

# **ISSUES TO CONSIDER IN BUILDING SECURE COMPUTER NETWORK DEFENSE SYSTEMS**

## **NuParadigm Government Systems**

The entire contents of this Presentation are a copyrighted work owned by NuParadigm Government Systems, Inc. Distribution or use in whole or in part without express written consent of NuParadigm Systems is prohibited. Copyright - NuParadigm Government Systems, Inc. - 2004 This presentation contains confidential proprietary information owned by NuParadigm and should be returned or destroyed if you are not the intended recipient or are unable to keep it confidential.



# **Distributed Applications in a Web Services Environment Create Opportunity but present New Challenges:**

- Asynchronous non-deterministic network behavior
- Object and/or message driven dependencies
- Security/Information Assurance
- Complicated provisioning/deployment when distributed
- Flat peering of control domains



# Distributed Applications in a Web Services Environment Create

## Opportunity but present New Challenges:

- Modeling, simulation and testing are complex
- Life cycle development spreads across multiple modules/nodes with differing completion levels complicating the testing and certification of app
- Centralized versus distributed management
- Shared services and orthogonal use cases create chaos and noise in the mixed domains/communities/applications
- Inherited behavior



# More Specific Problems in GIG and NCES Oriented Programs:

- Overwhelming Scale
- Interoperability and Rational Legacy Integration N-Square Permutations
- Cross Boundary Security - Traditional Firewalls are not Content Aware
- Ad-Hoc bi-directional Connectivity between users, domains, and systems
- IDM: Persistence, Asynch, Push, Pull, Smart Pull, Assurance, Pub-Sub



# System Development Reqs Include:

- Composeable distributed web services application development environment
- Rapid, visually driven application prototyping and development
- Highly granular, complex transaction and systems modeling & simulation; directly interface live code to simulation environments such as OPNET
- Complete end to end control of entire object framework
- Directly extensible

