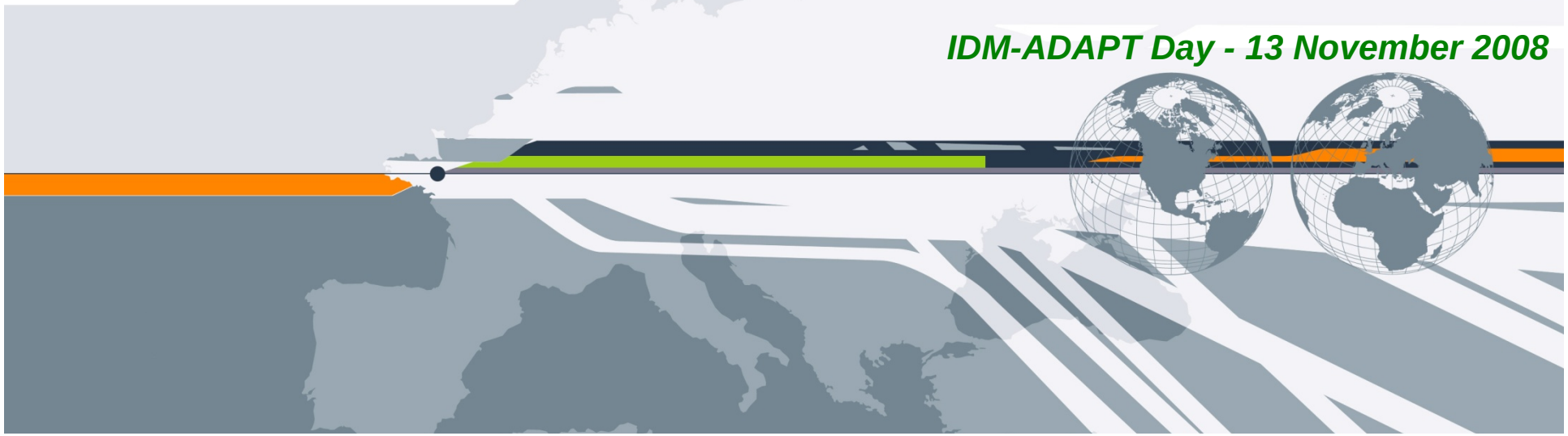
A model for distributed adaptation applied to replicated data management

Mohamed Zouari

Supervisors: Françoise André & Maria-Teresa Segarra



IDM-ADAPT Day - 13 November 2008

Issues

- Execution environment
  - Heterogeneity of hardware, network, software
  - Variation of execution conditions (connections/disconnections, failure…)
- Various and changing needs of users

Need for context management and dynamic adaptation

Issues

- Complex distributed applications
  - Many entities with different adaptation needs
- Single adaptation system
  - Complexity, Scalability, Adaptation overhead, Autonomy…

Need for distributed adaptation

- Long life time → non envisaged situations

Need for adaptable adaptation system

Contents

- Goals
- Illustrative application: data replication
- Our approach
- Case study: adaptation of replication
- Conclusion and future work

Goals

- To design a software architecture to ensure the dynamic self-adaptation of distributed applications
  - Granularity of entities to adapt (several applications, component…)
  - Model for dynamic adaptation
    - Adaptation functionalities

