A Brief History Of The Sheaf Data Model

David M. Butler

Limit Point Systems, Inc.

Outline

- objective
- Limit Point Systems, Inc. (LPS)
- data model paradigm
- relational data model
- Sandia National Labs 1987-1989
- fiber bundle data model
- Advanced Strategic Computing Initiative (ASCI) 1998-2003
- sheaf data model

Objective

- use historical narrative to
  - introduce Limit Point Systems
  - introduce basic concepts of sheaf data model

Limit Point Systems, Inc. (LPS)

- consulting and contract software development organization
  - founded 1985
  - 6 employees (3 PhD, 2 BS, 1 AA)
- corporate focus
  - integration of scientific computing into industrial workflows
  - "systems for processing scientific data"
- customers in a wide range of industries
  - oil and gas
  - medical imaging
  - manufacturing
  - defense

➔*key to successful system development is the data model*

➔*data model R&D an on-going interest at LPS*

Data Model Paradigm

- data model is a "theory of data"
- data model specifies
  - class of mathematical objects
  - operations on those objects
  - constraints valid instances have to satisfy
- languages, libraries, tools based on data model
- applications are developed on top of tools

➔*best known example is relational data model*

Relational Data Model

- E.F. Codd, Comm. of the ACM, 1970

  "A Relational Model for Large Shared Data Banks"

- objects
  - relations on sets
  - table metaphor

$A = \{a_1, a_2, ...a_m\}$
$B = \{b_1, b_2, ... b_n\}$

A	B
a_2	b_{36}
a_1	b_2
a_{15}	b_0

- operations
  - relational algebra & calculus
    - select rows
    - project columns
    - join tables
    - etc.

➔*revolutionized business data management*

Revolutionized Business Data Management

- enabled much more sophisticated interaction with data
  - interactive queries using SQL
- enabled data sharing between diverse applications
  - SQL became "intergalactic dataspeak"
- success as integration platform enabled by mathematics
  - objects of data model are mathematical abstractions
  - shared by many business applications

➔*did not revolutionize scientific data management*

Did Not Revolutionize Scientific Data Management

- scientific data is overwhelmingly (physics) field data
- relational model doesn't support field data very well
- relational model not used for field data

Scientific Data Is Overwhelmingly (Physics) Field Data

- physics field is a function of several variables
  - *F*(*r*)
- independent variables are coordinates in some space and/or time
  - *r* = (t) or (x, y) or (u,v,w) or ...
- dependent variable is some physical property
  - porosity (scalar): $F(r) = s(r)$
  - flow rate (vector): $F(r) = (v_x(r), v_y(r), v_z(r))$
  - stress (tensor): $F(r) = (t_{xx}(r), t_{xy}(r), \ldots)$

➔ *property as function of coordinates*

Relational Model Doesn't Support Field Data Very Well

- doesn't support how we want to use field data
  - want to add, subtract, differentiate, visualize
  - table operations are too low level
- doesn't support how we want to store field data
  - technicalities prevent simple table interpretation
  - efficiency issues as well

➔*relational model typically not used for field data*

Relational Model Not Used For Field Data

- scientific computing community focused on file formats
- first generation were just ad hoc files with FORMAT statement
- next generation encapsulated format in APIs
- later generations attempted to extend scope of integration

Next Generation Encapsulated Format In APIs

- CDF (Common Data Format) NASA Goddard
  - computational fluid dynamics grids
- net CDF (network CDF) NCAR
  - CDF with distributed computing extensions
- Exodus (Sandia)
  - finite element meshes
- vendor specific formats in oil & gas industry
- many others

