



E-mail Address Reliability

Vandy BERTEN
Isabelle BOYDENS

Section Recherche



Table des matières

Introduction

Contexte

Difficultés et incertitudes

Protocoles et mécanismes

Syntaxe

Contraintes syntaxiques générales et spécifiques

Méthodes traditionnelles

Propositions

Validation

Existence d'une adresse

Contrôle de consultation

Matching

Présomptions d'erreurs

Dédoublonnage

Bonnes pratiques & Conclusions



A conceptual image showing a hand pointing at a glowing envelope icon on a world map. The map is overlaid with a network of glowing white lines and dots, representing global connectivity. Several other envelope icons are scattered across the map, connected by these lines. The background is dark blue with a subtle world map outline.

Introduction

Table des matières

Introduction

Contexte & enjeux

Organisation

On-line vs Batch

Exemples

Difficultés et incertitudes

Protocoles et mécanismes

Syntaxe

Validation

Matching

Bonnes pratiques
& Conclusions



Problématique

- De plus en plus d'administrations ou sociétés utilisent/veulent utiliser les **adresses e-mail** :
 - Contacts avec les citoyens/clients
 - Recommandé électronique (notifications)
 - V-ICT-OR veut les ajouter au Registre National
- Or :
 - Les DB d'e-mail sont souvent de **mauvaise qualité**
 - Les processus mis en place ne permettent en général pas de **maintenir un niveau correct**



Mauvaise qualité

- Campagnes de communication:
 - Jusqu'à 20% de « bounce »
 - Taux de lecture confirmé faible ($\pm 20-25\%$)
- Certains organismes n'osent pas utiliser leur propres DB ...
- Pourquoi ?
 - Quasiment jamais de contrôle en entrée, ou minimaliste
 - Aucun suivi dans le temps
 - Incohérence de flux
 - Cumul d'incertitudes

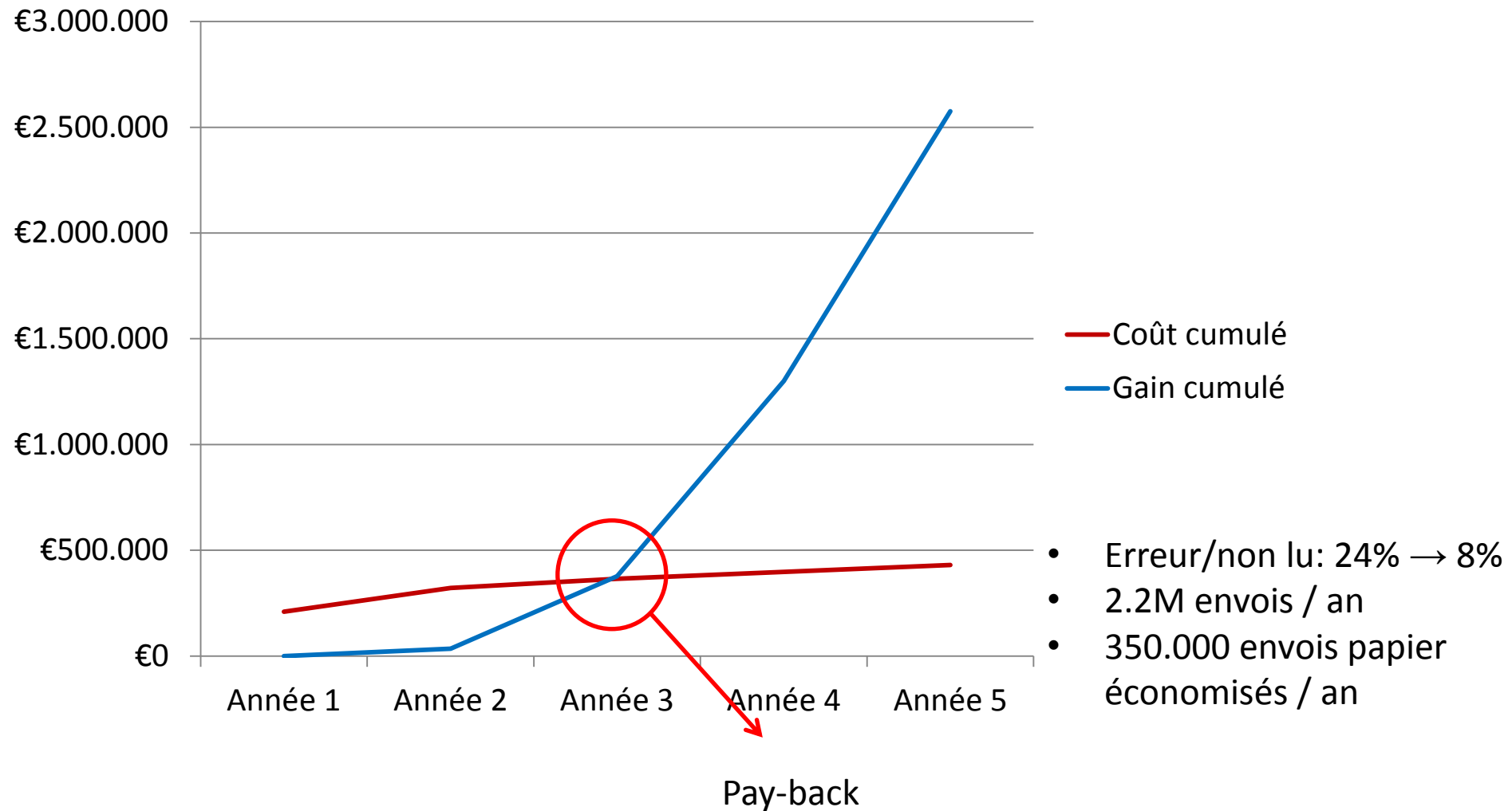


Qualité : pourquoi l'améliorer

- Gains **directs** : ROI fonction des usages et du contexte:
 - Man-power réduit si bonne qualité
 - Communications officielles : diminutions des **envois papiers**
 - Gains potentiels à terme en **millions d'euros**
- Gains **indirects** :
 - Amélioration des services rendus et de l'efficacité
 - Crédibilité, législation



ROI



Qualité : comment l'améliorer

Il existe de nombreuses techniques connues,
pas/peu/mal appliquées :

- Vérification syntaxique
- Validation par envoi d'e-mail de confirmation
- Suivi dans le temps par indicateurs de lecture
- Rétroaction en cas d'erreur à l'envoi



Qualité : comment l'améliorer

Apports originaux de notre étude :

- Amélioration notable de la **qualité** des tests
 - Syntaxe **spécifique** (↗ 15-20%)
 - Mise en évidence d'adresses **suspectes** (↗ 5-10%)
 - Amélioration des **techniques de validation** « batch »
- **Suggestions** de correction
 - Erreurs syntaxiques
 - Comparaison avec info annexes : nom, prénom
- Compilation de **best-practices**

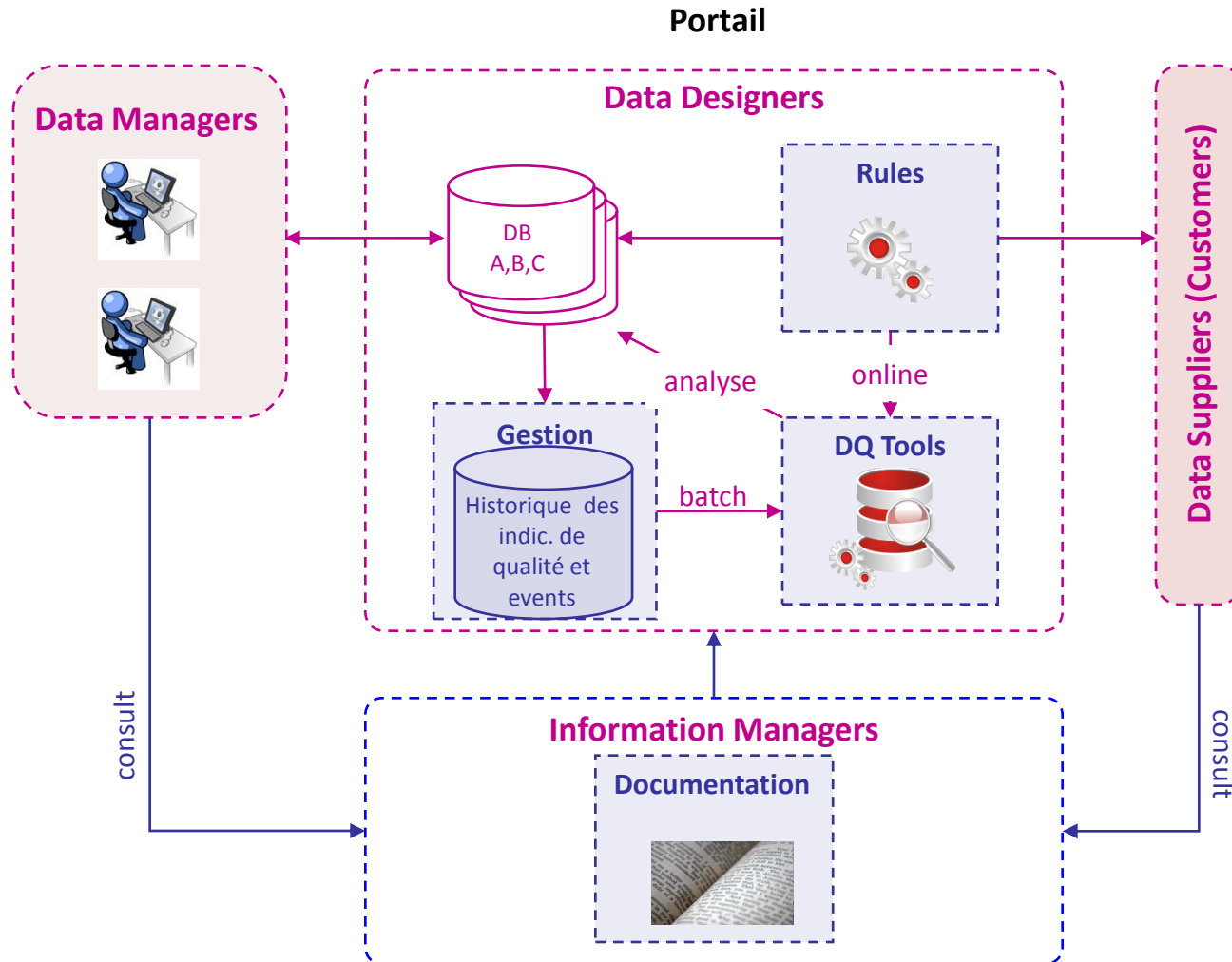


Contexte et enjeux

- Cumul d'**incertitudes** :
 - Volatilité des usages
 - Dynamique des noms de domaines
 - Syntaxe non standard
- Nécessité d'un **bénéfice** ou d'un intérêt des mises à jour pour les utilisateurs
- Objectifs de l'étude :
 - Contrôles et outils performants
 - Indicateurs de qualité en vue d'un monitoring
 - Bonnes pratiques de gestion & d'amélioration continue
 - Organisation adéquate



Organisation



On-line vs Batch

On-line	Batch
Sur un portail, à l'enregistrement	DB existante
Doit être rapide !	On a le temps, étude poussée possible
Interroger l'utilisateur → facile	Interroger l'utilisateur → plus difficile
Envoyer un e-mail + action → facile	Envoyer un e-mail → plus difficile
Ne dispense pas du batch par la suite!	S'applique sur des DB avec ou sans contrôle à l'entrée Si contrôle, se focalise sur l'évolution (DN, usage)

La plupart des méthodes s'appliquent aux deux

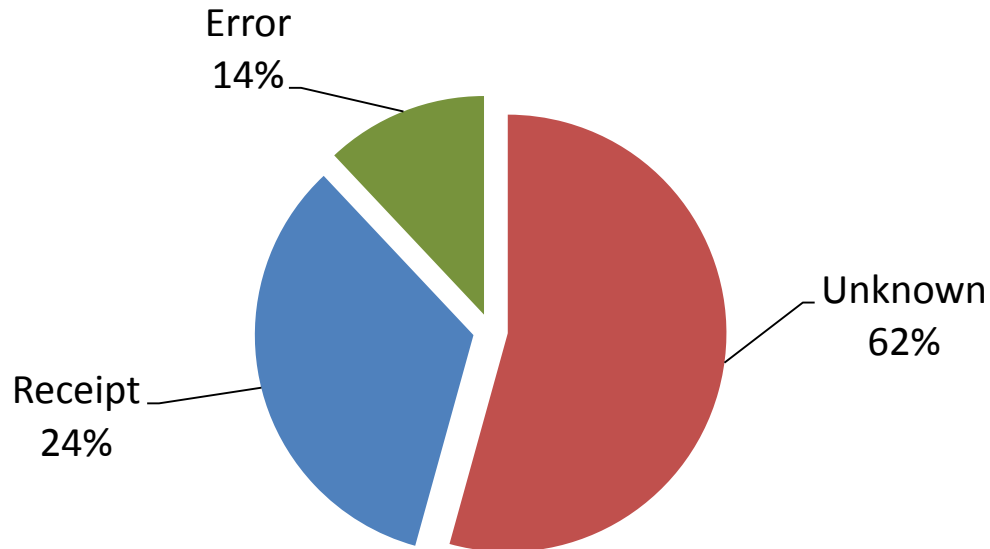


Étude basée sur ...

