



Addressing MVC's Technology Needs
Rewriting the Comprehensive & Agency Systems

State of New Jersey
Motor Vehicle Commission

July 2005



Comprehensive & Agency System Problems

The systems are 20 years old and difficult to maintain and enhance.

■ Technology Issues

- Outdated Architecture – inflexible, less open to new tools
- Outdated Database Design – limited adaptability
- Business Rules Are Hard Coded – not easily changed
- 100% Custom Coding – no opportunity for vendor developed upgrades
- Personnel Skills for Old Technology – limited and declining

■ Maintenance & Expansion Issues

- Limited Expandability - No flexibility to support significant changes like preprocessing for all transactions.
- Minimal ability to cleanly accommodate web and other interfaces
 - Web, IVR, Handheld, Kiosk
- No request is small – individual change requests and reports are often put off for larger efforts
- Limited Documentation – common, often due to overburdened staff

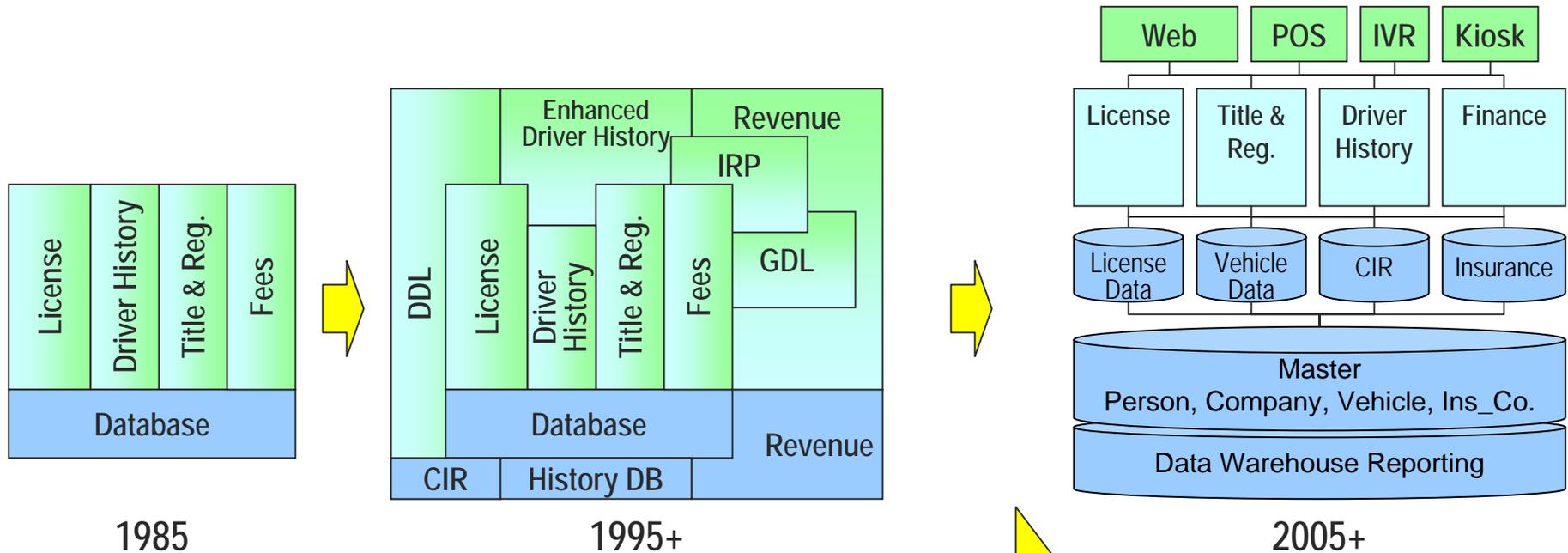
Comprehensive & Agency System Problems (continued)

- Functional Issues
 - User interface is inflexible and difficult to teach
 - Reporting capabilities are very limited
 - History Tracking and Auditability is lacking
- Process Issues
 - Definition-documentation-design processes is ineffective
 - Testing can be improved and more formalized

Architecture Team is Developing a Rewrite & Implementation Strategy

The strategy will...

- Systematically address the current Comp./Agency System issues
- Implement a new system so that MVC can move forward with its mission.



