

Grand mère mode

@framiere
Jaxio
Programmeur/Développeur/Codeur

#grandmere

@framiere



Plan

- Contexte projet
- Excel
- POI
- Antlr
- Google Spreadsheet
- Négatif ?
- Conclusion

#grandmere

@framiere

Quick show of hand.

Qui a déjà utilisé Haskell ? Ocaml ? Clojure ? Scala ?

Quid de java8 ? ... anciennes version de java ?

Qui a déjà utilisé excel. Ok :) Vous seriez peut-être une mamie qui s'ignore...

Vous avez remarqué qu'on va parler d'excel, et qu'on est dans le track langage alternatif ?
Je trouve que c'est déjà très intéressant...

Les langages alternatifs ne sont pas ceux qu'on croit.

Qui a déjà utilisé POI ?

Qui a déjà créé un langage ?

Qui a déjà réalisé une grammaire ?

Qui a déjà utilisé google spreadsheet ?

A User-Centred Approach to Functions in Excel

30th June 2003

Simon Peyton Jones
Microsoft Research, Cambridge
simon@research.microsoft.com

Alan Blackwell
Cambridge University
alan.blackwell@cl.cam.ac.uk

Margaret Burnett
Oregon State University
burnett@cs.oregonstate.edu

Abstract

We describe extensions to the Excel spreadsheet that integrate user-defined functions into the spreadsheet grid, rather than treating them as a “bolt-on”. Our first objective was to bring the benefits of additional programming language features to a system that is often not recognised as a programming language. Second, in a project involving the evolution of a well-established language, compatibility with previous versions is a major issue, and maintaining this compatibility was our second objective. Third and most important, the commercial success of spreadsheets is largely due to the fact that many people find them more usable than programming languages for programming-like tasks. Thus, our third objective (with resulting constraints) was to maintain this usability advantage.

Simply making Excel more like a conventional programming language would not meet these objectives and constraints. We have therefore taken an approach to our design work that emphasises the cognitive requirements of the user as a primary design criterion. The analytic approach that we demonstrate in this project is based on recent developments in the study of programming usability, including the Cognitive Dimensions of Notations and the Attention Investment model of abstraction use. We believe that this approach is also applicable to the design and extension of other programming languages and environments.

Categories and Subject Descriptors

D.3.2 [Programming Languages]: Language classifications—Functional languages; D.3.3 [Programming Languages]: Language constructs and features—Procedures, functions and subroutines

1 Introduction

For many people, the programming language of choice is a spreadsheet. This is especially true of people who are not employed as programmers, but write programs for their own use — often defined as “end-user” programmers [Nar93]. An end-user programmer is a *scientist, an engineer, a projective, a manager, an accountant, a manager, in fact almost anything except a trained programmer*. These people use computers to get their job done, but often they are not interested in programming *per se*. End-user programmers outnumber professional programmers, and their numbers are projected to increase more rapidly; in fact, the number of end-user programmers in the U.S. alone is expected to reach 38 million by 2005, as compared to only 2.75 million professional programmers [BAB’00]. These facts suggest that the spreadsheet, which is a widely used and commercially successful end-user programming language, is also a particularly significant target for the broader application of programming-language design principles.

It may seem odd to describe a spreadsheet as a programming language. Indeed, one of the great merits of spreadsheets is that users need not think of themselves as doing “programming”, let alone functional programming — rather, they simply “write formulae” or “build a model”. However, one can imagine printing the cells of a spreadsheet in textual form, like this:

$$\begin{aligned} A1 &= 3 \\ A2 &= A1+3 \\ A3 &= A2 * 6/9 \end{aligned}$$

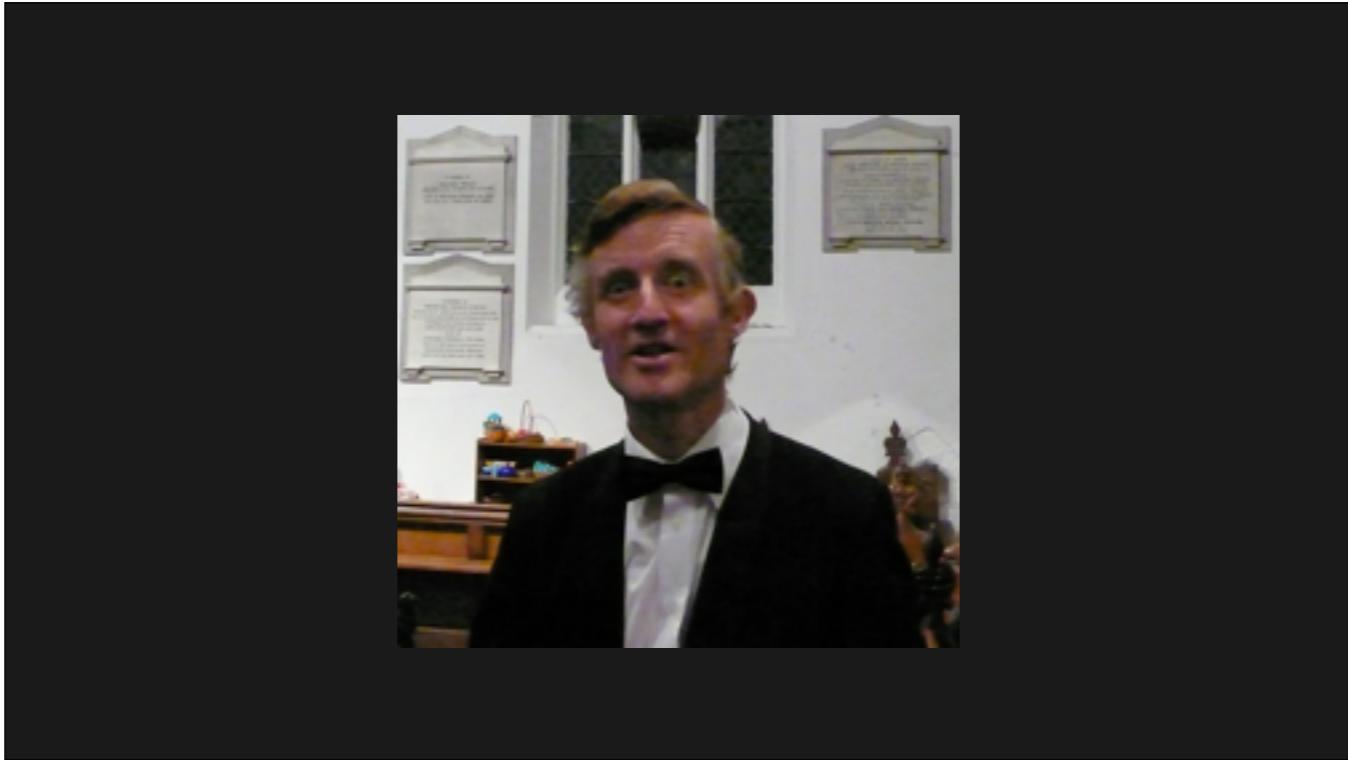
and then it plainly is a (functional) program. Thought of as a programming language, though, a spreadsheet is a very strange one.

For many people, the programming language of choice is a spreadsheet. This is especially true of people who are not employed as programmers, but write programs for their own use — often defined as “end-user” programmers [Nar93].

Je vous la traduit en mode barbare "les end-users programment en Excel"

Bref, les ends-users ils ont un marteau magique, et c'est excel.

Qui sait qui est Simon Peyton Jones ?



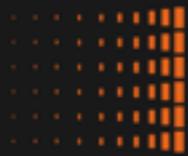
Je vous aide ... C'est lui là ...



Yup c'est un des gars derrière Haskell. Le Glasgow haskell compiler (ghc) c'est lui.
Si des mecs comme lui s'interessent à Excel, on peut aussi :)
Et en plus on reste dans le ton des langages alternatifs !



Objectifs



Bon c'est pas tout, mais quels sont mes objectifs ?

A retenir

- Ne refaites pas ce que peut faire Excel
 - Faites bosser vos utilisateurs !
 - Parlez le langage de vos utilisateurs
 - Utilisez des outils collaboratifs
-
- Créez ce langage si il n'existe pas déjà
 - ... si possible proche d'Excel qu'ils savent manipuler



#grandmere

@framiere

Ne pas refaire ce que fais Excel très bien ? Captain obvious ! Mais pourquoi est-ce que je le rappelle ? Car **c'est oublié trop souvent** ! J'ai bien sûr moi même réalisé des devs qui auraient du rester en Excel. Si on a une spec exprimée sous forme excel, et que vous n'avez pas de besoin avéré de perf, vous serez agréablement surpris !

On connaît pas mal de contraintes dans excel, donc utilisez google spreadsheet et autres feuilles interactives collaboratives si nécessaire. Ca résout plein de problèmes. Vous vous êtes tous pris des lock sur des feuilles magiques partagées.

Si vous donnez les moyens à vos utilisateurs d'exprimer les règles, vous n'avez pas besoin d'aller les chercher. Ils sont **de facto** acteur car ils font leur règles eux même.

Mais pour ça il faut parler le langage des vos utilisateurs !

Si vous arrivez à un exprimer ce métier avec excel simple tant mieux ! Sinon créez ce langage !

Les specs sont **executables**, pas d'ambiguïté de spec ou sinon on a un prob fondamental qu'on doit adresser.

Contexte

- Migration moteur de calcul
- Création d'indicateurs financiers
- Plus de 7 000 règles
- 21 étapes
- Plusieurs équipes
- Pas de compétence technique
- **Un seul et unique modèle !**
- des milliards de résultats à calculer
- le métier perdu dans la technique

#grandmere

@framiere

Ce talk est issu d'une mission sur laquelle je suis en ce moment où on fait une migration d'un moteur de calcul qui était sur 11 bases de données distinctes développé en sql, c#, vb, excel, access, procédures stockées, trigger, scripts.

11h pour produire des résultats pour 900 boîtes, un concurrent arrive qui vend un service avec plus de boîtes trackées. Faut faire kkchose !

Ok côté business, mais techniquement personne ne savait comment ça marchait **en détail**, il y avait un Bus factor en plein milieu du bouzin... qui est parti.

Bref, ce projet c'est marche ou crève.

