The Microsoft data platform

Grid computing presentation
November 24th, 2009

Murata Laboratory
09M38482
Pham TuongVan

Outline

- Application development: the old way, and the data-centric perspective
- The trends of new developing environment for data-centric applications
- Microsoft data platform: key tenets and the architecture
- Summary

Outline

- Application development: the old way, and the data-centric perspective
- The trends of new developing environment for data-centric applications
- Microsoft data platform: key tenets and the architecture
- Summary

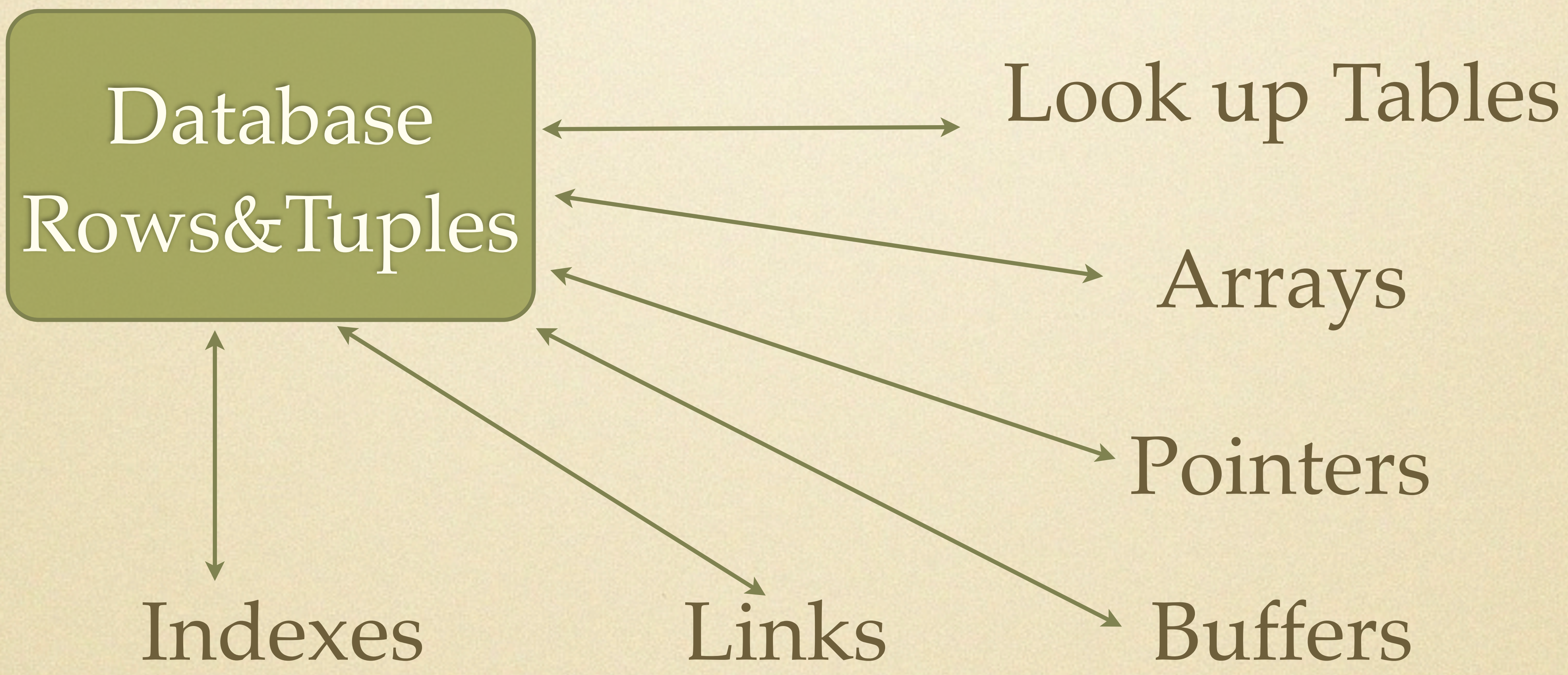
Application Development

The old way

- Defining, initializing, and managing data structures such as look-up tables, arrays, linked lists, indexes, pointers, buffers
- Acquiring, aggregating, and transforming data with such mechanisms as message queues, filters, data logging, fill and stop buffers, circular buffers, device drivers
- Using data to drive control and event processes such as control loops, events, alarms, and signal analysis

Application Development

The old way



Database Rows&Tuples

Look up Tables

Arrays

Pointers

Indexes

Links

Buffers

Application Development

The old way

- Approximately 50% of application code is for data-management
- Nearly 90% of the serious bugs can be traced back to that data-management code.

