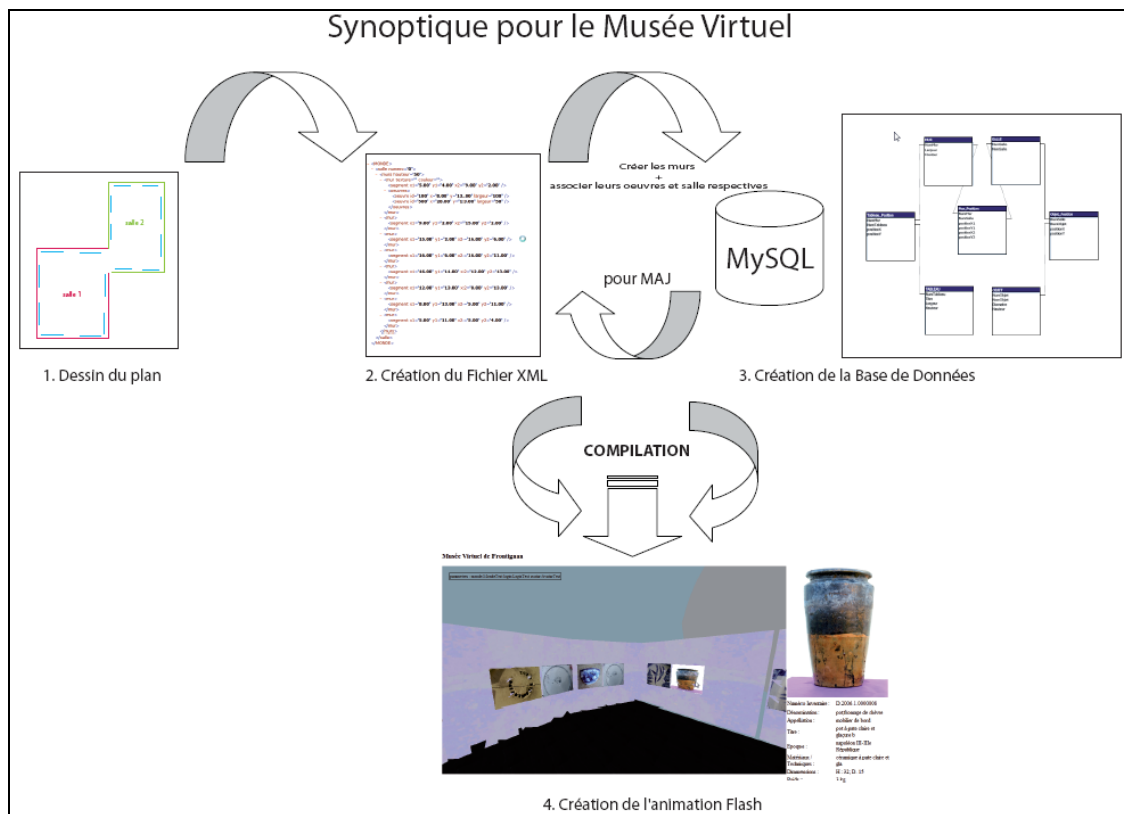
Pour l'inventaire



37- Modèle Logique de Données pour le musée virtuel (MLD)



38- représentation du SI



39- Synoptique du musée virtuel

Pour la gestion des œuvres positionnées dans les salles respectives du Musée Virtuel, différents formulaires sont dédiés afin d'insérer, modifier, supprimer et rechercher des objets inventoriés ou à inventorier. Ils permettront de gérer la base de données pour le musée virtuel en 2D, mais également de faire la liaison et ainsi maintenir une corrélation entre les objets présentés dans le musée virtuel en 3D (le prototype).

Le fichier XML « scène.xml » sert de point de départ pour la création du musée et permet de présenter les œuvres sélectionnées au préalable pour ce dernier.

Lors de modification de ce fichier ou de la base de données, deux scripts sont utilisables pour mettre à jour le fichier ou la base selon la nécessité.

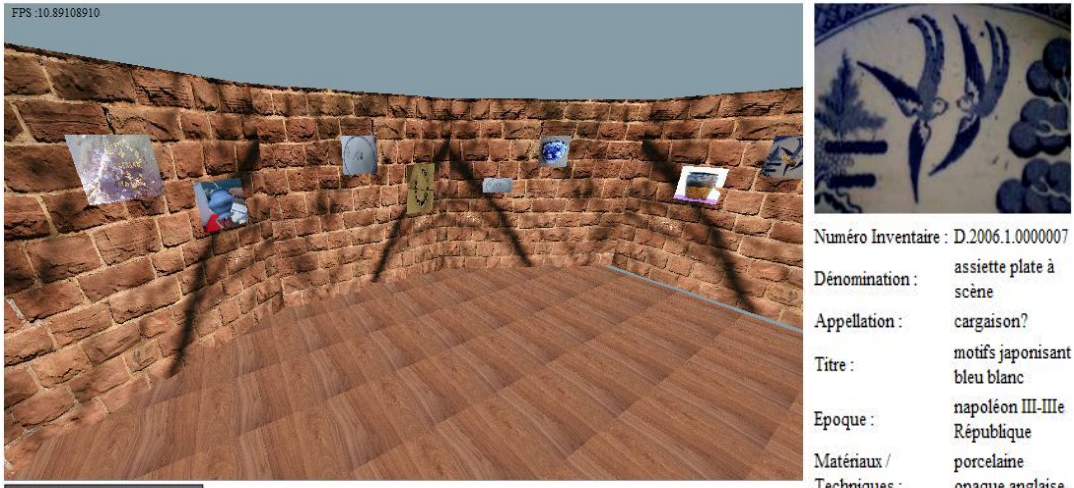
Effectivement une page PHP permettra de générer à partir du fichier XML la base de données MySQL afin de pouvoir ultérieurement modifier chacune des œuvres déjà présentes sur leur mur respectif en parcourant le fichier.

Voici ci-dessous une version du prototype réalisé

Voici une impression écran de la maquette du Musée Virtuel de Frontignan :

Musée Virtuel de Frontignan

FPS :10.89108910



Numéro Inventaire :	D.2006.1.0000007
Dénomination :	assiette plate à scène
Appellation :	cargaison?
Titre :	motifs japonisant bleu blanc
Epoque :	napoléon III-IIIe République
Matériaux / Techniques :	porcelaine opaque anglaise
Dimensions :	23
Poids :	30 g
Inventeur :	Claude Bené ; Michel Aubertel

**Aide à la navigation**

Pour tourner

1 clic pour activer la navigation

Gauche Avancer Droite

Reculer

40- vision générale du site du musée virtuel de Frontignan

L'interactivité entre le site et l'internaute est effective par le biais de l'utilisation du clavier pour naviguer dans les salles créées avec les œuvres sur chacun des murs. La souris permet la rotation à partir du point de position courant. Il aura la possibilité de cliquer sur une œuvre et d'en avoir le détail affiché sur la droite. La technologie AJAX combinée à une classe d'AS 3.0 permet de ne modifier que les données concernées (la partie qui affiche le détail de la dernière œuvre cliquée).

Une aide à la navigation est présente plus bas sur la gauche et explique brièvement les fonctionnalités possibles avec le clavier et la souris.

## **7.2 Aperçu du Framework PV3D**

Suite à l'étude de l'existant des technologies utilisées pour mettre en œuvre un environnement 3D, nous avons pu nous rendre compte de certaines différences de performances. Un des atouts du Framework PV3D est qu'il reste beaucoup plus commenté sur la toile que les autres. En plus de cela, ce Framework est gratuit contrairement à Alternativa 3D. Nous devons nous rappeler que le budget consacré à ce projet est de zéro euro. C'est pourquoi, le choix de ce Framework correspondait en tous points avec les exigences du projet. Nous allons voir quelques points essentiels pour comprendre rapidement les bases de PV3D.

### **7.2.1 Introduction**

Papervision3D ou PV3D, est un moteur 3D open-source pour Flash. Il permet de simuler un environnement en trois dimensions à partir d'une animation Flash et de tout ce qu'elle comporte (clips, boutons, photos, vidéos, etc.).

Il existe deux versions, l'une en ActionScript 2.0 et l'autre en ActionScript 3.0, cependant il est fortement recommandé d'utiliser la version AS3 qui est plus stable et plus avancée. Pour ce faire, j'ai choisi d'utiliser le SDK de Flex 3 d'Adobe comme outil de programmation pour générer le code nécessaire et définir les différentes classes. Adobe Flex 3 est un framework Open Source multi-plateforme permettant la création d'application internet riches (RIA) avec un rendu identique sur la plupart des navigateurs et des systèmes d'exploitation (Windows, Linux, Mac). Au départ, Flex donne la possibilité de créer des fichiers .SWF pouvant être lu par Adobe Flash

Player. Flex contient des classes, des composants, un compilateur, un debugger, des librairies et utilise XML (MXML) pour déclarer les balises de ses composants.

Ce sont des objets de la classe « Sprite ». Toute animation Flash est un objet de la classe « Sprite ». Nous avons donc la possibilité de créer une animation entière à partir d'un simple programme en ActionScript 3.0 et la classe « Sprite ». Bien sûr, il est nécessaire d'apporter des modifications à cette classe afin qu'elle génère l'animation. On dit alors que la classe « Sprite » est étendue et elle devient la classe du document.

### **7.2.2 Création de l'environnement 3D**

PV3D n'est autre qu'un ensemble de classes, des plans utiles à Flash pour représenter un environnement en 3 dimensions. Comme pour les classes « Sprite » et « MovieClip » de Flash, il suffit d'importer les classes PV3D dans vos animations (dans lesdites classes du document) afin de pouvoir les utiliser.

Ainsi, vos animations nécessitent l'importation d'un certain nombre de classes pour pouvoir utiliser le Framework Papervision3D. Et comme un réalisateur qui définit les éléments essentiels de son film, il existe des éléments immuables qui sont nécessaires pour faire fonctionner l'environnement 3D. Un environnement PV3D nécessite au moins une scène, une caméra, un acteur et des costumes.

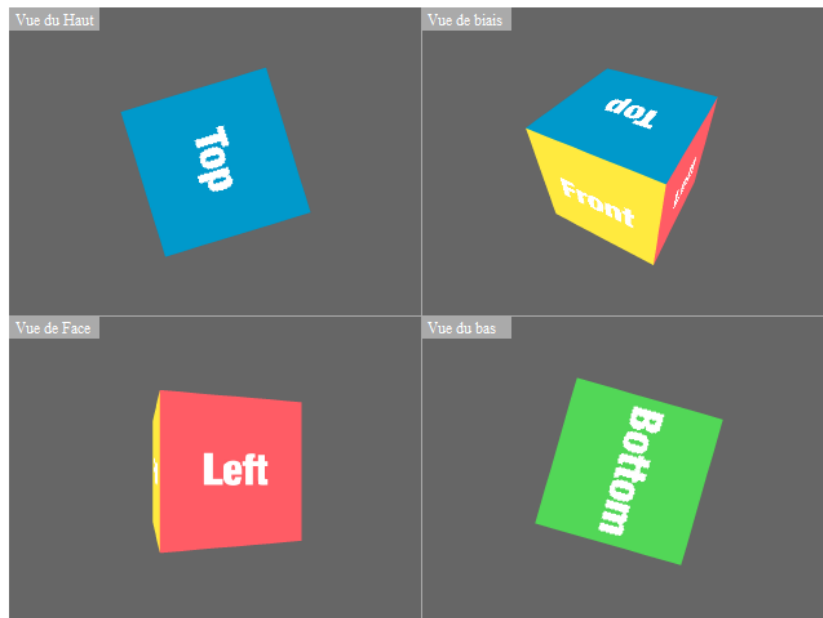
### **7.2.3 Fonctionnement d'un environnement Papervision3D**

La liste des éléments (ou classes) nécessaires mis conjointement en relation pour créer un environnement sous PaperVision 3D permet de modéliser un espace virtuel. Pour cet exemple, nous avons disposé de 4 caméras afin d'obtenir une vision différente de l'objet « Cube » en rotation avec les faces identifiées textuellement et utilisant également des codes couleurs.

Bien évidemment on peut utiliser ce paramétrage pour présenter un objet en 3 dimensions, une voiture qui aurait été modélisée en 3D par exemple pour une publicité ou autre besoin.

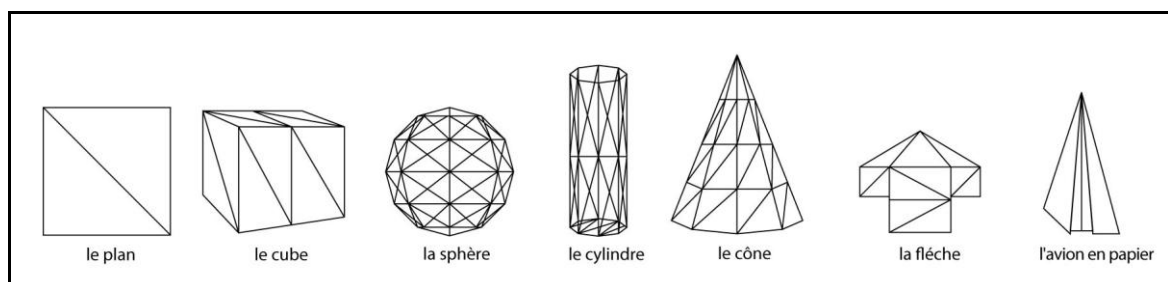
Un exemple disponible à ce lien : <http://musee.delcol.fr/PV3D/>

Papervision 3D en action



33- visualisation d'une scène avec PV3D découpée en 4 vues

Pour réaliser cet exemple, nous avons dû utiliser une des primitives fournies dans le Framework. Ci-dessous, nous voyons l'ensemble des primitives disponibles.



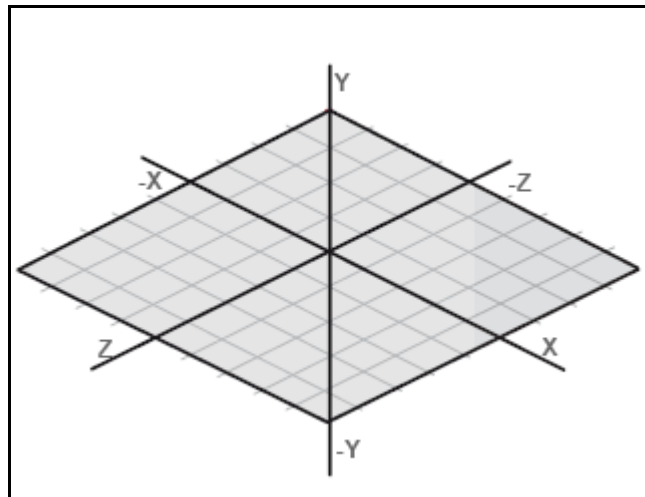
34- Représentation des différentes primitives fournies par PV3D

Ces dernières peuvent facilement être instanciées et donc ajoutées à la scène. Il suffit d'importer la classe correspondante à la primitive à utiliser.

## 7.2.4 Espace global et espace local



Lorsque nous ajoutons à un objet parent, un autre objet (ce dernier devient enfant du premier), sa référence spatiale par défaut correspond à l'espace local. C'est l'enfant parent qui est susceptible de pouvoir se déplacer dans l'espace global. Il ne faut pas confondre ces 2 espaces au risque de faire des erreurs.



35- Représentation tridimensionnelle d'une scène

### 7.2.5 Les Matériels :

Il existe une multitude de types de matériel qui peuvent être utilisés pour habiller, texturer les objets 3D. On parle aussi de « mappage ». On peut partir du plus simple matériel comme les images en passant par des animations et finir avec une vidéo en « stream ».

Pour chaque type, il y a une approche différente de mise en œuvre. Associé à l'interaction de la souris, l'animation PV3D offre de multiples artifices pour agrémenter au besoin, nos différentes réalisations.

Un objet tel un cube sans texture ou matériel est transparent, invisible. En fait la texture est la source du matériel qui sera utilisée pour appliquer cette texture sur l'objet 3D. La texture est étendue sur toute la surface de l'objet.

