

BIND

Une boîte à outils pour l'analyse de données
expérimentales

Arnaud Bonnard, Damien Sornette

4 octobre 2010

Le LESCOT

- Laboratoire de sciences humaines
- Analyse du comportement des « conducteurs » et de leur activité de conduite
- Différents outils pour recueillir des données comportementales



Problématique

- Différents outils de recueil
- De nombreuses expérimentations
- D'énormes quantités de données

Comment formaliser le processus d'analyse des données ?

Problématique

Comment formaliser le processus d'analyse des données ?

- Des processus d'analyse
 - Accès
 - Visualisation
 - Enrichissement
 - Extraction
- Des données
 - Paramètres continus
 - Données événementielles
 - Données situationnelles

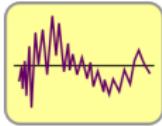
Approche

- Définition des concepts métiers
- Définition des contraintes
- Choix d'un environnement logiciel

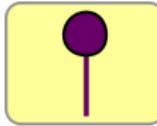
Définition des concepts métiers

- Notre élément unitaire est un trajet ou « trip » composé de trois types de données :

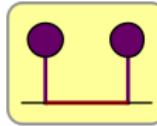
Data



Event



Situation



Définition des contraintes

- Répondre aux questions :
 - Qu'est ce qu'il y a dans les données ?
 - Comment est ce qu'on va analyser nos données ?
- On a aussi besoin de :
 - Connaître
 - Partager
 - Capitaliser

Définition des contraintes

Connaître

- Les expérimentations sont de nature variée
- On collecte des données hétérogènes en provenance de sources différentes

Système auto-descriptif

Définition des contraintes

Partager

- Ce qui a été réalisé doit
 - être transférable
 - être utilisable par d'autres
 - rester utilisable longtemps après

Système solidement documenté

Définition des contraintes

Capitaliser

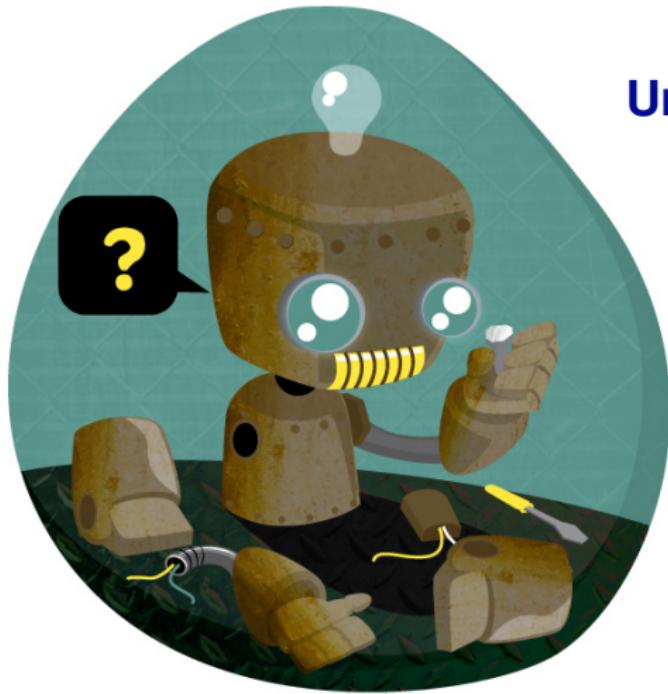
- On a besoin de s'appuyer sur nos expériences pour mieux se focaliser sur l'innovation

Système porteur de nos compétences techniques

Définir un environnement logiciel

- Déjà connu au sein du laboratoire
- Usuel dans le monde scientifique
- Suffisamment polyvalent

Matlab



Une problématique...

... des contraintes...

... une solution !

Vers une solution

- Un framework logiciel pour
 - Répondre aux contraintes
 - Implémenter les compétences métier
 - Standardiser les développements du laboratoire

Un framework logiciel

- Qu'est ce que c'est ?

- Boite à outils logicielle
- Squelette applicatif de référence
- Architecture cible



BiND

BIND

- Se prononce « *Baïnde* »
- « To bind » : *lier* ou *attacher*

**BIND créé le lien entre les données et les utilisateurs,
via ses outils**



**Quelles sont les
fonctionnalités de BIND ?**

BIND

- Une structuration métier et autodescriptive des données



BIND

- Une structuration métier et autodescriptive des données
- Un moteur d'accès aux données

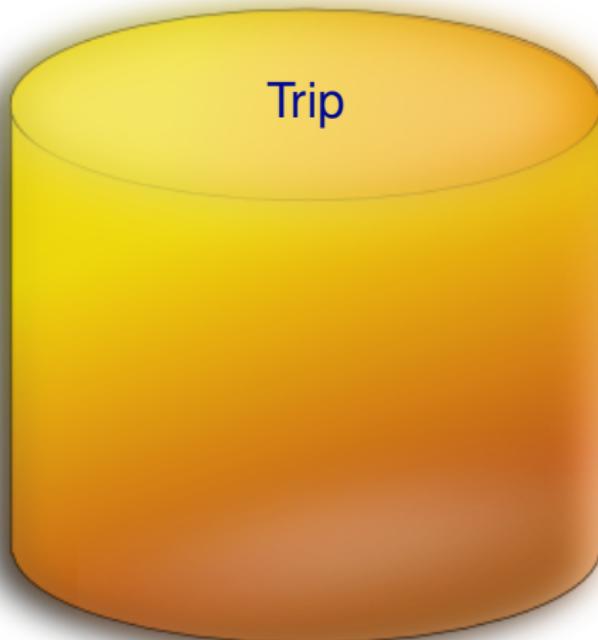


BIND

- Une structuration métier et autodescriptive des données
- Un moteur d'accès aux données
- Une architecture de modules graphiques synchronisables