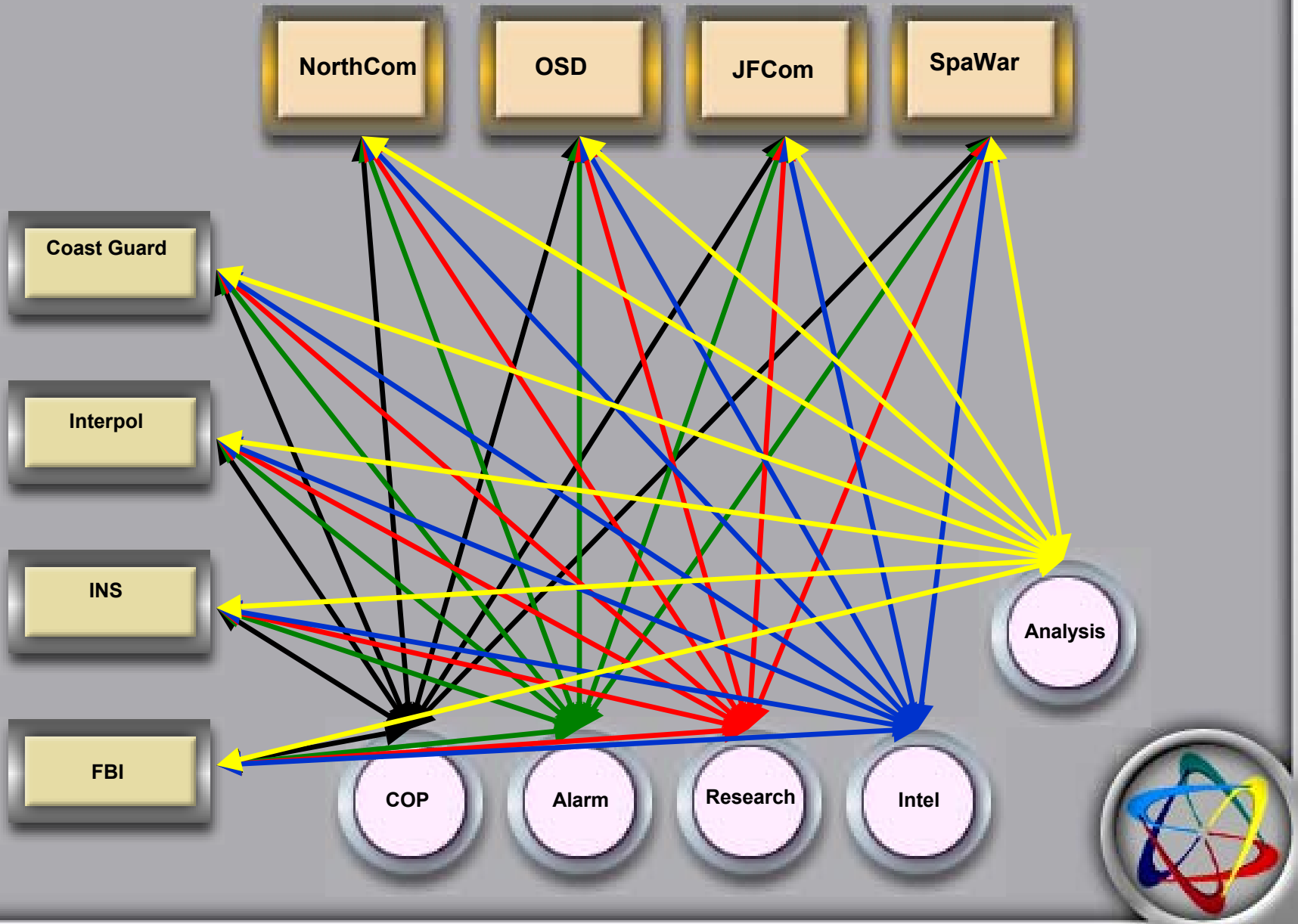


# A Secure Object Framework for Development:

- Robust object-based security model
- Isolates community, control, private and local data
- Active agent-based fabric for network router defense, provisioning, control, redundancy and management
- Large to massive scale heterogeneous systems integration
- Supports distribution of policy and pre-positioning of contingent policy
- Active control of networked application



# The N-square Problem as an Application Issue



# The N-Square Problem:

## ■ N-Square Permutations

- Multiple manifestations
  - Application Interfaces
  - Permission Mapping
  - Community Data Set Noise versus Granular Differentiation

## ■ Interoperability

- Incompatible Hardware, OS, and Applications
- Abstraction and permission difficulties
  - Orthogonality
  - Unfiltered
- Incompatible Data Structures