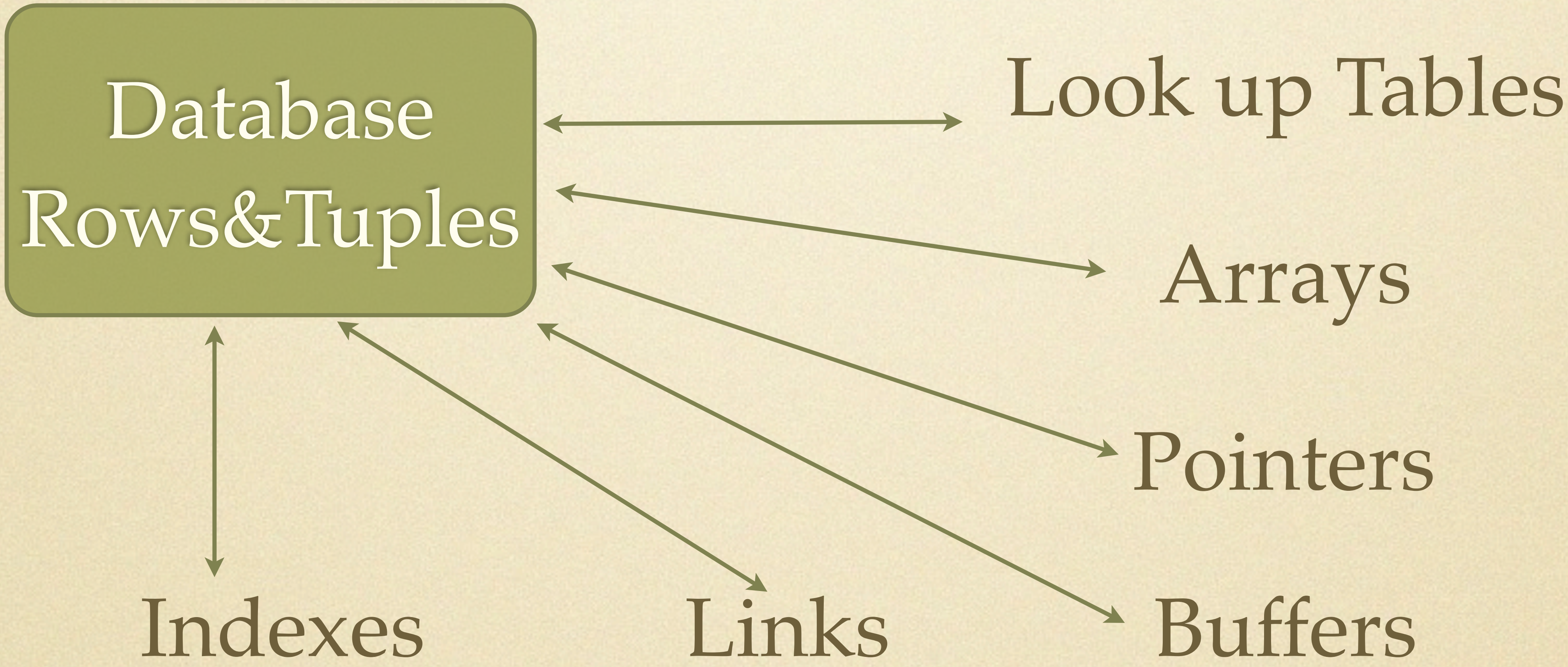
Application Development

Data-centric characteristics

- Data flows in from one or more sources
- Some data is static and stored, but other data is dynamic and accessed in different ways
- Data can be ordered and structured or unstructured
- Data is replicated from one system to another
- Data must be integrated and shared across multiple tables or applications (or both)
- Data may have to be stored in different storage media internal and external to the device or the cloud
- The amount of data to be managed is growing in both size and complexity
- Data can be large while the devices operating on that data have limited system resources (usually for cost reasons)

