

# **Online Business Systems (San Francisco): Building a Multi-Tenant CICS Service Bureau Before CICS/VS (1973–1982)**

## **1. Scope and Position**

This document is a technical and historical recollection of Online Business Systems (OBS), a San Francisco-based IBM mainframe service bureau, focusing on the period 1973–1982. It is based primarily on direct participation and memory rather than surviving corporate documentation. It describes the computing environment, the heavily modified CICS V1 system, the WYLBUR/MILTEN ecosystem, the multi-client architecture, the people who built and maintained it, and the major production systems, especially the Princess Cruises reservation system. Where relevant, it also mentions events that predated the author's employment, but such earlier material should be read as second-hand or oral history unless otherwise noted. It also acknowledges that many essential business functions—especially operations—are described only at a high level because the author was not directly involved in them.

## **2. Hardware and Base Platform**

By 1973, OBS was running an IBM System/370. The earlier System/360 era predated the author's arrival. During this period, the system was upgraded to a 370/AP. The shop operated as a true service bureau, hosting multiple unrelated commercial clients on the same complex.

## **3. The Service Bureau Model**

OBS hosted and operated multiple production client systems concurrently. Known major clients included Princess Cruises (online reservation system), US Leasing, Levi Strauss, and Pacific Vegetable Oil. These were production, revenue-critical business systems from different industries, sharing the same infrastructure. There were additional client systems, including at least one inventory control system and a workflow/ticketing system for a shoe company that made heavy use of OBMOS-driven 3270 printing, but the company names are no longer recalled with certainty.

## **4. Pre-1973 Background (Second-Hand Accounts)**

What follows is based on later recollections and informal accounts rather than direct personal experience. By the time the author arrived in 1973, the system already existed in its heavily modified form, and any earlier projects had either been completed or retired.

It was said that, prior to 1973, OBS had undertaken at least one demanding contract involving CICS and 3270 Model 1 (12x40) terminals, and that the limitations of stock CICS of that era forced a major internal redesign. According to these accounts, Phil Humphries took the system off IBM maintenance and replaced large portions of the core, including KCP and TCP, moving away from TCT polling toward

a more event-driven architecture.

The author does not remember the client system itself, what it did, or what became of it, and it appears to have been gone by the time of his arrival. This episode is therefore best regarded as part of the shop's internal origin story rather than as a documented project history. Its importance here is simply to explain why the CICS environment already looked the way it did in 1973.

## 5. Entry Point

The author was hired in 1973 at age 24, with no prior professional employment experience, under the title 'Member of Technical Staff III', which was effectively the bottom of the technical org chart. The environment consisted of large assembler codebases, production-critical systems, and real operational risk.

## 6. The OBS CICS System

### 6.1 Baseline

The system was based originally on IBM CICS V1, predating CICS/VS. At that time, IBM's CICS was not designed for true multi-tenant service bureau use and lacked many features later considered standard.

### 6.2 Scale of Modification

This was not a lightly customized CICS. Major components were completely rewritten, including TCP, KCP, and others. The result was effectively a forked and heavily extended transaction processing monitor.

### 6.3 Multi-Client Architecture

The modified system supported multiple independent client application environments concurrently, providing logical separation while sharing infrastructure.

### 6.4 Transaction Accounting and Billing

The multi-tenant design included per-transaction accounting. At completion of each transaction, a record was written to a CICS Transient Data destination backed by a tape volume mounted at system startup. This produced a chronological transaction log used primarily for billing. At least one contract billed at a rate of \$0.05 per transaction. This made usage metering a first-class architectural feature.

## 7. Advanced Subsystems

## **7.1 Virtual Terminal Service (VTS)**

VTS provided terminal abstraction and screen management years before IBM introduced BMS, fulfilling a similar role.

## **7.2 OBS Message Output Service (OBMOS)**

OBMOS managed 3270 printers, queued output, and interfaced with HASP to allow online systems to produce spooled output.

## **7.3 3270 Terminal Support**

Full 3270 terminal support was engineered into the environment to support serious commercial online systems.

## **7.4 CICS Simulator**

A CICS simulation environment was built to allow development and testing without endangering production.

# **8. WYLBUR and MILTEN**

WYLBUR originated at Stanford University. OBS took over its maintenance and development, licensed it, resold it, and used it internally for development and operations. MILTEN served as the terminal/communications monitor for the WYLBUR environment and was named for Milten Wright. WYLBUR/MILTEN and OBS CICS were entirely separate systems with no shared code. Later, both environments were enhanced to use VTAM, but they remained independent.

# **9. Operations and the 24x7 Data Center**

In parallel with development and testing, OBS ran a continuous, twenty-four hour production workload for its clients. The data center processed online transactions and batch work around the clock, and this operational responsibility was central to the business. While the author was not directly part of the operations organization, it is clear in retrospect that the success of OBS depended as much on disciplined, reliable operations as on the engineering of the software itself. The coexistence of nonstop production processing with ongoing development and testing shaped many technical and procedural decisions, even if those details are only partially visible from a development perspective.

# **10. The Princess Cruises Reservation System**

One of OBS's most important systems was the Princess Cruises online reservation system, a real-time, revenue-critical workload implemented with assembler for the online components and COBOL for batch processing. It integrated terminals, printers, spooling, and operational workflows.

## **11. The Organization and People**

### **11.1 Executive and Administrative**

Jeff Stein was President. Linda Brisbois was his secretary. Don Eichler was an executive (exact title not recalled), and his secretary was Diana Walsh. Barry Hansen and Malcolm Foster were heavily involved in client interface and data center operations and attended to many vital operational matters.

### **11.2 System Software**

The System Software group was responsible not only for OBS CICS and the WYLBUR/MILTEN environment, but also for the IBM System/370 operating system in use over time, including MFT, MVT, VS1, SVS, and MVS, along with HASP/JES2. This responsibility included system generation (SYSGEN), installation, upgrades, performance tuning, and ongoing maintenance. Personnel in this group included: Phil Humphries, Steve Bursch, Pete Sawyer, Milt Demaray, Ken Scott, Raymond Revis, and John Shangler.

### **11.3 Client Application Development**

Responsible for client systems: Sharry Bash, Andrea Keane, Mel Bachmeier, Jerry Day, and Paul D'ana.

## **12. Phil Humphries' Role**

Phil Humphries was the principal architect and implementer of the heavily modified CICS V1 system, the multi-tenant architecture, Virtual Terminal Service (VTS), OBMOS, 3270 terminal support, the CICS simulator, and VTAM support for both OBS CICS and the WYLBUR/MILTEN environment.

## **13. Significance**

OBS implemented a production-grade, multi-tenant transaction processing environment with usage-based billing and features IBM would not standardize until years later, anticipating many ideas common in modern computing.