The best expansion efforts can not avoid creating a patchwork of subsystems that continually complicate future expansion.

- Reorganize Functionality
- Transform Technology
- Redefine Responsibilities

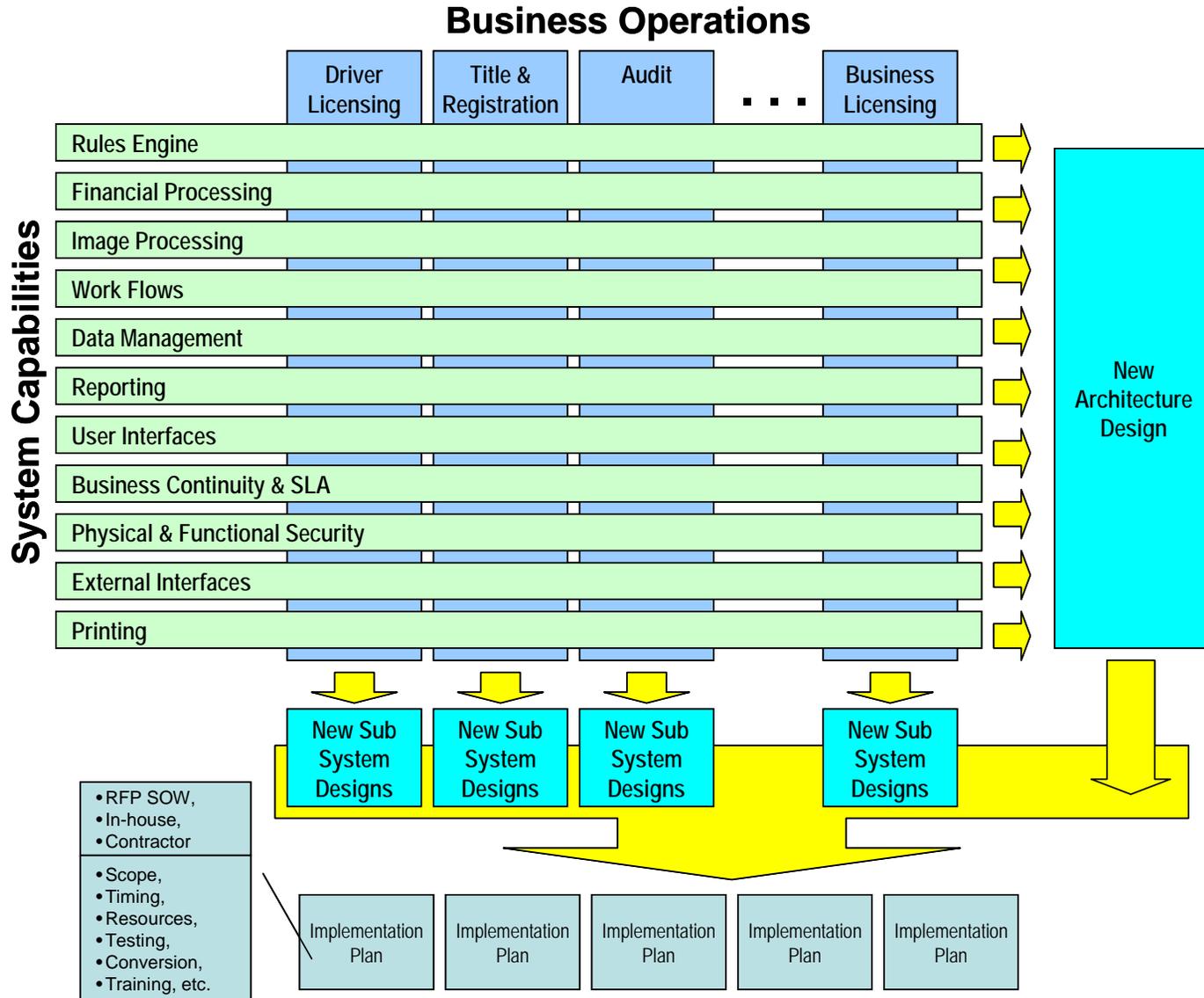
Our new system will use current technologies and development approaches.

Appendix A

Analysis & Design Approach

We are looking at both business needs and system needs.

