Cozy Cloud anatomy

cozy.io

A PERSONAL CLOUD PLATFORM TO CONNECT YOUR DATA
SNAPSHOT

Cozy in a nutshell

KEY FIGURES

2012 5,2 M€ 20
Creation Funds raised Team members

MANAGEMENT TEAM

- Benjamin ANDRÉ, CEO, experienced entrepreneur
- Tristan NITOT, CPO, founder of Mozilla Europe + former member of French Gov’t Digital Advisory Board.

CLIENTS

EDF OVH.com LA POSTE MAIF

STRATEGIC PARTNERS

orange Inria Fing

INCUBATED AT

Firefox
WHY ?
ASSUMPTIONS

1. we want **more smart services**

2. digital services relevance comes from:
   - frictionless user experience
   - **data mashups**, data mixing
BARRIERS:

1. Data are walled gardened into the big internet platforms & our vendors

2. This fluidity requires trust. Without, it will be un-sustainable.
CONSEQUENCE:

- We want more fluidity for our data,
- but with some strong guaranties to enforce our trust.
TWO POSSIBILITIES:

1. Interoperability and consent management. keywords: APIs, protocols, agreements, regulation...: the technical challenge is huge and in the end, everybody will progressively require access to all your data, like apps on your phone...

2. Personal Cloud:
   - a personal data storage connected to all your digital third parties (devices, IoT, online services, energy provider, telco, insurance, bank...)
   - an ecosystem of personal apps you can choose to run on your personal cloud
OUR SOLUTION: a personal cloud
Cozy Cloud’s approach
MY PERSONAL CLOUD

Cozy: connect all your data
KEY ADVANTAGES

True added value for users

- SINGLE SIGN-ON
- GLOBAL SEARCH
- FRICTIONLESS APPS INTEGRATION
- IoT HUB
- PERSONAL BIG DATA
- PRIVACY
TECHNOLOGY
Open source architecture

- Modular → Adaptive
- Sandboxed → Safe
- Simple → Fast to learn (no SDK, REST, JSON)
- Data Centric → Sharing, notifications, sync...
- With 2 main technical concepts:
  - pPaas (personal Paas)
  - DataSystem (semantic storage)
CONCLUSION:

- It's all about services and usages
- Privacy is a requirement, an enabler, not an end.
- Personal storage is not enough: personal apps ecosystem is key
- Open source: for trust and hackability (ecosystem)
- Come at 14:00 – workshop about business models
CRAZY?
1977

Ken Olsen, DEC’s founder (Digital Equipment Corporation)

“There is no reason anyone would want a computer in their home”
1977

Steve Jobs & Steve Wozniak launched the Apple I ...
OUR VISION

Change the paradigm to change the rules

1980
Computer → Personal Computer

Today
Cloud → Personal Cloud
• P2P Big Data?
• Architecture?
• Mobile?
• Security, trust issues?
• Business model?
• Beyond geeks?
• What can we do together?
• Open source? …
ANNEXES
TRUST & SAFETY

Why our platform is secure

- OPEN SOURCE
- APP AUDITING
- SECURITY ALERTS
- PARTITIONED ARCHITECTURE
- DATA DISTRIBUTION
- VIRTUOUS BUSINESS MODEL
ASSUMPTIONS
1. we want more smart services
2. digital services relevance comes from:
   • frictionless user experience
   • data mashups, data mixing

BARRIERS:
1. Data are walled gardened into the big internet platforms & our vendors
2. This fluidity requires trust, because in a world where our decisions are driven by data, people who access your data will have a great power on you.

CONSEQUENCE: we want more fluidity with our data, but with some strong guaranties to enforce our trust.

TWO POSSIBILITIES:
1. Interoperability and consent management. keywords: APIs, protocols, agreements, regulation... the technical challenge is huge and in the end, everybody will progressively require access to all your data, like apps on your phone...
2. Personal Cloud:
   • a personnal data storage connected to all your digital third parties (devices, IoT, online services, energy provider, telco, insurance, bank...)
   • an ecosystem of personal apps you can choose to run on your personnal cloud

THE COZY ANATOMY: demo & architecture

CONCLUSION:
TECHNOLOGY
Open source architecture

- Modular → Adaptive
- Sandboxed → Safe
- Simple → Fast to learn (no SDK, REST, JSON)
- Data Centric → Sharing, notifications, sync...
- With 2 main technical concepts:
  - pPaas (personal Paas)
  - DataSystem (semantic storage)
TECHNOLOGY

Host-side architecture

CLUSTER CONTROLLER
- Distributed instance creation & routing
- Backup
- Monitoring
- Update
- Load balancing
- Default restoration
- Link with client base (invoicing, status…)

CHARACTERISTICS
- Not Open Source
- Top R&D to create a product which is:
  - Robust (less human interactions needed)
  - Scalable (less machine needs, Cozy addition is sub-linear)

FARM 1
- Machine
- Machine
- Machine
- Machine

Machine with inter-Cozy mutualized components
- Proxy
- Couch
- App X
- App Y
- App Y

FARM N
- Machine
- Machine
- Machine
- Machine

Machine with inter-Cozy mutualized components
- Proxy
- Couch
- App X
- App Y
- App Y
Multi-platform apps

“Deploy once run everywhere" apps are:

• A set of static files on the server side (app.js on the scheme)
• Those files are uploaded in the browser or in native apps on the Cozy mobile client (Cordova) or the Cozy desktop client (NW.js)
• The developer interacts with the persistence with the ODM (Object Document Mapper) library.
• The ODM will use the DataProxy whether of the browser or of the Native Cozy app on the device.
• App.js doesn’t have to deal with being online or not, this is the role of the DataProxy.
• The DataProxy is powered by the synchronization capacities of CouchDB / PouchDB.
• This architecture is inspired by offline first approaches.

⇒ the only constraint for the developer is to develop a responsive app and to use the ODM for persistence !
⇒ Then his app can run in the browser or on a native app on desktops or mobiles
⇒ Users don’t have to install the apps nor to sync them, it is automatic…
CONCLUSION

Cozy:

1. An aggregator of aggregators
2. Focused on usages, services, stockage
3. Join a deep decentralization movement (cf Blockchain)
4. Gives services an absolute and non intrusive digital intimacy
5. An ecosystem rather than isolated services.

=> Together, let's Uberize GAFAs!