

OPERATION-CENTRIC ARCHITECTURES

For Supporting Integrated Information Access Systems

This document presents terms and concepts of a new discipline for relating a body of formal mathematics to the design and development of high-performance integrated information access systems.

XST: Extended Set Theory adapts the axioms of Zermelo-Fraenkel set theory to accommodate a compound set-membership condition. With an extended definition for *sets* new mathematical objects can be defined that are not visible to Classical set theories. These new mathematical objects include: unambiguously defined n-tuples, arrays, RDM-Relations, SQL-tables, XML-structures, physical RAM, and physical storage devices.

XOPS: Extended Set Operations are rigorously defined, executable XST operations which operate on mathematical objects in a digital computing environment. XOP definitions are mathematical entities. XOP implementations are software entities. The operations defined by Codd's Relational Data Model (RDM) are XOPS. They are well-defined in XST and have many different commercial implementations. RDM-operations and RDM-operands are the data access and manipulation core of all SQL implementations.

XSP: Extended Set Processing is the application of XOPS to XST representations of physical data resident in a digital computing environment. These set-processing systems rely on the set-theoretic definitions, or *mathematical identity*, of data, unlike traditional record-processing systems that rely solely on the physical representations of data.

Mathematical Identity: The key (and most challenging) concept of XSP is the ability to share the set-membership, or mathematical identity, of data relationships between system components without any shared knowledge of how the data is physically represented. In a Gamma Interface Architecture (where Gamma implies mathematical identity preservation) application data is represented by XST sets, which are mapped to RAM resident data representations, RAM-Sets, which in turn are mapped to storage representations of data, DISK-Sets. These set representations of data are well-defined mathematical entities and can thus be treated as operands to well-defined mathematical operations.

Operation-Centric Architectures: Traditional record-processing (structure-dependent) architectures do not recognize data representations as mathematically well-defined entities, but rather treat all data representations as navigable structures. In set-processing (operation-centric) architectures all data representations are recognized as well-defined mathematical operands. Structure-dependent architectures focus on the representation and organization of data, then choose the operations and order of application required to execute a transaction with optimal performance. Operation-centric architectures focus on the operations and data relationships required to complete a transaction, then adapt the data representation and organization to provide optimal performance. This 'backwards' approach to systems design is what makes operation-centric architectures so very difficult to accept by structure-dependent systems experts.

