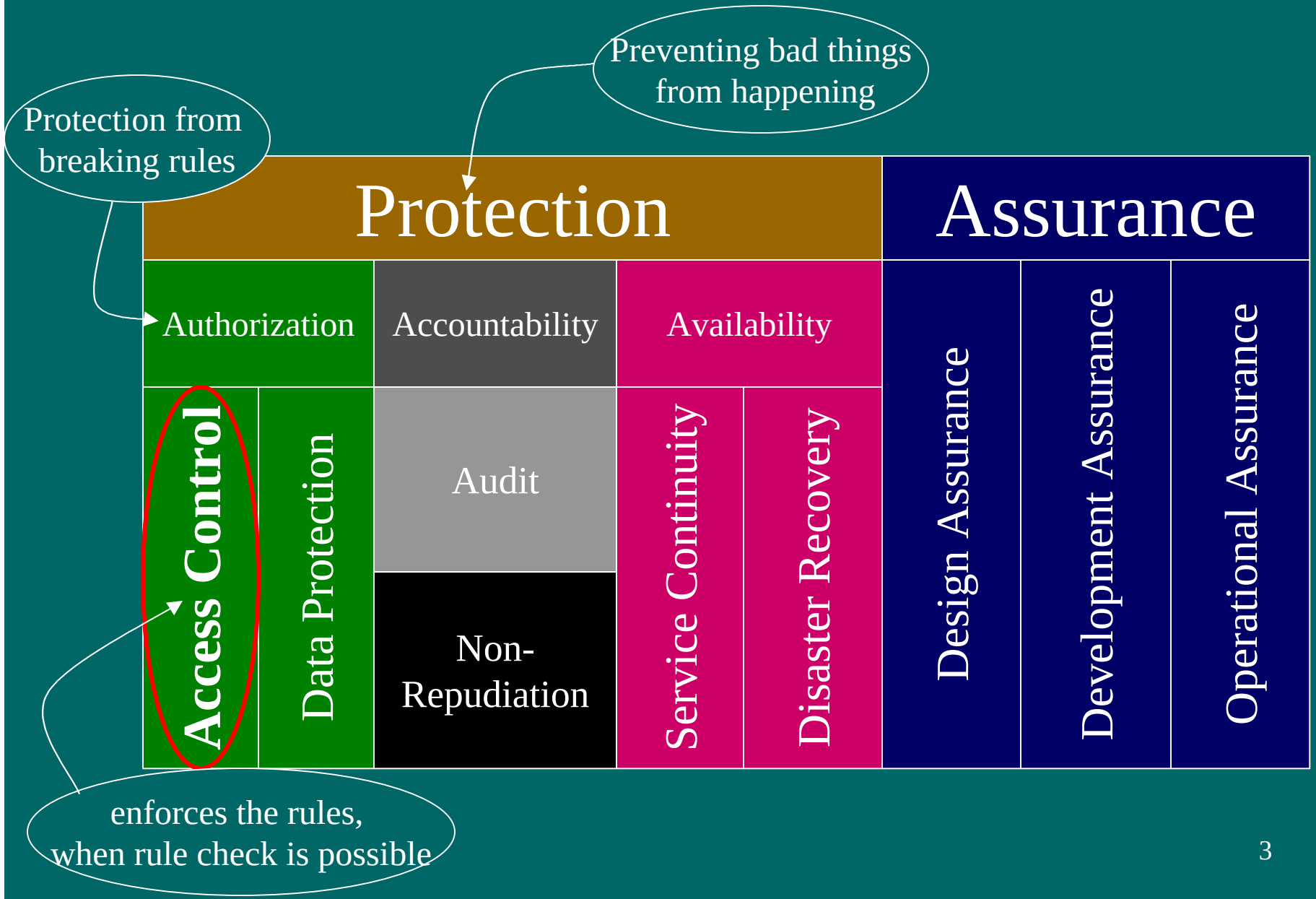
Architectural Separation of Authorization and Application Logic in Distributed Systems

Konstantin Beznosov, Research Associate
Center for Advanced Distributed Systems Engineering
School of Computer Science
Florida International University, Miami
http://www.cs.fiu.edu/~beznosov

Outline

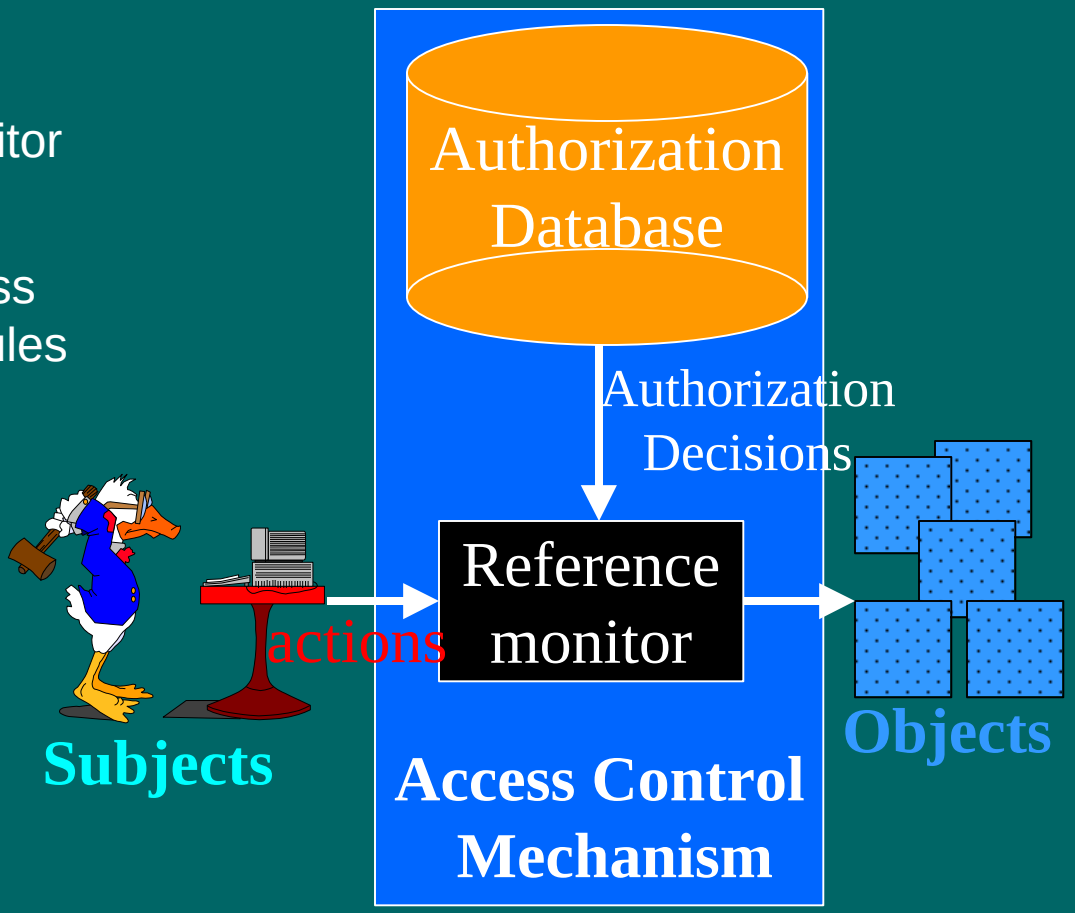
- Introduction
  - Access control
  - Problem Statement
  - Related work
- Resource Access Decision Architecture
- Application Authorization Service
  - Implementation
  - Performance considerations
  - Distributed architecture
- Conclusions

Conventional Computer Security



Access Control

- Access control
  - enforced by a reference monitor
- Authorization
  - concerned with making access control decisions based on rules
  - Rule example
    "subject physician
    can do action read
    on object patient record"



What Is Application Security?

