Grid Architecture Work Products

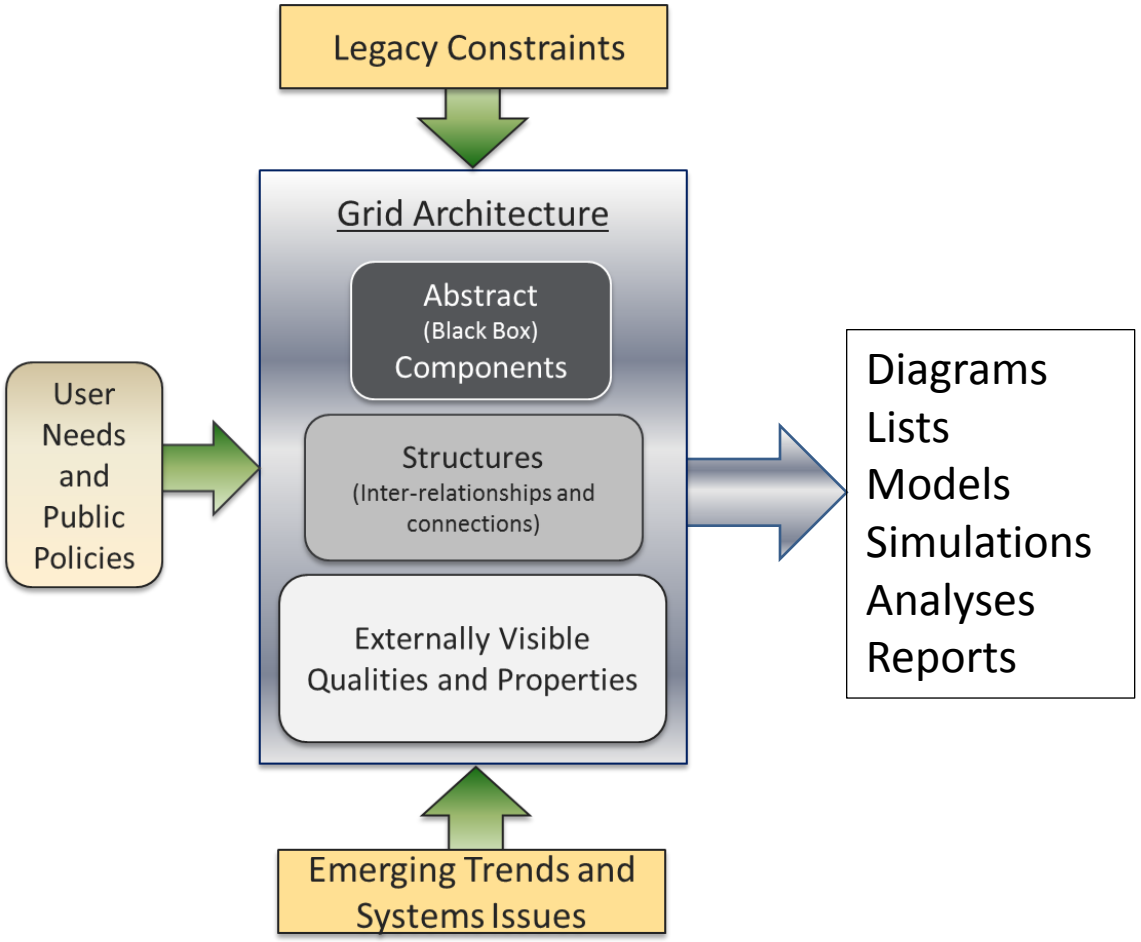
Pacific Northwest
NATIONAL LABORATORY

Purposes of the Work Products

- The architecture development process generates a variety of work product documents
  - Some are documentation of inputs (requirements, constraints)
  - Some are internal intermediate works (analyses, simulations)
  - Some are deliverables for use by stakeholders
- The primary tangible outputs are structures and component classes with support detail
- The ultimate purpose is to enable a shared stakeholder vision of the grid, so many architectural views may be generated to aid in managing complexity