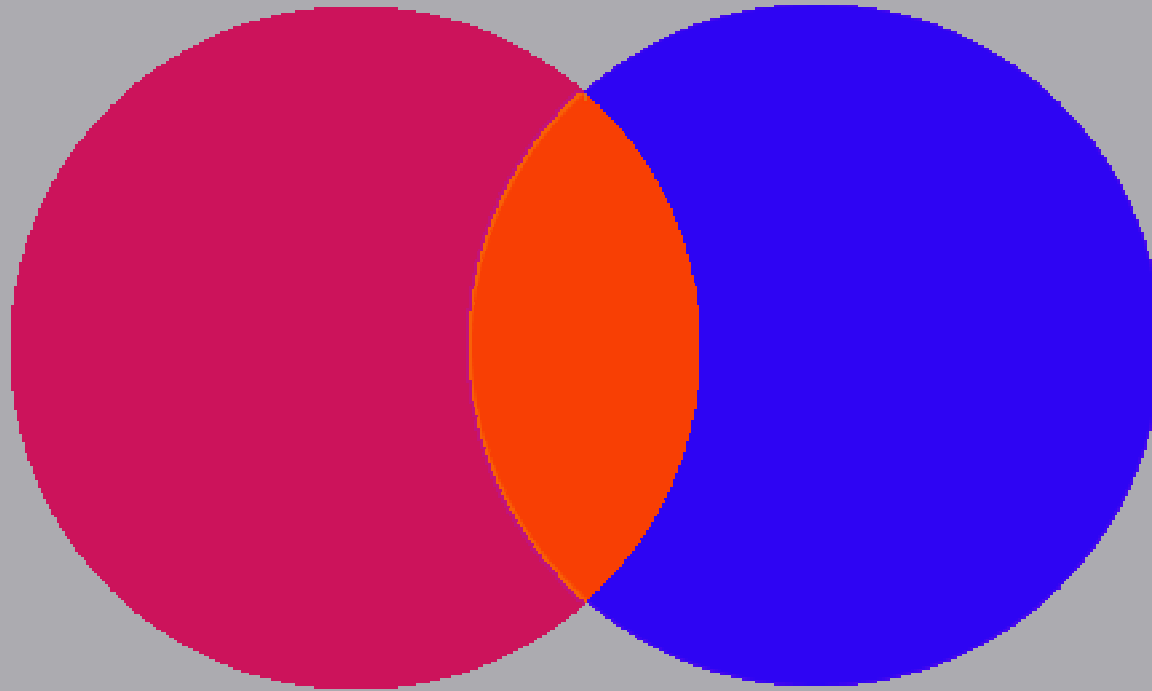


# The Problem and How to Solve It:

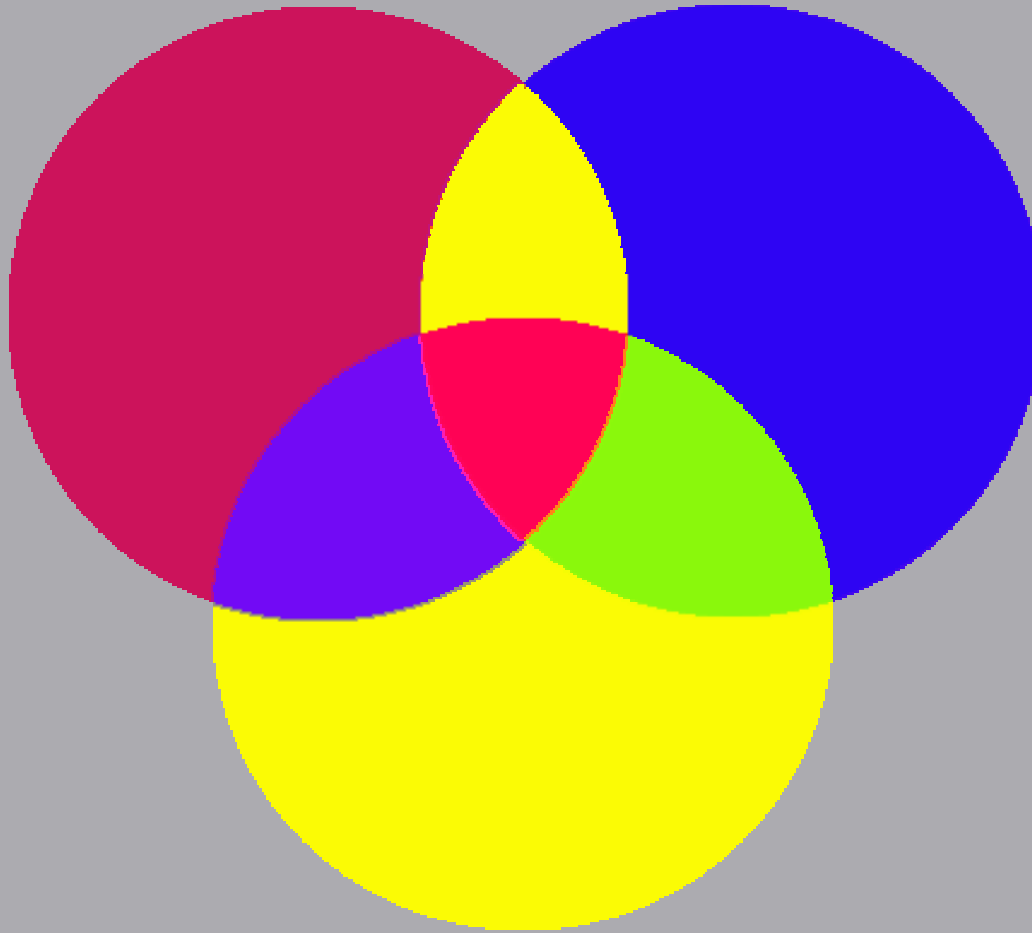
- **The Solution Must Support:**
  - Platform Independence
  - Multi-tier Abstraction
  - Transformation



# N-square as it Applies to Data Sets or Heterogeneous User Communities – the Simple Intersection of Two Data Communities



