

# Voice of the Customer

Real-time Data Connectivity in Healthcare Using Progress DataDirect

Jim Kosloskey  
Integration Architect/Engineer, MD Anderson Cancer Center  
October 7, 2013

Sumit Sarkar  
Systems Engineer  
October 7, 2013

**PROGRESS**  
**EXCHANGE** 2013  
DISCOVER. DEVELOP. DELIVER.

# Agenda

---

- Who is UT MD Anderson Cancer Center?
- What is Cloverleaf®?
- Why does MD Anderson Cancer Center use ODBC?
- Using ODBC in Cloverleaf: General Concept
- Why Deploy This Way?
- Some Details
- More Depth for Those Who Are Interested

# Who Is University of Texas MD Anderson Cancer Center?

---

- Number one Cancer Treatment/Research Center in the USA
  - (US News & World Report Best Hospitals 2013)
- Part of the University of Texas
- Headquartered in Houston TX
  - Affiliated locations
    - Orlando, FL
    - Phoenix, AZ

THE UNIVERSITY OF TEXAS  
**MD Anderson**  
**Cancer Center**  
Making Cancer History®



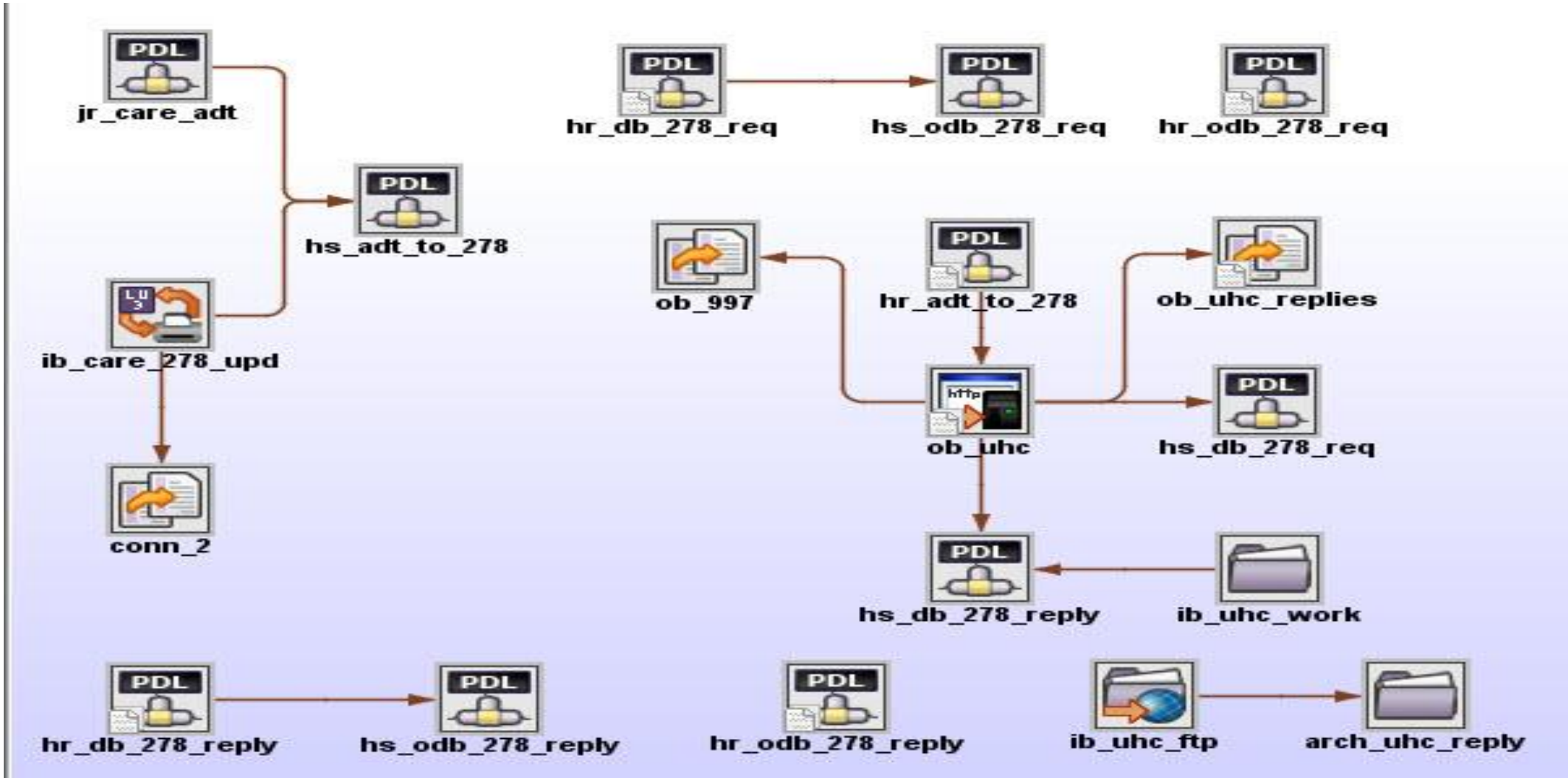
# What Is Cloverleaf®?