- La **littérature** (marketing, technique et e-gov à l'étranger)
- Des **tests et expériences** abondants sur des **grandes bases de données** (échantillons)
- Les **Data Quality Tools** et des **développements** propres
- 10 ans d'expérience en Data Quality (**DQ Cell**)
- Des contacts multiples avec le **terrain**, le développement, des services opérationnels et juridiques



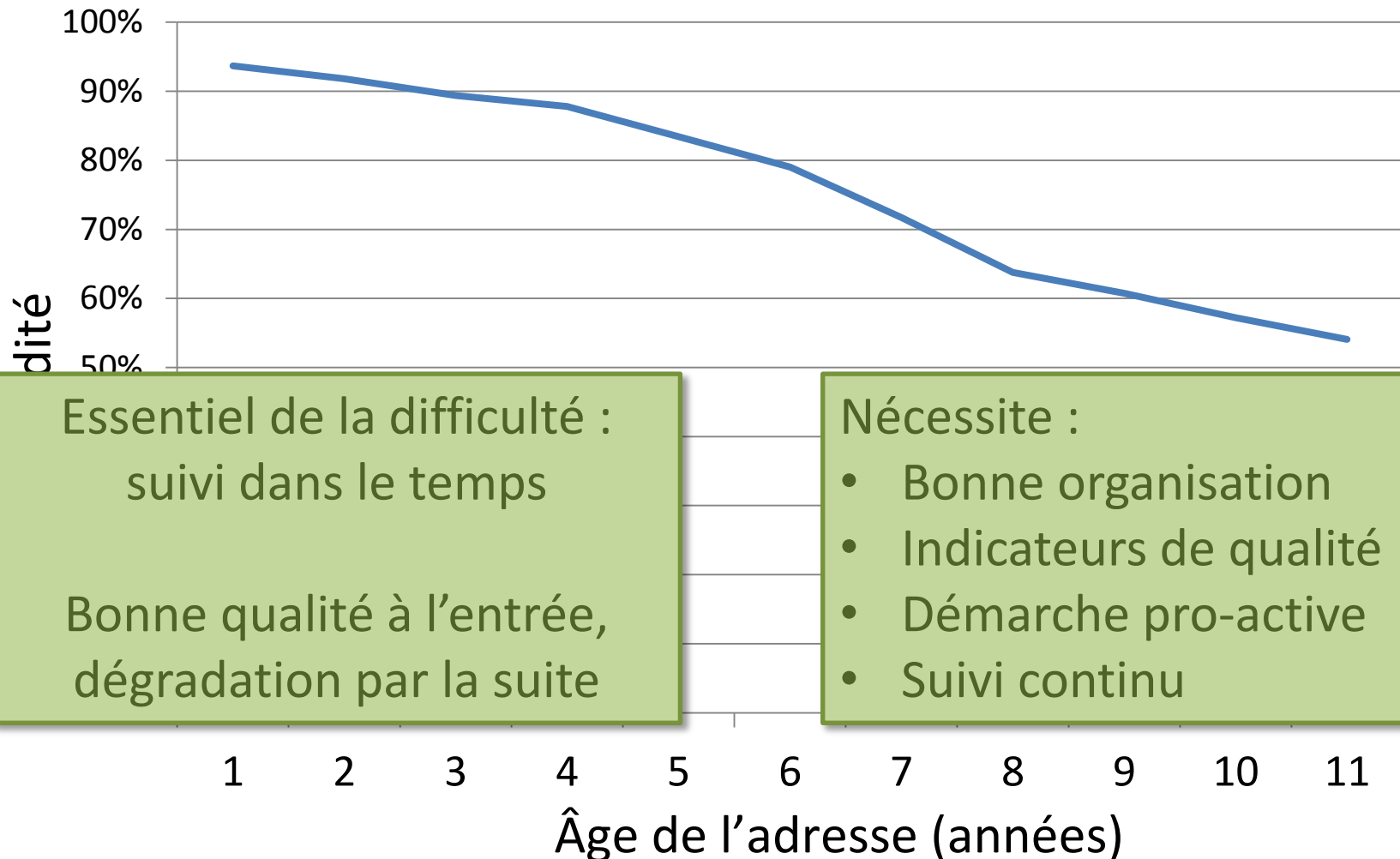
Exemple typique de situation



- Coûts directs et indirects
- Importance d'un contrôle à la source et continu !
- Nécessité d'une rétro-action



Dégressivité de la validité



Vérification/validation : Étapes

Albert.Leroy@smals.be



Vérification/validation : Étapes

Vérification syntaxique

- Présence d'un (et un seul) « @ », absence d'espace, ...
- **Standards non respectés** : restrictions et extensions

Albert.Leroy@smals.be



Vérification/validation : Étapes

Albert.Leroy@smals.be

Top Level Domain (TLD)

- Aujourd'hui : plutôt statique, facile à valider (+/- 280)
- Prochainement : .brussels, .vlaanderen, .中国, .இந்தியா, .آزمایشی



Vérification/validation : Étapes

Nom de domaine (DN)

- Dans la pratique : a-z, 0-9, « - », « . »
- Dans le futur : accents, autres alphabets (IDN)
- Dynamique ; .be : **1300 changements**/jour, monde : **300k!**

Albert.Leroy@smals.be



Vérification/validation : Étapes

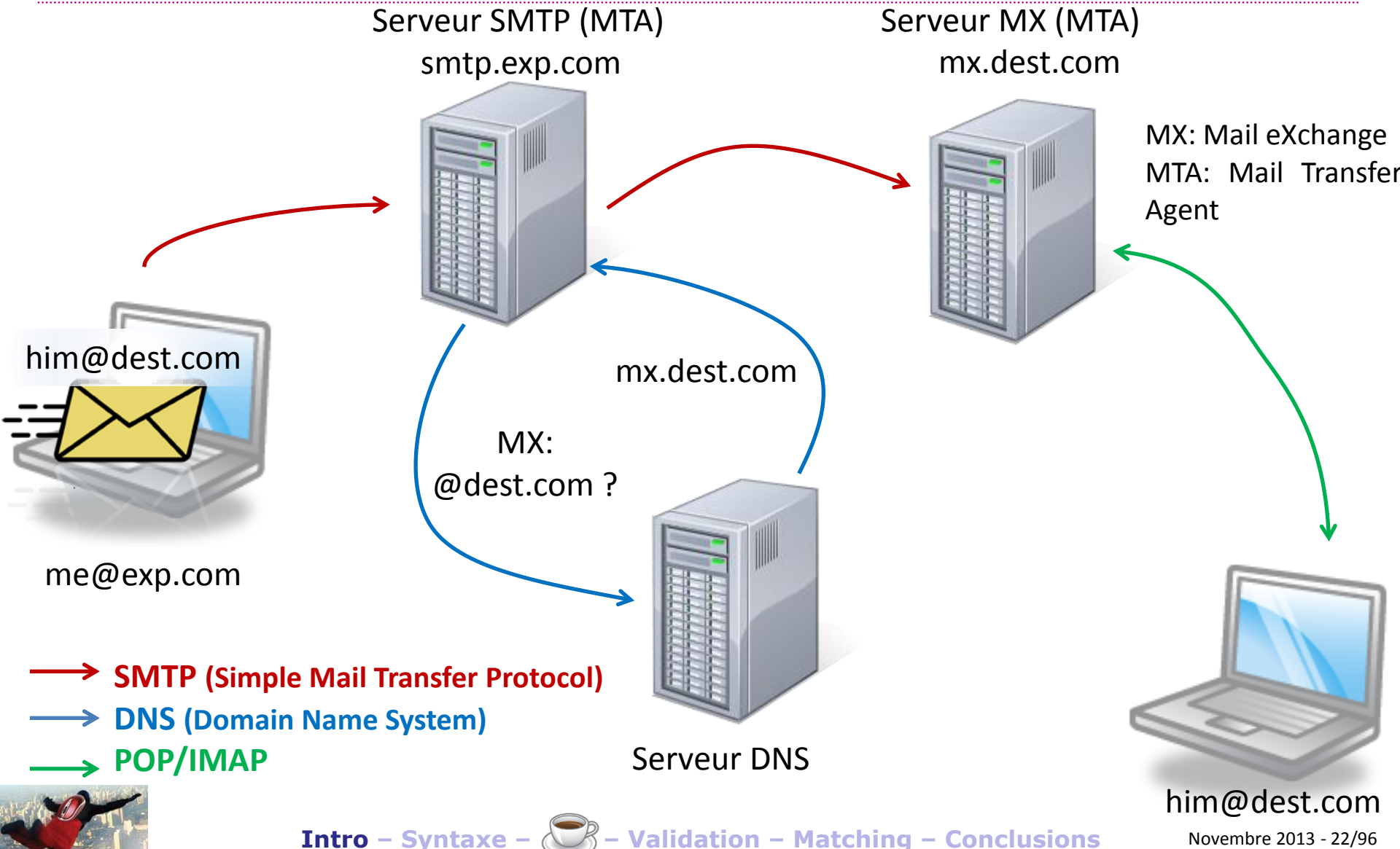
Albert.Leroy@smals.be

Username

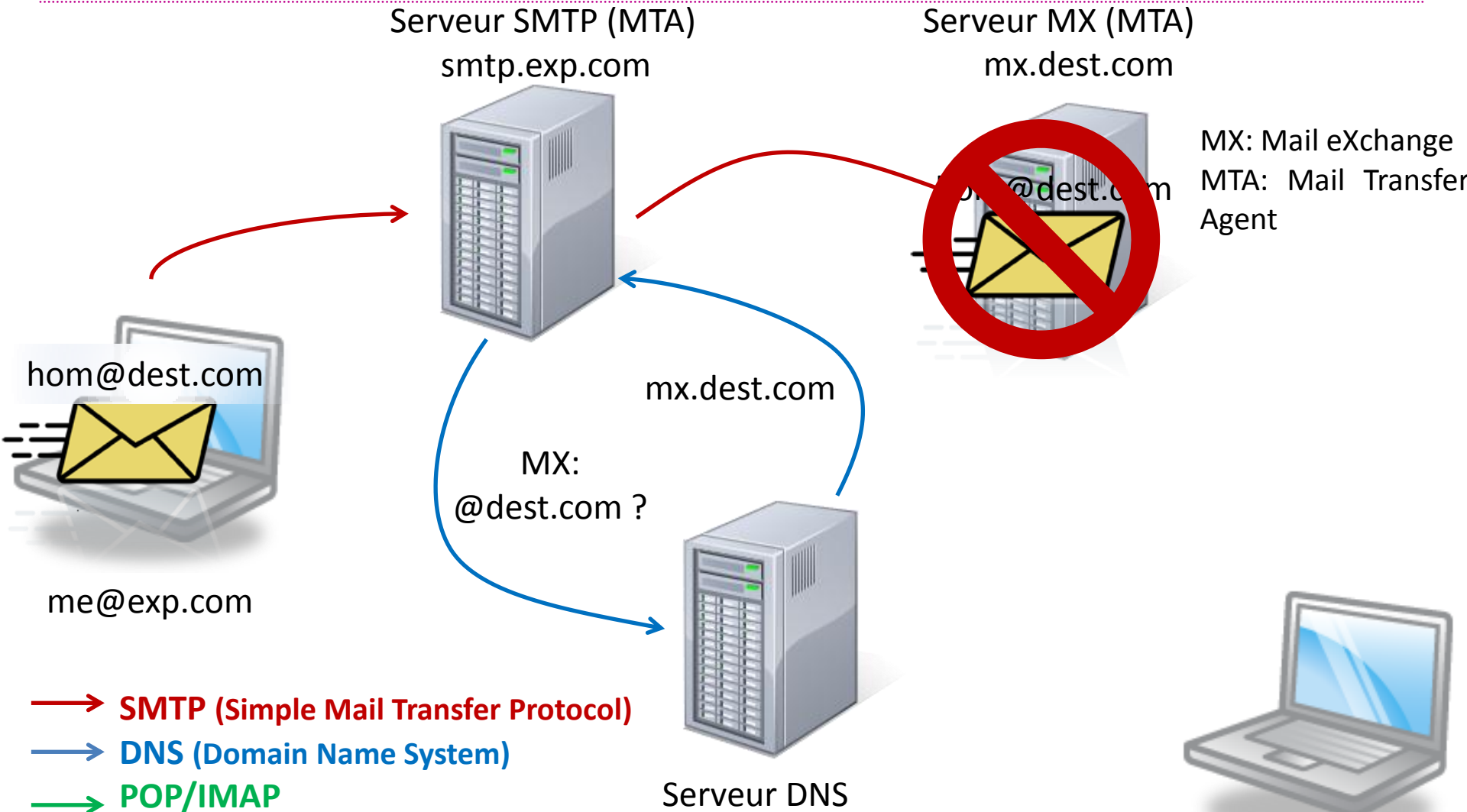
- Validation *a priori* (avant envoi) :
 - Utilisation du protocole SMTP
- Validation *a posteriori* (après envoi) :
 - Analyse de « bounce »
 - Contrôle de lecture (image, lien, ...)



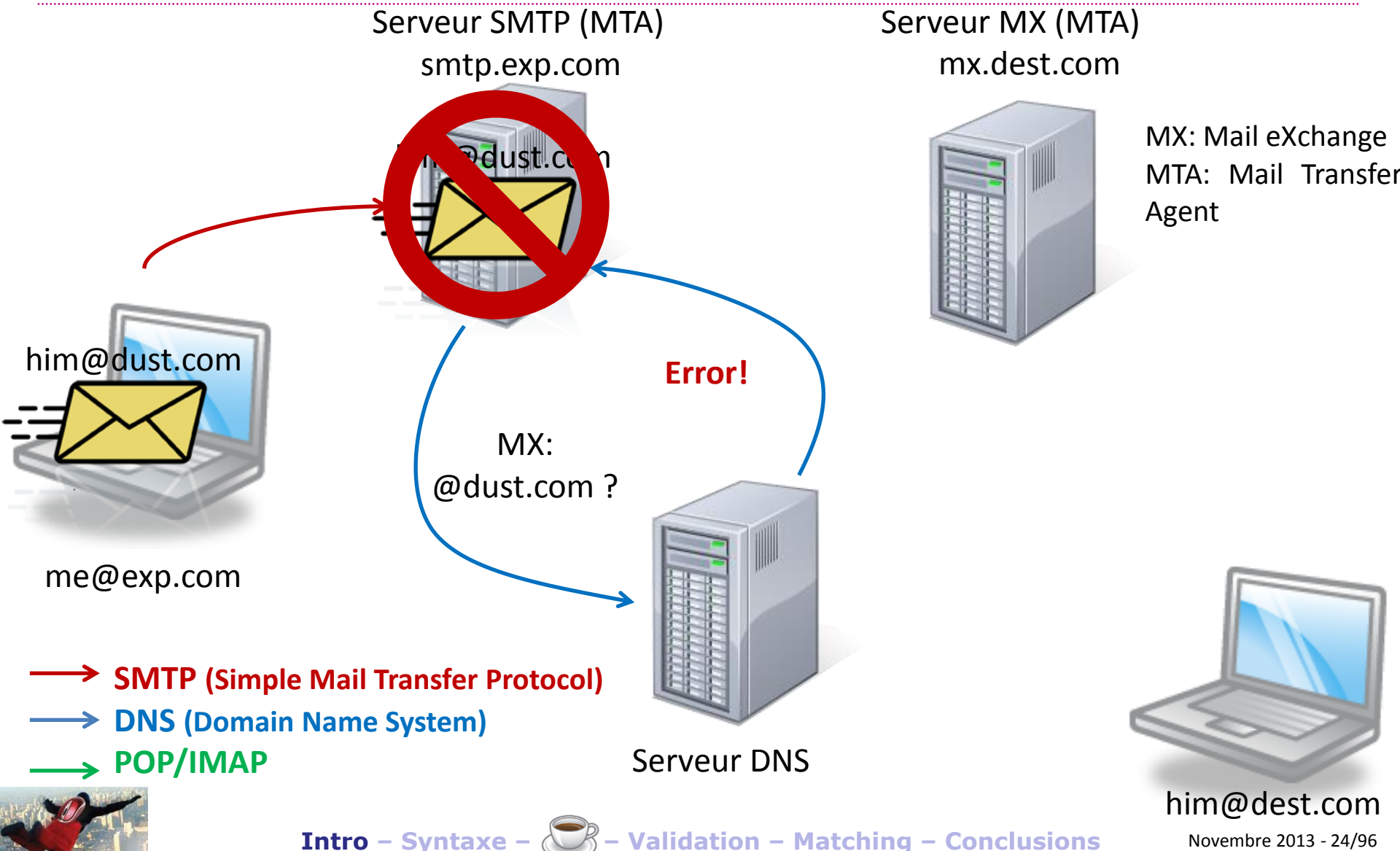
Mécanisme d'envoi d'un e-mail



Envoi d'un e-mail : bounce



Envoi d'un e-mail : bounce





Syntaxe

Table des matières

Introduction

Syntaxe

Contexte

Les standards et la pratique

Techniques de contrôle et limites

Syntaxes spécifiques

Propositions

Outils

Validation

Matching

Bonnes pratiques &
Conclusions



Contexte

- Syntaxe = vérification « orthographique » :
 - **Obligatoire** : un (et un seul) arobase (@)
 - **Interdit** : espace, virgule, point-virgule
 - Points non consécutifs, ...
- Ne dit pas si l'adresse/le domaine/le TLD **existe** !
- Analogie :
 - Code postal belge : 4 chiffres
 - 1234 respecte la syntaxe, mais n'est pas un CP !
 - Idem avec les numéros de téléphone
- Standards : RFC 5321 et 5322



Contexte

Erreurs évidentes

albert.leroy#smals.be

albert.leroy@smals,be

albert leroy@smals.be

0471/257 800

Rue Fonsy 20

www.smals.be



Erreurs ??

albêrt.leroy@简体中文.com

albert.leroy@be

albert-leroy@gmail.com

albert..leroy@smals.be

albert%leroy@blahblah.be

albert.leroy@a--b.be

-----@hotmail.com

albert@ma_boite.be



Contexte

- Intérêt de vérifier la syntaxe ?
 - Batch : Identifier des **erreurs présentes**
 - On-line : Détection des erreurs à un **stade précoce** (avant envoi d'e-mail de confirmation)
- Dans la pratique :
 - Beaucoup de portails (officiels) **sans le moindre contrôle** ou minime
 - Souvent : contrôles beaucoup **trop stricts**
- Problème des **flux d'entrée multiples** :
 - Un point d'entrée avec contrôle, un autre sans contrôle



Syntaxe : que disent les standards ?

