STEVENS
INSTITUTE of TECHNOLOGY
THE INNOVATION UNIVERSITY

Relational Database Design

Danny T. Ho
Center for Business Process Innovation
Howe School of Technology Management
Stevens Institute of Technology
Hoboken NJ
tho2@stevens.edu

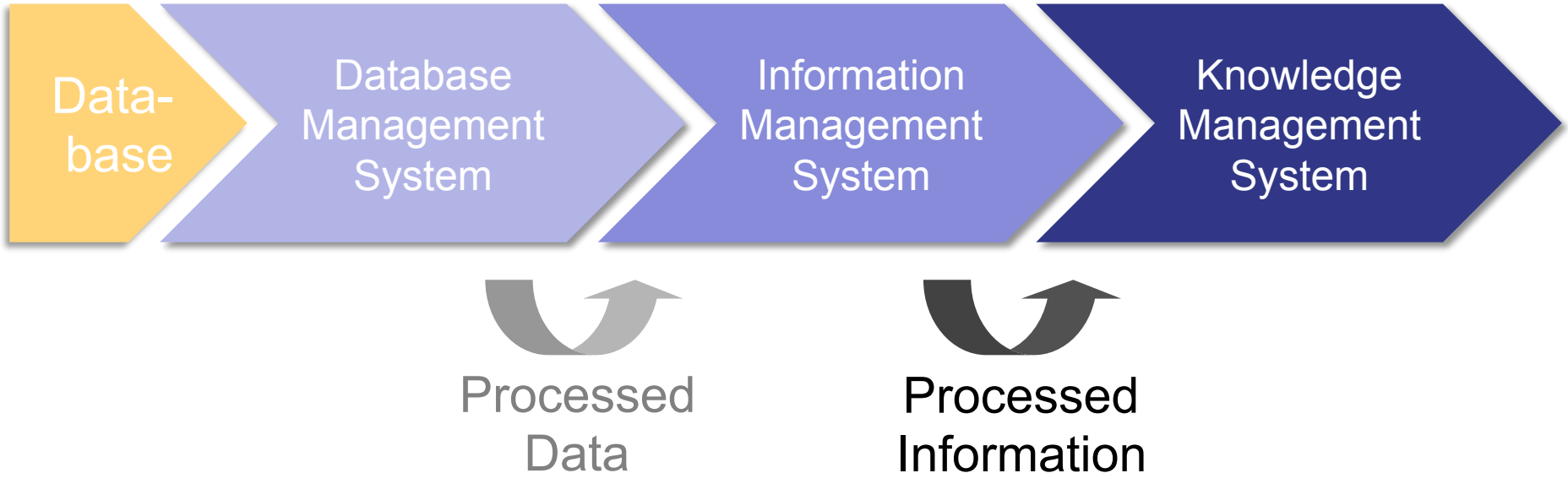
*Lecture in part based on "The Essence of Databases" by F.D. Rolland

HOWE SCHOOL
of Technology Management

Databases, Data, and Information

- Data
  - Collection of unprocessed items (e.g., text, numbers, images, etc.)
- Database
  - Collection of data organized in a manner that allows access, retrieval, and use of that data
- Information
  - Data that has been organized or presented in a meaningful fashion
  - Process data (organized, meaningful, useful)
- Database Management System (DBMS)
  - Controls access to the data and provides for required data management facilities
  - Issues requests to use the the data storage facilities in order to service different applications

