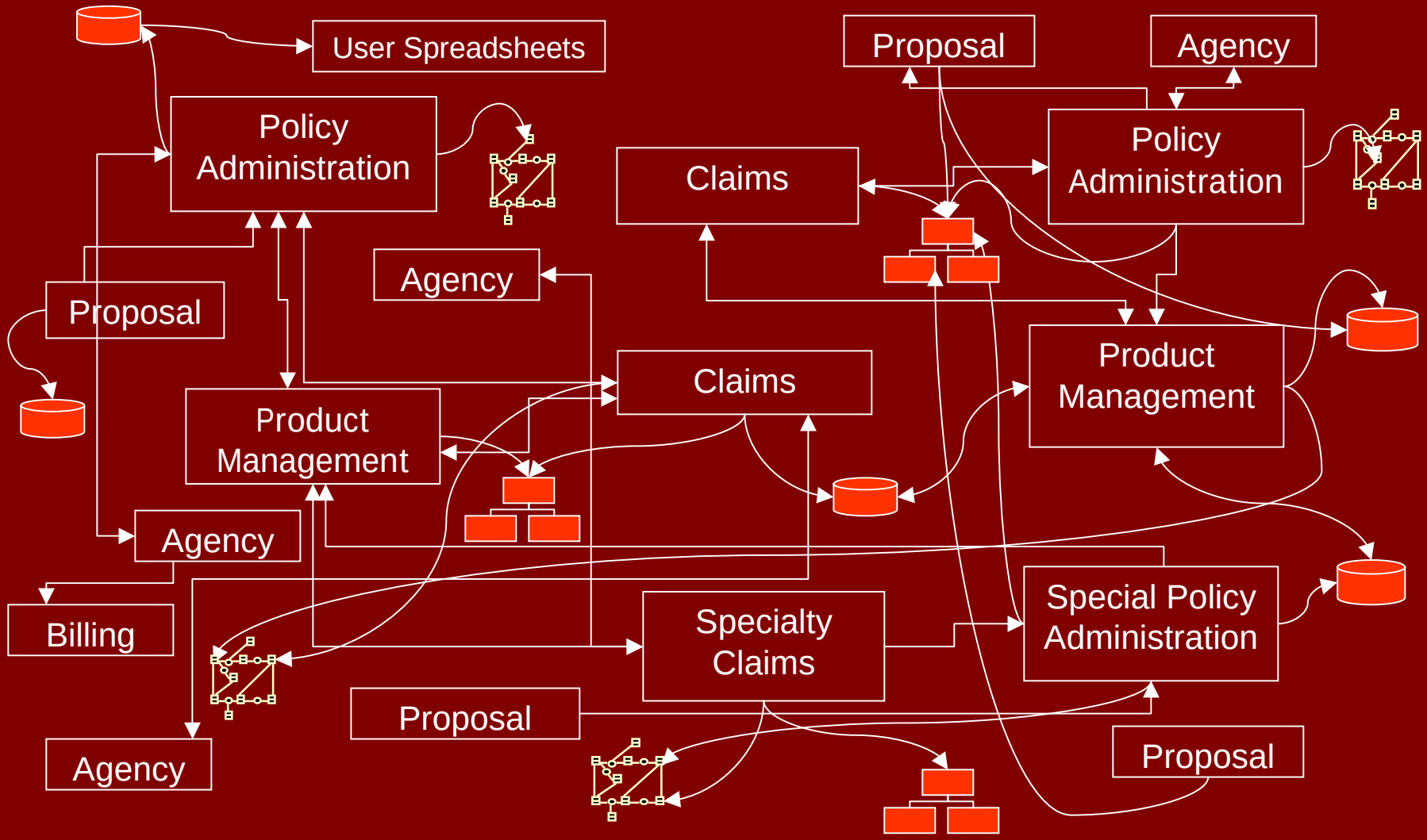
Systems Consolidation:
A Claims System Consolidation Scenario

William M. Ulrich
Tactical Strategy Group, Inc.
tsginc@cruzio.com

Background & Challenge

- Health care insurance provider grew via acquisitions resulting in:
  - Duplication in core applications and data
- Redundant functions, systems and data:
  - Increased business operating costs
  - Decreased ability to service customers
  - Reduced ability to bring new products to market
- The Challenge:
  - Effectively streamline business by consolidating common functions, customers and data