- Syntaxe « classique » :
 - [a-z], [0-9]
 - « . » et « - » (non consécutifs, pas en début ou en fin)
 - Dernière partie (TLD) : [a-z]{2,6}
- Nouveautés :
 - gTLD : .brussels, .vlaanderen, ...
 - IDN : ñandú.cl , 简体中文.com, ...
 - IDN ccTLD : .中国, .இந்தியா, .آزمایشی, .pф, ...

Albert.Leroy@smals.be

- Caractères normaux et accentués (case sensitive)
- .!#\$%&'()*+,-/=:?^_`{|~"
- Points non consécutifs, pas en début ou en fin



Dans la pratique ?

- Partie « nom de domaine » :
 - Respect imposé par la **structure « DNS »**
 - Un nom de domaine **non-conforme** n'apparaît **pas dans les tables**
- Partie « username » :
 - Uniquement évaluée par le serveur **mail de destination**
 - Username attribué par l'organisation qui le gère → une certaine **liberté** malgré les standards !
 - Dans la pratique : beaucoup plus **restrictif** que la norme
 - Rarement **case sensitive**
 - Certains acceptent des **extensions**



Syntaxe spécifique : intérêt ?

- Il est facile de trouver la **syntaxe spécifique** pour la plupart des grands fournisseurs (Gmail, Hotmail, Belgacom, Telenet, ...)
- Sur certaines DB étudiées : **85% des adresses !**
- Permet d'être **beaucoup plus restrictif** sur ce qu'on laisse passer, sans risquer les « **faux négatifs** »



Syntaxe spécifique : exemples

- Hotmail-live-outlook-..., Belgacom-skynet-..., telenet, pandora :

- « a-z » « 0-9 » « - » « _ » « . »
- Pas de points consécutifs, en début ou en fin



- Yahoo :

- « a-z » « 0-9 » « _ » « . » (pas le tiret !)
- Maximum un point
- 1^{er} : a-z ; dernier : a-z, 0-9
- Entre 4 et 32 caractères
- Si un point : 4 caractères après

YAHOO!



Syntaxe spécifique : Gmail

- « a-z », « 0-9 », « . », « + »
 - Entre 6 et 30 caractères, en ignorant les points et ce qui suit le « + »
 - Plus de 8 caractères : minimum une lettre
 - Les points sont ignorés, « + » débute un commentaire. Sont équivalents et légaux :
 - albert.leroy@gmail.com
 - albertleroy@gmail.com
 - albert.leroy+blahblah@gmail.com
 - albert..leroy@gmail.com
 - .albert.leroy.@gmail.com
- } Interdits par les standards !

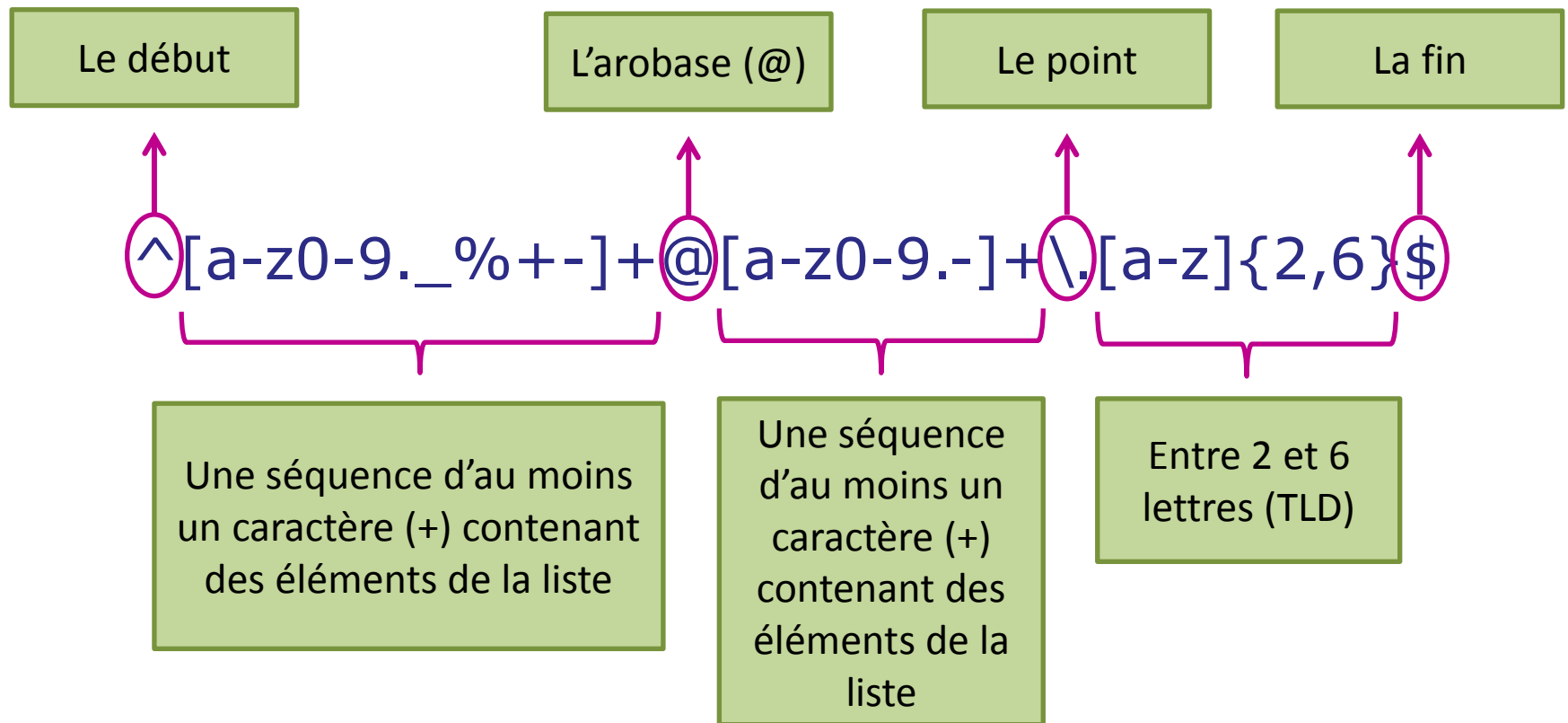


Comment vérifier ?

- Technique de vérification la plus répandue : **les expressions régulières**
- Sorte de **mini-langage de programmation**, utilisable de façon (quasi) standard par (quasi) tous les langages
- **Très puissant** pour vérifier qu'une **chaîne de caractères** rencontre bien certaines **contraintes**
- A malgré tout quelques **limites**



Expressions régulières



Expressions régulières

- On trouve beaucoup de variantes d'expressions régulières de vérification d'e-mail
- La plupart conviennent pour l'énorme majorité des adresses en cours
- Certaines acceptent beaucoup trop
- D'autres refusent des adresses valides
- Parfois : totalement illisible



Expression régulière : exemples

- Champ « email » en HTML :

$^{\wedge}[a-z0-9.!\$ \% \& ' * + / = ? ^ _ \backslash \{ | \} \sim -] + @$

$[a-z0-9-] + (\backslash . [a-z0-9-] +) * \$$

- N'accepte pas les accents
- Accepte ...@--.--, a@b.55

- Expression commune :

$^{\wedge}[a-z0-9._ \% + -] + @[a-z0-9.-] + \backslash . [a-z] \{ 2, 4 \} \$$

- Petite liste de caractères
- N'accepte pas les TLD .museum ou .travel
- Accepte ...@---...be
- Refuse albert@be



- Dans une librairie Javascript répandue :
- ```
^(((([a-z]|\d|![#\$\%&'*\+|-\/=\?\\^_`{\}|}~])|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF]))+(\.([a-z]|\d|![#\$\%&'*\+|-\/=\?\\^_`{\}|}~])|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF]))+)*|((\x22)((((\x20|\x09)*(\x0d\x0a))?(\x20|\x09)+)?(([\x01-\x08\x0b\x0c\x0e-\x1f\x7f]|\x21|[\x23-\x5b]|[\x5d-\x7e]|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])|(\x01-\x09\x0b\x0c\x0d-\x7f)|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])))*)*(((\x20|\x09)*(\x0d\x0a))?(\x20|\x09)+)?(\x22)))@((((([a-z]|\d|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])|((([a-z]|\d|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF]))([a-z]|\d|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF]))*([a-z]|\d|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF]))))\.)+((([a-z]|\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF)|((([a-z]|\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF))([a-z]|\d|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF]))*([a-z]|\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF))))\.\?$.
```
- Accepte beaucoup de caractères (y compris chinois, arabe, ...)
- Accepte « " albert.leroy "@smals.be »
- Rejette « albert@be »
- Accepte « albert@gm.--ail.c0m »



# Expressions régulières spécifiques

---

- Pour les domaines pour lesquels on connaît une syntaxe spécifique (Hotmail, Yahoo, Gmail, ...), on peut proposer un test spécifique sur le « username ».
- Exemple pour Hotmail :

