

# Telex: A platform for decentralised sharing

Marc Shapiro

<http://lip6.fr/Marc.Shapiro/>

Jean-Michel Busca, Lamia Benmouffok, Pierre Sutra

INSTITUT NATIONAL  
DE RECHERCHE  
EN INFORMATIQUE  
ET EN AUTOMATIQUE

INRIA  
centre de recherche PARIS - ROCQUENCOURT

Grid4All

LIP6  
Dpt : Réseaux  
et Systèmes Répartis

## P2P collaborative work

Shared documents

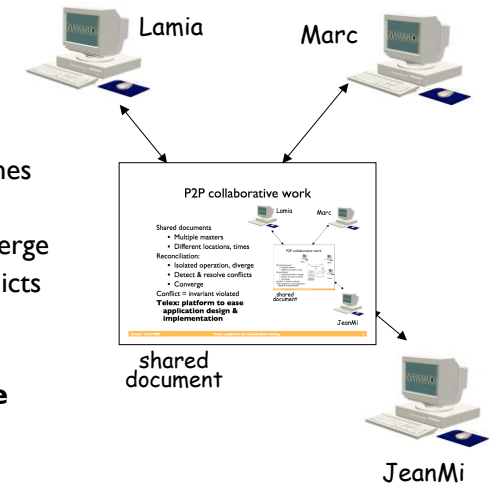
- Multiple masters
- Different locations, times

Reconciliation:

- Isolated operation, diverge
- Detect & resolve conflicts
- Converge

Conflict = invariant violated

**Telex: platform to ease application design & implementation**



## Competition

File systems, file synchronisers, database “replication mode”

- Patis, dot-mac, Windows Live, Oracle
- Last Writer Wins: silently loses updates

Collaboration suites

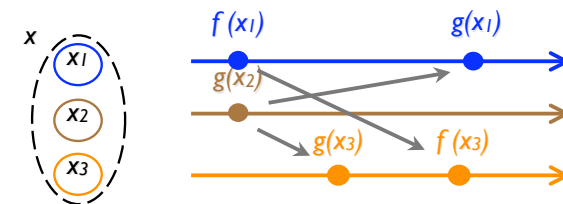
- Exchange calendar: centralised
- Groove: Last Writer Wins

Shared text editors

- Google Docs, Wikipedia: centralised, LWW
- subEthaEdit, Wooki: Operational Transformation
- SVN: central repository, text only

**Telex: principled, general-purpose, P2P**

## An abstract model



Objects  $x, y, \dots$

Replicas,  $x_2, x_3, \dots, y_1, y_2, \dots$  at sites 1, 2, 3, ...

One site proposes (atomic) operation  $f(x_1, y_1)$

Propagate to other sites; replay operation  $f(x_2, y_2), f(x_3, y_3), \dots$

If conflict: may rollback, abort



# Concurrency control with constraints

Action: reified operation

Constraint: reified Application-supplied concurrency control specifications

Binary relations:

- NotAfter  $\alpha \rightarrow \beta$
- Enables (implication)  $\alpha \triangleleft \beta$
- NonCommuting  $\alpha \# \beta$

Combinations:

- Antagonistic  $\alpha \rightarrow \leftarrow \beta$
- Atomic  $\alpha \triangleleft \triangleright \beta$
- Causal dependence  $\alpha \rightarrow \beta \wedge \alpha \triangleleft \beta$

Action-constraint graph ACG

## Example: Sakura Shared Calendar over Telex

Marc  $\triangleleft \triangleright$  Lamia

Tues.  $\rightarrow \leftarrow$  Fri.

Private calendar + common meetings

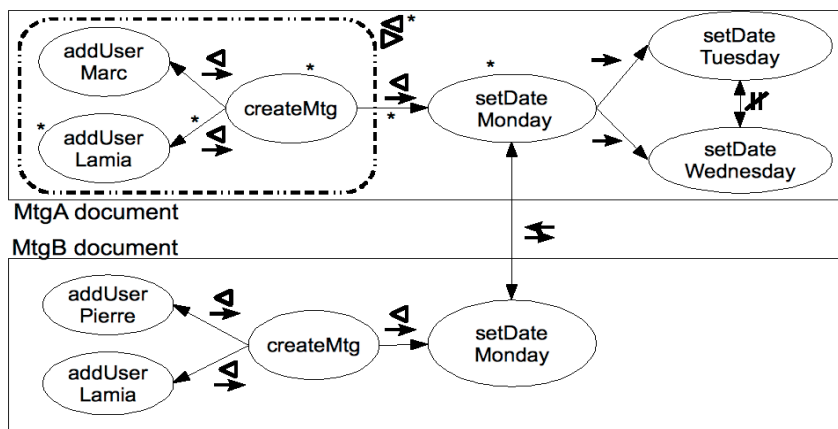
Example proposals:

- M1: Marc & Lamia & JeanMi, Monday | Tuesday | Friday
- M2: Lamia & Marc & Pierre, Tuesday | Wed. | Friday
- M3: JeanMi & Pierre & Marc, Monday | Tues. | Thurs.