Multiple Redundant Data Structures & Systems Across Information Architecture



Represents mix of individual and group medical policies and customers

Redundancy Challenges

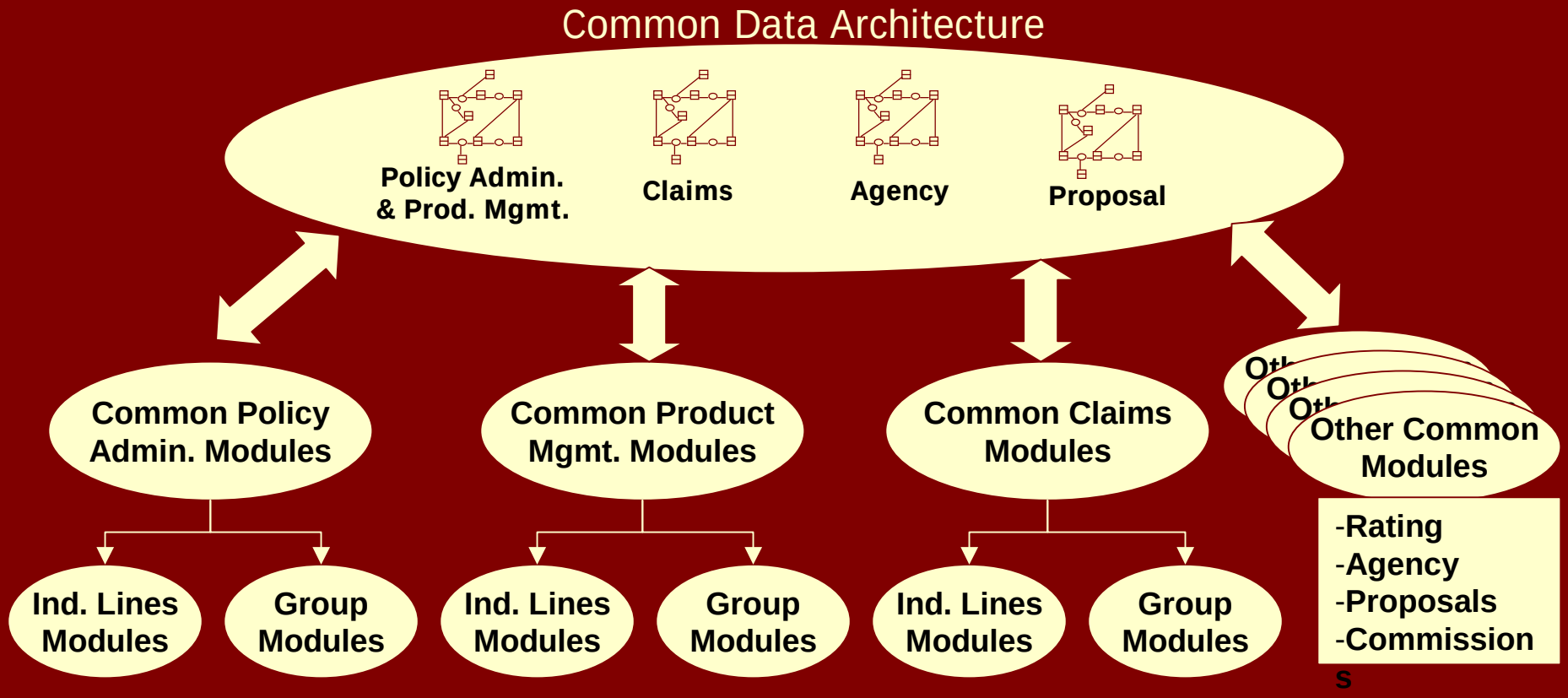
- Redundant data and functions scattered across infrastructure – For example:
  - Same customer data in different data structures
  - Claim calculations differ across claims systems
  - Requires cross-functional reconciliation
- Data and systems are not aligned around market requirements
  - Requires realigning as well as consolidating data and applications
- No one-to-one functional correspondence
  - Requires careful functional realignment

Consolidation Roadmap

- Define target data and functional architecture
- Functionally decompose redundant systems
- Develop phased consolidation plan
- Rationalize data definitions across systems of interest
- Refactor baseline systems in preparation for consolidation
- Using phased approach, consolidate and phase out redundant systems

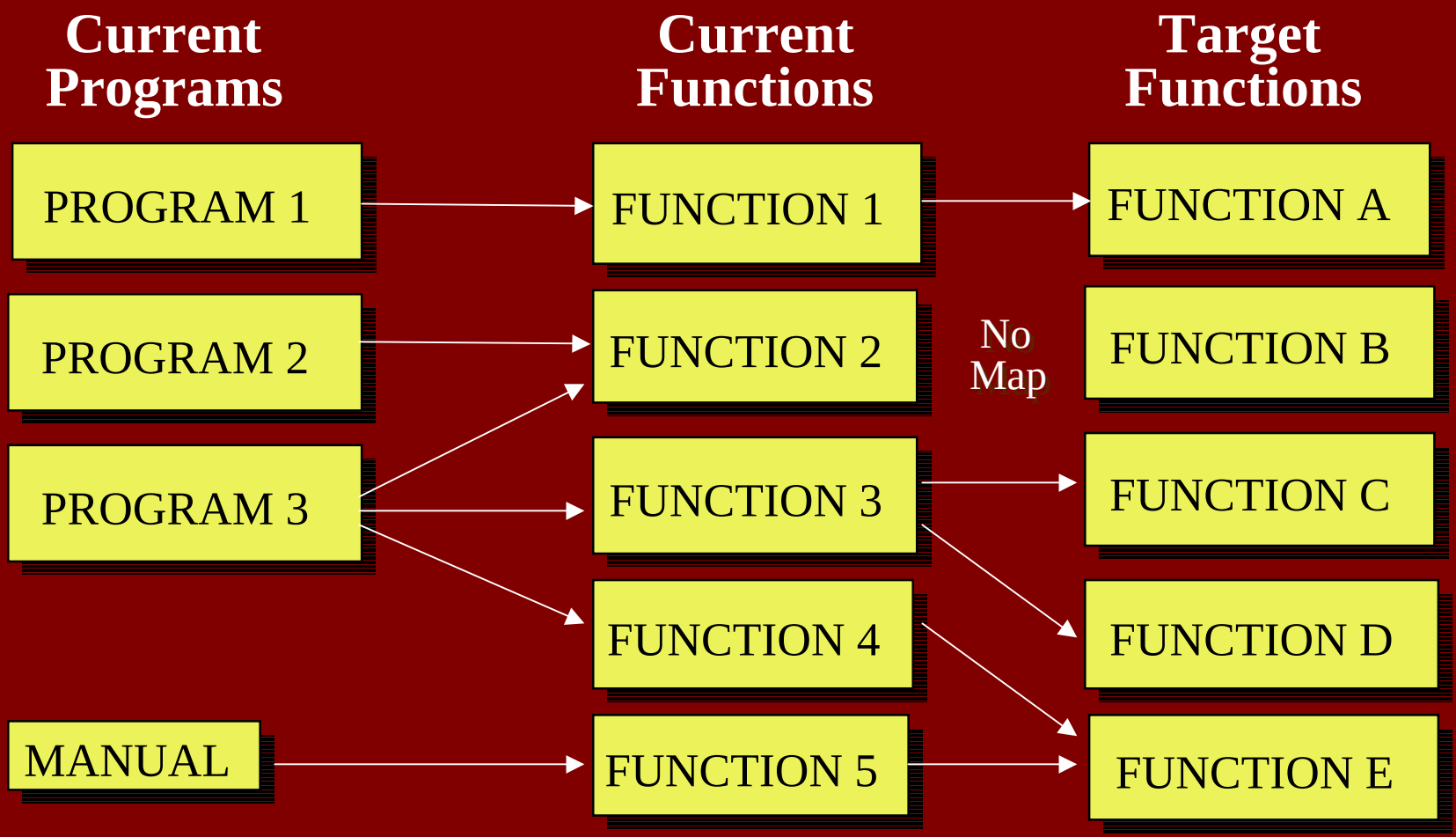
Defining End-State Data & Functional Architecture