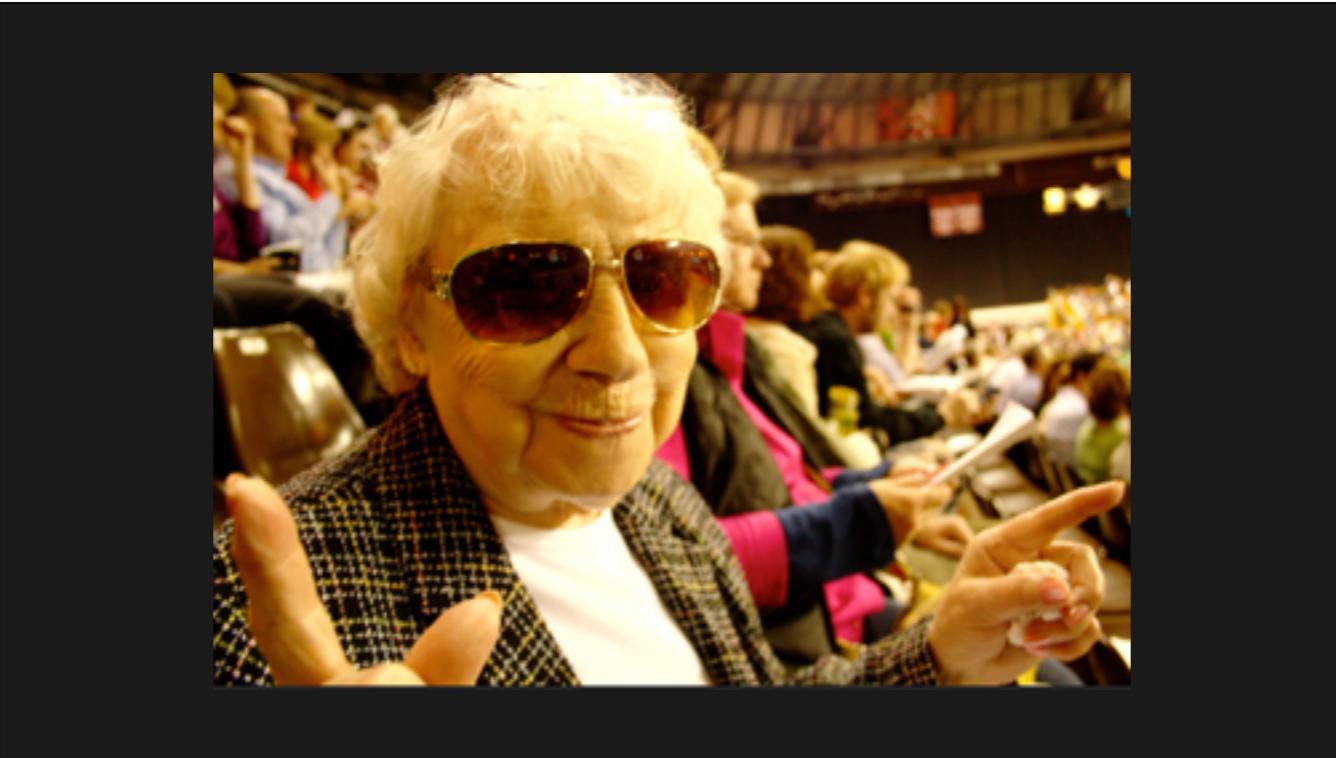
Aïe !

#grandmere

DEVO^X France

@framiere

Vraiment, le métier est compliqué !

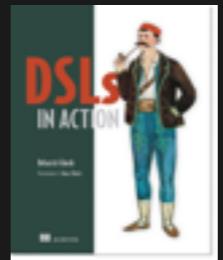


Mais les grand mères, elles en ont vues d'autres elle savent gérer.

Ce que savent faire les grand mères, c'est ... communiquer.

Donnons nous les moyens de communiquer sur ce projet alors !

DSL !



#grandmere

DEVOXXTM France

@framiere

Une seule solution, les obliger à exprimer leur métier sous forme précise et COMPLETE. Qu'ils soient au cœur du projet, et les faire bosser en "vrai". Bref, on va leur mettre dans les pattes un DSL !

Domain specific language. Le bouquin de Debasish Ghosh est très bien.

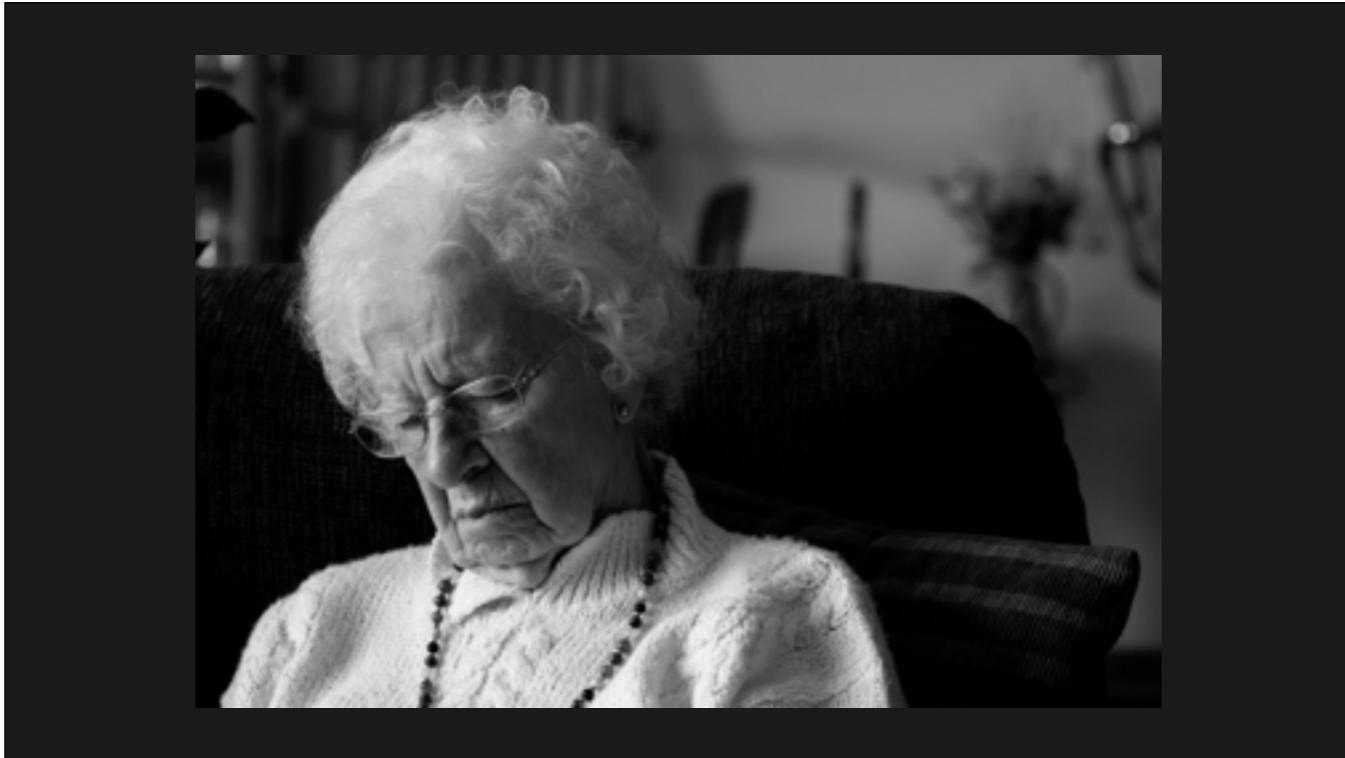
Pour ceux qui auraient plus vu un moteur de règles sur les 7000 règles, ce n'est pas possible sous peine de péter le bras du moteur. Il y a des maths, du control flow, de la perf, des débranchements, du temps.Bref c'est juste pas adapté.



Comme on est français, on connaît tous Guillaume des castcodeurs... et de groovy.
Et comme il est sympa: facile on sort Groovy !

On va faire un joli dsl qui se lit comme une phrase !

Yeah !



Po po po, pas trop vite, on veut un truc simple, un truc qu'on connaît.
Un langage du type "si le cfroc average du sous-secteur de l'univers World developed est au dessus de la moyenne mobile de 5 and du consensus forecast le cfroc_consensus ... " bref ça marche pas ça

Leur vrai langage

#grandmere

DEVO^X France

@framiere

Y a pas photo ... C'est excel. Et c'est encore plus évident dans la finance !



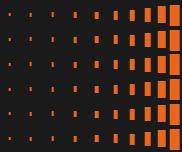
Bref ca doit être un Excel like. Autant ils ne sont pas techniques, autant on sait se parler via une feuille excel.

Et si ils n'arrivent pas a s'exprimer sous une forme excel-like –même draft–, posez-vous des questions.
Comment ne peuvent-ils pas exprimer leur métier ? Sont-ils les bons interlocuteurs ?

Sinon juste pour la blague, certaines de leur feuilles excel sont absolument giganormous, et laissent songeur quand à leur santé mentale. Mais bon CA MARCHE ! Et si ca marche, moi ca me va.

J'ai même du faire des outils pour suivre leur feuilles excel tellement elles était compliquées.

Demo POI



#grandmere

DEVO^X France

@framiere

Qui a déjà lu des feuilles excel programmatiquement ?

Qui a déjà utilisé POI ?

