almost all cases, there will be a reasonably obvious candidate organization to take on this role, for each industry sector or large organization that decides it is appropriate to interoperate with an AXN.

The solid arrowed lines in Figure 4.13b indicate agreements with the trust framework provider for implementing technical, operations, and legal requirements. The dashed arrowed lines indicate other agreements potentially affected by these requirements. In general terms, the model illustrated in Figure 4.13b would operate in the following way. Responsible persons within participating organizations determine the technical, operational, and legal requirements for exchanges of identity information that fall under their authority. They then select OITF providers to implement these requirements. These OITF providers translate the requirements into a blueprint for a trust framework that may include additional conditions of the OITF provider. The OITF provider vets identity service providers and RPs and contracts with them to follow its trust framework requirements when conducting exchanges of identity information. The contracts carry provisions relating to dispute resolvers and auditors for contract interpretation and enforcement.

4.9 CASE STUDY: RBAC SYSTEM FOR A BANK

The Dresdner Bank has implemented an RBAC system that serves as a useful practical example [SCHA01]. The bank uses a variety of computer applications. Many of these were initially developed for a mainframe environment; some of these older applications are now supported on a client-server network while others remain on mainframes. There are also newer applications on servers. Prior to 1990, a simple DAC system was used on each server and mainframe. Administrators maintained a local access control file on each host and defined the access rights for each employee on each application on each host. This system was cumbersome, time-consuming, and error-prone. To improve the system, the bank introduced an RBAC scheme, which is systemwide and in which the determination of access rights is compartmentalized into three different administrative units for greater security.

Roles within the organization are defined by a combination of official position and job function. Table 4.4a provides examples. This differs somewhat from the concept of role in the NIST standard, in which a role is defined by a job function. To some extent, the difference is a matter of terminology. In any case, the bank's role structuring leads to a natural means of developing an inheritance hierarchy based on official position. Within the bank, there is a strict partial ordering of official positions within each organization, reflecting a hierarchy of responsibility and power. For example, the positions Head of Division, Group Manager, and Clerk are in descending order. When the official position is combined with job function, there is a resulting ordering of access rights, as indicated in Table 4.4b. Thus, the financial analyst/Group Manager role (role B) has more access rights than the financial analyst/Clerk role (role A). The table indicates that role B has as many or more access rights than role A in three applications and has access rights to a fourth application. On the other hand, there is no hierarchical relationship between office

Table 4.4 Functions and Roles for Banking Example

(a) Functions and Official Positions

Role	Function	Official Position
A	financial analyst	Clerk
B	financial analyst	Group Manager
C	financial analyst	Head of Division
D	financial analyst	Junior
E	financial analyst	Senior
F	financial analyst	Specialist
G	financial analyst	Assistant
. . .	. . .	. . .
X	share technician	Clerk
Y	support e-commerce	Junior
Z	office banking	Head of Division

(b) Permission Assignments

Role	Application	Access Right
A	money market instruments	1, 2, 3, 4
	derivatives trading	1, 2, 3, 7, 10, 12
	interest instruments	1, 4, 8, 12, 14, 16
B	money market instruments	1, 2, 3, 4, 7
	derivatives trading	1, 2, 3, 7, 10, 12, 14
	interest instruments	1, 4, 8, 12, 14, 16
	private consumer instruments	1, 2, 4, 7
• • •	• • •	• • •

(c) Permission Assignment with Inheritance

Role	Application	Access Right
A	money market instruments	1, 2, 3, 4
	derivatives trading	1, 2, 3, 7, 10, 12
	interest instruments	1, 4, 8, 12, 14, 16
B	money market instruments	7
	derivatives trading	14
	private consumer instruments	1, 2, 4, 7
• • •	• • •	• • •

banking/Group Manager and financial analyst/Clerk because they work in different functional areas. We can therefore define a role hierarchy in which one role is superior to another if its position is superior and their functions are identical. The role hierarchy makes it possible to economize on access rights definitions, as suggested in Table 4.4c.

In the original scheme, the direct assignment of access rights to the individual user occurred at the application level and was associated with the individual application. In the new scheme, an application administration determines the set of access rights associated with each individual application. However, a given user performing a given task may not be permitted all of the access rights associated with the application. When a user invokes an application, the application grants access on the basis of a centrally provided security profile. A separate authorization administration associated access rights with roles and creates the security profile for a use on the basis of the user's role.

A user is statically assigned a role. In principle (in this example), each user may be statically assigned up to four roles and select a given role for use in invoking a particular application. This corresponds to the NIST concept of session. In practice, most users are statically assigned a single role based on the user's position and job function.

All of these ingredients are depicted in Figure 4.14. The Human Resource Department assigns a unique User ID to each employee who will be using the system. Based on the user's position and job function, the department also assigns one or more roles to the user. The user/role information is provided to the Authorization Administration, which creates a security profile for each user that associates the User ID and role with a set of access rights. When a user invokes an application, the application consults the security profile for that user to determine what subset of the application's access rights are in force for this user in this role.

