

CadsolOnline

Jean-Luc Astre & Yvon Massé

Résumé :

CadsolOnline est un logiciel en ligne de tracé de cadrans solaires en 3D

Tracés de cadrans solaires classiques, bifilaires et analemmatiques

Présentation de l'utilisation des méthodes de tracé de rayon 'en gnomonique.

Mots clés :

gnomonique, 3D, cadrans bifilaires, cadrans analemmatiques, infographie, ray-tracing, logiciel libre

Introduction

L'informatique graphique est née il y a près de trente-cinq ans, et elle s'est depuis considérablement développée, pénétrant de nombreux domaines comme l'architecture et l'urbanisme, l'audiovisuel, la CAO, la chimie, la médecine, le multimédia, la simulation, la visualisation scientifique ...

Les algorithmes utilisés pour produire des images de synthèse permettent de:

- modéliser les courbes et les surfaces
- gérer l'éclairement, les couleurs, émises ou réfléchies
- créer des texturations sur les surfaces
- réaliser des animations
- calculer l'ombre d'un objet 3D sur un autre objet 3D

Il existe des bibliothèques logicielles qui permettent de calculer facilement l'équation du temps, la position apparente du soleil, les dates des solstices et équinoxes... etc..

Des serveurs en libre accès permettent de connaître les coordonnées géographiques de tout point de la terre, son fuseau horaire, l'heure d'été éventuelle...

**Nous avons donc à notre disposition
tous les outils indispensables au gnomoniste 2.0**



Historique de Cadsol

La version Windows du logiciel de cadrans solaires : « Cadsol » est disponible de puis longtemps sur [SourceForge](https://sourceforge.net/projects/cadsol/), il y a actuellement 30 à 40 téléchargements par mois. Ce n'est pas négligeable, mais c'est quand même peu satisfaisant. Probablement pour les raisons suivantes :

- il faut télécharger le setup, le lancer , puis passer outre aux avertissements signalant que le programme n'est pas validé par une autorité de certification, ce qui peut être inquiétant pour l'utilisateur. Sans compter que Windows et MacOS deviennent de plus en plus restrictifs pour l'installation de logiciels libres.
- le programme ne s'exécute nativement que sous Windows. PlayOnLinux et PlayOnMac ne donnent pas entièrement satisfaction. Et leur utilisation est assez complexe.
- le compilateur (Delphi) utilisé n'est pas libre de droits, c'est une version gratuite qui peut ne plus exister du jour au lendemain. De plus il est bien moins répandu qu'il y a 25 ans.
- la bibliothèque graphique utilisée (GLScene) est open source, mais elle commence aussi à dater.

CadsolOnLine

Le logiciel a été repris avec les techniques et les outils actuels :

- mise en ligne du logiciel sur un serveur web : rien à télécharger, rien à installer, pas de connexion ni de mot de passe, il suffit d'avoir l'adresse du serveur. Les mises à jour sont effectués par le gestionnaire du site. Sans intervention de l'utilisateur.
- le programme est écrit en **JavaScript**. Ce langage de programmation est open-source. C'est un langage objet, évènementiel, non typé, bien normalisé depuis 2019 (par l'ECMA) . C'est un peu verbeux, moins strict que le Pascal ou le C, mais on s'y habitue. L'interpréteur est **disponible nativement sur tous les navigateurs modernes (Chrome, Firefox, Safari, Opera ..), sur tous les système d'exploitation actuels (Windows, Linux, Mac, Android, IOS ...)** et tous les types de terminaux (**ordinateur de bureau, portables, tablettes, téléphones...**), Des outils de développement sont téléchargeables facilement et gratuitement.
- le code source est immédiatement disponible dans le navigateur. Il suffit de faire un clic droit avec la souris, puis sur « télécharger le code source »
- nativement, les menus sont en anglais mais les navigateurs actuels réalisent des **traductions** dans toutes les langues.

Bibliothèques logicielles

Les bibliothèques JavaScript utilisées par CadsolOnLine sont toutes open-source et libre de droits :

- Three.js est une bibliothèque JavaScript pour créer des scènes 3D dans un navigateur web. Les objets sont rangés en arborescence, avec un système parent/enfants. Le matériau par défaut est très simple mais il dispose de nombreuses options pour y ajouter divers effets. pour l'affichage en 3D, les

animations, la gestion des ombres et des lumières, l'import et l'export des fichiers numériques, 3D et 2D...Les primitives de ray-tracing (tracé de rayons) sont disponibles.