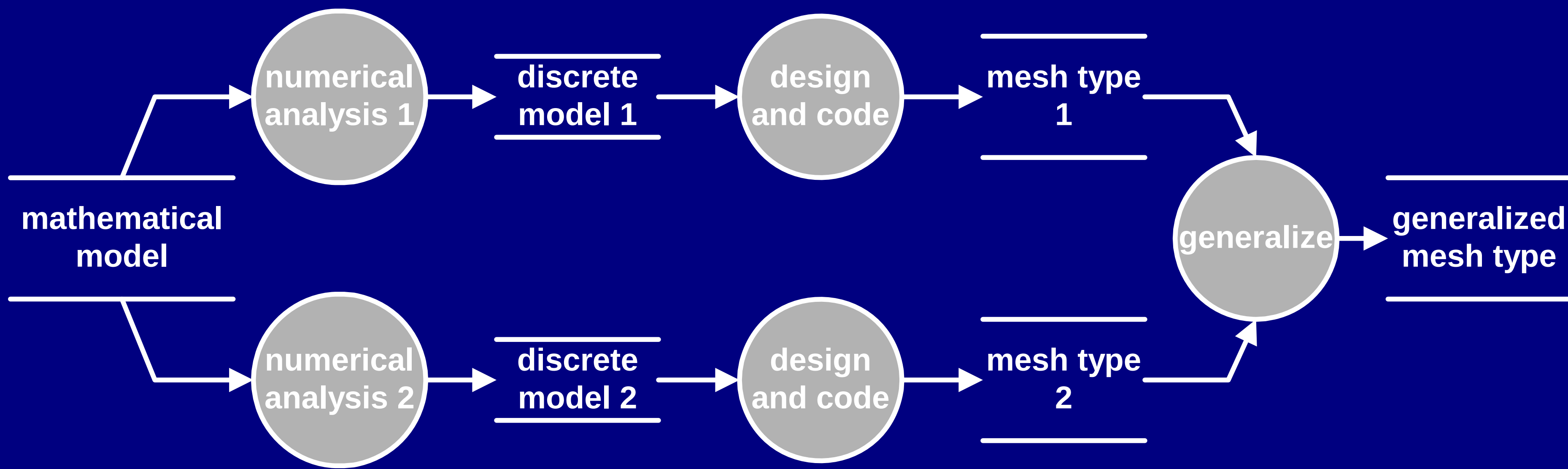
➔*created islands of integration around a given mesh type*

Later Generations Attempted To Extend Scope Of Integration

- support multiple mesh types
- HDF (U of Illinois/NCSA)
  - generalized external arrays
- Sampled Data type (Open Spirit)
  - various oil & gas mesh types
- islands of integration became "archipelagos"
- but lesson learned repeatedly ...

➔*integration based on file formats fails*

Integration Based On File Formats Fails



- integration depends on shared abstractions
  - mathematical model is what's shared
- file format generalizes mesh types
  - mesh types are what's not shared

➔*the right data model for fields is mathematical*

Sandia National Labs 1987-1989

- visualization just emerging as important computational tool
- visualization applications particularly sensitive to data model
- Sandia National Labs, CA
  - just beginning to implement visualization tools
  - asked LPS for data model proposal

➔*fiber bundle data model*

Fiber Bundle Data Model

- D. M. Butler & M. H. Pendley, Computers in Physics,1989

  "A Visualization Model Based on the Mathematics of Fiber Bundles"
- recall two problems with relational model for fields
  - doesn't describe how we want to use data
  - doesn't describe how we store data
- fiber bundle model focused on how we want to use data
- objects are sections of fiber bundles
- operations on sections