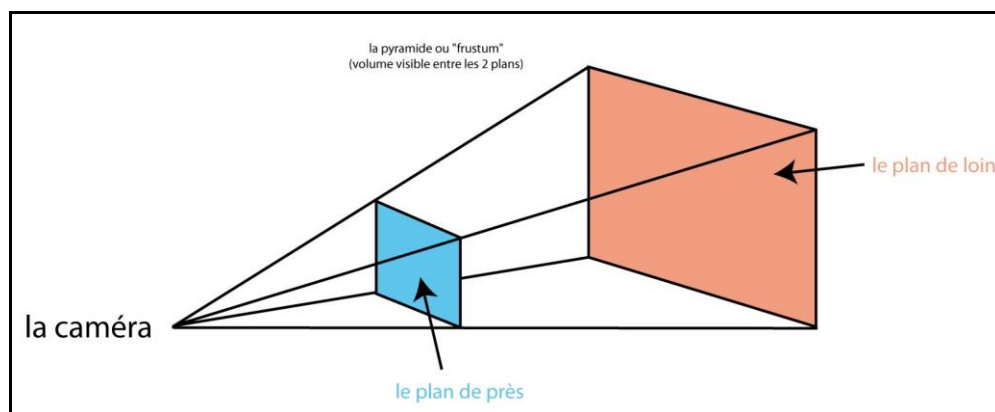
Juste avant nous parlions de triangles qui associés entre eux forment le squelette de l'objet. Le matériel représenterait, par similitude, à la peau de l'objet.

### 7.2.6 Les caméras :

L'utilisation des caméras dans PV3D n'a rien en commun avec celle des caméras classiques comme on peut se l'imaginer. La camera n'est pas visible sur la scène, dans la mesure où la scène représente la vision que l'on a depuis ce point virtuel qu'elle représente.

Les caméras héritent de la classe `displayObject3D` et de ce fait peuvent bouger, tourner, se positionner n'importe où dans l'espace afin de visionner les objets contenus dans son champ de vision avec plus ou moins de détail selon la valeur du zoom.

Différents paramètres sont utiles pour définir ces derniers dans un but précis qui diffère selon les résultats attendus, comme le zoom, le focus, le champ de vision.



36- Schémas explicatif de l'utilisation d'une caméra

Il existe 4 types de cameras incluses dans le Framework PV3D qui sont à utiliser en fonction de l'environnement à créer. Deux d'entre-elles sont les plus utilisées, à savoir la camera cible et la camera libre.

### **7.2.7 Optimisation de l'application PV3D :**

La principale problématique dans la réalisation d'une animation Flash reste à gérer et optimiser la fluidité de l'application tout en conservant une résolution adéquate et satisfaisante. Pour y parvenir il existe différentes techniques que nous allons voir plus en détail ci-dessous.

Le choix des techniques à utiliser dépend du type d'application PV3D qui est à réaliser. En effet, plusieurs facteurs sont à prendre en considération pour optimiser les performances de l'application et ainsi éviter de surcharger l'utilisation du CPU.

Il existe différentes techniques à utiliser et à adapté pour optimiser la fluidité de l'animation Flash crée et ainsi apporté un confort visuel satisfaisant.

Pour plus de détails, le Framework est expliqué dans les Annexes page 111.

### **7.2.8 Les navigateurs :**

Un facteur non négligeable pour la lecture des animations Flash est le navigateur. Hormis Internet Explorer, qui reste le navigateur le plus populaire avec Firefox, il en existe d'autres, toutes plateformes confondues, comme Google Chrome, Safari, Opéra qui offrent de meilleures performances en la matière. Pour une même animation flash, les résultats diffèrent selon le navigateur utilisé.

Comme exemple concrets, nous pouvons se rendre compte de ces différences en ouvrant l'exemple précédemment cité avec les 4 angles de vues avec les cubes, une fois avec Internet Explorer, et une autre fois avec Firefox. Le rendu n'est pas identique. Il existe encore quelques lacunes de ce côté-là.

## **VIII. Conclusion**

Après avoir étudié le Framework PV3D, celui-ci semble offrir de bonnes possibilités de virtualisation qui rendent le projet d'un musée virtuel tout à fait réalisable.

Cette version du musée offre la possibilité de créer autant de salles que nécessaires, tout en ayant la possibilité de personnaliser chacune d'elles par leur nombre de murs respectivement affectés ainsi que le nombre d'œuvres.

La génération du musée virtuel s'effectue de façon générique. Toutes les informations nécessaires sont disponibles dans le fichier XML qui servent à sa création. Chacune des œuvres (des tableaux) présentes dans les différentes salles permettent par un simple clic sur celles-ci d'avoir des informations complémentaires stockées dans une base de données qui sert également pour l'inventaire des œuvres du Musée Municipal de la ville de Frontignan.

Cependant le choix des technologies à mettre en œuvre dépend du réalisme attendu et des interactions voulues. Nous avons analysé les différentes possibilités qu'offrait ce Framework pour optimiser l'affichage dans un but de garder l'application aussi fluide que possible sans perdre en qualité de résolution des photos des œuvres.

Nous avons aussi remarqué que les différents navigateurs existants n'offrent pas tous les mêmes capacités pour afficher une animation Flash. Cette réalité est d'autant plus vraie que les postes des internautes n'ont pas tous les mêmes capacités à afficher une animation. C'est autant de contraintes qu'il faut garder en mémoire lors de la réalisation d'une animation 3D.

Enfin nous avons pu noter l'existence d'un logiciel « VizualPV3D », de Juxt Sr. ActionScripter et Gary Stasiuk, qui est une application de visualisation Flash qui fournit une interface pour créer et manipuler des objets en 3 dimensions dans une

scène en utilisant le Framework Papervision3D. « VizualPV3D » est une application qui aide à produire des sites internet en flash avec un design agréable et interactif pour créer des environnements PV3D.

Ce stage m'a permis non seulement de mettre en pratique des compétences techniques et de gestion de projet abordées durant la formation au CNAM, mais également d'acquérir un « savoir-faire » et une maîtrise de certaines technologies qui m'étaient jusqu'alors inconnues.

En outre, la réalisation du cahier des charges a été très intéressante pour l'avoir rédigé dans un cas réel avec la complicité de tous les acteurs de ce projet. Ce sont les différents échanges et rendez-vous qui m'ont fait aussi prendre conscience de la valeur des mots et de l'intérêt qu'il y a, et pas des moindres, pour que les informations passées soient comprises comme elles devaient l'être.

La communication est un des points clés dans le bon déroulement d'un projet quel qu'il soit. Entendre et comprendre sont les maîtres mots pour une communication efficace et bénéfique dans le cadre de la réalisation d'un projet.

Ce fût une très bonne expérience de passer ces dix mois au sein de cette collectivité. Mes compétences s'en trouvent enrichies que par la découverte de ces technologies nouvelles que j'ai su appréhender pour mener à bien ce projet.

## Liste des figures

1	Vision satellite de Frontignan et la région	P 9
2	Vue satellite de frontignan et ses alentours	P 9
3	Extrait de l'Organigramme fonctionnel des services de Frontignan	P 10
4	Diagramme de Gant	P 18
5	Visualisation du site de Musée d'Orsay	P 21
6	Visualisation du site de Musée du Louvre	P 22
7	Visualisation du site de Musée virtuel du Louvre	P 23
8	Exemple de visualisation d'une fiche détaillée d'une œuvre	P 23
9	Visualisation du site « Organismuseum »(vue 1)	P 24
10	Visualisation du site « Organismuseum »(vue 2)	P 25
11	Schéma représentant l'utilisation de la mémoire par Alternativa3D	P 36
12	Schéma représentant l'utilisation de la mémoire par Away 3D	P 36
13	Schéma représentant l'utilisation de la mémoire par PV3D	P 36
14	Schéma représentant l'utilisation de la mémoire par Sandy 3D	P 37
15	Récapitulatif des couleurs du site internet officiel de la ville	P 45
16	Modèle Conceptuel de Données de l'inventaire	P 46
17	Modèle Logique de Données de l'inventaire	P 47
18	Arborescence du site du musée	P 48
19	Maquette de la page d'accueil du site du musée municipal	P 50
20	Exemple de menus permettant le tri des œuvres par collections	P 50
21	Schéma de la 1 <sup>ère</sup> solution de création d'un circuit touristique	P 53
22	Schéma de la 2 <sup>ème</sup> solution de création d'un circuit touristique	P 55
23	Exemple de visualisation du site d'un parcours touristique	P 55
24	Visualisation des informations d'un marqueur (onglet Titre)	P 56
25	Visualisation des informations d'un marqueur (onglet Description)	P 56
26	Cas général des uses cases	P 59
27	Diagramme de classes	P 63
28	MCD pour le musée virtuel	P 65
29	MLD pour le musée virtuel	P 66
30	Représentation du SI	P 67
31	Synoptique du Musée virtuel	P 67
32	Prototype du site du musée virtuel de Frontignan en 3D	P 68
33	Visualisation d'une scène 3D avec PV3D sous 4 angles de vue	P 71
34	Représentation des différentes primitives sous PV3D	P 71

35	Représentation tridimensionnelle d'une scène	P 72
36	Schéma explicatif de l'utilisation d'une caméra	P 73
37	Explication de la formation d'un carré sous PV3D	P 118
38	Compréhension du terme et de l'utilisation de « segments »	P 119
39	Schéma explicatif du champ de vision et du focus d'une caméra	P 128

## Liste des tableaux

1	Tableau comparatif de la taille de l'animation générée pour les 4 moteurs 3D	P 34
2	Tableau comparatif de temps et de taille pour la création d'un objet filaire pour les 4 moteurs 3D	P 37
3	Comparatif d'utilisation des temps de création d'objets 3D pour les 4 moteurs 3D	P 39
4	Comparatif détaillé de la mémoire utilisée lors de la création d'objets 3D pour les 4 moteurs 3D	P 40
5	Use case pour la visite du musée virtuel	P 59
6	Use case pour administrer le musée virtuel	P 60
7	Liste des paramètres de création d'un plan	P 119
8	Liste des paramètres de création d'un cube	P 120
9	Liste des paramètres de création d'une sphère	P 121
10	Liste des paramètres de création d'un cylindre	P 121
11	Liste des paramètres de création d'un cône	P 122

## Glossaire

### **PV3D :**

PV3D est l'acronyme de PaperVision 3D qui est un moteur de rendu 3D pour générer des applications flash facilement intégrable dans un site Internet.

### **ActiveX :**

Les contrôle ActiveX sont une solution de création d'applications développée par Microsoft. Les principes de fonctionnement sont très semblables au fonctionnement des applets. Le code est dépendant de la plate-forme, le plus souvent.

### **Adobe Flex :**

Adobe Flex est une solution de développement créée par Macromedia en 2004 puis reprise par Adobe en 2006, permettant de créer et de déployer des applications Internet riches (RIA) multi plates-formes grâce à la technologie Flash et particulièrement son lecteur. Son modèle de programmation fait appel à MXML (basé sur XML) et AS 3.0, reposant sur ECMAScript. La technologie Flex produit un fichier « .swf » intégré dans une page html.

### **ActionScript :**

ActionScript est un langage de programmation utilisé au sein d'applications clientes (Adobe Flash, Adobe Flex) et serveur (Flash media server, JRun, Macromedia Generator). ActionScript est un langage de script, orienté objet et prototype, basé sur ECMAScript. ECMAScript étant une version standardisée de JavaScript, ActionScript et JavaScript partagent une syntaxe semblable.

### **Flash Player :**

Flash Player, développé et distribué par Macromedia (racheté en 2005 par Adobe Systems), est une application client fonctionnant sur la plupart des navigateurs Web. Ce logiciel permet la création de graphiques vectoriels et de bitmap animés par un langage script appelé ActionScript, et la diffusion de flux (*stream*) bi-directionnels audio et vidéo.



**Adobe Flash :**

Adobe Flash est un logiciel multimédia utilisé et spécialisé dans l'animation et l'interactivité des pages web via un format fermé de données d'animations vectorielles interactives.

**runtime :**

Un moteur d'exécution, bibliothèque d'exécution ou runtime (abréviation de *runtime system* ou *runtime library*, *runtime* signifiant, en anglais, « au moment de l'exécution ») est un programme ou une bibliothèque qui permet l'exécution d'un autre programme.

**MIP mapping :**

Le « MIP mapping » est une technique d'application de textures, les *MIP maps*, qui permet d'améliorer la qualité de l'affichage.

**Frustum :**

En géométrie, le “frustum” représente la portion d'un solide (plus souvent un cône ou une pyramide) qui est coupé par 2 plans parallèles. Ce terme est souvent utilisé en dessin numérique pour décrire ce qui est visible sur un écran

## Webographie

### **Action Script 3 :**

[http://livedocs.adobe.com/flash/9.0\\_fr/ActionScriptLangRefV3/compilerErrors.html](http://livedocs.adobe.com/flash/9.0_fr/ActionScriptLangRefV3/compilerErrors.html)  
<http://www.bases-as3.fr/>

### **Flex :**

[http://www.flashmagazine.com/tutorials/detail/flash\\_3d\\_basics/](http://www.flashmagazine.com/tutorials/detail/flash_3d_basics/)  
<http://www.adobe.com/devnet/flex/videotraining/>  
<http://www.adobe.com/devnet/flex/?view=documentation>  
<http://blog.phiphou.com/index.php/?2007/06/03/26-creer-des-composants-flex-avec-flash-cs3>  
[http://livedocs.adobe.com/flex/3/html/help.html?content=controls\\_06.html](http://livedocs.adobe.com/flex/3/html/help.html?content=controls_06.html)

### **Papervision 3D :**

<http://www.papervision3d.org/>  
<http://www.helioflash.com/adobe-flash/tutos-papervision-3d/francais/1/installer-papervision-3d/>  
<http://www.australopitech.com/883-collada-papervision3d>  
<http://malavoie.com/blog/2009/02/02/geolocalisation-avec-flash-et-papervision3d/>  
<http://www.tsoin.com/asblog/2008/11/20/comparatif-moteurs-3d-et-les-plus-performants-sont/>  
<http://papervision3d-fr.com/2008/05/23/les-cubes-sous-papervision/>  
<http://www.helioflash.com/adobe-flash/tutos-papervision-3d/francais/2/initiation-a-papervision-3d/#3>  
<http://www.juxtinteractive.com/work/vizualpv3d/>  
<http://blog.emmanuelbonnet.com/2007/06/08/papervision-panorama-360/>  
<http://www.hebiflux.com/blog/2008/11/19/3d-en-flash-comparatif-des-moteurs-et-des-nouveautes-chez-alternativa/>  
<http://docs.pv3d.org>  
<http://www.flashperfection.com/tutorials/3D-Cube-Gallery-70735.html>

### **Autres moteur 3D Flash :**

<http://www.hebiflux.com/blog/2008/11/19/3d-en-flash-comparatif-des-moteurs-et-des-nouveautes-chez-alternativa/>  
[http://www.3d-test.com/interviews/sandy3d\\_1.htm](http://www.3d-test.com/interviews/sandy3d_1.htm) (sandy 3D)  
<http://www.flashsandy.org/blog/category/fr> (sandy 3D)  
<http://www.mnt-tech.fr/blog/flash-flex/tutoriel-flex-et-alternativa3d-helloworld/> (alternativa 3D)  
<http://www.paperblog.fr/2791158/alternativa3d-le-moteur-flash-3d-qui-va-faire-mal/> (alternativa 3D)  
<http://papervision3d-fr.com/2008/06/22/vos-premiers-pas-avec-alternativa3d/> (alternativa 3D)  
<http://gdesplac.free.fr/index.php/?alternativa3d-50-sandy-papervision-and-co> (alternativa 3D)  
<http://msdn.microsoft.com/fr-fr/directx/default.aspx> (DirectX)

### **Musées :**

[http://www.trimaran.com/Interactif/References/\(detail\)/Film-en-3D-pour-la-decouverte-du-futur-musee-Magritte-a-Bruxelles-pour-GDF-Suez](http://www.trimaran.com/Interactif/References/(detail)/Film-en-3D-pour-la-decouverte-du-futur-musee-Magritte-a-Bruxelles-pour-GDF-Suez)  
(Animation sur le Musée de Renée Magritte à Bruxelles)

<http://www.musees.gc.ca/mad/reserve/index.php?flash=1>  
[http://www.paris.fr/portail/Culture/Portal.lut?page\\_id=4644&document\\_type\\_id=5&document\\_id=8407&portlet\\_id=9878](http://www.paris.fr/portail/Culture/Portal.lut?page_id=4644&document_type_id=5&document_id=8407&portlet_id=9878)  
[http://www.picturalissime.com/musee\\_paris\\_france.htm](http://www.picturalissime.com/musee_paris_france.htm)  
<http://www.culture.gouv.fr/documentation/museo/Musees.htm>  
<http://www.musee-orsay.fr/fr/accueil.html>  
<http://www.organismedia.com/virtualmuseum/>

**Autres sources pour les définitions et les traductions :**

[http://fr.wikipedia.org/wiki/Wikip%C3%A9dia:Accueil\\_principal](http://fr.wikipedia.org/wiki/Wikip%C3%A9dia:Accueil_principal) (Wikipedia)  
[http://www.reverso.net/text\\_translation.asp?lang=FR](http://www.reverso.net/text_translation.asp?lang=FR)

## **Bibliographie**

Il n'existe à ce jour aucun livre édité sur le Framework Papervision 3D. Les seules ressources se trouvent sur Internet.