- Astronomia** : pour tout ce qui est calcul astronomique (déclinaison du soleil, équation du temps...etc..) Cette bibliothèque est une traduction en JavaScript de l'ouvrage bien connu de **Jean Meeus** : Astronomical Algorithms (Jean Meeus est membre de la SAF),

- L'API lil-dat.GUI** : pour l'interface utilisateur.

Tous ces codes source sont disponibles sur gitHub dans des répertoires en libre accès, avec toutes les fonctionnalités habituelles (chargement et sauvegardes des versions, création de branches, de forks...etc)

Le logiciel, la documentation et l'aide sont actuellement disponibles sur les site :

<https://cadsol.fr> ou <https://cadsolonline.web-pages.fr>

Les sources, et l'environnement de développement sont librement téléchargeables sur github, à l'adresse :

<https://github.com/cadsol/COLMod>

Spécificité des logiciels « en ligne »

CadsolOnLine est un logiciel "en ligne", ce qui signifie que le code (en javascript) est téléchargé depuis le serveur. Ce code est ensuite "interprété" par votre navigateur. En effet les navigateurs (ou Browsers, en anglais) contiennent un logiciel capable de traduire le Javascript en langage machine, ce langage est le seul que votre ordinateur (ou votre smartphone) puisse comprendre et exécuter.

CadsolOnLine a été testé avec :

les navigateurs les plus courants : Chrome, FireFox, Edge, Opera, Safari...

les systèmes d'exploitation : Windows, Android, Linux, Mac, IOS...

des ordinateurs de bureau ou portables, des tablettes, des telephones Android ou Apple...

Le développement a été réalisé avec Chrome. Ce navigateur est donc recommandé (c'est aussi le plus utilisé actuellement). Il est conseillé d'utiliser les versions les plus récentes de votre navigateur et de votre système d'exploitation. (Remarque : le navigateur Internet Explorer, est complètement obsolète, et ne devrait plus être utilisé).

L'avantage des logiciels en ligne, c'est que vous utilisez votre navigateur, celui auquel vous êtes habitué :

- Pour réinitialiser, il suffit de recharger la page .

- Vous pouvez utiliser les commandes habituelles :pour imprimer une page (control P), afficher en plein écran (F11), agrandir ou diminuer la police de caractère (control + ou -) , traduire l'interface dans pratiquement toutes les langues (utiliser l'icone de traduction), afficher le code source (control U), utiliser le clic droit et tous les menus habituels. (Les raccourcis signalés ici sont ceux de Chrome, mais il y a l'équivalent sous tous les navigateurs modernes)

- Vos données personnelles sont préservées (pas d'enregistrement, pas de traqueur, pas de publicité). .
- Les téléchargements que vous pouvez demander sont des fichiers texte, visualisables et modifiables avec tout éditeur de texte simple. Vous les retrouverez dans le répertoire affecté aux téléchargements. Rien d'autre n'est écrit (ou lu) sur votre disque dur.

Les logiciels en ligne ne peuvent être exécutés que par les navigateurs, ceux ci ont de fortes contraintes de sécurité :

- un site web ne peut écrire que sur le répertoire de téléchargement et à la demande expresse de l'utilisateur
- un fichier ne peut être lu que si l'utilisateur demande expressément la lecture d'un fichier particulier, qu'il a lui même choisi.
- l'arborescence des fichiers utilisateur ne peut pas être mémorisé.

Ceci est dans l'intérêt de l'utilisateur. Sans ces contraintes, un site web pourrait lire impunément les données personnelles, effacer n'importe quel fichier (voire tout le disque dur) ou télécharger des logiciels malveillants...

Fonctionnalités gnomoniques

- Heures solaires
- heures légales (avec géolocalisation par adresse, par appel à OpenStreetMap))
- Heures avec correction de l'équation du temps (courbes en huit)
- Heures antiques
- Heures italiques
- Heures babyloniennes
- Heures sidérales
- Arcs diurnes

Cadrans plans classiques à gnomon ou style polaire

algorithmes de J.Meeus, Fer j. De Vries, D.Davoie

La bibliothèque Astronomia contient une procédure relative aux tracés des cadrans solaires plans, déclinant et inclinés. Cette fonction a servi de point de départ au code utilisé dans CadsolOnLine.