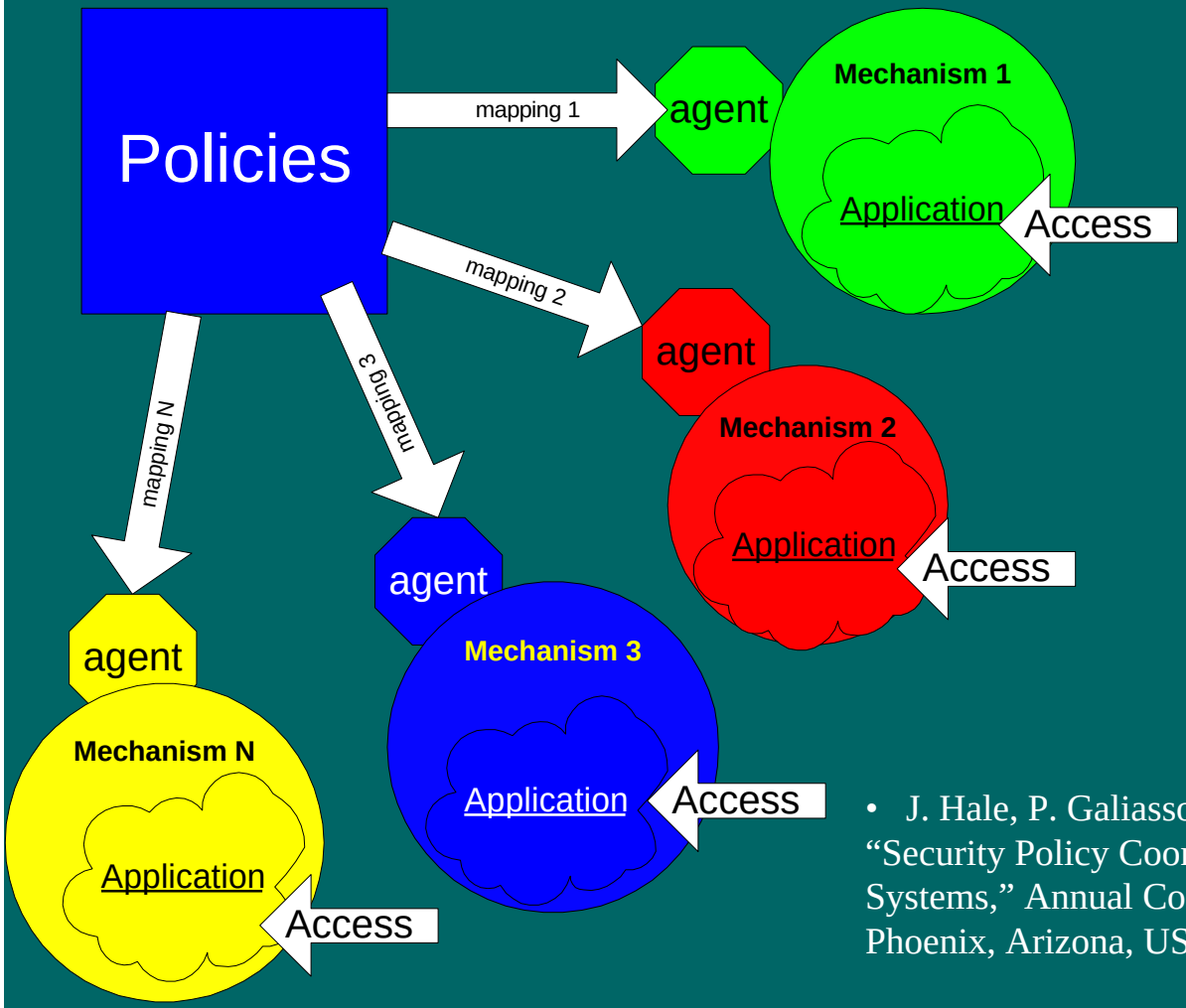
- Complex policies:
  - example

  "subject attending physician can do action read on object current episode sensitive records of the patient"
  - fine grain, domain-specific, dynamic and/or context sensitive. E.g.
    - based on user-patient relationship
    - emergency context
- Need organization-wide enforcement
  - potentially large number of heterogeneous distributed applications and users

Problems of Enterprise Application Security

- Can not be easily handled by existing general purpose security mechanisms
- Largely embedded in application systems today
  - because of
    - need for fine grain access control
    - factors for authorization decision known only to application
- Costly, error-prone multiple points of control
- Expensive life-cycle
- Lack of means to assure organization-wide consistency and end-to-end properties

Approaches to Application Access Control: Policy Agents

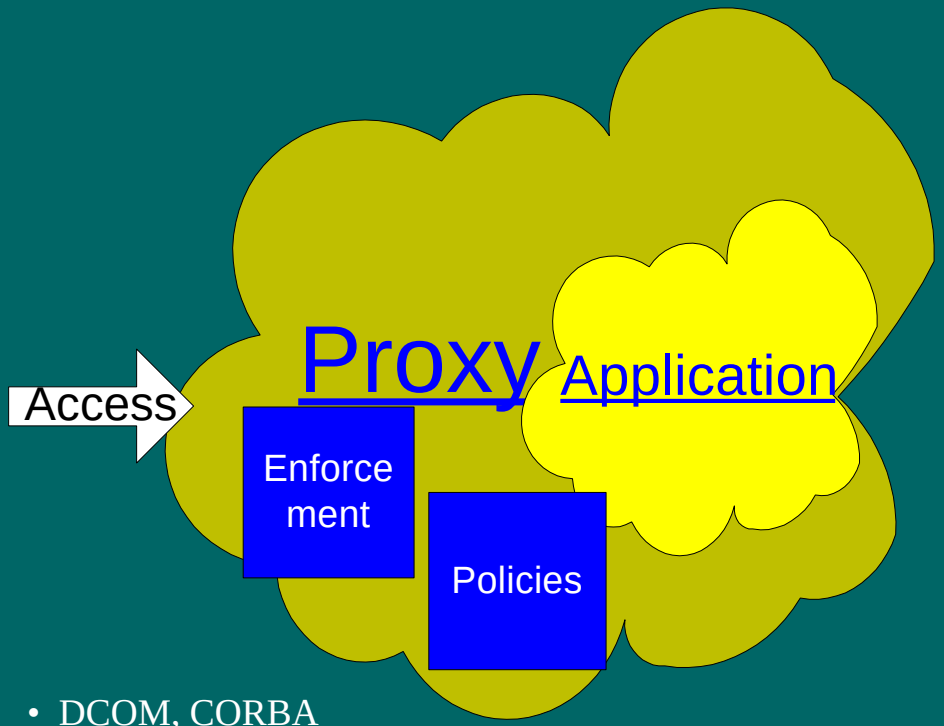


- Accommodates existing body of products and technologies
- Inherent fault tolerance
- Enterprise security is naturally compartmentalized
- Nominal performance overhead
- High degree of run time autonomy

- How to map?!
- Least expressive and most coarse-grain policy supported
- Distribution of policy updates

- J. Hale, P. Galiasso, M. Papa, and S. Shenoi, "Security Policy Coordination for Heterogeneous Information Systems," Annual Computer Security Applications Conference, Phoenix, Arizona, USA, 1999.

Approaches to Application Access Control: Proxies

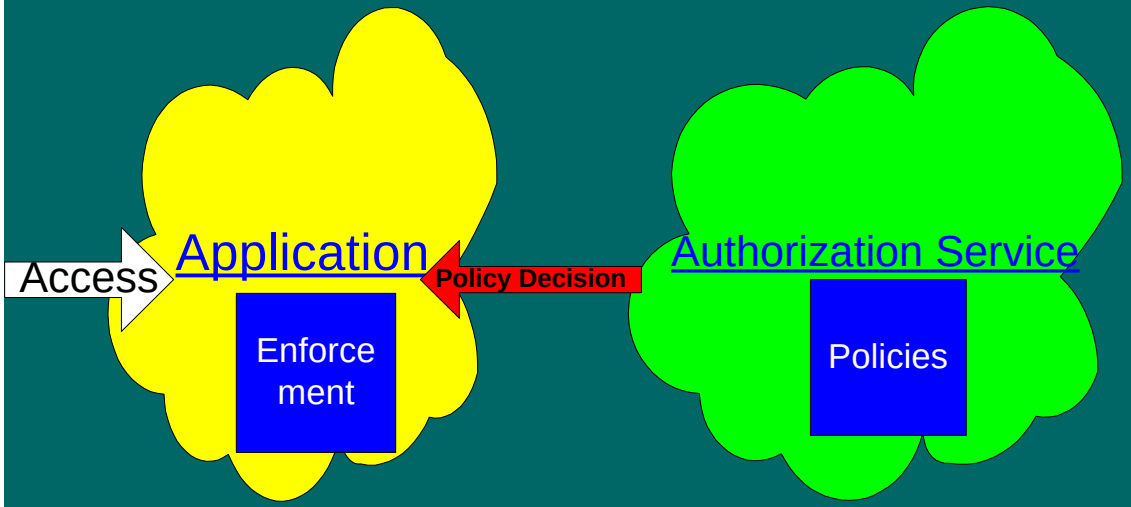


- No changes to an application system
- External enforcement
- Reference monitor size is controlled

- Coarse granularity of access control
- Decisions and enforcement outside of application
- No application-specific enforcement
- Policy and authorization data consistency?