Sub-System and Architecture designs are combined to create new implementation projects

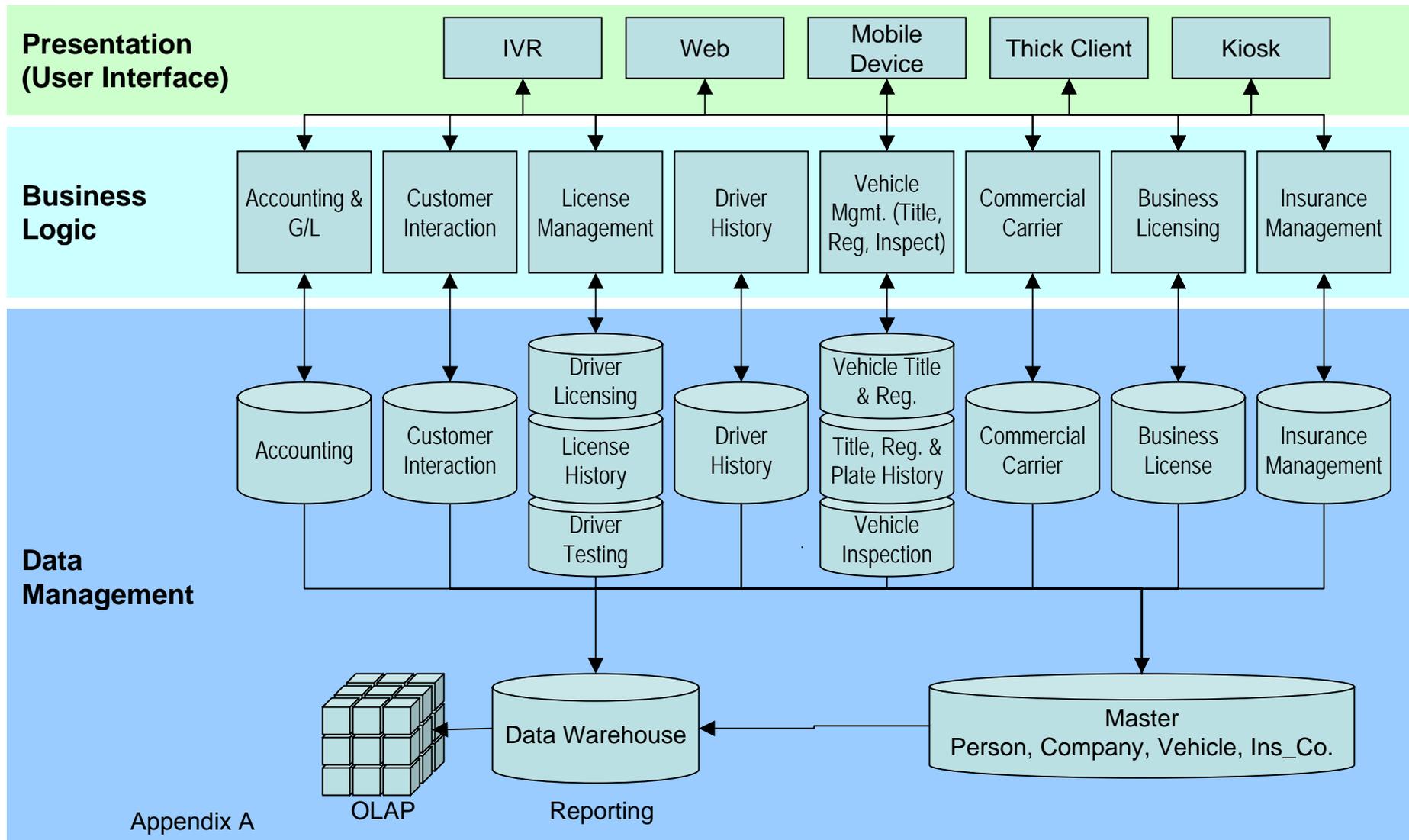


How Will We Do It? Rewriting the System...

The following approaches are part of the overall strategy for successfully redeveloping & maintaining MVC's new systems.