$$^[a-z0-9_-](\.[a-z0-9_-]+)*\$$$

- Pour d'autres, des tests supplémentaires sont nécessaires



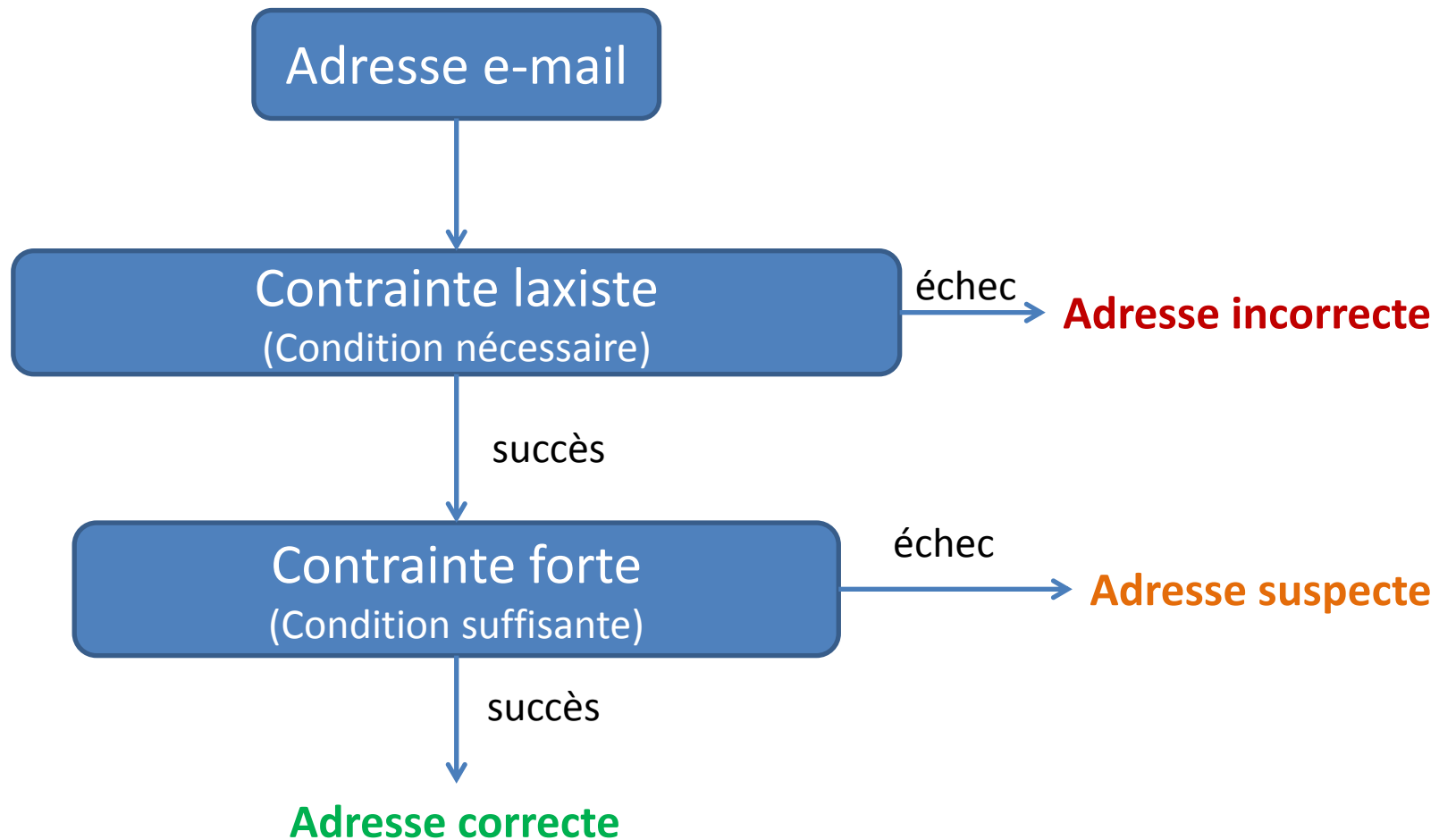
# Expressions régulières : les limites

---

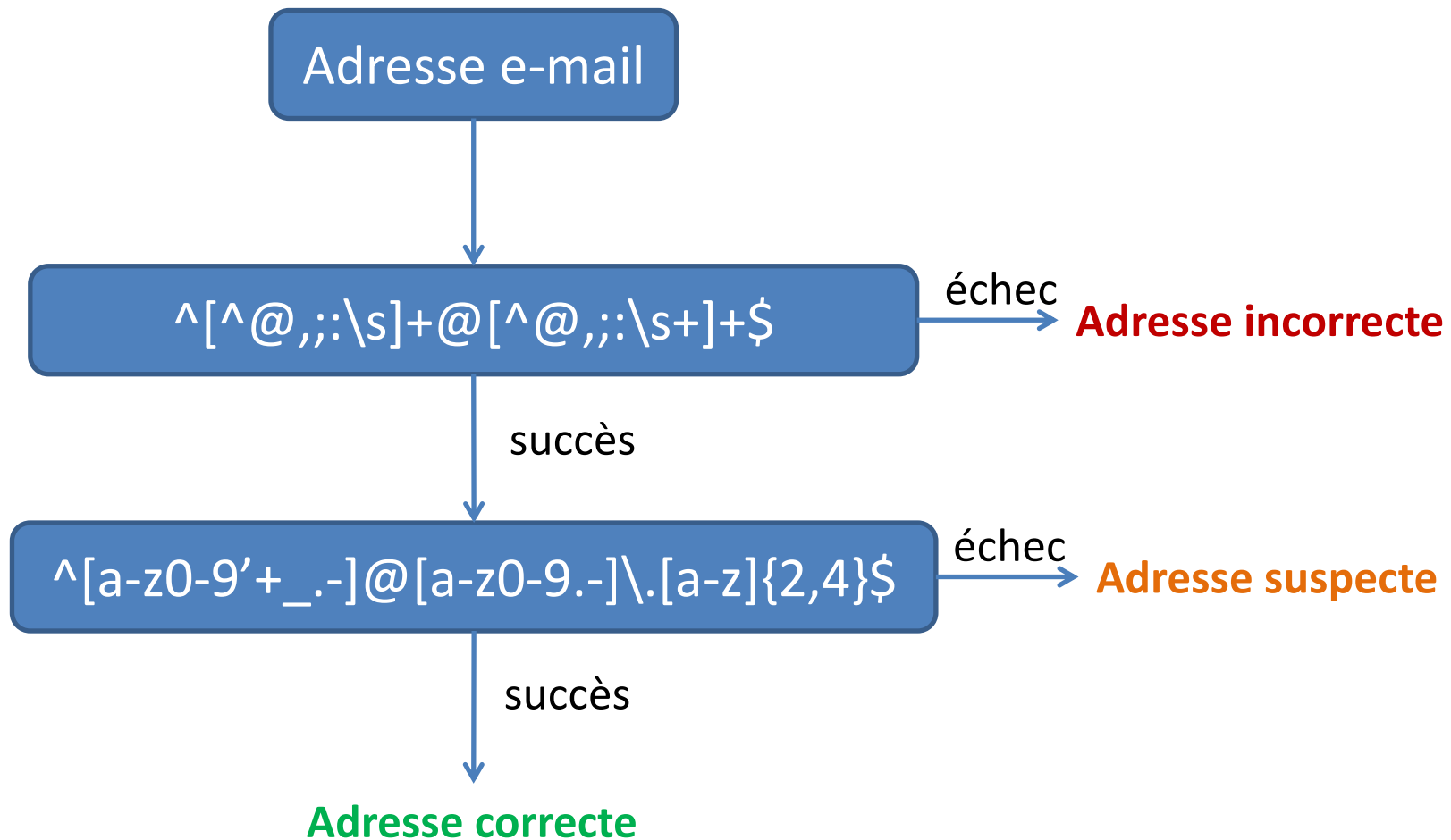
- Problème :
  - Soit trop contraignant
  - Soit trop laxiste
- Difficile de tout vérifier avec une expr. régulière
- Notre proposition : vérifier en **2 temps** :
  - Éliminer des adresses **certainement fausses** avec un test **laxiste**
  - Identifier des adresses **suspectes** avec un test très (trop) **contraignant**
    - On-line : demander confirmation
    - Batch : ajouter dans une liste « à contrôler »
  - Ce qui reste : **correct** (→ tests spécifiques)



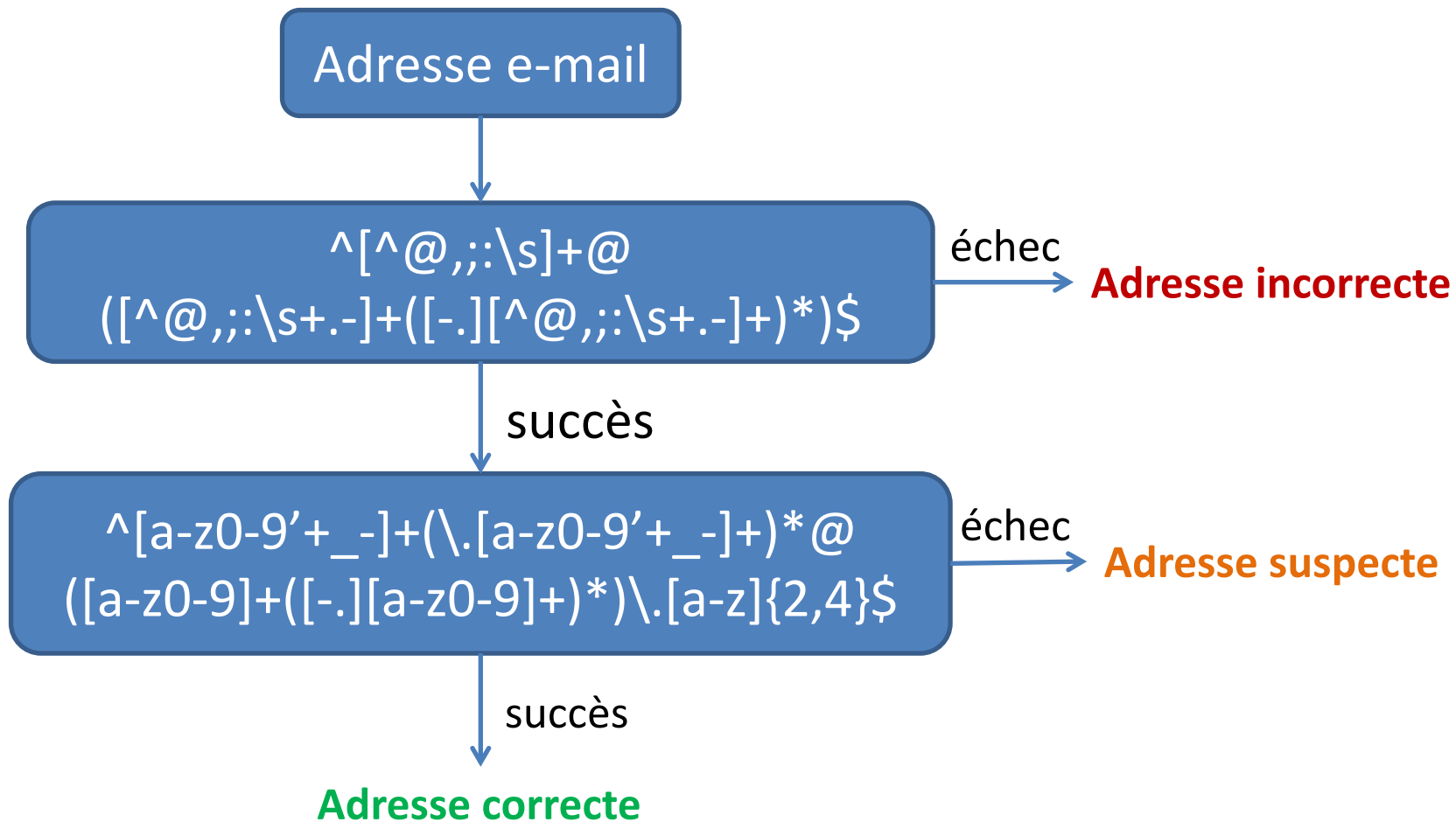
# Proposition : Trois catégories



# Proposition : Trois catégories

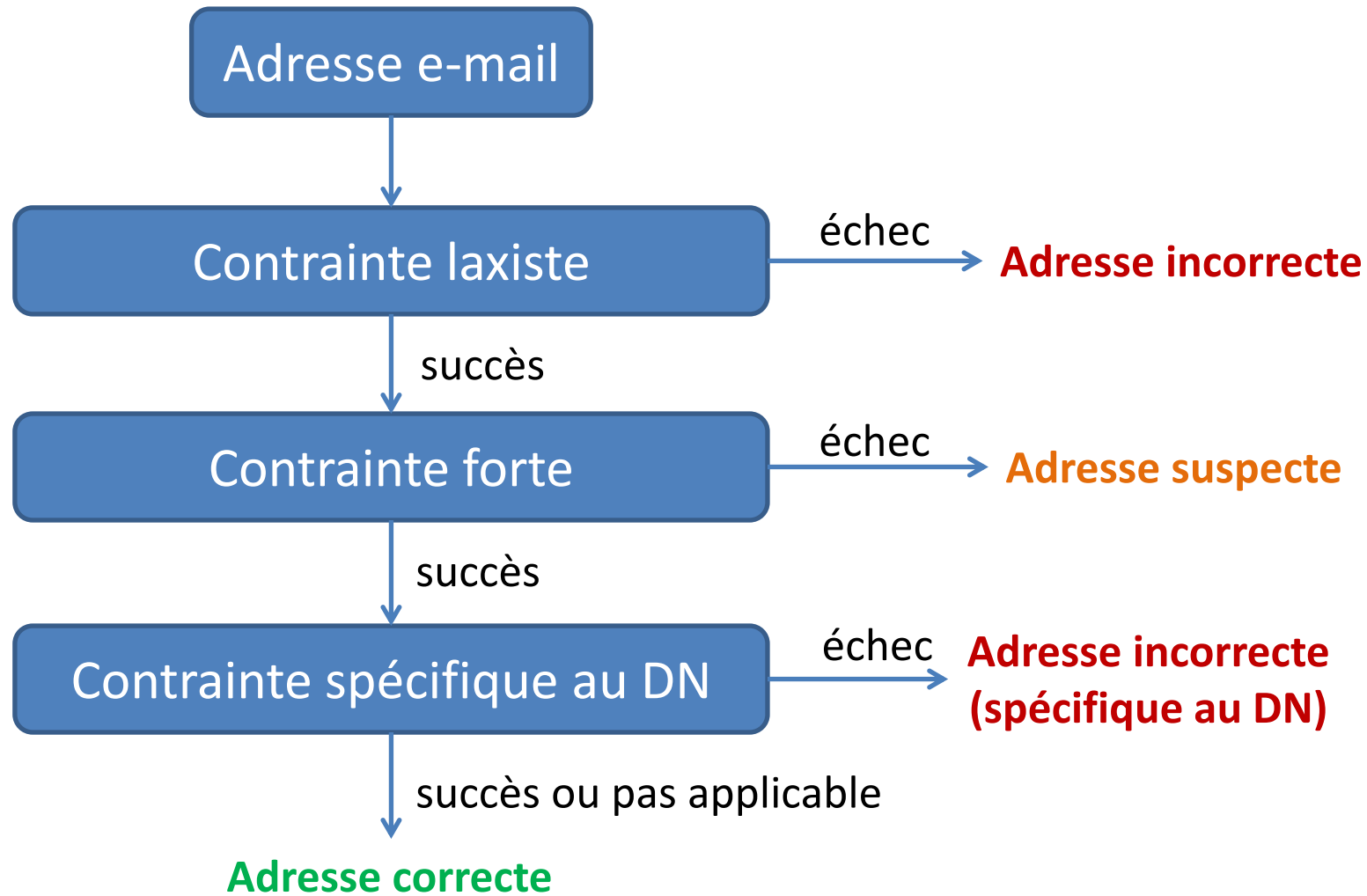


# Proposition : Trois catégories





# Proposition : Quatre catégories



# Suggestions de correction

On-line  
only

- Idée : si possible, proposer une correction
  - Avantage : souvent difficile de localiser une faute de frappe
  - Erreurs classiques :
    - albert.leroysmals.be
    - albert.leroy@|smals.be
    - albert.leroy#smals.be
    - albert,leroy@smals.be
    - albert.leroy @smals.be
    - albert.leroy@smals
    - albêrt.leroy@smals.be
    - albert-leroy@gmail.com
- albert.leroy@smals.be
- albert.leroy@gmail.com



# Outils

---

- Librairies : <http://isemail.info>
- Discussion : <http://www.regular-expressions.info>
- HTML 5 : champ « email »
- Librairies standards dans certains langages. Java : `org.apache.commons.validator.EmailValidator`
- Développement propres
- À notre connaissance :
  - Jamais de tests « spécifiques »
  - Toujours correct/incorrect, pas de cas suspects



# PoC

**Mail:**  ☐ Full check

**Firstname:**

**Lastname:**

## Verification and Validation

**Syntax check:** Incorrect syntax

## Matching

**Database matching:** No match  
**Name matching:** No name provided  
**Domain name matching:** No match

## Submission

Cannot submit!

## Suggestions

- alberleroy@smals.be

**Address:** alberléroy@smals.be  
**Firstname:**  
**Lastname:**

## Verification and Validation

**Syntax check:** Unusual format, please double check!  
 Suspicious characters : alberléroy@smals.be

**TLD (.be):** Passed  
**Domain (smals.be):** Passed

## Matching

**Database matching:**  
**Name matching:**  
**Domain name matching:**

## Submission

We have some doubt,

## Suggestions

- alberleroy@smals.be

**Mail:**  ☐ Full check

**Firstname:**

**Lastname:**

## Verification and Validation

**Syntax check:** Incorrect syntax  
 Forbidden characters : alberleroy@smals.be

## Matching

**Database matching:** No match  
**Name matching:**  
**Domain name matching:**

## Submission

Cannot submit!

## Suggestions

- alberleroy@smals.be

**Mail:**  ☐ Full check

**Firstname:**

**Lastname:**

## Verification and Validation

**Syntax check:** Incorrect syntax for domain gmail.com

## Matching

**Database matching:** No match  
**Name matching:** No name provided  
**Domain name matching:** Common domain name

## Submission

Cannot submit!

## Suggestions

- albert.leroy@gmail.com





**Mail:**

☐ Full check

**Firstname:**

Submit

**Lastname:**

Reset

# Syntaxe : l'essentiel

---

Plus  
complexe  
que ce que  
l'on pense !