Relational Database Design: The Goals



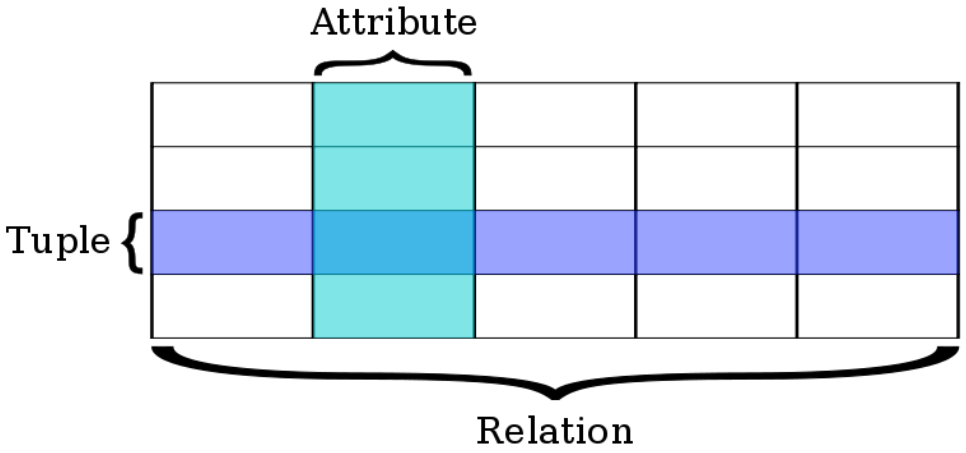
- **To store information without unnecessary redundancy**
- **To retrieve information easily**

Types of Databases

- Relational Databases
  - stores data in tables that consist of rows and columns
  - ideal for simple or structured data
  - may be queried through *Structured Query Language (SQL)*
- Object-oriented Databases
  - stores data in objects
  - ideal for complex or unstructured data
  - may be queried through *Object Query Language (OQL)*
- Multidimensional Databases
  - stores data in more than two dimensions of data
  - can consolidate data much faster than a relational database

Relational Databases: Basic Structure

- Tables: each is assigned with a unique name, a.k.a. **relations**.
- Rows: each represents a relationship among a set of values, a.k.a. **tuples**.
- Columns: each represents an **attribute** used to describe a relationship.
  - **Domain**: the set of permitted values for each attributes

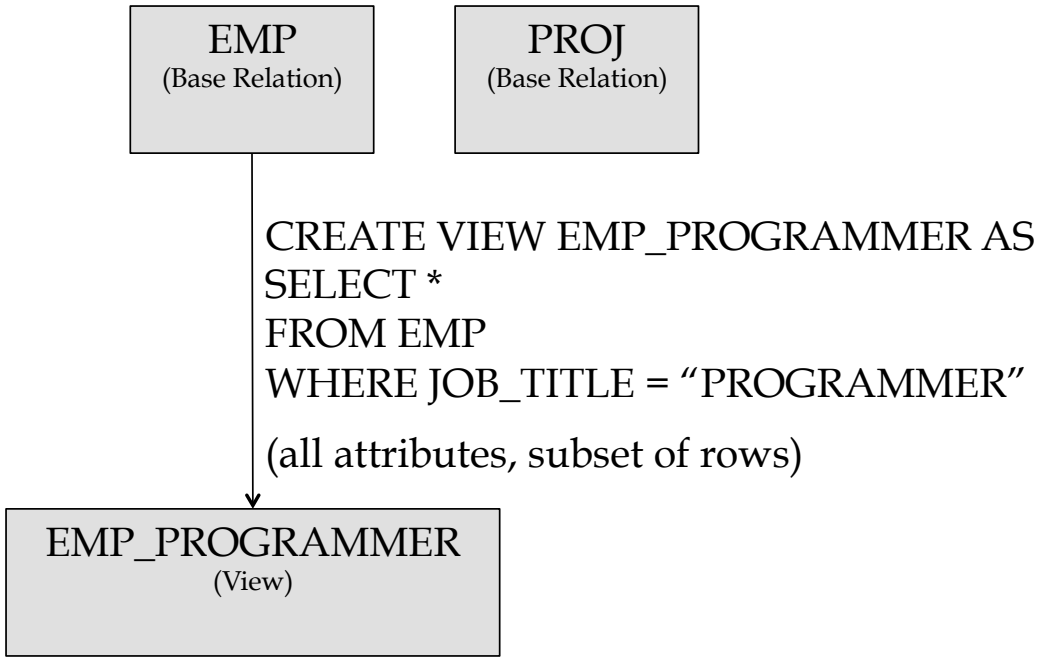


A *Relation* is defined as a set of *tuples* that have the same *attributes*.