Structure des données



Structure des données

- L'élément unitaire est le **trip**
- Un trip est décrit par :
 - des Attributes
 - des informations sur le participant
 - des informations sur les vidéos



Structure des données

- Les données contenues dans le trip sont décrites par :
 - des MetaDatas
 - des MetaEvents
 - des MetaSituations



Structure des données

- Le trip contient également les données
 - Datas(timecode, variables)
 - Events(timecode, variables)
 - Situations(startTimecode, endTimecode, variable)



Structure des données

Exemple

MetaDatas

- GPS
 - timecode
 - latitude
 - longitude

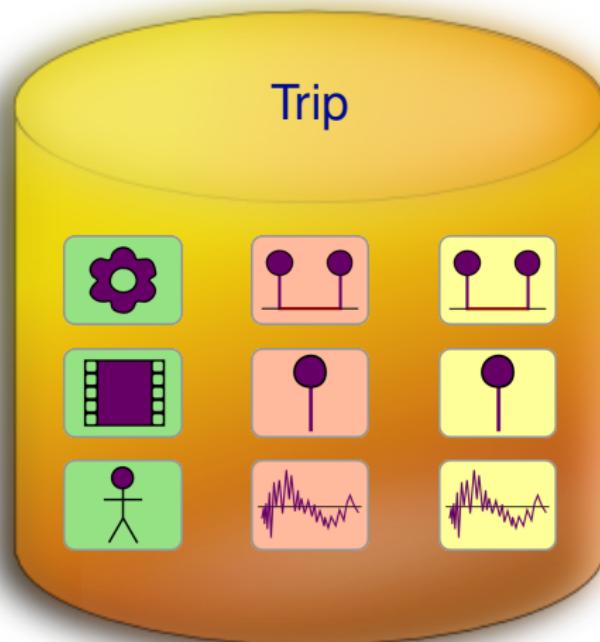
Data

timecode	latitude	longitude
0	45.5698	31.2587
0.1	45.4501	31.2590
0.2	45.5890	32.7750
...

- Données capteurs
 - timecode
 - accélérateur
 - frein

timecode	accélérateur	frein
0	10	0
0.1	13.5	0
0.2	0	21.2
...

En résumé



Moteur d'accès aux données



Moteur d'accès aux données

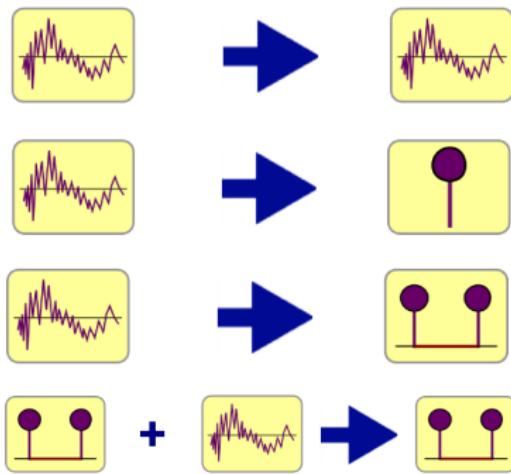
- BIND prend en charge l'accès aux données via son interface
- On ne touche **jamais** aux données sans passer par le framework
- L'implémentation de l'interface logicielle prend en charge les fonctions de bas niveau

Sauvegarde des trips

- BIND définit une interface pour les formats de données
 - Tout format respectant l'interface est dit BINDable
- Une implémentation de cette interface existe : **SQLite**
- On peut en imaginer d'autres

Traitement des données

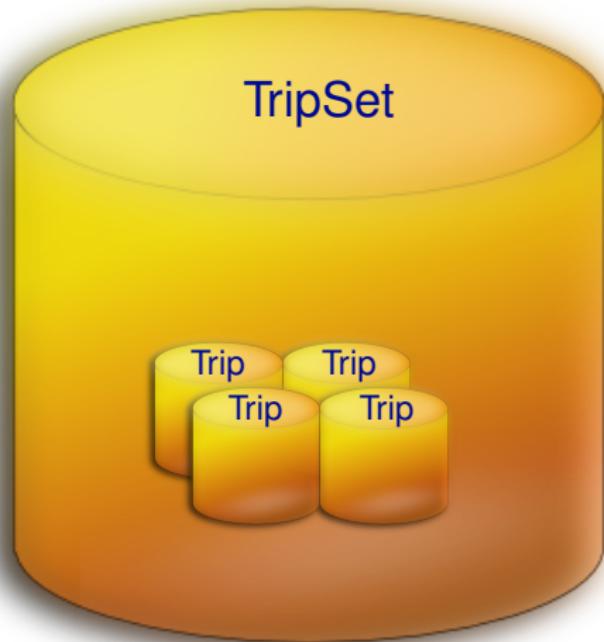
- BIND prend en charge certains mécanismes de traitement des données