A role may be used to access several applications. Thus, the set of access rights associated with a role may include access rights that are not associated with one of the applications the user invokes. This is illustrated in Table 4.4b. Role A has numerous access rights, but only a subset of those rights are applicable to each of the three applications that role A may invoke.

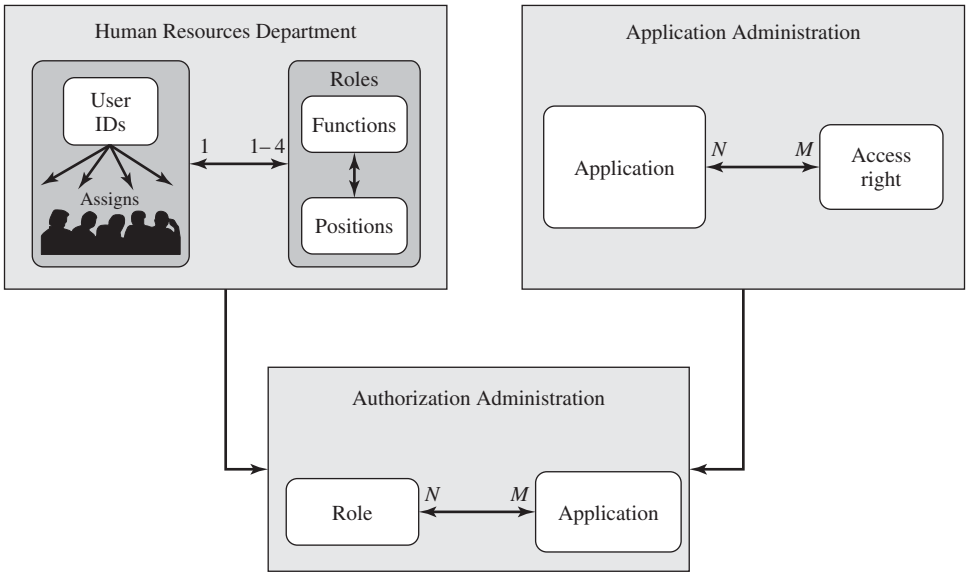

Figure 4.14 **Example of Access Control Administration**

Some figures about this system are of interest. Within the bank, there are 65 official positions, ranging from a Clerk in a branch, through the Branch Manager, to a Member of the Board. These positions are combined with 368 different job functions provided by the human resources database. Potentially, there are 23,920 different roles, but the number of roles in current use is about 1,300. This is in line with the experience other RBAC implementations. On average, 42,000 security profiles are distributed to applications each day by the Authorization Administration module.

4.10 RECOMMENDED READING

[DOWN85] provides a good review of the basic elements of DAC. [KAIN87] is a clear discussion of capability-based access control.

[SAND96] is a comprehensive overview of RBAC. [FERR92] also provides some useful insights. [BARK97] looks at the similarities in functionality between RBAC and DAC based on access control lists. [SAUN01] is a more general comparison of RBAC and DAC. [FRAN12] discusses the strengths and weakness of RBAC and its applicability in various contexts. [FERR01] and [SAND00] present the NIST RBAC standard in exhaustive detail. [COYN08] describes the INCITS RBAC implementation and interoperability standard, based in the NIST standard.

NIST SP 800-16 [HU13] provides an excellent overview of ABAC models and their application. [CIOC11] is a clear and thorough introduction to ICAM. [RUND10] is a useful overview of OITF. [OIX13] is a much more detailed treatment of trust frameworks and attribute exchange networks.

BARK97 Barkley, J. "Comparing Simple Role-Based Access Control Models and Access Control Lists." *Proceedings of the Second ACM Workshop on Role-Based Access Control*, 1997.

CIOC11 CIO Council. Federal Identity, Credential, and Access Management (FICAM) Roadmap and Implementation Guidance. cio.gov. December 2011.

COYN08 Coyne, E., and Weil, T. "An RBAC Implementation and Interoperability Standard." *IEEE Security & Privacy*, January/February 2008.

DOWN85 Down, D., et al. "Issues in Discretionary Access Control." *Proceedings of the 1985 Symposium on Security and Privacy*, 1985.

FERR92 Ferraiolo, D., and Kuhn, R. "Role-Based Access Control." *Proceedings of the 15th National Computer Security Conference*, 1992.

FERR01 Ferraiolo, D., et al. "Proposed NIST Standard for Role-Based Access Control." *ACM Transactions on Information and System Security*, August 2001.

FRAN12 Franqueira, V., and Wieringa, R. "Role-Based Access Control in Retrospect." *Computer*, June 2012.

HU13 Hu, V. et al. *Guide to Attribute Based Access Control (ABAC) Definition and Considerations*. NIST SP 800-62, September 2013.

KAIN87 Kain, R., and Landwehr. "On Access Checking in Capability-Based System." *IEEE Transactions on Software Engineering*, February 1987.