Contents

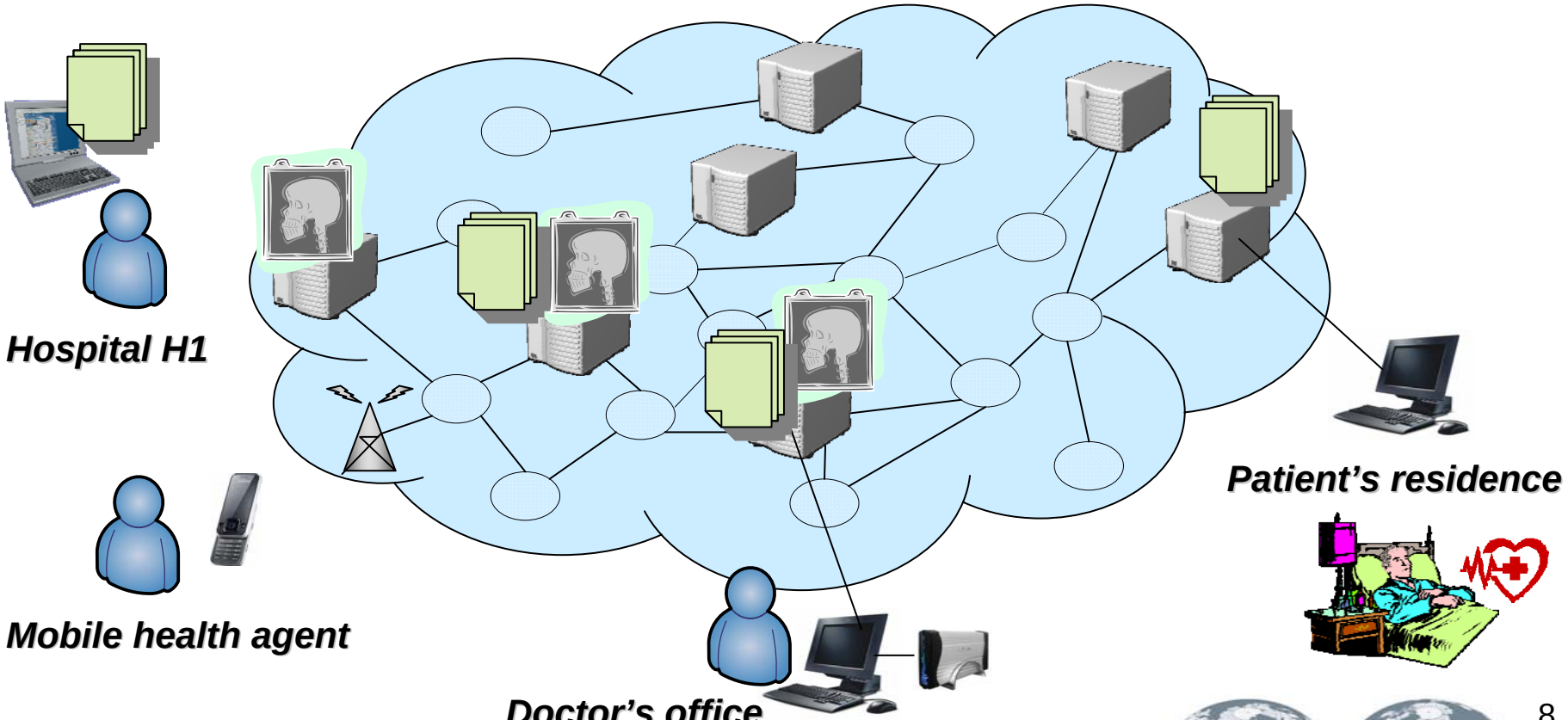
- Goals
- Illustrative application: data replication
- Our approach
- Case study: adaptation of replication
- Conclusion and future work

Illustrative application

- Distributed data replication system
  - Traditional technique to improve the quality of data sharing (performance, availability…)
- Need for dynamic distributed adaptation
  - Application has many algorithms
  - Replication services are distributed
  - Variability of execution context
    - Storage and treatment capacities of devices, bandwidth…
  - Varieties of user requirement
    - Latency of access, consistency level…

Illustrative application

- Replication of medical data among different health practitioners



Limits of existing works

- Framework for replicated data management not complete and/or not modular
  - e.g. RS2.7 is focused in consistency protocol [DRA03]
- Static mode of replication adaptation
  - e.g. rules to change the consistency protocol are embedded in the model [YU02]