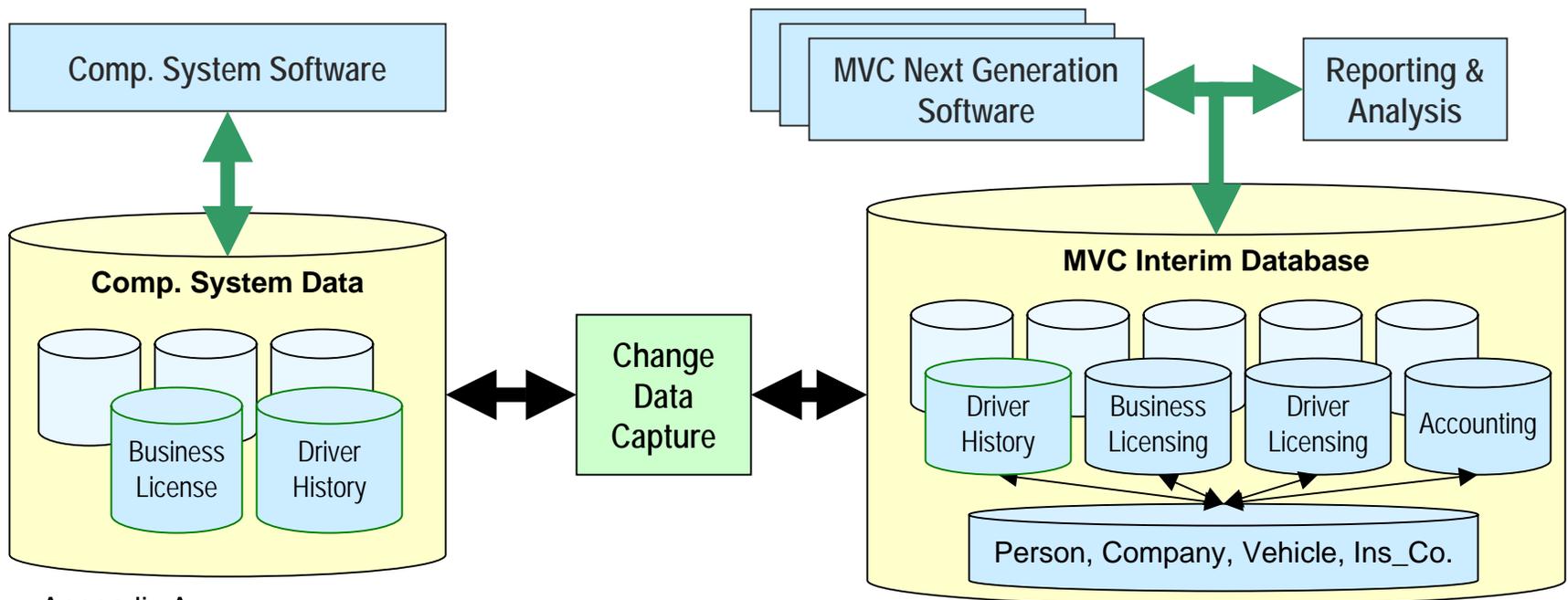
- “Partition” the subsystems into *Data, Business Logic, & User Interfaces*
- Maximize Table-driven and Rule Engine-driven coding
- Leverage Commercial Off-the-Shelf Software
- Migrate subsystems individually – not all at once
- Separate Data Management from System Management
- Implement a reporting and analysis system consistent with data warehousing project
- Develop a definition and documentation process with automated tools that will support OIT and MVC analysts
- User Groups for Subject Matter Management & Requirements Development
- Architecture & Management Oversight