Qui sait que POI sait executer des feuilles excel ?





```
@Test
public void should_load_execute() {
    FormulaEvaluator evaluator = creationHelper.createFormulaEvaluator();

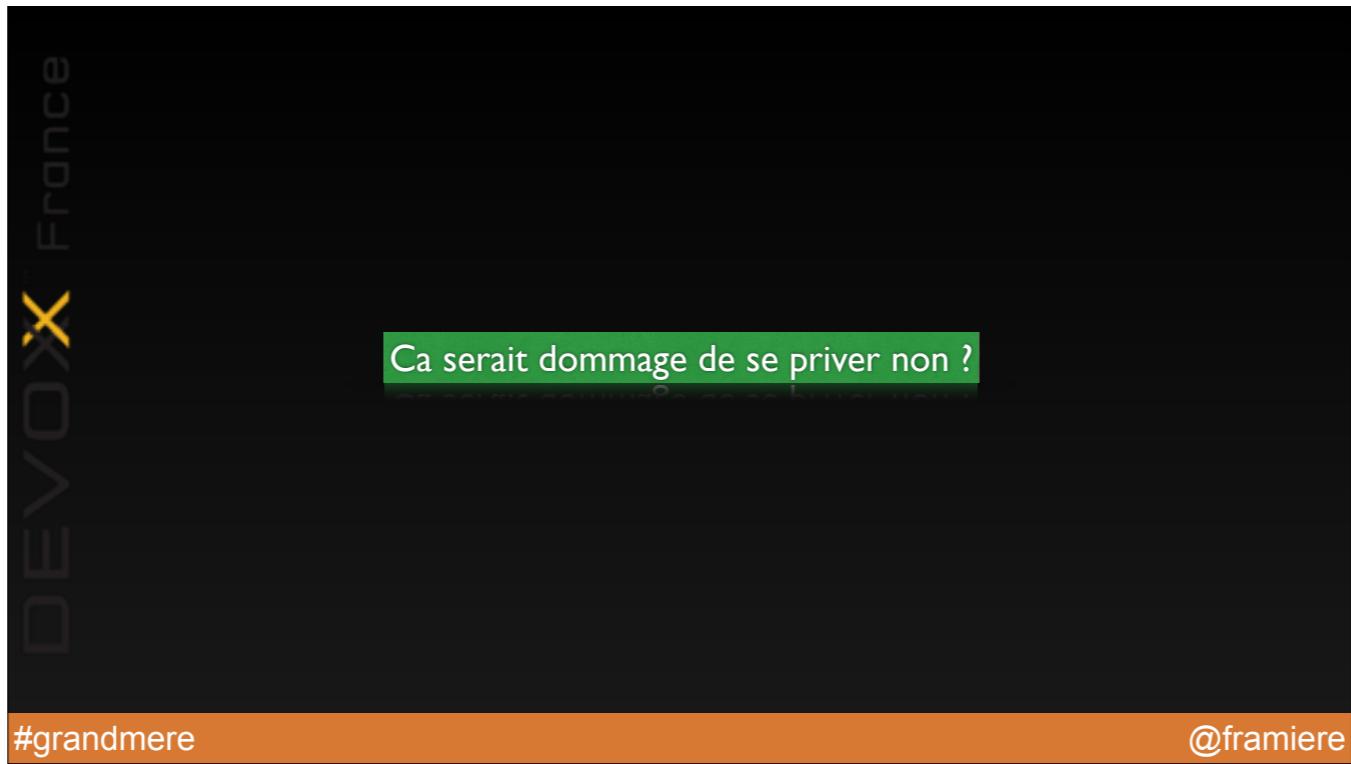
    Sheet perf = workbook.getSheetAt(0);
    for (int rowId = perf.getFirstRowNum(); rowId < perf.getLastRowNum(); rowId++) {
        Row row = perf.getRow(rowId);
        for (int column = 0; column < row.getLastCellNum(); column++) {
            assertThat(cell(row.getCell(column), evaluator)).isNotNull();
        }
    }

    private String cell(Cell cell, FormulaEvaluator evaluator) {
        switch (cell.getCellType()) {
            case Cell.CELL_TYPE_FORMULA:
                return evaluator.evaluate(cell).formatAsString();
            case Cell.CELL_TYPE_NUMERIC:
                return "" + cell.getNumericCellValue();
            case Cell.CELL_TYPE_STRING:
                return cell.getStringCellValue();
            case Cell.CELL_TYPE_BOOLEAN:
                return "" + cell.getBooleanCellValue();
            case Cell.CELL_TYPE_BLANK:
                return "";
            case Cell.CELL_TYPE_ERROR:
                return null;
            default:
                throw new IllegalStateException("Could not handle " + cell.getCellType());
        }
    }
}
```

30 lignes !

#grandmere

@framiere



D'autant que ça permet de ne pas bosser sur des interfaces de saisie. De validation etc.
Finalement ce qu'ils font eux, ils le valident eux-même. On s'occupe uniquement des données d'entrée et de celles à récupérer.

Ok mais mes Excel ont plein de VB dedans
...et ça piquouille quand même pas mal !

#grandmere

@framiere

Ben ouaip, je connais bien le problème, j'ai eu du C#, VB, SQL à migrer.

Fonctions VB

- VB ou pas VB il y a du métier à prendre en compte !
- Redéveloppez ce code en Java à la main
- Traduisez ce code legacy automatiquement en java
- Puis déclarez ces fonctions à POI

```
private static DefaultBiffFinder registerFunctions() {  
    String[] functionNames = {"WEEKE", "PIRA", "ABYE", "WEE", "PVTE4", "PVTE5", "OPEN",  
    "PrestFunction"};  
    FunctionData[] functionData = new BiffFunction[5], new PVEMacro(), new ABYEMacro(), new WE  
    Macro(), new PVTEMacro(), new PVTEMacro(), new PVTEMacro(), new PVTEMacro(), new CustomRateFunc  
    tion();  
    return new DefaultBiffFinder(functionNames, functionData);  
}  
  
private static void registerFunctions() {  
    FunctionData.registerFunction("WE", new ESRMFunction());  
    FunctionData.registerFunction("WE", new RSDFunction());
```

Setup

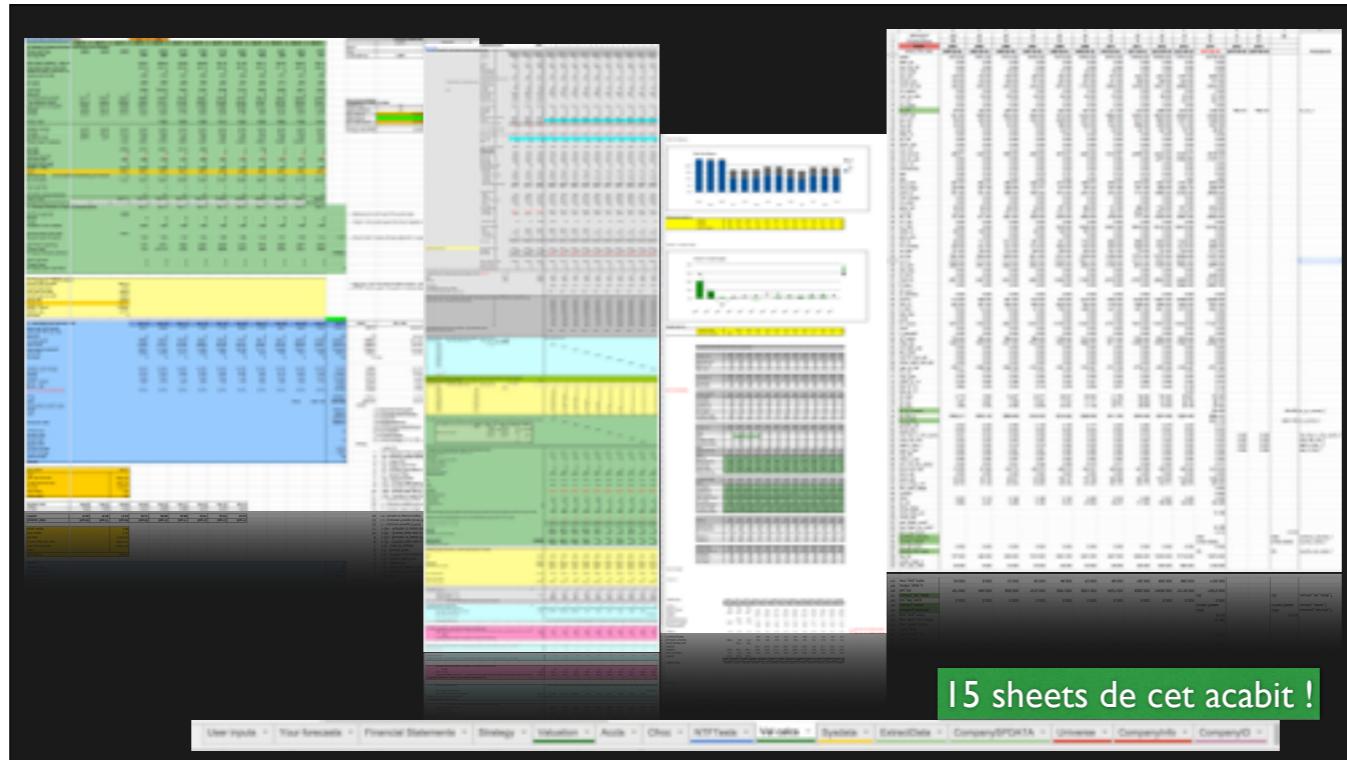
#grandmere

Contrat simplissime

```
(S110) public class ABYEMacro implements ISheetFunction {  
    @Override  
    public ValueEval evaluate(ValueEval[] args, OperationEvaluationContext ec) {  
        if (args.length != 1) {  
            return ErrorValue.ERROR_INVALID; // S110  
        }  
        try {  
            ValueEval singleValue = getSingleValue(args[0],  
                ec.getRowIndex(),  
                ec.getColumnIndex());  
            Double value = OperandResolver.coerceToDouble(singleValue);  
            return new NumberEval(ec.getRowIndex(), ec.getColumnIndex(), value);  
        } catch (EvaluationException e) {  
            log.warn("Error at " + ec.getRowIndex() + ":" + ec.getColumnIndex() + ":" + e);  
            return e.getErrorValue();  
        }  
    }  
}
```

@framiere

VB ou pas VB il y a du savoir faire, faut donc bien le gérer.
Poi permet d'ajouter des fonctions donc go.
En plus les contrats sont triviaux.



Voici une application qui MARCHE. C'est hyper complexe.

Ca vole a 100km au dessus de ma tête.

Toutefois, les BA maîtrisent ce sujet. Ca fait le taff.

Moi je me contente de remplir des cellules avec des données et roule ma poule.

J'en extrait certains résultats pour continuer mes traitements.

On a transformé un sujet gigantesque principalement métier a un sujet technique assez facile. On se contente de mapper des données in, et extraire des données.

En vrai, faut valider ce bazar: ma technique demander une version "normale", et une version "data" où il n'y a plus du tout de formula. Et moi je fais une validation cellule a cellule pour vérifier que je calcule bien la même chose. Les BA ont la facheuse tendance à oublier que c'est un outil qui chope le bazar, dont on fait pleins de checks histoire de pas perdre de temps. Genre les headers etc.

Quizz



#grandmere

DEVO^X France

@framiere

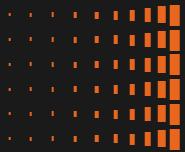


Qui sait qui est cette personne ?
Eric Schmidt ex ceo de Google.
Pourquoi je vous pose la question ?

Car il a écrit Lex le grand père de flex, et donc l'arrière grand père d'antlr ... en 75. Soit 40 ans !

On est bien dans de la techno de grand père.