---

## ■ Integration Engine

- Multiple Protocols
- Multiple message/record formats
- Alerting
- Monitoring
- High Availability
- Multi-platform
- Extensible (Tcl)

# What Is Cloverleaf?



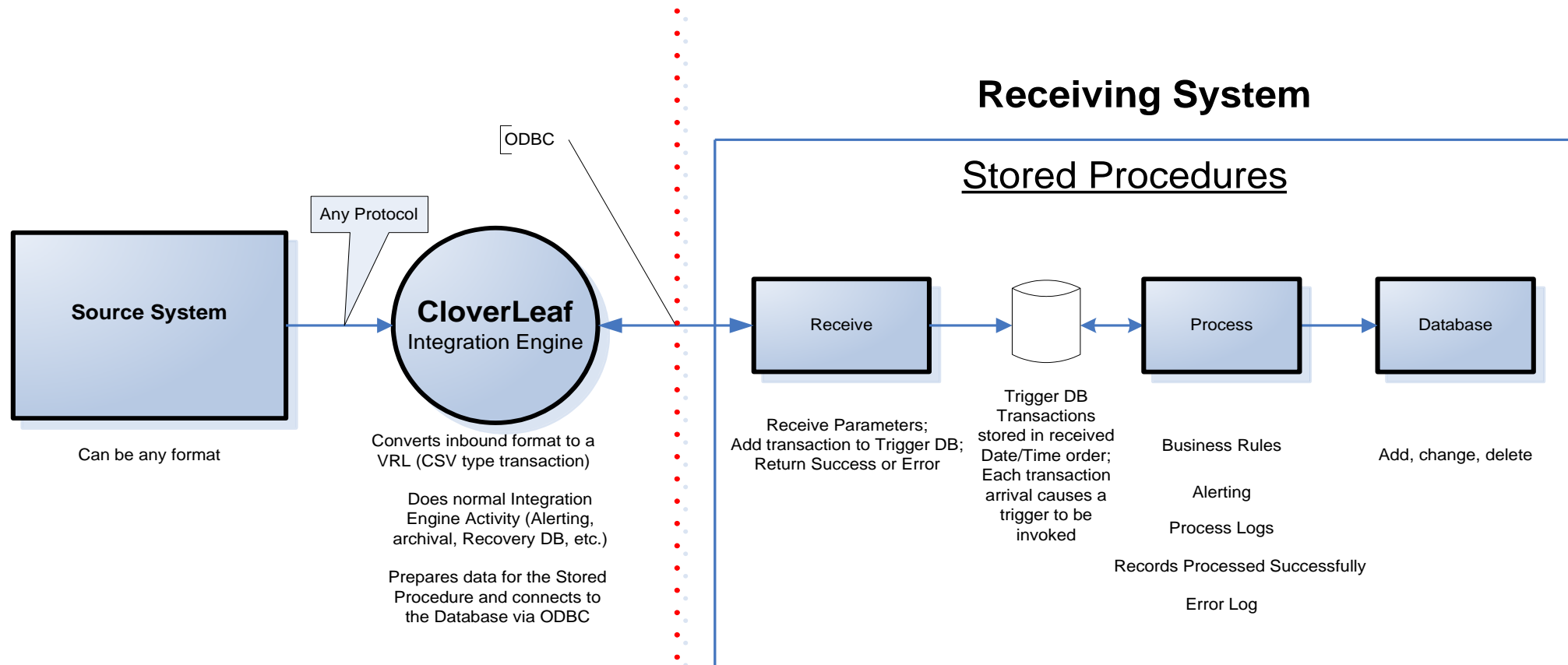
# Why Does MD Anderson Cancer Center Use ODBC with Cloverleaf

---

- Receiving system:
  - Needs information real time
  - Does not have communication capability
  - Does have a database
  - Does have the skill set to write Stored Procedures

# General Concept

## Update via Stored Procedure General Concept



# Why Deploy This Way

---

- Each toolset used for its strength
  - Integration Engine – convert protocols, transform data, exchange data
  - Data Base – Business Rules, Stored Procedures are common.
- Independence of Business Rules from data delivery
- More closely aligns with the concepts of SOA
- Simplify the work needed to be done in Cloverleaf



# Some Details

---

- A single Tcl proc used (argument driven)
- Connect String rather than maintaining ODBC.ini
- Any message/record format can be used
- Message exchange and acknowledgements via Stored Procedure parameters



**PROGRESS**