

Le point sur la messagerie instantanée Jabber/XMPP

Mickaël Rémond
XMPP Standards Foundation Board Member
<mickael.remond@process-one.net>



Plan

- **Chat et Messagerie Instantanée, une longue histoire**
- **Jabber et XMPP**
- **XMPP: Applications concrètes et déploiements**
- **L'avenir**

Chat et Messagerie instantanée: Une longue histoire



Historique: Messagerie Instantanée



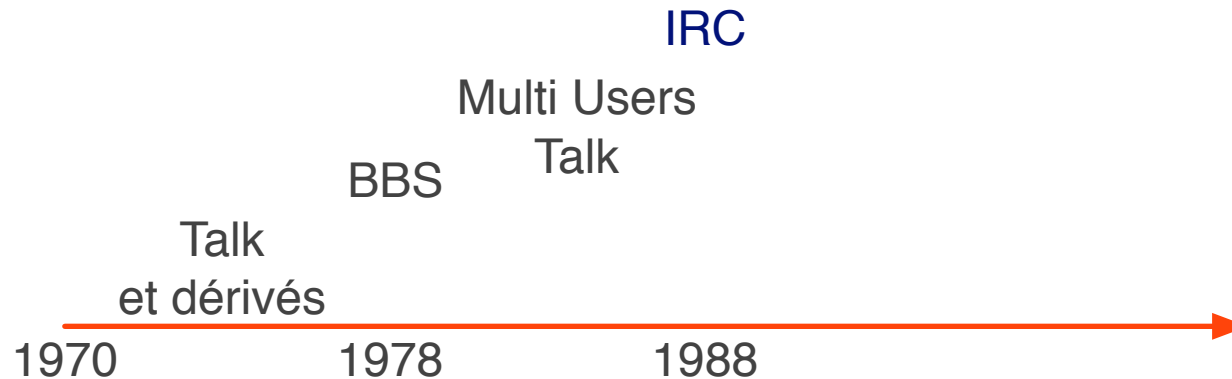
- **Communication entre utilisateurs sur le même serveur**
- **Limité à deux personnes**
- **Orienté vers la discussion entre utilisateurs d'une même machine**

Historique: Messagerie Instantanée



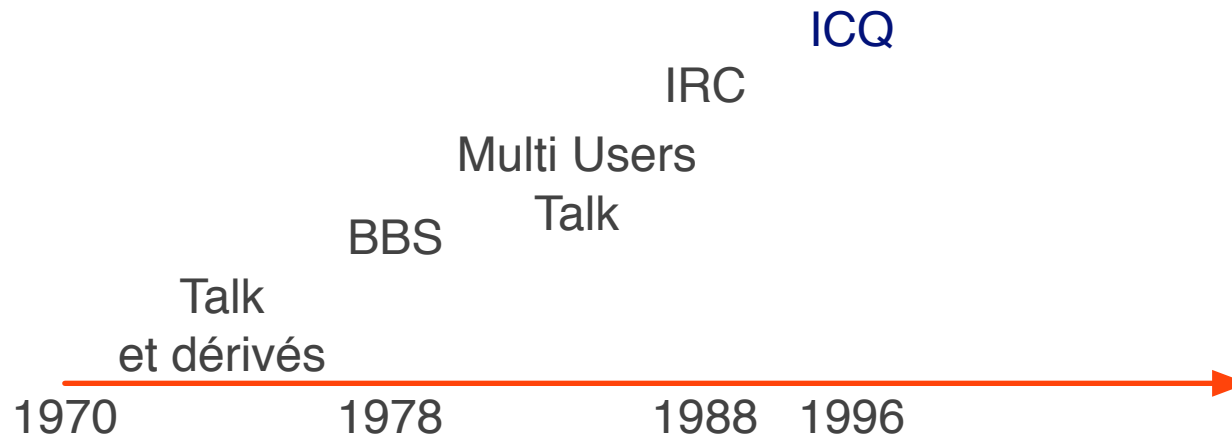
- **Bulletin Board Systems**
- **Ancêtre des forums**
- **Introduction des discussions multi utilisateurs**