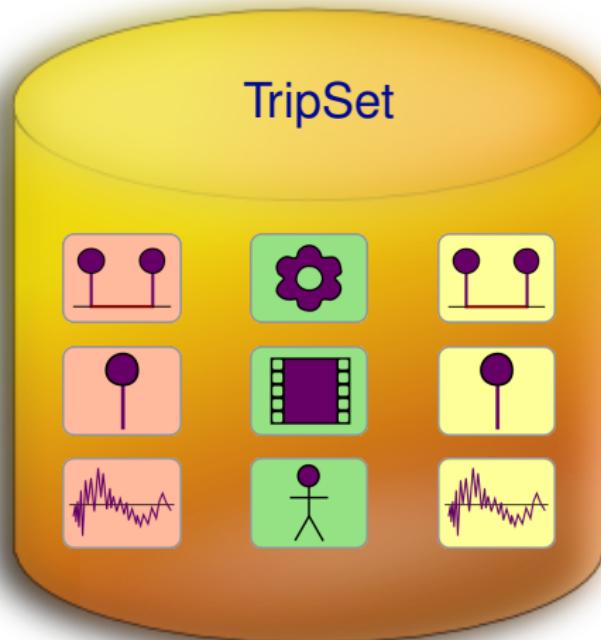
TripSet

- Permet d'agréger des « trips »
- Liens vers les trips
- Union et intersection des informations des trips
 - Attributes
 - MetaInformations

TripSet



TripSet



Modules graphiques



Modules graphiques

- Objets de base pour la création d'applications graphiques
 - Pas de plugins inclus dans BIND
- Mécanisme de synchronisation
 - Chaque trip est doté d'un timer interne
 - Utilisation du design pattern observer / observable



BIND

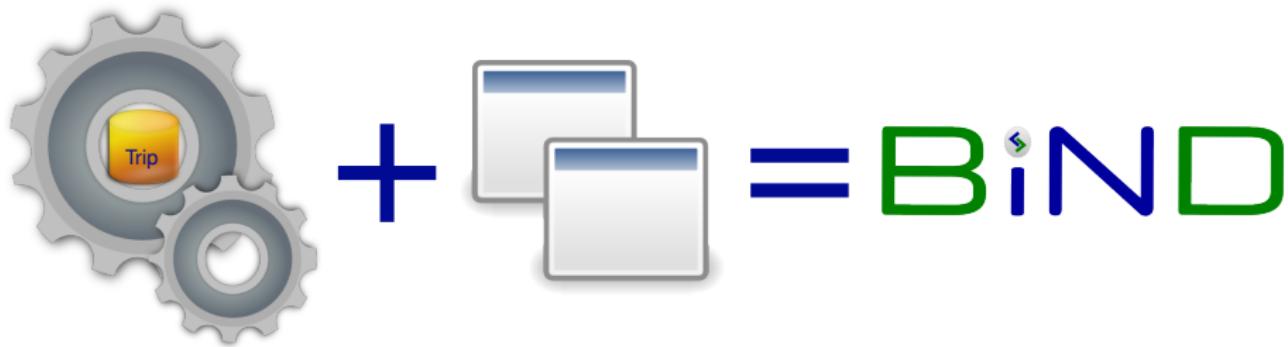
Paradigme et usages

- Création de 2 types d'applications métier

```
    . or ever
    refreshdir=0
EndIf
Color bgr,bgg,bgb:Rect 0,0,640,640
If showbgimage=1 Then TileImage by image
Color schriftr,schriftg,schriftb:Text
If mode=0 Then Text 518-81Len("Version")
If fillbox=1 Then Color box,bgg,bgb
Color borderr,borderg,borderb
Color schriftr,schriftg,schriftb:Text
Color box,bgg,bgb
If fillbox=1 Then Color box,bgg,bgb
Color borderr,borderg,borderb
Color schriftr,schriftg,schriftb:Text
If fillbox=0 Then Color box,bgg,bgb
If selection=0 Then Color selec,selec,selec
Color borderr,borderg,borderb
If Len(trypaders$)>67 Then
    showPaders$=Mid$(trypaders$,1,34)
    trypaders$=Left$(trypaders$,Len(trypaders$)-34)
    doc$=trypaders$
```



Synthèse





Qu'est-ce que ça change ?

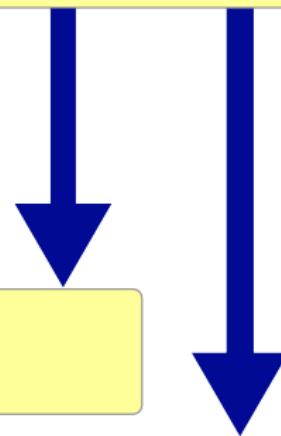
Rien... et tout

- On continuera de développer
 - des scripts de traitement de données
 - des logiciels de visualisation
 - des outils de modélisation, de simulation
 - ...

Mais on les développera en utilisant BIND, ou on les connectera à BIND

Avant

Applications métiers



Matlab

OS (Windows, Linux, ...)

