Transactional Workflows:

Research, Enabling Technologies, and Applications

Amit Sheth

Bell Communications Research, Inc.

[Invited Talk: 10th Data Engineering, Feb. 1994] & VLDB'92, '93

Collaborations: U. of Houston (M. Rusinkiewicz, ..), Bellcore (N. Krishnakumar,..), MCC (M. Singh,..), ETH-Zurich, A Client Company.

Business Challenges

- Improve flexibility for re-engineering
- Increase automation to reduce cost and improve response time
- Support evolution and migration (accommodate both existing systems and new systems; increase use of the latter)

Example (Case Study of a Service Automation):

- 40 persons -> 12 persons (work center reorganization, re-engineering) -> 2-3 person (automation)
- Six Weeks -> Next Day -> Few Minutes

About the environment:

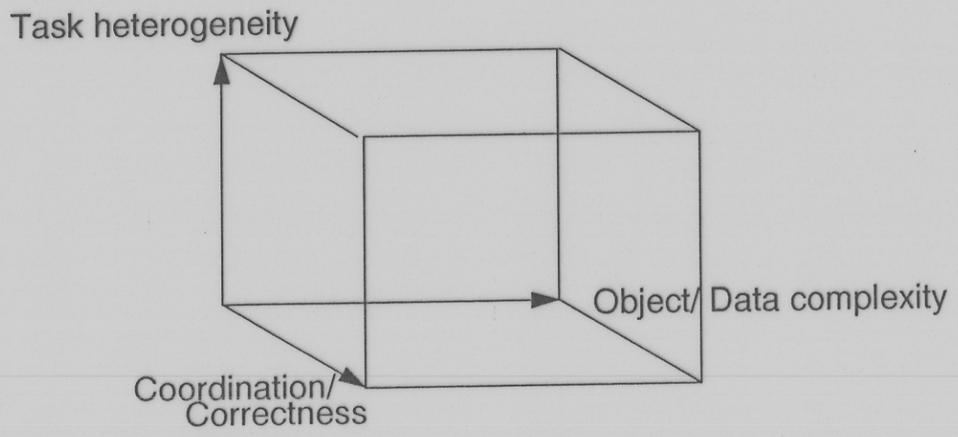
Types of Tasks

- **user tasks** involving humans in processing task
- **application tasks**:
  - scripts involving terminal emulations to remote systems
  - application programs/systems providing data manipulation (filters)
  - contracts (predefined interfaces) to large application systems
  - client programs or servers invoking other servers
  - database transactions

[Sheth and Krishnakumar 94]

Technology & Markets

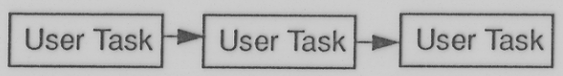
- Workflow Automation Software
  - Administrative/E-mail-based: e.g., expense reporting
  - Ad-hoc/Project-based [Groupware]: e.g., proposal preparation
  - Production/"Transaction"-oriented: e.g, loan appl. processing
- Transaction Processing Systems
- Relaxed Multidatabase Transactions



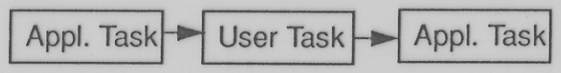
Workflow (WF) Automation Software vs Transactional Workflow (TWF)

(Typical Case)

Current workflow/workgroup software supports *office automation* functions (involving user tasks) (most products in PC and mainframe env.). Transactional workflows involves *application automation* (both user and application tasks with varied level of transactional properties).



In Current WF



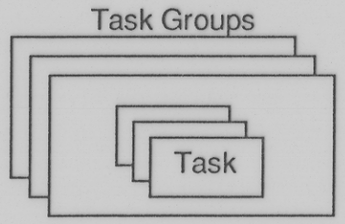
In TWF

[See Special Issue of Data Engg. Bulletin]

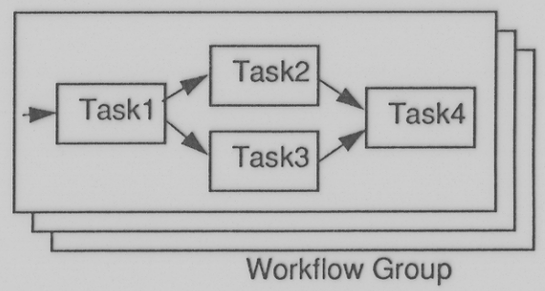
(Distributed) Transaction Processing (DTP) vs Transaction Workflow (TWF)

(Typical Case)

DTP/D-OLTP is focused on efficient execution of relatively simple tasks with no coordination across heterogeneous tasks (in different task groups). Transactional workflows requires coordinated execution of heterogeneous tasks, with varied levels of transactional properties, on a variety of systems.



In DTP



In TWF

- Earlier queued message systems and "chaining of transactions".
  Problems: insufficient control over transaction properties, one type of task, interactions among concurrent activities difficult.

Scheduling Approaches

- Based on Petri-net Models
- Executor for Flex. Trans. in a logically parallel language L.0
- Interpreter of MDB transaction specification Language (VLP)
- Interpreter of ECA rules
- Games vs. Nature
- Fine-state Automata
- Scheduling and enforcing intertask dependencies using temporal propositional logic

[Rusinkeiwicz and Sheth 94]

Application/Task and System Semantics

Semantics (Application/Task, System)	Impact (CC: Con. Control, R: Recovery)
limited commutativity (appl)	fewer exclusive locks (CC)
relaxed isolation (appl)	no global commitment (CC)
order preservation + rigorousness (sys)	early release of locks (CC)
idempotency (sys)	resubmit transactions (R)
+ monotonicity	roll-forward recovery (R)

[Jin et al 93]

Conclusions

- Defined a generic transaction workflow model. Motivation from real applications and environments. Developed high-level languages.
  - Specification: based on a formal model
    - tasks: task structures, types, classes, instances
    - intertask dependencies, controllable transitions, ...
    - interfaces, processing entities
  - Execution
    - executable/postulative specification
      - correct and safe execution
    - one approach to scheduling- implemented
- Completed two prototypes. Demonstrated with a real application.
  Technology Transfer leading to real systems.