Historique: Messagerie Instantanée



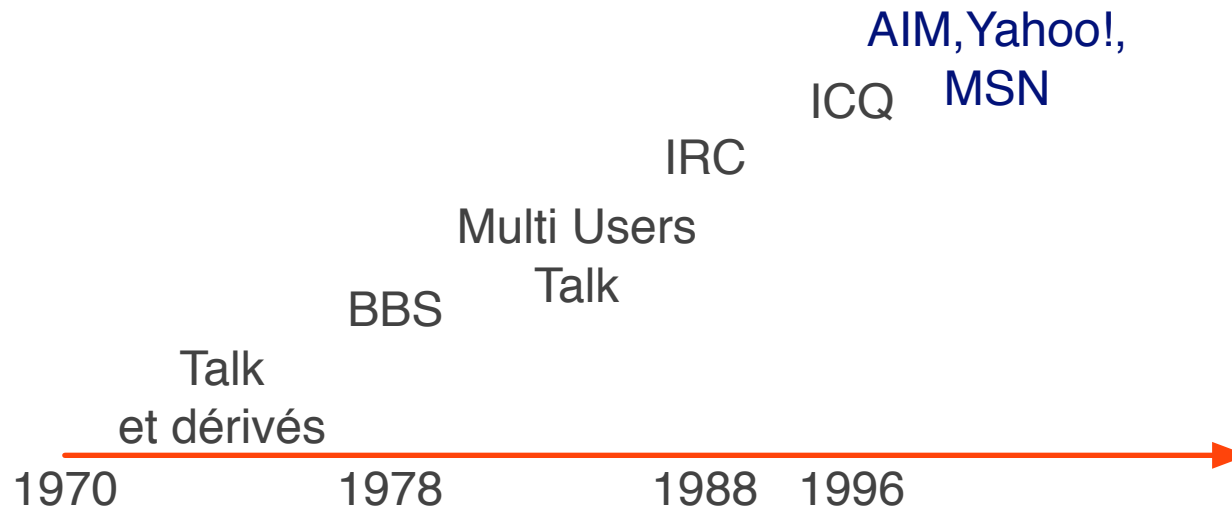
- **Internet Relay Chat**
- **Discussion de groupe**
- **Premier serveur en Finlande**
- **Reste en mode texte et réservé aux initiés**

Historique: Messagerie Instantanée



- **ICQ (“I Seek You”)**
- **développé par la société Mirabilis**
- **Démocratisation grâce à:**
 - Liste de contacts
 - Intégration de la présence et du statut