Data architecture designed around market & business-driven requirements



Applications decompose into single function modules (i.e. components)

Program / Function GAP Analysis



Program-to-function mapping helps determine level of legacy application conformance to target architecture.

Program / Process / Function Map Example

Functional decomposition maps existing programs to functions and processes (sub-functions) that those programs implement.

Results of this analysis, which can be mapped to a target functions for the purpose of reuse analysis, provides a roadmap to more fine grain analysis, consolidation and the realignment of existing architecture.

Mapping functions to program source modules

PROGRAM SOURCE	CURRENT PROCESS	CURRENT FUNCTION	TARGET FUNCTION
IC300200	Edit Ind. Claim	Individual Claims	
IC300201	Setup Ind. Claim	Individual Claims	
IC300202	Approve Ind. Claim	Individual Claims	
IC300520	Reject Ind. Claim	Individual Claims	
IC500000	Pay Ind. Claim	Individual Claims	
PM774200	Increase Rate	Prod. Mgmt.	
PM600350	Calculate Rate	Prod. Mgmt.	
PA08040	Establish Policy	Policy Admin.	
PA0922	Add Policy	Policy Admin.	
PA08041	Delete Policy	Policy Admin.	
PM400100	Ship corporate products	Prod. Mgmt.	
PM400200	Ship corporate products	Prod. Mgmt.	
GM906600	Edit Group Claim	Group Claims	
GM906601	Setup Group Claim	Group Claims	
GM665000	Approve Group Claim	Group Claims	
MA240030	Reject Group Claim	Group Claims	
AM240050	Establish Agent	Agency Mgmt.	
AM300100	Adjust Compensation	Agency Mgmt.	
PP805000	Process Proposal	Proposal Processing	
Etc...	Etc...	Etc...	

Consolidation Approach

"Insurance Provider Architecture Consolidation Example"

REFACTORING STEPS

Step 1 Code structuring & program cleanup

Step 2 Data definition rationalization

Step 3 Baseline system modularization

Step 4 Create consolidated data architecture

TRANSFORMATION STEPS

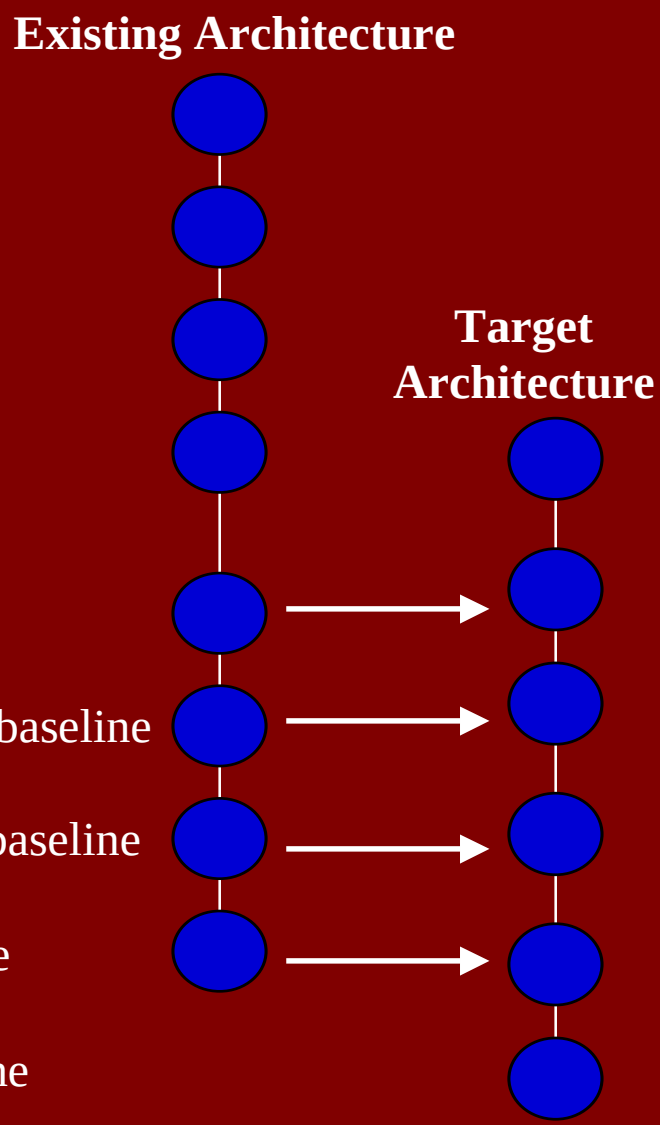
Step 5 Migrate common Claims functions to baseline

