Actionable Strategies

Case Study: Smart Grid Enterprise Architecture

Client Profile

The Central Power Corporation of Vietnam is a state-owned power distribution utility that also operates some hydropower generators and EV charging infrastructure. It is a subsidiary of Vietnam Electricity (EVN) which is headquartered in Hanoi. EVN and the Electricity Regulatory Authority of Vietnam (ERAV) set national level directives for technologies including Cybersecurity.

CPC has a reputation in the Vietnamese energy sector for innovation. The organization has invested in IT systems for workflow automation. Substation automation is world class. ERAV noted the advances made and EVN has adopted CPC applications for enterprise use. EVN has also increased the capital projects entrusted to and overseen by CPC.

Project Background

Despite the success of individual applications, CPC was severely constrained in building enterprise

applications as well as integrating new applications with existing software. Supporting a growing and transforming business was an increasing technology challenge.

Business drivers for the project included:

- Growing customer base
- Increased electricity demands
- Pressure to improve the reliability and quality of supply and service

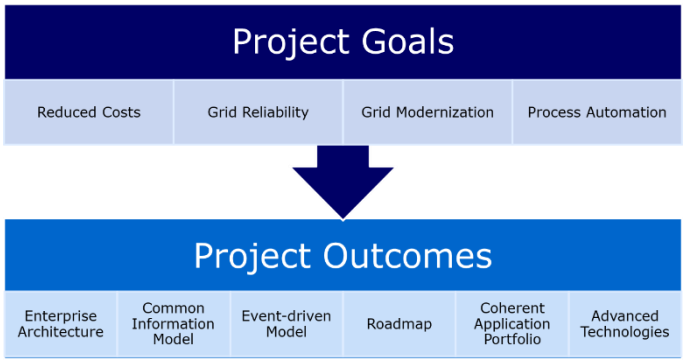
Technical drivers included:

- Business process automation
- Lack of an enterprise architecture
- Need to incorporate power system operations and management functions

CPC's stated objective was to develop an overall enterprise architecture, IT system infrastructure, and smart grid development strategy that includes an implementation plan that will improve operations and business processes. Targeted benefits included reduced system losses, improved management and operational efficiency, and improved quality of supply and electricity services to CPC customers.

Strategic Alignment

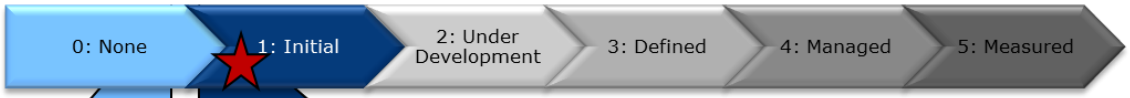
As part of project inception for the Enterprise Achitecture, technical work was directly aligned to project goals. This allowed the team to maintain focus and prevent "scope creep" in an already large and complex project.



Maturity Assessment

While there were effective point solutions built internally, such as automated process approvals that included a mobile app, software development was not a mature function. This fractured application portfolio reflected the overall state of IT.

- Lack of governance of IT ranged from the strategic including business alignment to managerial (e.g., data management) down to ongoing management of service delivery
- IT processes were ad-hoc and there was a lack of standards for technologies, methods and tools
- Dearth of enterprise scale IT underpinnings for integration, data management and standardized components forced developers to maintain "their" applications because of the different technologies
- Bias towards building rather than buying led to silos of independent applications including redundant functionality that is available in Oracle ERP, for example.
- Lack of technology risk management, notably continuity of business and information security, meant that applications were not resilient.