Les techniques décrites par Fer de Vries dans son article ([A uniform method to compute flat sundials](#)) ont permis de calculer les différentes heures (solaires, légales, antiques, italiques, babyloniennes , sidérales...etc)

L'ouvrage : La gnomonique - [Denis Savoie](#) a servi constamment de référence générale.

Cadrans bifilaires

algorithmes de D.Collin

Les cadrans solaires bifilaires à gnomons rectilignes quelconques (2002)

Cet article généralise les cadrans bifilaires inventé par H. Michnik à des cadrans déclinants dont les fils ont une orientation quelconque (mais toujours parallèle à la table)

HUGO logiciel pour cadrans bifilaires Eric Mercier & Dominique Collin (2013)

On trouve dans cet article un algorithme très astucieux (l'œuf de Christophe Colomb de la gnomonique !) qui permet de généraliser complètement la théorie des cadrans bifilaires. Les cadrans peuvent être déclinants et inclinés. Les fils ont une orientation quelconque et ils ne sont pas forcément parallèles à la table. Il suffit de donner les hauteurs et les positions de chacun des 4 « piliers » supportant les fils, puis de calculer les coordonnées de l'ombre de ces piliers (par les formules classiques) Cela donne 4 points d'ombre sur le cadran, donc deux droites dont on peut calculer l'intersection par les formules de Cramer.

CadSolOnLine réalise une simulation 3D de ce calcul. Le point obtenu n'est marqué que si l'ombre des deux fils est bien visible au voisinage de leur intersection.

Cadrans analemématiques

algorithme de Y.Massé

Cadrans construits par « ray-tracing »

algorithme de JL ASTRE

On trouve dans l'ouvrage de D.Savoie la remarque suivante :

« Certains gnomonistes ont eu l'idée de d'utiliser une technique simple pour tracer des cadrans sur une surface absolument quelconque. On prend en diapositive une rose horaire ou l'espacement entre les lignes est de 15°. On cale un projecteur de diapositive sur le méridien, en l'inclinant de la latitude du lieu. Il suffit ensuite de projeter l'image de la rose horaire sur la surface pour tracer les lignes horaires »

La bibliothèque THREE.JS permet de réaliser des choses assez semblables en utilisant les méthodes de tracé de rayon. Un solide quelconque étant construit en 3D, il suffit de placer (dans l'espace lié à ce solide) un objet également quelconque qui va projeter son ombre sur le solide.

Dans THREE.JS, la lumière utilisée peut être émise dans une direction spécifique. Cette lumière se comporte comme si elle était infiniment éloignée et les rayons produits par elle sont tous parallèles. Cette lumière peut projeter des ombres semblables à celles du soleil.

CadSolOnLine utilise l'ombre d'un tore dont la position 3D est fixée par l'utilisateur. L'ombre du tore est projetée sur le cadran, de forme et d'orientation quelconque. Il suffit

ensuite de faire varier la direction de la lumière, pour obtenir les lignes horaires, en mémorisant les positions successives du centre de l'ombre du tore.

En pratique, il suffit de choisir dans le menu principal « Ray tracing sundial » puis de déterminer la forme du cadran 3d . Pour l'instant, CadolOnline propose :

Box : on retrouve les cadrans plans classiques, mais obtenus par ray-tracing

Cylinder, Cone, Sphere : tracé des lignes horaires sur la surface extérieure des solides

Equatorial (ou armilaire) : tracés sur l'intérieur d'un demi-cylindre

Scaphe : tracés sur l'intérieur d'un demi-sphère

File 3D... : tracé sur un solide quelconque, le fichier 3D correspondant (obj, ply, stl..) est choisi par l'utilisateur

Les versions ultérieures proposeront d'autres types de cadran

Le cadran peut être orienté dans n'importe quelle direction dans l'espace global 3D, il suffit de préciser les valeurs en degrés des 3 angles : déclinaison gnomonique, inclinaison et rotation (dans l'algorithme, ces angles correspondent aux [angles d'Euler](#))

Le tore peut être placé n'importe où dans l'espace 3D local (lié au cadran) les axes x, y et z peuvent être représentés, pour faciliter le placement du tore.