Tests  
spécifiques au  
domaine  
⇒ ↗ précision

Tests binaires =  
trop laxistes, ou  
trop contraignants



Adresses  
suspectes ⇒  
limite les faux  
positif/négatif



# Questions ?

---



# Pause !

---







**Validation**

# Table des matières

---

---

Introduction

---

Syntaxe

---

**Validation**

Validation du nom de domaine

---

Validation d'adresse

---

Contrôle de lecture

---

Limites et difficultés

---

Outils

---

Matching

---

Bonnes pratiques  
& Conclusions



# Contexte

---

- Une adresse syntaxiquement correcte **n'implique pas qu'elle existe !**
- Une adresse qui existe **n'est pas toujours consultée**
- **Existence** d'une adresse :
  - Le nom de domaine
  - L'adresse elle-même
- Contrôle de **lecture** :
  - Insertion d'image
  - Lien unique



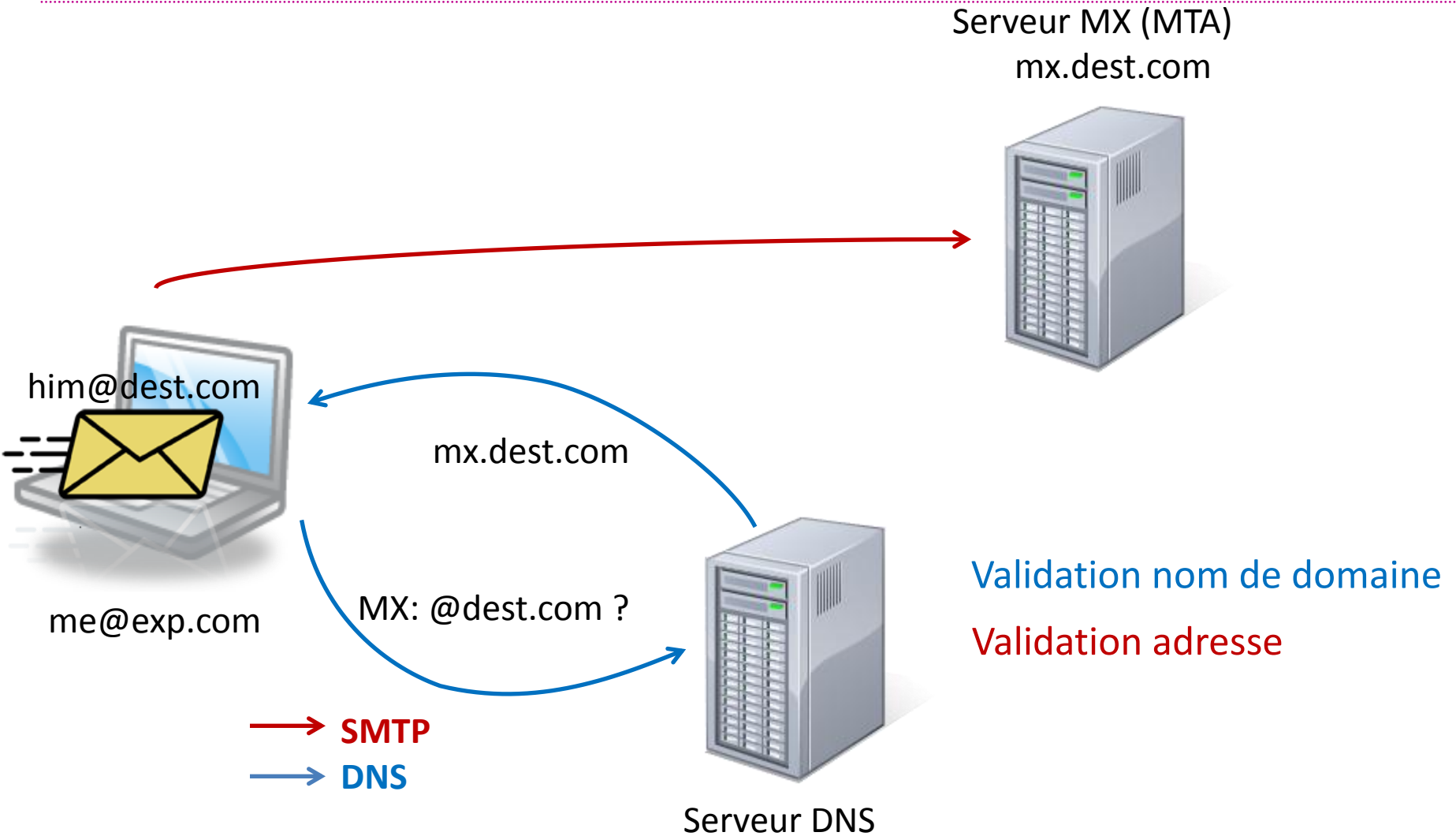
# Contexte : portail on-line

---

- Une nouvelle adresse devrait être **confirmée** (envoi d'un lien à cliquer)
- Beaucoup de portails ne le font pas → à **nettoyer** en batch
- Validation initiale : nécessaire, mais ne permet pas de **maintenir la qualité** dans le temps
- Forcer une revalidation fréquente peut être contraignant et intrusif
- On peut parfois automatiser cette validation



# Mécanisme de validation



# Validation : conversation SMTP

Batch  
only

```
C:\>telnet gmail-smtp-in.l.google.com. 25
Trying 173.194.78.26...
Connected to gmail-smtp-in.l.google.com. [...]
EHLO bxl.mapetitesociete.be
250-mx.google.com at your service, [91.xx.xx.xx][...]
MAIL FROM:<albert.leroy@bxl.mapetitesociete.be>
250 2.1.0 OK pn9si6796wjc.42 - gsmtip
RCPT TO:<leroy.mariecelestine@gmail.com>
550-5.1.1 The email account that you tried to reach
does not exist. [...]
RCPT TO:<mariecelestine.leroy@gmail.com>
250 2.1.5 OK pn9si6796wjc.42 - gsmtip
QUIT
221 2.0.0 closing connection pn9si6796wjc.42 - gsmtip
```



# Limites et difficultés (domaine)

---

- Validation nom de domaine : **très fiable**
- Si incorrect, deux cas :
  - Le nom de domaine n'existe pas
  - Il existe, mais pas d'e-mail associé (No MX record)
- Quelques cas rares de « time-out » → réessayer
- Faible risque de blacklisting en vérification batch



# Limites et difficultés (adresse)



Batch  
only

- Validation adresse : **beaucoup d'incertitudes !**
- Réponse difficile à interpréter

550 Service unavailable; Client host [xxx.xxx.xxx.xxx] blocked using dnsbl.njabl.org; spam source

550 MAILBOX NOT FOUND 550 <john@foo.bar>... User unknown

550 5.7.2 This smells like Spam

550 <john@foo.de>. Benutzer hat zuviele Mails auf dem Server.

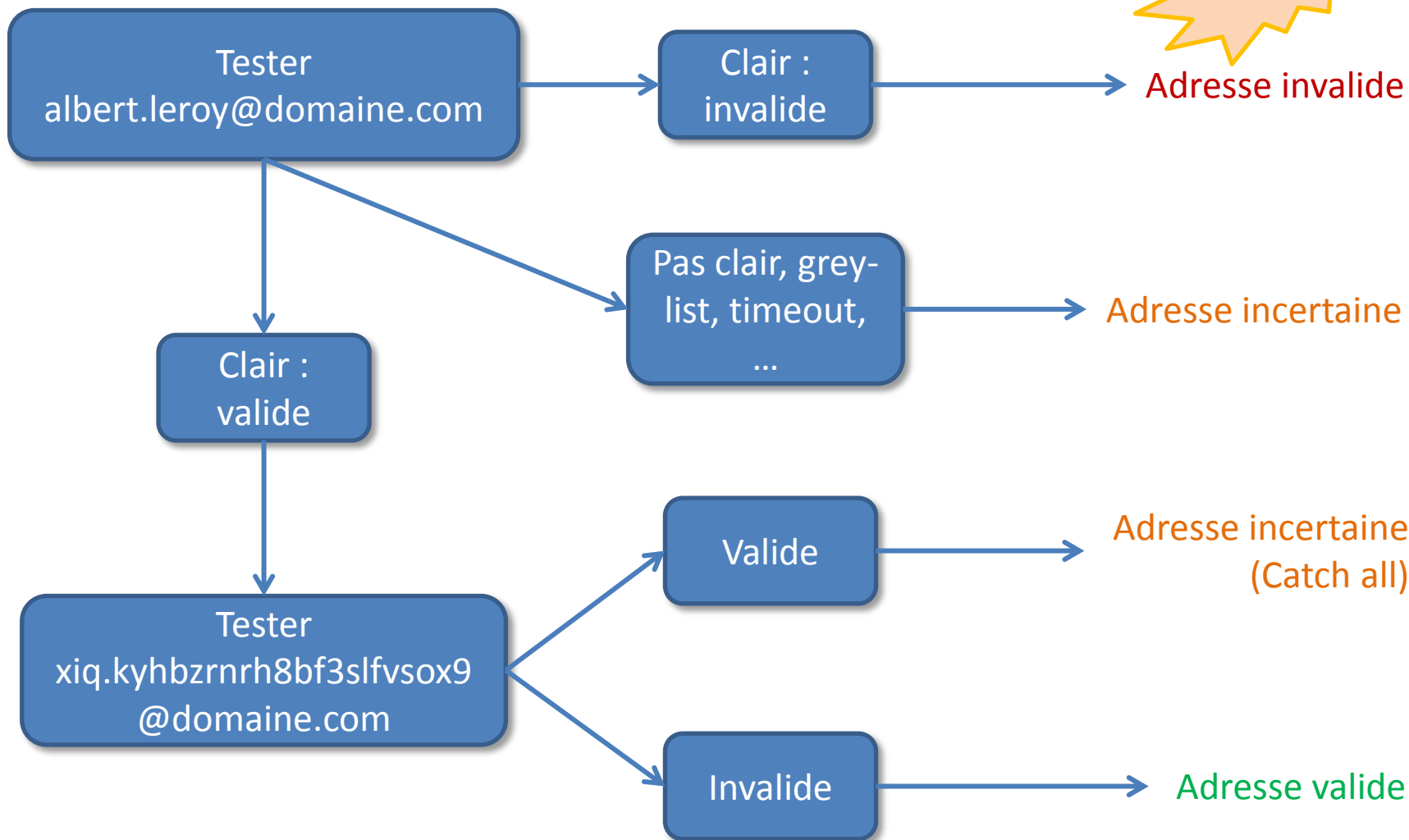
554 Delivery error Sorry your message to joe@foo.bar cannot be delivered. [#102]

- Si source pas clairement identifiée :
  - Greylisting (explicite)
  - Catch-all (implicite) } > 20-25% depuis un PC « standard »
- En cas de requête massive : risque de blacklist !
- Recyclage des adresses

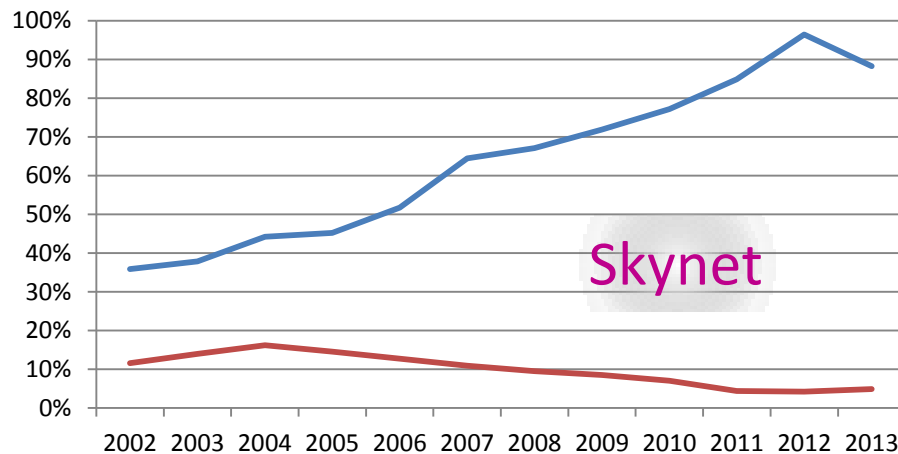
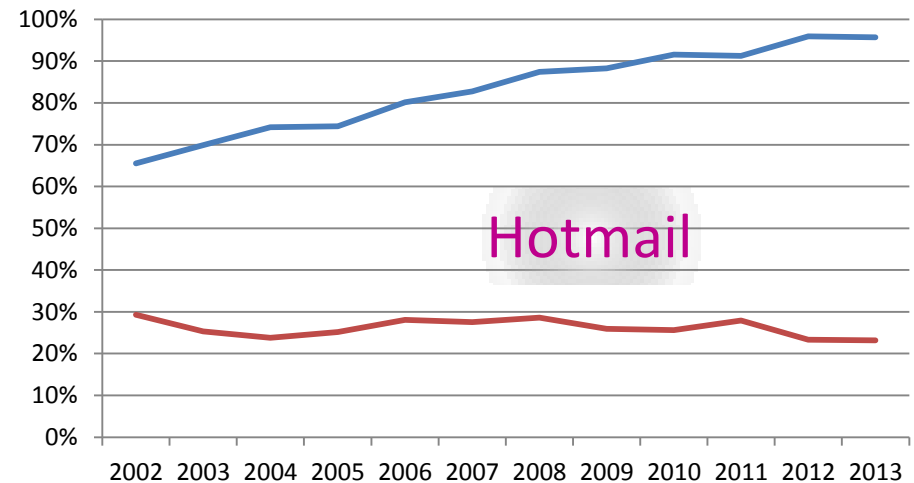
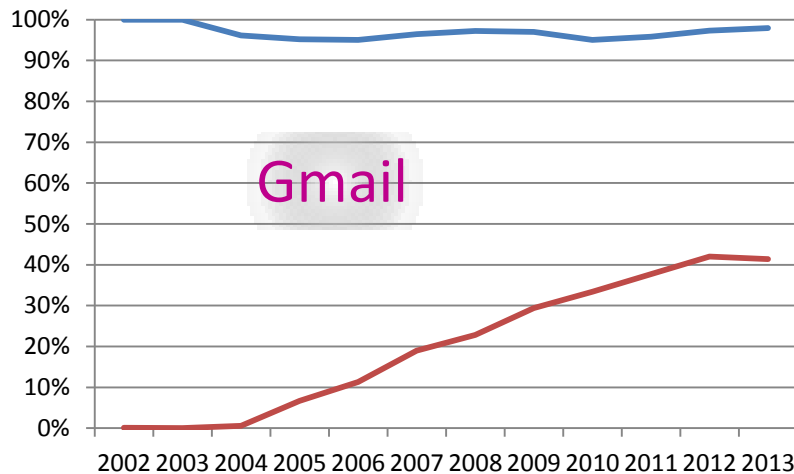




# Validation d'adresse



# Dégressivité par domaine



— Validité adresses — Proportion vs total



# Contrôle de lecture

---

- Quelques techniques pour s'assurer qu'une adresse est toujours active, mais **rien de très fiable**
- **Accusé de réception** (Outlook & co) : pas standard, pas inter-plateforme
- Présence de **lien « unique »** à cliquer
  - Il faut une raison de cliquer !
- Présence d'une **image « unique »**
  - Il faut accepter d'afficher les images



# Lien unique

```
<a href="http://mysite.com/redir?
m=albert.leroy@smals.be&a=www.smals.be">
www.smals.be
```

To:albert.leroy@smals.be

[www.mysite.com/redir](http://www.mysite.com/redir)