1. No recognition for the need for Enterprise Architecture. Success depends on individual efforts
2. Enterprise Architecture Development methods ignored
3. Business requirements specific to a problem
4. Stakeholders limited to users of the solution
5. Product focused
6. Specifications delivered after the solution!
7. Governance seen as bureaucracy.
8. Architecture projects considered to be time wasting – let's get down to that code!!!!
9. Legacy management is spiraling out of control
10. Benefits not measured

1. Processes are ad hoc and localized. Some EA processes are defined. There is no unified architecture process across technologies or business processes. Success depends on individual efforts
2. EA processes, documentation and standards are established by a variety of ad hoc means and are localized or informal
3. Minimal, or implicit linkage to business strategies or business drivers
4. Limited management team awareness or involvement in the architecture process
5. Enterprise Architecture under development
6. The latest version of the Enterprise Architecture documentation is on the Web. Little communication exists about the Enterprise Architecture process and possible process improvements
7. No explicit governance of architectural standards.
8. No Enterprise Architecture projects
9. Legacy architectures not subject to Architecture Change Management
10. No benefits achieved from the EA Program

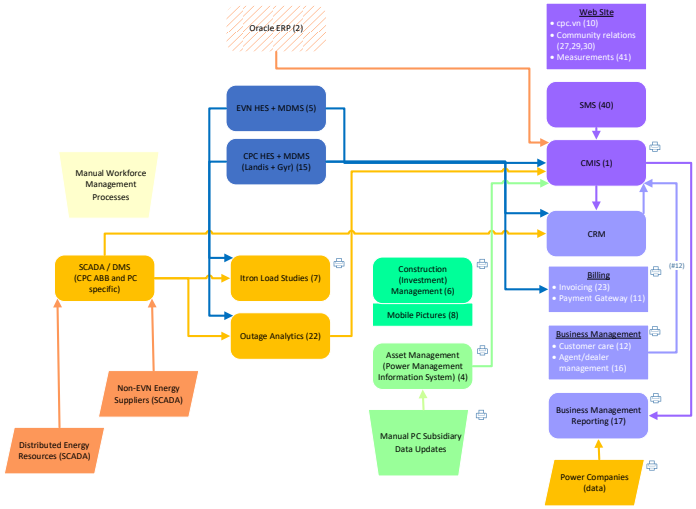
The Enterprise Architecture Maturity of CPC was assessed as at the Initial stage on the Open Group Maturity Model. The client agreed that it was at the very beginning level of maturity in the Initial stage and just beyond having no mature practices.

Existing Integration

Point-to-point integration led to the expected issues.

- Brittleness (changes to interfaces cause breakage in applications)
- Delays and cost overruns when updating existing applications
- High implementation costs for new application integration
- Accelerating maintenance costs

Operational applications were integrated via the point-to-point interfaces in the following diagram.



Reporting was coded into individual applications with the majority of the reports embedded into the applications with the printer icons. Back office applications were similarly siloed. Reporting errors

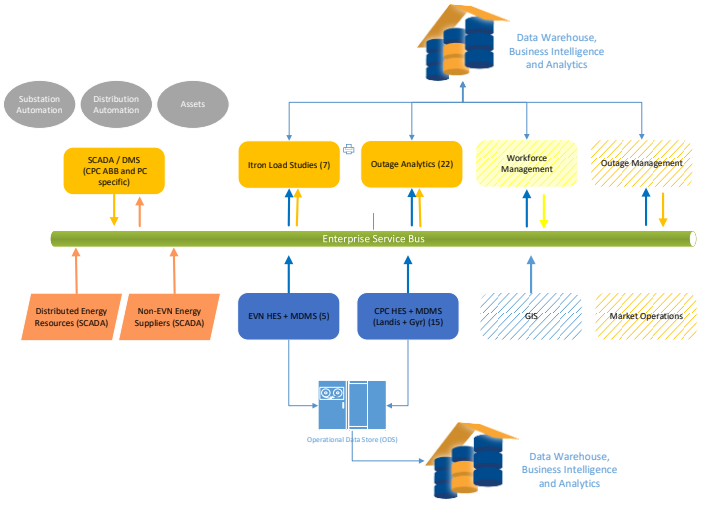
due to data issues were cited as critical to key business users.

Target State Architecture

The target state architecture applied the following core architectural principles.

