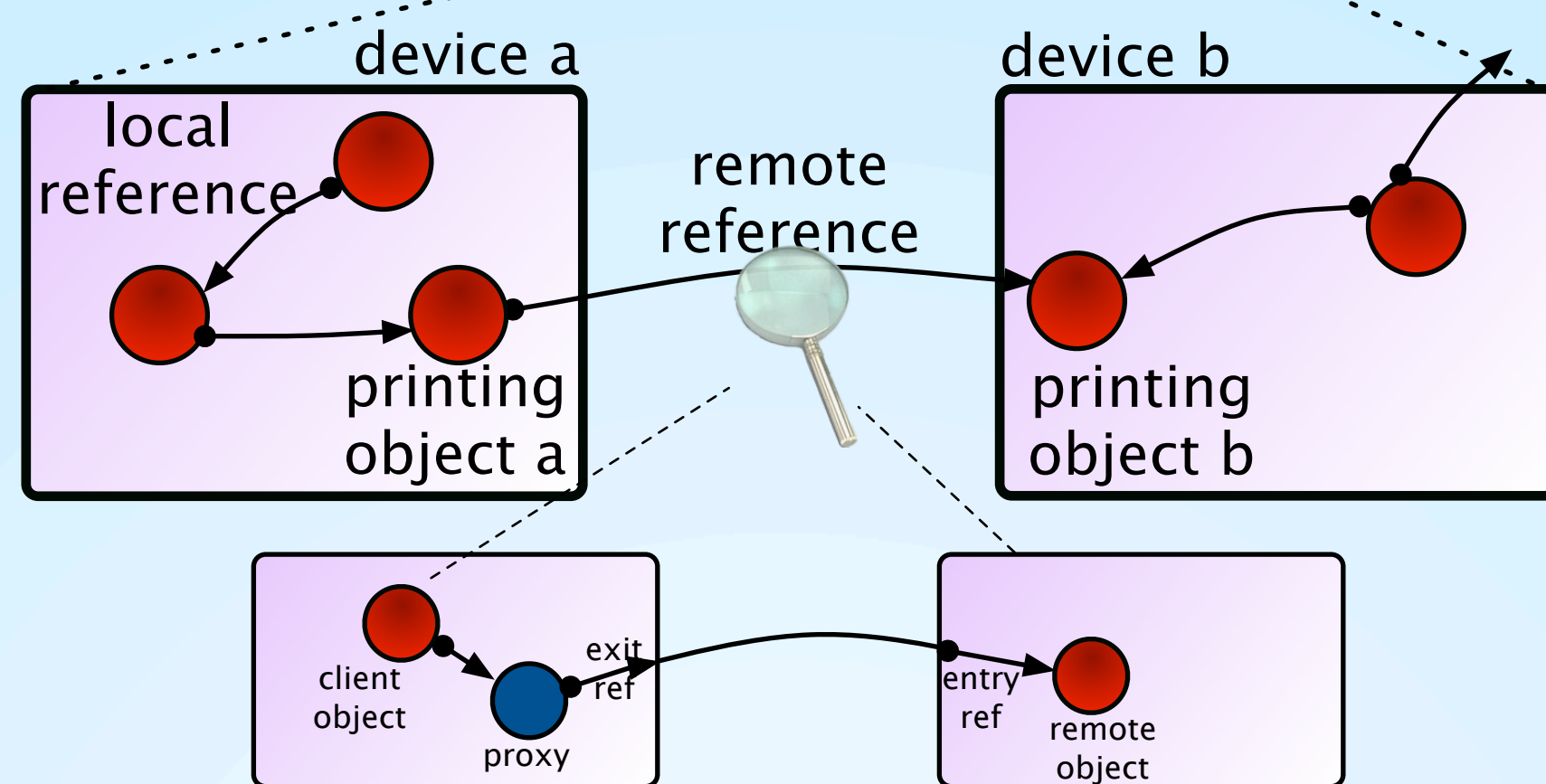