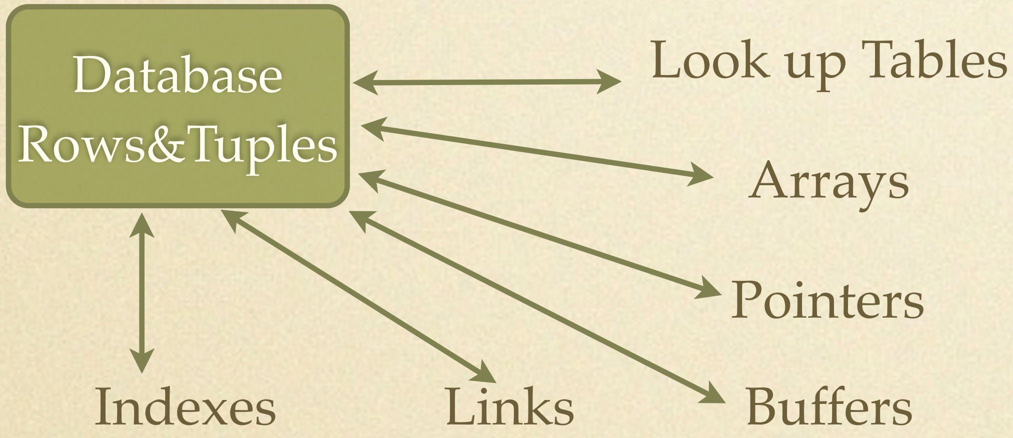
Application Development

The old way



Application Development

The old way



Database Rows&Tuples

Look up Tables

Arrays

Pointers

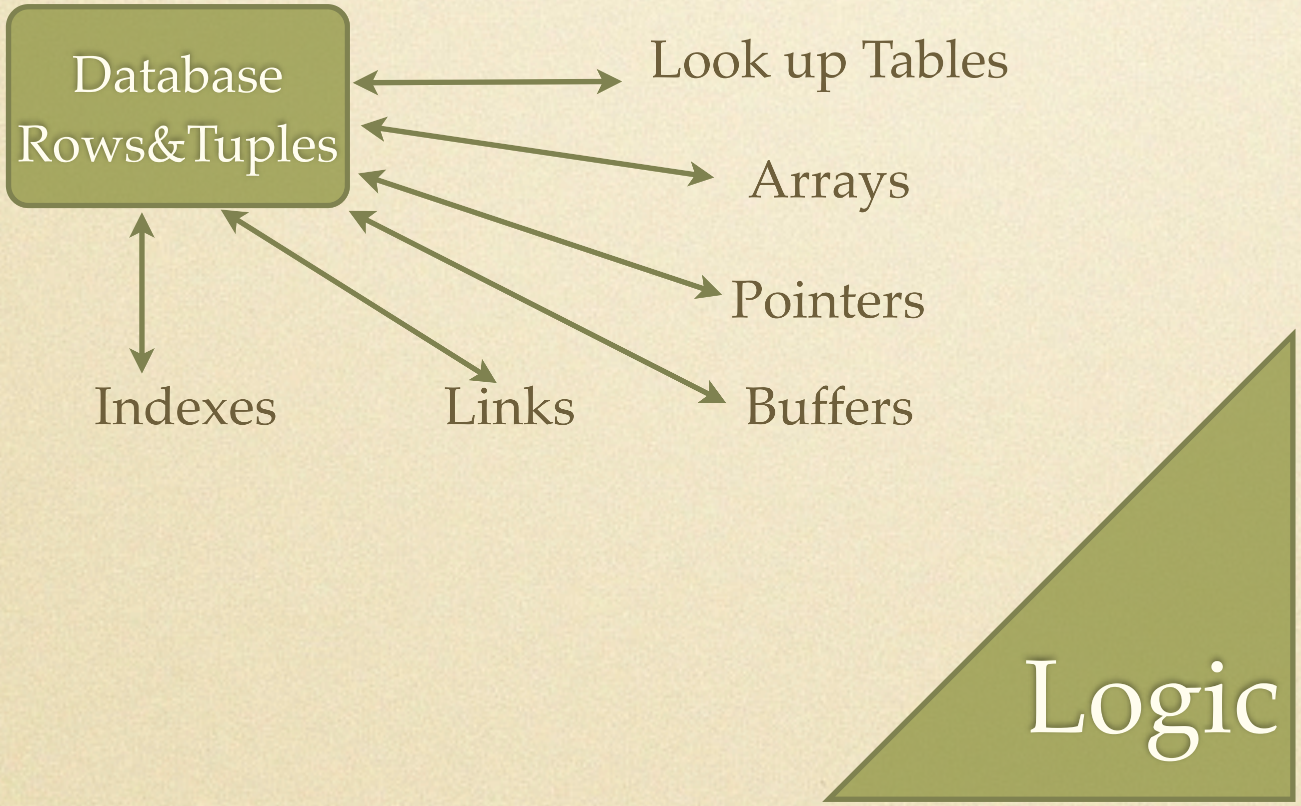
Buffers

Indexes

Links

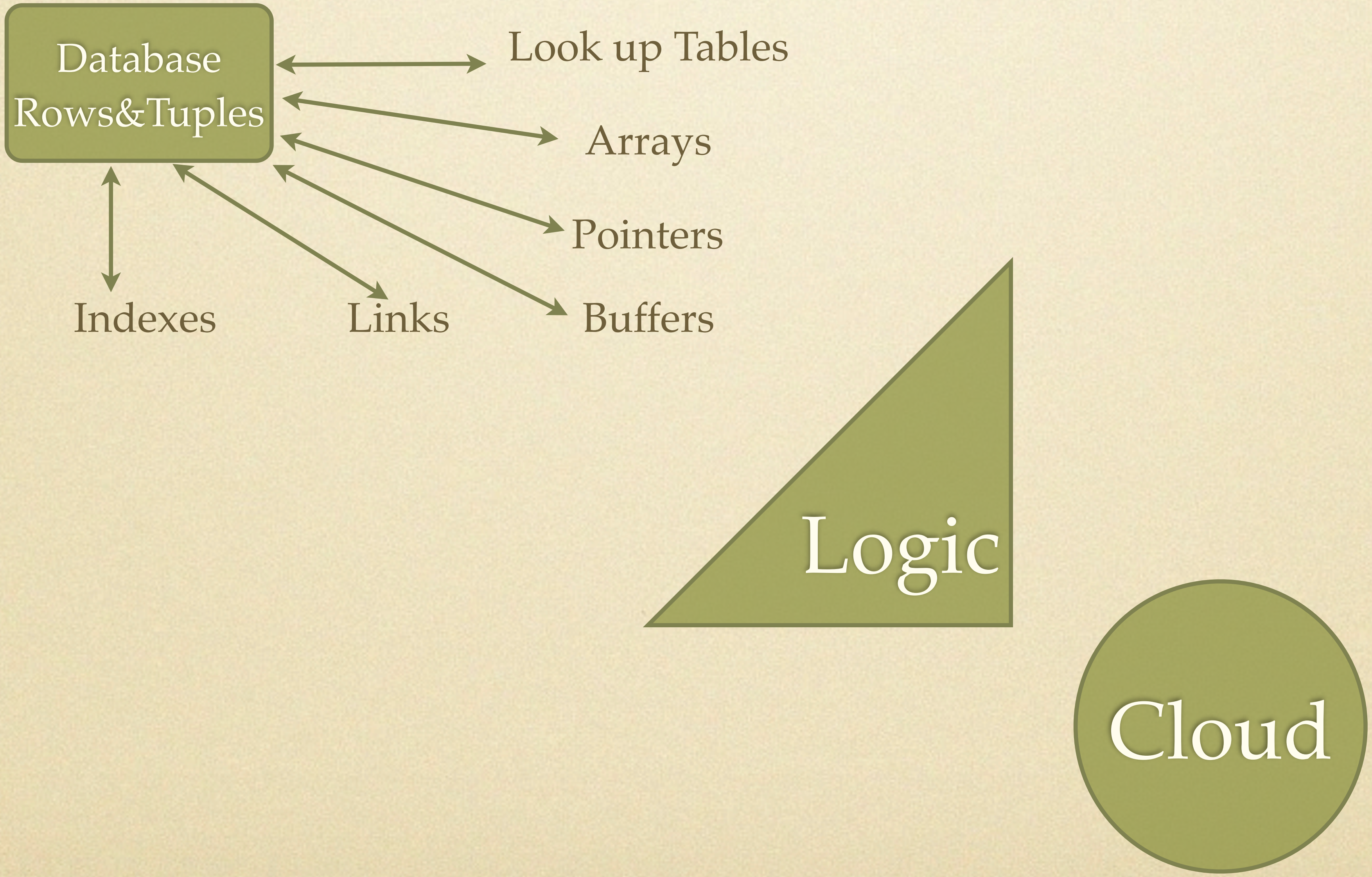
Application Development

The old way



Application Development

The old way



Database Rows&Tuples

Look up Tables

Arrays

Pointers

Buffers