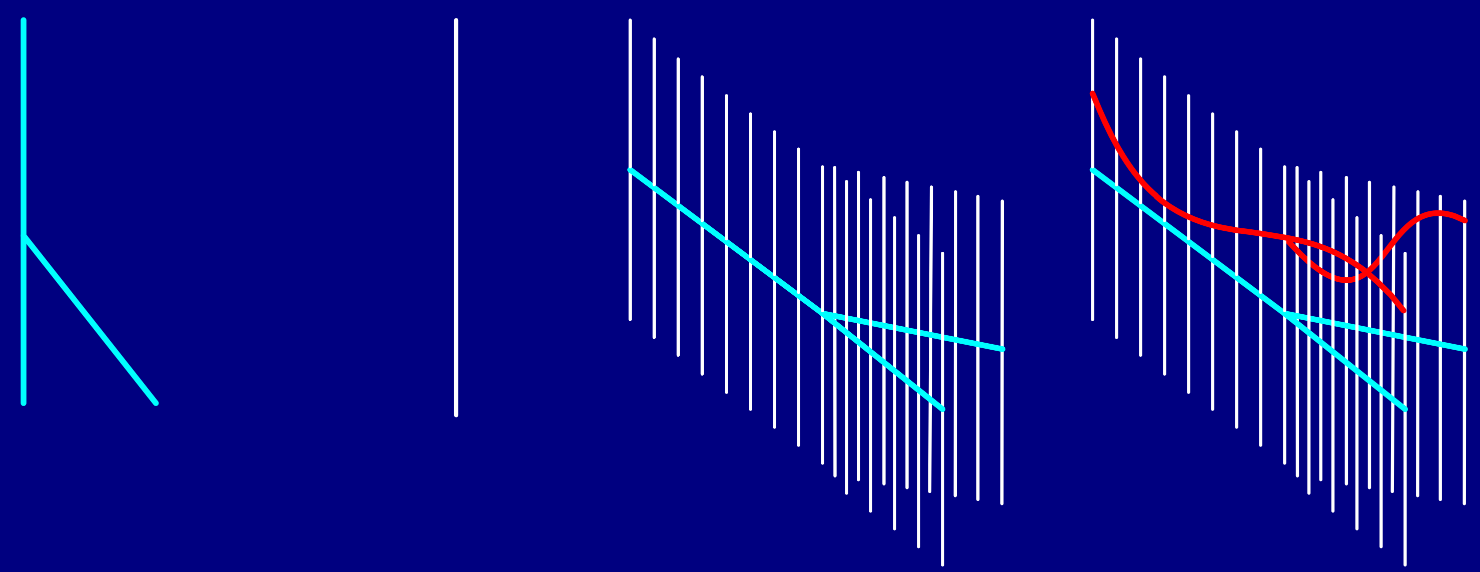
➔*very well received*

Objects Are Sections Of Fiber Bundles

- modern mathematical formalism for fields
- reformulation of notion of function
- main benefits

Reformulation Of Notion Of Function

- reformulate ***F(r)*** to emphasize geometry and topology
- focus on spaces instead of variables



base space = well **fiber = R^1** **bundle = well x R^1** **section $\subset$ well x R**

Main Benefits

- more general than function
- explicitly identifies structure
- emphasizes topology

➔*supports problems with complex topology*

Operations On Sections

- algebra
  - add, subtract, scale sections
- calculus
  - integrate and differentiate sections
- visualization
  - visualize section vs section
- number of other useful operations

➔*describes how we want to use field data*

Very Well Received

- heralded as "break through" by reviewers
- several groups built systems based on it
  - IBM Data Explorer product
- benefits
  - very broad coverage, any sort of field
  - good description of large scale structure of a field
  - operations fit how we want to use the data

➔ *adopted as starting point for ASCI data model*

Advanced Strategic Computing Initiative (ASCI) 1998-2003

- US Department of Energy
- 5 year program to advance state of the art of HPC
- led by US National Labs
- participation by other labs, academia

➔*Data Models and Formats (DMF) effort*

Data Models And Formats (DMF) Effort

- core participants
  - Lawrence Livermore National Lab
  - Los Alamos National Lab
  - Sandia
  - LPS
- input and review from other labs, academia
- charter
  - analyze data representation requirements
    - high performance computing in general
    - DOE complex in particular
  - identify data model for complex-wide data integration
  - implement software to support model

➔*began with extensive requirements gathering and analysis*

➔*limitations of fiber bundle model*

Began With Extensive Requirements Gathering And Analysis

- started with fiber bundle data model
- iterated for over a year at all three labs
  - review model capabilities with stakeholders
  - elicit additional requirements
  - analyze requirements
  - revise model as needed