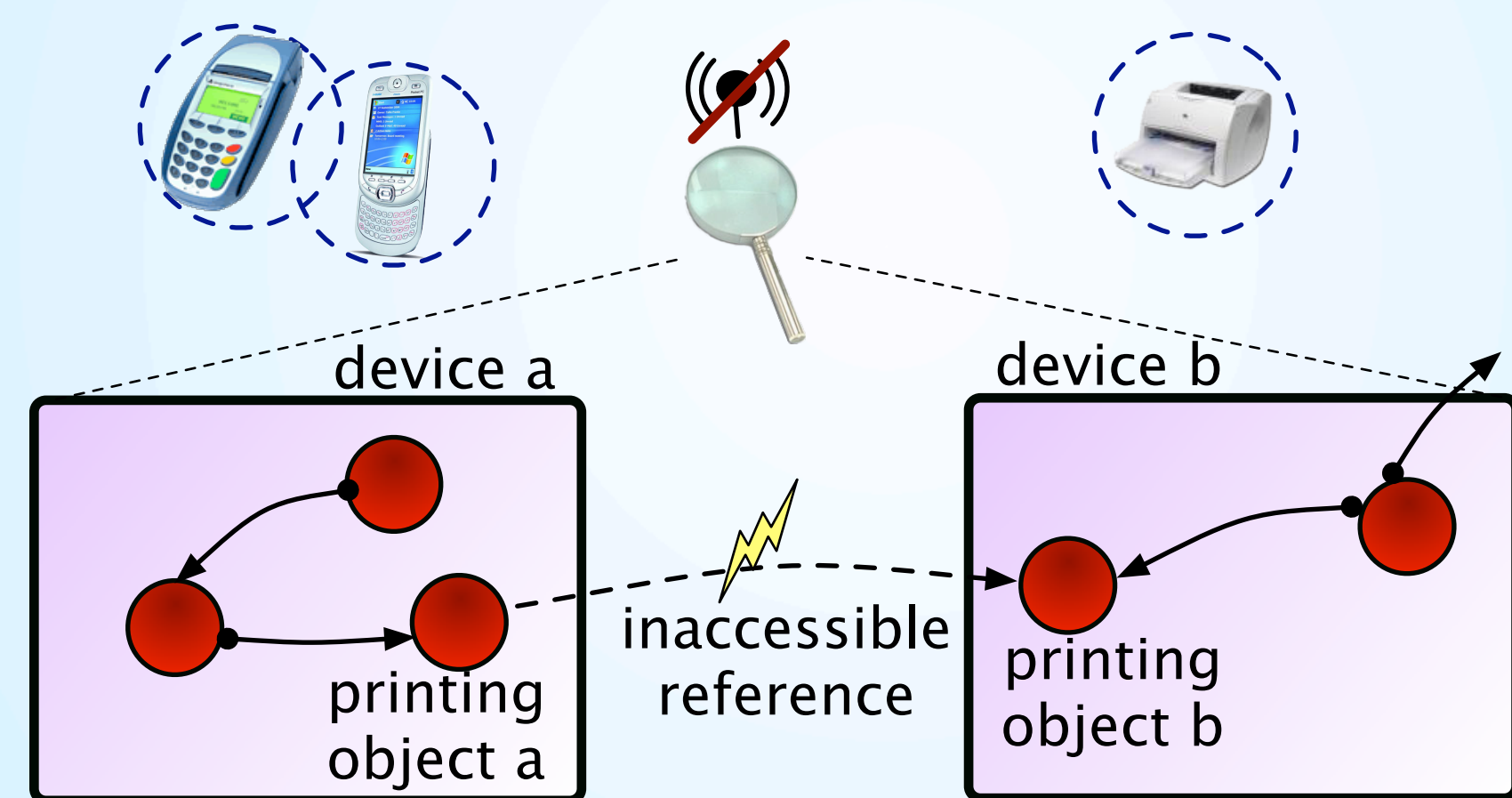