*Figure adapted from Wikipedia

Relational Databases: Basic Structure (con't)

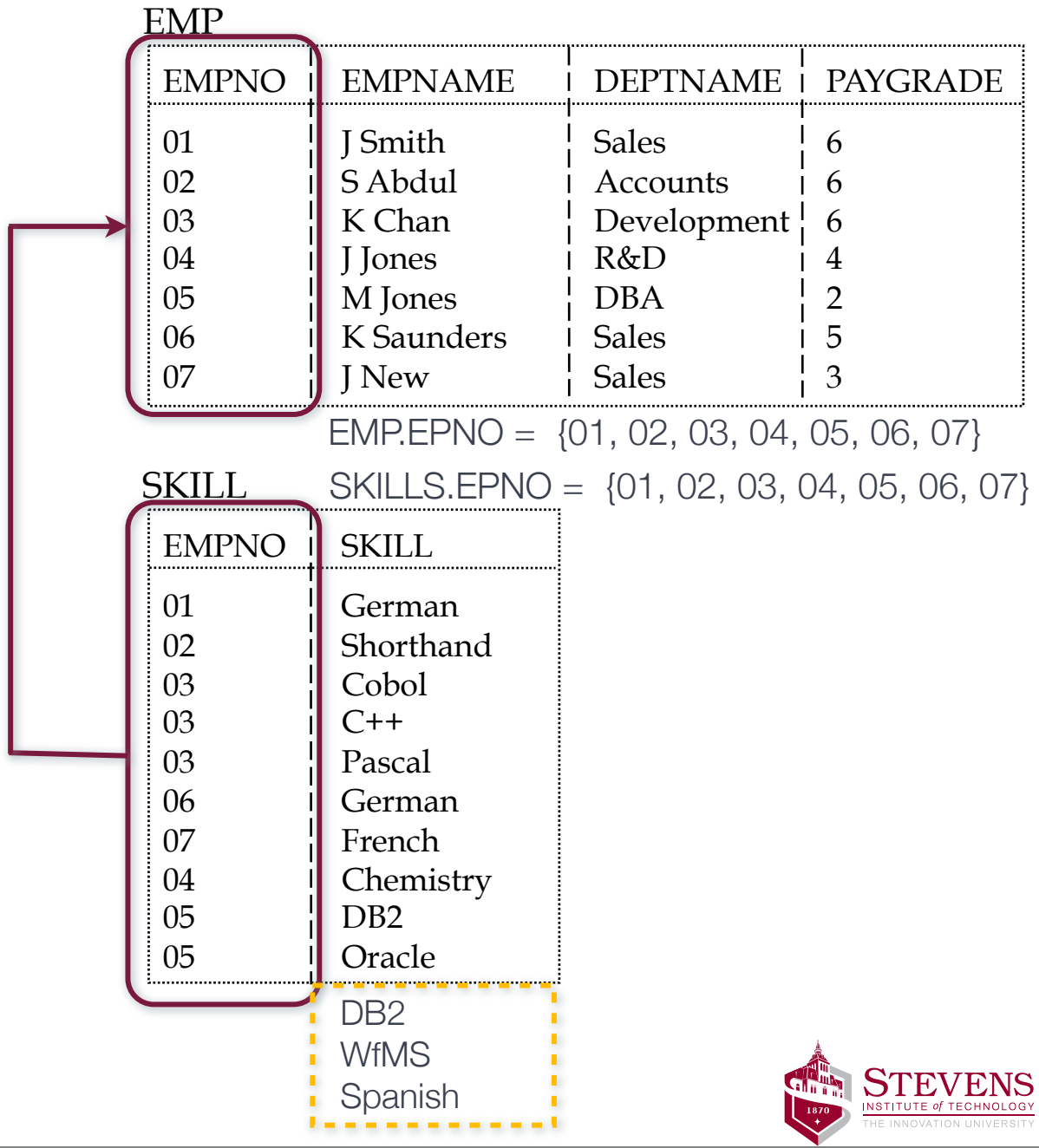
- In a relationship database, all data are stored and accessed via relations, a.k.a. **tables**.
  - Base relation: relations used for storing data.
  - Derived relation: relations computed by applying relational operations to other relations, a.k.a. **view** or **queries**.