- DCOM, CORBA
- B. Hailpern and H. Ossher, "Extending Objects to Support Multiple Interfaces and Access Control," *IEEE Transactions on Software Engineering*, vol. 16, pp. 1247-1257, 1990.
- J. Barkley, "Implementing Role-based Access Control Using Object Technology," The First ACM Workshop on Role-Based Access Control, Fairfax, Virginia, USA, 1995.
- R. Filman and T. Linden, "SafeBots: a Paradigm for Software Security Controls," New Security Paradigms Workshop, Lake Arrowhead, CA USA, 1996.
- C. W. W.A. Wulf, and D. Kienxle, "A new model of security for distributed systems," 1995.
- T. Riechmann and F. J. Hauck, "Meta Objects for Access Control: A Formal Model for Role-based Principals," New Security Paradigms Workshop, 1998.

Authorization Services: A Solution to Enterprise Application Security



- Access control decisions external to application
- Logically centralized administration of enterprise wide policies
- Simplified application development
- Any level of granularity
- Easy policy changes and updates
- Just is time authorization decisions

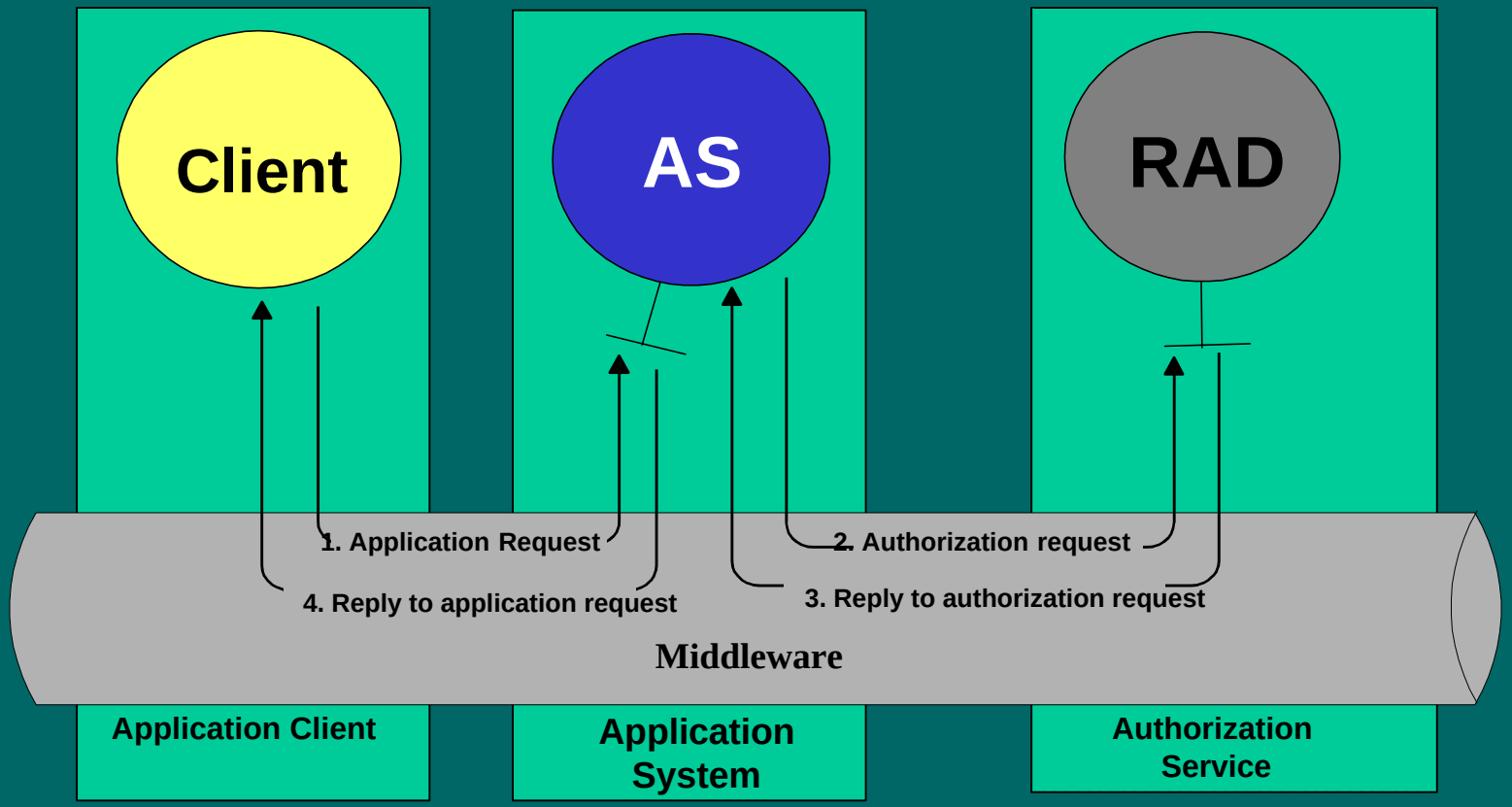
- Application system part of reference monitor
- Performance, fault tolerance, scalability, resource representation

- *V. Varadharajan and C. C. a. J. Pato*, "Authorization in Enterprise-wide Distributed System: A Practical Design and Application," 14th Annual Computer Security Applications Conference, 1998.
- T. Y. C. Woo and S. S. Lam, "Designing a Distributed Authorization Service," IEEE INFOCOM, San Francisco, 1998.
- R. Simon and M. E. Zurko, "Adage: An Architecture for Distributed Authorization," OSF Research Institute, Cambridge 1997.

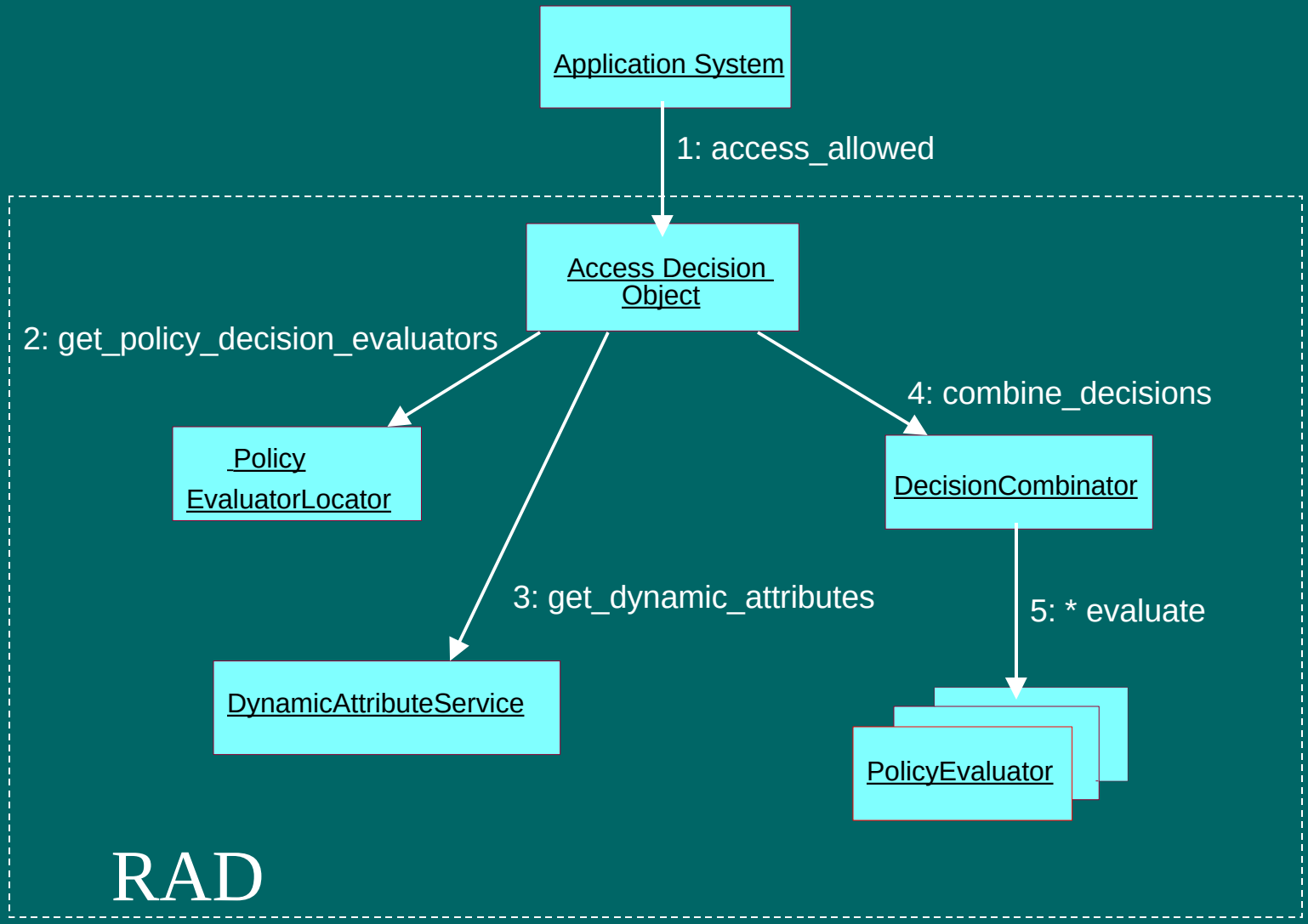
Resource Access Decision Service
Architecture Objectives