Indexes

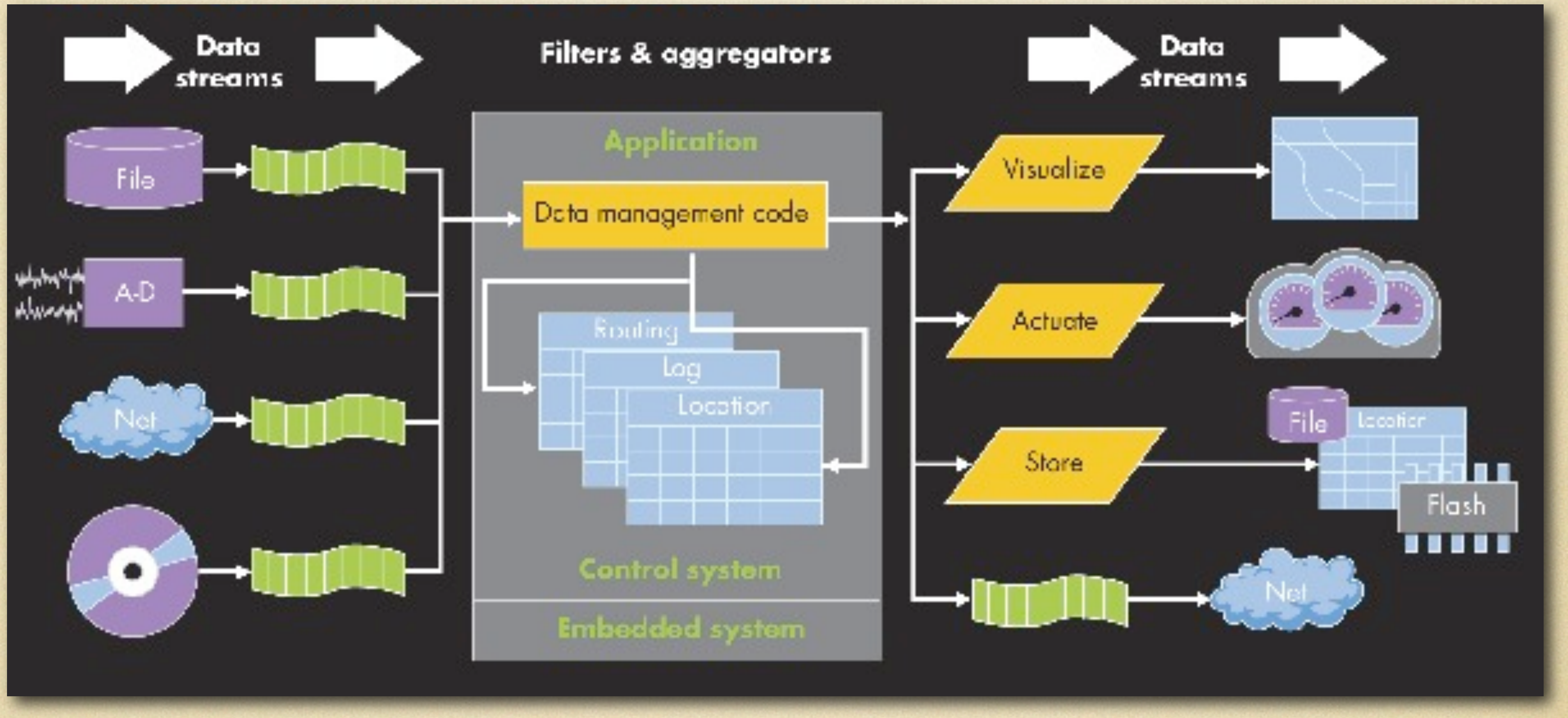
Links

Logic

Cloud

Application Development

Data-centric application model



Eugene K. Buechele - Embedded Development Community 11/09/05

Outline

- ~~Application development: the old way, and the data-centric perspective~~
- The trends of new developing environment for data-centric applications
- Microsoft data platform: key tenets and the architecture
- Summary