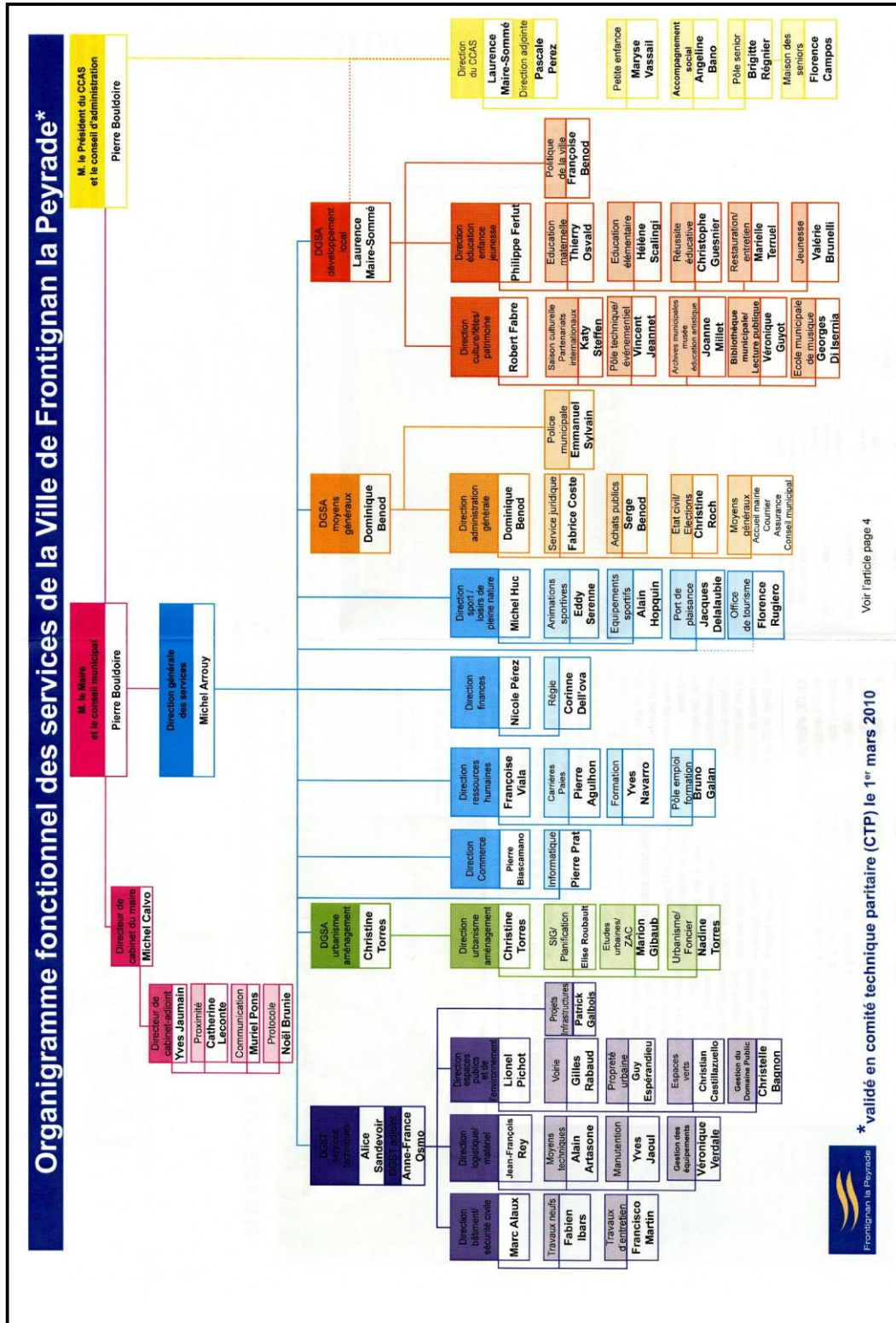
IMBERT T., 2009. **Pratique d'ActionScript 3.0**, Paris, 1080 p.

Macromedia Inc., 2004. **ActionScript 2.0**, CampusPress, Paris, 616 p.

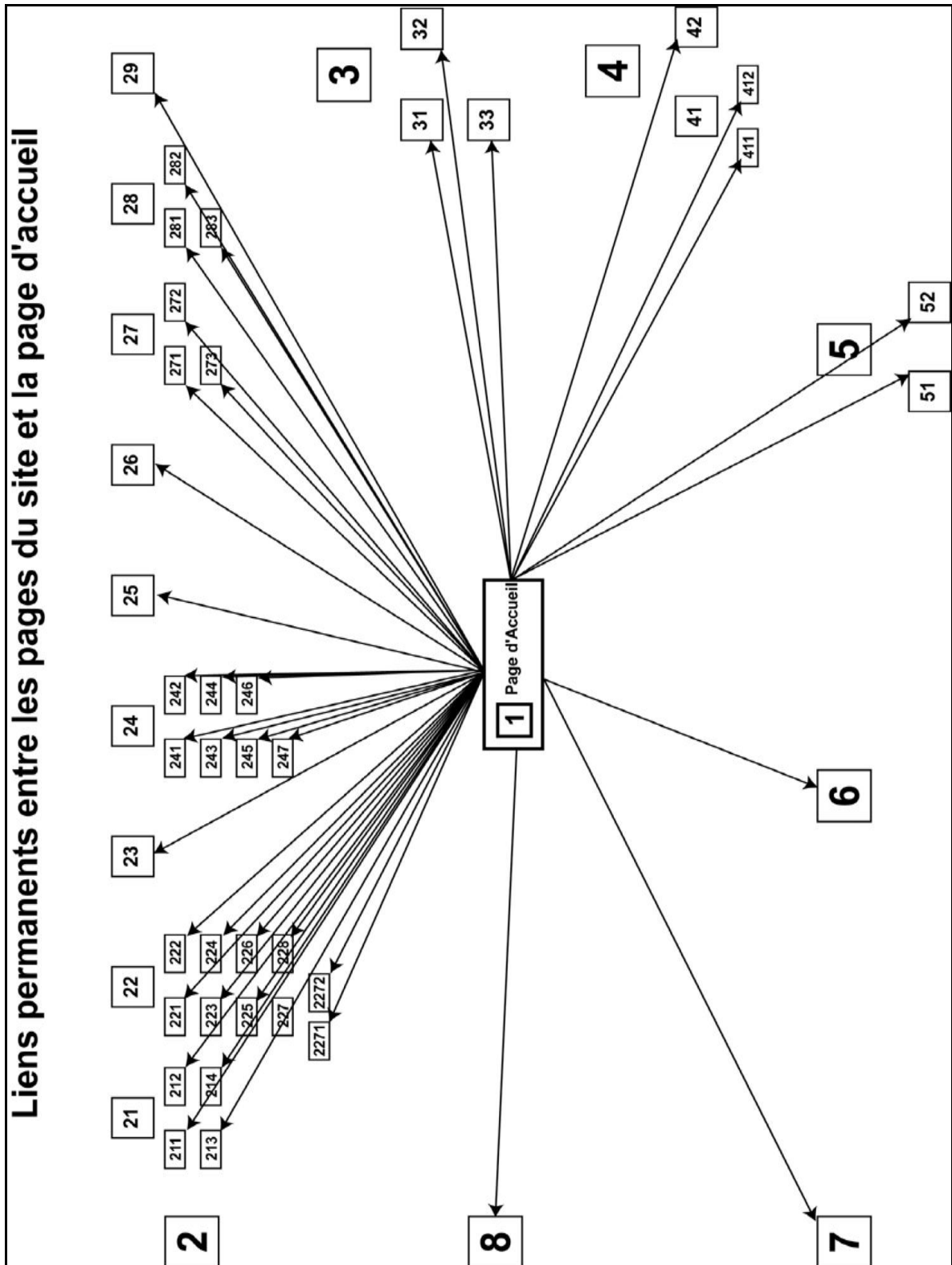
Webster S., Yard T., McSharry S., 2008. **Foundation ActionScript 3.0 with Flash CS3 and Flex**, USA, 578 p

## Annexes

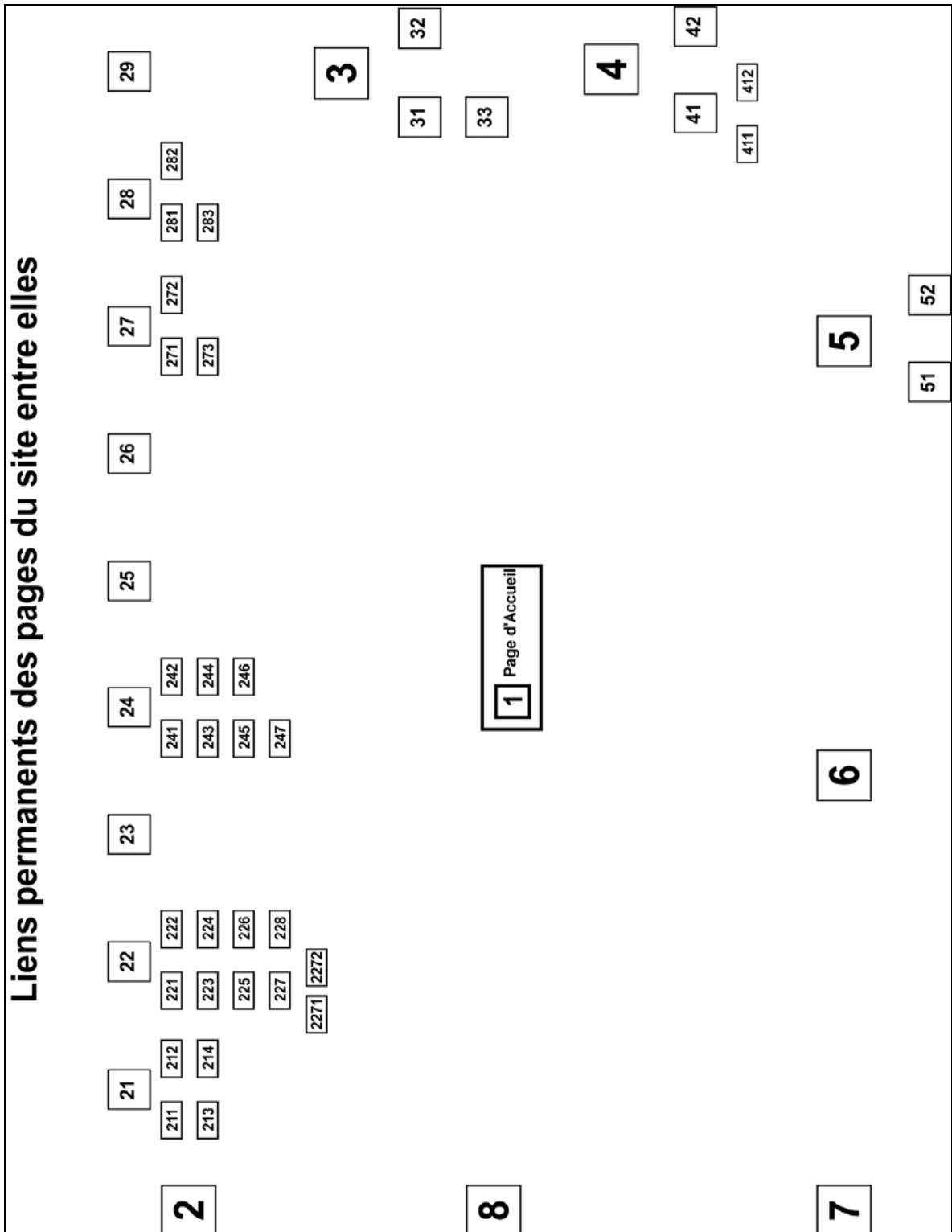
A1 - Organigramme des services de la municipalité de Frontignan	Page 85
A2 - schéma des liens permanents inter pages du site en 2D	Page 86
A3 - schéma des liens permanents inter pages du site en 2D	Page 88
A4 - Cahier des charges du site du musée en 2D	Page 88
A5 - Exemple d'un compte rendu de réunion	Page 103
A5 - Le Framework PV3D	Page 112



A1 – Organigramme de l'organisation des services au sein de la municipalité



A2 –schéma des liens permanents inter pages du site du musée en 2D



A3 –schéma des liens permanents des pages du site du musée en 2D



## VILLE DE FRONTIGNAN

## REALISATION DE SITE INTERNET Pour le MUSEE

## CAHIER DES CHARGES

# SOMMAIRE

## 1. PRESENTATION

### **1.1 L'objet de la consultation**

- 1.1.1 *Objectifs du site*
- 1.1.2 *A qui s'adresse le site*
- 1.1.3 *Principaux résultats attendus*

### **1.2 Le projet**

- 1.2.1 *La page d'accueil*
- 1.2.2 *Les rubriques*
- 1.2.3 *Arborescence du site*
- 1.2.4 *La constitution des rubriques*
- 1.2.5 *Le planning de réalisation*

### **1.3 Rôle de la ville de FRONTIGNAN et du prestataire**

- 1.3.1 *La ville de FRONTIGNAN*
- 1.3.2 *Le prestataire*
- 1.3.3 *Les interlocuteurs*

## 2. PRESTATIONS

- 2.1 *Charte graphique*
- 2.2 *Création des contenus*
- 2.3 *Développement*
- 2.4 *Nom de domaine et référencement*
- 2.5 *Référencement*
- 2.6 *Spécifications techniques du site*
- 2.7 *Maintenance et suivi*
- 2.8 *Statistiques de connections*

## 3. LIVRABLES ATTENDUS

## 4. PROPRIETE et DROITS

## **1. PRESENTATION**

La ville de Frontignan souhaite créer en plus du site actuel, un site internet pour son Musée.

A travers ce projet de création, la ville souhaite valoriser ce nouveau site internet sur le Musée comme un outil de communication, afin de fournir des informations qualitatives sur les actions conduites par les acteurs du secteur culturel. De plus le musée de la ville ne dispose pas d'une structure suffisante pour y accueillir toutes les œuvres en sa possession.

Ce dernier proposera une vitrine de ses œuvres en y associant une partie 2D et une autre 3D (un musée virtuel).

Cette création est l'occasion pour la ville de Frontignan de revêtir une image plus actuelle sur son site et de proposer une communication autour du Musée et de ses manifestations culturelles.