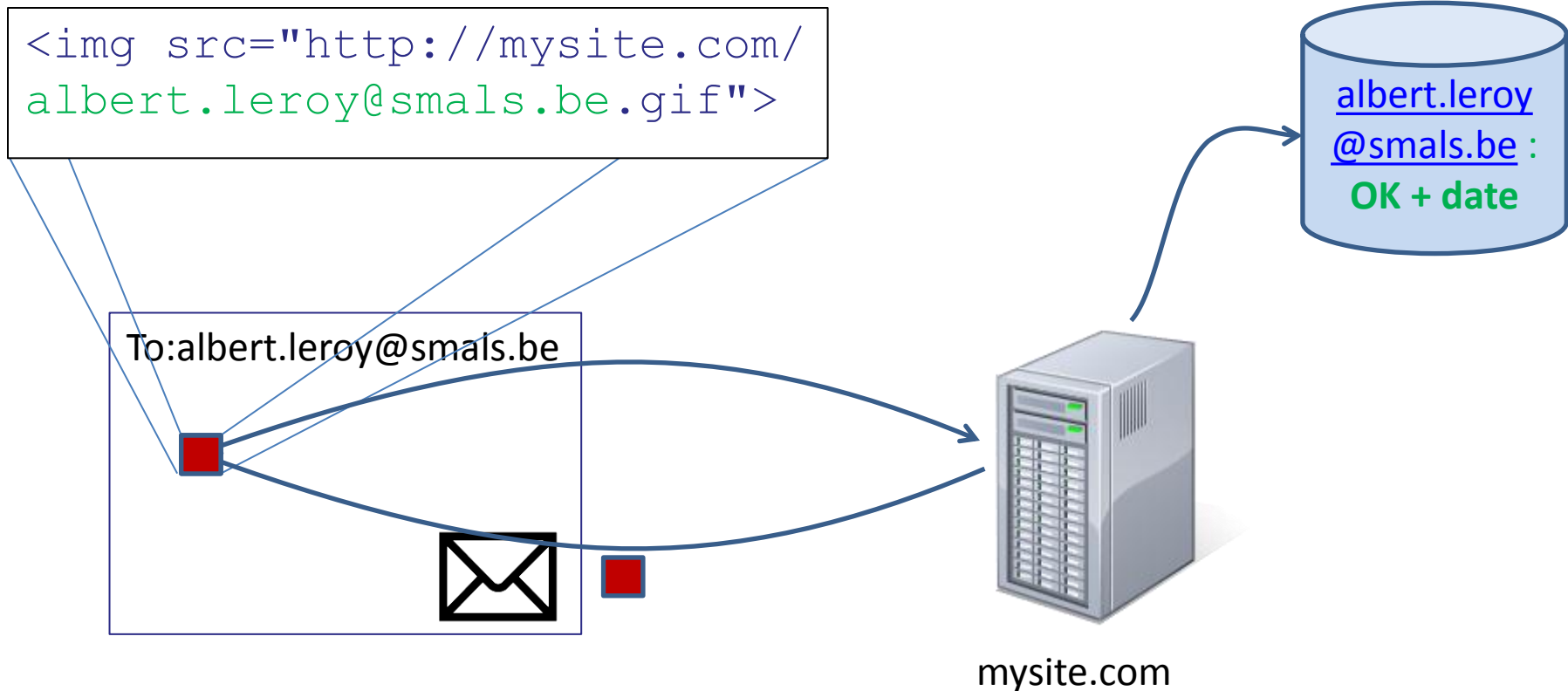
[www.smals.be](http://www.smals.be)

[albert.leroy@smals.be](mailto:albert.leroy@smals.be) :  
OK + date

- Variantes : crypter/masquer le contenu, passer via une base de données
- Il faut une bonne raison de cliquer !
- Indispensable à l'enregistrement (e-mail de confirmation)



# Image unique



Remarque :

- Souvent un pixel blanc (invisible)
- Il faut accepter les images !



# Image unique

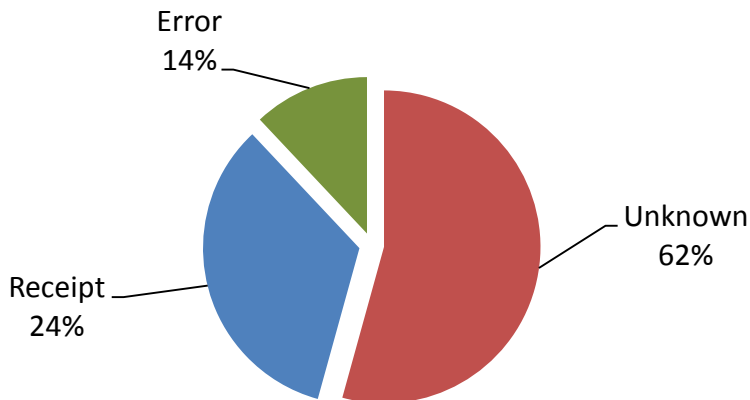
- Inconvénient : beaucoup de systèmes désactivent les images distantes par défaut

Security:

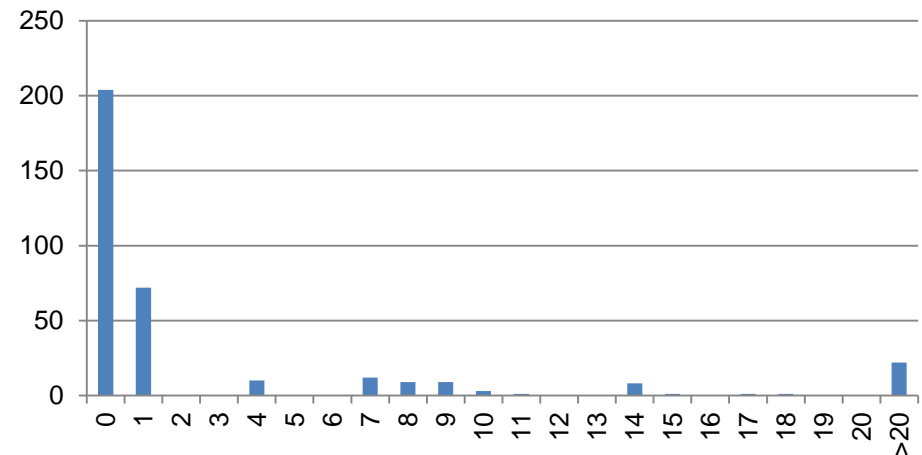
To ensure privacy, images from remote sites were prevented from downloading. [Show Images](#)



Images are not displayed. [Display images below](#)

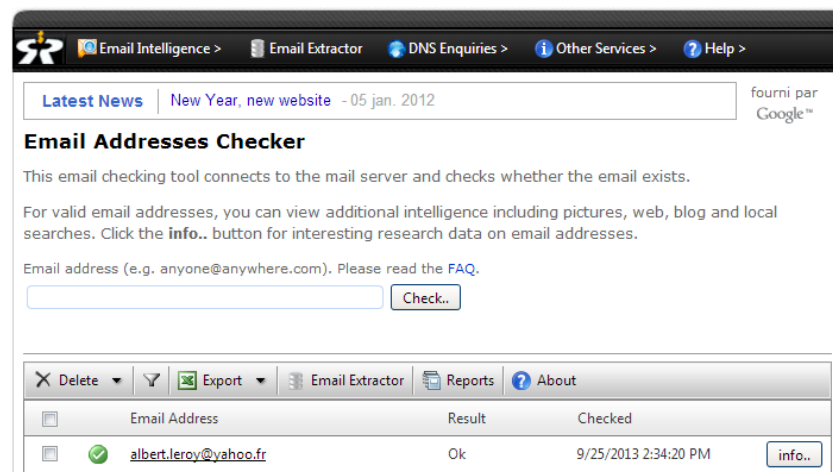
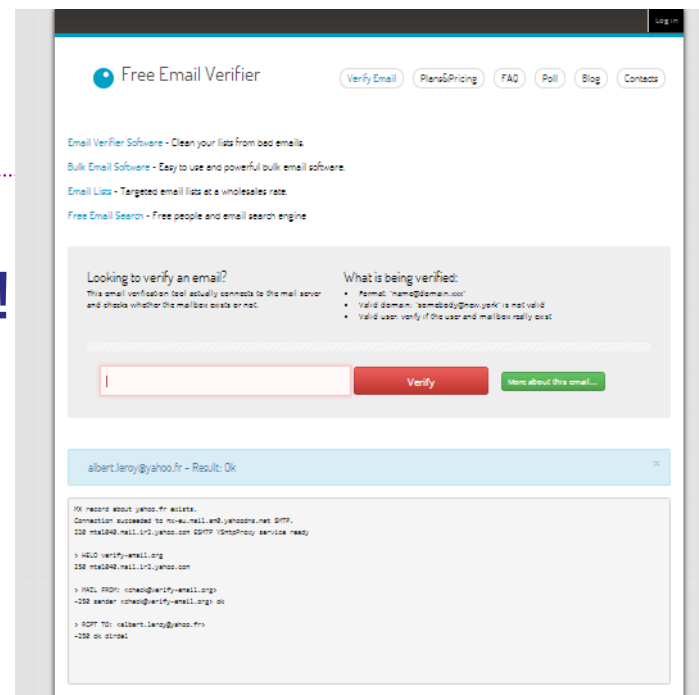


**Délai lecture**



# Outils (Existence)

- Peu d'outils du marché fiables !
- Les plus fiables :
  - <http://verify-email.org>,
  - <http://tools.email-checker.com>
  - Limités à quelques requêtes/24h
  - Version professionnelle payante
- Quelques logiciels, mais tournent en « local »
- Développement propre plus performant



# Outils : ServiceObjects



- Web service avancé de vérification et validation
- Vérification syntaxique spécifique
- Correction d'erreurs
- Identification des serveurs « catch-all »
- Réponse avec degré de certitude
- Outil similaire moins avancé : StrikeIron





# Outils (Contrôle de lecture)

- [www.bananatag.com](http://www.bananatag.com) : intégré à Gmail ou Outlook. Ajoute une image + adapte les liens
- [www.spypig.com](http://www.spypig.com), TailMail : génère une image à intégrer manuellement
- [www.MsgTag.com](http://www.MsgTag.com) : « proxy SMTP » pour client mail. Ajoute une image



# PoC

**Address:** albert.leroy@gmail.com  
**Firstname:**  
**Lastname:**

## Verification and Validation

**Syntax check:** Passed  
**TLD (.com):** Passed  
**Domain (gmail.com):** Passed

## Matching

**Database matching:** No match  
**Name matching:** No name provided  
**Domain name matching:** Common domain name

## Submission

**Address:** albert.leroy@smals.be  
**Firstname:**  
**Lastname:**

## Verification and Validation

**Syntax check:** Passed  
**TLD (.be):** Passed  
**Domain (smals.be):** Passed  
**Address check:** This mail server is known to ignore all requests (catch-all server) ...  
 Cannot tell whether this address exists or not!

**Matching**

**Mail:** albert.leroy@skynet.be ☐ Full check  
**Firstname:**    
**Lastname:**

**Submission**

## Verification and Validation

**Syntax check:** Passed  
**TLD (.be):** Passed  
**Domain (skynet.be):** Address check:  
 Address check:

## Matching

**Database matching:** No match  
**Name matching:** No name provided  
**Domain name matching:** Found match: smals.be ⇔ smals.be

## Submission

Cannot submit!

## Suggestions

- albert.leroy@smals.be





**Mail:**

☐ Full check

**Firstname:**

Submit

**Lastname:**

Reset





POC Email Addresses Reli x

vandyssh.smalsrech.be/lo x

Inbox - vandy.berten@gn x



vandyssh.smalsrech.be/mail



**Mail:**

☐ Full check

**Firstname:**

Submit

**Lastname:**

Reset

# Validation : l'essentiel

Existence du  
domaine :  
très fiable

Existence de l'adresse :

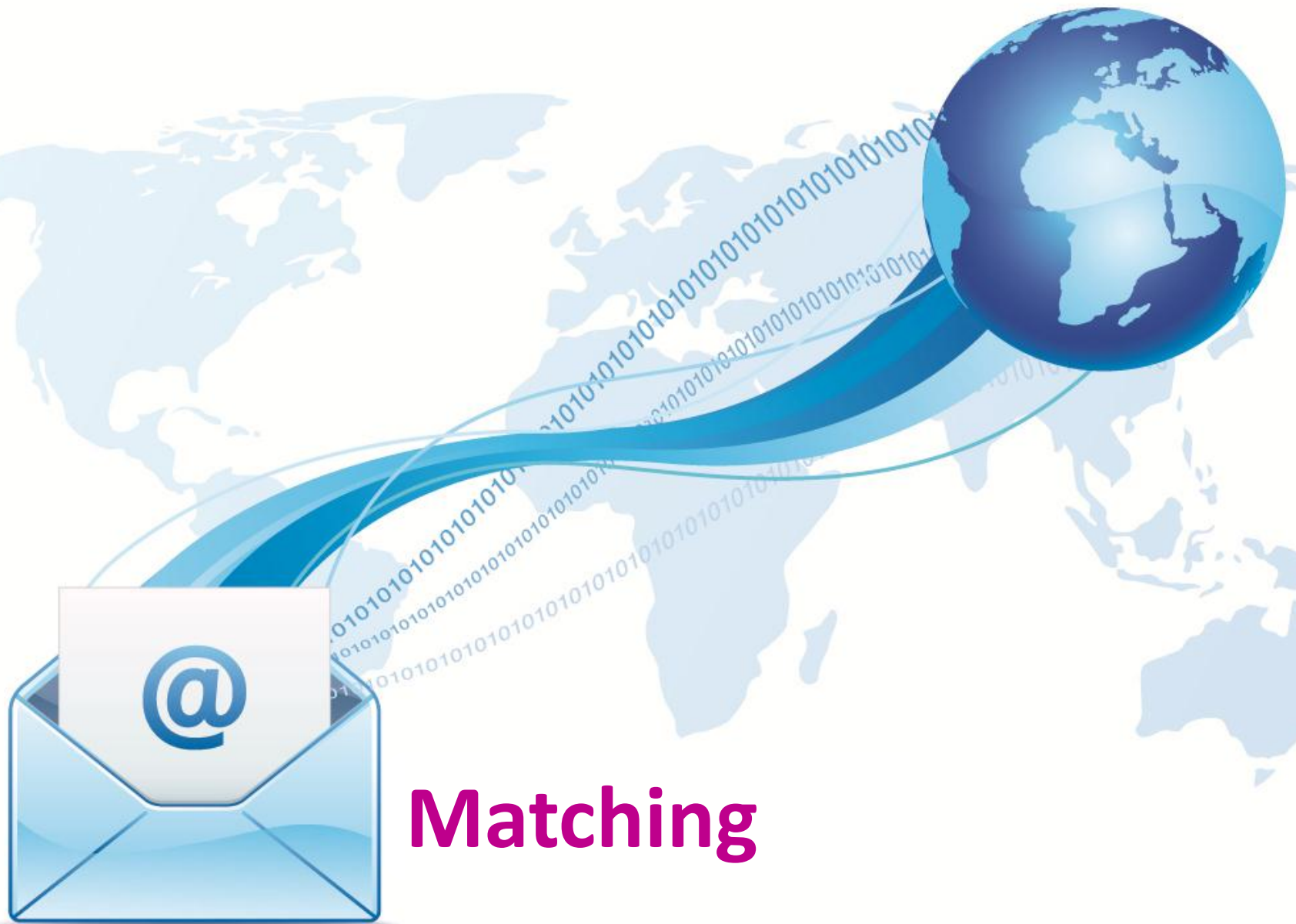
- Fausse/Correcte :  $\pm$  fiable
- Beaucoup d'inconnu  
(catch-all, greylist, ...)