Après

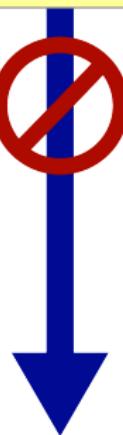
Applications métiers



BIND

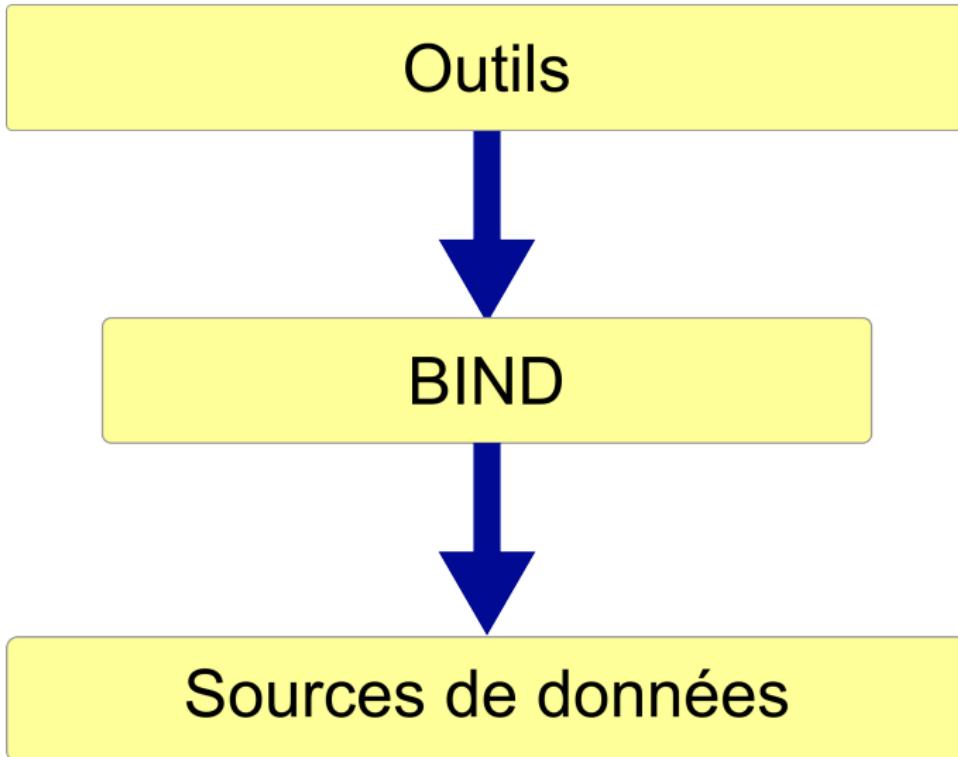


Matlab



OS (Windows, Linux, ...)

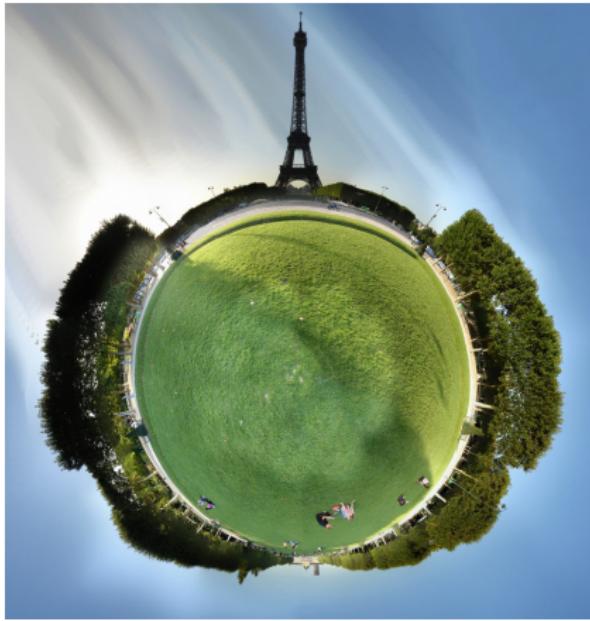
Démarche de conception



Finalement... BIND

- Fait plein de choses... mais ne fait rien !
- Sert à créer des applications et des outils métiers

La façon de faire change



BIND et son environnement

Lecture des données

- Format standard et ouvert
 - Partage de données
 - Utilisation avec d'autres outils
- SQLite
 - Visualisation rapide avec SQLite Manager

The screenshot shows the SQLite Manager interface with the database 'trip_park01.db' open. The left pane displays the table structure for 'SensorsMeasure'. The right pane shows the data for the 'SensorsMeasure' table, which has columns: rowid, timecode, timestamp, pulse, Brake, Acceler., Clutch, and Accr. The data consists of 16 rows of sensor measurements.

rowid	timecode	timestamp	pulse	Brake	Acceler.	Clutch	Accr
1	111.0787	3292204	0	0	0.02	-0.16	0.01
2	111.1187	3292204	0	0	0.02	-0.16	0.01
3	111.1387	3292204	0	0	0.02	-0.16	0.04
4	111.1787	3292204	0	0	0.01	-0.16	-0.03
5	111.1987	3292204	0	0	0.01	-0.16	0
6	111.2387	3292204	0	0	0.01	-0.16	0.01
7	111.2587	3292204	0	0	0.01	-0.16	-0.02
8	111.2787	3292204	0	0	0.01	-0.16	0.01
9	111.2987	3292204	0	0	0.02	-0.16	-0.03
10	111.3387	3292204	0	0	0.01	-0.17	-0.02
11	111.3787	3292204	0	0	0.01	-0.16	-0.03
12	111.3987	3292204	0	0	0.02	-0.16	0
13	111.4387	3292204	0	0	0.01	-0.16	0.01
14	111.4587	3292204	0	0	0.02	-0.16	-0.03
15	111.4787	3292204	0	0	0.01	-0.16	0.01
16	111.5187	3292204	0	0	0.02	-0.16	0

Documentation

Pour les développeurs

BIND	fr.lescot.bind.data.Record
Doe	handle
Src	
CONFIGURATORS	
DATA	
MetaBase	
MetaData	
MetaEventData	
MetaEvent	
MetaEventVariable	
MetaInformations	
MetaParticipant	
MetaSituation	
MetaSituationVariable	
MetaVariableBase	
MetaVideoFile	
Record	fr.lescot.bind.data.Record
KERNEL	
LOADING	
OBSERVATION	
PLUGINS	
PROCESSING	
UTILS	
INDEX	
Search	

This class represents a result from a database request. It is implemented as a struct array, to avoid costly operations when converting from sqlite4m output. Efficient, but hard to manipulate, because some syntaxes aren't very well documented in Matlab.