Grid Architecture Development Process

- Work products document both inputs and outputs of the process
- Various analyses are created as intermediate products



Work Products

- Inputs
  - User Requirements and Public Policies
  - Emerging Trends and Constraints Lists
  - Reference Models and Systemic Issues Lists
  - Use Case Documents
  - Architectural Bases and Principles List
  - Architecture and Industry Technical Glossary
- Outputs
  - System Qualities, Properties, and Elements Mappings
  - Component class models and external properties
  - Structures and external properties
  - Validation Studies and Analyses
  - Reports and Presentations

The Core Architect Team determines the exact list of work products to be created for any specific architecture project.

INPUTS

User Requirements and Public Policies

- Word document listing and describing the requirements gathered from stakeholders and the set of relevant public policies used as inputs to the architecture process
- Reference materials as appropriate
  - Policy documents
  - Requirements studies
  - User surveys and reports
  - Related industry white papers
- Compilations of stakeholder interview/focus group discussions comments

Emerging Trends & Constraints

- Two lists, one for trends, one for constraints
- Emerging Trends
  - Typically a spreadsheet or slide deck
  - Trend name, detailed description, comments on significance/impact
  - As many as needed, but typically 10-20
- Constraints
- Also a spreadsheet or slide deck
  - Name, description, comments, and references if needed
  - As many as are relevant

Reference Models, Systemic Issues, As-Is Depictions

- Reference models are depictions of a problem domain from a particular point of view
  - Diagrams
  - Text to explain PoV (context) and diagram contents
  - As many as needed; architect team decides
- List of Systemic (Cross-Cutting) Issues
  - Derive from reference models and knowledge of constraints, relevant technologies, and
  - Typically a spreadsheet or slide deck
  - Issue name, detailed description, comments on significance/impact
  - As many as needed, but typically 30-50
- As-Is Grid Depictions
  - Architectural representations (structure/components and detail) for existing aspects of the grid where needed

Use Cases

- Unlike design processes, in architecture development, use cases are primarily for conceptual testing of proposed architectures
- Typical use case format
- As such, the detail level and granularity of these use cases is less than for typical design processes
- It is often useful to create use case *scenarios* involving multiple activities in order to help capture couplings and interactions – this makes them inherently less granular than typical design use cases

Architectural Bases and Principles

- List and description of formal bases and underlying principles used in the development of the architecture
- Provides part of the foundation for conceptual integrity
- Word doc plus reference materials