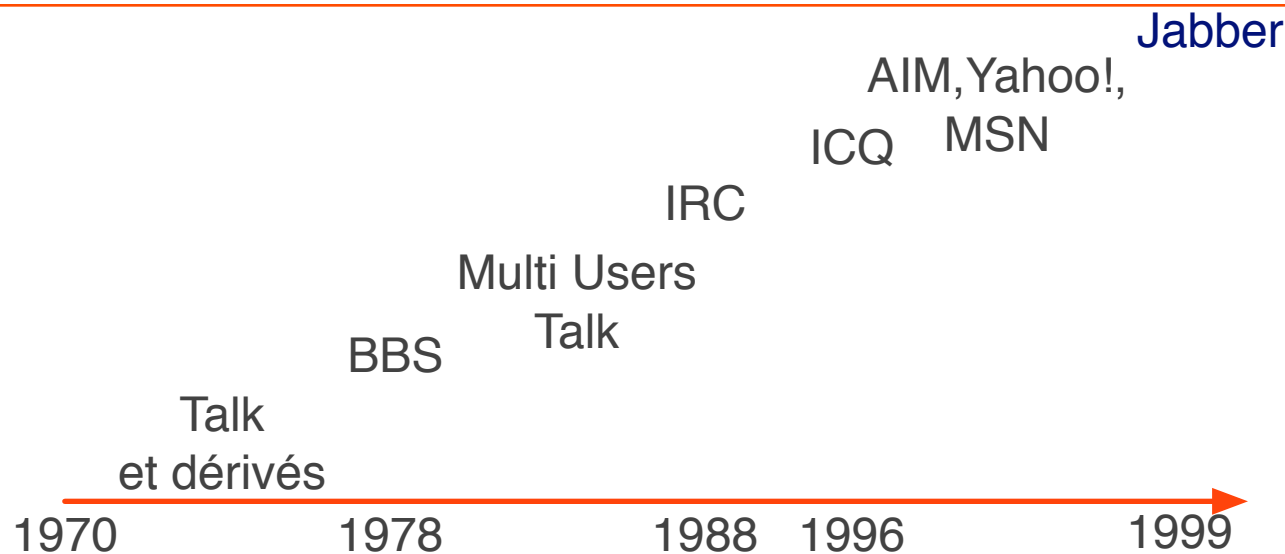
Historique: Messagerie Instantanée



■ **Devant le succès d'ICQ:**

- AOL IM
- Yahoo! Messenger
- MSN Messenger

Historique: Messagerie Instantanée



- **Tous les nouveaux réseaux sont incompatibles**
- **Jabber est créé pour permettre à un utilisateur de réutiliser tous ses comptes depuis un même système**

Jabber et XMPP



Jabber

- **Inventé par Jérémie Miller en 1999.**
- **Premier server public: Août 1999**
- **Basé sur XML pour faciliter l'interopérabilité**
- **Objectif de standardisation dès l'origine**
 - Rapprochement avec l'IETF qui échoue à trouver un consensus sur IMPP (Instant Messaging and Presence Protocol)

Jabber repose sur deux piliers

- **Protocole**
- **Communauté**

Forces du protocole Jabber

- **Dès l'origine le protocole est conçu avec les objectifs suivants:**
 - Extensibilité: usage d'XML et des espaces de nommage.
 - Distribution naturelle: comme le mail, connexions de serveur à serveur (S2S).
 - Notion de passerelle pour l'interopérabilité
 - Conçu comme un bus de données générique, dont une des applications est la messagerie instantanée

Simplicité du protocole

- **Un des facteurs de succès a été la simplicité du protocole, utilisable avec *telnet*.**

A quoi ressemble le protocole XMPP ?

- **Documents XML entrelacés**
 - Le flux client et le flux serveur forment chacun un document XML.
- **Un tag de premier niveau**
 - `<stream>`
- **Trois types de tag de premier niveau:**
 - **message**: communications asynchrones
 - **iq**: communications synchrones
 - **presence**: données de présence et statut

XMPP / SIMPLE

- **SIMPLE est un dérivé de SIP pour la messagerie instantanée.**
 - SIP domine aujourd'hui sur la voix mais reste trop pauvre sur la présence.
- **XMPP domine sur la présence et la voix est seulement en émergence:**
 - Jingle, protocole de Google Talk

Jabber et la communauté de développement

- **La communauté de développement a fait la richesse du protocole**
 - Grande et rapide innovation
 - Projets concurrents et forte émulation
 - Développement de bibliothèques dans de nombreux langages de programmation
- **Cette richesse est également sa faiblesse:**
 - Faible coordination
 - Nombreux projets concurrents: le novice est perdu

La Jabber Software Foundation

- **Créée en août 2001 pour:**
 - rassembler la communauté de développement
 - lui donner une voix et une visibilité
 - devenir un interlocuteur de l'IETF en vue de la standardisation du protocole
 - se concentrer sur les extensions du protocole