- Decouple authorization and application logic
- Generic
- Fine-grain resources
- Use of underlying middleware and its security
- Existing authorization mechanisms
- Policy-neutral
- Minimum application involvement
- Multi-policy systems
- Request-specific and dynamic factors
- Co-existence of parts from different vendors

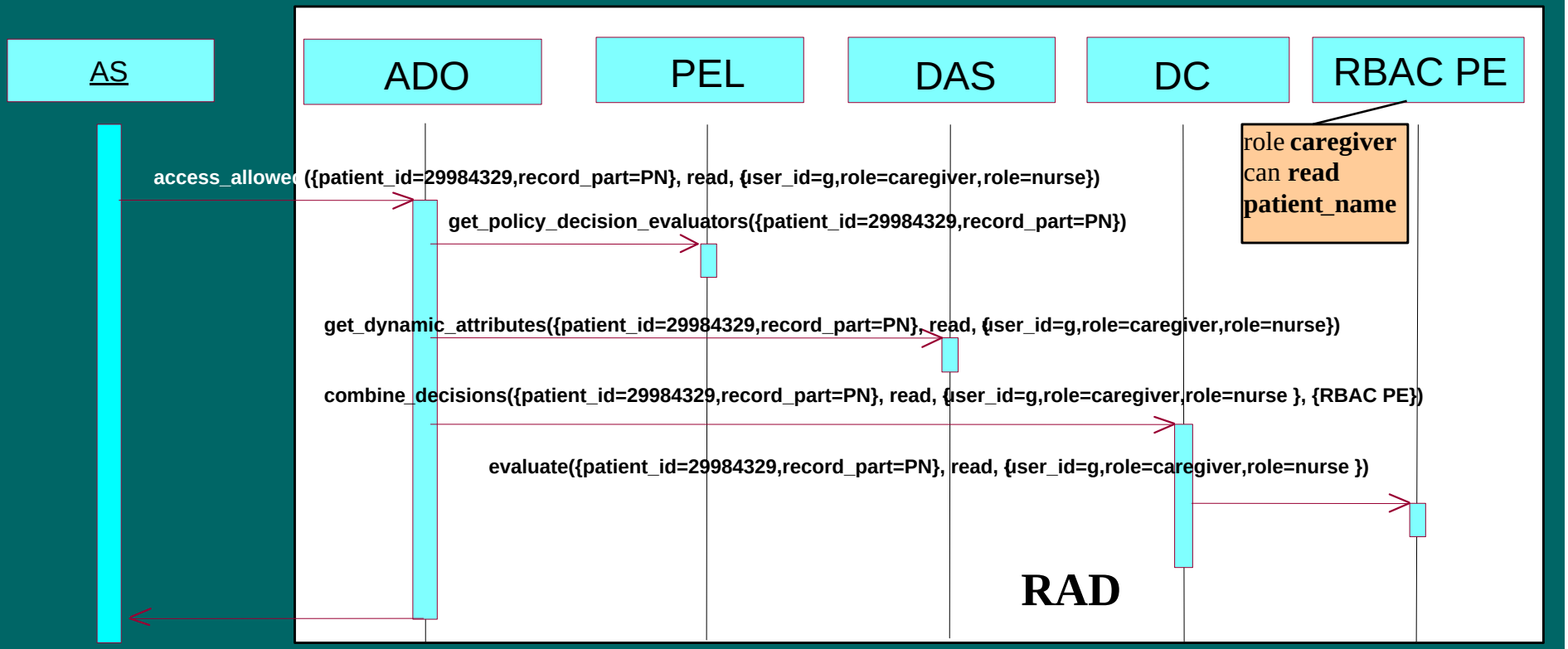
Resource Access Decision (RAD): External View