Demo antlr



#grandmere

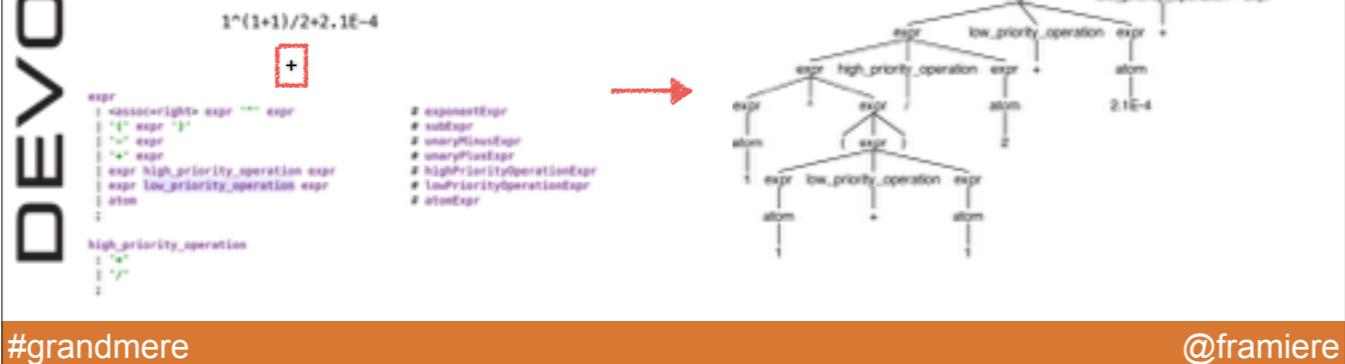
DEVO^X France

@framiere

Quand excel ne suffit plus il faut trouver une solution

Antlr

- Grossso modo une regex qui crée un arbre avec des noeuds nommés que l'on peut visiter.
- ... et c'est tout.



Vous connaissez les regexp avec les group. C'est l'équivalent en plus puissant.
On crée des match, on leur donne des noms, et on peut se faire appeler lorsque le visiteur tombe sur le nom.

Dans notre projet on produit du java pur pour faire de la compilation avec Javac et avoir les meilleures performances.

On a également le typage de java gratuitement. Javac est appellable par le jdk.

Pour commencer je vous recommande de produire du Groovy pour ne pas avoir à s'embêter avec les technicalités.

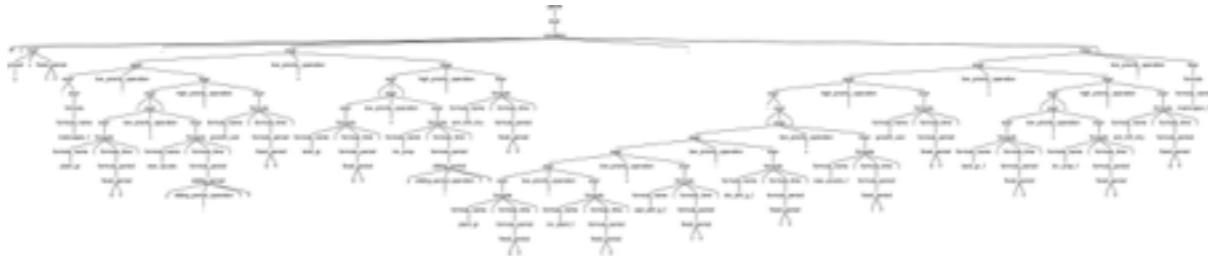
Ce sujet du typage est très interressant. C'est une découverte, qui après coup n'est pas une. Mais en tant qu'informaticien.

Et plus spécifiquement dev java, pour moi le typage est kkchose de très important.

Hors excel fait de la coercion de type, c'est très utile, mais ca fait rapidement du bordel. On a pas les memes attentes, les BA sont très rapidement réfractaires, et trouvent ca limité. Il faut se battre pour conserver cet outil indispensable

1 des 7000 règles

```
IF( y=1Y, - maincapex_f - ( plant_gr(0Y) + new_assets(y-1Y) ) *  
growth_use(0Y) + (land_gr(0Y) + inv_prop(y-1Y) ) *  
ann_infl_ctry(0Y), - ( plant_gr(0Y) - inc_plant_f(1Y) +  
aqn_plnt_g_f(1Y) - dis_plnt_g_f(1Y) + new_assets_f(1Y) ) *  
growth_use(0Y) + (land_gr_f(1Y) + inv_prop_f(1Y) ) *  
ann_infl_ctry(0Y) - maincapex_f)
```



#grandmere

@framiere

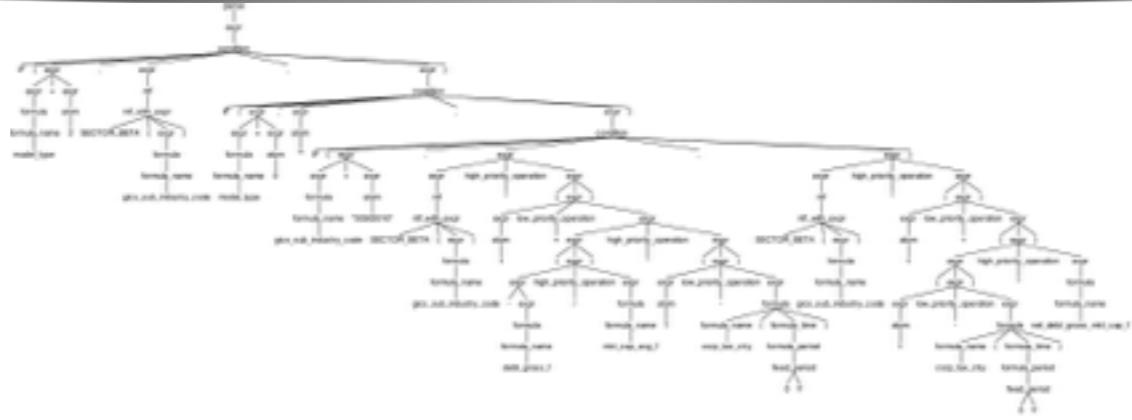
Prenons quelques instants pour piger ce que c'est .

100 métier

Vous leur donnez des outils, ils les utilisent !

1 des 7000 règles

```
IF( model_type = 2 , SECTOR_BETA( gics_sub_industry_code ), IF( model_type = 0 , "", IF(gics_sub_industry_code = "20302010" , SECTOR_BETA( gics_sub_industry_code ) * ( 1 + ( - debt_gross_f / mkt_cap_avg_f) * (1- corp_tax_ctry(0Y) ) ) ,SECTOR_BETA( gics_sub_industry_code ) * ( 1+( (1- corp_tax_ctry(0Y) ) * net_debt_gross_mkt_cap_f ) ) ) ) )
```



#grandmere

@framiere

Combien d'allez retour aurions-nous eu avec des specs classiques ?

1 des 7000 règles

```
IF( y=2Y, IF( AND( -bs_gdbt_f(1Y) >= bs_cash_f(1Y), -bs_gdbt_f(1Y) >
chg_ntdbt_f ), bs_cash_f(1Y), IF( AND( -bs_gdbt_f(1Y) >= bs_cash_f(1Y), -
bs_gdbt_f(1Y) <= chg_ntdbt_f ), bs_cash_f(1Y) + chg_ntdbt_f + bs_gdbt_f(1Y),
IF( AND(-bs_gdbt_f(1Y) < bs_cash_f(1Y), -bs_cash_f(1Y) <= chg_ntdbt_f ),
bs_cash_f(1Y) + chg_ntdbt_f, IF( AND(-bs_gdbt_f(1Y) < bs_cash_f(1Y), -
bs_cash_f(1Y) <= chg_ntdbt_f ), 0, "" ) ) ) + bs_cash_adj_f, 0)
```

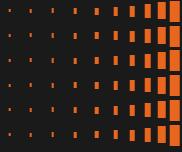


#grandmere

@framiere

Et là ?