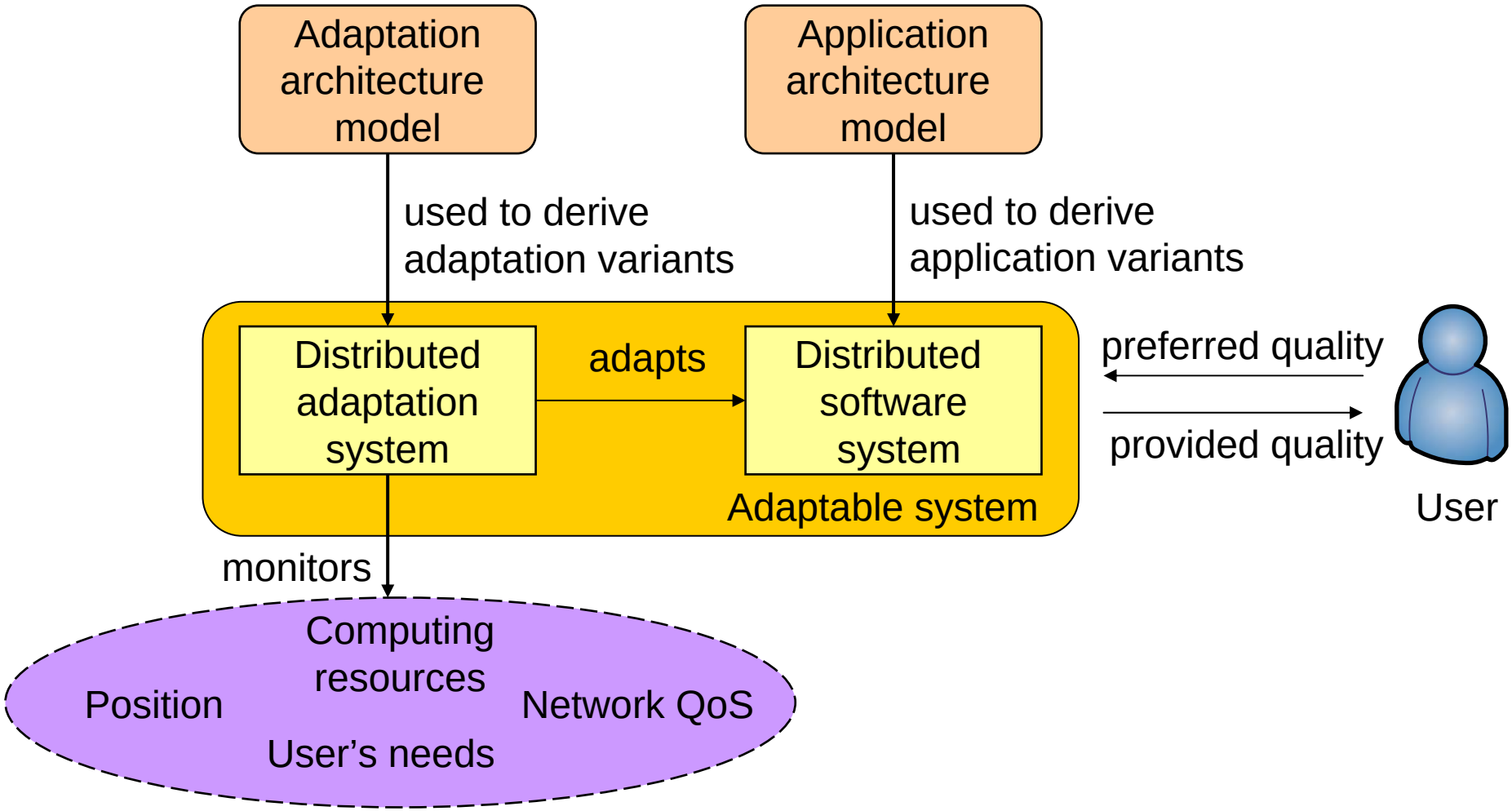
Contents

- Goals
- Illustrative application: data replication
- Our approach
  - Framework for replication system
  - Framework for distributed adaptation system
  - Cooperation of adaptation systems
- Case study: adaptation of replication
- Conclusion and future work

Our approach

- Distributed software system is a set of interacting entities
  - An entity is a component which can be composite and distributed
- Component-based framework for adaptation
  - Adaptation system is a set of components

Our approach

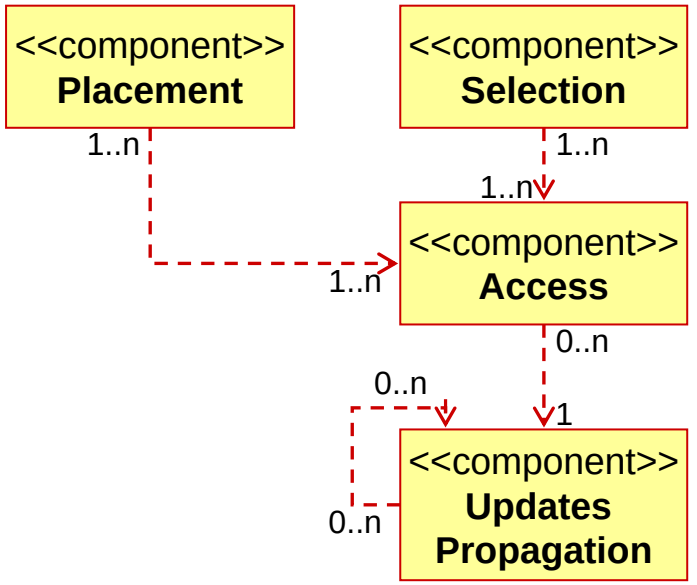


Contents

- Goals
- Illustrative application: data replication
- Our approach
  - Framework for replication system
  - Framework for adaptation system
  - Cooperation of adaptation systems
- Case study: adaptation of replication
- Conclusion and future work