### **1.1 L'objet de la consultation**

**Le but du site créé dans le cadre du présent cahier des charges est de permettre la mise en ligne des œuvres de différentes collections en 2D classique (photo +descriptif) et une vision 3D du musée virtuel.**

**Ce nouveau site [www.musee.ville-frontignan.fr](http://www.musee.ville-frontignan.fr) a pour objectif à terme de proposer un éventail de services et d'informations régulièrement actualisées par le Service du Patrimoine afin d'offrir un outil de promotion culturelle.**

Cette création à présent envisagée doit permettre d'étendre l'autonomie de la ville dans l'administration du site. L'administration de ce dernier ne doit pas surcharger la tâche du webmaster au quotidien.

Et notamment :

- Présenter une interface moderne, dynamique et interactive afin de répondre aux nouvelles attentes des internautes. Ce site ne doit pas seulement présenter une vitrine informative, il faut qu'il soit un outil performant de valorisation de Frontignan.

Pour ce faire, nous devons :

- Alléger le contenu, travailler sur des accroches et des descriptifs incitatifs pour la mise en valeur de chaque rubrique et de chacune des œuvres.
- Mettre en avant le Musée via des visuels et laisser une place plus importante à l'image
- Réduire les pages statiques pour permettre au commanditaire d'éviter systématiquement l'intervention du webmaster et lui permettre d'agir directement sur la majeure partie du site pour changer des textes, des visuels.

Aujourd'hui, les informations visibles sur le site de la ville sont alimentées directement par la Mairie et le service de Communication qui intervient pour actualiser les différentes rubriques désirées.

### 1.1.1 Objectifs du site

Les principaux objectifs du site sont :

- Donner une image séduisante et attractive du Musée de la ville de Frontignan,
- Offrir des informations complètes, pouvant s'enrichir,
- Développer l'interactivité avec une vision 3D d'un musée virtuel,
- Organiser une ergonomie claire, pour sa lecture et son administration.

### 1.1.2 A qui s'adresse le site

- Grand public
- Citoyens
- Professionnels du Tourisme
- Presse
- Commerçants & artisans
- Partenaires institutionnels
- ...Pour tous

### 1.1.3 Principaux résultats attendus

Sur la base des desideratas exprimés par les différents interlocuteurs, et formalisés dans le présent cahier des charges, voici les principaux résultats attendus du site :

- Augmenter la visibilité et la fréquentation du site
- Affecter un design plus actuel et cohérent avec le service Communication de la ville
- Enrichir les contenus (base de données, etc...)
- Offrir la possibilité de télécharger de la documentation via le site internet sur les différents thèmes du site (tourisme, actualités, etc.),
- Mettre en place des outils de mises à jour, simples et accessibles par le webmaster, Mr Christophe MONTAGNE ou toutes autres personnes désignées.

Une réunion est prévue pour affiner les différents besoins du site, valider le choix des rubriques proposées, déterminer s'il y a ou non intégration du site du musée dans le site de la ville actuel, déterminer la ou les personnes qui seront en charge des mises à jour du site du Musée.

## 1.2 Le projet

### 1.2.1 La page d'accueil

Dés la page d'accueil, l'internaute pourra visualiser les différentes actualités culturelles et les différentes rubriques accessibles. La ville de Frontignan aura la possibilité de gérer les actualités figurant sur la page d'accueil.

- > **Recherches intuitives** : une facilité et gain de temps pour l'internaute, une valeur ajoutée pour la destination
- > **Cartographie dynamique** : affichage des points touristiques du centre ville de Frontignan (photos + descriptif pour chacun des points). La carte est reliée à une base de données pour faciliter la création, de nouveaux points patrimoniaux.
- > **Brochures** : téléchargement de brochures culturelles (format PDF)

La page d'accueil du Musée doit donc s'intégrer dans le site de la ville de Frontignan (attente de validation) afin d'offrir de multiples accès vers les rubriques et sous rubriques en gardant à l'esprit d'améliorer l'attractivité des informations culturelle et du patrimoine.

La page d'accueil doit pouvoir conserver une personnalité forte par son design afin d'illustrer le quotidien vivant de Frontignan et sa volonté de promouvoir tout ce qui a attiré à la culture aux internautes.

### 1.2.2 Les rubriques

Les principales rubriques énumérées dans la page d'accueil ci-dessous sont pour la plupart les thématiques à partir desquelles se déclinent les contenus.

La navigation à partir de ces rubriques ouvrira sur des listes de résultats en liaison avec une base de données.

Ces rubriques doivent pouvoir se différencier visuellement.

#### La liste des rubriques et sous-rubriques pour les collections:

- Art
  - Peinture à l'huile
  - Aquarelle
  - Pastels
  - Lithographies – Eaux Fortes – Gravures
  - Dessins
  - Photographie
  - Techniques Mixtes
    - Mosaïques
    - Marqueteries
  - Sculptures
- Collections d'objets sacrés
- Archéologie
  - Subaquatique

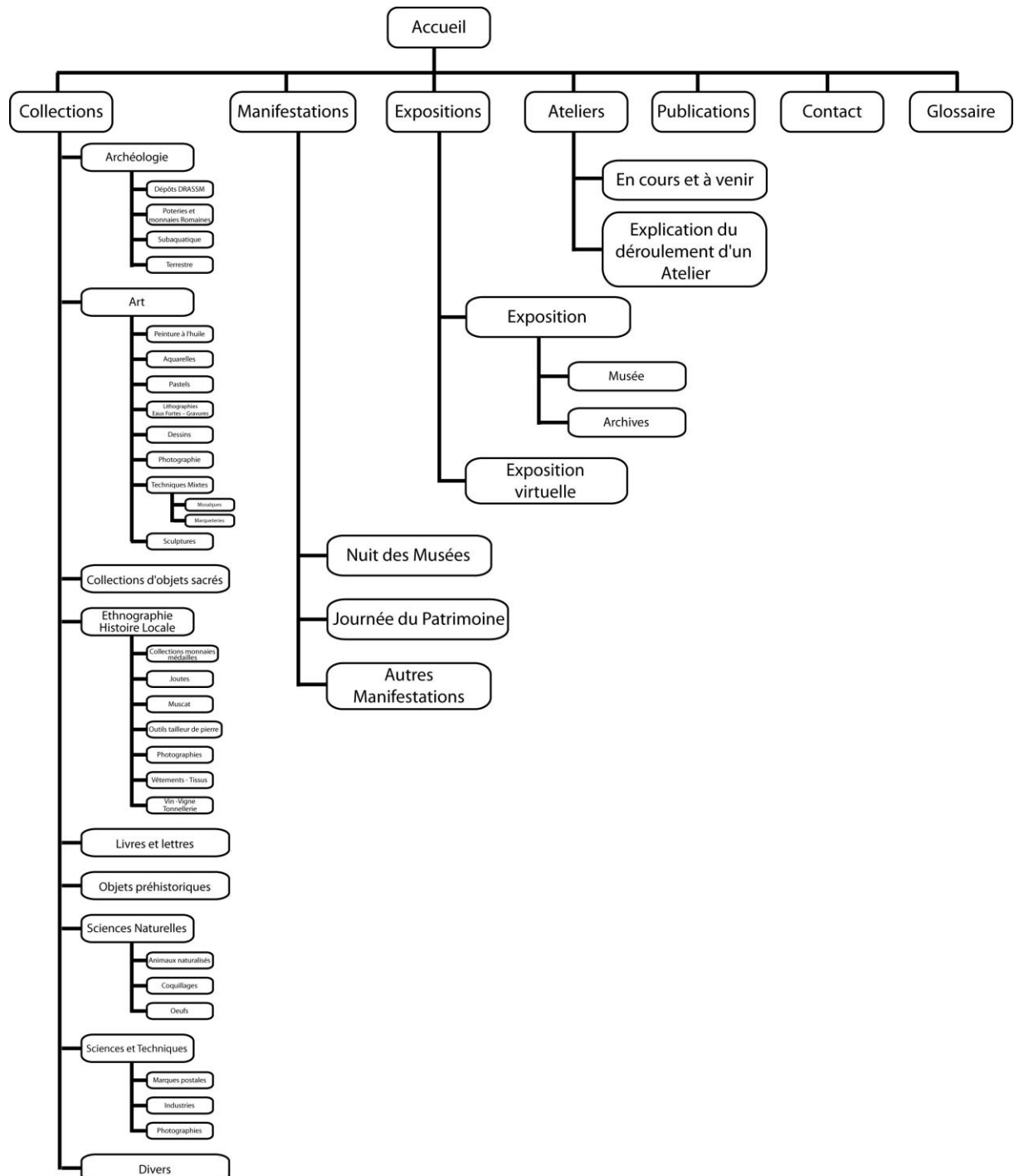
- Terrestre
- Dépôts DRASSM
- Poteries et Monnaies Romaines
  
- Science et Technique
  - Marques Postales
  - Industries
  - Photographie
  
- Livres et lettres
  
- Ethnographie – Histoire locale
  - Joutes
  - Muscat
  - Vin – Vigne – Tonnellerie
  - Outils de tailleur de pierre
  - Vêtements – Tissus
  - Photographies
  - Collections monnaie - Médailles
  
- Sciences Naturelles
  - Animaux naturalisés
  - Œufs
  - Coquillages
  
- Objets préhistoriques
  
- Divers

**La liste des rubriques et sous-rubriques pour les autres informations:**

- Accueil
- Manifestations
  - Nuit des Musées
  - Journée du patrimoine
  - Autres manifestations
- Expositions
  - Expositions
    - Musée
    - Archive
  - Exposition virtuelle
  
- Ateliers
  - En cours et à venir
  - Explication du déroulement d'un atelier

- Publications
- Contact
- Glossaire

### 1.2.3 Arborescence du site



### 1.2.4. La constitution des rubriques

La constitution précise de chaque rubrique sera validée ultérieurement par les services concernés.

**Téléchargement de documentation dans les rubriques concernées**



L'intérêt est d'éviter l'impression systématique et proposer des versions électroniques des différents supports papiers habituels conçus par le service Communication de la ville.

Les formats des fichiers téléchargeables seront les formats PDF ou JPG (essentiellement).

### **Construction des pages**

La réalisation du site fournira les deux premiers niveaux qui sont l'accueil et les rubriques thématiques, l'administrateur du site sera en charge d'intégrer les informations supplémentaires pour construire automatiquement les pages du troisième niveau et inférieurs, qui constituent les contenus pour chaque rubrique thématique, hormis les pages statiques.

- Le contenu de chaque rubrique est automatiquement lié aux données contenues dans la base de données.

### **1.2.5. Administration**

Le choix d'utiliser un CMS (Content Management System) permettra et facilitera l'administration du site par les personnes désignées à cette charge.

#### **La solution de gestion de contenu**

La solution de gestion de contenu (CMS) correspond à la volonté d'indépendance de la ville, aucune solution n'est désignée dans cette consultation, en revanche la solution recherchée doit pouvoir répondre à certains besoins :

- L'utilisation de ce CMS ne doit pas recourir à une formation lourde, la charge de travail du webmaster en sera facilitée.
- La solution CMS doit être utilisable en ligne de tout poste connecté sur une liste d'utilisateurs agréés par les services responsable du site (certaines personnes du service Patrimoine pourraient être susceptibles d'accéder aux données, à déterminer).

Après discussion avec le webmaster de la ville, Il semblerait que le choix se porte sur le CMS JOOMLA (à confirmer).

Une formation pourra être mise en place pour permettre aux personnes une parfaite autonomie.

### **1.2.6. Compatibilité consultation**

Le site pourra être visible par tous les visiteurs disposant au minimum de la version 5.x et ultérieurs d'internet Explorer, la version 7.0 de Netscape et compatible Mozilla sur PC et Mac.

Le site sera développé afin d'être visible par tous les utilisateurs disposant au minimum d'un affichage 16 bits en 800x600 ou 1024x768 (à déterminer).

### **1.2.7. Plug-in**

Le site utilisera sur la page d'accueil un ou des développements réalisées à l'aide de technologie FLASH 8, les visiteurs ne disposant pas du Player Flash compatible seront inviter à pouvoir le télécharger afin de visualiser correctement les différentes animations du site.

### **1.2.8. Méta balises, bas de page et liens**

#### ➤ **Copyright**

« © 2009 – Ville de Frontignan la Peyrade »

#### ➤ **Auteur**

« service du Patrimoine »

#### ➤ **Description et mots clés**

Cette balise permet de faire référencer efficacement le site par le choix de mots clés judicieusement choisis en rapport avec le contenu du site.

#### ➤ **Title**

Les pages pourront avoir des titres différents ou un titre unique (à confirmer).

#### ➤ **Bas de page**

Toutes les pages du site contiendront le même bas de page (communément appelé footer) avec les éléments suivants :

- Lien pour les « mentions légales »,
- Lien pour le « Plan du site »,
- Lien pour la page « Contact ».

#### ➤ **Liens**

Les liens seront formatés de la façon suivante (à confirmer) :

- Par défaut, une couleur différente du texte normal
- Lors du passage de la souris : changement de couleur et soulignement

### **1.2.9. Le planning prévisionnel de réalisation du site en 2D**

Dans l'attente d'avoir toutes les informations pour terminer le présent cahier des charges.

- début de la mission : mi-mars 2009.
- Entretiens pour établir les besoins du site avec les interlocuteurs référents.  
RDV avec le webmestre de la ville (vendredi 17 avril 2009)
- Livraison d'une version test du site Internet : 15 mai 2009.

- Formation des utilisateurs et première période de test : fin mai 2009.
- Corrections et améliorations éventuelles : première semaine de juin 2009.
- deuxième période de test : deuxième semaine de juin 2009
- nommage, référencement, hébergement : mi juin 2009
- mise en ligne officielle du site Internet dans le site existant (intégration) : fin juin 2009

### **1.3 Rôle de la ville de Frontignan et du prestataire**

#### **1.3.1 La ville de Frontignan**

- Valider les phases de choix (graphisme, ergonomie, contenus),
- Valider le respect du cahier des charges,
- Fournir les contenus de base (textes et documents, logos, images, photos)

#### **1.3.2 Le prestataire**

- Analyse de l'existant (site de la ville actuel),
- Concevoir la charte graphique du site,
- Adapter à une présentation web les contenus fournis,
- Créer les éléments graphiques (fond de page, boutons, icônes...),
- Mettre en place les différentes fonctions permettant la gestion des mises à jour par la ville.
- Proposer un planning de réalisation en accord avec la ville de Frontignan et s'engager à le respecter,
- Réaliser le développement du site,
- L'hébergement du site du musée sera intégrer au site de la ville (à confirmer),
- le référencement se fera sur le choix pertinent des mots clés à vocation culturelle,
- Déterminer pour la commune de Frontignan un ou des interlocuteur(s) « référent »

#### **1.3.3 Les interlocuteurs**

Mr Robert FABRE  
Mme Joanne MILLET  
Mme Murielle PONS  
M. Christophe MONTAGNE (webmaster)  
Mme Carole BRIFFAUD  
M. Florent CHALIFOUR  
M. Joachim

## **2. PRESTATIONS ATTENDUES**

## 2.1 Charte graphique et charte éditoriale

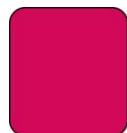
La charte graphique sera établie en collaboration avec le service Communication et le webmaster

- recherche graphique et iconographique,
- propositions de mise en page et de charte éditoriale,
- animation d'écran (utilisation d'animation flash éventuellement),
- recherche d'un système de navigation (schéma des différentes navigations possibles dans le site),
- frais, simple et ludique, le graphisme doit s'appuyer sur de nombreux visuels illustrant l'esprit de la page,
- le graphisme devra mettre en valeur les éléments clés de chaque page,
- L'ensemble sera basé sur des feuilles de style CSS, de manière à pouvoir changer rapidement de style et pour pérenniser sa fraîcheur.

Cette charte graphique permettra de créer un modèle de présentation élégante déclinée à toutes les pages du site avec l'utilisation de couleur thématique pour une meilleure lisibilité et de respecter les règles fondamentales d'ergonomie.

Moderne, créatif, ergonomique et qualitatif sont les mots clé pour désigner l'esprit du site.

