


Why is CICS Still Alive?

Dr Geoff Sharman
Visiting Professor in Computer Science
Birkbeck College

Agenda

- ♦ Middleware – the hidden part of IT
 - ♦ CICS (Customer Information Control System) track record as a middleware product
 - ♦ The future environment for middleware
- 

CICS Today

- ◆ **CICS Transaction Server V3.1** for zOS on IBM zSeries systems
 - ◆ Includes enhanced support for Web Services, application modernisation, and systems management
- ◆ **TXSeries V6.0 for AIX** on IBM pSeries systems and other platforms including Windows
 - ◆ A fully functional CICS system with no dependencies
- ◆ **CICS Transaction Gateway V6.0** for ... many
 - ◆ Connects Web to older CICS systems
- ◆ One of the most successful s/w products ever
 - ◆ *Most people in the developed world use it every day*

Why? How?

- ▶ I thought mainframes died years ago ...?
- ▶ I never see adverts for CICS programmers ...?
- ▶ I don't recall the last time I heard about about CICS ... so how can this be?

Did you do any of these today?

- ♦ Buy something in a supermarket?
- ♦ Use a mobile phone (or any phone)?
- ♦ Travel by public transport?
- ♦ Attend an entertainment event?
- ♦ Visit a hospital?
- ♦ Use electricity, gas or water?
 - ♦ *The chances are you used CICS ... 65% of transactions are processed on CICS mainframes ... or a competitor*

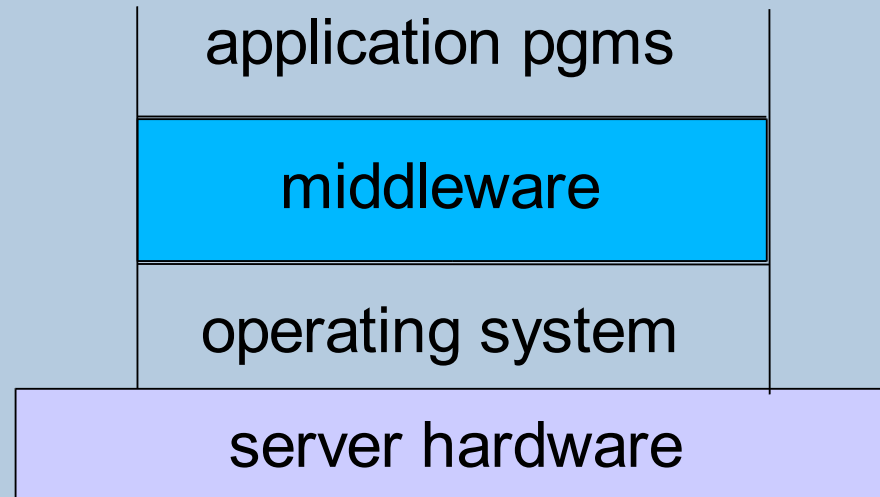
Invisible Transactions

- ♦ A huge range of daily activities involve automated transactions which are performed on *remote servers*
- ♦ The user interface is *embedded* in a supermarket checkout, mobile phone, ticketing machine etc. and is therefore *invisible*
- ♦ The number and range of transaction applications will only continue to *grow*
- ♦ And you can also do transactions on the *Web!*

So What's Different About Servers?

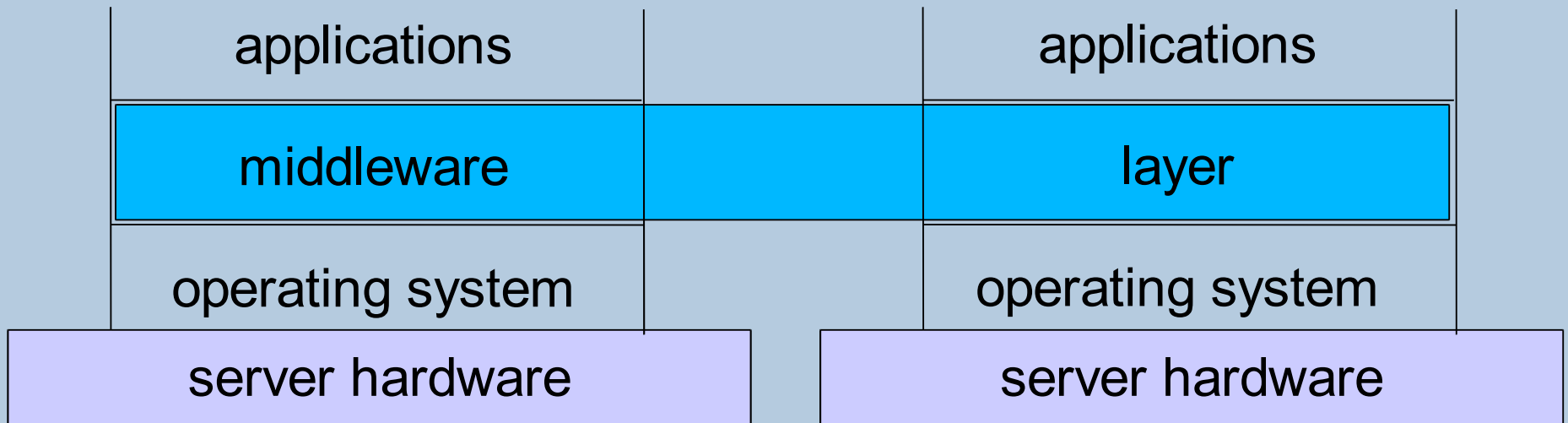
- ♦ We're all familiar with applications which run on personal systems, e.g. PCs, mobile phones, iPods
 - ♦ They offer instant access via low utilisation
- ♦ We tend to forget about *server applications*
 - ♦ Must offer instant access, but economics demand:
 - low cost per transaction, hence high utilisation
 - high scalability
 - continuous availability
 - ♦ So nearly always based on *Middleware*