CadSolOnLine virtualise un rayon lumineux, partant du soleil fictif (à une heure et un jour donné) et passant par le centre du tore. Il faut ensuite calculer les coordonnées du point d'intersection du rayon avec le solide. En infographie les solides sont représentés par un maillage de triangles (mesh en anglais). Il faut donc trouver le « bon » triangle, puis calculer les coordonnées du point d'intersection du rayon avec le triangle. Pour les surfaces planes ou à peu près planes, il suffit de peu de triangles et le calcul est assez rapide. Pour une surface plus complexe, la précision du calcul est liée à la finesse du maillage. Évidemment, si le nombre de triangles est élevé, la précision est meilleure, mais les temps de calcul s'allongent... Les ordinateurs actuels (et même les smartphones) réalisent assez rapidement tous ces calculs.

L'algorithme est détaillé dans l'article de Wikipedia sur le ray-tracing :

https://fr.wikipedia.org/wiki/Ray_tracing

La bibliothèque THREE.JS dispose de toutes les fonctions nécessaires au codage de ces algorithmes.

Code de la procédure de rayTracing de CadsolOnLine:

```
// -----  
  
function CoordRayTracing(hrad,dec) {  
  // hrad: angle horaire en radians de -Pi a Pi  dec: declinaison du soleil en radians  
  
  let rayOrigin = new THREE.Vector3();  
  // origine du rayon lumineux virtuel : centre du tore de corrd. cs.xgnomon, cs.ygnomon cs.zgnomon  
  // en cord. locales, liées au cadran, les axes x, y et z sont visibles sur l'écran  
  let rayDirection = new THREE.Vector3(); // direction rayon (vecteur de norme 1)  
  let coordRayCaster = new THREE.Vector3(); // coord. de l'ombre tore sur le cadran en coord.  
  locales (c'est le pont 3D à trouver)  
  
  const k = 1200; //constante arbitraire  
  
  rayOrigin.set( cs.xgnomon / cadran.scale.x,  
                cs.ygnomon / cadran.scale.y,  
                cs.zgnomon / cadran.scale.z)  
  // ces facteur d'echelle sont liés au dimensions d cadran  
  cadran.localToWorld(rayOrigin)  
  // changement de repere, il faut passer aux coord. locales aux coord. globales  
  
  sphereSun.position.z = k*Math.cos(dec) * Math.cos(-hrad); // calcul des coord. du soeil virtuel  
  sphereSun.position.x = k*Math.cos(dec) * Math.sin(-hrad);  
  sphereSun.position.y = k*Math.sin(dec);  
  sphereSun.getWorldPosition(rayDirection) //memorisation dans le vecteur 3D rayDirection  
  
  rayDirection.negate(); // changement de sens, le vecteur pointe vers l'origine du repère global  
  rayDirection.normalize(); // normalisation  
  
  rayCaster.set(rayOrigin, rayDirection)  
  // l'objet rayCaster a été initialisé au début du code par :  
  // const rayCaster = new THREE.Raycaster() (variable globale)  
  // rayCaster virtualise le rayon lumineux ayant direction du soleil (rayDirection)  
  // et passant par de centre du tore (rayOrigin)
```

```
const intersects = rayCaster.intersectObject(cadran);
```

```
// c'est la ligne la plus importante du code: recherche de l'intersection du rayon et du cadran
```

```
// le cadran est modélisé par un maillage de triangles
```

```
if (intersects.length > 0) {
```

```
// s'il y a au moins une intersection, les coord du point sont dans intersects[0].point
```

```
    coordRayCaster = cadran.worldToLocal(intersects[0].point)
```

```
    // on revient aux coord locales
```

```
    coordRayCaster.set(coordRayCaster.x * cadran.scale.x - cs.xgnomon,
```

```
                        coordRayCaster.y * cadran.scale.y - cs.ygnomon,
```

```
                        coordRayCaster.z * cadran.scale.z)
```

```
    // il reste à remettre à l'échelle, et changer d'origine
```

```
    return {
```

```
        x: coordRayCaster.x, //retour des coord de l'ombre, en coord locales, ok=true
```

```
        y: coordRayCaster.y,
```

```
        z: coordRayCaster.z,
```

```
        ok:true
```

```
    }}
```

```
    else return {          // si pas d'ombre, ok=false
```

```
        x: 0,
```

```
        y: 0,
```

```
        z: 0,
```

```
        ok:false}
```

```
}
```


Importation et exportation

Exportation :

Les cadrans solaires réalisés avec CadsolOnLine sont exportés au format json (JavaScript Object Notation). C'est un format texte, donc facilement lisible et éditable, qui permet de représenter de l'information structurée.