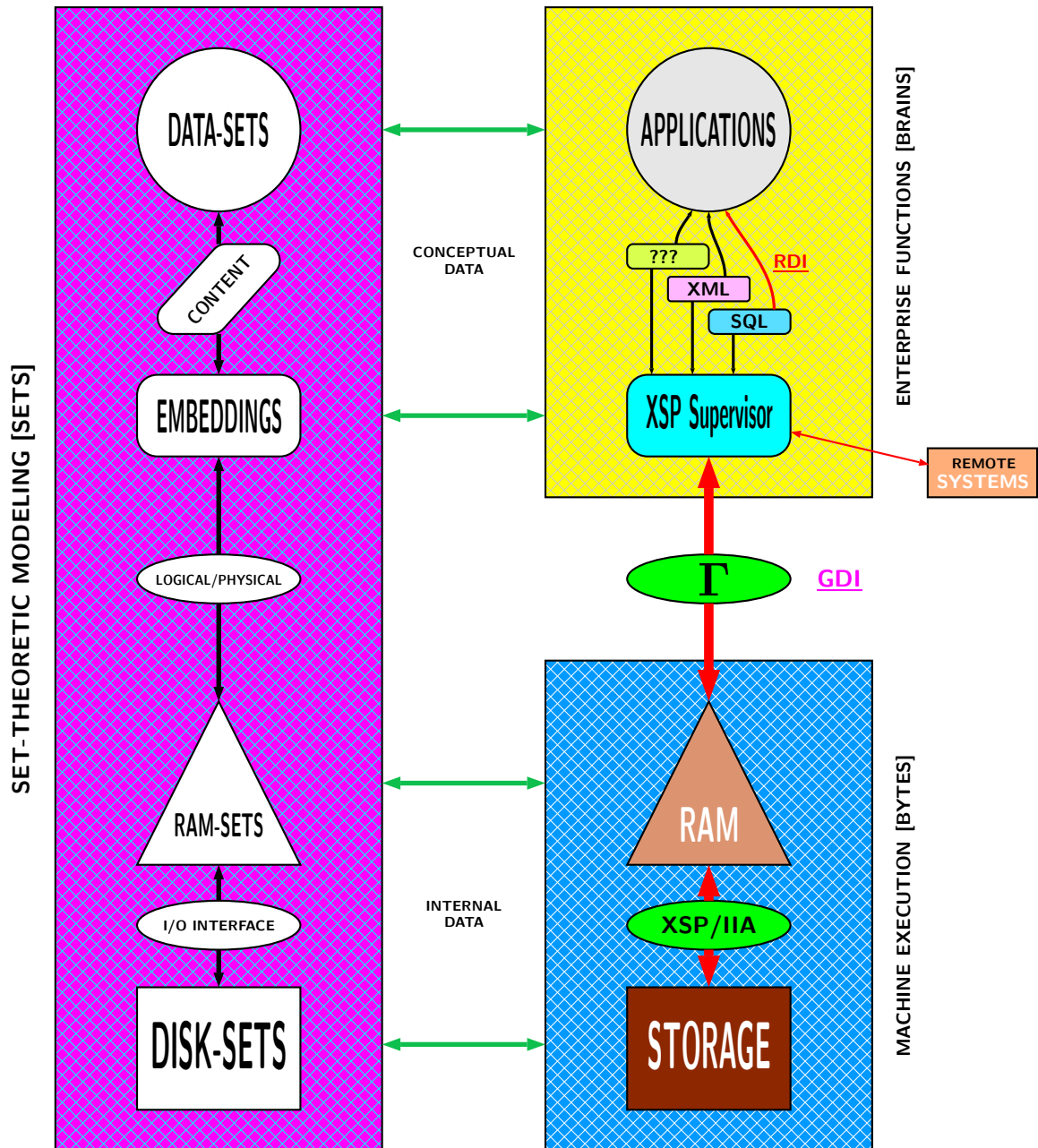
Functionality & Performance: The key distinction between operation-centric architectures and structure-dependent architectures, is the formal separation of high-level application specification from low-level system execution. This separation allows data access decisions to be made after functional requirements are known. With structure-dependent architectures data access options are predetermined by fixed index structure paths. In operation-centric architectures the data access options are adaptively determined by a sequence of XOPS. Index structures need to be constructed, validated, and maintained for each and every data item entering or leaving a system. XOPS need be constructed and validated only once. Index structures consume tremendous amounts of auxiliary storage, many terabytes to access just 100 gigabytes of raw data. XOPS require very little auxiliary storage, 200 gigabytes are adequate for processing 100 gigabytes of raw data. Index structures are stored, I/O access is slow. XOPS are executed, CPUs are fast.

iXSP: iXSP is a very specific low-level implementation of XOPS intended to support adaptive data access between secondary (local and remote) storage and RAM. Evaluations of properly chosen sequences of XOPS has demonstrated dramatic performance improvements over equivalent index structure dependent implementations.

Integrated Information Access Systems: Integrated Information Access Systems strive to share information embedded in disparate data representations. This is especially difficult in structure-dependent architectures where *form* takes precedence over *content*. In operation-centric architectures, where *content* takes precedence over *form*, the problem is much less difficult. The two-tiered, operation-centric architecture depicted on the next page is notable, not only because it is inherently more robust than its counterparts in the industry, but also because it is capable of supporting any integrated information access system.

XSP: GAMMA INTERFACE ARCHITECTURE

OPERATION-CENTRIC INTEGRATED INFORMATION ACCESS SYSTEMS



ALL ARROWS PRESERVE MATHEMATICAL IDENTITY (SET-MEMBERSHIP Γ).

RDI: LOGICAL DATA INDEPENDENCE, GDI: PHYSICAL DATA INDEPENDENCE.

XSP/IIA: XSP INTEGRATED INFORMATION ACCESS I/O INTERFACE(Γ XSP)

XSP (extended set processing) Technology is a modeling methodology for implementing systems characterized by a Gamma Interface Architecture (GIA). In a GIA all environments are data independent of each other. That is, set-membership is shared between environments while data representation is contained within environments. Thus, in a GIA, performance issues can be controlled independently of functional specifications. The only requirement is to capture set-membership without depending on data representation. Thus the need for extended set theory (XST). XST allows formal designs for computer implementations (of both logical and physical data representations) as they actually exist - instead of as artificial views for conceptual convenience.