Motivation: DGC in Mobile Networks

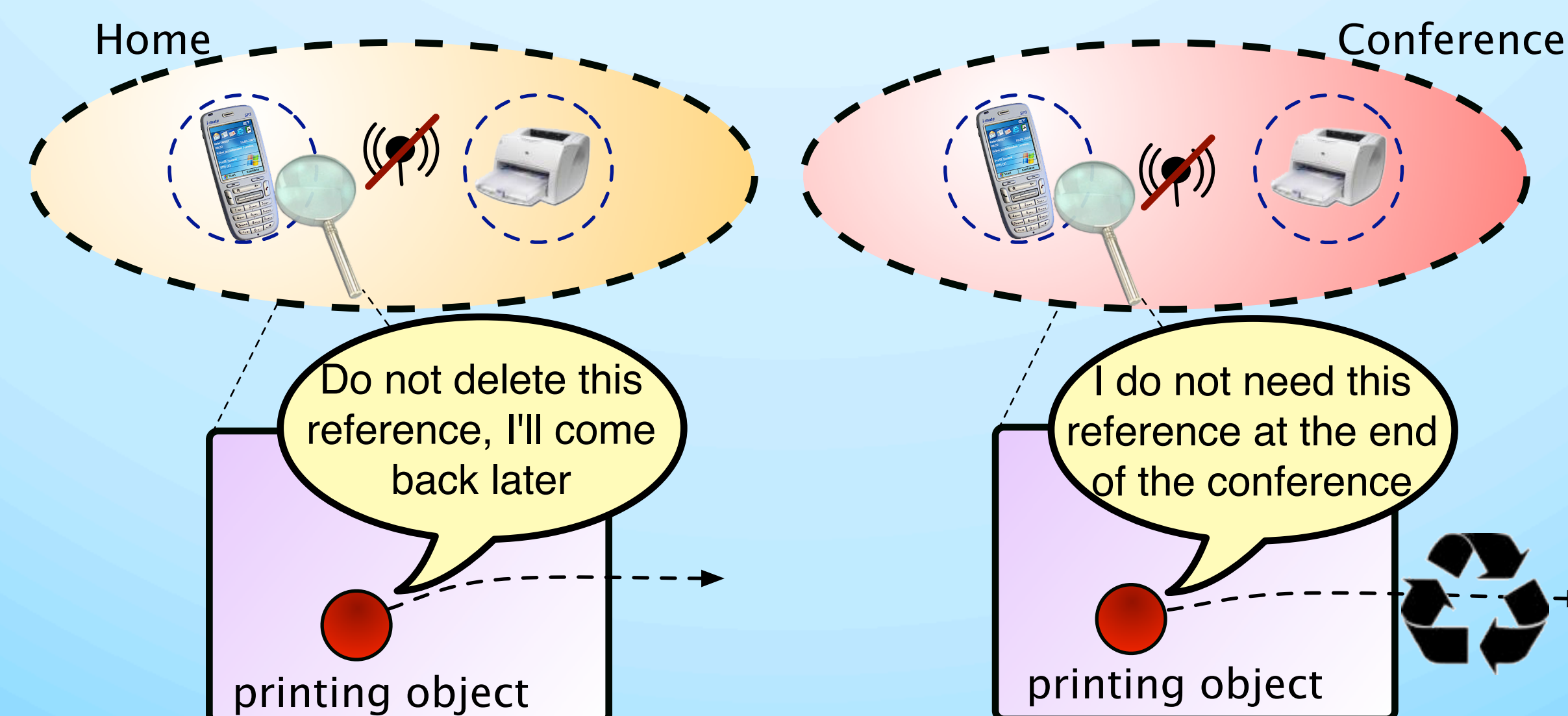


Volatile connections



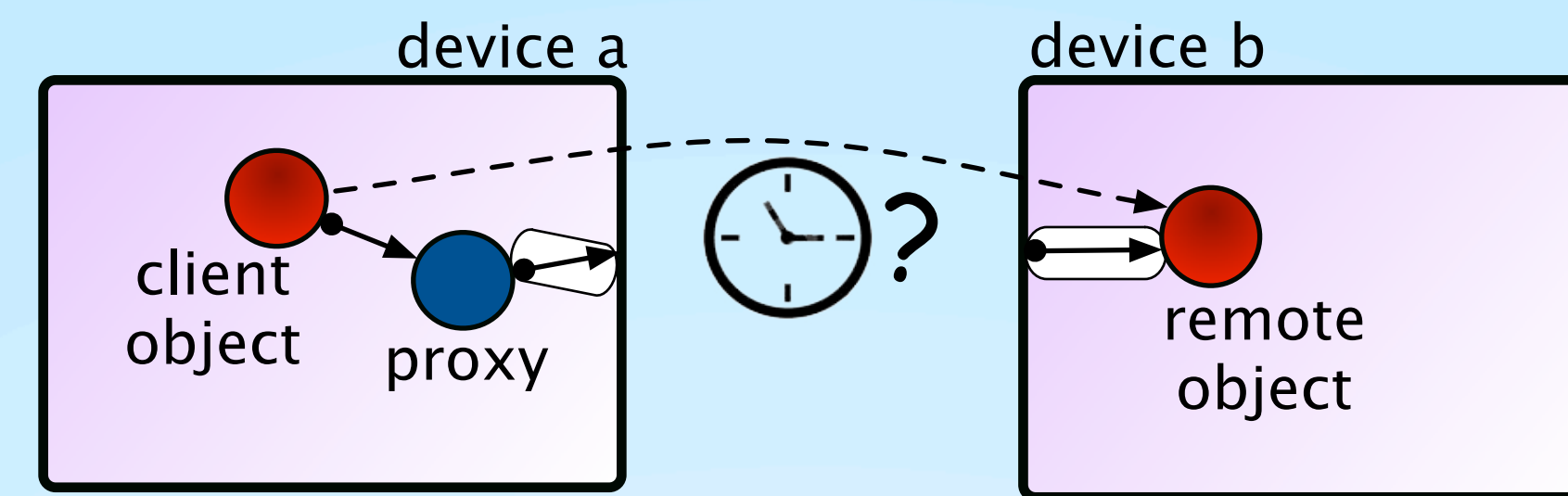
- Limited connectivity of the nodes.
- Inaccessible reference ≠ Broken reference.