Relational Databases: Basic Structure (con't)

- Primary key
  - an attributes or a set of attributes that *uniquely* defines each tuple (row/entry) in a relation (table).
- Foreign key
  - an attribute or a set of attributes in one relation (table) that matches the primary key of another relation (table).
- Two principle rules for the relational model
  - **Entity integrity**: an entity (tuple) must have an independent existence. It is enforced by:
    - requiring each base relation (table) to have a unique primary key
    - the primary key must not contain NULL values
  - **Referential integrity**: all values of all foreign keys must be valid.

All references from SKILLS to EMP are valid since: EMP.EPNO == SKILLS.EPNO

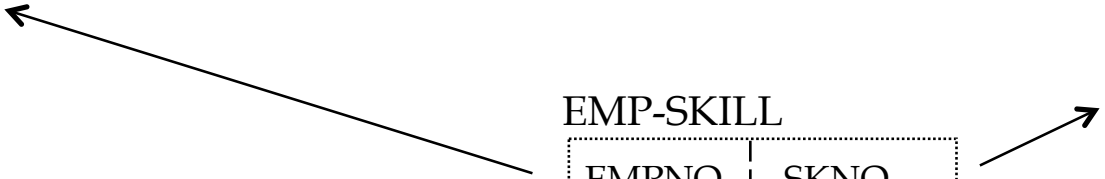


EMP

EMPNO	EMPNAME	DEPTNAME	PAYGRADE
01	J Smith	Sales	6
02	S Abdul	Accounts	6
03	K Chan	Development	6
04	J Jones	R&D	4
05	M Jones	DBA	2
06	K Saunders	Sales	5
07	J New	Sales	3

SKILL

SKNO	SKILL
01	German
02	Shorthand
03	Cobol
04	C++
05	Pascal
06	German
07	French
08	Chemistry
09	DB2
10	Oracle
11	Sybase



EMP-SKILL

EMPNO	SKNO
01	01
02	02
03	03
03	04
03	05
06	01
07	07
04	08
05	09
05	10

Database Normalization

- Goal:
  - To ensure the data structures in a RDBMS are “efficient”
    - Absence of redundancy (while free of modification anomalies)
    - Minimal use of null values
    - Prevention of loss of information
- Normalization Granularity
  - Unnormalized, 1NF, 2NF, 3NF, 4NF, and 5NF
  - Most 3NF relations (tables) are free of insertion, update, and deletion anomalies
- Tradeoffs in choosing a proper level of normalization granularity
  - many small tables may be fine for machines, but people prefer viewing denormalized data, even unnormalized data

Unnormalized Data

BANK

BRANCHNAME	ADDRESS	MANAGER_NO	ACCNO	BALANCE	TYPE
Crawley	3 High Street	1768	(120768, 678453, 348973)	(234, 456, 12567)	(S, C, C)
Stonehouse	2 Low Street	9823	(987654, 745363)	(789, 23)	(C, S)

CUSTOMER

CUSNO	NAME	ADDRESS	STATUS	ACCNO
2345	Abdul	23 High Street	Business	(120768, 348973)
7654	Peters	45 Ash Street	Personal	(987654)
8764	Jones	17 Low Street	Business	745363, 678453, 348973)

From Unnormalized From to 1NF

- For a table to be a relation in 1NF:
  - The cells of the table must be single valued
  - The entries in a column must be of the same kind
  - Each column must have a unique name
  - Each row must be unique
  - The order of rows and columns is insignificant