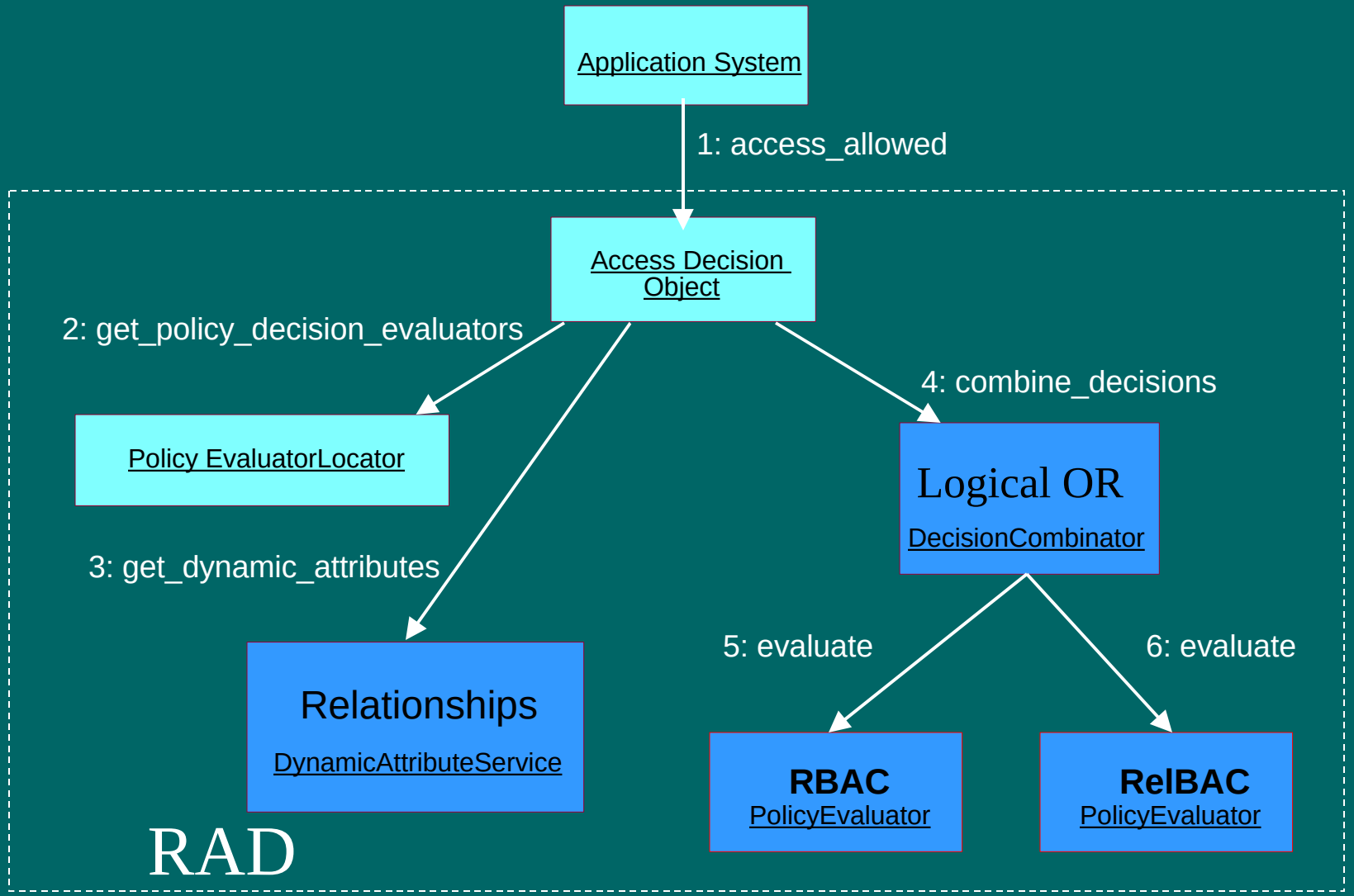
RAD Architecture



Walk Through



RAD Configuration Example



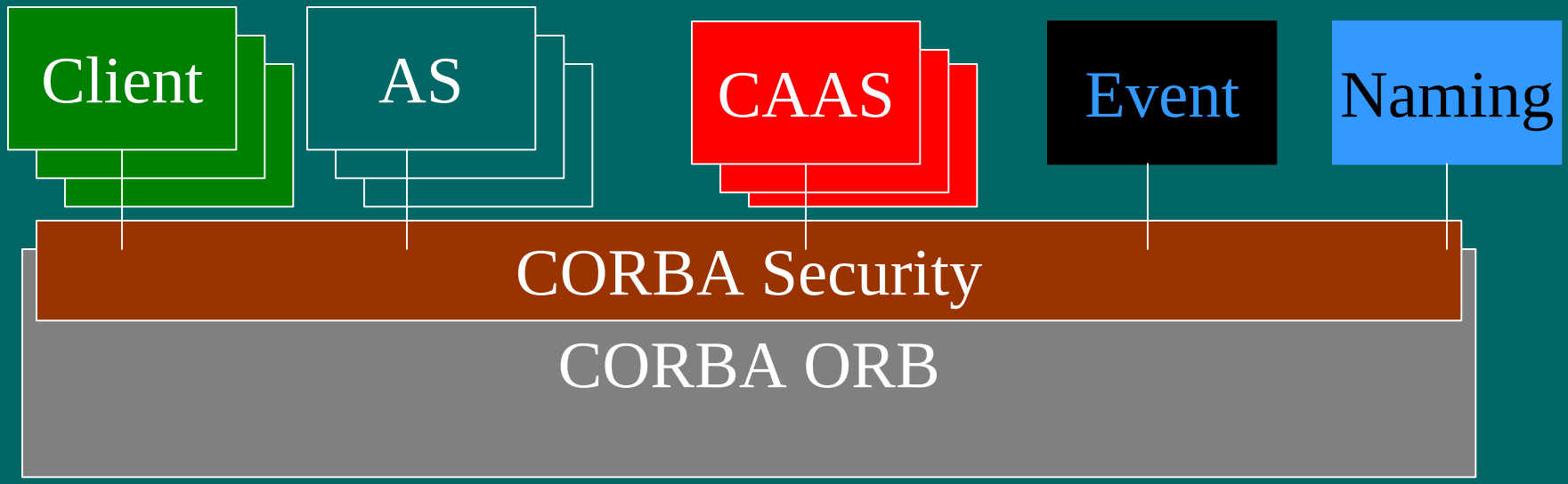
RAD Features and Issues

- Features:
  - Centralized administration of access control mechanisms
  - Dynamic change of access control policies
  - Independent development and evolution of application and authorization services
  - Policy-neutral by encapsulating policy evaluation in independent policy evaluators
  - Support for domain/request-specific factors
- Major issues
  - Performance in terms of response time
  - Policy modeling

CORBA-based Application Authorization Service (CAAS)

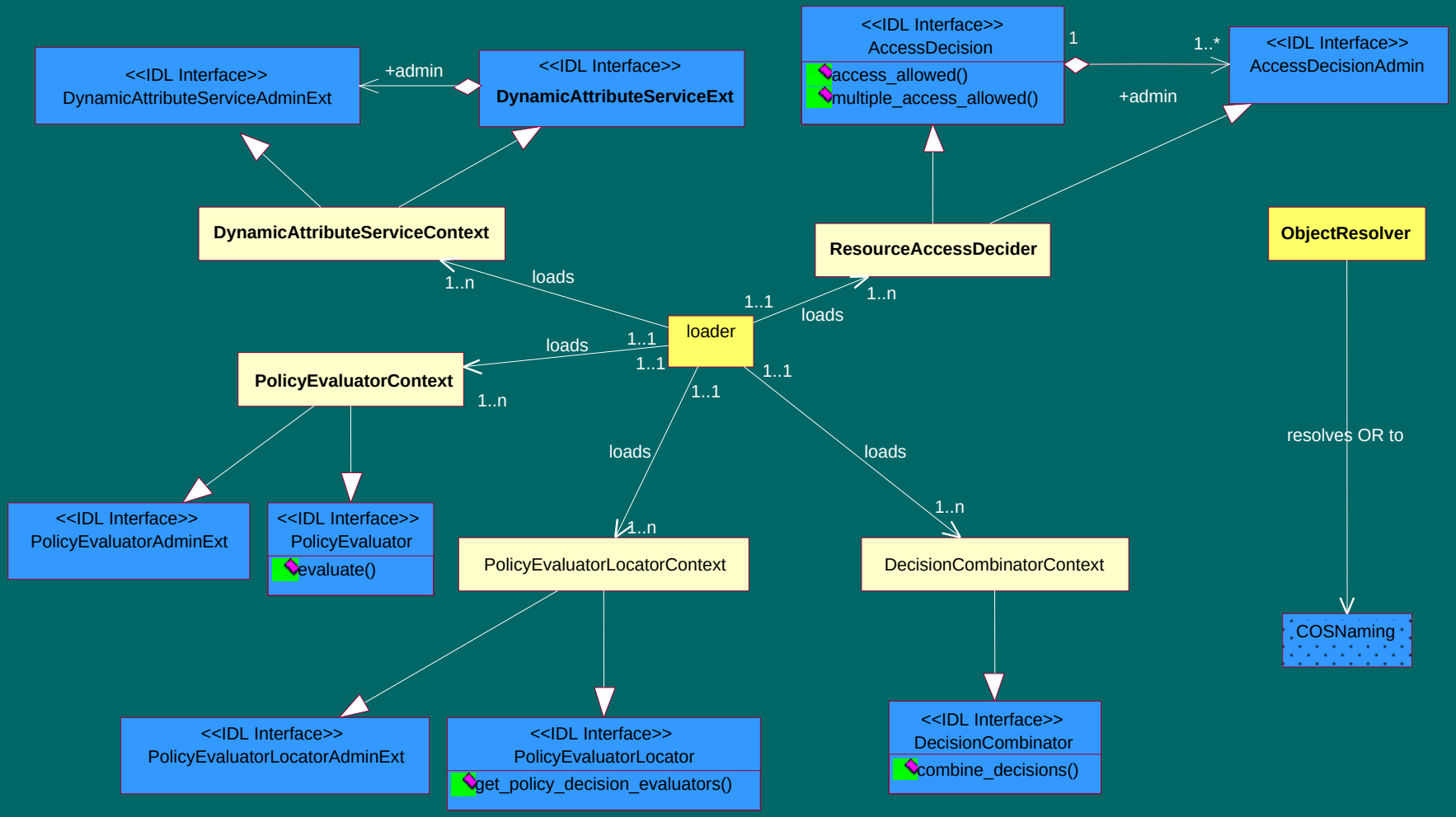
- Test-bed for research on RAD:
  - Performance
  - Policy Modeling
- Requirements
  - Re-configurable
  - Easy to implement
  - Portable to different platforms
  - First step towards future research

CAAS: CORBA-based Application Authorization Service

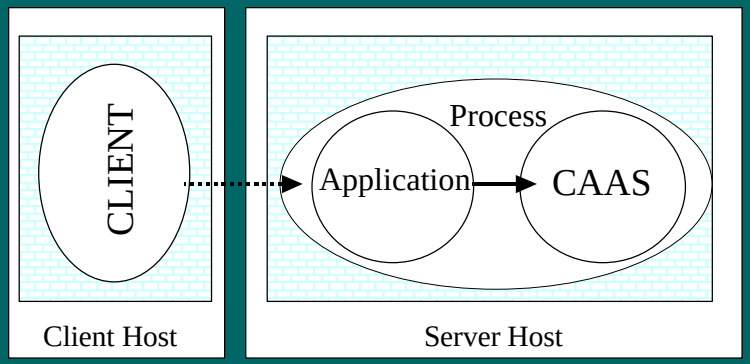


- Compliant with RAD
- Defined in OMG IDL
- Implemented in Java
- ORB-independent