Model for replication system

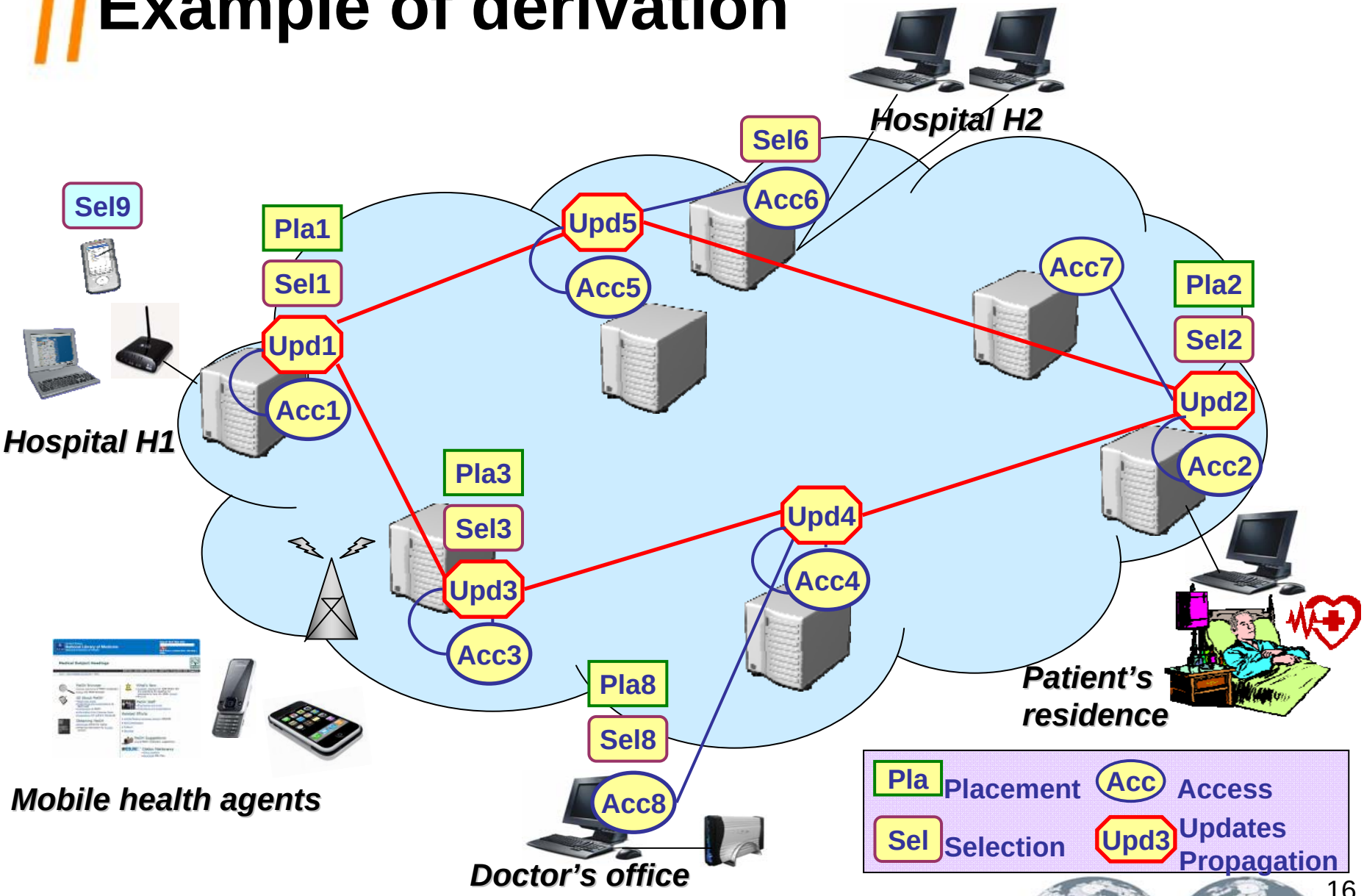
- Placement
  - Creates replicas and place them on sites
  - Deletes replicas
- Selection
  - Selects replicas to execute operations (read/write)
- Updates propagation
  - Ensures the functions contributing to the update of the replicas
- Access
  - Manages the local access (insert/delete and read/write)



Deriving a replication system

- Make a choice in points of variation
- Types of points of variation
  - Internal variation: define variables that determine program behavior
  - Behavioural variation: choose algorithms
  - Structural variation: select components that compose an replication system and bind them
  - Distribution variation: choose components placement over distributed nodes

Example of derivation

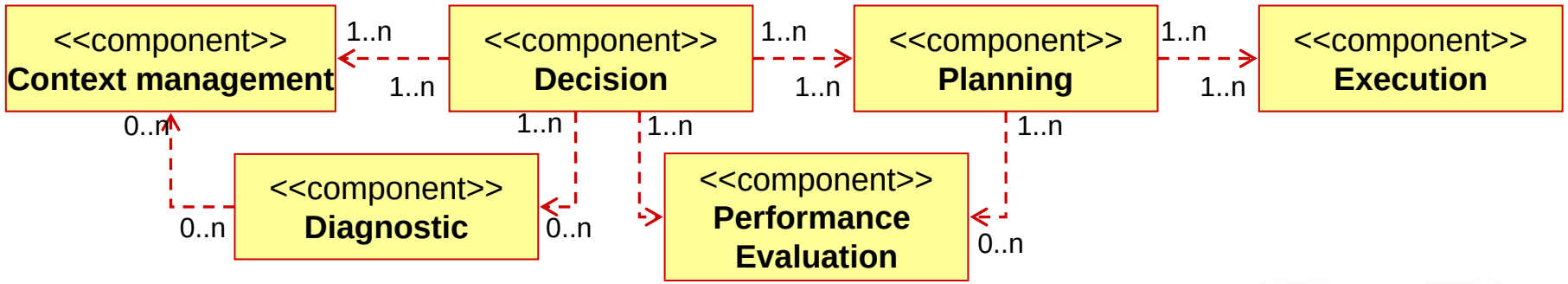


Contents

- Goals
- Illustrative application: data replication
- Our approach
  - Framework for replication system
  - Framework for adaptation system
  - Cooperation of adaptation systems
- Case study: adaptation of replication
- Conclusion and future work