Google spreadsheet



#grandmere

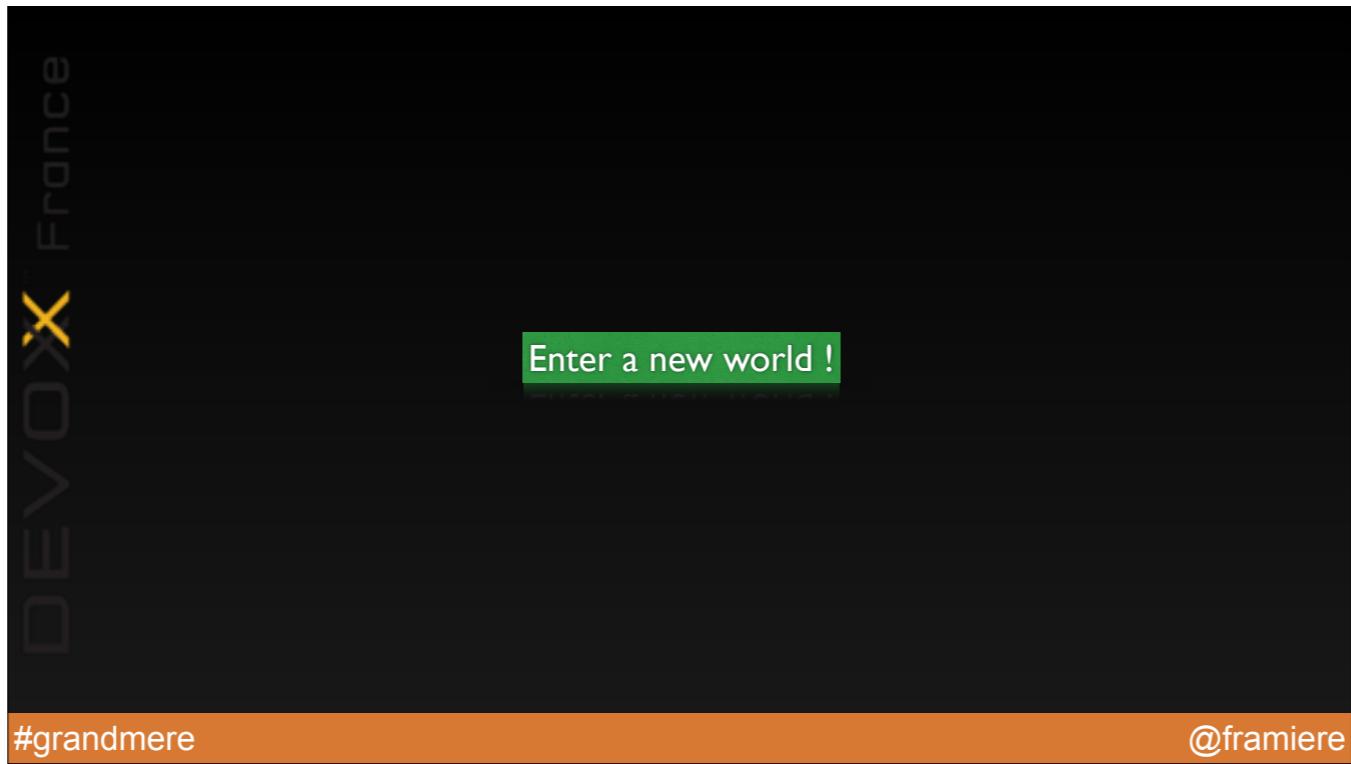
DEVO^X France

@framiere

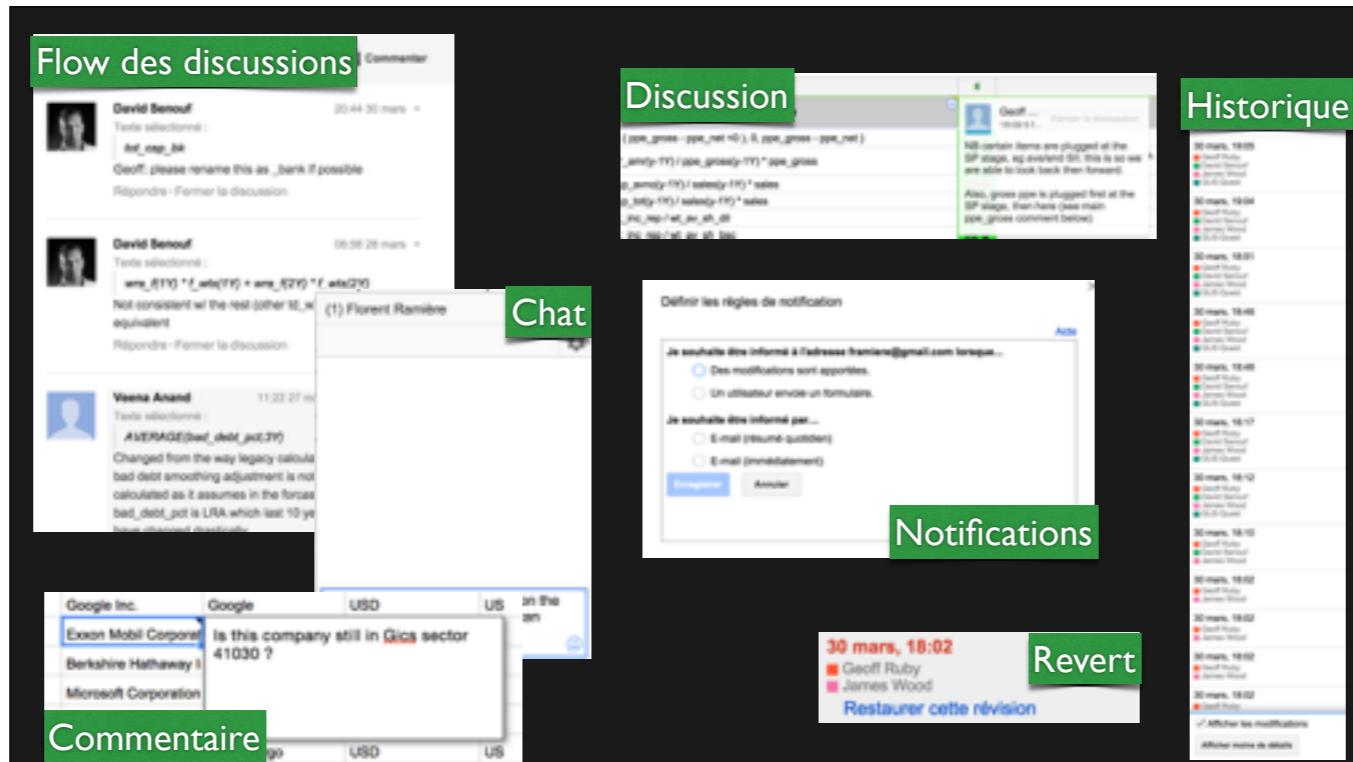
Où vont-ils exprimer leur domaine ?
... là où ils savent déjà l'écrire partiellement.

Dans excel.

Or ils sont une équipe, donc pas possible, on va juste utiliser de vrais outils modernes.



Attention ... Là c'est le choc pour les métiers !

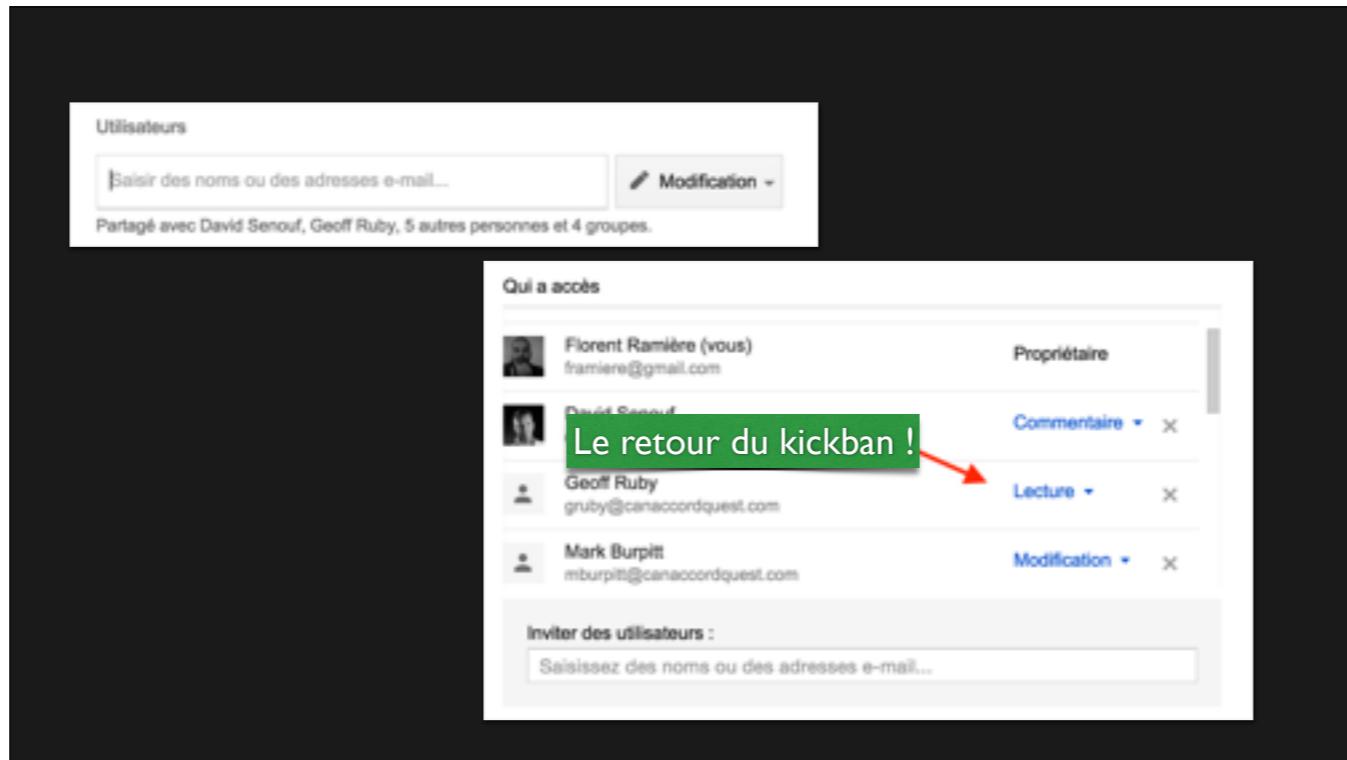


Rendez-vous compte combien ce genre de choses va pouvoir aider l'équipe de BA à enfin communiquer ? Cela va bien au delà de la technique, ils ont désormais tous les moyens de vraiment bosser ensemble alors qu'avant elles étaient avant en silo ! Le pré carré est désormais percé.

Si on avait bossé en mode "normal". Chacun aurait fait sa spec, on aurait eu la misère, tout le monde aurait alors joué au blame game. Et si on s'en serait sorti, ce serait avec un code imbuvable.

Là tout le monde est sur le pont.

C'est un jira de business... executable !



Ma feature préférée : Le retour du kickban !

Tout ça gratuit !

Réimplémenter toutes ces features pour votre appli de gestion, c'est juste impensable, on n'a pas le temps, et surtout on est pas payé pour !

Google script

#grandmere

DEVO^X France

@framiere

Ca piquouille le retour

- Vous vous rappelez du slide précédent ?



- Vous pouvez réaliser vous même votre legacy en javascript !
- ... ou alors mettez votre complexité métier derrières des api rest
- ... que vous partagez avec les feuilles excel executables !

#grandmere

@framiere

Ca fait 99% du job.

Les points négatifs

#grandmere

DEVO^X France

@framiere

Rien n'est gratuit dans ce bas monde

Dur dur

- Ca va tanguer !
- Découvrir le langage adéquat est difficile
- Le temps de support est très important
- Les excel utilisateur c'est vraiment le bazar
- Les visiteurs antlr ca se fait en TDD (pour une fois)
- Le reporting d'erreurs google scripts est moi si
- Google excel download downtime (pas cool, mais rare)

#grandmere

@framiere

Ca va tanguer... car c'est une nouvelle façon de travailler les clients ont "plus de travail"

Trouver un langage adéquat c'est difficile : ... mais peut-on s'en passer ? 400 révisions sur les grammaires, 3 génération de versions pour le langage principal.

Support : ... ils n'ont pas l'esprit de développeur et loupent des choses simples
... mais cela permet de rester près d'eux et d'échanger
... ca permet de découvrir avec eux des difficultés de leur modèle et de trouver des solutions simples
contrairement a ce qu'ils font tout seuls avec Excel.

Excel bazar: On doit enforcer un contrat clair entre le monde du dev, et le monde d'excel

Les visiteurs doivent se faire en TDD

Le reporting d'erreurs google scripts: il est naze.

Résumé

#grandmere

DEVO^X France

@framiere

Créez vos langages

C'est tout bête

#grandmere

DEVO^X France

@framiere

Utilisez Excel

Vite fait bien fait !

#grandmere

DEVO^X France

@framiere

Utilisez Google sheet

On peut bosser en équipe !

#grandmere

DEVOXTM France

@framiere

Accompagnez le business

Plutôt que le coder...