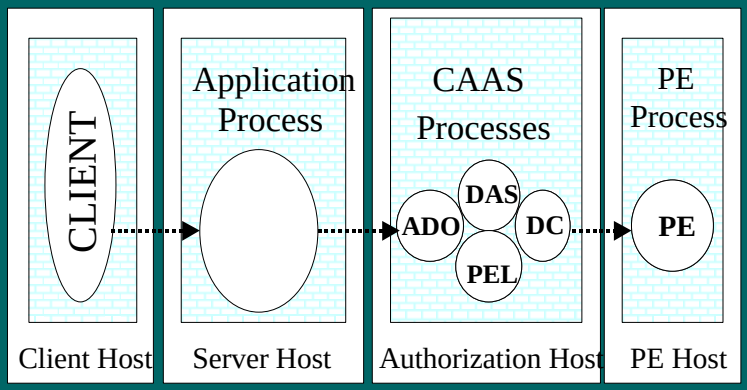
CAAS Architecture



CAAS: Highly Configurable



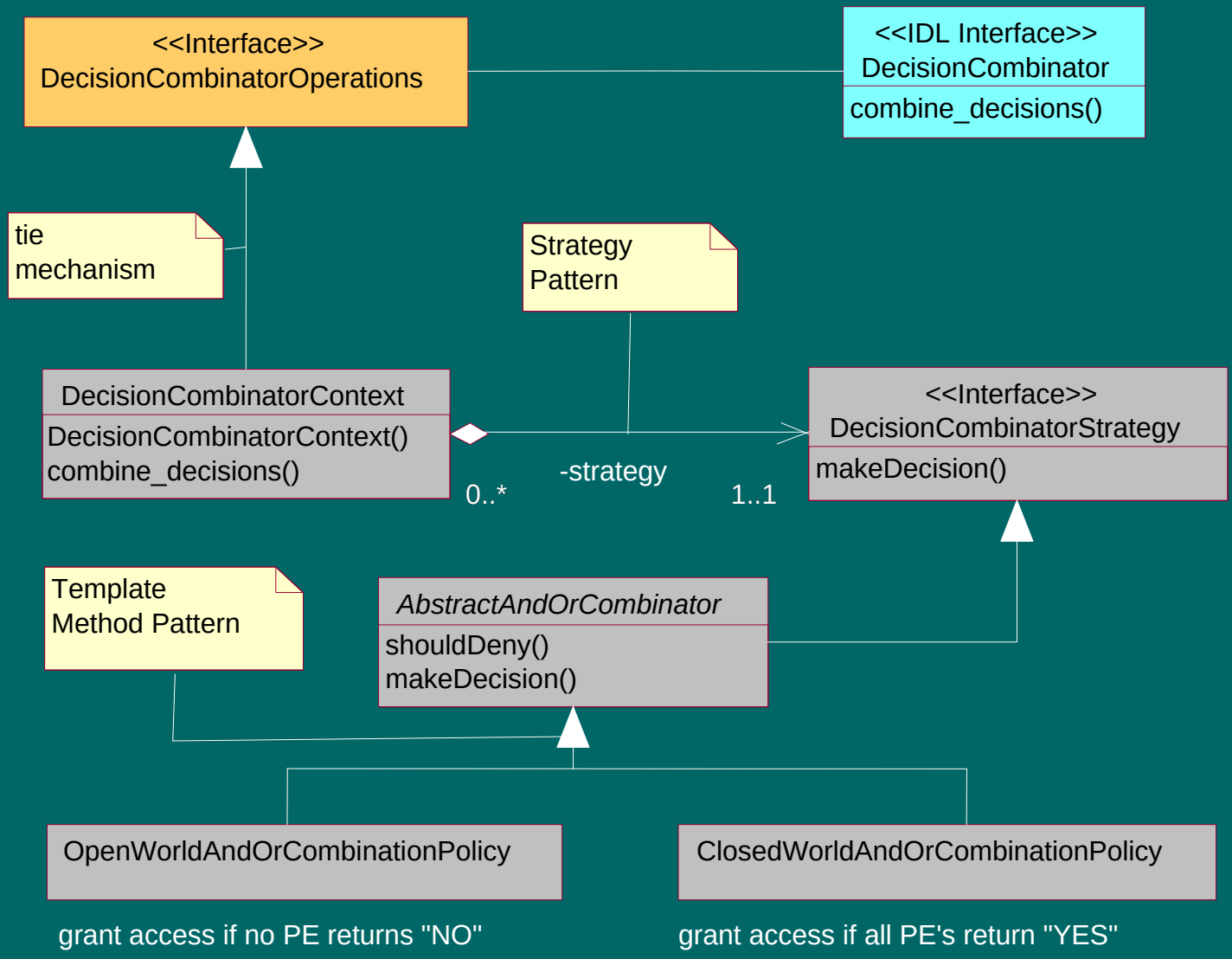
Reference Configuration



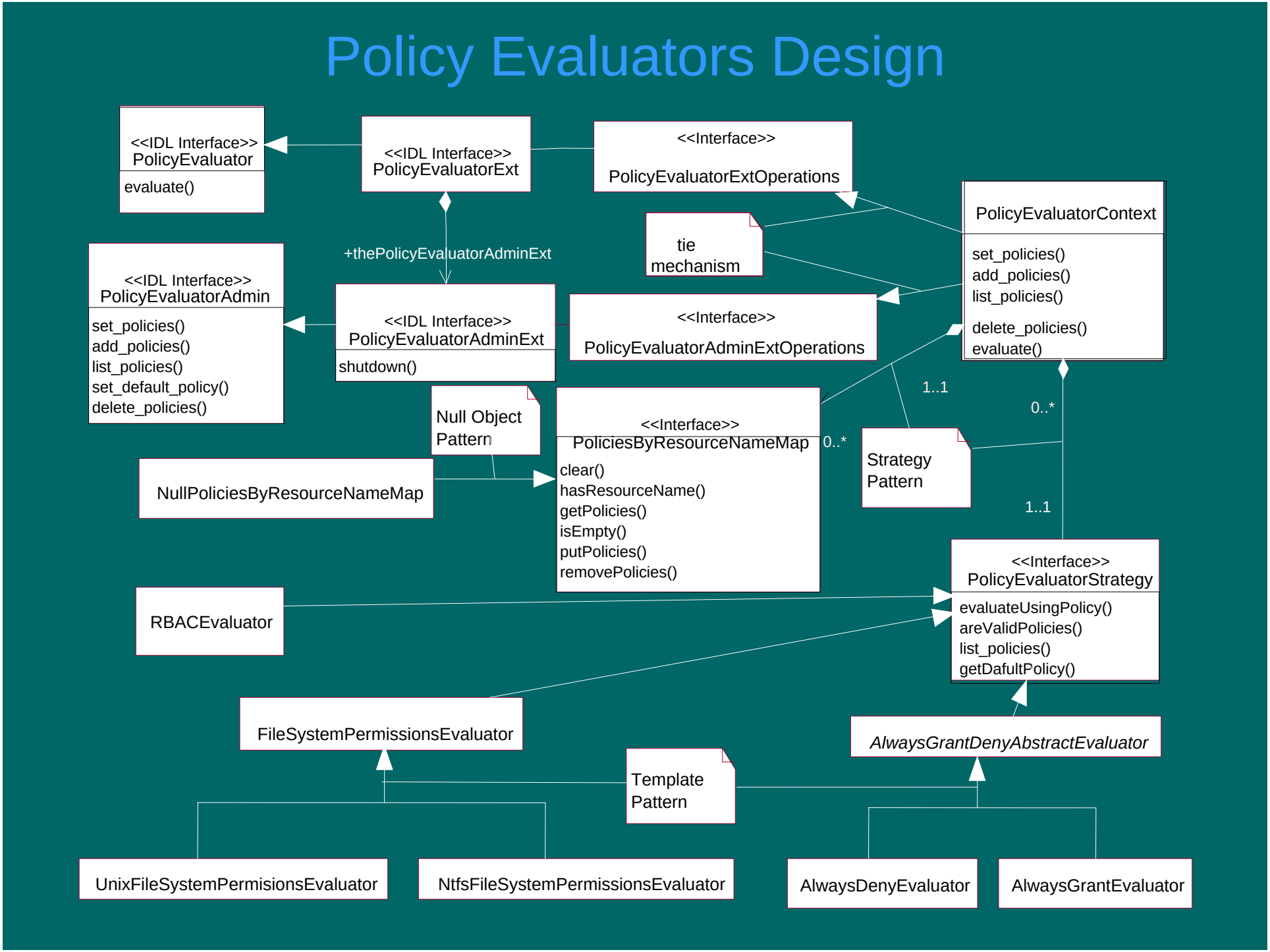
Host/Process/PE-Host Configuration

- Changeable and portable
  - e.g. provides both run-time interface for authorization and administrative interface for configuring CAAS components
- Supports different types of policies
  - federations, multi-policy, relationship-based access control (RelBAC)
- For details: http://cadse.cs.fiu.edu/research_projects/RAD