What is Middleware?



- *Software layer that sits between the operating system and the application programs*
- *Provides higher **level of abstraction** than operating system – makes application programming easier*
- *May provide greater **scalability**, reliability, and availability*

Middleware may Span Systems



◆ *Middleware layer:*

- ◆ *May be a class library/subroutine library*
- ◆ *May exist on multiple physical systems*
- ◆ *May exist on different operating systems – so may become a **virtual multi-system environment***

- ◆ *May enable portable applications*


Common Middleware Services

- ◆ *Program directory service*
- ◆ *Program scheduling service*
- ◆ *Program synchronisation service*
- ◆ *Time of day/time interval service*
- ◆ *Presentation services (device class specific)*
- ◆ *Reliable messaging*
- ◆ *High performance data management*
- ◆ *Transaction commitment service*
- ◆ *Journalling service*
- ◆ *Monitoring service*
- ◆ *Authentication service*
- ◆ *Authorisation service*
- ◆ *and more...*

More on Middleware

- ♦ *Middleware* is a generic term which includes:
 - ♦ TP Monitors
 - ♦ Web Application Servers
 - ♦ Message Queue Managers
 - ♦ Remote Procedure Call
 - ♦ Various flavours of Web software
- ♦ Normally runs on servers or clusters of servers and requires *systems management*
- ♦ *CICS* is the *market leader* in middleware

How? Why?

- ◆ 3270 is dead, SNA is dead ...
 - ◆ Does anyone still run COBOL applications ...?
 - ◆ How did CICS get there?
- 

Some Highlights from History

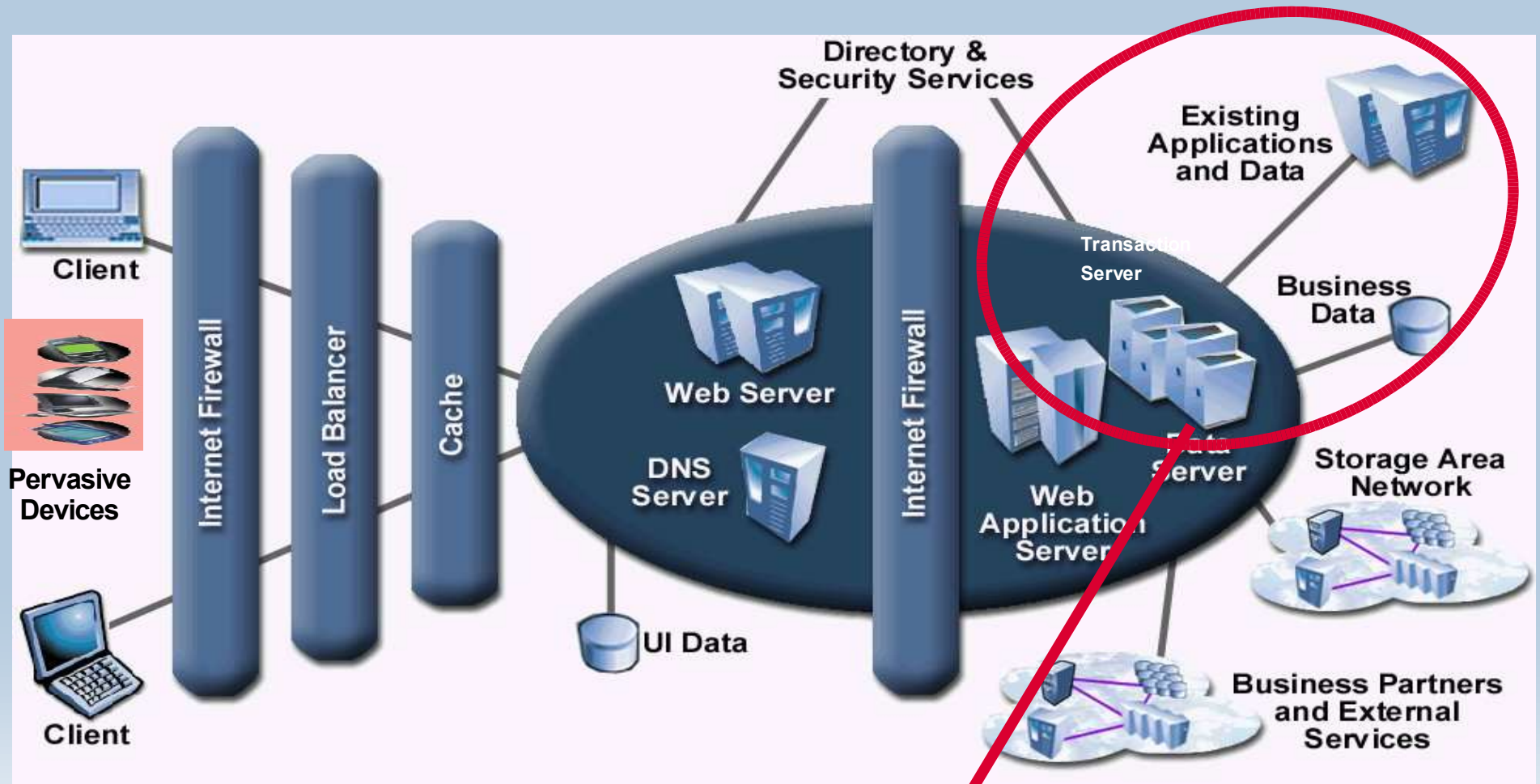
- ♦ 1968 CICS, IMS and GIS announced as IBM's first “unbundled” (i.e. priced) software products
- ♦ 1974 CICS development transferred to Hursley (in exchange for PL/I); expected to die but new command level API reverses the trend
- ♦ 1980 CICS 1.5 remote access to data and invocation of programs (ISC/MRO)
- ♦ 1990 CICS 3.1 large virtual storage via 31 bit addressing; enables very large networks
- ♦ 2005 CICS TS 3.1 supports “transactional Web”