#grandmere

DEVOXTM France

@framiere

Q/A

#grandmere

DEVO^X France

@framiere

Transformé en javascript

TERM	DEFINITION
Net income	Net income, Net profit, Net earnings. Profit attributable to the equity holders of the parent incurring亏损. Should disclose other changes in equity items.
Net income	NET to net income on a consolidated basis
Net loss	Loss for the year ended April 30 (NLT)
Net loss	Loss of parent and Gross profit
Net loss	Other operating income
Net loss	Operating expenses
Net loss	Selling, general & administrative
Net loss	Depreciation & depletion (D)
Net loss	Asset write-downs
Net loss	Other assets

Produit des pages web

Macmillan Resources Inc.

Period	2014	2013	2012	2011	2010	2009	2008	2007	2006	2005	2004	2003	2002	2001	2000
Assets	200,973	176,433	148,000	120,912	108,000	90,000	120,244	95,000	77,000	55,000	31,000	21,000	12,000	1,200	1,200
Operating profit (loss) accumulated	86,101	51,110	1,000	11,000	20,000	-20,000	25,100	40,000	-10,000	33,000	21,000	21,000	21,000	21,000	21,000
Provision	50,170	10,000	1,000	70,000	20,000	-40,000	25,000	25,000	-10,000	10,000	10,000	10,000	10,000	10,000	10,000
Provision after tax (PAT)	35,030	5,140	0,000	10,011	0,000	-10,000	20,000	20,000	-10,000	10,000	10,000	10,000	10,000	10,000	10,000
Minority interest in controlling/held joint venture	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
Dividend paid & Extraordinary items	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
Less: Provisions and Reserves	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
Income	20,001	5,130	0,000	10,011	0,000	-10,000	20,000	20,000	-10,000	10,000	10,000	10,000	10,000	10,000	10,000
Interest	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
DEPIK:															
Turnover (PAT/200 million Andy & VAT)	200,973	176,433	148,000	120,912	108,000	90,000	120,244	95,000	77,000	55,000	31,000	21,000	12,000	1,200	1,200
Gross profit	86,101	51,110	1,000	10,000	20,000	-20,000	25,100	40,000	-10,000	33,000	21,000	21,000	21,000	21,000	21,000
Other operating income	0,000	1,000	0,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000

2 langages

un pour créer le web

un pour la validation interne

Pas besoin que ca soit compliqué pour que ca soit efficace

4 mots. Title, Value, Newline, Compute

L'autre produit le code javascript utilisé pour de l'aide a la validation



100% est piloté par des langages

	'1wk'(8D)	'{1,3,6}mth'(80)	'{1,3,5,10,15}yr'(80)
label:pr_abs	pr_abs_1W_curr	pr_abs_{1,3,6}M_curr	pr_abs_{12M,3Y,5Y,10Y,15Y}_curr
label:pr_rel	pr_rel_1W_curr	pr_rel_{1,3,6}M_curr	pr_rel_{12M,3Y,5Y,10Y,15Y}_curr
label:pr_abs_agg	level0:pr_abs_1W_curr_agg	level0:pr_abs_{1,3,6}M_curr_agg	level0:pr_abs_{12M,3Y,5Y,10Y,15Y}_curr_agg
label:pr_abs_agg	level3:pr_abs_1W_curr_agg	level3:pr_abs_{1,3,6}M_curr_agg	level3:pr_abs_{12M,3Y,5Y,10Y,15Y}_curr_agg

↓

Price Performance %

	1 week	1 month	3 months	6 months	1 year	3 year	5 year	10 year
Share price relative	-0.9	-0.2	42.5	33.7	97.5	197.8	233.5	-2.4
Share price absolute	-0.3	-90.2	46.9	40.1	103.2	281.5	366.1	54.3
Industry price absolute	1.4	-0.6	19.9	11.9	10.8	91.9	187.5	-
Market price absolute	0.8	1.8	7.0	5.0	4.6	32.6	40.0	59.1

2 langages

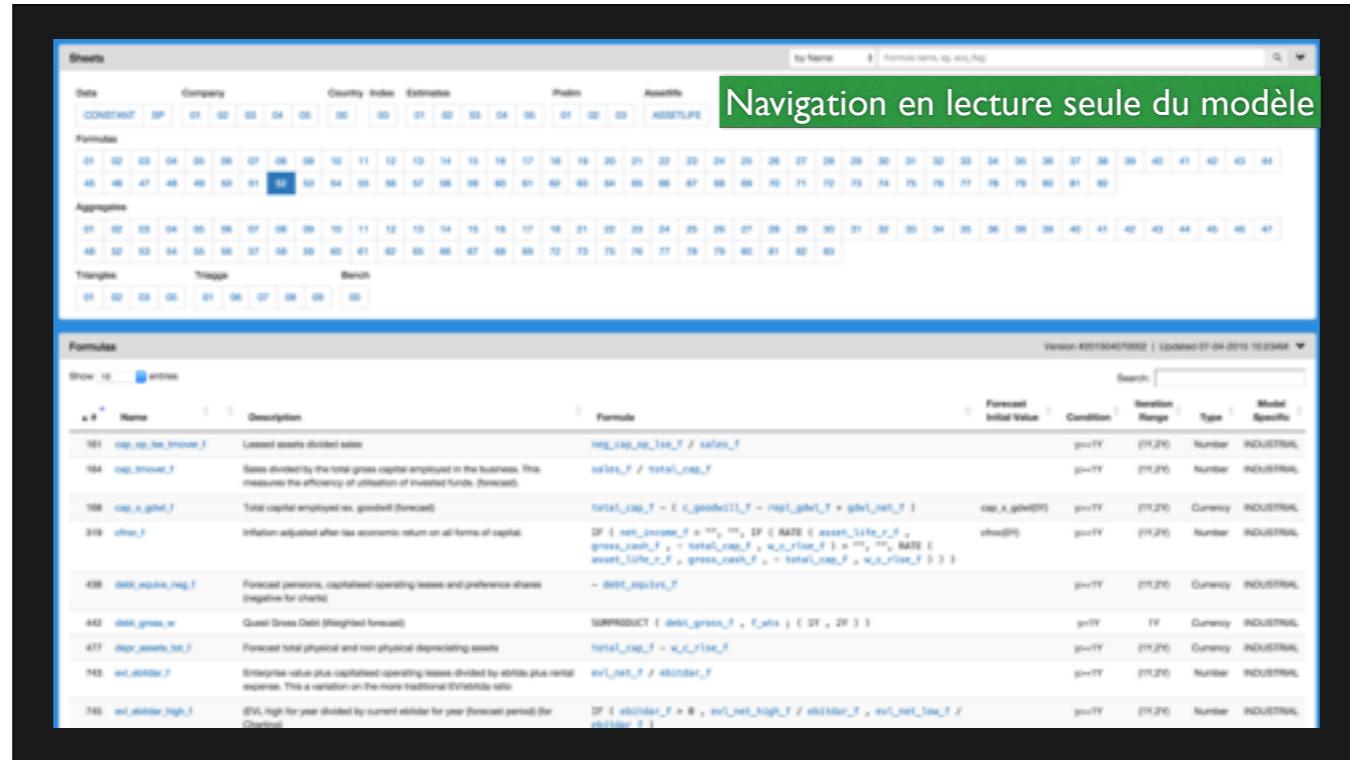
un pour créer le web

un pour la validation interne

Pas besoin que ca soit compliqué pour que ca soit efficace

4 mots. Title, Value, Newline, Compute

L'autre produit le code javascript utilisé pour de l'aide a la validation



Top left window (Formula Editor):

Type:	U.I.
Name:	FORMULA_E1_BY_AVG_1912
Description:	Used to calculate the average as intermediate step.
Formula:	$var_weight \times deviation / 100$

Top right window (Properties):

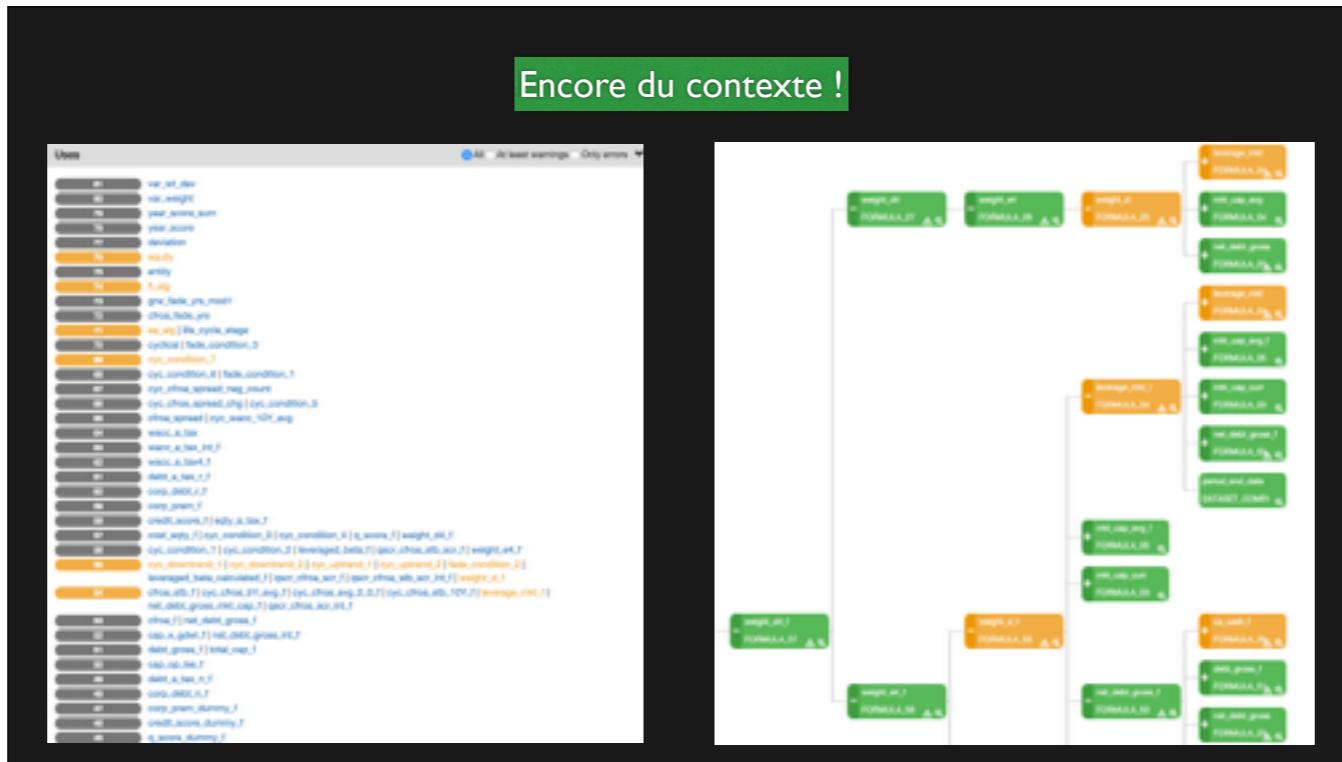
Additional values	
Condition	0.000000
Iteration range	1.00 (0)
Legacy Record Name	var_weight
Type	Number
Value type	%
Annotations	CHARTING (2,1,11%)
Data Item (1-Legacy)	4000
Dependent Item Range	1.00 (0)
Item ID	100
Item ID By	100, 200, 300, 400, 500, 1000
Last Modified By	John, John, John, John, John, John
Legacy Type	String (1..100)
Owner	UNSPECIFIED
Owner ID	00000000000000000000
Owner Name	UNSPECIFIED

Main window (Values):

Detailed description: Shows a table of values from 2014-12-01 to 2015-12-01. The table has columns: Date, Iteration, and Value. A green box highlights the header row. A green callout bubble "Du contexte!" points to the header row.