Data-centric Environment

Hardware Trends

- 64 bit CPUs, and 64 bit OSs
- Multi-core CPUs
- Main memory sizes are growing
- Non-violate RAMs, Flash

Data-centric Environment

Storage Trends

- Low cost commodity storage
- Data operational cost exceeds hardware cost
- Merging of structured data and unstructured data

Data-centric Environment

Device and Client Trends

- More capable & powerful Devices
- Mobility dramatically increases
- Streaming technology
- Occasionally Connected to the Cloud

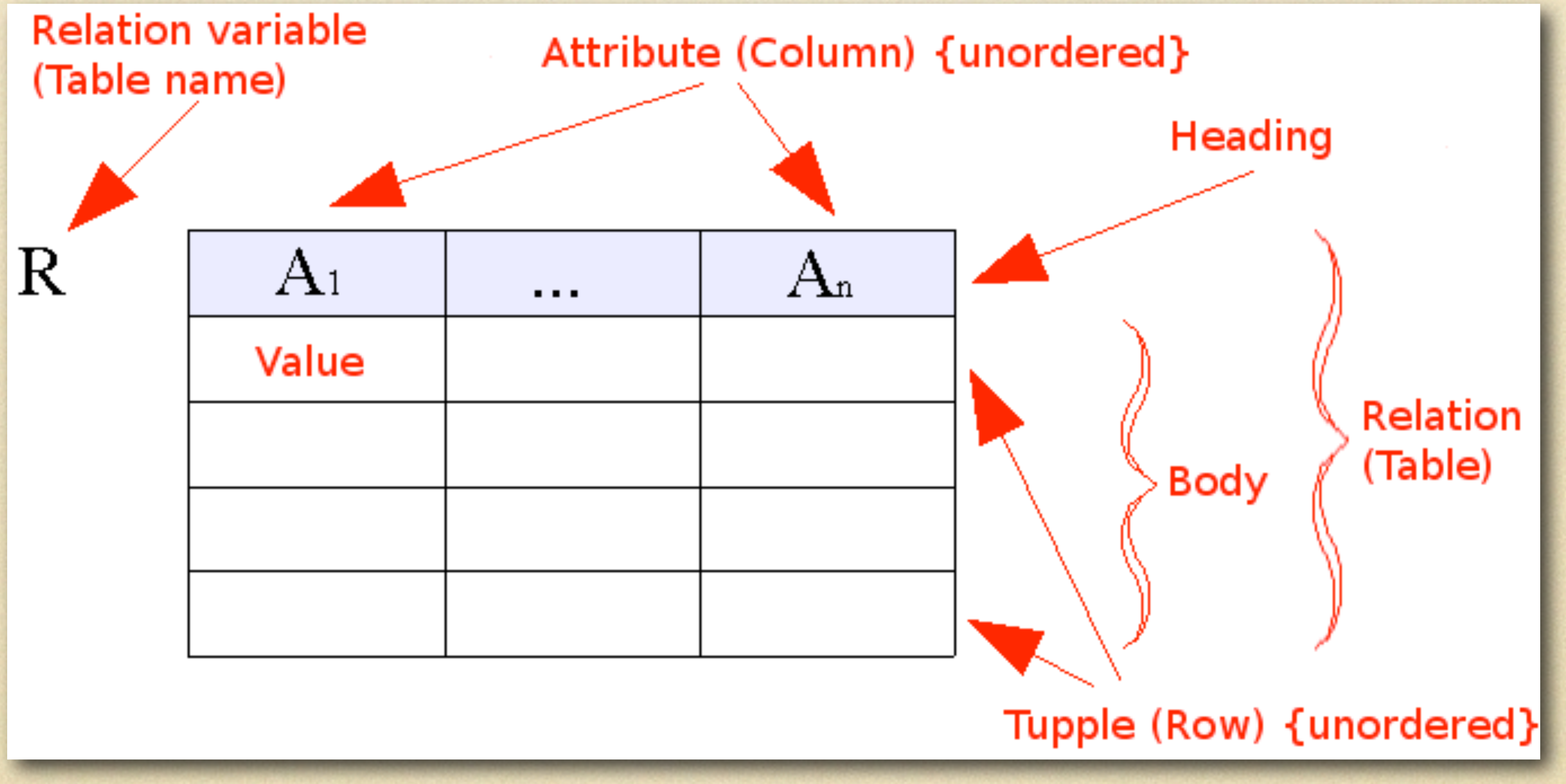
Data-centric Environment

Data and Applications Trends

- Data everywhere(devices, cloud...), every tier
- Data Proximity: Form & Process
- Embedding data application in other application
- Shifting from relational data model to entity-relationship data model