CAAS: Use of Design Patterns

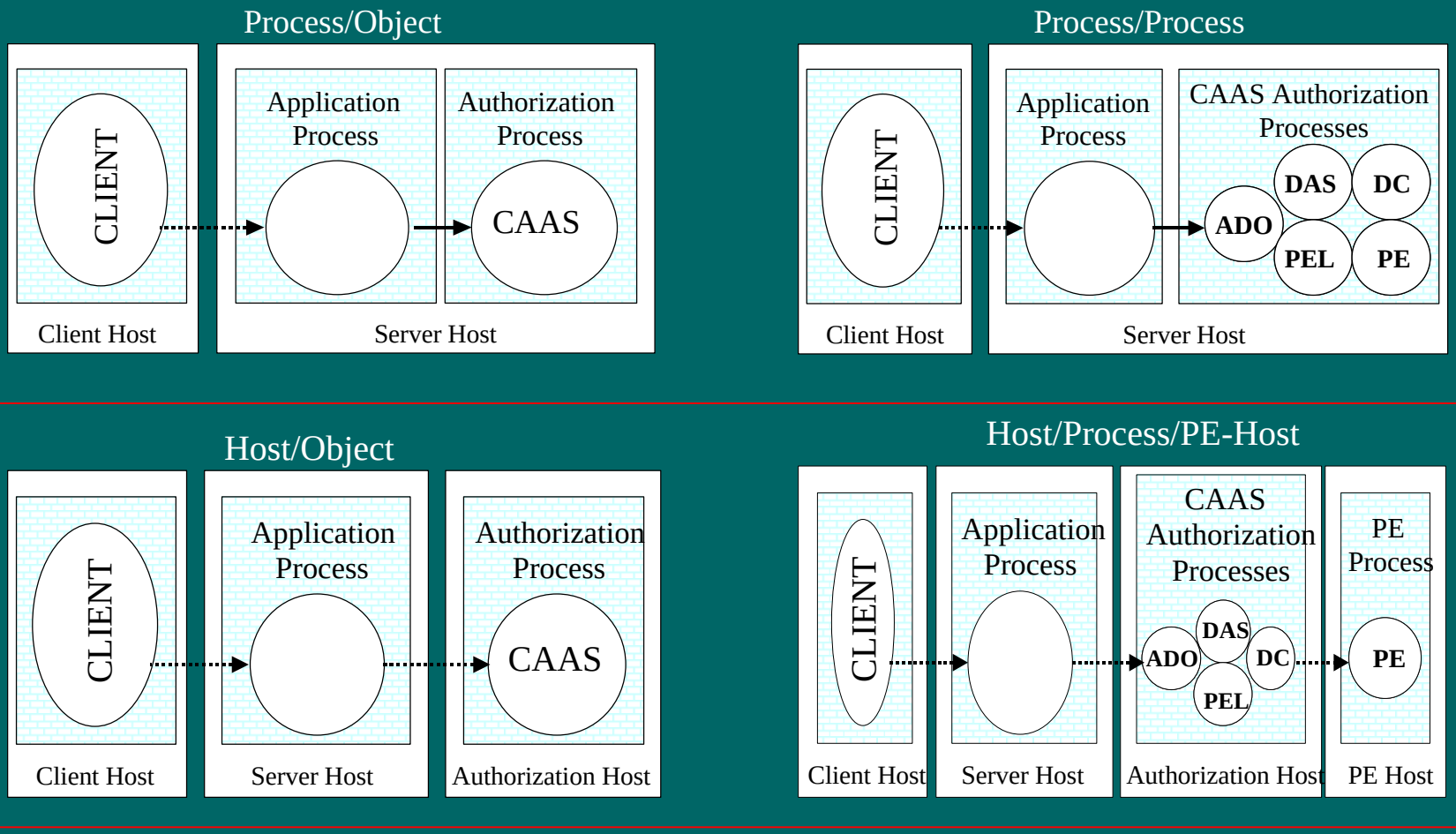


Policy Evaluators Design

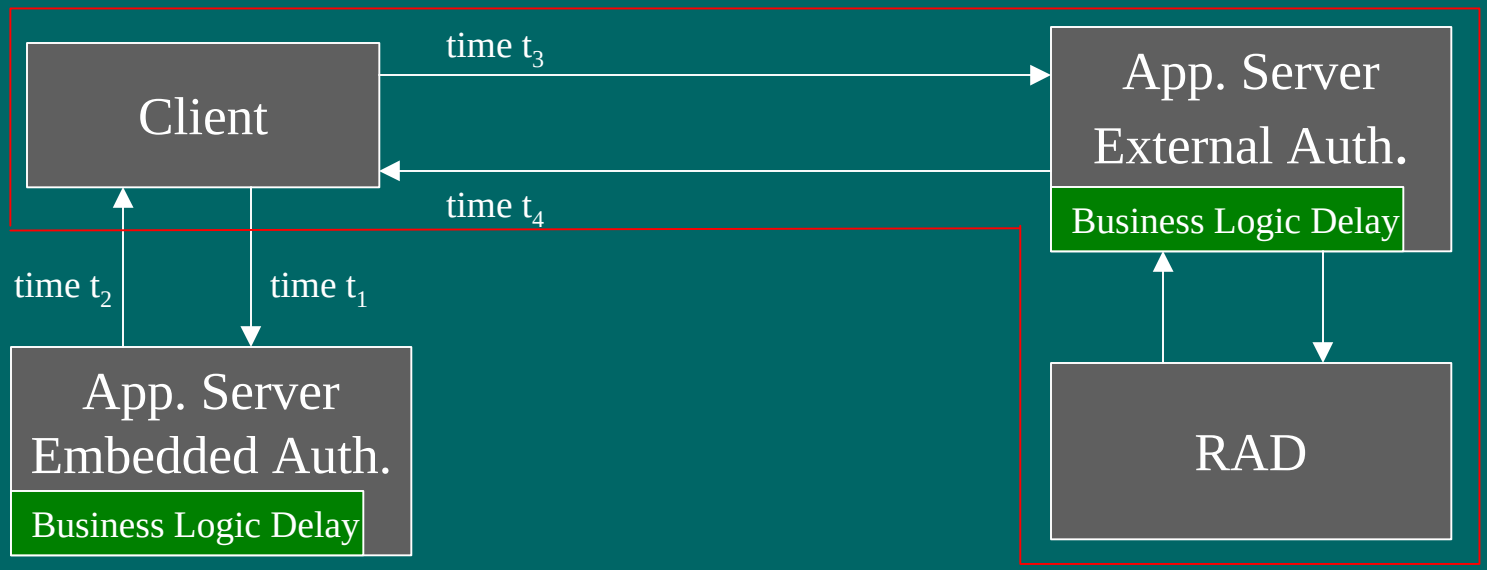


Configurations for Performance Test

Boundaries crossed: Application -> RAD/RAD Components
Host=ORB+network; Process=ORB+process; Object=function call