Telex:

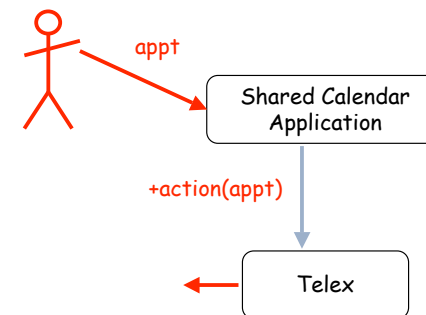
- (Local) Explore solutions, select
- (Global) Combine, Commit
- Persistence, replication, peer-to-peer
- Disconnected operation

## Example ACG: calendar application



Schedule: sound cut in the graph

## Telex lifecycle Application operations



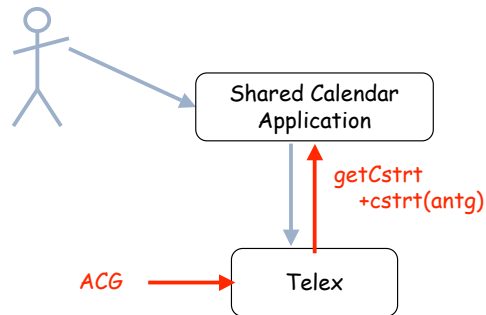
User requests

- > application: actions, dependence
- > Telex: add to ACG, transmit, merge



## Telex lifecycle

### Receive remote operations

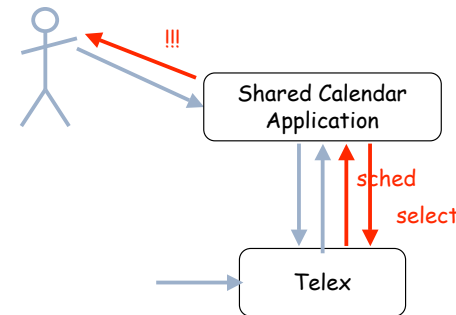


User requests

- > application: actions, dependence
- > Telex: add to ACG, transmit, merge

## Telex lifecycle

### Detect & display conflict

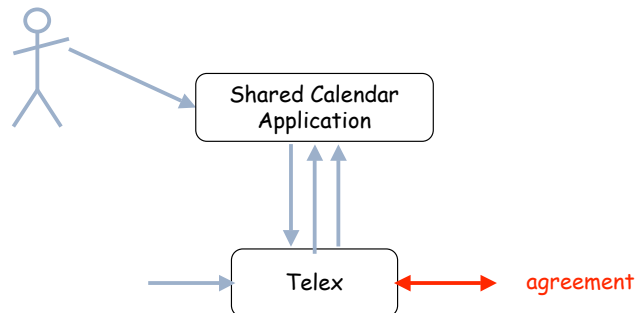


ACG

- > Telex: valid path = sound schedule
- > application: tentative execute
- > application: if OK, select

## Telex lifecycle

### Convergence protocol



Selected schedules

- > Telex: exchange, agree
- > commit/abort, serialise

## Scheduling & convergence

Schedule: satisfies constraints

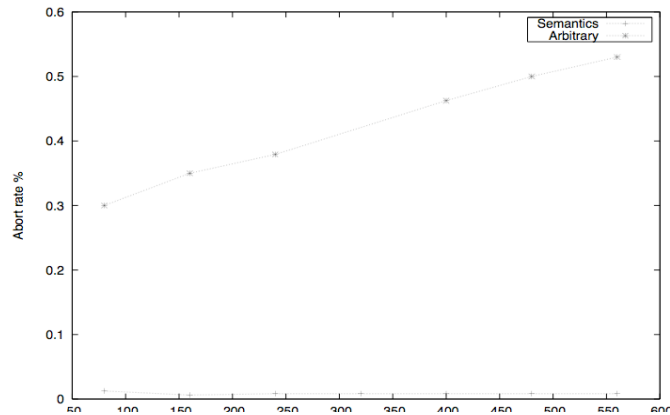
- No conflicts
- Ordering, implication
- Optimal (heuristic): penalise lost work
- If *selected*: proposed to commitment