# Real Data and User Relationships Are More Granular



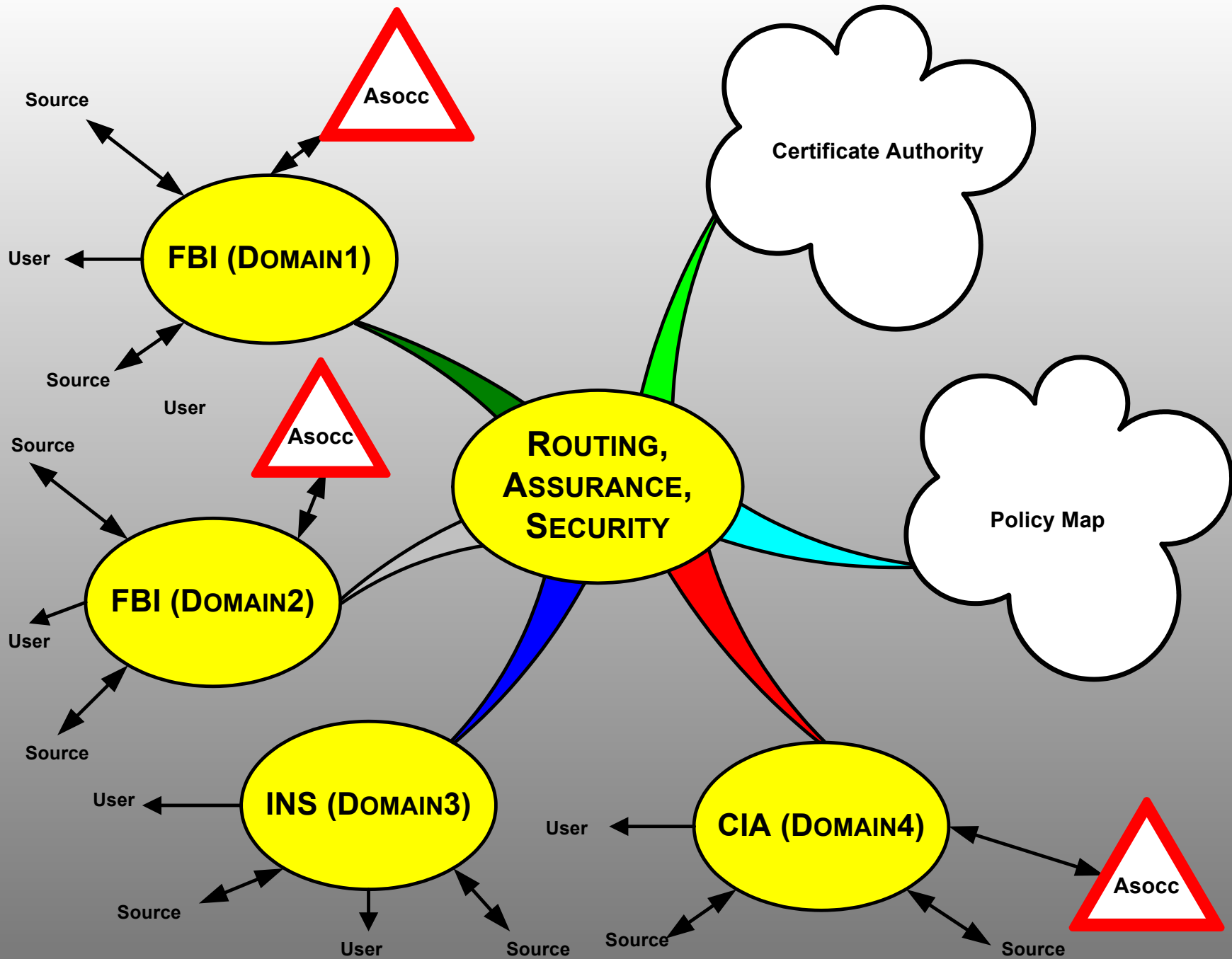


# The Problem and How to Solve It:

## The Solution must support:

- Object level granularity
- Intelligent mapping of use, purpose and permission policy on an ad-hoc basis
- Multiple encapsulated tiers separating object connections, user permissions, application connectors, and user presentation





# The Problem and How to Solve It:

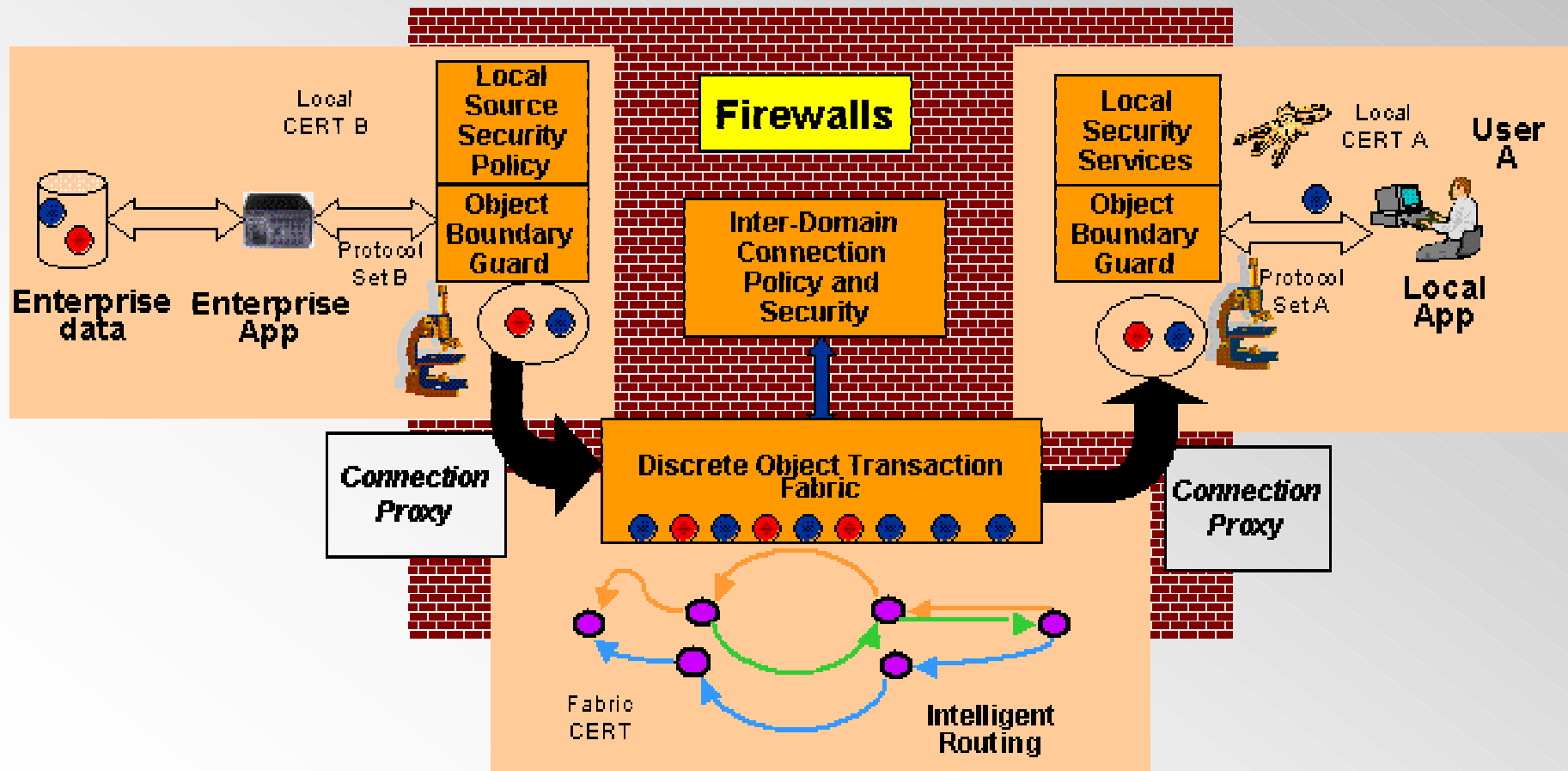
## ■ Cross Boundary Security - Traditional Firewalls Are Not Content Aware

- Susceptible to DoS attack
- Spoofing
- Mal-formed or illegal requests
- Permission hacking
- Trojan or insider manipulation