Context-dependent information



- Semantics of the application are required to clear objects.

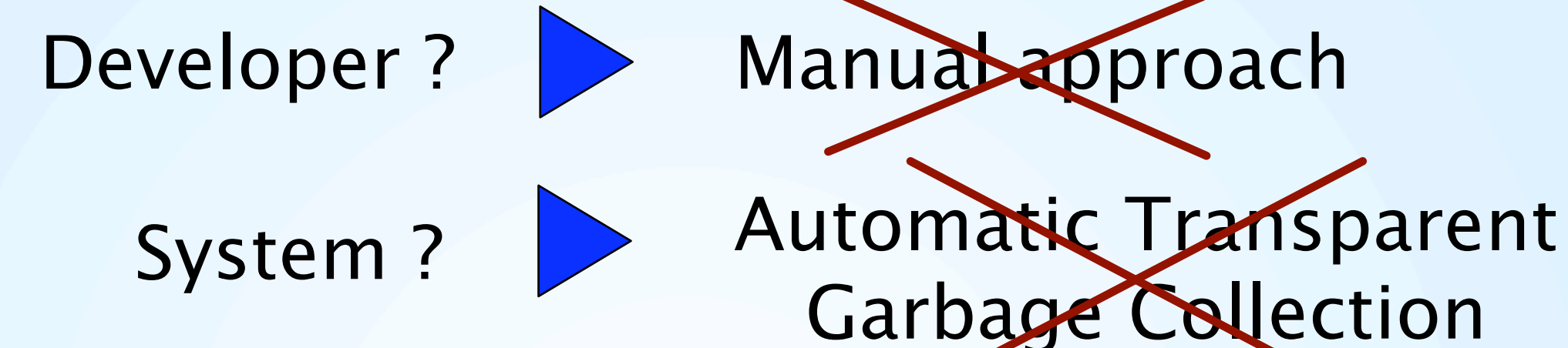
Problem Statement



How long should the system wait for the connectivity of the devices?