Contrôle de lecture :

- Lu : très fiable
- Pas d'accusé = pas  
d'information

Seule validation  
fiable : envoi  
d'un e-mail avec  
action obligatoire





**Matching**

# Table des matières

---

---

Introduction

---

Syntaxe

---

Validation

---

**Matching**

Contexte

---

Matching interne (informations annexes)

---

Matching sur nom de domaine

---

Dédoublonnage

---

Suggestions

---

Difficultés

---

Outils

---

Bonnes pratiques &  
Conclusions



# Contexte

---

- Matching : classique en Data Quality
- But : comparer des infos identiques ou similaires
- Différentes utilités :
  - Matching interne : dans un « record », utiliser la redondance pour croiser des info → **suspicion d'erreurs** (nom/prénom/e-mail)
  - Dédoublonnage : « records » similaires → **suspicion de doublons**
  - Domaine connu : similitude avec des domaines fréquents → **suspicion d'erreurs** (nom de domaine)
- Ne permet pas de détecter des erreurs, mais de les **soupçonner**





# Contexte

---

- Basé sur des algorithmes de **similitude** (Métaphone, Soundex, Jaro, Levenshtein,...)
- On pourra **suggérer** des corrections :
  - En on-line : à l'utilisateur à l'encodage
  - En batch : aux gestionnaires
  - Décision automatique difficile ou risquée
- Dans certaines DB observées : présence du nom/prénom dans **85%** des adresses !



# Matching interne : info annexes

| Nom   | Prénom | E-mail                |
|-------|--------|-----------------------|
| Leroy | Albert | albert.leroy@smals.be |

| Nom   | Prénom | E-mail                |
|-------|--------|-----------------------|
| Leroy | Albert | ablert.leroy@smals.be |

| Nom   | Prénom | E-mail                |
|-------|--------|-----------------------|
| Leroy | Ablert | albert.leroy@smals.be |

| Nom de société     | E-mail                 |
|--------------------|------------------------|
| Les meilleurs asbl | lesmeilleurs@gmail.com |

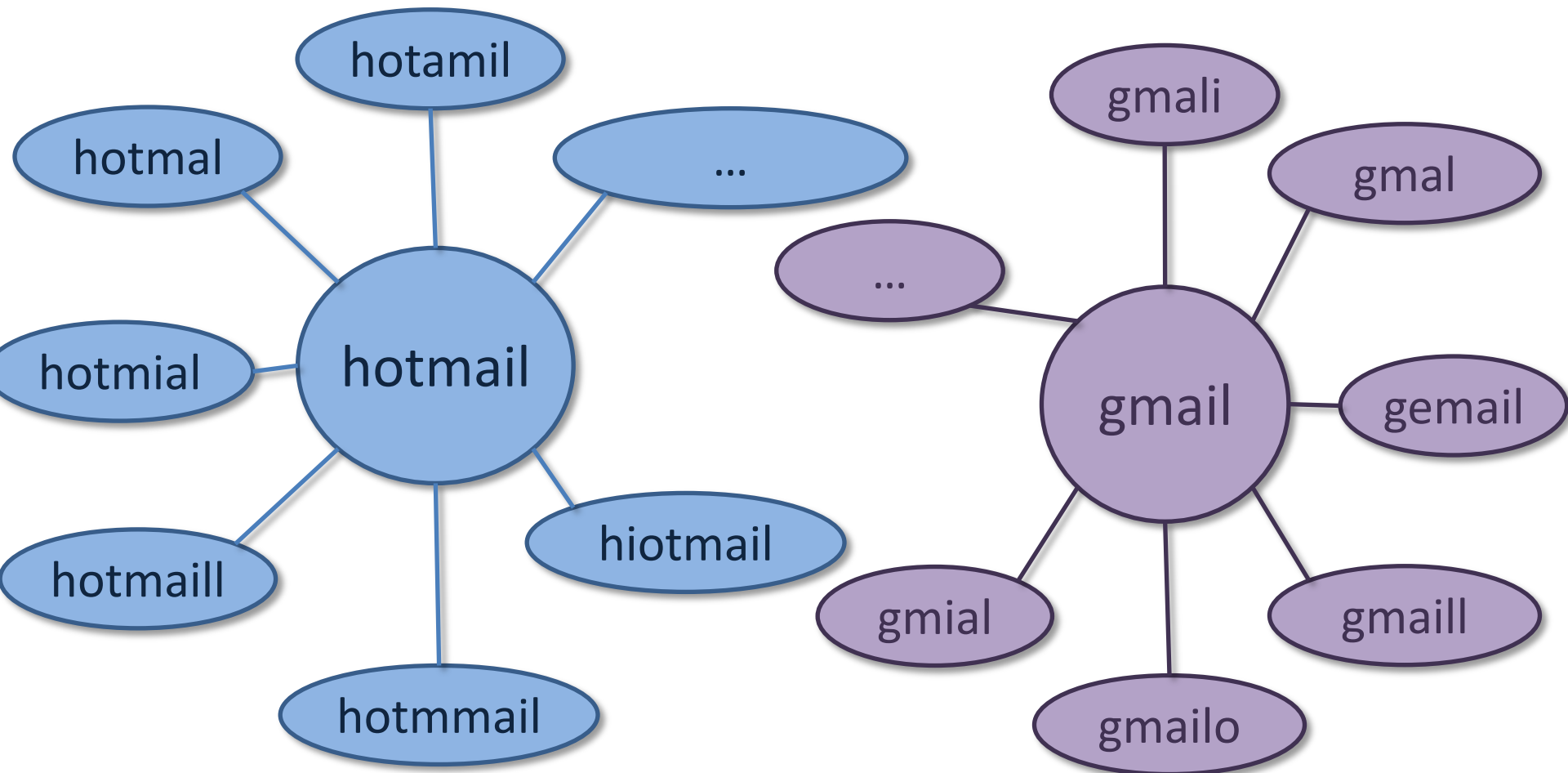


# Exemple réel (anonymisé)

| Nom                    | Prénom             | E-mail                                    |
|------------------------|--------------------|-------------------------------------------|
| El <b>l</b> arkani     | Ahmine             | ahmine.el <b>l</b> arkani@hotmail.com     |
| De <b>t</b> immerman   | Alain              | alain.de <b>t</b> imemrman@gmail.com      |
| De <b>s</b> aux        | Adèle              | adeledel <b>v</b> aux@swing.be            |
| Vanden <b>l</b> berghe | Arnaud             | arnaud.vanden <b>l</b> eghe@skynet.be     |
| Des <b>l</b> medt      | Geert              | geertdes <b>l</b> met@hotmail.com         |
| Piotrowski             | Alexander <b>l</b> | alexander <b>l</b> .piotrowski@telenet.be |
| Lanoy                  | Caroline <b>l</b>  | carol <b>l</b> anoy@hotmail.fr            |
| Michel                 | Charles-Édouard    | charlesedou.michel@yahoo.fr               |
| Hammoud                | Haffsa             | haf <b>l</b> ssa.hammoud@gmail.com        |
| Mabrouk                | Tarik              | mabrouk.tare <b>l</b> k@gmail.com         |
| Wouters                | Idalie             | wouterso <b>l</b> dalie@yahoo.fr          |

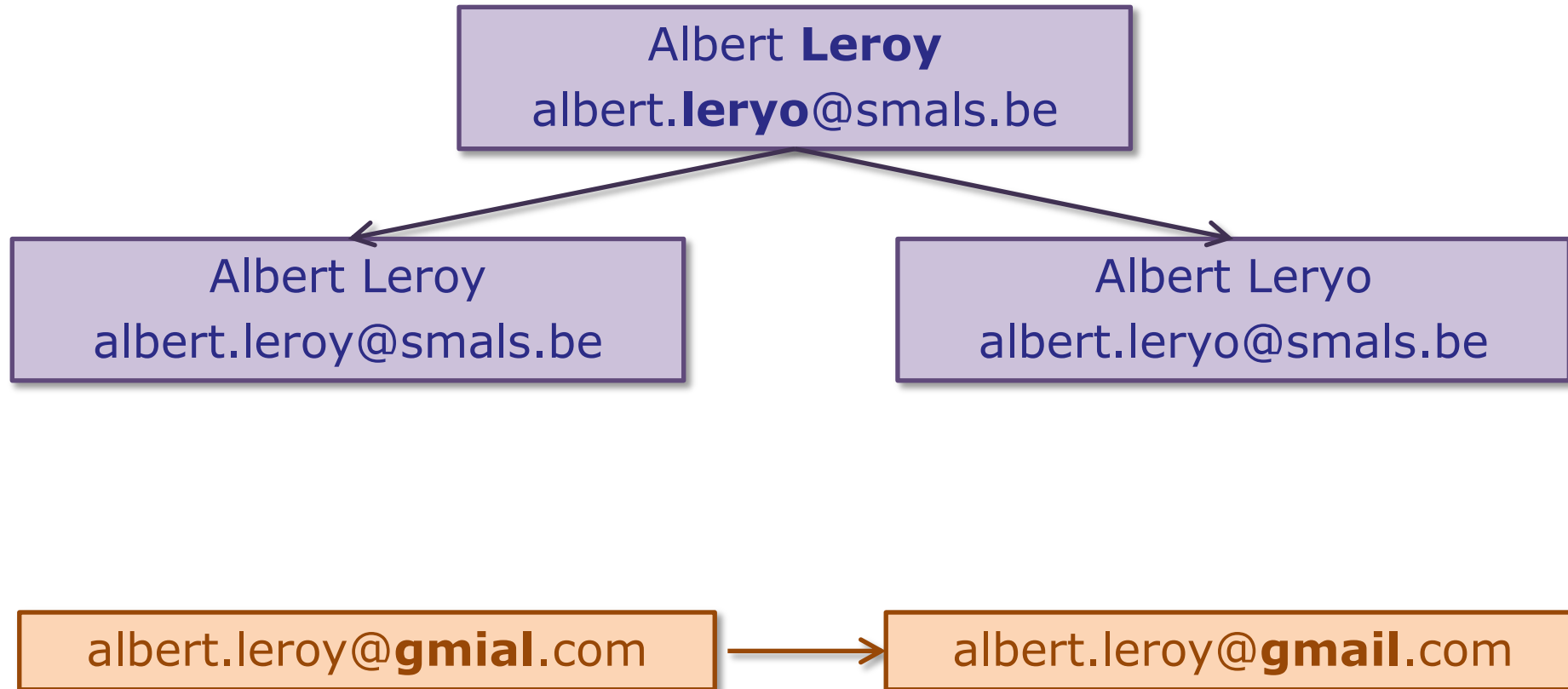


# Matching interne : domaines connus



# Suggestion de correction

---



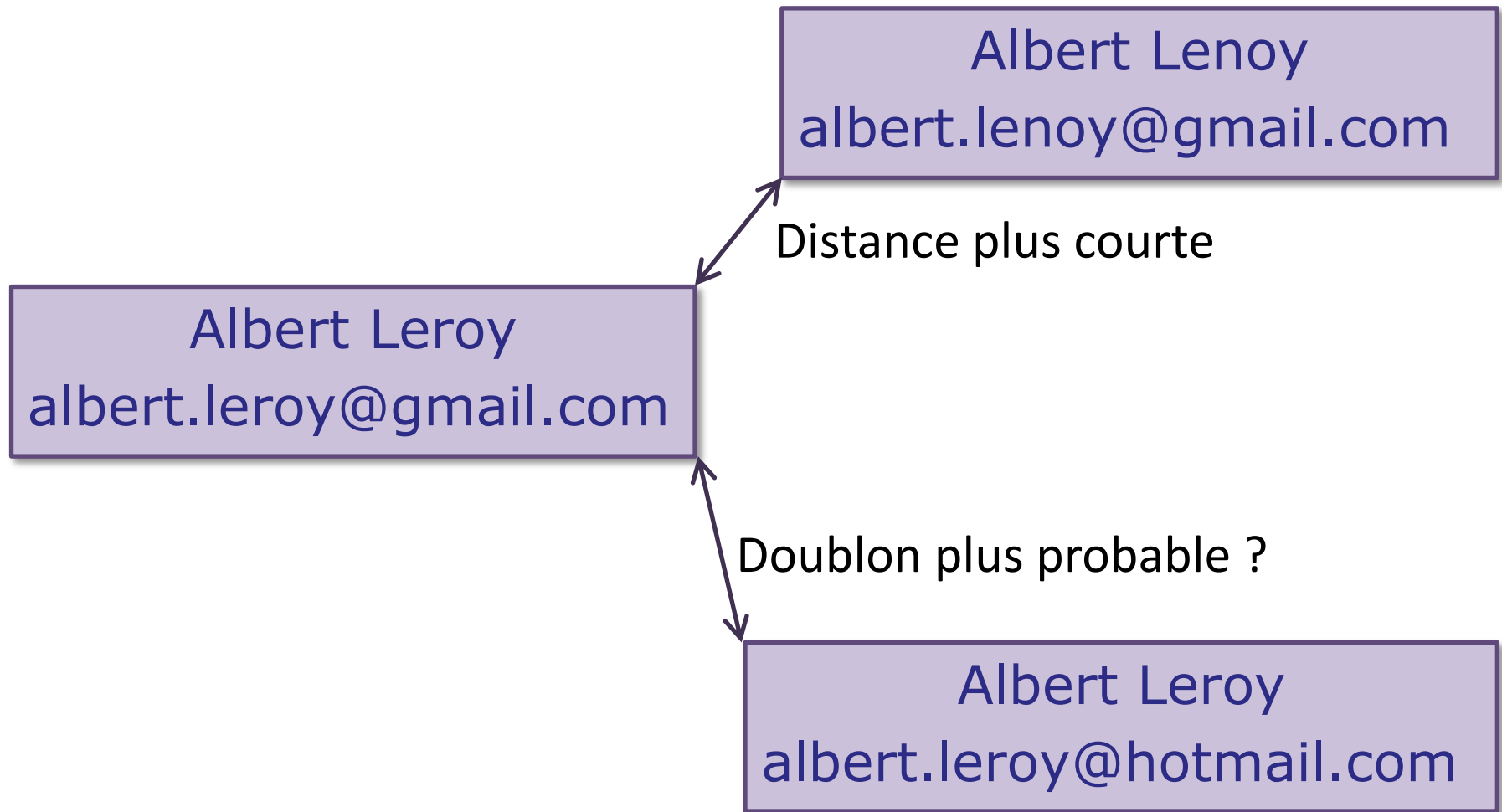
# Dédoublonnage

---

- Les **techniques classiques** de dédoublonnage peuvent être utilisées pour détecter des **doublons** dans une DB
- Il faut se servir de **différentes informations** pour établir un double
- En général, on établit un « **classement** » : double très probable (beaucoup d'information similaire) → double moins probable
- Parfois **beaucoup plus que deux** !