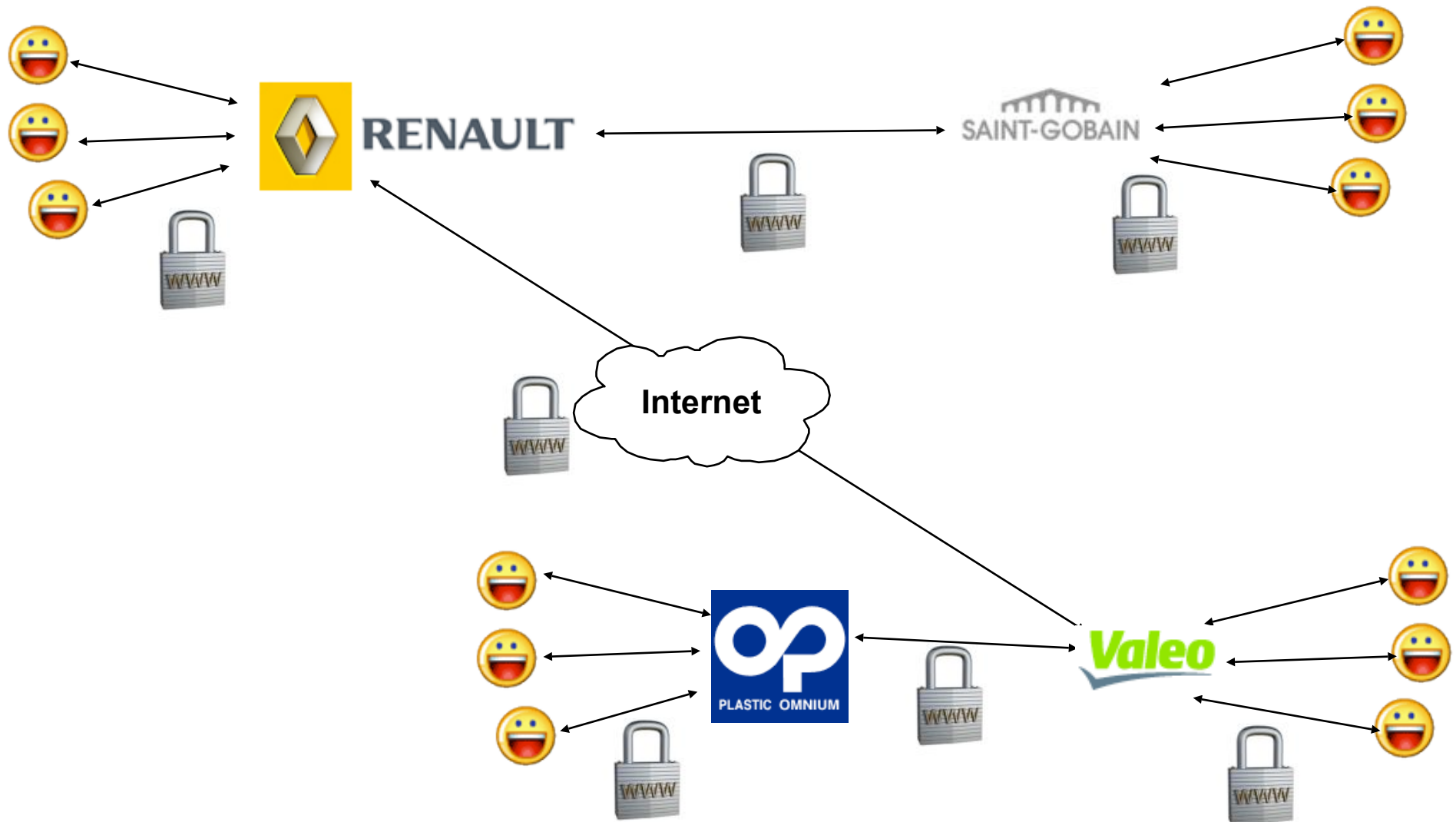
XMPP: un standard IETF

- **L'Internet Engineering Task Force est probablement un des organismes de standardisation du Web le plus connu, avec le W3C.**
- **XMPP est devenu un standard pour la messagerie instantanée en octobre 2004.**
- **Plusieurs RFCs dont les deux plus importantes:**
 - RFC 3920: XMPP Core (XML streams, SASL, TLS, stringprep profiles, stanza semantics)
 - RFC 3921: XMPP IM (XMPP extensions for basic instant messaging and presence)

JSF devient la XSF

- **La Jabber Software Foundation est devenu début Janvier 2007 la XMPP Standards Foundation**
 - Se concentrer sur la promotion et l'amélioration du standard
 - Accélérer l'adoption