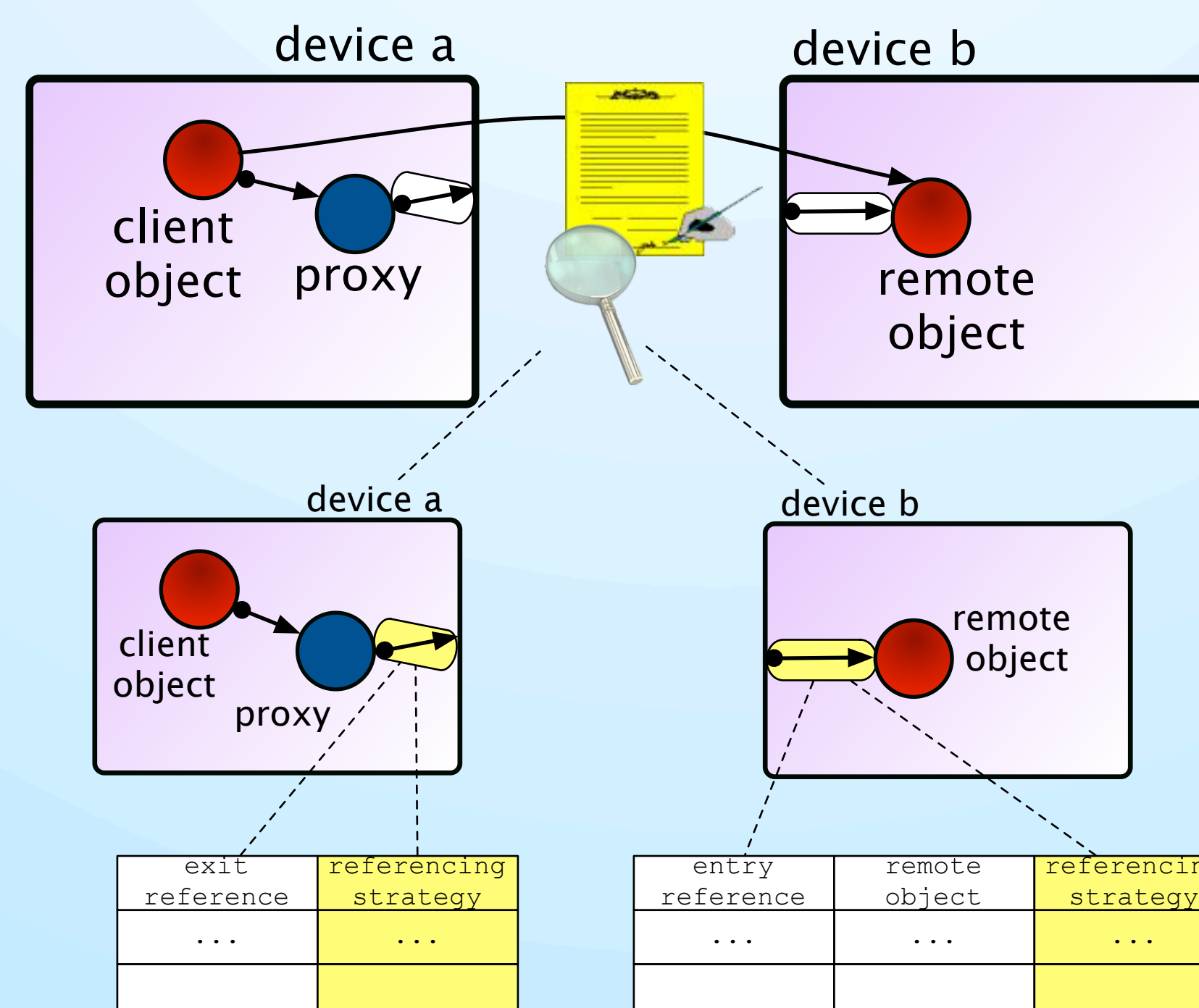
- Application-dependent!
- Even context-dependent!

Who is responsible for garbage collection?



A combination of both:
Semi-Automatic Garbage Collection

- Remote-references as a two-party contract.



- Referencing strategies express collection policies (remote object) + kind of rebinding (client object) upon disconnection.

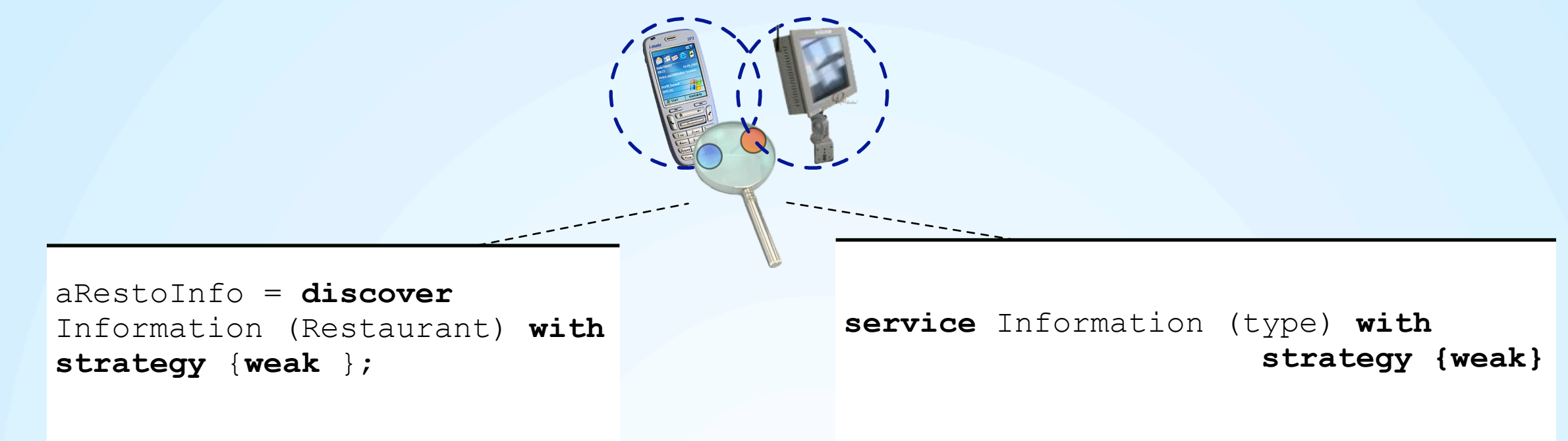
Referencing Strategies

- Language support to apply a collection strategy to the remote reference.
- Express the disposability of a reference upon disconnection:

Temporal disposability

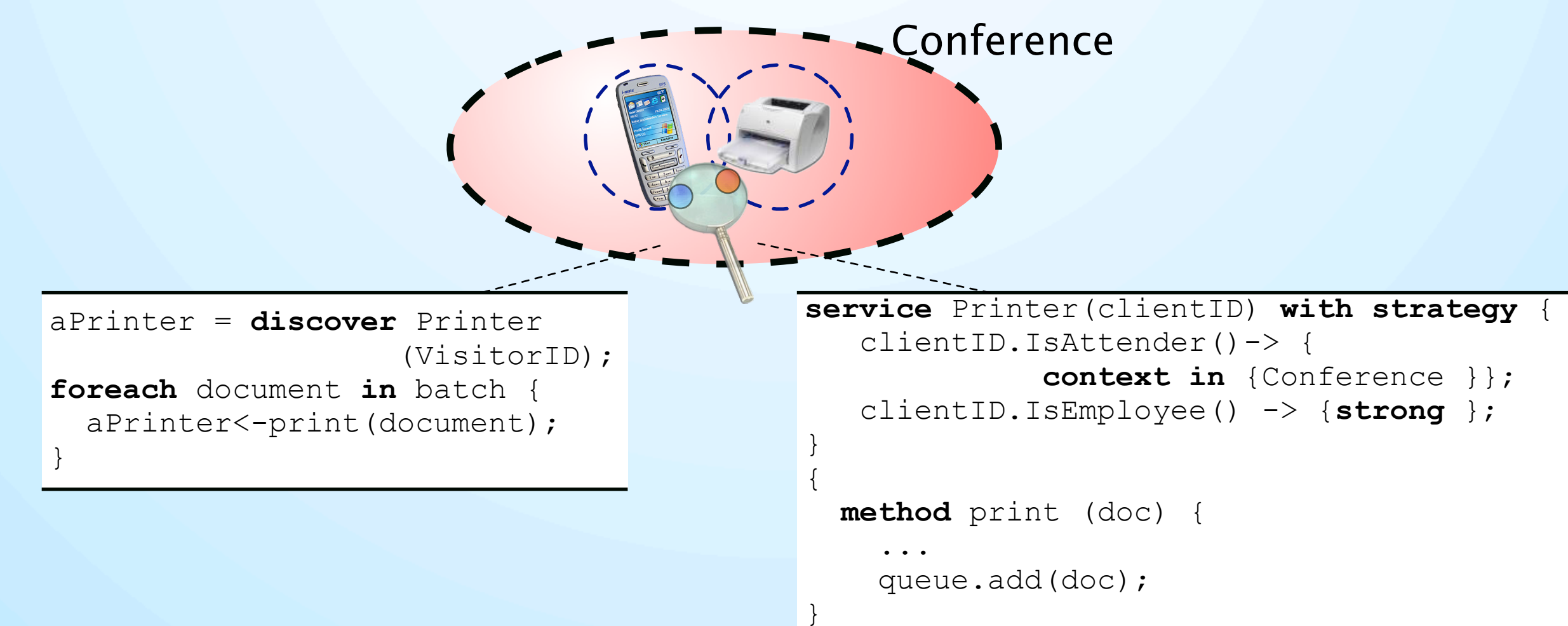
... expressing disposability based on time constraints.

- Weak References: always reclaim.
- Temporal References: reclaim after time period.
- Strong References: never reclaim.



Domain-specific disposability

... expressing disposability based on context information.



... some challenges

- Resolving conflicts.
- Indirect References.
- Language support to annotate groups of remote references.

	weak	temporal	strong
weak	✓	✓	✓
temporal	✗	✓ (t1<t2)	✓
strong	✗	✗	✓

Read more?

[1]Gonzalez Boix E., Van Cutsem T., Mostinckx S., Dedecker J., De Meuter W., D'Hondt T. *Semi-Automatic Garbage Collection for Mobile Networks*. In workshop on Object Technology for Ambient Intelligence and Pervasive Computing (OT4Aml) co-located at the 20th European Conference on Object-Oriented Programming (ECOOP), Nantes, France. 2006.