Model for adaptation system

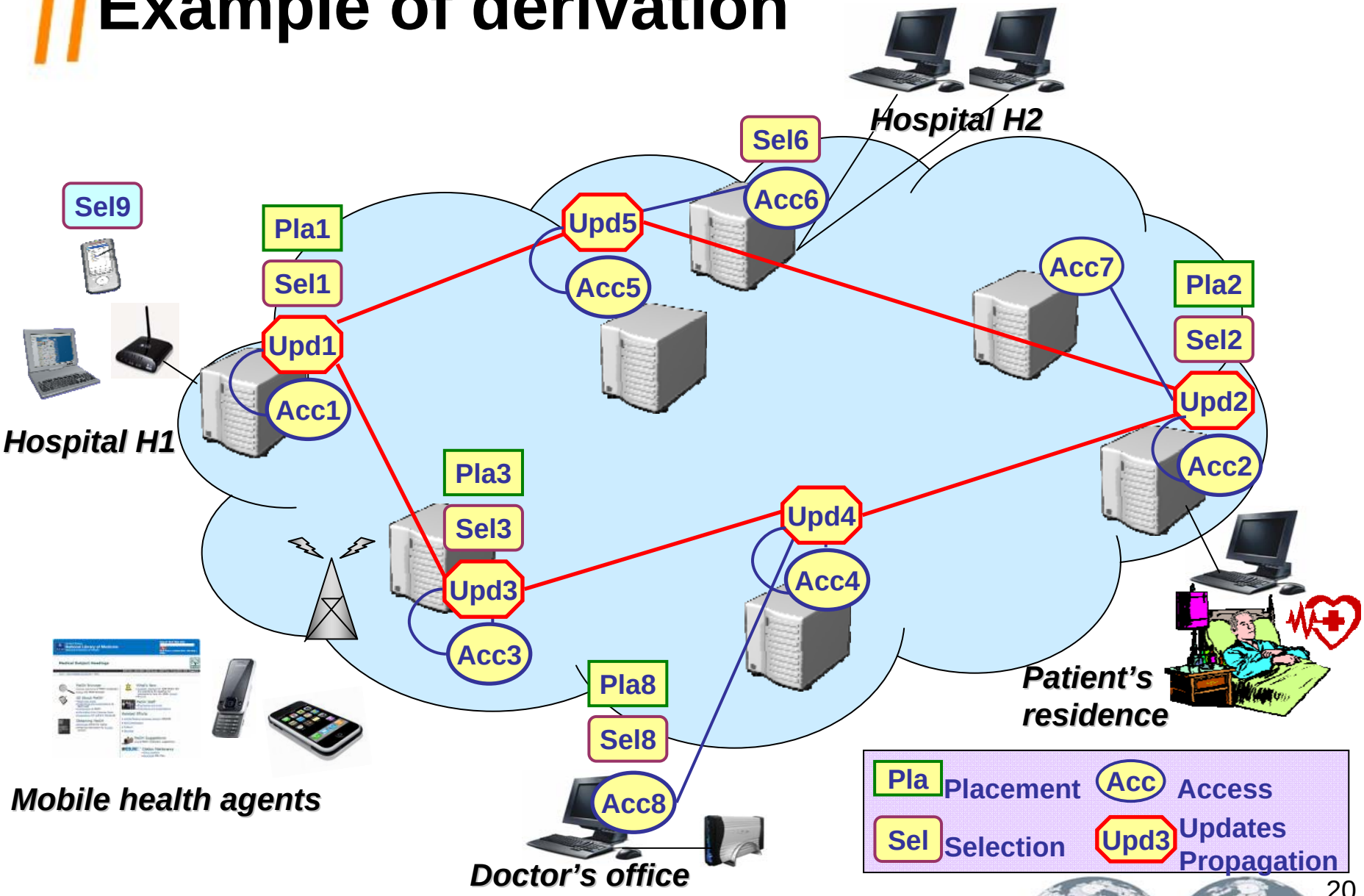
- Context management
  - Monitors and analyzes the context to make an entity context-aware
- Decision
  - Chooses the strategy to adapt an entity according to context
- Planning
  - Determines the adaptation plan which orders the actions that will make the entity achieve the strategy
- Execution
  - Executes the planned actions
- …