➔*identified critical limitations of fiber bundle model*

Limitations Of Fiber Bundle Model

- base space issues
- fiber space issues
- technical mathematical issues

➔*combinatorial explosion of ad hoc extensions*

Base Space Issues

- discretization issues
- part hierarchy issues
- parallelization issues

Discretization Issues

- zoo of different mesh types
- adaptive mesh refinement of particular interest
- no obvious, natural support for describing decomposition of base space into cells in mesh

➔*ad hoc extensions*

Part Hierarchy Issues

- assembly structures (solid dynamics, mechanics)
- material structures (fluid dynamics, climate models)
- political and geographic structures
- multiple, concurrent part hierarchies
- no obvious, natural support for describing decomposition of base space into parts

➔*ad hoc extensions*

Parallelization Issues

- multiple concurrent domain decompositions
  - different decompositions for different platforms
- dynamic decompositions
  - load balancing
  - adaptive mesh refinement
- no obvious natural support for describing domain decomposition

➔*ad hoc extensions*

Fiber Space Issues

- users required arbitrary client-defined fiber types
- no obvious, natural support for describing decomposition of client-defined objects into primitive data members for storage

➔*ad hoc extensions*

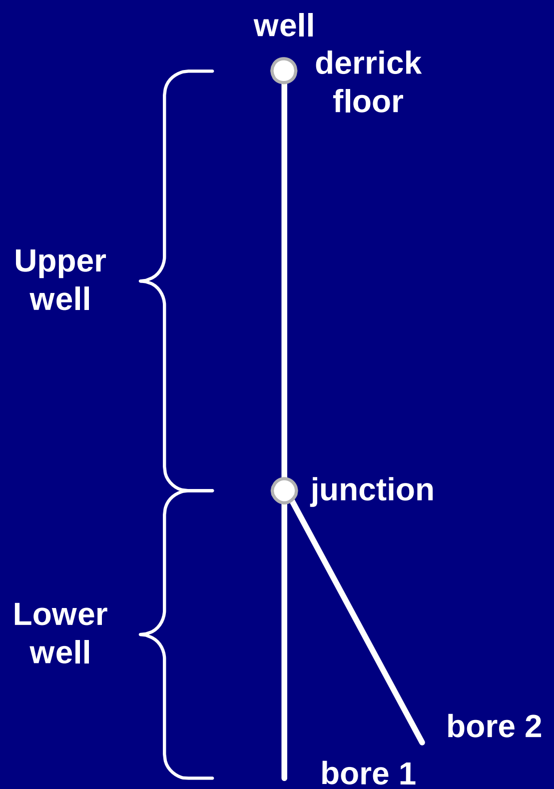
Combinatorial Explosion Of Ad Hoc Extensions

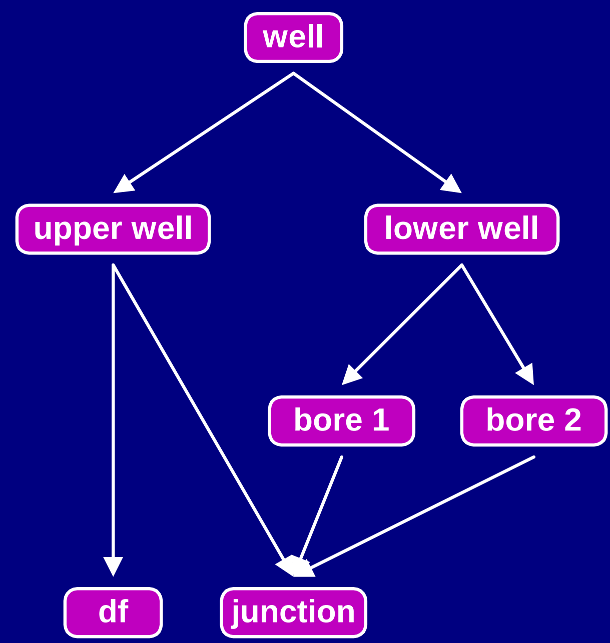
- ad hoc extensions got out of control
- fundamental problem
  - fiber bundle data model describes interface
    - how we want to use the data
  - doesn't describe the representation
    - decomposition of objects into primitive data
    - "data" model doesn't describe the data!