**Voici les couleurs récupérées depuis le site officiel de la ville :**



D10958



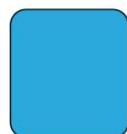
012D5A



97BD23



F0A232



00AEEF



097EB4



EEE084

## 2.2 Création et récupération de contenus

- adapter à une présentation web les contenus fournis,
- numériser les données qui sont nécessaires,
- créer les éléments graphiques (fond de page, boutons, icônes...).

## 2.2 Contraintes d'ergonomie et de navigation de base

Les contraintes Web de base qui doivent être respectées :

- ❖ Savoir des la première page de quoi traite le site,
- ❖ Savoir à tout moment à quel endroit on se trouve dans le site,
- ❖ Pouvoir à tout moment choisir une rubrique quelconque du site,
- ❖ Pouvoir à tout moment revenir à la page d'accueil.

### 2.3 Développement

Les éléments faisant appel à de la programmation et à des bases de données.

- L'entrée du site du musée se fera directement sur une page d'accueil informative sur laquelle on trouvera immédiatement l'ensemble de l'information et l'ensemble des rubriques du site déterminées,
- base de données (pour la gestion des œuvres et des actualités),
- gestion automatique d'actualités (en fonction de la base si besoin),
- interface de mise à jour du site,

### 2.4 Nom de domaine et Hébergement

Le nom de domaine déposé et existant est [www.ville-frontignan.fr](http://www.ville-frontignan.fr). L'hébergement, le nom de domaine et le maintien du site internet de la ville de Frontignan sont actuellement gérés par un prestataire « Pyxis ». (à valider)

### 2.5 Référencement

La ville de Frontignan souhaite continuer sa politique de promotion sur internet, notamment sur le thème de la culture, Frontignan vise une présence et un positionnement visible dans les moteurs et les annuaires sur des mots clés pertinents (à déterminer).

### 2.6 Spécifications techniques du site

- Structure arborescente
- Nom de domaine [www.ville-frontignan.fr](http://www.ville-frontignan.fr) (à déterminer)
- Nommer chaque page dans la balise TITLE par des titres parlants
- Remplir les balises META pour les descriptions et les mots clés de chaque page
- Enregistrer les pages avec des noms parlants pour les URL en ligne
- Préciser la date de création et des mises à jour sur la page d'accueil (si besoin),
- Pouvoir revenir sur l'accueil en cliquant sur le logo de n'importe quelle page du site.
- Le site devra être compatible avec les navigateurs les plus utilisés à savoir :
  - Internet explorer 6, 7 et suivant
  - Netscape 7
  - Mozilla
- Mise en page (à voir 800x600 ou autre)
- L'ensemble du site sera dynamique basé sur un langage PHP associé à des bases de données MySQL (à définir).

Pour construire ces pages et l'accès aux données, on pourra s'appuyer sur un modèle de construction MVC (Modèle – Vue – Contrôleur) qu'offre déjà certains CMS.

## **2.7 Maintenance et suivi**

La maintenance et le bon fonctionnement ininterrompu du site du musée sera assuré par le service informatique de la ville.

## **2.8 Statistiques de connections**

On pourra utiliser le service fournit par « Google Analytics » pour accéder en ligne aux statistiques de fréquentation du site.

Voici une liste non exhaustive :

- nombre d'accès visiteurs,
- mots clés tapés par les internautes,
- moteurs et annuaires de provenance,
- nombre de visites,
- pages visitées,
- provenance géographique des internautes,
- jours et tranches horaires des visites,
- types de navigateurs utilisés,
- ...

## **3. LIVRABLES ATTENDUS**

Les pièces mises à disposition pendant et après la réalisation du site sont les suivantes :

- arborescence détaillée du site,
- fichiers informatiques source (pages (HTML et PHP), graphismes, bases de données ...),
- documents décrivant le site, son fonctionnement, son hébergement et son intégration dans le site actuel,
- identifiant et mot de passe pour accéder au serveur et/ou à la partie administration du CMS.

## **4. PROPRIETE ET DROITS**

La Ville de Frontignan acquiert l'intégralité de la propriété de l'ensemble des images, graphismes, icônes et autres contenus créés pour le site.

## Compte Rendu Réunion Du 26/03/2009

**Personnes présentes :**

Joanne MILLET  
Bruno DELCOL

Version : 1.0

**Rapporteur :**

Bruno DELCOL

**Diffusion :**

Tous les participants

### Sujet : Création du site Internet 2D du Musée

#### Table des matières

1. ....	Etude et amélioration de l'existant	
.....		103
1.	Liste des Collections Modifiée (avec sous collections):	103
2.	Normalisation du Numéro d'inventaire :	104
3.	Logiciel GESPIMUS :	104
2. ....	Etude du Site Internet du Musée	
.....		105
1.	Objectif du site	105
2.	Hébergement	105
3.	Référencement	105
4.	Charte graphique	105
5.	Administration du site	105
6.	Liste des rubriques	105
7.	BDD	106
a.	Pour la Gestion des œuvres (+ édition du listing de l'inventaire du Musée)	106
b.	Pour la gestion des Manifestations	107
c.	Pour la gestion des Expositions	108
d.	Pour la gestion des Ateliers	108
e.	Pour la gestion des Publications	108
f.	Pour la gestion du Glossaire	108
8.	Maquette du site	110
a.	Vision d'ensemble	110
b.	Modèles des pages	110
c.	Contenu des pages	110

Cette réunion a pour but, à partir des éléments existants (classeur inventaire, GESPIMUS), d'améliorer l'organisation des collections et d'étudier les besoins pour l'élaboration du site Internet du Musée et de la base de données associée.

Les différents recueils des besoins permettront d'élaborer une première maquette pour le site internet qui sera modifiée selon tous les besoins recensés.

Cette étape permettra par la suite d'élaborer l'architecture et la création de la base de données MySQL qui servira par la suite à enrichir le site Internet dynamiquement.

Après consultations de la liste des collections, il s'avère utile de modifier cette liste.

## 1. Etude et amélioration de l'existant

### 1. Liste des Collections Modifiée (avec sous collections):

- Art
  - o Peinture à l'huile
  - o Aquarelle
  - o Pastels
  - o Lithographies – Eaux Fortes – Gravures
  - o Dessins
  - o Photographie
  
- Collections d'objets sacrés
  
- Archéologie
  - o Subaquatique
  - o Terrestre
  - o Dépôts DRASSM
  - o Poteries et Monnaies Romaines
  
- Science et Technique
  - o Marques Postales
  - o Industries
  - o Photographie
  
- Livres et lettres
  
- Techniques Mixtes
  - o Mosaïques
  - o Marqueteries



- Sculptures
- Ethnographie – Histoire locale
  - o Joutes
  - o Muscat
  - o Vin – Vigne – Tonnellerie
  - o Outils de tailleur de pierre
  - o Vêtements – Tissus
  - o Photographies
  - o Collections monnaie - Médailles
- Sciences Naturelles
  - o Animaux naturalisés
  - o Œufs
  - o Coquillages
- Objets préhistoriques
- Divers

Voici la liste des collections dans lesquelles sont réparties les œuvres.

Les collections avec sous collections ne seront jamais utilisé dans le but d'afficher toutes les œuvres des sous collections, le but est de pouvoir connaître le nombre d'œuvres dans la collection (donc la somme des œuvres contenus dans chacune de ses sous collections).

## **2. Normalisation du Numéro d'inventaire :**

On conservera la normalisation actuelle pour l'inventaire.

Année – N°Donateur – N°Collection – N° dans Collection – N° Fiche (= N°Œuvre)

Exemple : 1901 -335 – 01 – 0122 – 00905

## **3. Logiciel GESPIMUS :**

Le logiciel développé sous ACCESS permettant actuellement d'enregistrer des fiches (une fiche par œuvre), ne sera plus utilisé une fois la nouvelle base de données et le site créés.

## 2. Etude du Site Internet du Musée

### 1. Objectif du site

Promouvoir le Musée municipal  
Promouvoir les expositions (Musée, Archives et Patrimoine)  
Promouvoir les Ateliers  
Afficher les actualités sur la page d'accueil.

### 2. Hébergement

Il semblerait que le site du Musée soit intégré au site de la ville et non pas un site Internet à part entière.

Un prochain rendez-vous avec le webmaster permettra d'avoir plus de renseignements.

### 3. Référencement

Il faudra voir avec le webmaster comment est l'architecture du site de la ville pour s'intégrer au mieux et ajouter des « mots clés ».

### 4. Charte graphique

A déterminer avec Mme PONS du service Communication.

Cependant les couleurs, polices et photos du prototype ne sont pas définitives et ne servent qu'à définir les différentes zones et proposer une vision du site du Musée.

Le « bleu » et « muscat » sont actuellement les couleurs du site de la ville.

### 5. Administration du site

Il y aura une partie accessible à une liste de personnes à établir pour ce qui concerne les requêtes d'interrogation seulement.

Il y aura une seconde partie accessible au Webmaster pour tout ce qui se rapporte à l'administration de la base de données.

### 6. Liste des rubriques

#### Accueil

(Afficher les actualités sur 1 mois (vernissage, ateliers...))

#### Manifestations

- Nuit des Musées (x1)
- Journée du Patrimoine (x1)
- Autres

#### Exposition

- Exposition

- Musée (en cours et à venir)
- Archive (en cours et à venir)
- Exposition virtuelle  
Il s'agit de créer une sorte de diaporama qui juxtapose 2 photos  
(une ancienne à gauche et une récente à droite)

**Ateliers (de médiation culturelle)**

- En cours et à venir (x4 max)
- Explication du déroulement des ateliers  
Il s'agit de créer une sorte de diaporama qui juxtapose 2 photos  
(une ancienne à gauche et une récente à droite) par 2 photos

**Publications**

(Permet d'afficher les dernières publications de la ville)

**Contact**

(Plan d'accès + Téléphone, adresse, email et autres)

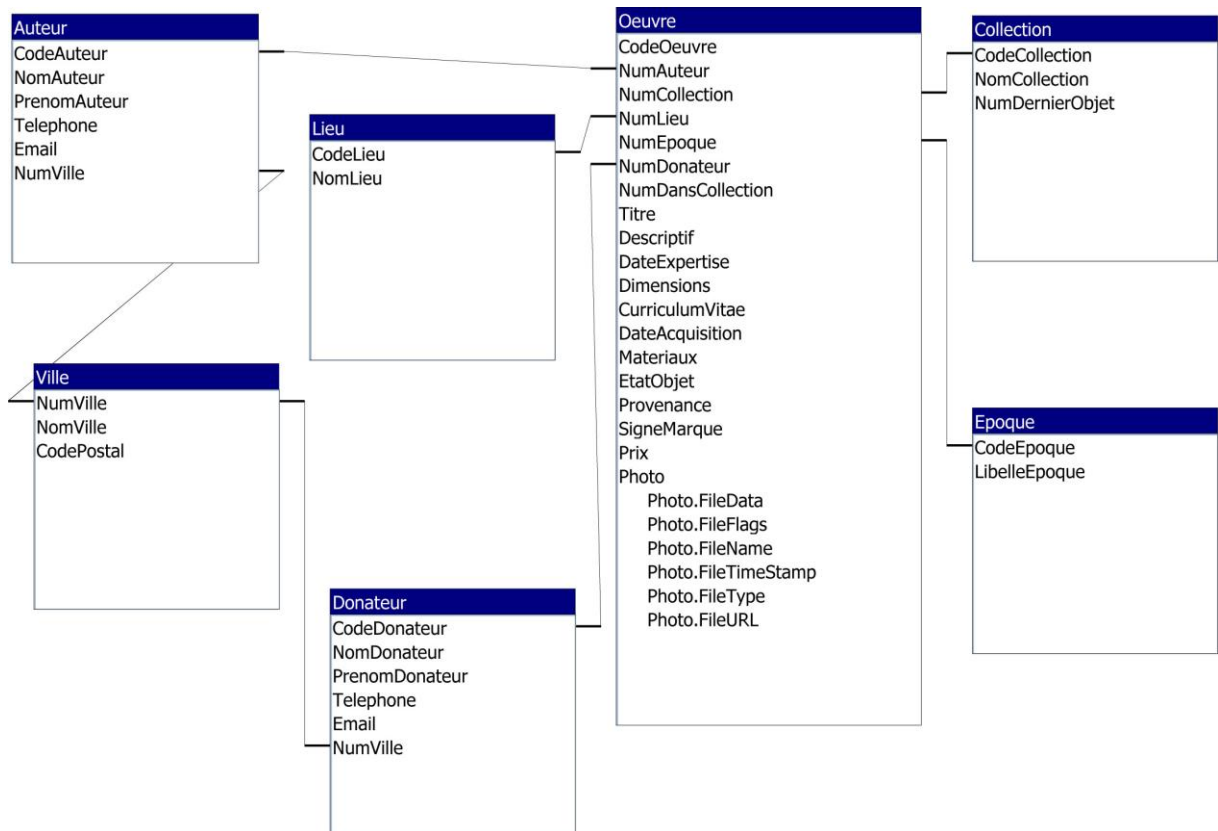
**Glossaire**

(Permet de lister et de définir certains mots techniques)

## **7. BDD**

### **a. Pour la Gestion des œuvres (+ édition du listing de l'inventaire du Musée)**

Voici une proposition d'un schéma de base de données optimisé :



Il s'avère que les champs ci-dessous seront formatés en texte libre :

- Etat
- Provenance
- Matériaux

On admettra que l'on ne supprime aucune œuvre de la base (l'attribut « EtatObjet » sera renseigné) pour que l'attribut « NumDernierObjet » dans la table COLLECTION corresponde au nombre d'œuvre dans la collection. Doit-on prévoir l'enregistrement de l'adresse du donateur et auteur (la ville suffit-elle ?).

### b. Pour la gestion des Manifestations

Voir si on utilise ou pas une base de données, pour permettre un affichage automatique (par date par exemple).

Voici les sous parties :

- Nuit des Musées (x1)
- Journée du Patrimoine (x1)
- Autres

### c. Pour la gestion des Expositions

Voir si on utilise ou pas une base de données, pour permettre un affichage automatique (par date par exemple).

Voici les sous parties :

→ Exposition

→ Musée (en cours et à venir)

→ Archive (en cours et à venir)

→ Exposition virtuelle

Il s'agit de créer une sorte de diaporama qui juxtapose 2 photos (une ancienne à gauche et une récente à droite)

### d. Pour la gestion des Ateliers

Voir si on utilise ou pas une base de données, pour permettre un affichage automatique (par date par exemple).