# Four Key Enablers

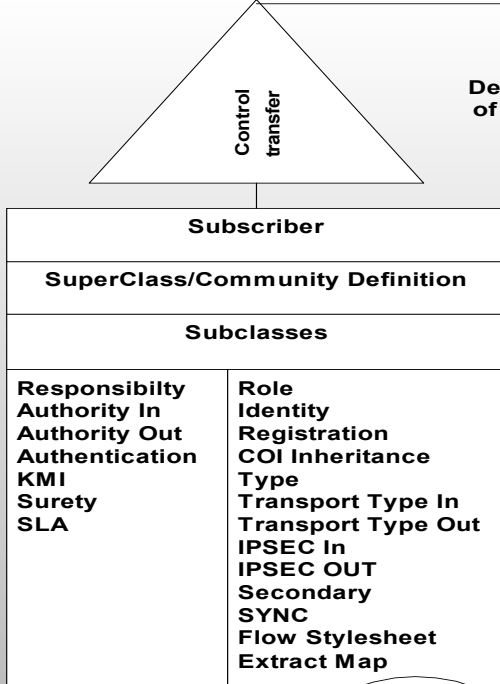
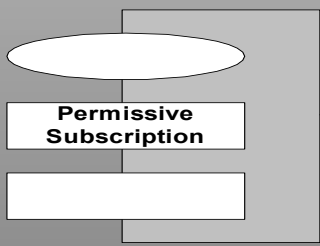
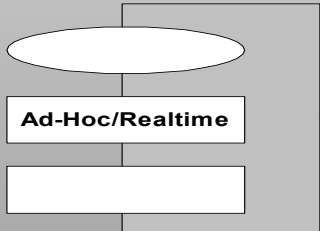
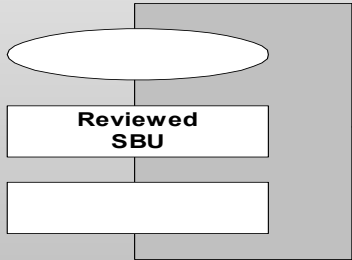
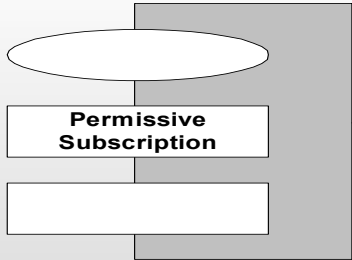
- Protocol independence through customizable gateway proxy of format, security and protocol
- Granular object inspection/authentication inside well understood boundaries
- Encapsulation for both source data and user application domains
- Object fabric distributes routing policy without requiring end-point control



# The Problem and How to Solve It:

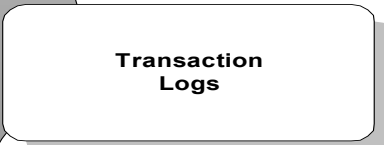
- The Solution Must Support:
  - System Wide Distribution of Policies
  - Role-Based Access Control
  - Object Level Encryption with Nested Security Levels within a Single Object
  - Boundary or Domain Guard Verifying Content Signatures/Permissions





Manual Or Automated Depending On Map of Cert, Policy and Request

Add, Edit, Delete Domains



Distributed  
HOLS Framework  
Roles  
Policy Map  
SLA  
Information Assurance  
Transport  
Surety



Role Definition

# The Problem and How to Solve It:

## ■ VPN type tunneling; like HAIPE:

- Does not scale well due to point-to-point dependency
- Wastes bandwidth because of multiple point-to-point transmission and retransmissions of data
- Results in degraded reliability without support for asynchronous connections
- Often exposes content in actual application due to uncontrolled local caching

## ■ The Solution Must Support:

- Object tunneling
- Caching of objects across network fabric
- Object level encryption and keys with nested security levels



# Other Practical Deployment Issues Must be Addressed:

- Survivability / Reliability
  - Confirmation of delivery
  - Redundant delivery routes
  - No central hubs or transfer points
- Ease and speed of deployment
- Flexibility to address wide variation of requirements for information creation, delivery and access
- Accommodation of Ever-changing Requirements
  - New Relationships
  - Flexibility
  - Change
  - Escalation
- Web platform connectivity