➔*needed mechanism for describing decomposition*

Needed Mechanism For Describing Decomposition

- LPS had already explored use of poset/lattice theory
  - informal "subset inclusion lattice" for assembly structures

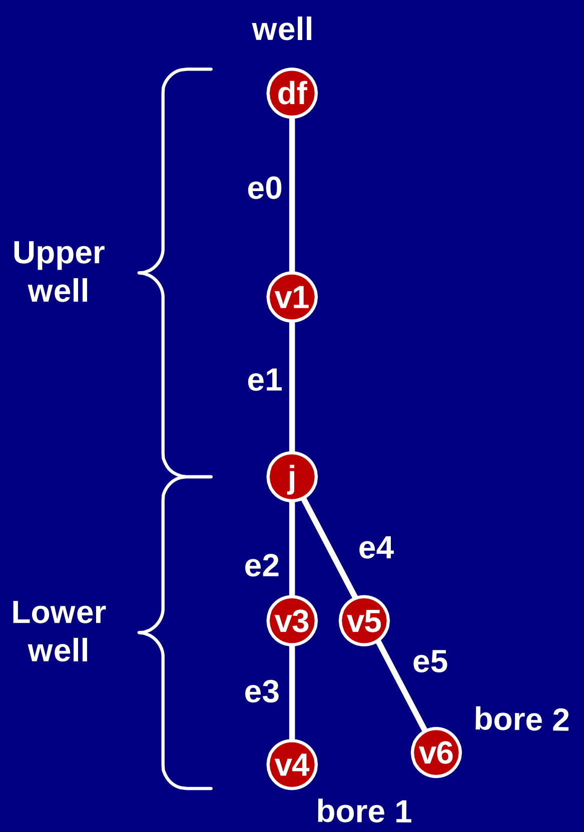


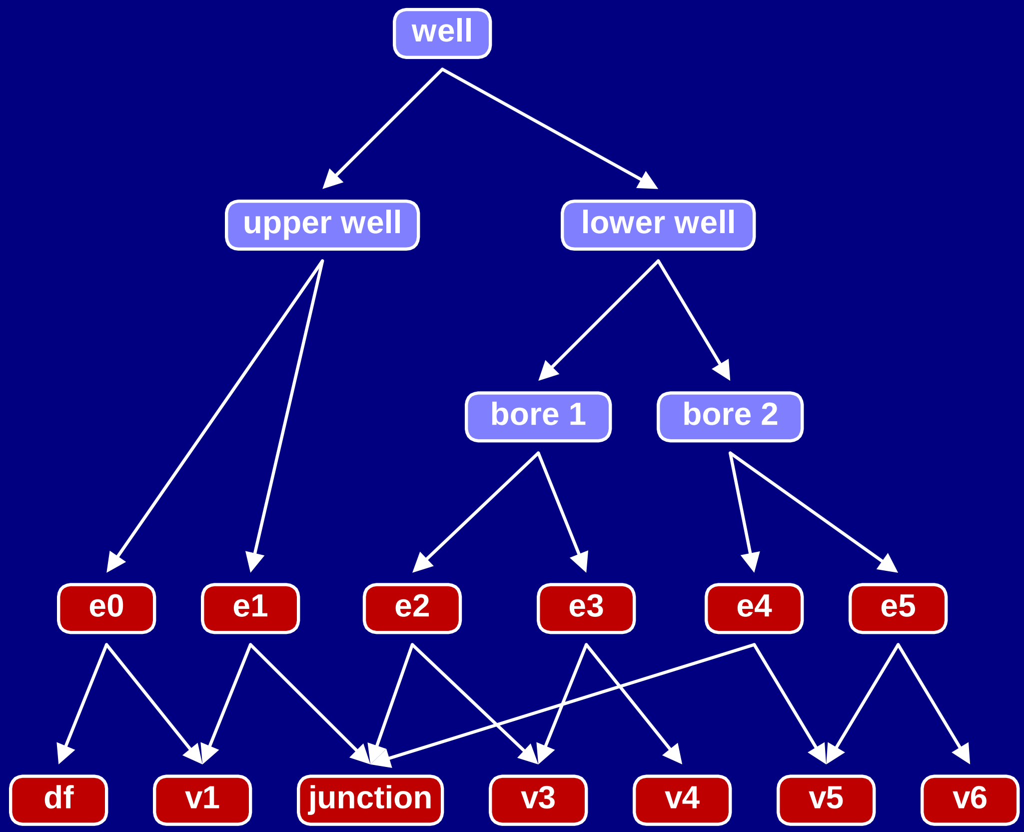


➔*decided take lattice theory seriously*

Decided Take Lattice Theory Seriously

- describe everything as a poset, including discretization





➔*support sections of arbitrary fiber types over lattices*

➔*sheaf theory*

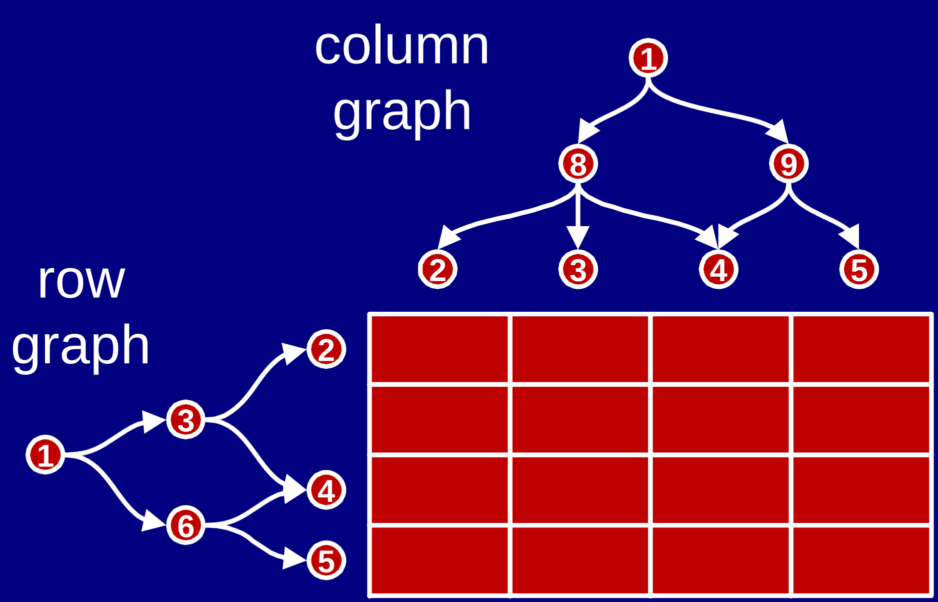
Sheaf Data Model