Summary

fr.lescot.bind.data.Record This class represents a result from a database request.

PROPERTIES

PRIVATE

- data** A struct array with fields representing the Variables
- isEmptyProperty** A logical marking whether the Record is empty or not

FUNCTIONS

PUBLIC

- buildCellArrayWithVariables** Builds a $n \times n$ cell array, where n is the number of elements passed on variablesList.
- getVariableNames** Returns a cell array of strings containing the names of the variables available in this Record.
- getVariableValues** Retrieves the values stored in record for the given variable name (case insensitive search).
- isEmpty** Returns a logical that represents the fact that the number of elements in the Record is 0 or not..
- Record** Constructor of Record.

PROPERTIES

PRIVATE

private data

A struct array with fields representing the Variables

private isEmptyProperty

A logical marking whether the Record is empty or not

FUNCTIONS

Connecter un logiciel à BIND ?

- BIND parle le « XMPP »
 - Protocole de communication réseau
 - Service de « chat »
- Grâce à Matjab
 - Bibliothèque XMPP pour Matlab
 - Connexion vers applications tierces (autres OS, autres langages...)

BIND et logiciels tiers

Applications tierces :

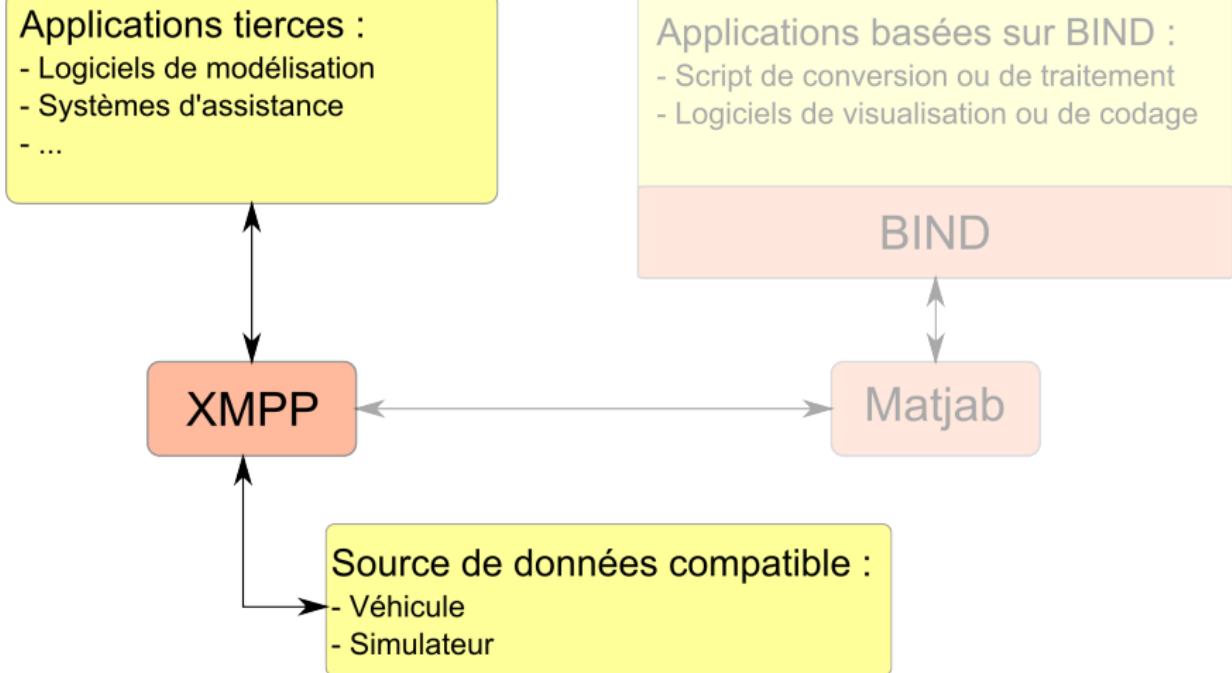
- Logiciels de modélisation
- Systèmes d'assistance
- ...

Applications basées sur BIND :

- Script de conversion ou de traitement
- Logiciels de visualisation ou de codage



BIND et logiciels tiers



Distribution

- On peut facilement distribuer des logiciels bâtis sur BIND
- Création d'application autonomes
 - Possibilité d'empaqueter un ensemble de scripts
 - Possibilité d'empaqueter des interfaces graphiques

Partage des outils



**Est-ce que BIND répond à
notre problématique ?**

Réponse à la problématique

- BIND : Une boîte à outils pour l'analyse de données expérimentales
- Utiliser BIND permet de
 - Guider l'activité de conception
 - Réutiliser les développements
 - Faciliter l'échange d'outils et de données

Formalisation des processus d'analyse des données



Comment travaille-t-on avec BIND ?