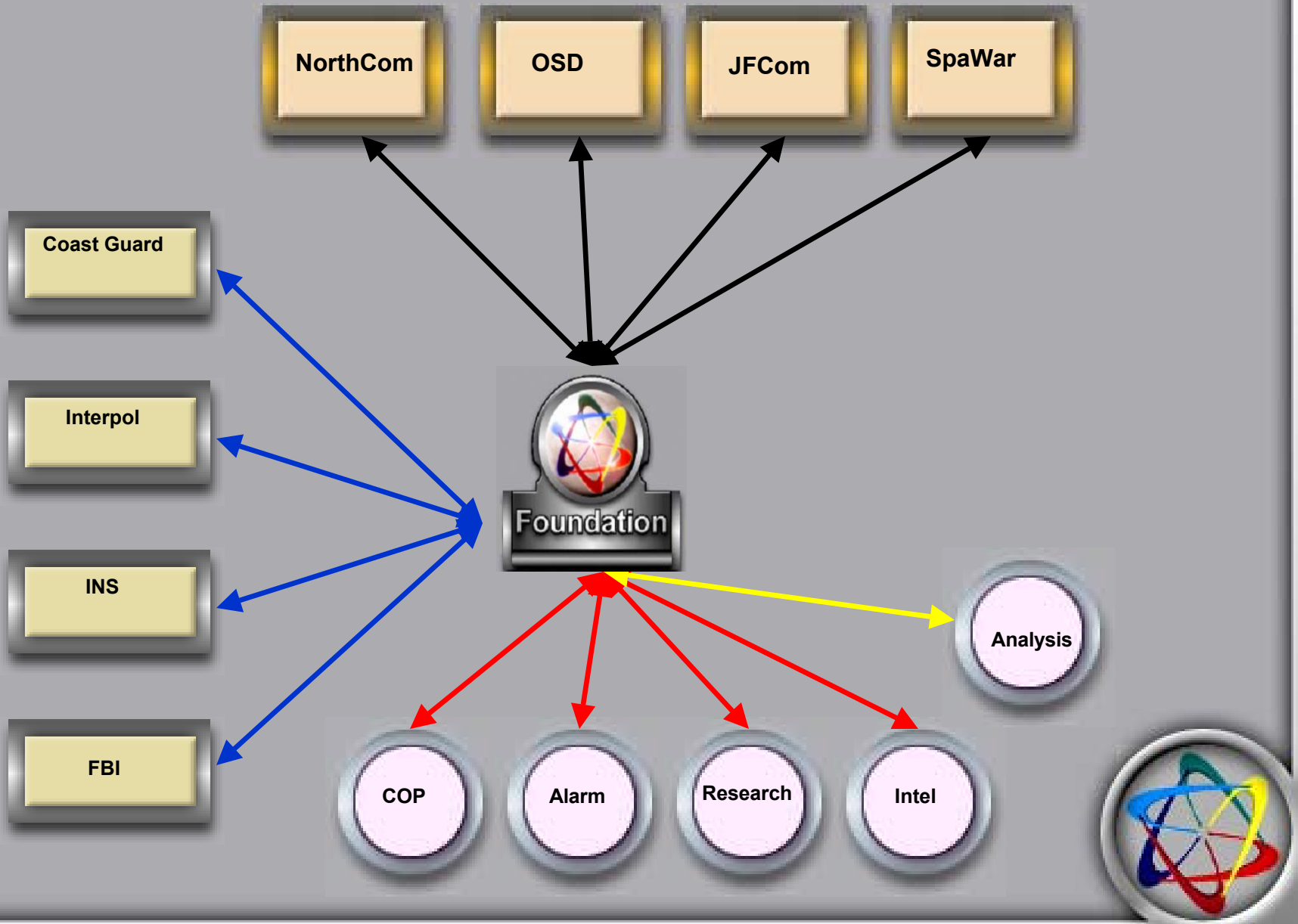
# The Secure Object Framework must address all of these requirements:

The framework provides an agent-based distributed grid computing platform that has been optimized for rapid development, high performance, flexibility and cross platform application deployment supporting:

- An unlimited number of participating users and systems,
- A robust multi-level, multi-tier security model,
- Simple implementation,
- Visual framework implementation, and
- Direct extensibility