Importance d'un standard d'IM



XMPP: Applications concrètes et déploiements



Une adoption qui s'accélère

■ Grands éditeurs

- Google
- Sun
- Apple
- Adobe
- Nero

■ Grands groupes

- Atos
- EDS

Une adoption qui s'accélère

- **Grand de la VOIP**

- Wengo
- Sipphone

- **Grands Fournisseurs d'accès:**

- Orange
- Portugal Telecom

- **Grands sites publics**

- Meetic
- Skyrock
- GoowyMedia
- MxIT

L'avenir



Evolution du standard

- **Réseau sécurisé et distribué**
- **Renforcement de la domination est de la fiabilité dans le domaine du chat texte**
- **Evolution du standard de la voix**
- **Accélération de l'adoption avec une extension du domaine d'application**

L'avenir de l'IM passe plus que jamais par XMPP



Aperçu du protocole XMPP



Exemple de session XMPP (I)

- étape 0: telnet localhost 5222
- étape 1: Open XMPP stream
 - Client envoie:
 - `<?xml version='1.0'?>`
 - `<stream:stream xmlns:stream="http://etherx.jabber.org/streams" to="localhost" xmlns="jabber:client">`
 - Serveur envoie:
 - `<?xml version='1.0'?>`
 - `<stream:stream xmlns='jabber:client' xmlns:stream='http://etherx.jabber.org/streams' id='3694886828' from='localhost'>`

Exemple de session XMPP (2)

- étape 2: Login
 - Client envoie les informations de login:
 - `<iq type='set' id='auth'>`
 - `<query xmlns='jabber:iq:auth'>`
 - `<username>mremond</username>`
 - `<password>azerty</password>`
 - `<resource>TelnetClient</resource></`
 - `query></iq>`
 - Server confirme le login:
 - `<iq type='result' id='auth' />`

Exemple de session XMPP (3)

- étape 2: Login
 - Le serveur peut renvoyer une erreur si l'authentification échoue:
- `<iq type='error' id='auth'>`
 - `<query xmlns='jabber:iq:auth'>`
 - `<username>mremond</username>`
 - `<password>D</password>`
 - `<resource>TelnetClient</resource></query>`
- `<error code='401' type='auth'>`
 - `<not-authorized`
 - `xmlns='urn:ietf:params:xml:ns:xmpp-stanzas' /> </error></iq>`

Exemple de session XMPP (4)

- étape 3: Présence
 - Client envoie les données de presence
- `<presence/>`
 - Si les contacts dans notre roster sont en lignes, ils reçoivent un paquet présence:
- ```
<presence from='mremond5@localhost/OneTeam'
 to='mremond@localhost/TelnetClient'>
 <priority>8</priority> </
 presence>
```



# Exemple de session XMPP (5)

---

- Etape 4: Envoie d'un message
  - Client envoie un message à un autre utilisateur / application:
  - ```
<message to='mremond5@localhost' >  
  <subject>Hello</subject>  
  <body>I am chatting with a Telnet  
  client !           </body>  
</message>
```
 - Pas de réponse du serveur (asynchrone)

Exemple de session XMPP (6)

- étape 5: Recevoir un message
 - Nous pouvons recevoir des messages d'autres clients:
- ```
<message from='mremond5@localhost/tkabber'
 to='mremond@localhost/TelnetClient'
 type='chat'
 xml:lang='fr-FR'>
 <body>Answer from the other side</body>
 <x xmlns='jabber:x:event'>
 <offline/>
 <delivered/>
 <displayed/>
 <composing/></x></message>
```

# Exemple de session XMPP (7)

---

- étape 6: Mise à jour de la présence
  - Les informations de présence sont mises à jour, par exemple lorsque un utilisateur se déconnecte:
- ```
<presence from='mremond5@localhost/OneTeam'  
to='mremond@localhost/TelnetClient'  
type='unavailable' />
```

Exemple de session XMPP (8)

- étape 7: Fermer le flux / document XML:
 - Client ferme le tag stream (Cela termine un document XML valide):
 - `</stream:stream>`
 - Le serveur ferme alors le tag XML et se déconnecte (cela ferme le second document XML):
 - `</stream:stream>`