Pour démarrer

- Ingrédients

- 1 licence Matlab (>=2009b)
- 1 code source de BIND
- 1 documentation
- Quelques connaissances en programmation

- Recette

- Lire la documentation
- Lire la documentation
- Lire la documentation
- Lire les exemples

Cas d'utilisation

Script Matlab

- Dérivation d'un signal
 - Ouverture du fichier contenant le trip
 - Sélection des variables pertinentes
 - Application du traitement
 - Sauvegarde du résultat
- En utilisant les outils de BIND : **5 lignes**

Cas d'utilisation

Script Matlab

```
① theTrip =  
fr.lescot.bind.kernel.SQLiteTrip(file,0.04, false);  
  
② dataRecord =  
theTrip.getAllDataOccurrences ('DonneesBrutes');  
  
③ signalToProcess =  
dataRecord.buildCellArrayWithVariables ('timecode'  
'%Acc');  
  
④ signal = Derivator.process(signalToProcess,'50');  
  
⑤ theTrip.setBatchOfTimeDataVariablePairs  
( 'DonneesTraitées' , 'Derivee%Acc' , signal );
```

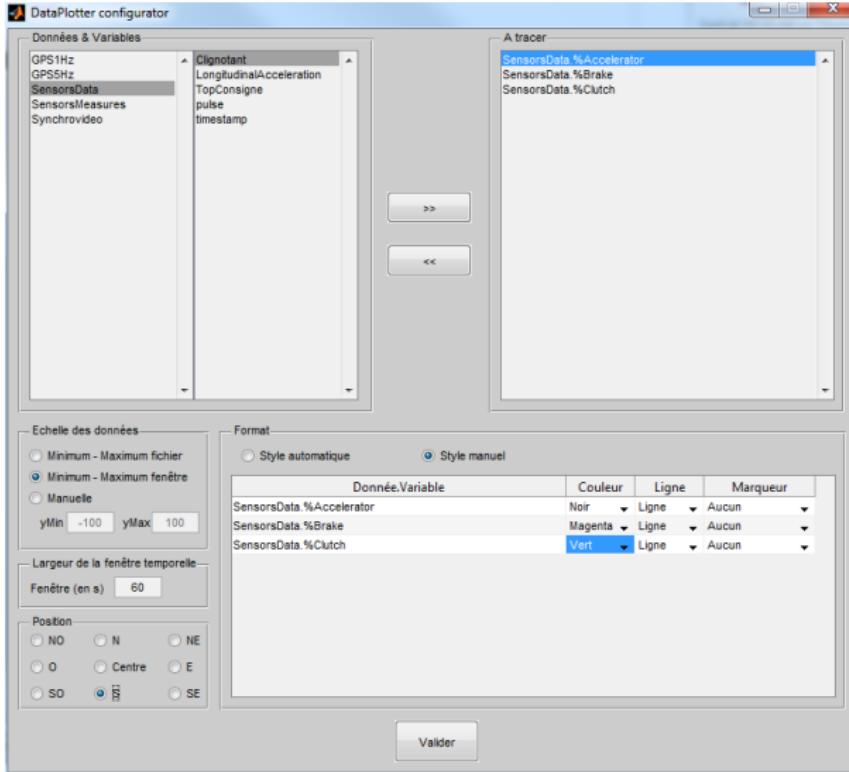
Cas d'utilisation

Interfaces graphiques Matlab

- Environnement de visualisation de données
 - Lecture audio/vidéo
 - Affichage de courbes
 - Contrôle du temps de la visualisation
- Avec BIND : 3 modules
 - Videoplayer (200 lignes)
 - DataPlotter (600 lignes)
 - Magneto (500 lignes)

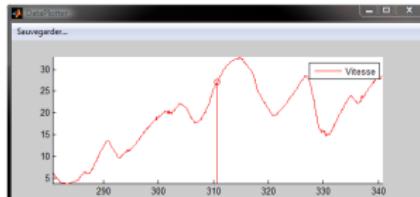
Cas d'utilisation

Configurateurs de plugins



Cas d'utilisation

Plugins





Un autre exemple ?

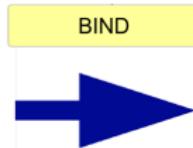
Exemple d'application

- Objectif
 - Différences entre 2 populations de conducteurs
- Démarche
 - Utilisation de données du véhicule instrumenté
 - Situations de conduite en intersection
 - Calcul de paramètres agrégés
 - Export vers un logiciel de statistique

Exemple d'application (1/7)

Utilisation de données provenant du véhicule instrumenté

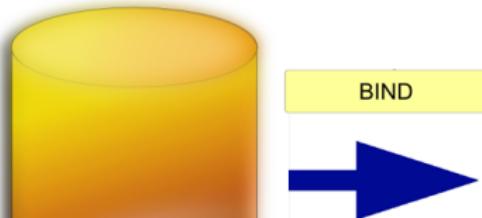
- Elaboration d'un script de conversion de données



Exemple d'application (2/7)

