Technologies for Developing Systems

Chapter 9

Information Systems Management In Practice 6E

McNurlin & Sprague
Introduction

• This chapter reviews the evolution of system development to provide an understanding of the underlying principles of building applications.

• It discusses underlying technologies, development methodologies, Internet-based systems, and project management.
Introduction

The Evolution of System Development
Introduction

• The system development life cycle emerged in the 1970s.
• More friendly languages and automation of portions of development occurred in the 1980s.
• The 1990s focused more on packages and on building systems by linking components (system integration).
Introduction

• These days, virtually every application is a network application, since the network is becoming the system.

• Web-based applications were the first generation of Internet-centric computing; the new field, Web Services, is as the second.
Introduction

• In addition, the increasing interconnectedness of supply chains is leading companies to build inter-organizational systems.
Foundations of System Development

- Structured development
- 4GLs
- Prototyping
- CASE
- Object-oriented
- Client-server systems
Structured Development

• This type of development introduced more discipline by establishing standards for processes and documentation.

• It raised reliability by introducing inspections, and it aimed to increase productivity.
Fourth Generation Languages

• 4GLs increased productivity and even allowed users to program.
Software Prototyping

• Prototyping introduced the concept of iterative development, by first introducing a quick and inexpensive prototype to be tested and refined.
Computer-Aided Software Engineering (CASE) —

• In the 1980s, CASE automated the structured development techniques of the 1970s — an evolutionary change.
• It consisted of an information repository, front-end tools for planning through design, back-end tools for generating code, and a development workstation.
• Timeboxing (fixed time-frame development) was a technique used.
Object-Oriented Development

- Object-oriented (OO) development was a revolutionary change in the 1980s – develop objects that can be reused.
- It allowed point-and-click programming of graphical user interfaces.
- It is very modular, so a change in one part of a system need not affect the other parts.
Client-Server Computing –

- These systems split work between a client and a server, so they bridged the PC and mainframe worlds.
- The desktop and laptop client machines handle graphics, animation, and video, while the servers handle production updating.
- It is a clever way to fuse the pizzazz of the PC world with the necessary back-end production strengths of the mainframe world.
System Integration

Three approaches:

1. Database Management Systems
   - integrate by sharing the data stored in a single or distributed database

2. Enterprise Resource Planning
   - all applications come from a single vendor and are specifically designed to communicate with each other

3. Middleware
   - applications communicate with each other through third-party translation software
Enterprise Resource Planning

- It aims to integrate corporate systems by providing a single set of applications, from a single vendor, operating with a single database.
- The goal is to provide the means to integrate business departments and functions across an organization.
ERP

• Often, that model differs from the way the customer actually operates, so the customer must change its organizational structure, culture, and processes.
System Integration

Types of Middleware Used in Client-Server Applications

*Inter-application communications facilities: Link components*
- Application programming interfaces (APIs): provide a standard way of interfacing
- Remote procedure call (RPCs): enable a dialogue between two geographically dispersed applications
- Object request brokers (ORBs): allow applications or utilities to interwork in standard ways
- Message-oriented middleware (MOM): uses asynchronous message passing for interapplication communications

*Transaction managers: Handle transactions across multiple platforms*
- Standard query languages (SQLs): standardize the way in which databases are accessed
- TP monitors (CICS, for example): monitor online transaction processing with a database
- Two-phase commit: a protective mechanism for transactions that fail to complete successfully

*Utilities: Provide general services*
- Directory services: resource allocation
- Time services: timing
- Security services: encryption, and so on
- Software distribution: including configuration control
Inter-Organizational System Development

- Business Ecosystems
- Supply Chain Management
- Platform Development
INTER-ORGANIZATIONAL SYSTEM DEVELOPMENT

• One of the main business trends is the appearance of business ecosystems — “groupings” of businesses that work closely together.

1. Supply Chain Management systems integrate supply chains, which are now a major trend as they compete against one another on their ability to reduce costs and time across their entire chains.

• Development of such inter-organizational systems requires teams from the different organizations to work together.
2. Another type of inter-organizational system is a *platform*, which provides the infrastructure for the operation of a business ecosystem, a region, or an industry.