Level 1 Normalization

BANK (Unnormalized)

BRANCHNAME	ADDRESS	MANAGER_NO	ACCNO	BALANCE	TYPE
Crawley	3 High Street	1768	(120768, 678453, 348973)	(234, 456, 12567)	(S, C, C)
Stonehouse	2 Low Street	9823	(987654, 745363)	(789, 23)	(C, S)

BANK (1NF)

BRANCHNAME	ADDRESS	MANAGER_NO	ACCNO	BALANCE	TYPE
Crawley	3 High Street	1768	120768	234	S
Crawley	3 High Street	1768	678453	456	C
Crawley	3 High Street	1768	348973	12567	C
Stonehouse	2 Low Street	9823	987654	789	C
Stonehouse	2 Low Street	9823	745363	23	S

Modification Anomalies in 1NF

Employees' Skills

Employee ID	Employee Address	Skill
426	87 Sycamore Grove	Typing
426	87 Sycamore Grove	Shorthand
519	94 Chestnut Street	Public Speaking
519	96 Walnut Avenue	Carpentry

An **update anomaly**.
Employee 519 is shown as having different addresses on different records

Faculty and Their Courses

Faculty ID	Faculty Name	Faculty Hire Date	Course Code
389	Dr. Giddens	10-Feb-1985	ENG-206
407	Dr. Saperstein	19-Apr-1999	CMP-101
407	Dr. Saperstein	19-Apr-1999	CMP-201
424	Dr. Newsome	29-Mar-2007	?

An **insert anomaly**.
Until the new faculty member, Dr. Newsome, is assigned to teach at least one course, his details cannot be recorded

Faculty and Their Courses

Faculty ID	Faculty Name	Faculty Hire Date	Course Code
389	Dr. Giddens	10-Feb-1985	ENG-206
407	Dr. Saperstein	19-Apr-1999	CMP-101
407	Dr. Saperstein	19-Apr-1999	CMP-201

DELETE

An **deletion anomaly**.
All information about Dr. Giddens is lost when he temporarily ceases to be assigned to any courses.

*Figures adapted from Wikipedia

From 1NF to 2NF

- For a table to be a relation in 2NF, a table (relation) must be:
  - in 1NF
  - have no nonkey attributes that are not fully *functionally dependent* upon the primary key. (That is, all of its nonkey attributes has to be fully functionally dependent upon the primary key)
- Functional Dependency
  - In a given table, an *attribute* Y is said to have functional dependency on a set of *attributes* X (written X ⟶ Y) *if and only if* each X value is associated with precisely one Y value.
  - e.g., BRANCHNAME ⟶ ADDRESS (for every BRANCHNAME, there is *one and only one* ADDRESS)

BANK

BRANCHNAME	ADDRESS	MANAGER_NO	ACCNO	BALANCE	TYPE
Crawley	3 High Street	1768	120768	234	S
Crawley	3 High Street	1768	678453	456	C
Crawley	3 High Street	1768	348973	12567	C
Stonehouse	2 Low Street	9823	987654	789	C
Stonehouse	2 Low Street	9823	745363	23	S

From 1NF to 2NF

BANK

BRANCHNAME	ADDRESS	MANAGER_NO	ACCNO	BALANCE	TYPE
Crawley	3 High Street	1768	120768	234	S
Crawley	3 High Street	1768	678453	456	C
Crawley	3 High Street	1768	348973	12567	C
Stonehouse	2 Low Street	9823	987654	789	C
Stonehouse	2 Low Street	9823	745363	23	S

- BRANCHNAME
  1. BRANCHNAME ⟶ ADDRESS
  2. BRANCHNAME ⟶ MANAGER_NO
- ACCNO
  1. ACCNO ⟶ BRANCHNAME
  2. ACCNO ⟶ BALANCE
  3. ACCNO ⟶ TYPE
  4. ACCNO ⟶ ADDRESS
  5. ACCNO ⟶ MANAGER_NO