Voici les sous parties :

→ En cours et à venir (x4 max)

→ Explication du déroulement des ateliers

Il s'agit de créer une sorte de diaporama qui juxtapose 2 photos (une ancienne à gauche et une récente à droite) par 2 photos

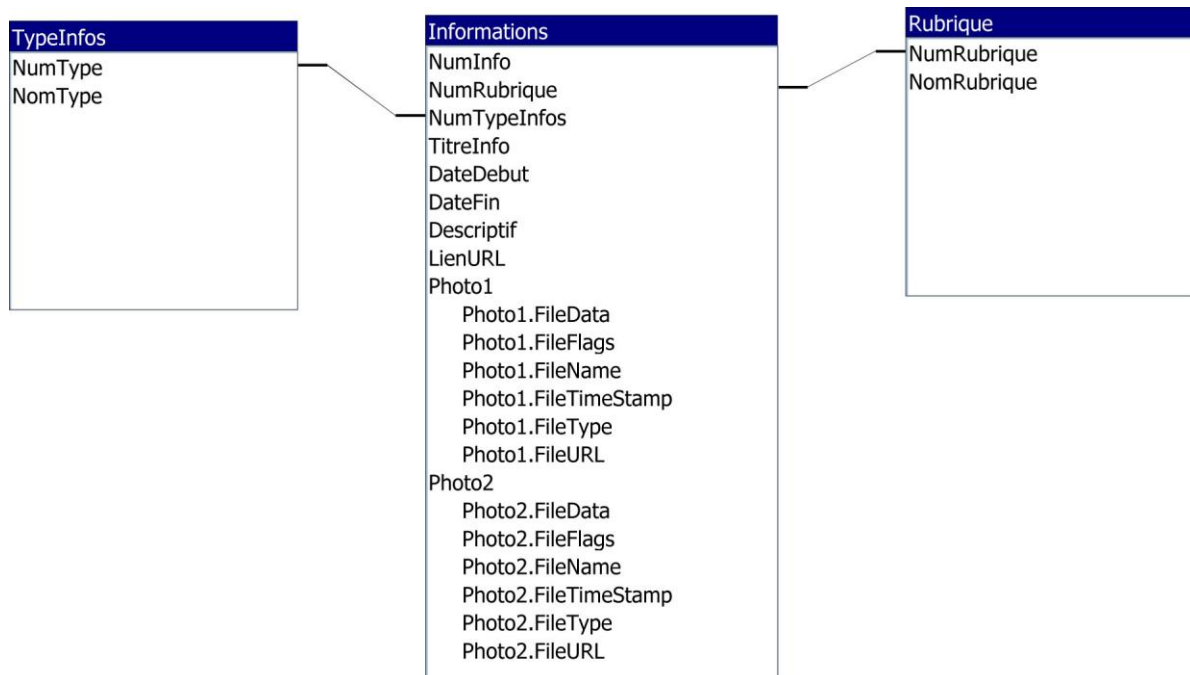
### e. Pour la gestion des Publications

Voir si on utilise ou pas une base de données, pour permettre un affichage automatique (par date par exemple).

### f. Pour la gestion du Glossaire

Voir si on utilise ou pas une base de données ou fichier XML par exemple (extraction).

Voici un schéma de base de données qui pourrait convenir à gérer les informations de chacune des rubriques. (À affiner selon les besoins choisis ultérieurement)



Il y a 2 champs photos dans la Table « Informations » pour l'exposition virtuelle qui nécessite 2 photos (1 ancienne et 1 récente), champs non obligatoire.

Il faudra déterminer si les photos doivent être dans la base ou en fichier.

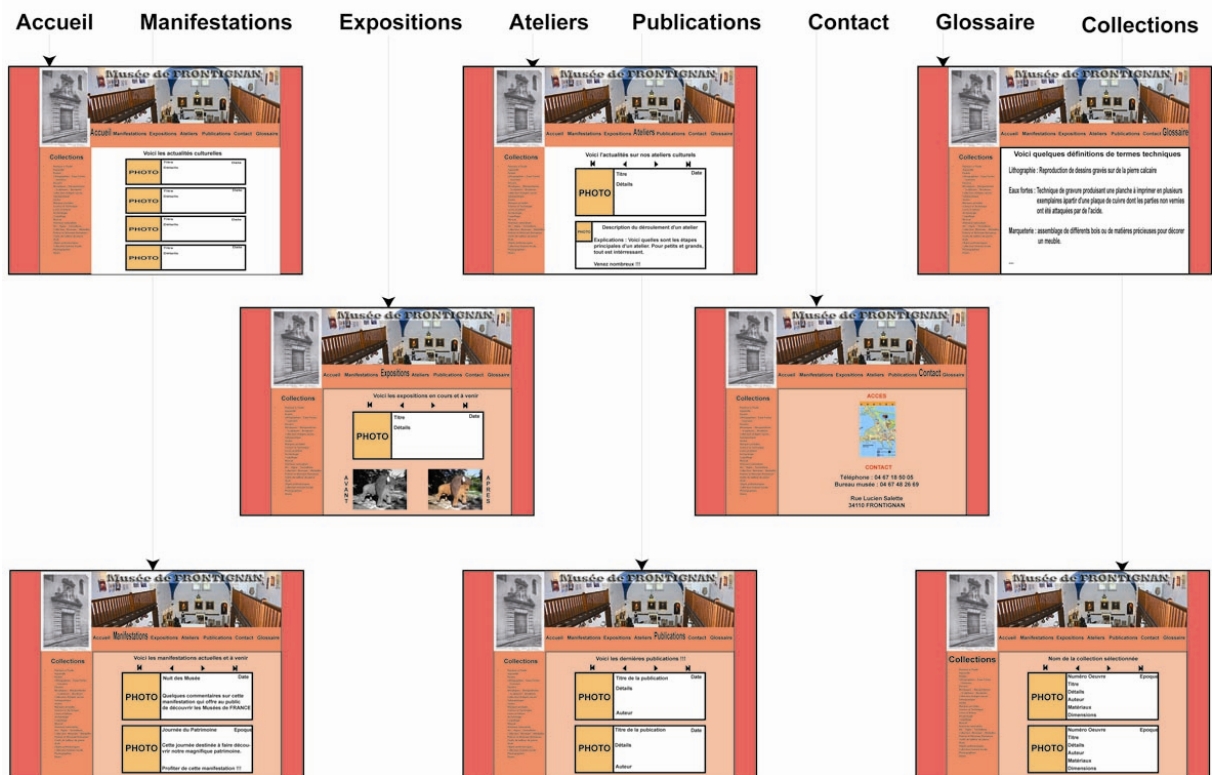
## 8. Maquette du site Internet

### a. Vision d'ensemble

Voir la photo jointe qui montre le site dans sa globalité en meilleure définition.

27-03-2009

#### Proposition de maquettes du site Internet du Musée de FRONTIGNAN



Toutes les pages ne sont pas représentées par soucis de place et de clarté.  
C'est pour donner une idée d'ensemble du site de Musée de Frontignan.  
Certaines parties de page sont en fond blanc pour faire varier l'apparence.

### b. Modèles des pages

Pour chacune des rubriques et sous rubriques il faudra déterminer une présentation ou un modèle des informations à afficher, la structure de la page.  
On utilisera les feuilles de styles CSS.

### c. Contenu des pages

Il faudra déterminer :

- quelles sont les pages non dynamiques  
comme la page « Contact » par exemple
- quelles sont les pages dynamiques qui seront liées à la base de données.

Une fois cela déterminé, on pourra créer la base de données en fonction des différentes informations à afficher, une fois les schémas validés.



## ***Le Framework PV3D***

### **1. Introduction**

Papervision3D ou PV3D, est un moteur 3D open-source pour Flash. Il permet de simuler un environnement en trois dimensions à partir d'une animation Flash et de tout ce qu'elle comporte (clips, boutons, photos, vidéos, etc).

Il existe deux versions, l'une en ActionScript 2.0 et l'autre en ActionScript 3.0, cependant il est fortement recommandé d'utiliser la version AS3 qui est plus stable et plus avancée. Pour ce faire j'ai choisi d'utiliser le SDK de Flex 3 de Adobe comme outil de programmation pour générer le code nécessaire pour définir les différentes classes. Adobe Flex 3 est un framework Open Source multi-plateforme permettant la création d'application internet riches (RIA) avec un rendu identique sur la plupart des navigateurs et des systèmes d'exploitation (Windows, Linux, Mac). Au départ, Flex donne la possibilité de créer des fichiers .SWF pouvant être lu par Adobe Flash Player. Flex contient des classes, des composants, un compilateur, un debugger, des librairies et utilise XML (MXML) pour déclarer les balises de ses composants.

Ce sont des objets de la classe « Sprite ». Toute animation Flash est un objet de la classe « Sprite ». Nous avons donc la possibilité de créer une animation entière à partir d'un simple programme en ActionScript 3.0 et la classe « Sprite ». Bien sûr, il est nécessaire d'apporter des modifications à cette classe afin qu'elle génère l'animation. On dit alors que la classe « Sprite » est étendue et elle devient la classe du document.

### **2. Création de l'environnement 3D**

PV3D n'est autre qu'un ensemble de classes, des plans utiles à Flash pour représenter un environnement en 3 dimensions. Comme pour les classes « Sprite » et « MovieClip » de Flash, il suffit d'importer les classes PV3D dans vos animations (dans lesdites classes du document) afin de pouvoir les utiliser.

Ainsi, vos animations nécessitent l'importation d'un certain nombre de classes Papervision3D. Chacune d'elles crée un objet utile à un environnement 3D.

Un environnement Papervision3D nécessite au moins une scène, un caméraman, un acteur et des costumes.

Chacun de ces types contient une série de classes pour générer des objets :

- la scène : les classes de type « scenes » : Scene3D, MovieScene3D, InteractiveScene3D,
- le caméraman : les classes de type cameras : Camera3D, FreeCamera3D
- les acteurs : les classes de type objects : Plane, Cube, Sphere
- les costumes : les classes de type materials ( BitmapMaterial, MovieMaterial, ColorMaterial)

### **La scène :**

Une animation Flash classique peut être classifiée comme étant une animation en 2D, c'est-à-dire comme ayant des objets animés par rapport à deux axes, X et Y. L'axe X représentant l'horizontal et l'axe Y la verticale. Ainsi chacune de vos interpolations de mouvement font jouer les coordonnées de vos clips sur ces deux axes. Une animation 3D comporte aussi les axes X et Y, auxquels on ajoute un troisième axe Z pour représenter la 3ème dimension, ou la profondeur. C'est ainsi que l'on obtient un effet de perspective et de 3D.

Une scène Papervision3D agit comme un repère à trois dimensions. C'est un objet destiné à contenir d'autres objets (les acteurs) afin de les positionner sur trois axes X, Y, et Z. Mais pour fonctionner une scène a besoin d'un conteneur, un objet qui va lui permettre d'accueillir des acteurs. Cet objet doit être de la classe « Sprite ».

### **Le caméraman :**

Pour se représenter une scène, Papervision3D ou pas, vous devez définir un observateur, l'endroit où il se situe et dans quelle direction il regarde. Ce sont les

objets de type cameras qui remplissent cette fonction. Il en existe deux différents, Camera3D et FreeCamera3D :

### **Cameras 3D :**

L'objet FreeCamera3D se définit avec 6 propriétés principales : les coordonnées x, y et z de la caméra, ainsi que les rotations sur les axes X, Y et Z. Le paramétrage d'un tel objet est assez complexe lorsque l'on débute, c'est pourquoi nous utiliserons plutôt l'objet Camera3D.

L'objet Camera3D se définit aussi avec 6 propriétés principales : les coordonnées x, y et z des deux objets « goto » et « target ». Ce sont des objets de la classe DisplayObject3D qui n'ont pour fonction que de représenter la position d'un point dans la scène 3D. En l'occurrence la position de la caméra et celle de sa cible. Les rotations de la caméra sont calculées automatiquement.

### **Les acteurs :**

Voici venus les objets de type « objects » : Plane, Cube et Sphère. Ce sont les acteurs d'une scène Papervision3D. Comme leur nom l'indique, ils vous permettent de créer des surfaces, des cubes ou encore des sphères.

Plane, Cube, Sphère, Papervision3D (ou PV3D)

Pour les personnaliser et leur donner des textures, nous les utilisons conjointement avec des objets de type « materials », les costumes en quelque sorte, leur « peau ». Par exemple, pour donner de la couleur à une surface on utilise l'objet « ColorMaterial ». Ou encore pour afficher le contenu d'un clip « MovieClip » on utilise l'objet « MovieMaterial ».

Une surface « Plane », de couleur 0x0000FF et de dimension 480 pixels par 320 pixels.

### 3. Fonctionnement d'un environnement Papervision3D

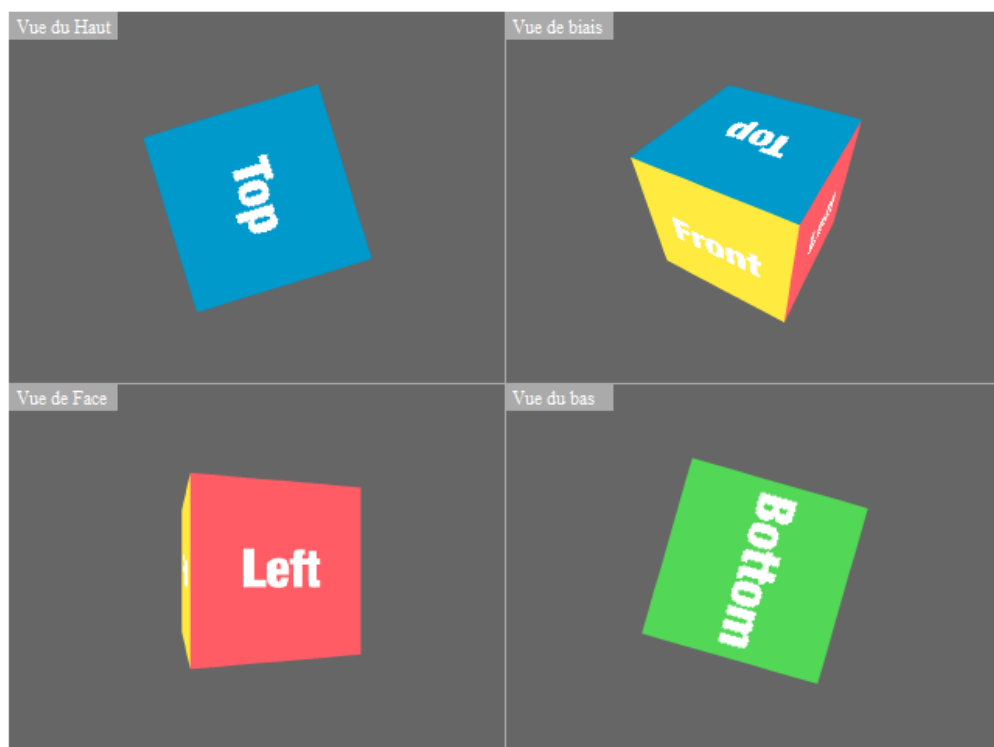
La liste des éléments (ou classes) nécessaires mis conjointement en relation pour créer un environnement sous PaperVision 3D permette de modéliser un espace virtuel.

Pour cet exemple, nous avons disposé de 4 caméras afin d'obtenir une vision différente de l'objet « Cube » en rotation avec les faces identifiées textuellement et utilisant également des couleurs.

Bien évidemment on peut utiliser ce paramétrage pour présenter un objet en 3 dimensions, une voiture qui aurait été modélisée en 3D par exemple pour une publicité ou autre besoin.

Lien : <http://musee.delcol.fr/PV3D/>

#### Papervision 3D en action



27- visualisation d'une scène avec PV3D découpée en 4 vues

#### 4. La méthode de rendu

PV3D utilise et gère un ensemble de processus qui permet de donner l'illusion de 3D dans un environnement 2D à partir d'objets instanciés à partir des primitives fournies par défaut.

Ce processus se décompose en 3 étapes ordonnées :

Initialisation → Projection → Affichage ou Rendu

Détaillons à présent ces 3 étapes clés.

##### L'initialisation :

Cette étape correspond à la création d'un « viewport », d'une scène, d'une caméra et d'un rendu avec un ou plusieurs objets.

##### La projection :

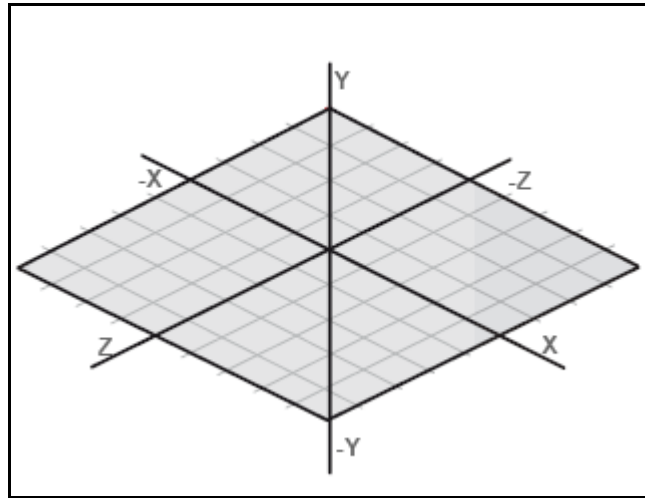
Cette étape consiste à projeter les objets 3D sur une surface 2D. Il faut donc convertir les coordonnées de chacun des points de l'objet sur une surface plane. PV3D calcule alors la position des points sur cette surface. A chaque fois qu'il y a un mouvement dans la scène, le rendu est recalculé systématiquement.

