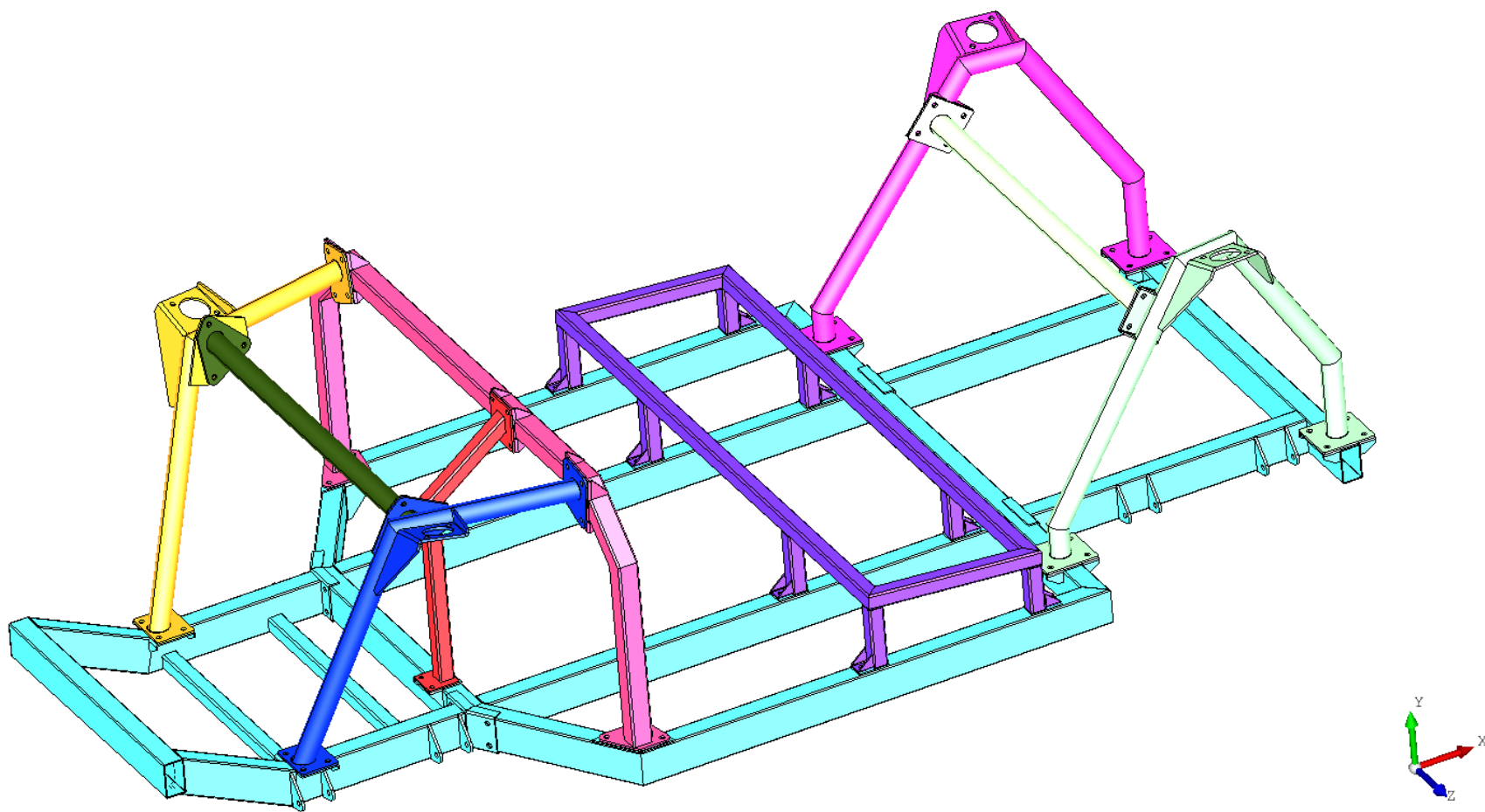
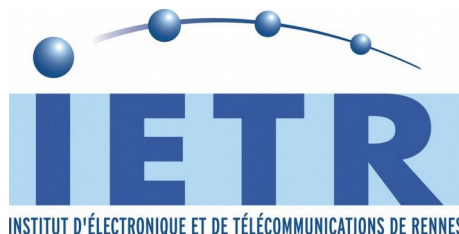


Modélisation Open Source d'objets complexes avec Python



L'équipe et ses partenaires

IETR / Bernard Uguen et Valérie Guichon



PythonOCC / Thomas Paviot



Rennes Métropole

FEDER



... vous

Un peu d'histoire ...

- | | |
|-----------------------|--|
| Avril 2016 | Premier CampOSV
Constatation des problèmes de modélisation (fichiers Open Source mais difficilement exploitables) |
| Mars 2017 | Coup d'envoi et début du développement
Vers un SI autour de l'OSV |
| Décembre 2017 | Présentation PyOSV au Paris Open Source Summit (POSS) |
| Mars 2018 | inOut / CampOSV / Hackathon |
| Septembre 2018 | CampOSV / Hackathon ← (Vous êtes ici) |

Quel problèmes à résoudre ?

Objet complexe + librairies de pièces + PLM + collaboratif → logiciels propriétaires (fermés) et chers

Ticket d'entrée élevé pour les petites structures

Modélisation d'objets complexes

- assemblage de pièces
- assemblage de pièces avec des assemblages
- assemblage d'assemblages
- documentation et lifecycle management

Modélisation collaborative et hétérogène

- contributeurs éparpillés géographiquement
- moyens de conception (logiciels CAO / librairies CAO) différents

Pièces standard / Librairies

- ensembles cohérents réservés au logiciels ... chers !