Another Look at History

- ♦ 1968 CICS ~ 50 teletype terminals
- ♦ 1974 CICS ~ 500 3270 terminals
- ♦ 1980 CICS ~ 5000 3270 terminals
- ♦ 1990 CICS ~ 50,000 3270 terminals/emulators
- ♦ 2005 CICS ~ 500,000 browsers/emulators etc

CICS systems handle ~ 10^{11} transactions with a financial value ~ $\$10^{13}$, per day

Typical View of today's Enterprise Network



huge investments in applications based on CICS here

Some Lessons from History

- ♦ Look after your customers – protect the value of their *past investments*
- ♦ Enable *technology change*:
 - ♦ Increased processor speed, memory capacity
 - ♦ Hierarchical to relational database
 - ♦ SNA to TCP/IP and HTTP networking
 - ♦ 3270 terminals to Web browsers
- ♦ Embrace new *application development* styles:
 - ♦ Assembler, COBOL, C/C++, Java, Eclipse tools
- ♦ Watch the *competition*

How? Why?

- ♦ Well, OK, it looks good so far ...
- ♦ ... but it can't last ... can it?

21st Century Market for TP

- ◆ Most networking is based on the *Internet*
- ◆ Growing number of *consumers* use traditional devices, web browsers, and pervasive devices to access services
- ◆ IT *substitutes* for labour - enables changed business processes*, leading to falling labour costs which provide the main source of increased productivity
- ◆ *Energy* becomes more *expensive* but usage must fall:
 - ◆ IT based communication substitutes for physical travel and reduces movement of goods
- ◆ Continued *growth* in transactions processed by a global IT infrastructure

* Harvard Business Review

Market Trends

- ◆ Consumers want unique products (i.e. “market of one”, *mass customisation*)
- ◆ Most enterprises must support *multi-channel service delivery* (browser, mobile phone, retail, etc.)
 - ◆ anytime, anyplace service
- ◆ Continued pressure for *cost reduction*
 - ◆ commoditisation of IT
 - ◆ on demand acquisition and pricing
 - ◆ utility computing
- ◆ Enterprises will continue to *consolidate & outsource*
 - ◆ Consolidation often involves migrating workloads or rehosting applications, to achieve economy of scale

The Transactional Web

- ◆ Enterprises will outsource by *purchasing IT-based services* (“B2B”)
 - ◆ Using a *Service Oriented Architecture*
 - ◆ Enabled by standards for *Web Services*
 - ◆ A Service is an *encapsulated component application* – the supplier runs it as well as providing it (may also contain *non-IT elements*)
 - ◆ SOA enables dynamic, *loosely coupled integration* of services
- ◆ Enables a business to focus on its *core competency*, offered as a Web Service to other businesses
 - ◆ When there is an effective way of *charging* for Services

Where is the Business Value?

- ◆ Business value is created by *applications*
 - ◆ delivered as components or services
- ◆ Infrastructure is perceived as an *enabler*
 - ◆ deployed on clusters and networks
 - ◆ based on middleware architectures
 - ◆ invisible except at the point of delivery
 - ◆ cost must be defrayed by large scale use
- ◆ Middleware must *compete* for applications
 - ◆ Multiple languages and re-use of past investments

Summary

- ◆ CICS gained an early lead in TP applications
 - ◆ Survived by adapting to changed technologies
 - ◆ Continued to offer customer value
 - ◆ Is well positioned for the future
 - ◆ See www.ibm.com/cics/ for more information
- 