##### Le rendu :

Cette étape affiche à partir du calcul de projection et du type de matière utilisé, l'image finale en dessinant des triangles, plus ou moins nombreux selon la résolution souhaitée. Cette étape est la plus coûteuse en termes de performance. C'est là que le processeur est le plus sollicité.

#### 5. Espace global et espace local

Lorsque nous ajoutons à un objet parent, un autre objet (ce dernier devient enfant du premier), sa référence spatiale par défaut correspond à l'espace local. C'est l'enfant parent qui est susceptible de pouvoir se déplacer dans l'espace global. Il ne faut pas confondre ces 2 espaces au risque de faire des erreurs.



28- Représentation tridimensionnelle d'une scène

## 6. Les primitives

### 1. Les éléments basics d'un objet 3D :

Il est facile de se représenter un objet 3D dans la vie de tous les jours. Il est moins évident de donner l'illusion de la troisième dimension sur une surface en deux dimensions. C'est ce qu'offre le Framework PV3D.

#### a- Les « Points » :

Tout objet 3D est construit à partir d'un groupe de points formant les extrémités. Cet ensemble de points donne une vision générale de l'objet.

Les formes les plus simples peuvent être représentées alors qu'en combinant différentes formes entre elles, cela donne un objet unique. Un point possède 3 coordonnées dans l'espace. L'ordre de ces coordonnées est formalisé comme tel : x, y, z.

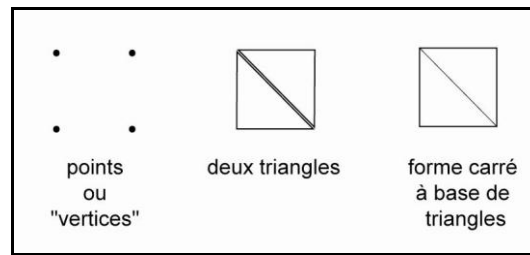
#### b- Les triangles :

Tout triangle est formé à partir de 3 points souvent appelé « face ».

PV3D se base sur les triangles ou des polygones triangulaires et seulement pour représenter un objet. C'est l'ensemble de triangles qui forme l'objet dessiné ou représenté.

Cette remarque n'est pas négligeable notamment lors d'import d'objets créés depuis une autre application comme Maya 3D ou autre. En effet selon le logiciel utilisé, les polygones par défaut ne sont pas forcément des triangles, ce qui peut engendrer des problèmes lors de son utilisation après son importation.

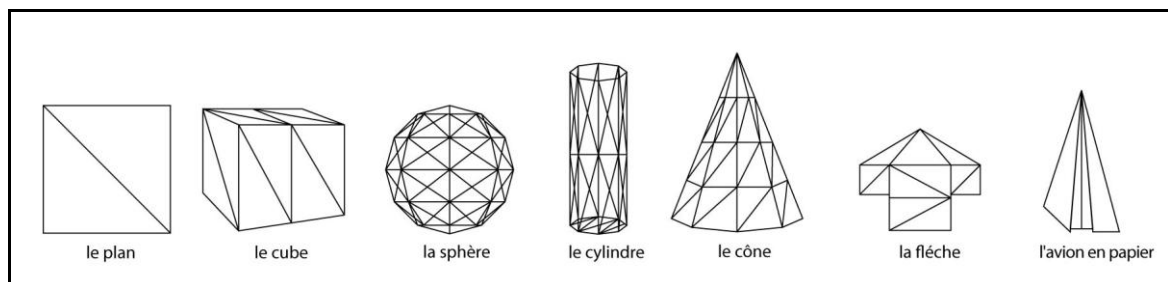
Pour comprendre la création d'un carré, nous allons voir un schéma qui explique son évolution en partant de point pour former des triangles qui ensemble forment alors le carré souhaité.



37- Explication de la formation d'un carré sous PV3D

## 2. Les primitives de PV3D :

PV3D fournit un ensemble de sept primitives qui sont le plan, le cube, la sphère, le cylindre, le cône, la flèche ou la pointe, et enfin l'avion en papier.



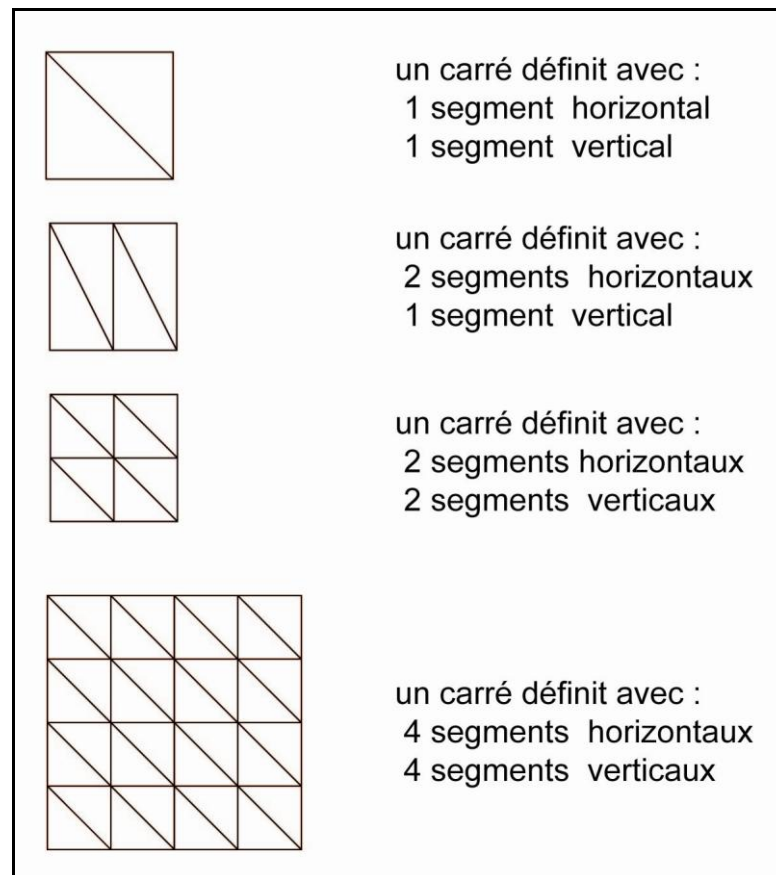
38- Représentation des différentes primitives fournies par PV3D

Ces dernières peuvent facilement être instanciées et donc ajouter à la scène.

Il suffit d'importer la classe correspondante à la primitive à utiliser.

Nous allons détailler, un peu plus, chacune d'entre elles en précisant leurs paramètres respectifs.

Pour comprendre les paramètres utilisant les termes de « segments », qui font parties des propriétés des primitives, nous allons utiliser un schéma.



38- Schéma pour la compréhension du terme et de l'utilisation de « segments »

Ceci s'applique à toutes les primitives listées ci-dessous. On se rend compte de la relation entre le nombre de points et le nombre de segments à afficher.

**Le Plan :**

C'est la primitive la plus simple de toutes.



VII- Liste des paramètres pour la création d'un plan :

	paramètres	Type de données	Valeur par défaut	Description
1	Matériel	MaterialObject3D	null	Définit le matériel à appliquer au plan, la texture.
2	largeur	nombre	0	Définit la taille du plan.
3	hauteur	nombre	0	Définit la hauteur du plan.
4	segmentsW	nombre	0	Définit le nombre de segments horizontaux.
5	segmentsH	nombre	0	Définit le nombre de segments verticaux.

**Le cube :**

VII- Liste des paramètres pour la création d'un cube :

	paramètres	Type de données	Valeur par défaut	Description
1	les matières	Liste de matières		Une liste qui définit les propriétés de matériel de l'objet.
2	Largeur	Nombre	500	Définit la largeur.
3	Profondeur	Nombre	500	Définit ma profondeur.
4	Hauteur	Nombre	500	Définit la hauteur.
5	segmentsS	Entier	1	Définit le nombre de segments pour la largeur.
6	segmentsT	Entier	1	Définit le nombre de segments pour la hauteur.
7	segmentsH	entier	1	Définit le nombre de

				segments pour la profondeur.
8	Faces internes	entier	0	Définit les faces qui sont visibles de l'intérieur du cube.
9	Faces exclues	Entier	0	Définit les faces qui ne seront pas créées.

### La sphère :

IX- Liste des paramètres pour la création d'une sphère :

	paramètres	Type de données	Valeur par défaut	Description
1	Matériel	MaterialObject3D	null	Définit le matériel qui sera appliqué à la sphère.
2	Rayon	Nombre	100	Définit la valeur du rayon de la sphère.
3	segmentsW	Nombre	8	Définit le nombre de segments horizontaux.
4	segmentsH	Nombre	6	Définit le nombre de segments horizontaux.

### Le cylindre :

X- Liste des paramètres pour la création d'un cylindre :

	paramètres	Type de données	Valeur par défaut	Description
1	Matériel	MaterialObject3D	Null	Définit le matériel qui sera appliqué au cylindre.
2	Rayon	Nombre	100	Définit le rayon du cylindre.

3	Hauteur	Nombre	100	Définit la hauteur du cylindre.
4	segmentsW	Nombre	8	Définit le nombre segments horizontaux.
5	segmentsH	Nombre	6	Définit le nombre de segments verticaux.
6	Epaisseur	Nombre	-1	Ce paramètre définit la valeur du rayon de la face du haut du cylindre, par défaut il est égal au rayon passé en second paramètre.
7	Face du haut	Booléen	Vrai	Définit si la face du haut est créée.
8	Face du bas	Booléen	Vrai	Définit si la face du bas est créée.

**Le cône :**

XI- Liste des paramètres pour la création d'un cône :

	paramètres	Type de données	Valeur par défaut	Description
1	Materiel	MaterialObject3D	Null	Définit le matériel qui sera appliqué au cylindre.
2	Rayon	Nombre	100	Définit le rayon du cône.
3	Hauteur	Nombre	100	Définit la hauteur du cône.
4	segmentsW	Nombre	8	Définit le nombre de segments horizontaux.
5	segmentsH			Définit le nombre de segments verticaux.

### **La flèche ou la pointe :**

La flèche n'est pas une primitive que l'on peut utiliser dans des projets. Cet objet permet, de par son utilisation, de tester des orientations dans un environnement en cours de réalisation.

Le seul paramètre que l'on peut passer est le matériel.

Ses dimensions sont fixes, 400 de largeur, 100 de hauteur et 600 pour la profondeur. Pour modifier ses propriétés, nous devons utiliser l'option « scale » qui met à l'échelle l'objet à la taille désirée. Ce n'est donc pas un objet dont on se sert pour des réalisations finales.

### **L'avion en papier :**

L'objet papier n'est pas non plus une primitive comme la flèche. Son utilisation sert d'outil pour contrôler un angle, une orientation d'objets en cours de mouvements.

## **7. Les Matériels**

Il existe une multitude de types de matériel qui peuvent être utilisés pour habiller, texturer les objets 3D. On parle aussi de « mappage ». On peut partir du plus simple matériel comme les images en passant par des animations et finir avec une vidéo en « stream ».

Pour chaque type, il y a une approche différente de mise en œuvre. Associé à l'interaction de la souris, l'animation PV3D offre de multiples artifices pour agrémenter au besoin, nos différentes réalisations.

Un objet tel un cube sans texture ou matériel est transparent, invisible. En fait la texture est la source du matériel qui sera utilisée pour appliquer cette texture sur l'objet 3D. La texture est étendue sur toute la surface de l'objet.

Juste avant nous parlions de triangles qui associés entre eux forment le squelette de l'objet. Le matériel représenterait par similitude à la peau de l'objet.

## 1. Les propriétés :

Cette classe hérite des propriétés de la classe mère « MaterialObject3D » qui sont communes à tous les matériels. Chacun des objets auront des propriétés intrinsèques à leur usage qui sont :

- Application simple face ou double face,
- Lissage,
- Application en opposition (ou inverse).

Leur utilisation dépend du résultat attendu. Lorsque nous souhaitons agrandir un objet, le matériel peut devenir pixélisé, c'est ici que la propriété de lissage (ou « smoothing ») intervient et peut résoudre le problème. L'utilisation abusive de cette propriété peut engendrer des coûts en termes de performances.

## 2. Les matériels de base :

Il existe 2 types de matériel de base que sont les matériels filaires ou de couleurs. L'utilisation de ces types n'est pas forcément agréable à l'œil en termes d'esthétique. La représentation des primitives fournies avec PV3D est sous forme filaire (voir figure page 43). Des fils ou traits sont créés entre chacun des points pour former l'objet 3D.

Par défaut dans PV3D, chaque objet créé est de type filaire. Pour ce type de matériel nous pouvons définir, la couleur, la transparence en pourcentage et l'épaisseur des fils (« 0 » définit l'épaisseur de fil aussi épais qu'un cheveu). Pour ce type de matériel nous utilisons de préférence des couleurs en 24 bits plutôt qu'en 32 bits qui gèrent la transparence en plus. Il n'y a donc pas d'intérêt d'utiliser la transparence pour ce type de matériel d'autant que la transparence est le deuxième paramètre à définir selon le cas.

Après le type filaire, voyons le type couleur qui permet d'appliquer au squelette une couleur. Pour définir ce type on peut utiliser les 3 paramètres qui sont la couleur à appliquer, sa transparence en enfin si ce dernier est interactif (booléen).

Ces deux types définissent les types de base qui peuvent être appliqués aux objets 3D.

L'utilisation de ces 2 types est assez facile.

### **3. Les images comme matériel :**

Pour rajouter de l'intérêt, du détail et plus de réalisme, on peut utiliser de 3 manières les images. Pour donner de l'intérêt à un objet, on pourrait s'appliquer à simuler une troisième dimension en utilisant différents moyens comme celui de l'utilisation de lumière d'une deuxième image ou d'autre subterfuge...

Le « BitmapMaterial » offre la possibilité d'utiliser une couleur donnée ou une image depuis un fichier.

Le « BitmapFileMaterial » offre, comme le premier, d'utiliser une image, mais avec plus de facilité, car on spécifie dans les paramètres et notamment le premier, le chemin du fichier). Une technique de programmation permet d'afficher l'image, une fois celle-ci téléchargée entièrement. Cela évite de voir un carré noir en attendant la fin du téléchargement du fichier. Tant que ce dernier n'est pas complètement chargé, on ne modifie pas l'aperçu de la scène. Pour ce faire nous avons recours aux « eventlistener » qui offrent 4 différents niveaux d'écoute (chargement complet, chargement en progression, erreur de chargement, erreur de sécurité). Ces « écouteurs » s'occupent de gérer au mieux les interactions voulues et attendues. Ils sont très importants et puissants en termes d'efficacité.

Le « BitmapAssetMaterial » est une classe qui définit une image comme texture. Cependant Flex n'intègre pas de librairie comme le logiciel Flash.

### **4. Les animations :**

Les animations Flash ont une réputation qui aujourd'hui reste encore d'actualité. Effectivement, Flash reste populaire et fiable dans la production d'animation.

Au-delà des images, nous pouvons utiliser des animations comme texture. Cela rend l'application finale plus attrayante.

Le « MovieMaterial » possède 5 paramètres qui permettent :

- De spécifier l'objet support qui utilisera cette animation,
- De spécifier s'il doit gérer la transparence,
- De spécifier s'il doit être recalculé à chaque fréquence d'image,
- De spécifier la précision du matériel,
- De spécifier s'il peut s'étendre en fonction de la taille de l'animation ou non.

L'activation de certains de ces paramètres peut engendrer des problèmes d'affichage. Il est donc judicieux les utiliser avec une attention particulière en fonction des besoins.

Le « VideoStreamMaterial » applique sur la surface d'un objet une vidéo. Il possède 4 paramètres qui offrent le moyen :

- De spécifier la source vidéo,
- De spécifier l'instance de contrôle de la vidéo,
- De spécifier la précision,
- De spécifier si la transparence doit être supportée.