Data-centric Environment

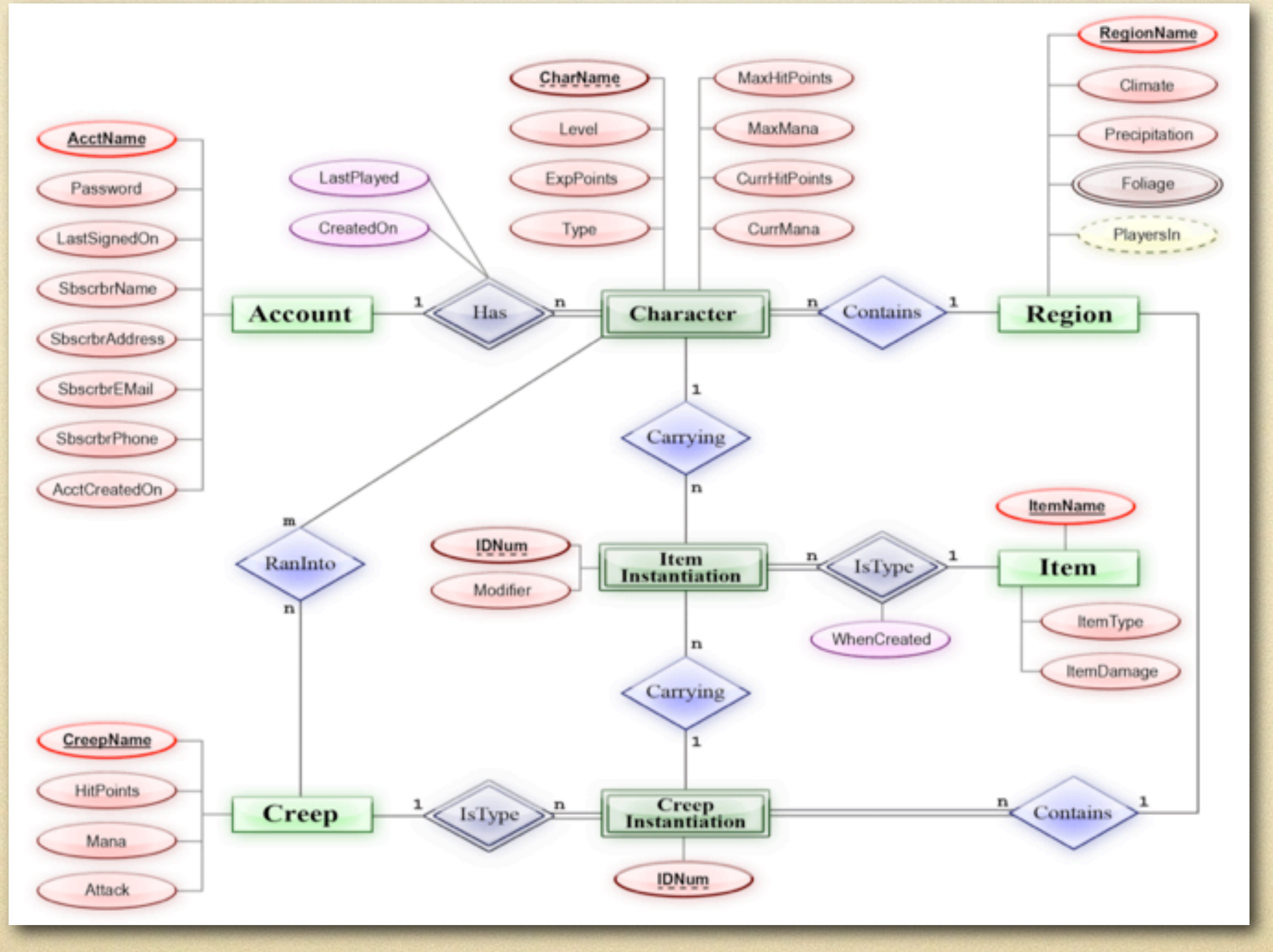
Relational data model(table-tuple)



A_1	...	A_n
Value		

Data-centric Environment

Entities datamodel



Outline

- ~~Application development: the old way, and the data-centric perspective~~
- ~~The trends of new developing environment for data-centric applications~~
- Microsoft data platform: key tenets and the architecture
- Summary

Microsoft Data Platform

Key Tenets: Platform for all data

- Traditional structured data; Byte stream data (audio, video); Semi-structured data; Web clickstream, feeds, audit logs...
- Better integration of database system (for query-ability) and file system (for performance and functionality)

Microsoft Data Platform

Key Tenets: Low cost, reliable storage

- Deliver significant improvement in database maintenance and management costs.
- Enable friction free addition of necessary resource such as process and disk, with zero-touch provisioning and resource balancing