Visualisation du parcours

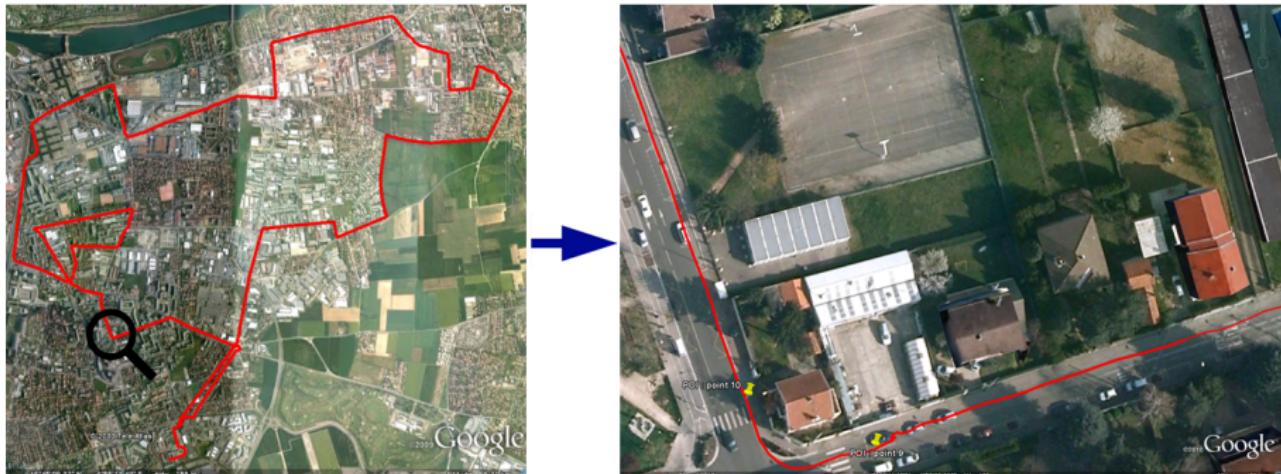
- Script d'export pour Google Earth



Exemple d'application (3/7)

Localisation des intersections

- Script de calcul de distance entre coordonnées GPS véhicule et points de référence
- Création des events POI



Exemple d'application (4/7)

Définition des situations de conduite en intersection

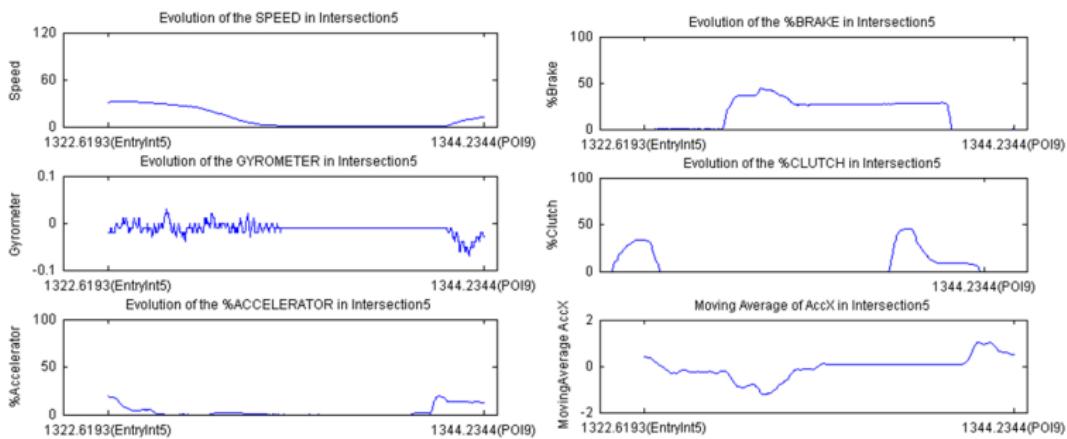
- Script utilisant les events POI et la distance parcourue
- Crédit des situations Intersections et BetweenIntersections



Exemple d'application (5/7)

Edition de courbes pour étudier les situations

- Création automatique d'un jeu de courbes pour chaque situation



Exemple d'application (6/7)

Calcul de paramètres agrégés pour chaque situation

- Utilisation des SituationAggregator
- Enrichissement des situations Intersections avec les valeurs de vitesse moyenne, l'écart type...



Exemple d'application (7/7)

Valeurs des paramètres agrégés pour tous les participants

- Utilisation du TripSet
- Création d'un fichier Microsoft Excel compatible SPSS

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Trip Number	Driver Code	Driver Status	entryInt5	int5	exitInt5	entryInt7	int7	exitInt7	entryInt18	int18	exitInt18	entryInt21
2	1	7	1	-7,2972	130,2753	-5,3082	-1,0616	97,1247	11,4027	10,3172	215,1098	-11,0745	-2,3578
3	2	2	1	-0,5695	116,0793	-2,0775	-3,4157	90,6929	19,37	161,3487	334,1773	-4,5166	6,4648
4	3	1	0	-0,3523	140,1816	-0,0747	2,0363	100,2037	6,4885	-21,2379	205,6577	-3,6881	6,2544
5	4	3	0	-0,4855	112,6456	5,2835	2,9738	98,1411	6,4935	-9,5618	236,2751	-5,3498	4,4021
6	5	1	1	3,6216	130,9991	5,6753	-5,4811	95,3141	23,9789	141,9831	242,1799	-3,4023	17,9251
7	6	22	0	-2,9183	105,2692	1,3569	-1,5936	100,7933	2,7093	-0,0093	230,8174	-11,891	0,0392
8	7	17	1	-1,4351	105,0707	-4,5614	-1,4875	83,1667	9,3767	76,2905	227,7859	-4,4656	2,584
9	8	3	1	-7,2995	131,3447	-9,9935	-2,8679	74,5175	21,2274	22,2481	243,2224	-8,9865	-7,6676
10	9	8	1	7,5541	141,1807	12,1723	10,5463	91,6677	49,4715	141,9766	247,916	4,046	15,3105
11	10	17	0	-0,957	115,0977	2,1459	4,641	79,4041	17,468	91,9853	223,6405	-6,6991	9,4347
12	11	5	1	-5,494	108,8735	-4,5378	-6,3579	94,8158	-3,2812	-4,1555	214,9727	-19,0954	-0,7754
13	12	5	0	-3,689	127,4767	11,3952	5,3661	85,6431	25,9487	174,1962	247,1185	-5,6727	10,0724
14	13	12	1	-2,7149	88,8419	-2,6962	-12,9699	90,9468	9,9317	102,4448	228,3043	-1,4365	8,0012
15	14	13	0	5,8879	130,9131	3,7283	2,4762	95,2448	17,9569	46,5638	205,6013	-6,9794	-1,3287