- D. M. Butler, U.S. Patent 6,917,943 B2, 2005
  "Sheaf Data Model"
- sheaf data model focuses on how to store the data
  - decomposition of complex objects into primitive data
- objects of the sheaf model
- operations of the sheaf model

➔*general system for managing decompositions*
➔*single integrated formalism*

Objects Of The Sheaf Model

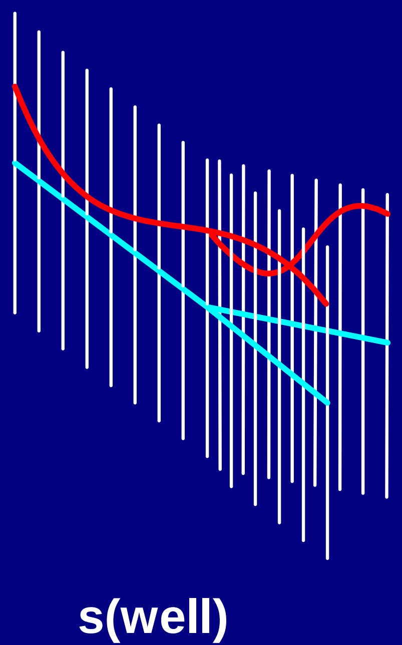
- sheaves over lattice-ordered relations
- data is logically table structured
    - like relational model
- rows and columns are lattice-ordered
    - represents part hierarchies
- table + graph metaphor