Commitment:

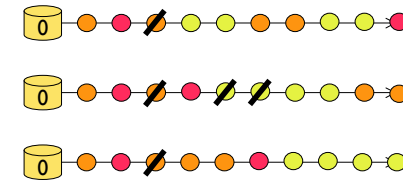
- Combine selected schedules
- Equivalent schedule at each site



## IceCube heuristic superior



## Eventual consistency



Optimistic: diverge arbitrarily

Common stable prefix

Liveness:

- Every action eventually (aborted or committed) in prefix
- Consensus on next extension of prefix

## Application design

MVC-like execution loop + rollback

Spend effort to reduce conflicts!

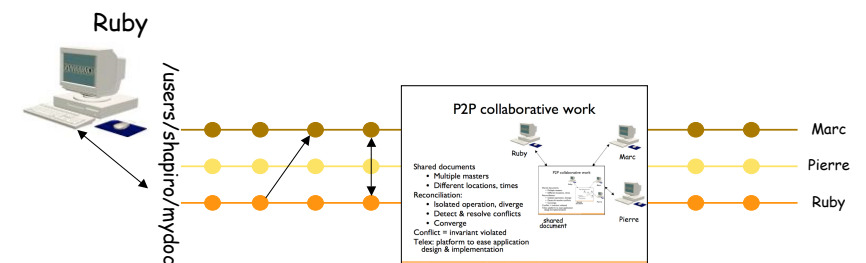
Concurrent action pairs

Commute (no constraints)

>> Antagonistic

>> Non-Commuting

## ACG representation = multilog

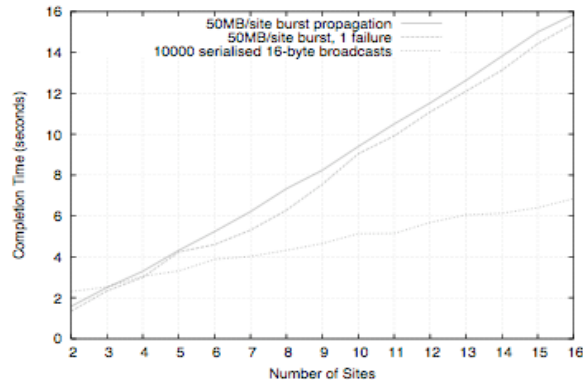


Avoid write contention: per-doc, per-writer logs

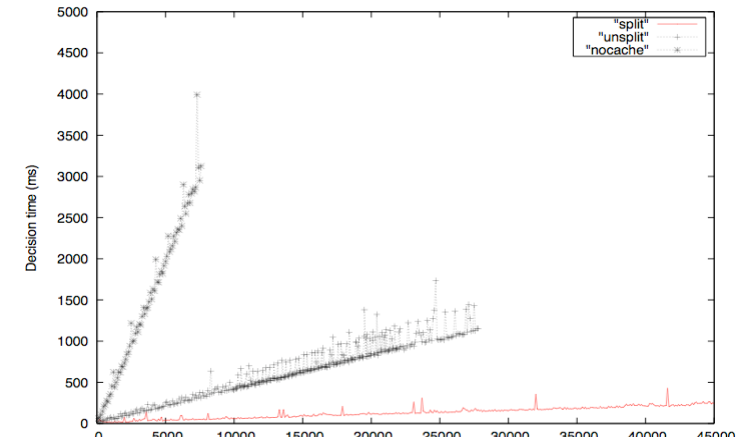
- Append to my document log
- Transmit sequentially
- Read all document logs



## Multilog avoids write contention



## Decomposing ACG improves performance



## Status of Telex

### ANR Respire + FP6 Grid4All

- 35,000 Java (well-commented) lines of code
- Academic & industry partners
- $\geq$  Nov. 2009: development engineer
- Available at [gforge.inria.fr](http://gforge.inria.fr), BSD license

### Applications:

- Sakura co-operative calendar (*demo*)
- Decentralised Collaborative Environment (UPC)
- Database, STMBench7 (UOC)
- Co-operative text editor (UNL, INRIA, UOC)
- Co-operative ontology editor (U. Aegean)

## Telex advantages

Telex: platform for sharing mutable data, decentralised, WAN operation

Separation of concerns

Optimistic application model

Takes over system issues

Eases application development

Available at [gforge.inria.fr](http://gforge.inria.fr) under BSD licence