Pour exporter un fichier :

- ✓ Ouvrir le sous-menu : Exporter
- ✓ Choisir un nom de fichier
- ✓ Choisir le type de fichier : Cadran solaire (JSON)
- ✓ Cliquer sur le bouton : Télécharger

Les fichiers exportés sont enregistrés dans le répertoire de votre disque dur affecté aux téléchargements. Vous pouvez ensuite les copier dans le répertoire de votre choix.

Importation :

Ces fichiers json peuvent être ouverts avec la commande de la page d'accueil :

‘Ouvrir le cadran solaire...’

Vous pouvez ouvrir n'importe quel répertoire de votre disque et lire le fichier json que vous y avez enregistré. Si ce fichier contient bien un cadran solaire, il sera ouvert par CadsolOnLine.

Autres type d'exportation :

Feuilles de calcul (CSV) : pour exporter les coordonnées de tous les points de tracé dans un tableur.

3D (PLY, OBJ, STL...) : pour générer des fichiers lisibles par tous les éditeurs 3D (MeshLab, Blender, FreeCad, 3D builder...etc), imprimables directement par toutes les imprimantes 3D

Ephémérides : exportation d'un fichier CSV contenant les éphémérides de l'année (déclinaison du soleil, équation du temps, jour julien, solstices et équinoxes...)

Scalable Vector Graphic (SVG) : exportation des tracés sous forme vectorielle, lisibles par tous les logiciels de dessin vectoriel (Inkscape, Adobe Illustrator...) ainsi que par les logiciels de commande des imprimantes 3D, fraiseuses numériques, traceuses laser...etc..

HTML : génération d'une fiche résumant des propriétés du cadran (en anglais, mais le navigateur peut traduire dans une langue quelconque)

Remarques : tous les navigateurs gèrent directement

- ✓ les impressions 2D (sur plusieurs pages, si nécessaire)
- ✓ la génération des fichiers PDF
- ✓ la lecture et l'impression des fichiers SVG.
- ✓ La lecture et l'impression des pages HTML
- ✓
- ✓

Bibliographie

Livres

- La gnomonique - [Denis Savoie](#) Les Belles Lettres (2001)
- Midi au Soleil - Jean Fulcrand & Pierre Bourge Editions P.Bourge (1978)
- Astronomical library - [Jean Meeus](#) Atlantic Books; (1998)
- Tout JavaScript 2em édition - Olivier Hondermarck DUNOD (2021)

Publications

- [A uniform method to compute flat sundials](#) - Fer j. De Vries (2002)
- [Cadrans solaires bifilaires](#) - Dominique Collin (2002)
- [HUGO logiciel pour cadrans bifilaires](#) - Dominique Collin & Eric Mercier (2013)

Librairies JavaScript

- [Javascript astronomical Library](#) - [Sonia Keys](#)
- [THREE3D](#)
- [lil-gui](#)

Serveur cartographique

- [OpenStreetMap](#)

Licence

CadsolOnLine est publié sous licence **CECILL 2.1**

(Ce[a] C[nrs] l[nria] L[ogiciel] L[ibre])

Distribuer un logiciel libre n'est pas renoncer à tout droit sur le produit : l'auteur comme l'utilisateur ont des droits, des devoirs et des responsabilités propres qui sont définis par la licence attachée au logiciel et la loi applicable. A ce jour, la majorité des logiciels libres français sont diffusés sous licences anglo-saxonnes, notamment la GNU General Public License (GNU GPL), car le logiciel libre s'est d'abord développé aux États-Unis. Ceci pose un certain nombre de questions juridiques délicates.

La démarche du CEA, du CNRS et de l'INRIA est d'élaborer un ensemble de licences adaptées au droit français et compatibles avec les principales licences libres anglo-saxonnes dont elles reprennent les principes. Deux critères président à leur élaboration :

- Le respect des principes de diffusion du logiciel libre. . L'exercice de ces libertés est assorti de certains devoirs à la charge des utilisateurs afin de préserver le caractère libre du logiciel, au cours des redistributions ultérieures.
- Leur conformité au droit français sous deux aspects, responsabilité civile d'une part et propriété intellectuelle d'autre part.

Copie d'écrans de CadsolOnLine

Photos : voir les fichier jpg.

Légendes : voir les fichiers txt