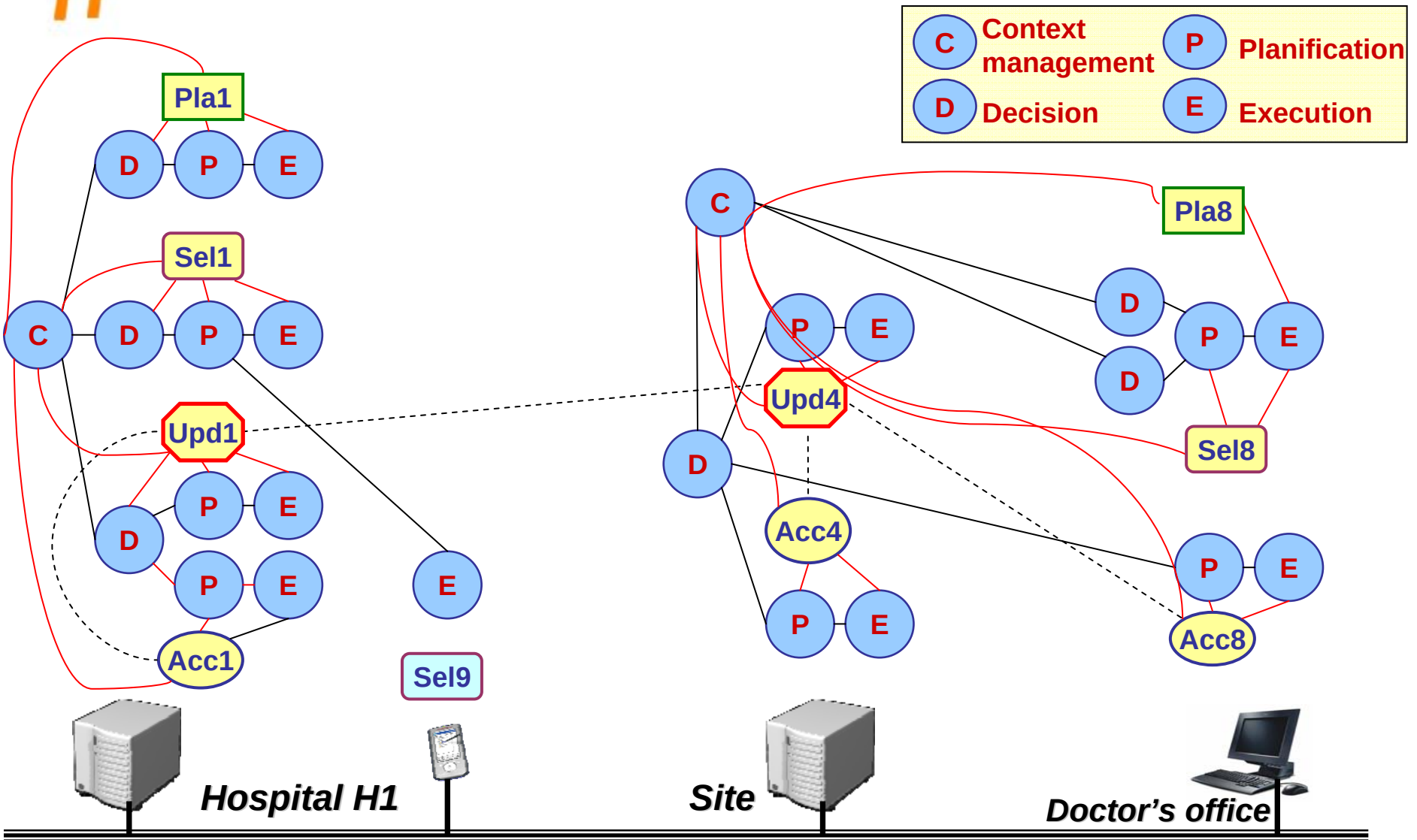
Deriving an adaptation system

- Make a choice in points of variation
- Types of points of variation
  - Internal variation: define variables that determine program behavior
  - Behavioural variation: choose algorithms
  - Specialization variation: define adaptation policy, adaptation guide…
  - Structural variation: select components that compose an adaptation system and bind them
  - Distribution variation: choose components placement over distributed nodes

Example of derivation



Example of derivation



Considered aspects when deriving

- Entities' adaptation requirements
- Evolution of adaptation system
- Flexibility
- Performance
- Scalability

Contents

- Goals
- Illustrative application: data replication
- Our approach
  - Framework for replication system
  - Framework for adaptation system
  - Cooperation of adaptation systems
- Case study : adaptation of replication
- Conclusion and future work