# Dédoublonnage



# Difficultés

---

- Difficile à paramétrer !
- On a vite beaucoup de faux positifs
- Faux positifs intentionnels :
  - Translittération : Tarik vs Tarek
  - Traduction : Alexander vs Alexandre
  - Diminutif/surnom : Jacques vs Jacky





# Outils

---

- Le matching et dédoublonnage sont des tâches **complexes**, nécessitant beaucoup de « **tuning** »
  - Seuls des **outils spécialisés de Data Quality** y arrivent correctement
  - Gratuit : OpenRefine
  - Propriétaire :
    - IntoDQ (Trillium)
    - RedPoint
    - HumanInference
- Pas de modules spécifiques aux e-mails mais largement paramétrables



# PoC

**Address:** albert.leroy@smals.be

**Firstname:** Albert

**Lastname:** Laroy

## Verification and Validation

**Syntax check:** Passed

**TLD (.be):** Passed

**Domain (smals.be):** Passed

## Matching

**Database matching:** No match

**Name matching:** Mail contains firstname  
Suspecting typo: leroy ⇒ laroy: l[e/a]roy

**Domain name matching:** Common domain name

## Submission

We have some doubt, please double check!

## Suggestions

- albert.laroy@smals.be (Albert Laroy)
- albert.leroy@smals.be (Albert Leroy)

**Address:** albert.leroy@smalz.be

**Firstname:**

**Lastname:**

## Verification and Validation

**Syntax check:** Passed

**TLD (.be):** Passed

**Domain (smalz.be):** Passed

## Matching

**Database matching:** No match

**Name matching:** No name provided

**Domain name matching:** Found match: smalz.be ⇔ smals.be

## Submission

We have some doubt, please double check!

## Suggestions

- albert.leroy@smals.be





**Mail:**

☐ Full check

**Firstname:**

Submit

**Lastname:**

Reset

Dédoublonnage

# Matching : l'essentiel

But :  
suspecter/identifier  
des cas douteux

Purement empirique !  
⇒ Traitement humain  
souvent nécessaire  
(facilité par les suspicions)

Matching interne :  
erreur dans le  
nom/prénom/e-mail

Domaine connu :  
erreur dans le nom  
de domaine

Dédoublonnage :  
données similaires





**Bonnes pratiques & Conclusions**

# Organisation

---

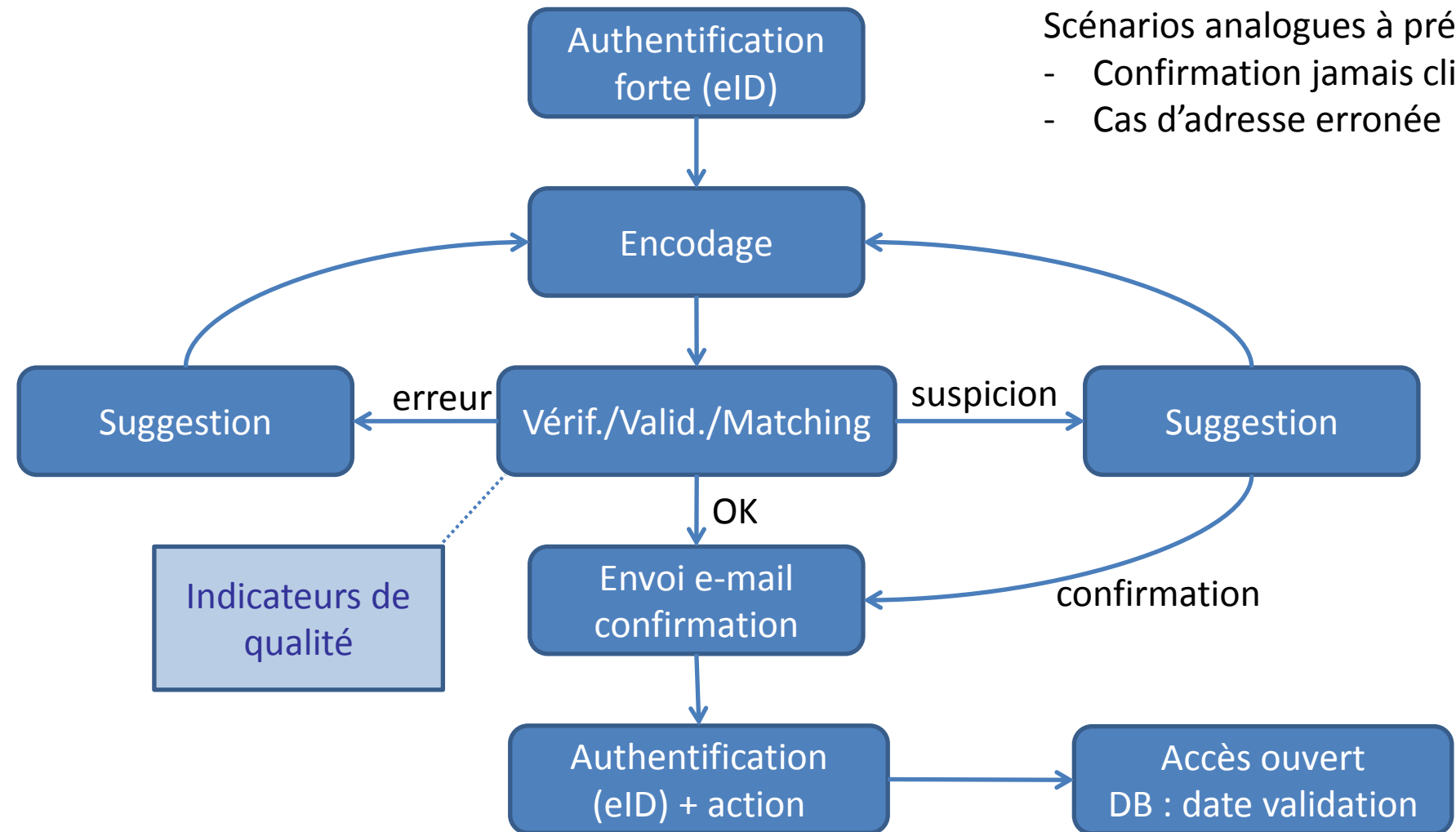
- Organisation rigoureuse et contrôlée du système d'information :
  - Définition de **services et sources authentiques validées**, en fonction des enjeux et usages
  - Flux, processus et intervenants humains
- **Arbitrage** en fonction des enjeux
  - Coûts *versus* moyens disponibles
  - Validité *versus* rapidité de traitement
  - Fréquence rappels *versus* caractère intrusif auprès du public
- Point fondamental : prise en compte de la **validité dégressive** dans le temps des adresses e-mail



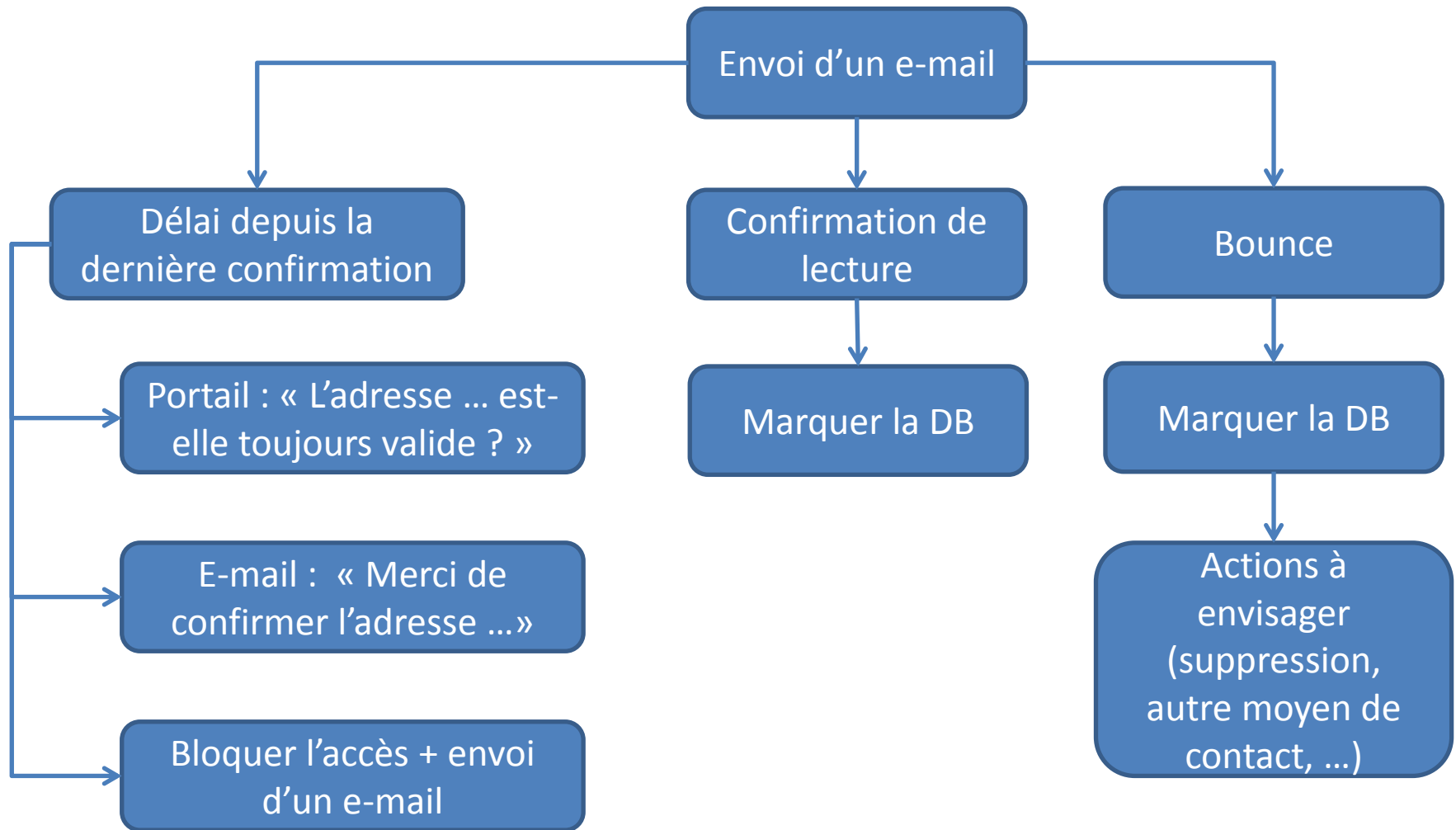
# Bonnes pratiques : encodage/update

Scénarios analogues à prévoir :

- Confirmation jamais cliquée
- Cas d'adresse erronée



# Bonnes pratiques : envoi d'un e-mail





# Stratégies de bonne gestion

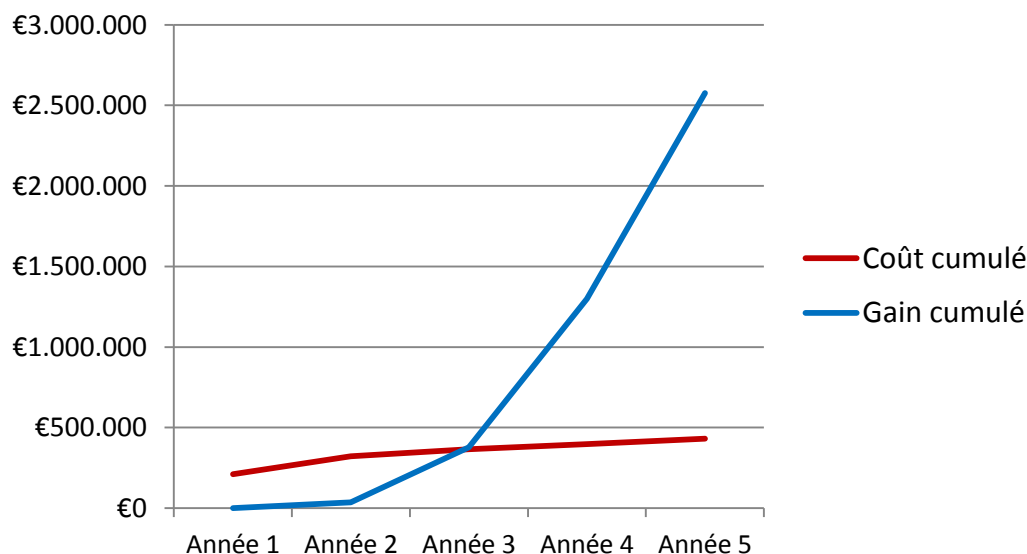
---

- Catégorisation pour tenir compte de l'incertitude :
  - Correct – Incorrect – Incertain
- Minimiser l'intervention manuelle :
  - Exemples : suggestions de corrections semi-automatiques
- Historique des événements et indicateurs de qualité
  - Datés (timestamp)
  - Associés à chaque record de la DB
  - Quantification continue des events et indicateurs



# Suivi de la validité dans le temps

- **Monitoring** de la base de données et stratégies de gestion en fonction des enjeux :
  - Exemples : actions en vue d'une meilleure visibilité, suivi des améliorations liées à une action, ...
- **Gains importants** si l'on adopte une stratégie proactive et continue



# Conclusions

Peu de  
déterminisme,  
beaucoup  
d'incertitude

Souvent,  
gains  
importants

Difficulté :  
dégressivité  
dans le temps

Syntaxe  
spécifique :  
améliore la  
qualité

Complexité  
de la gestion

Catégorisation  
binaire : laxiste  
ou restrictive  
→ Suspensions

Importance d'un suivi continu  
et d'une bonne organisation

Pro-action !



Vandy Berten  
02/787.57.32  
[vandy.berten@smals.be](mailto:vandy.berten@smals.be)



Isabelle Boydens  
02/787.59.92  
[isabelle.boydens@smals.be](mailto:isabelle.boydens@smals.be)



More on Smals Research  
Website Smals : [www.smals.be](http://www.smals.be)  
Blog : [blogresearch.smalsrech.be](http://blogresearch.smalsrech.be)  
Twitter : [twitter.com/smalsresearch](https://twitter.com/smalsresearch)



# Annexes



# Expressions régulières spécifiques



- Hotmail & Co (partie username) :



$^[a-z0-9_-](\.[a-z0-9_-]+)^*\$$

- Yahoo :

YAHOO!

$^[a-z][a-z0-9_]*([\.][a-z0-9_]{3,})?[a-z0-9]\$$

$^[a-z0-9_.]{4,32}\$$

En une seule expression ?



# Expressions régulières spécifiques

---

- Gmail :

$^[a-z0-9.+]^+ \$$



- Contraintes supplémentaires :
  - entre 6 et 30 caractères, sans compter les points et ce qui suit le « + ». Expression régulière ???
  - Si + de 8 caractères : au moins une lettre
- Alternative plus compacte mais incomplète :

$^[a-z0-9.+] \{6, \} \$$



# Références

---

- I. Boydens, «*Strategic Issues Relating to Data Quality for E-government: Learning from an Approach Adopted in Belgium*» In « Practical Studies in E-Government: Best Practices from Around the World », New York, Springer, pp. 113-130 (chapitre 7), 2011.
- Y. Bontemps, I. Boydens et D. Van Dromme, «*Data Quality: tools*», Bruxelles, Smals, 2007.
- I. Boydens, «*Informatique, normes et temps*», Bruxelles, Bruylant, 1999.
- I. Boydens, A. Hulstaert et D. Van Dromme, «*Gestion intégrée des anomalies*», Bruxelles, Smals, 2011.





# Glossaire

---

- DNS : Domain Name System
- gTLD : generic Top Level Domain
- IDN : Internationalized Domain Name
- MTA : Mail Transfer Agent
- MX : Mail eXchange
- SMTP : Simple Mail Transfer Protocol
- TLD : Top Level Domain