Step 6 Migrate common Product Mgmt. functions to baseline

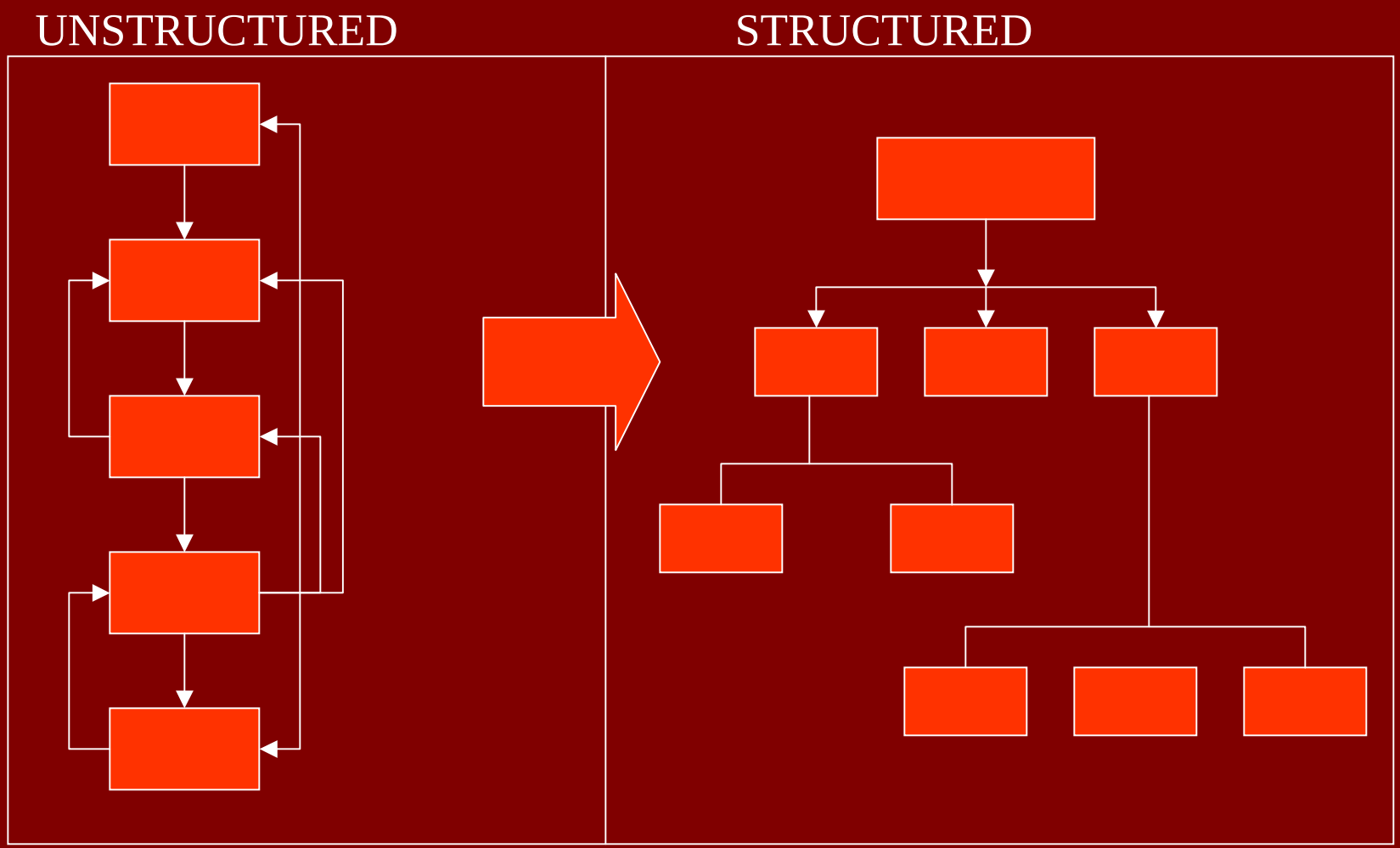
Step 7 Migrate common Policy Admin. functions to baseline