Cooperation of adaptation systems

- Entities are not aware about other entities' requirements → shared resources
  - Placement component uses a lot of bandwidth
- Some adaptations can affect the consistency of the global system
  - Replacing the placement strategy can affect the performance of consistency protocol

Some adaptation systems must cooperate

Contents

- Goals
- Illustrative application: data replication
- Our approach
- Case study: adaptation of replication
- Conclusion and future work

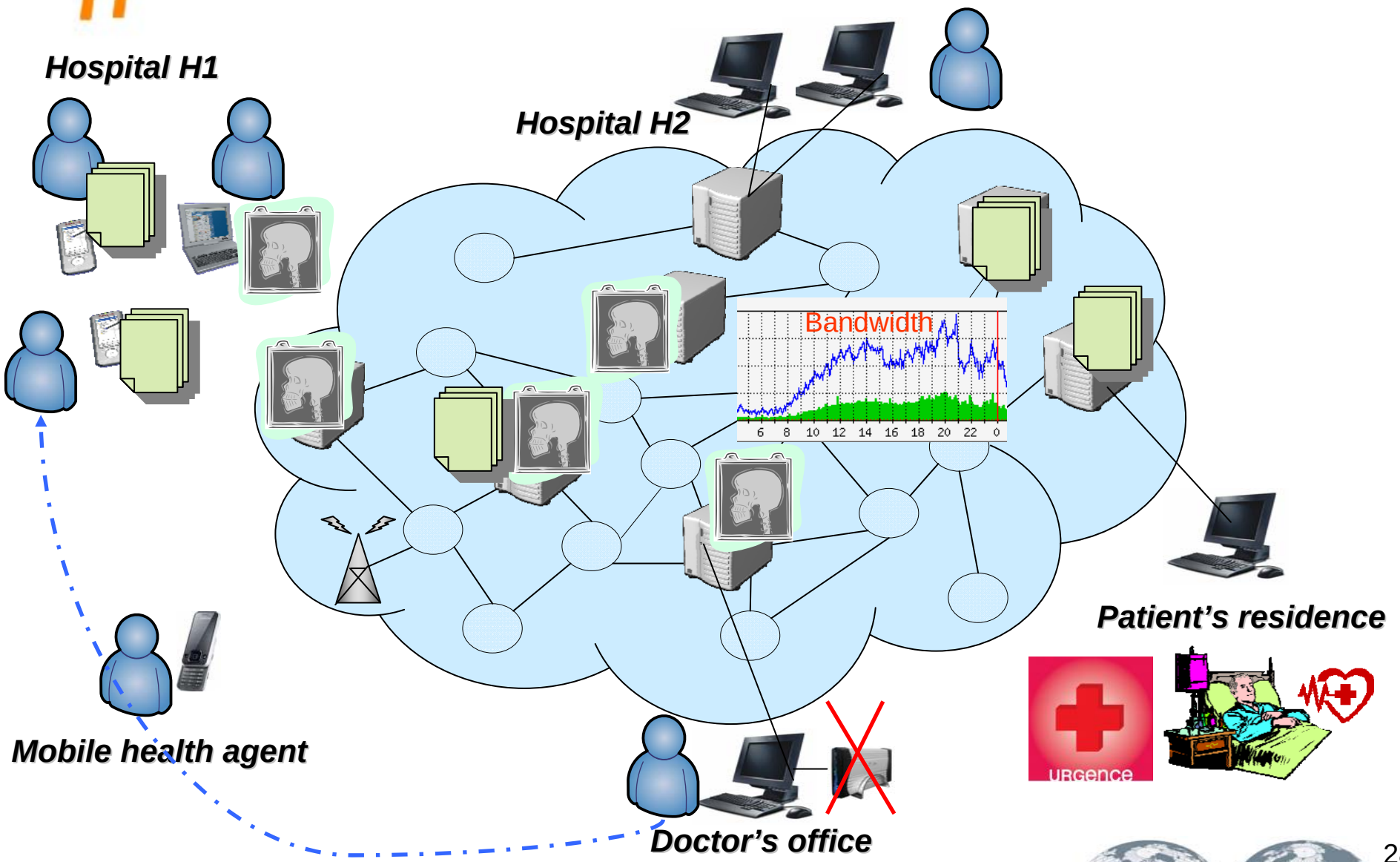
Case study

- Replication of medical data among different health practitioners
- Normal situation: Tele-surveillance for patient at home
  - Placement: Low replication degree, Random placement, Fixed machines
  - Selection: Random selection
  - Consistency: Weak