Microsoft Data Platform

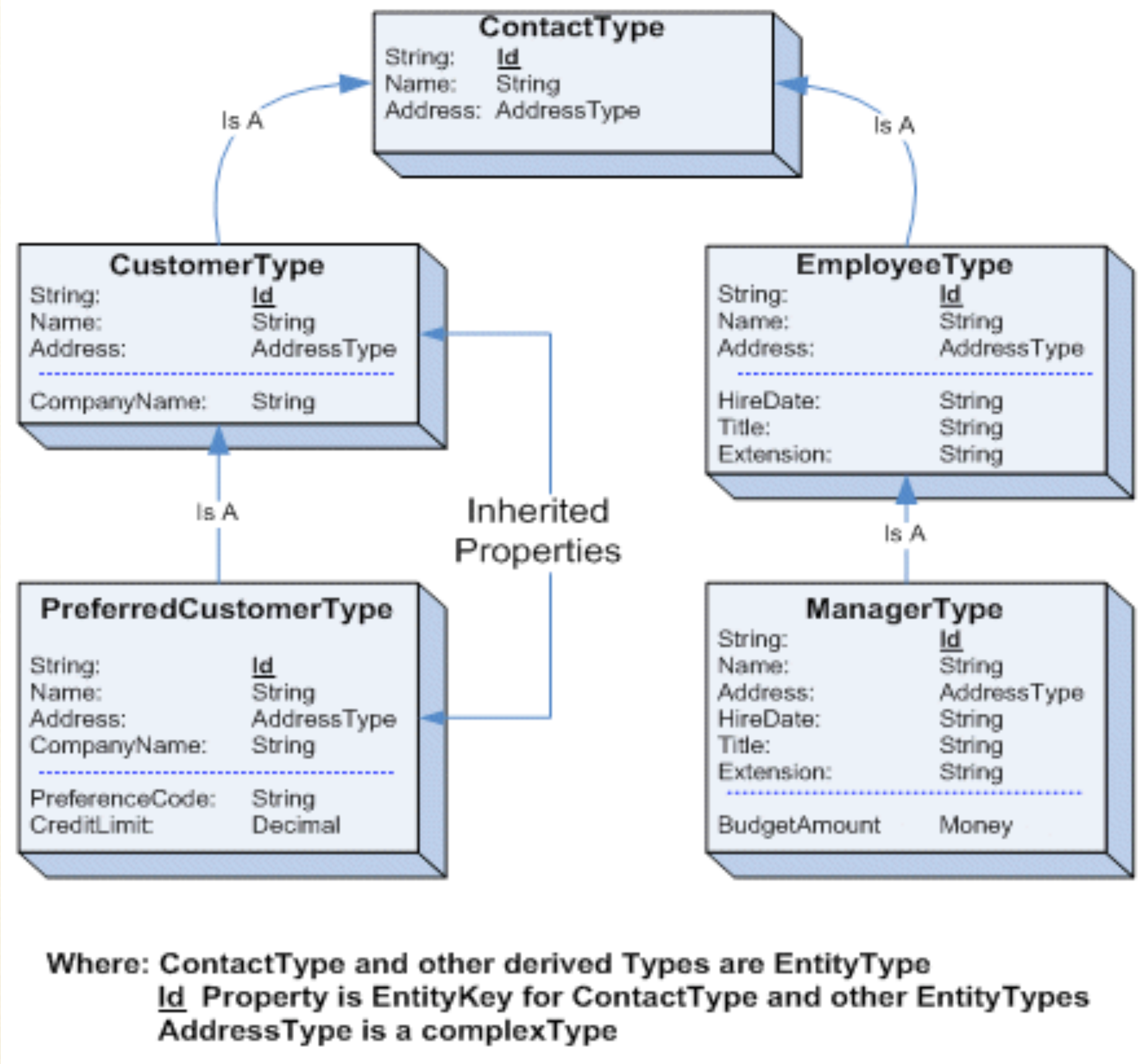
Key Tenets: Reach and Rich

- Allowing users to store data locally and to synchronize, to access their personal devices such as Phones, PCs, USBs, Cameras...
- Provide better offline experience for web applications.

Microsoft Data Platform

Key Tenets: Entity data model(EDM)

- Type: Each item (entity) are described by a type, can be inherited from another type



ContactType

String:	Id
Name:	String
Address:	AddressType

CustomerType

String:	Id
Name:	String
Address:	AddressType
CompanyName:	String

EmployeeType

String:	Id
Name:	String
Address:	AddressType
HireDate:	String
Title:	String
Extension:	String

PreferredCustomerType

String:	Id
Name:	String
Address:	AddressType
CompanyName:	String
PreferenceCode:	String
CreditLimit:	Decimal

ManagerType

String:	Id
Name:	String
Address:	AddressType
HireDate:	String
Title:	String
Extension:	String
BudgetAmount	Money

Where: ContactType and other derived Types are EntityType
Id Property is EntityKey for ContactType and other EntityTypes
AddressType is a complexType

Microsoft Data Platform

Key Tenets: Entity data model(EDM)

- Instances:
  - EntitySet: a set of instances of an entity type.
  - RelationshipSet: hold two or more entities which has same relation for a given relationship type.

Microsoft Data Platform

Key Tenets: Data Integration

- Dropping of data storage cost, data operation cost allow us focus on data mining
- MDP allows users and analysts to access, enrich and report on the data they need to do their job
- Allow data manager to control source quality, thus improve productivity.