Step 8 Migrate common Agency functions to baseline

Step 9 Migrate common Proposal functions to baseline

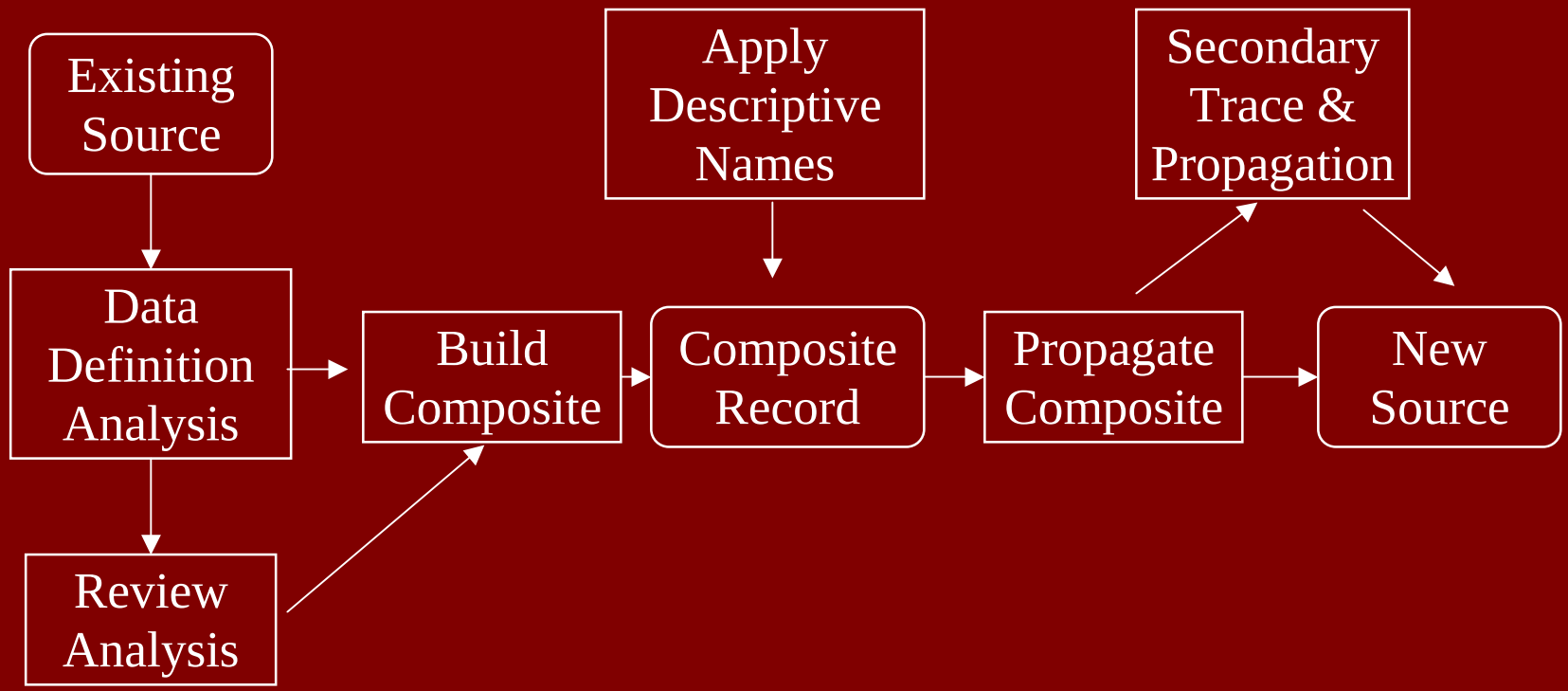


Code Restructuring Prepares Systems for Modularization

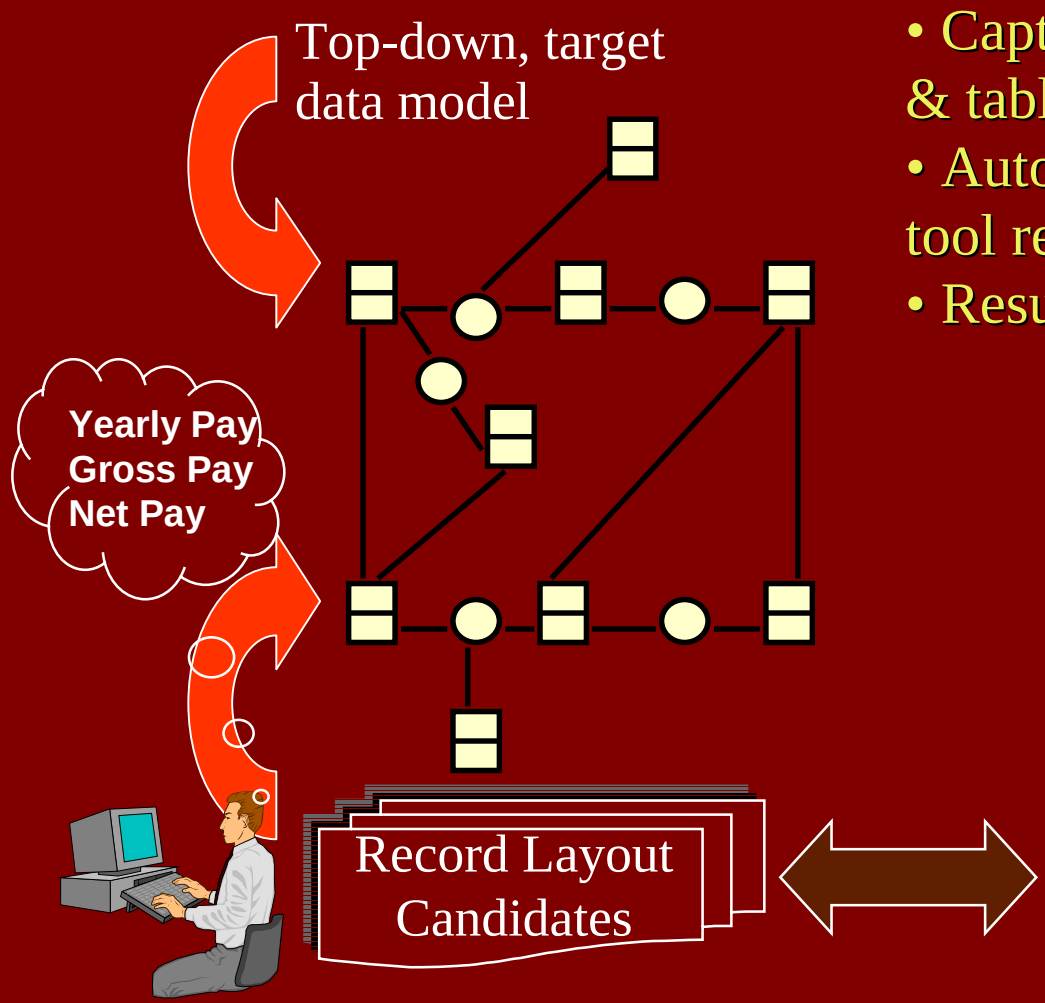


Restructuring creates single entry/single exit procedures in a program, isolates I/O logic and generally prepares a system for modularization.

Data Record/Segment/Table Rationalization Approach



Data Definition Extraction



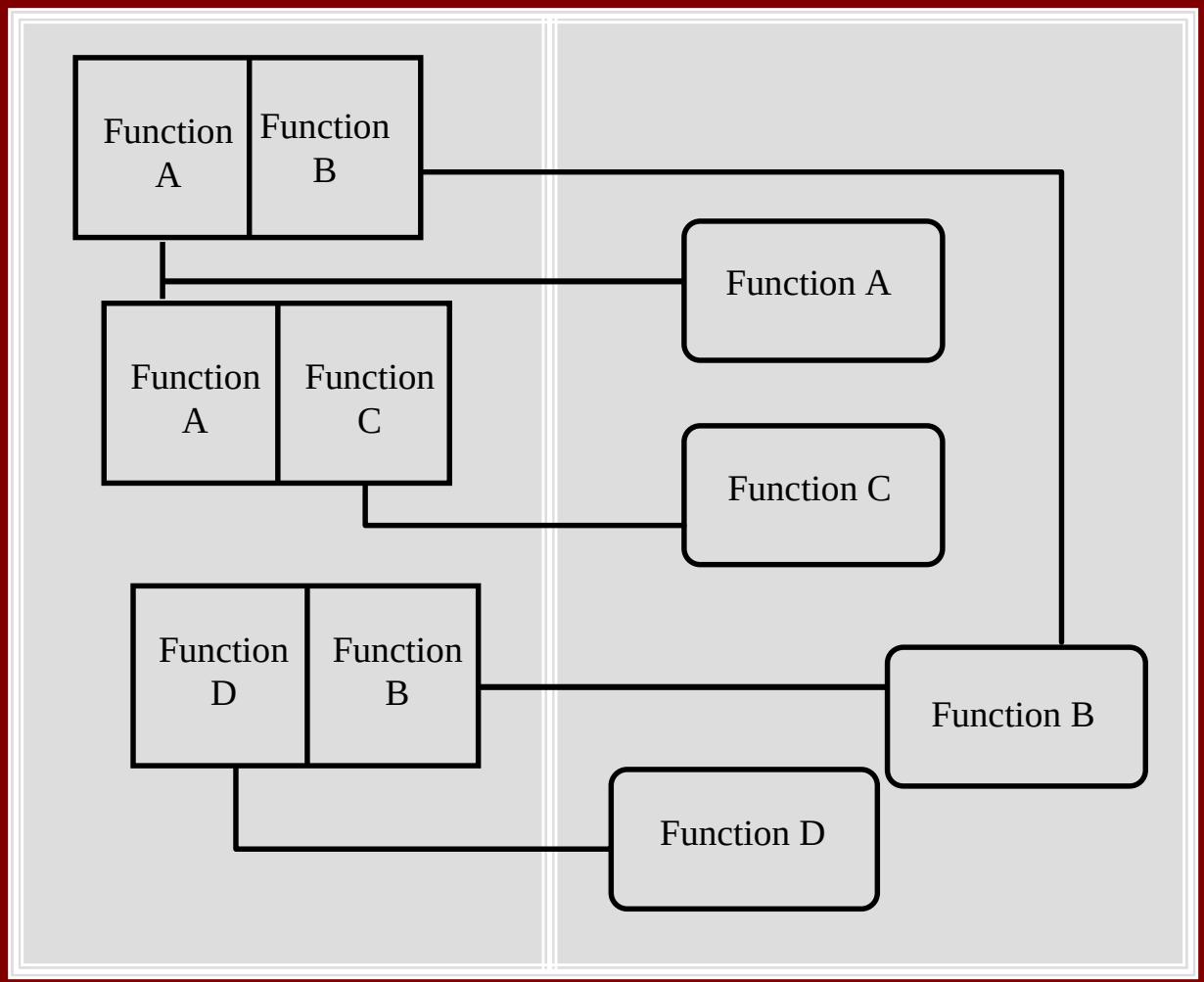
- Captures system-wide record, segment & table definitions
- Automated analysis stores results in tool repository
- Results drive business rule queries

```
MAST-REC.      (From Pgm PM22500.)

01  MAST-REC.
    05  MAST-HEADER.
        10  M-NAME              PIC X(31).
        10  M-NO                PIC 9(6).
        10  M-DOB               PIC 9(6).
    05  M-DOH                   PIC 9(6).
    05  M-SS-NO                 PIC 9(9).
    05  M-DIV-NO                PIC 9(3).
    05  M-DEPT-NO               PIC 9(3).
    05  M-MODE                  PIC X.
    05  M-YRLY-PAY              PIC 9(8).
    05  M-VAC-DAYS              PIC 99.
    05  M-GROSS-PAY             PIC 9(5).
    05  M-NET-PAY               PIC 9(5).
    05  FILLER                  PIC 9(36).
    05  M-YTD-GROSS             PIC 9(8).
    05  M-YTD-NET               PIC 9(8).
    05  FILLER                  PIC X(13).
```