From 1NF to 2NF

- Goal: to have no ‘nonkey’ attributes that are not fully *functionally dependent* upon the primary key.
- ‘Nonkey’ attributes: attributes that do not form any part of candidate key.
- A candidate key is a ‘superkey’ that cannot be reduced
  - A ‘superkey’ is the set of all attributes and uniquely identifies a tuple (row/entry)
  - for the BANK relation, the ‘superkey’ is:
    - BRANCHNAME, ADDRESS, MANAGER_NO, ACCNO, BALANCE, TYPE
  - the candidate keys can be identified by validating functional dependency
    - **CUSNO, ACCNO ⟶ STATUS, NAME, ADDRESS**

BANK

BRANCHNAME	ADDRESS	MANAGER_NO	ACCNO	BALANCE	TYPE
Crawley	3 High Street	1768	120768	234	S
Crawley	3 High Street	1768	678453	456	C
Crawley	3 High Street	1768	348973	12567	C
Stonehouse	2 Low Street	9823	987654	789	C
Stonehouse	2 Low Street	9823	745363	23	S

Level 2 Normalization

CUSTOMER (1NF)

CUSNO	NAME	ADDRESS	STATUS	ACCNO
2345	Abdul	23 High Street	Business	120768
2345	Abdul	23 High Street	Business	348973
7654	Peters	45 Ash Street	Personal	987654
8764	Jones	17 Low Street	Business	745363
8764	Jones	17 Low Street	Business	678453
8764	Jones	17 Low Street	Business	348973

CUSNO, ACCNO ⟶ STATUS, NAME, ADDRESS

CUSNO ⟶ NAME, ADDRESS, STATUS

CUSTOMER

CUSNO	NAME	ADDRESS	STATUS
2345	Abdul	23 High Street	Business
7654	Peters	45 Ash Street	Personal
8764	Jones	17 Low Street	Business

CUSTOMER_ACCOUNT

CUSNO	ACCNO
2345	120768
2345	348973
7654	987654
8764	745363
8764	678453
8764	348973

From 2NF From to 3NF

- For a table to be a relation in 3NF, we need to remove all *transitive dependencies* by decomposing
- A *transitive dependency* is a special type of functional dependency. It is a functional dependency between one non-key attribute and other non-key attributes.
  - e.g., BRANCHNAME ⟶ ADDRESS, MANAGER_NO
  - This is a functional dependency between a nonkey attribute (BRANCHNAME) and other attributes (ADDRESS, MANAGER_NO), and, therefore, is a *transitive dependency*.
- The existence of transitive dependency in a relation may result in replicated information

Level 3 Normalization

BANK (1NF)

BRANCHNAME	ADDRESS	MANAGER_NO	ACCNO	BALANCE	TYPE
Crawley	3 High Street	1768	120768	234	S
Crawley	3 High Street	1768	678453	456	C
Crawley	3 High Street	1768	348973	12567	C
Stonehouse	2 Low Street	9823	987654	789	C
Stonehouse	2 Low Street	9823	745363	23	S

BRANCH

BRANCHNAME	ADDRESS	MANAGER_NO
Crawley	3 High Street	1768
Stonehouse	2 Low Street	9823

ACCOUNT

ACCNO	BALANCE	TYPE	BRANCHNAME
120768	234	S	Crawley
678453	456	C	Crawley
348973	12567	C	Crawley
987654	789	C	Stonehouse
745363	23	S	Stonehouse

STEVENS
INSTITUTE of TECHNOLOGY
THE INNOVATION UNIVERSITY

Thank You - Questions?

Danny T. Ho
Center for Business Process Innovation
Howe School of Technology Management
Stevens Institute of Technology
Castle Point on the Hudson
Hoboken, NJ 07030
Phone: +1 (201) 216-8152
Fax: +1 (201) 216-5385
E-mail: tho2@stevens.edu
Web: http://www.cebpi.org

HOWE SCHOOL
of Technology Management