Date	Iteration	Value
2014-12-01	01	0.0514825416180408
2015-01-01	-1Y	0.0708868071771387
2015-02-01	-1Y	0.0660233675124504
2015-03-01	-1Y	0
2015-04-01	-1Y	0
2015-05-01	-1Y	0.070885
2015-06-01	-1Y	0.066402
2015-07-01	-1Y	0
2015-08-01	-1Y	0.066405
2015-09-01	-1Y	0.066405154480108
2015-10-01	-1Y	0.0660233675124509
2015-11-01	-1Y	0.066405154480108
2015-12-01	-1Y	0.066405154480108
2016-01-01	-1Y	0.066405154480108
2016-02-01	-1Y	0.066405154480108
2016-03-01	-1Y	0.066405154480108
2016-04-01	-1Y	0.066405154480108
2016-05-01	-1Y	0.066405154480108
2016-06-01	-1Y	0.066405154480108
2016-07-01	-1Y	0.066405154480108
2016-08-01	-1Y	0.066405154480108
2016-09-01	-1Y	0.066405154480108
2016-10-01	-1Y	0.066405154480108
2016-11-01	-1Y	0.066405154480108
2016-12-01	-1Y	0.066405154480108
2017-01-01	-1Y	0.066405154480108
2017-02-01	-1Y	0.066405154480108
2017-03-01	-1Y	0.066405154480108
2017-04-01	-1Y	0.066405154480108
2017-05-01	-1Y	0.066405154480108
2017-06-01	-1Y	0.066405154480108
2017-07-01	-1Y	0.066405154480108
2017-08-01	-1Y	0.066405154480108
2017-09-01	-1Y	0.066405154480108
2017-10-01	-1Y	0.066405154480108
2017-11-01	-1Y	0.066405154480108
2017-12-01	-1Y	0.066405154480108
2018-01-01	-1Y	0.066405154480108
2018-02-01	-1Y	0.066405154480108
2018-03-01	-1Y	0.066405154480108
2018-04-01	-1Y	0.066405154480108
2018-05-01	-1Y	0.066405154480108
2018-06-01	-1Y	0.066405154480108
2018-07-01	-1Y	0.066405154480108
2018-08-01	-1Y	0.066405154480108
2018-09-01	-1Y	0.066405154480108
2018-10-01	-1Y	0.066405154480108
2018-11-01	-1Y	0.066405154480108
2018-12-01	-1Y	0.066405154480108
2019-01-01	-1Y	0.066405154480108
2019-02-01	-1Y	0.066405154480108
2019-03-01	-1Y	0.066405154480108
2019-04-01	-1Y	0.066405154480108
2019-05-01	-1Y	0.066405154480108
2019-06-01	-1Y	0.066405154480108
2019-07-01	-1Y	0.066405154480108
2019-08-01	-1Y	0.066405154480108
2019-09-01	-1Y	0.066405154480108
2019-10-01	-1Y	0.066405154480108
2019-11-01	-1Y	0.066405154480108
2019-12-01	-1Y	0.066405154480108
2020-01-01	-1Y	0.066405154480108
2020-02-01	-1Y	0.066405154480108
2020-03-01	-1Y	0.066405154480108
2020-04-01	-1Y	0.066405154480108
2020-05-01	-1Y	0.066405154480108
2020-06-01	-1Y	0.066405154480108
2020-07-01	-1Y	0.066405154480108
2020-08-01	-1Y	0.066405154480108
2020-09-01	-1Y	0.066405154480108
2020-10-01	-1Y	0.066405154480108
2020-11-01	-1Y	0.066405154480108
2020-12-01	-1Y	0.066405154480108
2021-01-01	-1Y	0.066405154480108
2021-02-01	-1Y	0.066405154480108
2021-03-01	-1Y	0.066405154480108
2021-04-01	-1Y	0.066405154480108
2021-05-01	-1Y	0.066405154480108
2021-06-01	-1Y	0.066405154480108
2021-07-01	-1Y	0.066405154480108
2021-08-01	-1Y	0.066405154480108
2021-09-01	-1Y	0.066405154480108
2021-10-01	-1Y	0.066405154480108
2021-11-01	-1Y	0.066405154480108
2021-12-01	-1Y	0.066405154480108
2022-01-01	-1Y	0.066405154480108
2022-02-01	-1Y	0.066405154480108
2022-03-01	-1Y	0.066405154480108
2022-04-01	-1Y	0.066405154480108
2022-05-01	-1Y	0.066405154480108
2022-06-01	-1Y	0.066405154480108
2022-07-01	-1Y	0.066405154480108
2022-08-01	-1Y	0.066405154480108
2022-09-01	-1Y	0.066405154480108
2022-10-01	-1Y	0.066405154480108
2022-11-01	-1Y	0.066405154480108
2022-12-01	-1Y	0.066405154480108
2023-01-01	-1Y	0.066405154480108
2023-02-01	-1Y	0.066405154480108
2023-03-01	-1Y	0.066405154480108
2023-04-01	-1Y	0.066405154480108
2023-05-01	-1Y	0.066405154480108
2023-06-01	-1Y	0.066405154480108
2023-07-01	-1Y	0.066405154480108
2023-08-01	-1Y	0.066405154480108
2023-09-01	-1Y	0.066405154480108
2023-10-01	-1Y	0.066405154480108
2023-11-01	-1Y	0.066405154480108
2023-12-01	-1Y	0.066405154480108
2024-01-01	-1Y	0.066405154480108
2024-02-01	-1Y	0.066405154480108
2024-03-01	-1Y	0.066405154480108
2024-04-01	-1Y	0.066405154480108
2024-05-01	-1Y	0.066405154480108
2024-06-01	-1Y	0.066405154480108
2024-07-01	-1Y	0.066405154480108
2024-08-01	-1Y	0.066405154480108
2024-09-01	-1Y	0.066405154480108
2024-10-01	-1Y	0.066405154480108
2024-11-01	-1Y	0.066405154480108
2024-12-01	-1Y	0.066405154480108
2025-01-01	-1Y	0.066405154480108
2025-02-01	-1Y	0.066405154480108
2025-03-01	-1Y	0.066405154480108
2025-04-01	-1Y	0.066405154480108
2025-05-01	-1Y	0.066405154480108
2025-06-01	-1Y	0.066405154480108
2025-07-01	-1Y	0.066405154480108
2025-08-01	-1Y	0.066405154480108
2025-09-01	-1Y	0.066405154480108
2025-10-01	-1Y	0.066405154480108
2025-11-01	-1Y	0.066405154480108
2025-12-01	-1Y	0.066405154480108
2026-01-01	-1Y	0.066405154480108
2026-02-01	-1Y	0.066405154480108
2026-03-01	-1Y	0.066405154480108
2026-04-01	-1Y	0.066405154480108
2026-05-01	-1Y	0.066405154480108
2026-06-01	-1Y	0.066405154480108
2026-07-01	-1Y	0.066405154480108
2026-08-01	-1Y	0.066405154480108
2026-09-01	-1Y	0.066405154480108
2026-10-01	-1Y	0.066405154480108
2026-11-01	-1Y	0.066405154480108
2026-12-01	-1Y	0.066405154480108
2027-01-01	-1Y	0.066405154480108
2027-02-01	-1Y	0.066405154480108
2027-03-01	-1Y	0.066405154480108
2027-04-01	-1Y	0.066405154480108
2027-05-01	-1Y	0.066405154480108
2027-06-01	-1Y	0.066405154480108
2027-07-01	-1Y	0.066405154480108
2027-08-01	-1Y	0.066405154480108
2027-09-01	-1Y	0.066405154480108
2027-10-01	-1Y	0.066405154480108
2027-11-01	-1Y	0.066405154480108
2027-12-01	-1Y	0.066405154480108
2028-01-01	-1Y	0.066405154480108
2028-02-01	-1Y	0.066405154480108
2028-03-01	-1Y	0.066405154480108
2028-04-01	-1Y	0.066405154480108
2028-05-01	-1Y	0.066405154480108
2028-06-01	-1Y	0.066405154480108
2028-07-01	-1Y	0.066405154480108
2028-08-01	-1Y	0.066405154480108
2028-09-01	-1Y	0.066405154480108
2028-10-01	-1Y	0.066405154480108
2028-11-01	-1Y	0.066405154480108
2028-12-01	-1Y	0.066405154480108
2029-01-01	-1Y	0.066405154480108
2029-02-01	-1Y	0.066405154480108
2029-03-01	-1Y	0.066405154480108
2029-04-01	-1Y	0.066405154480108
2029-05-01	-1Y	0.066405154480108</td