System Modularization

Current System Re-aggregated System



Functions can be split out and recombined to re-aggregate application functionality. Tools provide high degree of automation for this process.

Business Logic Analysis Facilities Fine Grain Functional Consolidation

System Name: UNIT#1-GROUP-CLAIMS:

Program Name: GM906600

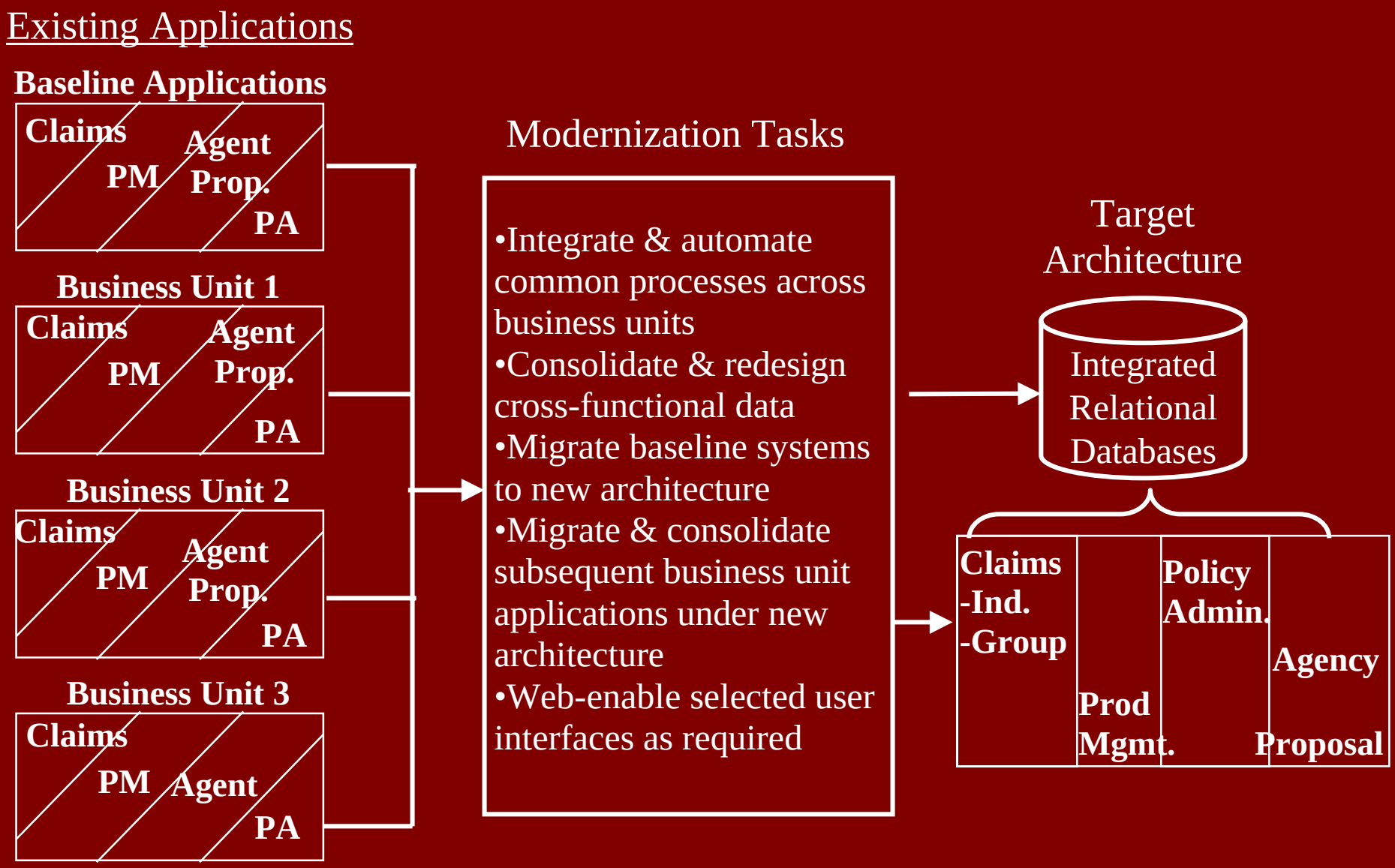
Paragraph Name: VALID-INCOMING-CLAIM-AMOUNT