- **Integration via service bus**: applications integrate with each other via the enterprise service bus
- **Event-driven**: applications publish relevant events on the bus in near real-time
- **Common Information Model**
  - IEC CIM standards are applied to semantically consistent enterprise data
  - CPC-specific information is conformed to a semantically consistent master data model and transactional data definitions
- Single source of truth: shared data is stored in the data warehouse

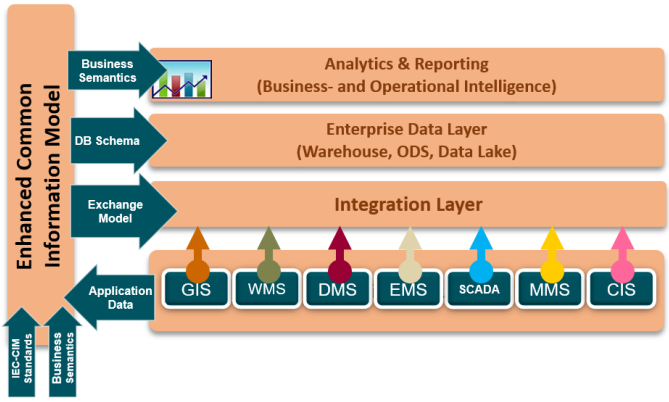
The to-be state of operational applications appears below. Back office applications would also use the Enterprise Service Bus and Data Warehouse.



Target Results

The major benefits create a technical foundation to enable CPC to build out Smart Grid functionality over the coming decades.

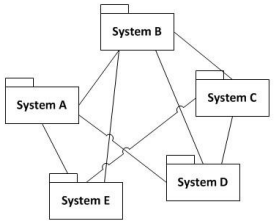
Enterprise Data Model



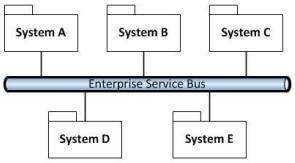
The planned enterprise data model provides:

- Consistent semantics which can be accelerated using the IEC Common Information Model which brings the ancillary benefit of the application of global standards
- Consolidation of data into a shared repository (data warehouse, ODS, data marts)
- Data governance and data quality management over enterprise data
- Geospatial data stored in a GIS

Loose Coupling



Tightly coupled – point to point



Loosely coupled with ESB

Applying a loosely coupled architecture provides many benefits to CPC in accelerating the delivery of applications and end-to-end solutions.

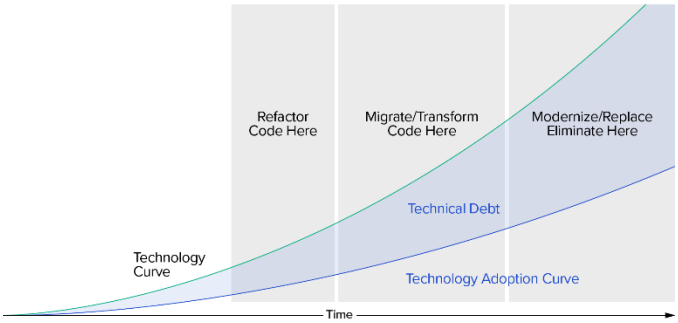
- Reduce development and maintenance costs related to integration
- Integrate third party services readily
- Provide cost-efficient scalability
- Promote agility in development of integrated systems
- Enable resilience from a distributed architecture

The best practices of ESB and CIM have been proven in other markets and the benefit documented. CPC has received a number of case studies illustrating the positive outcomes of the approach.

Retirement of Technical Debt

CPC has accumulated Technical Debt by building point (single purpose) applications to meet individual needs. While this provided value to a specific stakeholder, the ability for these applications to participate in an integrated portfolio servicing the CPC enterprise was diminished.

Technical debt increases as adoption rises which increases the number of integration points and data sharing requirements. These brittle interfaces drive up operating costs as the cost to enhance the applications also rise.



Retiring technical debt will mean that it no longer "crowds out" investment in new functionality. A technical model was provided to CPC to evaluate portfolio candidates and their disposition:

- Remain as-is
- Retire as no longer needed
- Rewire from point-to-point interfaces to ESB and the Data Warehouse (as a precursor to refactoring)
- Refactor under the Enterprise Architecture
- Rewrite using a CPC standard language, utilizing CPC tools and conforming to the EA
- Replace with a new or existing application