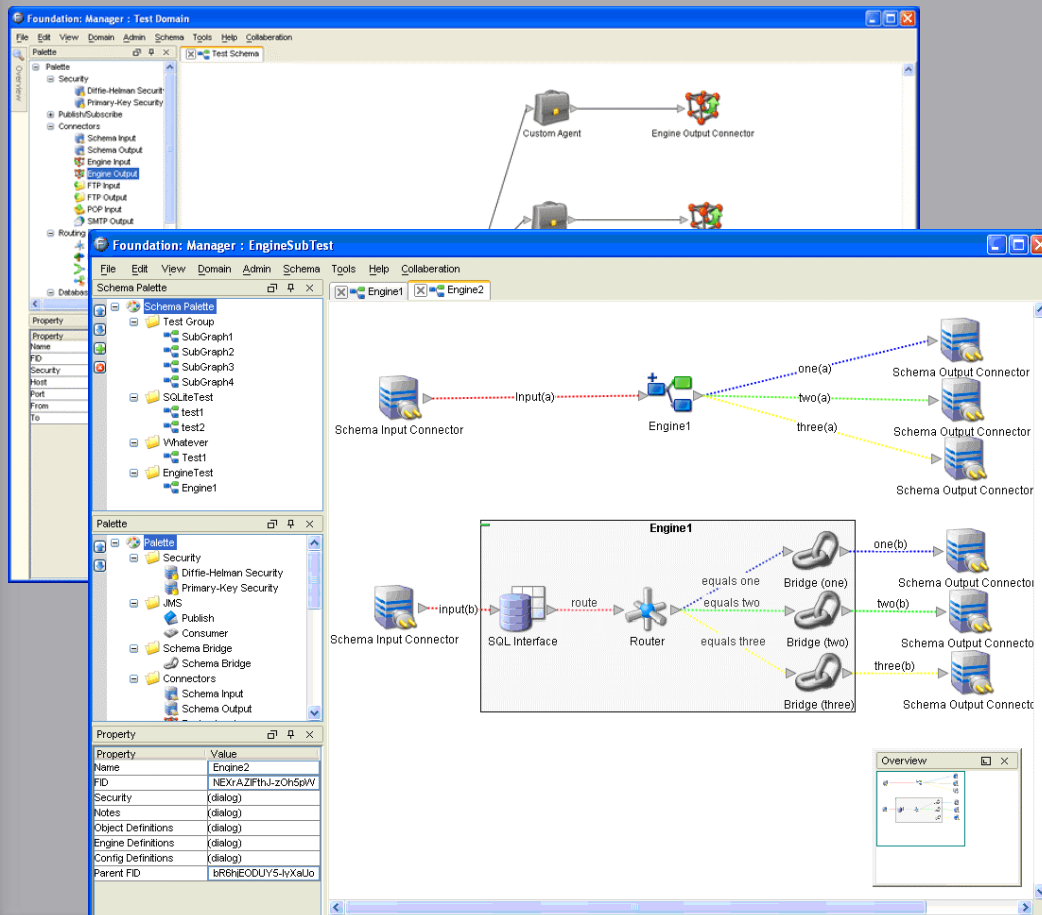


# Handling the N-square Application Problem



# Core Technology Base

**EXISTING OBJECT TRANSACTION FABRIC**  
(Standards-based at edge,  
proprietary if necessary)



## Supported Protocols

- HTTP(S)
- SAML
- FTP
- CORBA
- TN3270
- SNA
- SQL
- SMTP
- Others as needed

## Security Environments

- Commercial X.509 series
- PKI
- WSDL
- Proprietary

## App/Service Environments

- SOAP/XML/UDDI
- J2EE
- .NET
- SQL
- VB
- MQ Series
- Others as needed



# Network Centric Solution Components

## GATEWAYS/AGENTS

- *Independent software program independently running on various machines that create the network, each executing its own rule set or code*

## RULE SETS

- *Rule sets are business rules that determine how a piece of information is to be treated as it flows through the complex distribution network, implemented in an extremely powerful and flexible way, the rules support almost any processing requirement*

## COMMUNITY of INTERESTS/LIBRARIES

- *Groups that wants to receive or may generate different types of information*

## VISUAL CONFIGURATION MANAGER

- *Used to securely administer the operation of the gateways through remote configuration and implementation of business rules*

- *End User Gate Way*- acts as the user's portal into the Foundation network
- *Special Service Gateways*-move information in to and out of the network in the form that other systems can understand-ranging from fax and e-mail support to connection standards for applications like SOAP, .NET, JMS, Corba, MQSeries and many others
- *Traffic Gateways* are placed at the necessary points throughout the network to optimize traffic and routing allowing certain types of information to be preferentially sent down certain paths
- *Community of Interest*-when information is generated across the network it is automatically classified as belonging to one or more communities of interest
- *Community Library*-when it is delivered to the network it is then automatically replicated to the communities in the format those communities can utilize and is stored in the community library.
- End user gateways subscribe to a community library and are permitted to provide and use information according to their security rights
- Cross boundary security and internal control is maintained by the organization

**Foundation Architecture to support a network-centric computing platform that provides a platform for developing complex high performance distributed applications**

# Base Components –Solution Framework

Function	Differentiating Factor	
Platform Independence	Pure JAVA end to end deploys on any compatible JVM 1.4.1+	
Cross-Firewall Transportation	Moves information across firewall and physical boundaries and verifies that individual content delivery requests are permitted	
Filtering Domain Guard	Automates filtering of improper requests, broken instructions and non-permitted object transportation; request flooding never penetrates past the Gateway	
LDAP/UDDI/ Directory Service	Allows external user directory and system permissions reference	
Prioritization	Gateways will deliver information in priority order holding or interrupting large object transmission when higher priority information comes in	
Multiple Community Access	Routing capability extends a single request to multiple communities	
Service Proxy	Seamlessly connect and route multiple service connections in multiple formats	
Parallel Processing/ Routing	Supports breaking communication into many multiple paths, and manages parallel decision and delivery systems	

# Base Components –Solution Framework

Function	Differentiating Factor
<b>Pullbacks/Rerouting</b>	<b>Reroutes gateway information around bottlenecks and obstructions providing for network self healing and ensuring content delivery</b>
<b>Alternative Routing</b>	<b>Provides robust intelligent load balancing across the system allowing the efficient use of all resources</b>
<b>Instream Transformation</b>	<b>Supports the automatic reformatting of over 260 types of content, allowing other entities to receive and use information in a format they can accept; support for other formats easily installed</b>
<b>Delivery Assurance</b>	<b>Internal business rules, system logic and information queuing can be used to create sophisticated information delivery assurance processes beyond the already high level of assurance built into the product</b>
<b>Gateway/AGENT Independence</b>	<b>Enhances performance and provides immense scalability, systems have been installed running more that 100,000,000 routing decisions/hour; multiple platform support allows the Gateways to run on any system including wireless devices</b>
<b>Database Connectivity</b>	<b>Support for native and JDBC connectivity</b>
<b>3DES, RSA, AES, Strong Encryption</b>	<b>Ensures that the Gateways perform as authorized and protect content from being manipulated or improperly disclosed; support for nested and proprietary architectures</b>