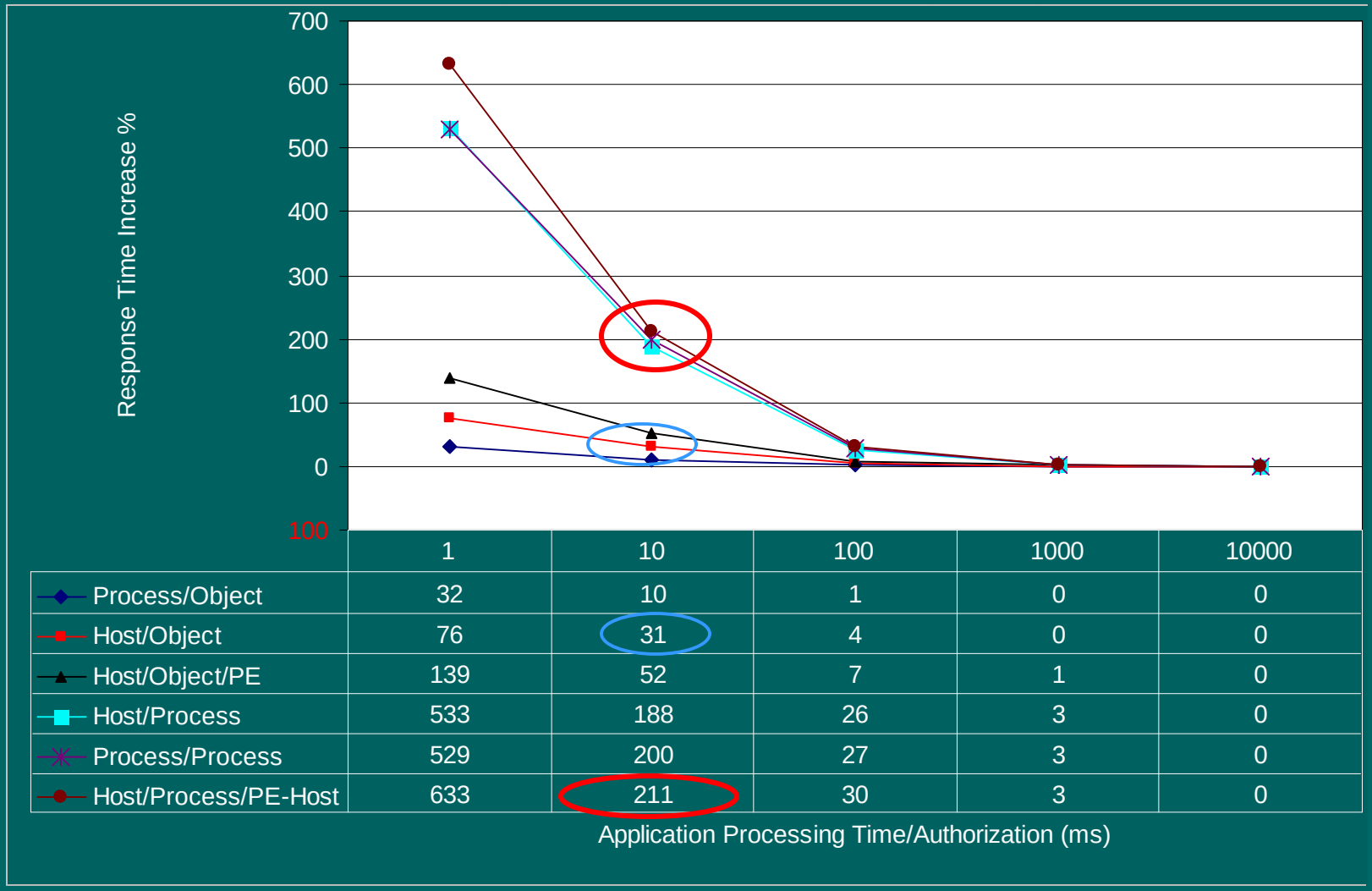


Conducting Performance Measurements



- Measure response time perceived by the client: $T_{emb} = (t_2 - t_1)$ and $T = (t_4 - t_3)$.
- Measure response time increase $I = (T \% T_{emb} - 1) * 100$
- Repeat for 1ms, 10ms, 100ms, 1sec, 10sec business logic delays.
- Repeat for 1, 10, 100, 1000 authorization requests.
- Repeat for different configurations.
- Conduct measurements under low network load (< 1%)

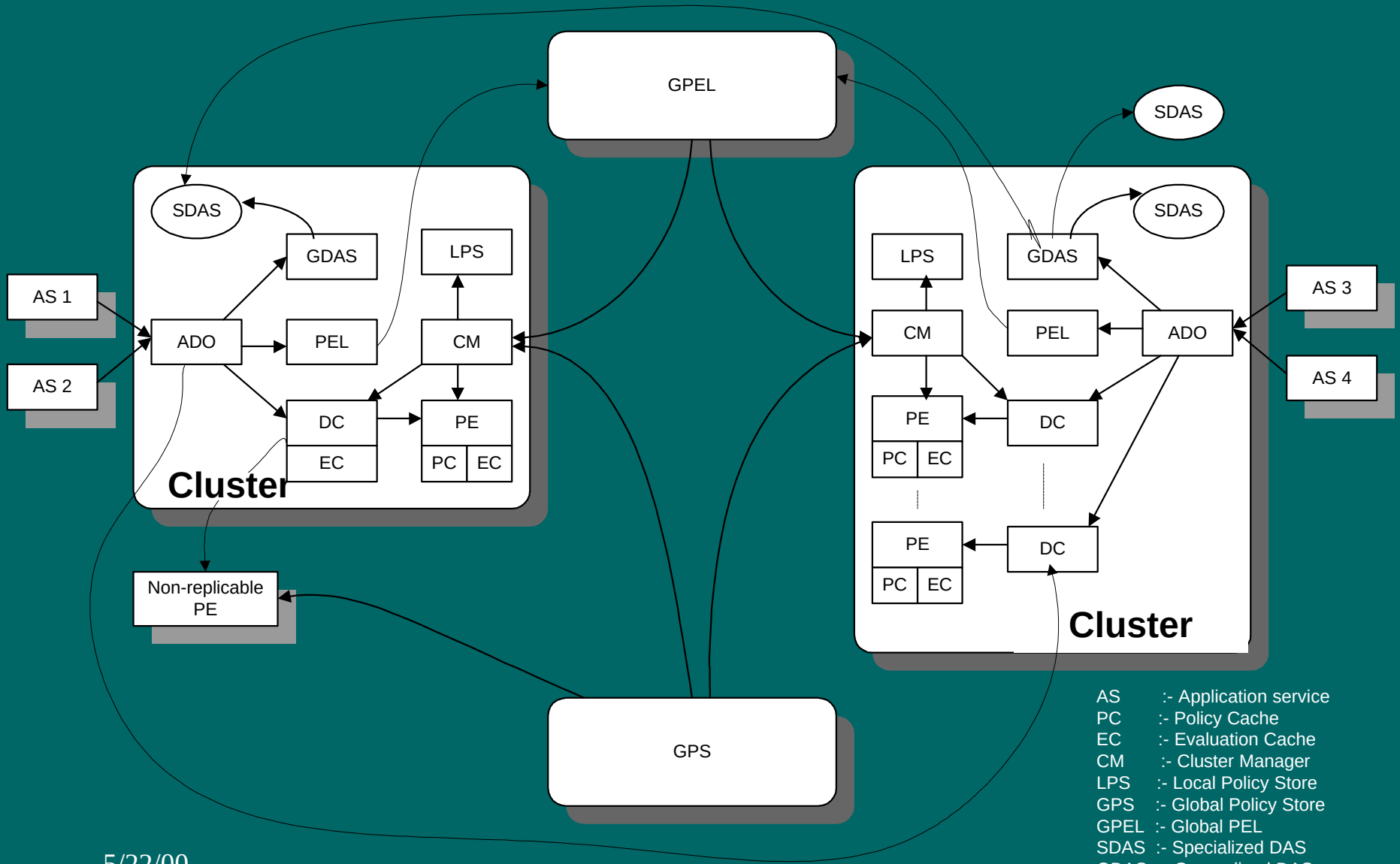
Performance Evaluation of CAAS



	1	10	100	1000	10000
Process/Object	32	10	1	0	0
Host/Object	76	31	4	0	0
Host/Object/PE	139	52	7	1	0
Host/Process	533	188	26	3	0
Process/Process	529	200	27	3	0
Host/Process/PE-Host	633	211	30	3	0

Application Processing Time/Authorization (ms)

Distributed AAS architecture



AS :- Application service
PC :- Policy Cache
EC :- Evaluation Cache
CM :- Cluster Manager
LPS :- Local Policy Store
GPS :- Global Policy Store
GPEL :- Global PEL
SDAS :- Specialized DAS
GDAS :- Generalized DAS

Next Steps

- Distributed AAS architecture
  - Configurability
    - dynamic policy changes support
    - support for different distributed (e.g. healthcare and Internet based e-commerce) environments
  - Adequate performance (distributed authorization and load balancing)
  - High availability (replication and fault tolerance)
  - Application composibility
- Case study
  - Real life policies in healthcare (HIPAA)
  - Sample application(s)
  - Workload and scenario simulation

Contributions and Publications

- Analysis of requirements for access control in US healthcare domain
  - K. Beznosov, "**Issues in the Security Architecture of the Computerized Patient Record Enterprise**," Second Workshop on Distributed Object Computing Security, Baltimore, Maryland, USA, 1998.
  - K. Beznosov, "**Requirements for Access Control: US Healthcare Domain**," Third ACM Workshop on Role-Based Access Control, 1998.
- Modeling of RBAC in CORBA access control
  - K. Beznosov and Y. Deng, "**A Framework for Implementing Role-based Access Control Using CORBA Security Service**," Fourth ACM Workshop on Role-Based Access Control, Fairfax, Virginia, USA, 1999.
- Introduction of relationships in access control and outlining implementation
  - J. Barkley, K. Beznosov, and J. Uppal, "**Supporting Relationships in Access Control Using Role Based Access Control**," Fourth ACM Role-based Access Control Workshop, Fairfax, Virginia, USA, 1999.
- Application-level access control
  - K. Beznosov, Y. Deng, B. Blakley, C. Burt, and J. Barkley, "**A Resource Access Decision Service for CORBA-based Distributed Systems**," Annual Computer Security Applications Conference, Phoenix, Arizona, USA, 1999.
  - OMG, "**Resource Access Decision Facility**," Object Management Group OMG document number: corbamed/99-05-04, May 1999.
  - K. Beznosov, L. Espinal, and Y. Deng, "**Performance Considerations for CORBA-based Application Authorization Service**," PODC Middleware Symposium (pending acceptance), 2000.
  - L. Espinal, K. Beznosov, and Y. Deng, "**Design Considerations for CORBA-based Application Authorization Service**," In Proceedings of National Information Systems Security Conference (pending acceptance), 2000.