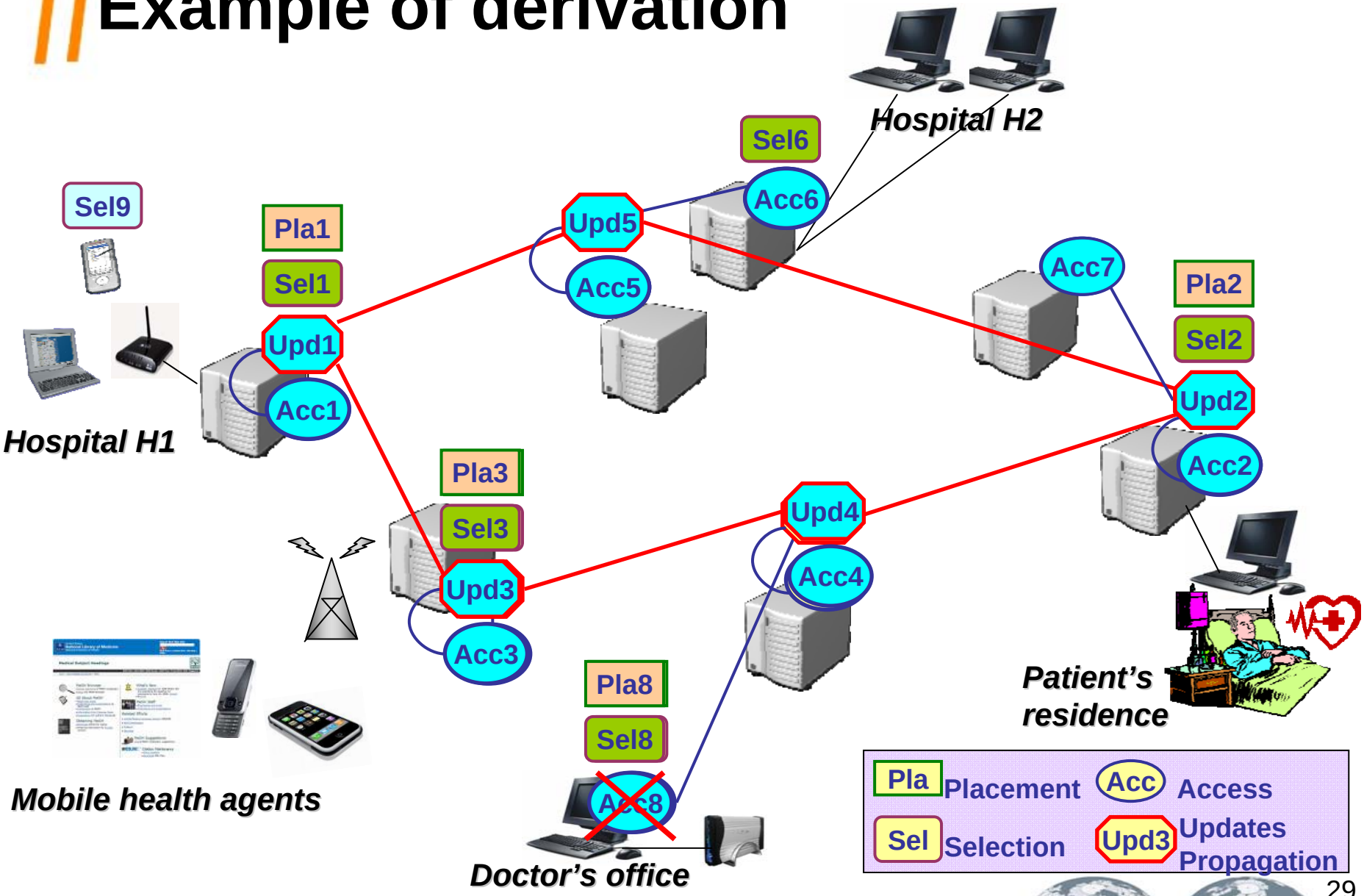
Case study

- Urgency situation: reduced access latency and freshness of replicas
  - Placement: High level of replication, Latency-based algorithm, Fixed and mobile machines
  - Selection: Latency-based algorithm
  - Consistency: Strong

Adaptation scenario



Example of derivation



Contents

- Goals
- Illustrative application: data replication
- Our approach
- Case study: adaptation of replication
- Conclusion and future work

Conclusion

- Distributed replication system reveals potential benefits of distributed adaptation
- Preliminary model-driven approach for constructing an adaptation system for distributed entities
- Experimentation
  - Fractal/Julia
  - Implementation of a preliminary prototype
  - Decisions in points of variation are made manually by the adaptation designer

Future work

- General approach for construct an adaptation system for distributed entities
- Study different cooperation techniques and cooperation strategies
- Make the adaptation system adaptable

References

[DRA03] Stéphane Drapeau. RS2.7 : un Canevas Adaptable de Services de Duplication. PhD thesis, Institut National Polytechnique de Grenoble, Grenoble, France, juin 2003

[YU02] YU H., VAHDAT A. Design and Evaluation of a Conit-based Continuous Consistency Model for Replicated Services. ACM Transactions on Computer Systems (TOCS). 2002.