# Key Inter-Domain Security Issues:

- Separation of User Identity Certification from Data Owner
  - User Identity is a Global Concept, it is the sign-on domain that provides User/Domain Authentication
  - Domain Authentication requires inter-domain policy administration with MOU between participants
  - No central hubs or transfer points but efficiency demands that MOU's between owners and subscribers be standardized
  - Within certain environments there will be central administration authorities
  - User cert inherits its domain specific privileges
- Mixed signing and message filtering must be supported
- Owner maintains sovereignty and defines the policy for access to their data

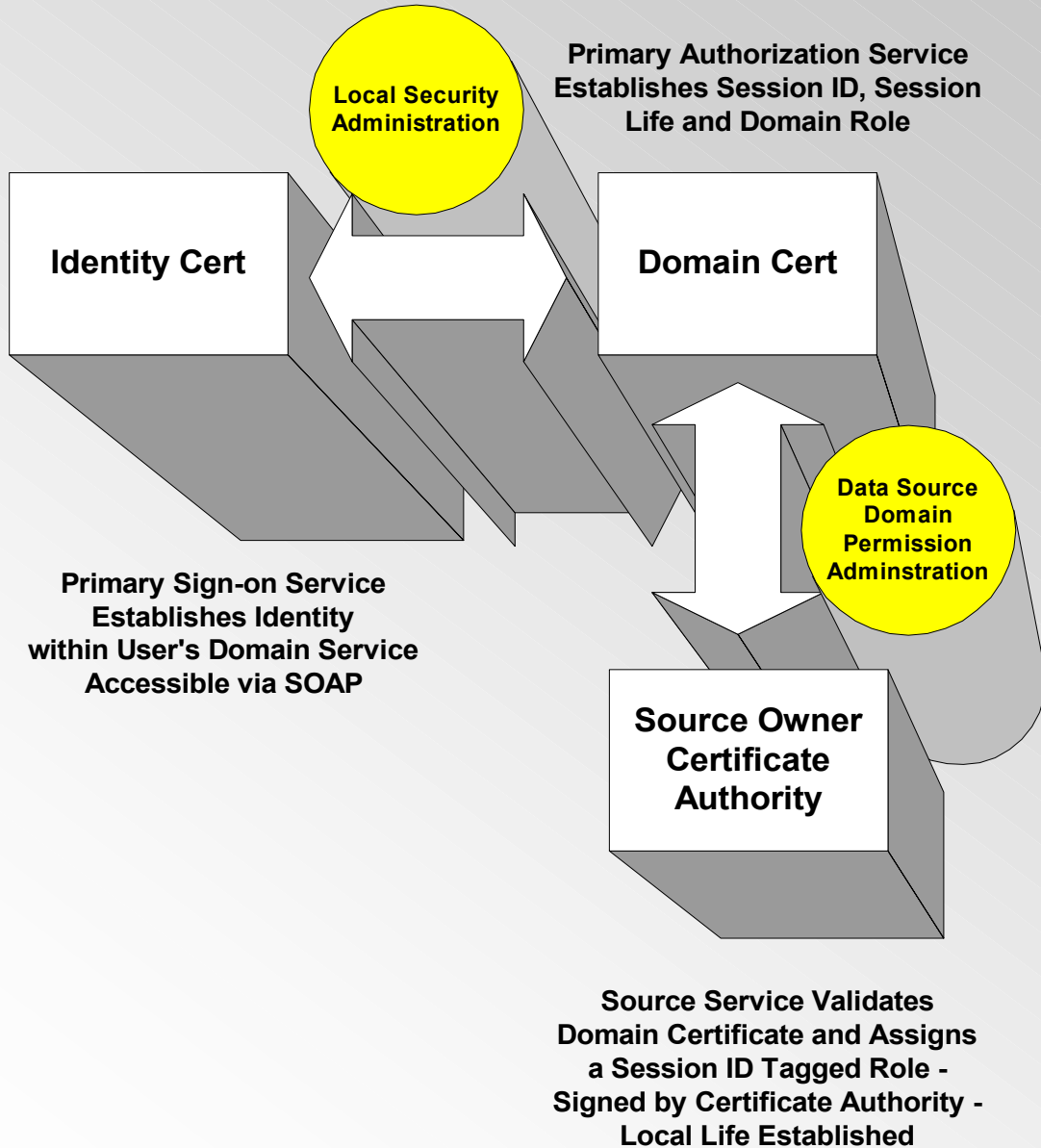


## Other Key Inter-Domain Security Issues:

- Certificate Authorities are separated from each other and available via web services
- Authorization dialog uses self-contained XML objects with permissive syntax
- Process is Asynchronous using internal session identifier(s) to track object history and permissions
- Recursive signing and encryption are permitted
- XML context tags may be contained within the encryption envelope
- Keys optionally contained within object
- Role assertion can include reference routing separate from access authority
- Side-band signaling points to reference object ID
- Owner and return tags can sit inside or outside encryption/signature boundary



# Example Multi-tier Security Model



# Application Use Cases:

- Object-based cross domain application integration,
- Seamless cross platform communication protocol integration,
- Role-based access control in a complex multi-application environment,
- Cross domain firewall/guard implementation,
- Silent data mining,
- Application integration,
- Dynamic assembly of Communities of Interest, and
- Application federation/reachback solutions.