Three-Tier Architecture is Flexible and Supports Multiple User Interfaces



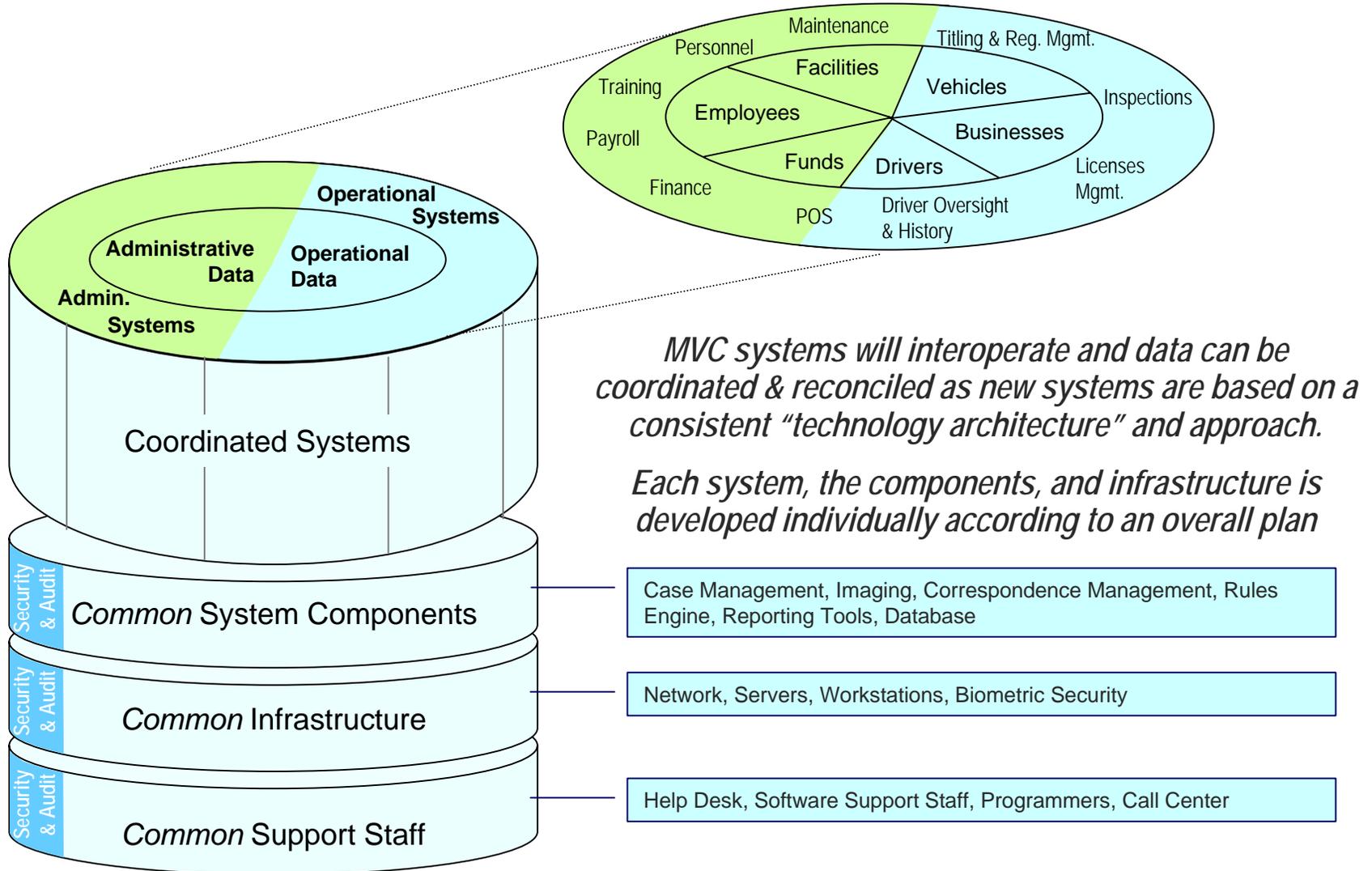
Managing MVC's Data

- A bi-directional change data capture process will allow us to phase over subsystems from old to new
- Some data will continue to exist in the 'old' Datacom database while we expand on the 'new' relational database.
- At the completion of subsystem migrations, the Datacom database and the change data capture process can be eliminated.



Appendix A

Common MVC Architecture



MVC systems will interoperate and data can be coordinated & reconciled as new systems are based on a consistent "technology architecture" and approach.

Each system, the components, and infrastructure is developed individually according to an overall plan

Architecture & Management Oversight

The rewrite is a complicated project that has many oversight needs.

- Orchestrate Many Projects & Subprojects
- Maintain A Strategic Vision
- Ensure Short & Long Term Successes
- Ensure Timely Resolution of Issues
- Identify & Manage Risk
- Coordination of Internal & Vendor Efforts
- Manage & Refine Technical Architecture

• Scope,
• Timing,
• Resources,
• Testing,
• Conversion,
• Risk
• Training, etc.