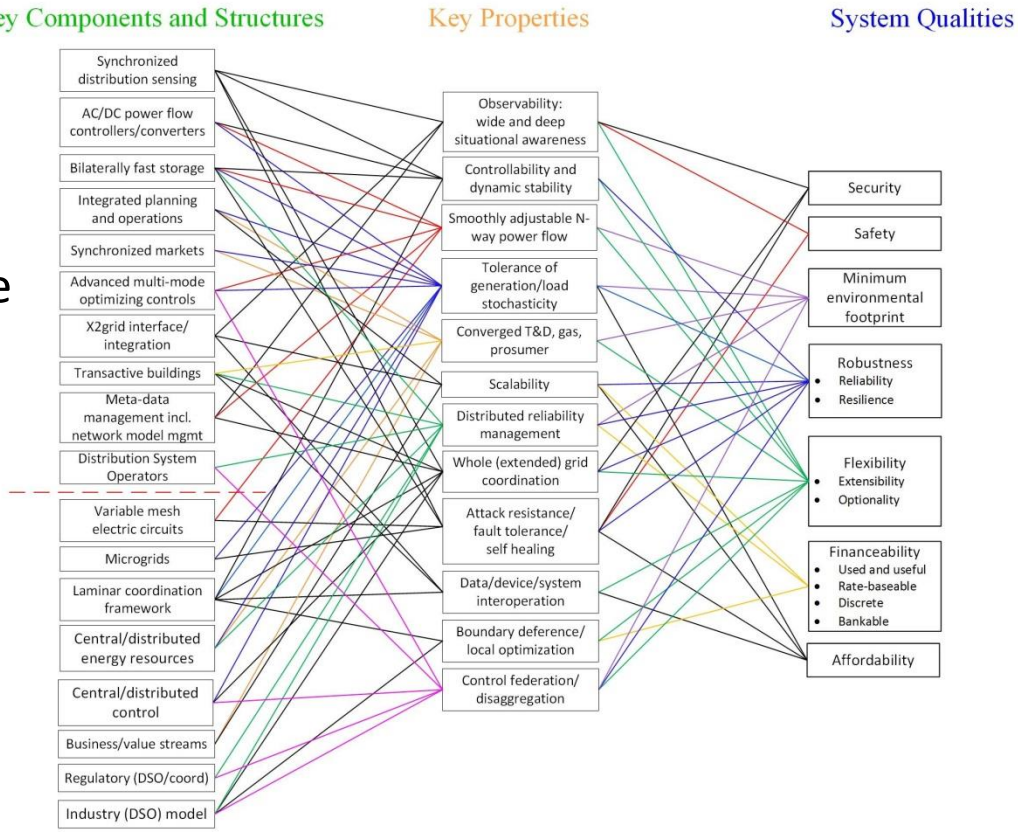
Technical Glossary

- List of architectural terms and industry technical terms with definitions
- Use to establish a common language for stakeholders to discuss grid architecture
- Word doc

OUTPUTS

System Qualities, System Properties and Architectural Elements Mappings

- Mapping diagrams with drilldown details
  - Three layer (tri-partite graph) map
  - Define each box (quality or property) and each mapping line (with numerical attribute)
- Properties-Qualities mapping is done early in the project; components and structures are mapped later



Component Classes, External Properties

- Component class models are abstractions; externally visible properties matter; internal implementation must remain hidden
  - Defined function, interfaces, and externally visible properties
  - No representation of how the component works or is implemented
- Do not over-abstract; this conceals important information
  - Demand Response is NOT a form of storage
- Formal text descriptions are fine; Word, Excel, or slides; some require diagrams and so Word or slides preferred
- ADL/SysML/UML not recommended (but prefer SysML to UML or ADL)
  - Most Grid Architecture stakeholders cannot consume
  - System (not IT) architects moving away from these anyway

Structures, External Properties

- Graphical depictions of structures with drilldown information
  - Power circuit bus and branch line diagrams
  - Industry structure diagrams
  - Regulatory scope diagrams
  - Application taxonomy and component models
  - Network connectivity diagrams
  - Measurement and control block/flow diagrams
  - Coordination skeleton diagrams
  - Value/intelligence/cash/energy flow diagrams
  - Network dependency/convergence diagrams

Validation Studies and Analyses

- Reports on technical support activities used to verify architectural concepts and approaches
  - Analyses
    - Two down/one up feasibility analyses
    - Theoretical and analytical studies on components and structures
    - DSM analysis
    - Use case concept test results
  - Simulations
    - Multi-structure co-simulations
  - Stakeholder reviews
    - End user qualities-based reviews (problem domain)
    - Developer/operator properties-based reviews (solution domain)

Architectural Views

- A view is a set of diagrams and drilldown detail depicting some aspect of the architecture from a particular point of view
  - Stakeholder interest
  - Industry or regional segment
  - Notional approach
- Views can depict alternate approaches but should have common conceptual integrity
- The entire set of views is the architecture.

Is There Just One Architecture for the Grid?

- There are an unlimited number of possible architectures; they are not all equally strong so part of the grid architecture process is to weed out the weak and identify the strong
- In the US, the diversity of the utility industry makes it impossible to have a universal one-size-adapts-to-all architecture. In addition, there are diverse competing approaches to various grid problems.

  We use multiple *views* to accommodate appropriate regional, industry segment, and notional variations while maintaining *conceptual integrity* across the set of views and so in that sense we construct a single (multi-view) architecture.

Reports and Presentations

- As needed to explain architecture or validation analyses to stakeholders
- No specific format
- Word
- PowerPoint