Data Element Search Argument:
Element = CLAIM-AMOUNT

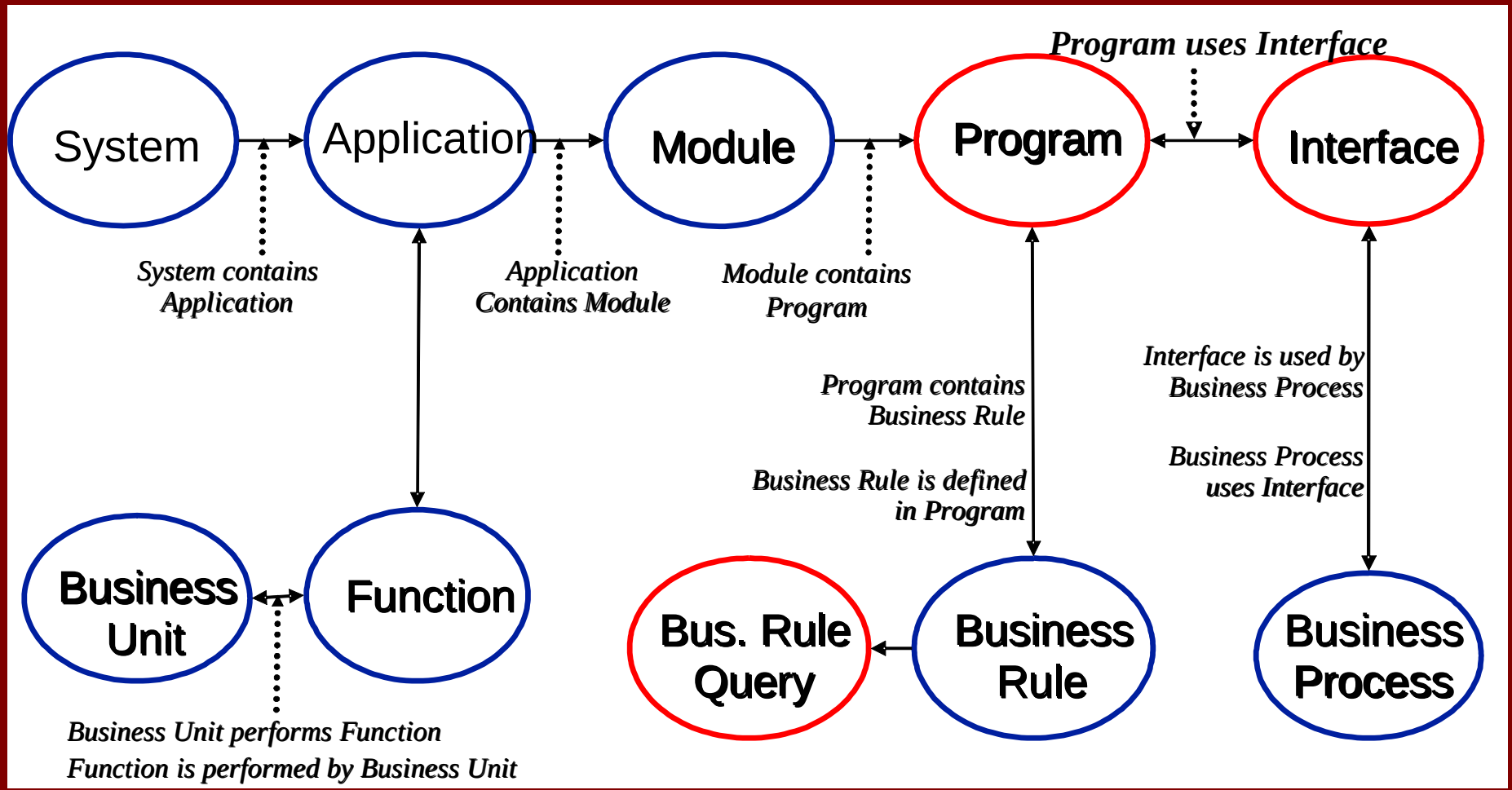
Business Logic:
IF CLAIM-NUMBER = VALID-CLAIM (IND)
AND CLAIM-AMOUNT NUMERIC
PERFORM PROCESS-CLAIM.

Modernization tools facilitate automated business logic extraction for purposes of analysis and modernization of applications.

Phased Consolidation: One Application From Many



Sample Program/Business Rule Tracking Repository



Actual model was implemented for various clients in MS Access & in tool repositories. Allows tracking of modules & functions to support modularization.

In Summary...

- Consolidation of data and application architectures is not only possible, but may be necessary
- A phased approach to modernization analysis, refactoring and transformation can be used to deliver a cost solution that manages change and risk

Systems Consolidation: A Claims System Consolidation Scenario

William M. Ulrich
Tactical Strategy Group, Inc.
tsginc@cruzio.com