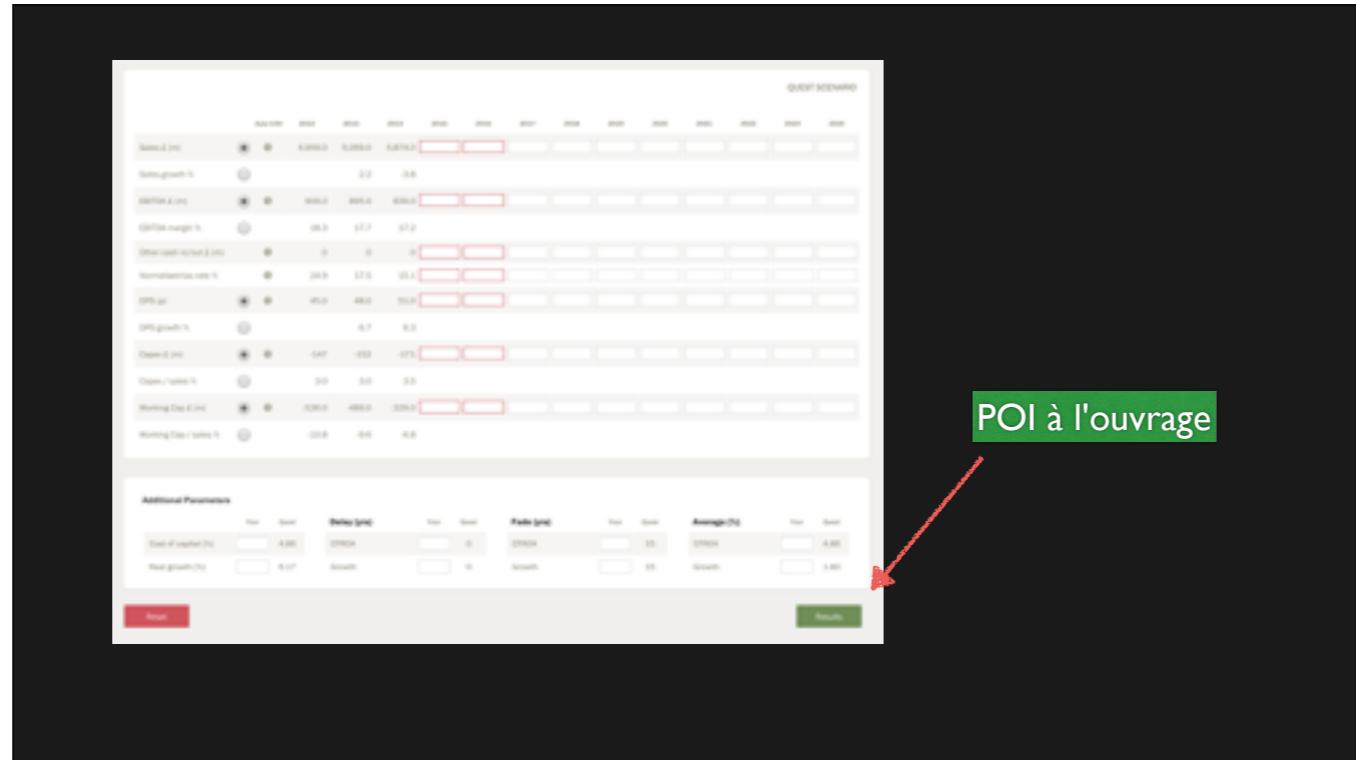
Encore du contexte !



Condition	Description		Condition	Description
C	Current customer without debt			
D	Debt			
E	Heavy material			
F	Highly leveraged			
G	Low credit risk value			
H	Low customer quality value			
I	Medium			
J	Medium			
K	Medium			
L	Medium			
M	Medium			
N	Medium			
O	Medium			
P	Medium			
Q	Medium			
R	Medium			
S	Medium			
T	Medium			
U	Medium			
V	Medium			
W	Medium			
X	Medium			
Y	Medium			
Z	Medium			
A1	Medium			
B1	Medium			
C1	Medium			
D1	Medium			
E1	Medium			
F1	Medium			
G1	Medium			
H1	Medium			
I1	Medium			
J1	Medium			
K1	Medium			
L1	Medium			
M1	Medium			
N1	Medium			
O1	Medium			
P1	Medium			
Q1	Medium			
R1	Medium			
S1	Medium			
T1	Medium			
U1	Medium			
V1	Medium			
W1	Medium			
X1	Medium			
Y1	Medium			
Z1	Medium			
A2	Medium			
B2	Medium			
C2	Medium			
D2	Medium			
E2	Medium			
F2	Medium			
G2	Medium			
H2	Medium			
I2	Medium			
J2	Medium			
K2	Medium			
L2	Medium			
M2	Medium			
N2	Medium			
O2	Medium			
P2	Medium			
Q2	Medium			
R2	Medium			
S2	Medium			
T2	Medium			
U2	Medium			
V2	Medium			
W2	Medium			
X2	Medium			
Y2	Medium			
Z2	Medium			
A3	Medium			
B3	Medium			
C3	Medium			
D3	Medium			
E3	Medium			
F3	Medium			
G3	Medium			
H3	Medium			
I3	Medium			
J3	Medium			
K3	Medium			
L3	Medium			
M3	Medium			
N3	Medium			
O3	Medium			
P3	Medium			
Q3	Medium			
R3	Medium			
S3	Medium			
T3	Medium			
U3	Medium			
V3	Medium			
W3	Medium			
X3	Medium			
Y3	Medium			
Z3	Medium			
A4	Medium			
B4	Medium			
C4	Medium			
D4	Medium			
E4	Medium			
F4	Medium			
G4	Medium			
H4	Medium			
I4	Medium			
J4	Medium			
K4	Medium			
L4	Medium			
M4	Medium			
N4	Medium			
O4	Medium			
P4	Medium			
Q4	Medium			
R4	Medium			
S4	Medium			
T4	Medium			
U4	Medium			
V4	Medium			
W4	Medium			
X4	Medium			
Y4	Medium			
Z4	Medium			
A5	Medium			
B5	Medium			
C5	Medium			
D5	Medium			
E5	Medium			
F5	Medium			
G5	Medium			
H5	Medium			
I5	Medium			
J5	Medium			
K5	Medium			
L5	Medium			
M5	Medium			
N5	Medium			
O5	Medium			
P5	Medium			
Q5	Medium			
R5	Medium			
S5	Medium			
T5	Medium			
U5	Medium			
V5	Medium			
W5	Medium			
X5	Medium			
Y5	Medium			
Z5	Medium			
A6	Medium			
B6	Medium			
C6	Medium			
D6	Medium			
E6	Medium			
F6	Medium			
G6	Medium			
H6	Medium			
I6	Medium			
J6	Medium			
K6	Medium			
L6	Medium			
M6	Medium			
N6	Medium			
O6	Medium			
P6	Medium			
Q6	Medium			
R6	Medium			
S6	Medium			
T6	Medium			
U6	Medium			
V6	Medium			
W6	Medium			
X6	Medium			
Y6	Medium			
Z6	Medium			
A7	Medium			
B7	Medium			
C7	Medium			
D7	Medium			
E7	Medium			
F7	Medium			
G7	Medium			
H7	Medium			
I7	Medium			
J7	Medium			
K7	Medium			
L7	Medium			
M7	Medium			
N7	Medium			
O7	Medium			
P7	Medium			
Q7	Medium			
R7	Medium			
S7	Medium			
T7	Medium			
U7	Medium			
V7	Medium			
W7	Medium			
X7	Medium			
Y7	Medium			
Z7	Medium			
A8	Medium			
B8	Medium			
C8	Medium			
D8	Medium			
E8	Medium			
F8	Medium			
G8	Medium			
H8	Medium			
I8	Medium			
J8	Medium			
K8	Medium			
L8	Medium			
M8	Medium			
N8	Medium			
O8	Medium			
P8	Medium			
Q8	Medium			
R8	Medium			
S8	Medium			
T8	Medium			
U8	Medium			
V8	Medium			
W8	Medium			
X8	Medium			
Y8	Medium			
Z8	Medium			
A9	Medium			
B9	Medium			
C9	Medium			
D9	Medium			
E9	Medium			
F9	Medium			
G9	Medium			
H9	Medium			
I9	Medium			
J9	Medium			
K9	Medium			
L9	Medium			
M9	Medium			
N9	Medium			
O9	Medium			
P9	Medium			
Q9	Medium			
R9	Medium			
S9	Medium			
T9	Medium			
U9	Medium			
V9	Medium			
W9	Medium			
X9	Medium			
Y9	Medium			
Z9	Medium			
A10	Medium			
B10	Medium			
C10	Medium			
D10	Medium			
E10	Medium			
F10	Medium			
G10	Medium			
H10	Medium			
I10	Medium			
J10	Medium			
K10	Medium			
L10	Medium			
M10	Medium			
N10	Medium			
O10	Medium			
P10	Medium			
Q10	Medium			
R10	Medium			
S10	Medium			
T10	Medium			
U10	Medium			
V10	Medium			
W10	Medium			
X10	Medium			
Y10	Medium			
Z10	Medium			
A11	Medium			
B11	Medium			
C11	Medium			
D11	Medium			
E11	Medium			
F11	Medium			
G11	Medium			
H11	Medium			
I11	Medium			
J11	Medium			
K11	Medium			
L11	Medium			
M11	Medium			
N11	Medium			
O11	Medium			
P11	Medium			
Q11	Medium			
R11	Medium			
S11	Medium			
T11	Medium			
U11	Medium			
V11	Medium			

#	Run	Date	By	Duration	VEDoc	S&P	Logs	Historical	Success	Website	Jump	Status	Actions
291504070029	Petroleum Geo-Services ASA	07-04-2015 15:24PM	Geoff Ruby	22.29 s	201504070018				1				Run Again
291504070028	UK	07-04-2015 15:21PM	Graham Simpson	1.401 min	201504070017				742				Run Again
291504070027	UK	07-04-2015 14:56PM	Graham Simpson	1.837 min	201504070016				742				Run Again
291504070026	Wowow Inc.	07-04-2015 14:48PM	Geoff Ruby	29.42 s	201504070015				1				Run Again
291504070025	Wowow Inc.	07-04-2015 14:46PM	Geoff Ruby	35.71 s	201504070014				1				Run Again
291504070024	UK	07-04-2015 14:39PM	Graham Simpson	1.548 min	201504070013				742				Run Again
291504070023	National Research Corp.	07-04-2015 14:26PM	Geoff Ruby	20.90 s	201504070012				1				Run Again
291504070022	Henkel AG & Co. KGaA	07-04-2015 14:18PM	Geoff Ruby	22.50 s	201504070011				1				Run Again
291504070021	UK	07-04-2015 14:14PM	Graham Simpson	1.422 min	201504070010				742				Run Again
291504070020	UK	07-04-2015 14:04PM	Graham Simpson	1.242 min	201504070009				742				Run Again

Business in control !



POI à l'ouvrage

Annexes

#grandmere

DEVO^X France

@framiere

Références

- <http://www.antlr.org/>
- <http://poi.apache.org/>
- <http://eclipse.org/Xtext/>
- <http://labs.carrotsearch.com/junit-benchmarks.html>
- Le code : <https://github.com/framiere/devoxx-grand-mere/>

#grandmere

@framiere