- Platform development is a major trend in an increasing number of industries.
Middleware

• Middleware is software that works between applications and connects them, allowing them to share data.

• A popular type of middleware is Enterprise Application Integration (EAI) products, which use a message broker to transfer data between applications.
  – EAI tools allow users to define business processes and make data integration subject to rules that govern those processes.
Internet-Based Systems

Diagram showing the components of internet-based systems:
- Web Browser Client
- Web Server
- Thin Clients
- Application Servers
- Virtual Application Server

Related terms:
- Database
- ERP
- Legacy Systems
- Middleware
INTERNET-BASED SYSTEMS

- Internet users have become so sophisticated that Internet-based systems must now be scalable, reliable, and integrated, both internally and externally, with the systems of customers and business partners.

- Cyberspace is where the action is in system development.
Application Servers

- Application servers have developed into a framework for developing Internet-based applications.

- In addition to providing middleware and integration functions, application servers have become application development platforms, with a wide range of development and automatic code generation tools.
Internet-Based Systems

- EJB (Enterprise Java Beans)
- J2EE (Java 2 Enterprise Edition)
INTERNET-BASED SYSTEMS: Java

- Java is a fairly open language that has evolved from client-side programming to a server-side application development standard.
- EJBs emerged on the developer scene in 1998 when Sun Microsystems unveiled a specification for creating server-based applications using software components.
  - EJBs are preconfigured pieces of code that IT staff no longer have to build from scratch.
INTERNET-BASED SYSTEMS: Java

– They can be as simple as an order-entry form or as complicated as a virtual shopping cart that even protects shopper privacy.
– Use of EJBs can greatly enhance programmer productivity.

• J2EE (the server-side Java standard) defines a standard for developing Internet-based enterprise applications.
INTERNET-BASED SYSTEMS: Java

- Java simplifies enterprise application development by basing it on a collection of standard server-side application programming interfaces (APIs), providing a set of services to modular components, and handling many of the core functions for the applications.
Internet-Based Systems

- Web Services

- Building a Web Service
INTERNET-BASED SYSTEMS: Web Services

• The vision of Web Services is that modules of code can be assembled into services, which, in turn, can be linked to create a business process at the moment it is needed and run across enterprises, computing platforms, and data models.

• There are two development modes:
  – one is to use a Web Service someone else has developed,
  – the second to create your own from existing in-house functions.
Project Management

- Keys to Success
- Internet Project Management
- Tips for Good Project Management
PROJECT MANAGEMENT

• Perhaps the most difficult component of IT project management is keeping in mind, and under control, all the interdependencies of the numerous tasks undertaken.

• Many of the best IT managers do not have a background in IT at all, but they possess the important skills of communication, organization, and motivation.
Keys to Project Management Success

• The keys are:
  – establish the ground rules;
  – foster discipline, planning, documentation, and management;
  – obtain and document user requirements;
  – obtain tenders from all appropriate potential vendors, include suppliers in decision making;
  – convert existing data; and
  – follow through after implementation.
Internet Project Management –

• The rules of project management do not change for Internet-based systems. However, Internet projects do tend to be more collaborative and iterative than traditional IT projects.

• Web-based project management tools are appearing; in fact, all project management tools will be Web-based and support virtual teams.
Tips for Good Project Management –

• Good project management depends on:
  – proper planning, appropriate user involvement and strong visible management support, project managers with authority and time, good change management, working as a team, proper project monitoring and control, and proper project closure.
Conclusion

The Current State of System Development
Conclusion

• The traditional approach to system development from the 1960s evolved to give the process more discipline, control, and efficiency.

• It was valuable in moving programming and system analysis from pure free-form “art” to a better defined “craft.”

• The 1970s and 1980s brought techniques and tools for more rapid development and even experimental development.
Conclusion

• The 1990s brought the need for integrated enterprise systems and Internet-based systems.

• Development now focuses on the Internet, inter-organizational development, and ecosystem applications – systems where project management skills are even more important due to the complexity of the systems.