Pour des raisons de sécurité, il faut au préalable s'assurer que l'accès ne posera aucun problème. C'est le lecteur Flash qui détermine s'il y a violation de droit ou non.

Pour diverses raisons, l'utilisation simultanée de matériel peut être requise et mise en œuvre pour répondre aux exigences visuelles attendues.

## 8. Les caméras

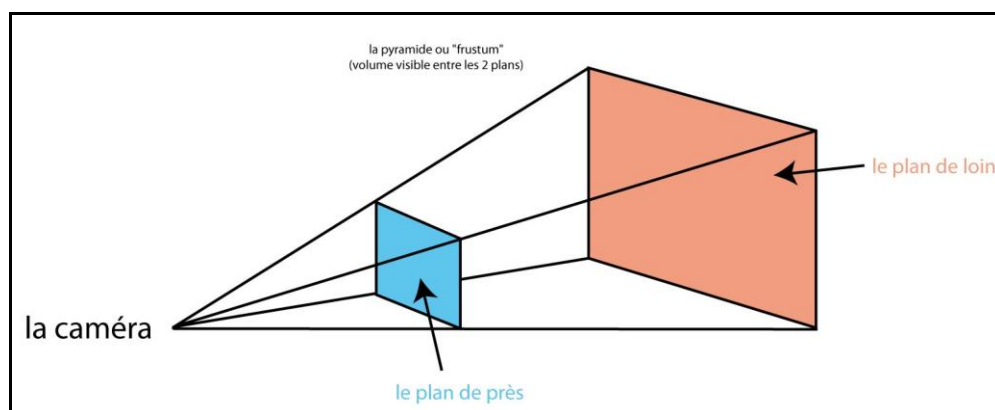
L'utilisation des caméras dans PV3D n'a rien en commun avec celle des caméras classiques comme on peut se l'imaginer. La camera n'est pas visible sur la scène, dans la mesure où la scène représente la vision que l'on a depuis ce point virtuel qu'elle représente.

Les caméras héritent de la classe dsplayObject3D et de ce fait peuvent bouger, tourner, se positionner n'importe où dans l'espace afin de visionner les objets contenus dans son champ de vision avec plus ou moins de détail selon la valeur du zoom.

### 1. Les paramètres :

Voici la liste des paramètres d'une caméra :

- Le zoom,
- Le focus,
- Le champ de vision,
- De près et de loin.



39- Schémas explicatif de l'utilisation d'une caméra

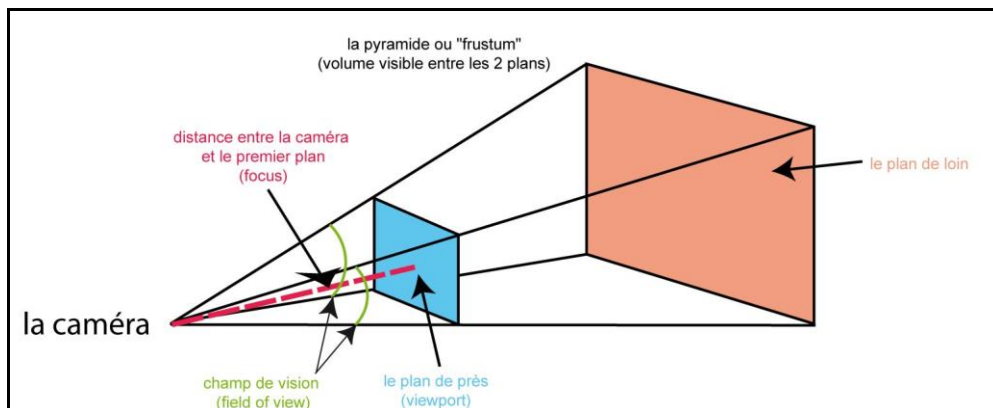
Le **focus** est la valeur numérique qui représente la distance entre la caméra et le premier plan (le plan de près). Ce premier plan représente aussi le « viewport » qui contient la scène avec tous les objets visibles en relation avec le champ de vision.

C'est la plus courte distance possible entre la camera et l'objet à voir.

Le champ de vision est l'angle de vue vertical de la caméra en degrés.

Le schéma ci-dessous montre les différentes propriétés et relations de la caméra.





39- Schéma explicatif du champ de vision et du focus d'une caméra

## 2. Les types de caméras :

Il existe 4 types de caméras incluses dans le Framework PV3D :

- Camera cible,
- Camera libre,
- Camera pour debugger,
- Camera de suivi.

Les deux types de camera les plus utilisés sont les deux premiers.

Nous allons voir en détail chacune d'elles pour voir leurs différences.

### **Camera cible (« ou target camera ») :**

C'est le type de camera par défaut dans PV3D et elle se place au centre de la scène si aucun paramètre lui est défini. Elle se place donc au centre du système global (la scène).

Do3D représente un objet 3D quelconque.

```
camera.target = Do3D ;
```

### **Camera libre (« ou free camera ») :**

Ce type de camera ressemble à la camera libre au détail près qu'elle n'a pas de cible définie. Le champ de vision correspond à ce qui se trouve devant elle.

Cependant, si la camera libre montre un objet et que ce dernier est ou se met en mouvement, l'objet sorti du champ de vision. La camera cible aurait suivi l'objet cible et ne serait alors jamais sorti du champ de vision.

Il est possible de changer de camera cible en camera libre à tout moment. Pour ce faire, il faut alors spécifier la propriété « cible » à « null ».

Pour revenir à une caméra cible, il est nécessaire de définir à nouveau l'objet cible.

```
camera.target = null ; (camera libre)
```

```
camera.target = Do3D ; (camera cible)
```

### **Camera pour debugger (« ou debug camera ») :**

Nous comprenons que dans un espace donnée, lors de la création d'une application, nous puissions ne plus nous rappeler où la camera se trouve, se positionne. Ce type de camera permet en effet de connaître les paramètres courants de la camera dans la scène. Un écran affiche sur la scène ces paramètres en permanence.

### **Camera de suivi (« ou spring camera ») :**

C'est le type de camera qui est utilisée pour suivre un objet comme si celle-ci ne le quittait jamais. Par similitude, nous pouvons illustrer son effet en imaginant le point de vue d'un pilote de Formule 1 assis dans sa voiture qui ne voit son tableau de bord et la route qui défile. Ce type de caméra est principalement utilisé dans la conception d'un environnement de jeu.

## **9. Optimisation de l'application PV3D**

La principale problématique dans la réalisation d'une animation Flash reste à gérer et optimiser la fluidité de l'application tout en conservant une résolution adéquate et

satisfaisante. Pour y parvenir il existe différentes techniques que nous allons voir plus en détail ci-dessous.

Le choix des techniques à utiliser dépend du type d'application PV3D qui est à réaliser. En effet, plusieurs facteurs sont à prendre en considération pour optimiser les performances de l'application et ainsi éviter de surcharger l'utilisation du CPU.

### **1. Définir le niveau de qualité du « Stage » :**

En changeant le niveau de qualité du stage, on peut gagner en performance sans nécessairement diminuer la qualité de ce qui est affiché à l'écran.

Il existe 4 niveaux pour définir le niveau de qualité du Stage (BEST, HIGH, MEDIUM et LOW).

Le niveau « High » est celui par défaut.

Dans la plupart des applications, nous ne devons pas descendre en dessous du niveau « Médium ».

Une pratique fréquente est de définir le stage en « Médium » ou « High » lorsque il s'agit d'une vue statique, et de définir le stage à « Low » lorsque qu'il y a une animation. En effet, il est parfois difficile de voir la qualité des textures lorsque qu'il y a un mouvement, une animation.

Une autre manière consiste à changer la qualité du Stage en fonction de « Frame Rate » permettant de s'adapter aux différentes performances de chacun des ordinateurs qui visionnent cette application. Cependant, cette méthode n'est pas optimisée pour du texte qui risquerait de devenir illisible.

### **2. Détruire les objets inutiles :**

Il est très utile de détruire les objets dont on n'aura plus besoin. Ainsi l'objet peut être supprimé de la mémoire par le GC ou le « Garbage Collector ».

Dans la mesure où PV3D utilise une multitude d'objet bitmap, il est judicieux de nettoyer les objets de l'application qui sont devenus inutiles.

Voici la syntaxe à utiliser :

Remarque : do3D est un objet de la classe DisplayObject3D.

```
scene.removeChild(do3D) ;  
do3D.material.destroy() ;  
do3D = null ;
```

### 3. La taille du « Viewport » :

La taille du « Viewport » influe également sur les performances de l'application. Plus sa taille est grande, plus les besoins en rendu sont importants.

Il est important de ne pas sur dimensionner la taille du « Viewport », elle doit être en relation avec les réels besoins et les limitations de Papervision 3D.

### 4. Le champ de vision de la camera :

Une caméra peut voir tout ce qui se passe sur la scène au même moment mais engendre par ailleurs une plus grosse charge de travail pour le lecteur Flash.

C'est pourquoi il est plus judicieux de limiter le champ de vision de la camera pour diminuer le nombre d'objets à afficher simultanément

Pour une meilleure optimisation de l'application, nous pouvons utiliser les propriétés « fov », « zoom » et « focus » de la camera.

### 5. La Sélection ou « CULLING » :

Cette méthode de sélection identifie les éléments qui sont partiellement ou totalement apparents dans le champ de vision et d'exclure qui n'y est pas.

Cette technique optimise les performances de l'application.

Il faut pour cela utiliser la propriété suivante, pour activer cette technique, camera.useCulling = true.

Ceci permet d'épargner l'utilisation de ressources inutilement, dans la mesure où les objets non vu dans le champ n'ont pas de raison d'être calculés. La mémoire et les ressources de la part du CPU sont moins sollicitées.

## 6. La pensée créative :

L'approche créative requiert une adaptation à chacune des applications. On n'utilisera pas les mêmes techniques d'optimisation pour toutes les applications car chaque application est différente.

Pour optimiser, nous pouvons avoir recours à la falsification (ou « faking 3D »), bien qu'ayant une connotation péjorative, cela permet néanmoins de contribuer à réduire la sollicitation du lecteur Flash pour un gain en performances.

L'utilisation de « Particles » est un exemple pour feindre un ou plusieurs objets en 3D.

## 7. Optimisations des différents types de matériaux :

Si les éléments à afficher ne sont pas optimisés, cela peut accroître la lourdeur de l'application. Le but est de les rendre les plus léger possible ainsi que de les dimensionner au plus juste en fonction des besoins.

Par exemple, l'utilisation d'images en 1024 X 768 n'a pas de sens. Il faut toujours avoir à l'esprit, de travailler dans le bon sens.

La transparence des éléments peut avoir un impact sur la vitesse et la rapidité à afficher le rendu. L'utilisation d'images 32 bits contre 24 bits par défaut doit être utilisée avec parcimonie.

Il est aussi recommandé d'utiliser la propriété « tiled » pour augmenter légèrement les performances de l'application.

L'utilisation de 2 textures active le « mipmapping » et utilise la propriété « smooth » sans perte de performance.

Cependant ceci engendre une utilisation de la mémoire plus conséquente que par défaut.

L'utilisation de ces types de matériels pour des animations requiert davantage de travail pour le lecteur Flash qui, pour chaque rendu, doit créer une image et la

restituer dans la scène en perspective 3D et l'afficher à l'écran. La création de ces images accroît véritablement l'utilisation du CPU. C'est pourquoi, nous devons limiter le nombre d'animations simultanées autant que possible.

### 8. Optimisation des objets :

Plus un objet est complexe, plus il y a de triangles à dessiner. Il est préférable de créer ces objets avec des outils, comme « Blender », avec un minimum de triangle et de polygones, et d'augmenter au fur et à mesure leur nombre pour avoir un objet optimisé. L'inverse représenterait davantage de travail.

Maintenant, prenons le cas où nous sommes dans le champ de vision qui contient un certain nombre d'objets à représenter. Supposons que certains objets se retrouvent cachés par d'autres. Ces derniers n'ont pas besoin d'être dans la scène, car visuellement depuis le point de vue de la caméra, ils ne peuvent être vus. C'est pourquoi nous avons intérêt à les supprimer. Effectivement, PV3D utilise « l'algorithme du peintre » par défaut pour créer tous les objets situés dans le champ de la caméra, ne se souciant pas des objets cachés ou non. Supprimer les objets cachés est un gain en performance considérable surtout si ces derniers sont définis avec un grand nombre de triangles.

Cependant, cela représente un travail conséquent de programmation pour appliquer cette technique.

Le niveau de détail (ou LOD) est une technique qui diminue la complexité des objets 3D en fonction de leur distance avec la caméra.

C'est l'utilisation d'une classe spécifique qui œuvre à mettre en pratique cette dernière.

L'utilisation **d'ombrage (ou shading)** diminue considérablement la sollicitation du CPU.

Pour y parvenir, nous devons ajouter l'objet 2 fois à la scène, le premier représenterait l'image normale ou originale et le second utiliserait une couleur choisie par avance pour simuler un ombrage. Chaque objet serait alors ajouté à des sous-calques de « Viewport » triés par index.

Cette technique donne l'illusion d'ombrage mais ne permet pas de gagner beaucoup en performances.

### 9. La puissance de l'utilisation de 2 textures :

Le « MIP-mapping » est une technologie d'application de textures. C'est une fonction native du lecteur Flash qui, par opération mathématique, crée plusieurs versions de l'image. Cette technique est extrêmement rapide et a un effet non négligeable sur les performances. Les dimensions de l'image n'ont pas un ratio de 1:1 en aspect réel.

Néanmoins quand il n'est pas possible d'utiliser cette technique, on peut utiliser une des fonctions qui est automatique et fournie par le Framework PV3D. Il s'agit d'une propriété statique.

L'ajout de ce paramètre, `BitmapMaterial.AUTO_MIP_MAPPING = true`, permet d'activer le « mip-mapping ».

Lorsque PV3D est configuré ainsi, toutes les images et les textures seront alors corrigées pour permettre le « mip-mapping ».

### 10. Les navigateurs :

Un facteur non négligeable pour la lecture des animations Flash est le navigateur. Hormis Internet Explorer, qui reste le navigateur le plus populaire avec FireFox, il en existe d'autres, toutes plateformes confondues, comme Google Chrome, Safari, Opéra qui offrent de meilleures performances en la matière.





Réalisation d'un musée virtuel générique en 3D avec le moteur PaperVision 3D et le SDK Flex.

Mémoire d'Ingénieur C.N.A.M., Montpellier 2010

---

## **RESUME**

Le Framework PaperVision 3D offre le réel moyen de virtualiser le musée municipal de Frontignan de façon générique et ainsi de générer une animation Flash. Ce choix de technologie par rapports aux autres fut pris en regard du fait que tous les navigateurs actuels lisent les animations flash de façon native et ne nécessite pas d'ajouter un composant supplémentaire pour chacun des navigateurs.

Il faut penser lors de la conception de projet de chercher à optimiser sans cesse les performances de l'application. En effet les performances sont aussi liées à des contraintes extérieures comme les ordinateurs des internautes qui n'ont pas tous les mêmes performances. D'autre part les navigateurs internet n'offre pas les même performances de lecture pour une même animation

**Mots clés : PaperVision 3D, Action Script 3.0, Flex SDK, Flash, primitives, musée virtuel.**

---

## **SUMMARY**

**3D Framework PaperVision offers the real means to virtualize the municipal museum of Frontignan in a generic way and so to generate an animation Flash. This choice of technology by reports in the others was taken compared to the fact that all the current browsers read the flash animations in a native way and do not require to add an additional component, plug-ins for each of the browsers.**

**It is necessary to think during the conception of project to try to optimize ceaselessly the performances of the application. Indeed the performances are also connected to constraints outside as the computers of the Internet users which do not have all even performances. On the other hand Internet browsers do not offer even performances of reading for the same animation.**

**Key words : PaperVision 3D, Action Script 3.0, Flex SDK, Flash, primitives, virtual muséum.**

---