Exemple d'application

Conclusion

- Ensemble de scripts Matlab / BIND
 - Analyse des données véhicules collectées lors de l'expérimentation de la thèse de Maud Ranchet
 - Développé au LESCOT par Qing Huang
- Réutilisable en partie pour d'autres projets

BIND est accessible à des stagiaires



Le projet BIND

Le projet BIND

- Historique
 - Hérite de la culture du LESCOT et du groupe « dépouillement »
 - Conçu pour traiter des données de conduite
- Développé au LESCOT
 - Damien Sornette
 - Arnaud Bonnard

Le projet BIND

- Emploi des outils de la qualité logicielle
 - Gestion des versions logicielles
 - Pilotage des développements
 - Génération automatique de documentation
 - Non régression du code

Le projet BIND

Gestion des versions logicielles : Subversion

- Subversion (SVN)
 - Historique des révisions, suivi des modifications, édition concurrente
 - Dépôt central vs. copies de travail
- Dépôt INRETS
 - Géré par le CIR
 - Droits d'accès en lecture/écriture

Le projet BIND

Pilotage des développements : Redmine

- Redmine

- Forge logicielle
- Gestion des demandes (bugs, évolutions...)

- Forge INRETS

- <https://redmine.inrets.fr>
- Géré par le CIR
- Droits d'accès manager/développeur/rédacteur

Le projet BIND

Génération automatique de documentation : NaturalDocs4Matlab

- NaturalDocs4Matlab
 - Fork du projet libre NaturalDocs
 - Support de la syntaxe orientée objet de Matlab
- ND4M
 - Développé au LESCOT par Bertrand Richard
 - Sous licence GPL

Le projet BIND

Non régression du code : Xunit

- XUnit
 - Framework de test unitaires
 - Test du comportement attendu de toutes les méthodes de toutes les classes
- XUnit pour Matlab
 - Outil pour développeurs
 - Evolution sans regression



Quel avenir pour BIND ?

Perspective

- Développement
- Utilisation
- Diffusion
- Valorisation

Perspectives de développement

- Tests unitaires
 - Finir tous les tests unitaires
- Documentation
 - Exemples et « getting started »
 - Mise au point de tutoriels vidéos
- TripSet
 - Ajouter des fonctionnalités pour les traitements de groupe

En fonction des besoins

Perspectives d'utilisation

- Scripts
 - Analyse de données de conduite
 - Programme de validation des données véhicules
- Applications graphiques
 - Module de codage d'informations depuis la vidéo
- Applications satellites
 - Visualisation de données sur cartographie
 - Modèle de conducteur

Selon votre imagination

Perspectives de diffusion

- Diffusion par étapes
 - Accompagnement de quelques intéressés
 - Consolidation
 - Diffusion plus large...
- Choix d'une licence libre GPL ou LGPL
 - Crédit à une communauté
 - <http://bind-project.inrets.fr>

Perspectives de valorisation

- Valorisation scientifique
 - Articles dans congrès ou revue
 - Thèmes : reproducible research, data sharing. . .
- Valorisation des « satellites »
 - Diffusion de Matjab, de NaturalDocs4Matlab
 - Diffusion des modules graphiques existant
 - Développement de nouveaux modules

Merci de votre attention !

Contacts

- Arnaud Bonnard : arnaud.bonnard@inrets.fr
- Damien Sornette : damien.sornette@inrets.fr

Crédits iconographiques

- Slide 12 : <http://www.flickr.com/photos/yomostro/2972523219/>
- Slide 14 : <http://www.flickr.com/photos/mesh/7876556/>
- Slide 17 : <http://www.flickr.com/photos/crystaljingsr/3915514724/>
- Slide 36 : <http://www.flickr.com/photos/crystaljingsr/3915513242/>
- Slide 51 : <http://www.flickr.com/photos/crystaljingsr/3915515338/>
- Slide 34 (à gauche) : <http://www.flickr.com/photos/crystaljingsr/3915514014/>
- Slide 49 : <http://www.flickr.com/photos/crystaljingsr/3915514014/>
- Slide 75 : <http://www.flickr.com/photos/john/4776861/>
- Slide 58 : <http://www.flickr.com/photos/crystaljingsr/3914731337/>
- Slide 68 : <http://www.flickr.com/photos/crystaljingsr/3915515536/>
- Slide 42 : <http://www.flickr.com/photos/gad1/237122671/>
- Les autres éléments iconographiques ont soit été créés pour la présentation, soit appartiennent au domaine public.