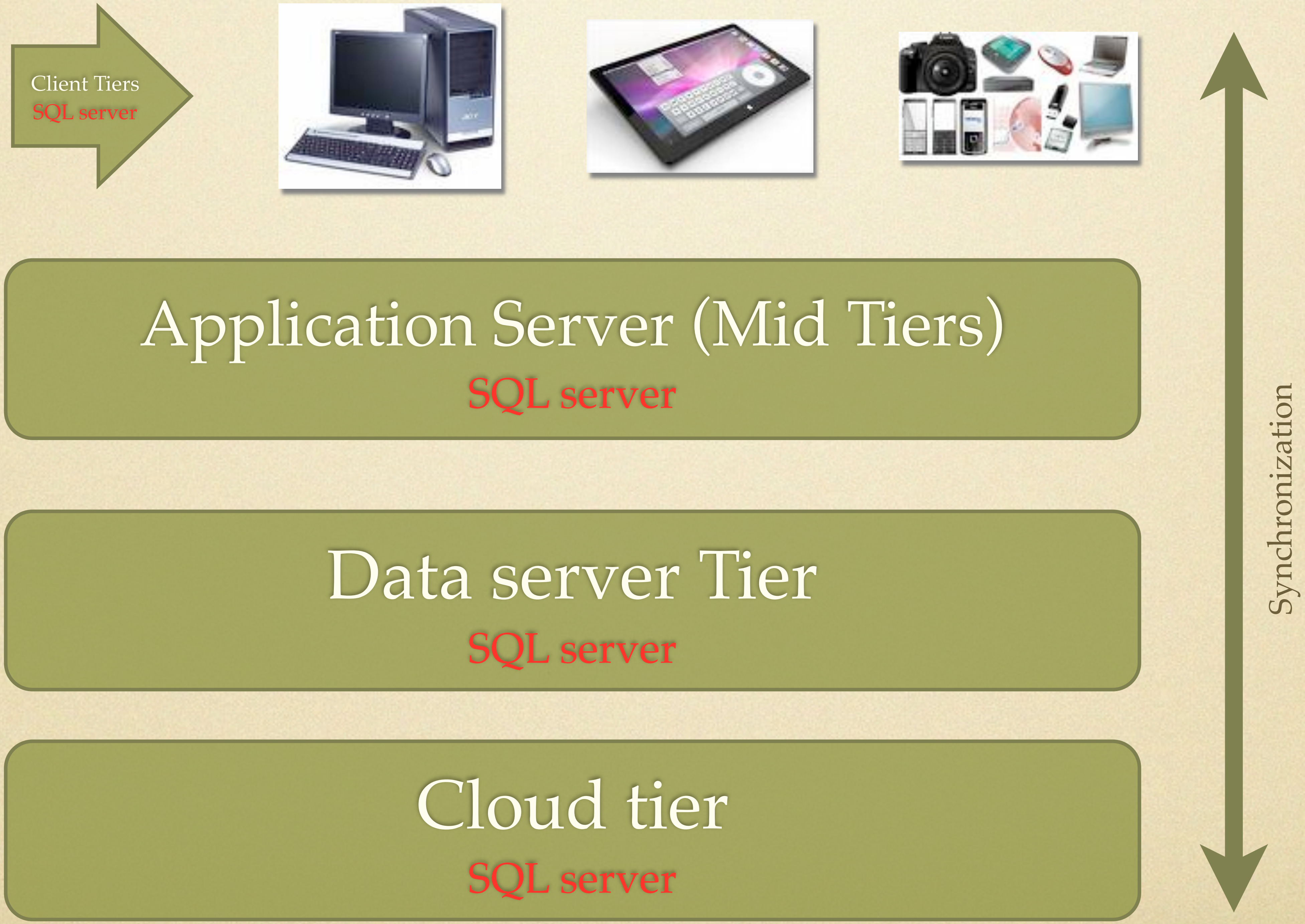
Microsoft Data Platform

Key Tenets: Cloud scale data service

- Provide a reference data service, that allows web clients to navigate entities via HTML.
- Provide means to upgrade shared data
- Provide space for third parties host application logic which integrates with existing storage and services.

Microsoft Data Platform

Architecture: The tiers



Client Tiers
SQL server

Application Server (Mid Tiers)
SQL server

Data server Tier
SQL server

Cloud tier
SQL server

Synchronization

Microsoft Data Platform

Architecture: Characteristics of data in tiers

- Cloud tier: Large data size, highly partitioned data, data operations such as maintenance, monitoring is essential
- Data server tier: data tends to be mostly structured, highly transactional
- Application server tier: Typically cached data, tends to be application-centric, must be self-managed
- Client tier: Data sizes are not very large, self-management is critical

Microsoft Data Platform

Architecture: Uniform application Development(1)

- Uniform database model:
  - The entity model, which supports row, XML and BLOB data (Binary Large OBject), is desired across all tiers.
- Uniform programming model:
  - Uniform query and data definition language such as Entity SQL, XQuery
  - Uniform APIs

Microsoft Data Platform

Architecture: Uniform application Development(2)

- Uniform development environment
  - Use same tools to develop applications such as Visual Studio
  - Possibility of deploy applications in different tiers
- Uniform management across tiers
  - Although data management experience is different on each tiers, however declarative management is crucial for achieve uniform management

Microsoft Data Platform

Architecture: Synchronization across tiers

- Synchronization from and to all tiers such as between servers, server and client, server and application
- Synchronization over different communication protocols and media such as wireless network, or connected network using HTTP
- Synchronization of data at different abstraction, such as file, tuples, entities

Summary

- Application development changes to data-centric due to huge impact of cloud to trends of hardware, software, storage, and applications
- Microsoft Data Platform is designed for supporting data processing of developer in all tiers

References

- The Microsoft data platform, David Campbell and Anil Noir, Microsoft Corp. WA 98052-6399
- ADO.NET Network. http://msdn.microsoft.com/data/ref/adonetnext/
- Embedded Development Community. http://www.embedded.com/