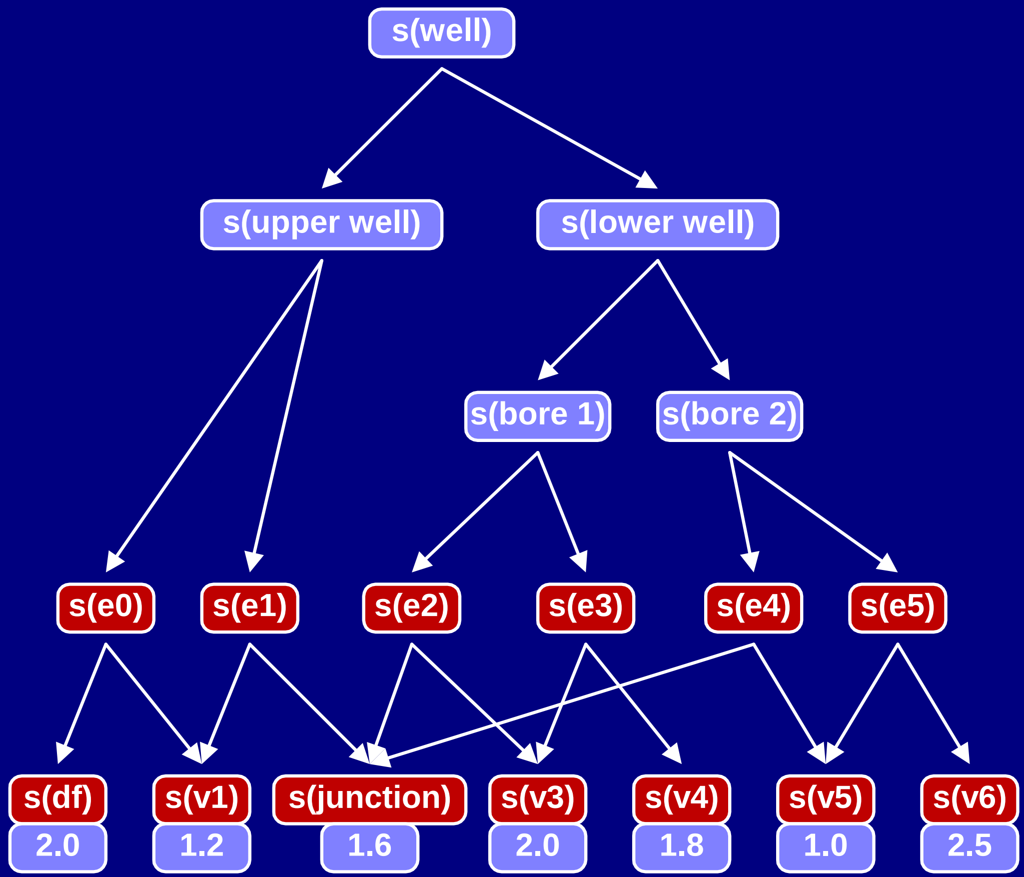


Operations Of The Sheaf Model

- "sheaf algebra" (for want of a better term)
- generalization of relational algebra
  - row and column operations
  - graph operations

General System For Managing Decompositions





- ➔*all the way down to the bytes*
- ➔*general bridge from abstract math to numerical representation*
- ➔*"intergalactic dataspeak" for field data*

Single Integrated Formalism

- traditional relational data
- object-oriented data
- topology/decomposition
- field data

➔*proven in 10 years of R&D at US national labs and major oil co.*

End