Formats neutres (STEP, IGES) sans 'intelligence', pas de capture du 'design intent'

Nomenclature (*osvcad/doc/nomenclature*)

Les idées

Idées générales

Open Source, évidemment

Appliquer les meilleures pratiques du développement logiciel au développement hardware

Idées techniques

Modélisation d'un objet complexe sous forme de graphe (acyclique dirigé)

Pas de limitation théorique (possibilité d'introduire des nouveaux types de noeuds et de liens)

Modélisation programmatique

Origines multiples (pièces standard, STEP, scripts)

Les ancres et les 'stezips'

Appliquer les meilleures pratiques logicielles au développement hardware : Use case cible

- | | |
|---|---|
| 0. Je crée mon environnement de développement | <code>./install_pyosv.sh</code> |
| 1. Je crée ou je clone un projet à partir de la plateforme pyosv | <code>git clone http://pyosv.org/<user>/<project></code> |
| 2. Je modifie le code de création des pièces, des assemblages etc ... | |
| 3. Je vérifie mes modifications grâce à l'outillage fourni par l'environnement | <code>pyosv-view <project></code> |
| 4. Je commite mes modifications | <code>git add && git commit -m ... && git push</code> |
| 5. La nouvelle version est en ligne et je peux visualiser en ligne le graphe et la vue 3D | |

... bref, le workflow open hardware est très proche du workflow logiciel open source

Modélisation sous forme de graphe acyclique dirigé

Des noeuds:

- pièce
- assemblage
- *documentation*
- *simulation*

Des liens :

- contraintes géométriques (e.g. `osvcad.edges.ConstraintAnchor`)
- *documente*
- *contient des résultats de simulation*

Modélisation programmatique et origines multiples

L'objet complexe est **modélisé par un script Python**.

Le graphe et la vue 3D sont des résultats (/des vues) de ce que produit le script.

Les plans de fabrication et la documentation d'un projet sont des exports.

Le script Python de modélisation 'ne fait que' créer la description de l'objet complexe à partir de noeuds et de liens.

Origines multiples

Un noeud 'géométrique' peut être créé à partir d'un script (Python), d'un fichier STEP ou d'une librairie de pièces. Une fois le noeud créé, son origine n'importe plus.

Les ancrs et les Stepzips

Qu'est-ce qu'une ancre ?

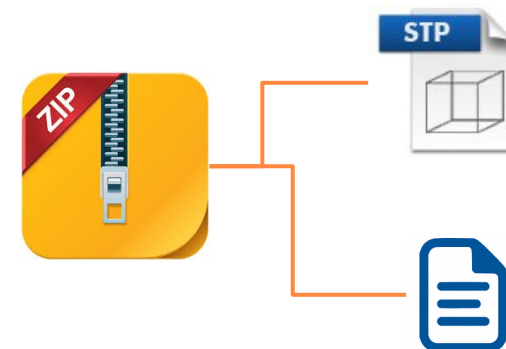
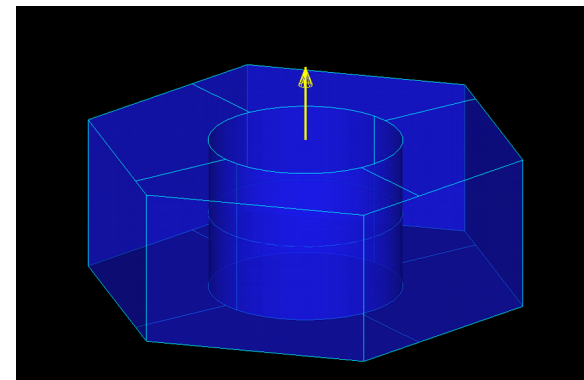
C'est un point et un vecteur attaché à une géométrie.

La plupart des pièces ont des points, des axes, des plans d'appui particuliers prévu pour l'assemblage.

Les ancrs permettent de placer des pièces les unes par rapport aux autres de manière programmatique

Qu'est-ce qu'un Stepzip ?

C'est un fichier zip contenant un fichier STEP et un fichier décrivant ses ancrs.



<https://github.com/osv-team>



pyosv Point d'entrée

pyosv-binderhub Notebooks online

osvcad Modélisation des objets complexes (assemblages, ancrs, graphes ...)

ccad Construction et manipulation scriptée d'objets simples (fork du projet de Charles Sherman)

party Génération et utilisation des librairies de pièces

standard-cad-parts Librairie de pièces générées à la volée

reversy Décomposition de fichiers STEP en pièces élémentaires (positionnées dans leur repère propre)

quaternions Implémentation des quaternions

<https://github.com/osv-team>

osvcad

UI !

osvcad2

Modélisation plus complète et ré-écriture des ancres et des contraintes

reversy

Reverse engineering des fichiers STEP

opm

Open Product Model

freecad-workbench-anchors

Faciliter la création des ancres

OSV

Modélisation de l'OSV

openplm + plugin

PLM old school

Qualité projet, Python 3.6, Dockerisation de l'environnement, Travis (Linux), Codacy

Newsletters



Roadmap

Faire sauter le bouchon des ancres !

Nouveaux types de noeuds et de contraintes + solveur de contraintes

Plateforme collaborative

Enrichir librairies de pièces standard



Vos remarques

Vos questions

Vos idées



LET'S TALK