

using CA IDMS™ in the cloud

Tom Hebert
ObjEX, Inc

maymainframemadness 2012



abstract

— Cloud Computing may be the best thing to happen to CA IDMS in many years. Your organization may be looking at Cloud options, but not fully understand where CA IDMS fits. What if you use software products that embrace SOA (Service Oriented Architecture)? What if you want to create new applications that use Cloud techniques and technology? How will you get there? With his expertise in SOA design and development and deep CA IDMS experience, Tom Hebert answers these questions and describes how you can wrap existing CA IDMS applications with an architecture that can take advantage of Cloud and other new technologies.

maymainframemadness 2012

Copyright © 2012 CA. All rights reserved.



scalable cloud computing with CA IDMS? introduction

- Cloud computing may be the best thing to happen to CA IDMS in many years
- How do you use your CA IDMS data to meet new integration demands?
 - Salesforce.com
 - Industry collaboration sites
 - Social networking sites
 - Marketplaces like Amazon.com and others
- How do you implement, securely, with performance - without having to rewrite your applications?

maymainframemadness 2012

Copyright © 2012 CA. All rights reserved.



does scalable cloud computing with CA IDMS require a port? don't focus on your navel

- Relational is losing its monopoly on data
 - Consumers of cloud services do not care which database is being used
 - Witness the NoSQL movement
 - There is no advantage to using a relational database on the cloud
 - It is counterproductive to port everything before you can start work on things that are critical to your organization's success
- The way to move forward is to:
 - Keep CA IDMS
 - Allow current applications to live out their useful lives
 - Wrap CA IDMS with an architecture that allows you to use your CA IDMS data in your cloud

maymainframemadness 2012

Copyright © 2012 CA. All rights reserved.



scalable cloud computing with CA IDMS? introduction

- You aren't going to do it with screen scrapers
 - Imagine what it would be like building a real-time bridge with an outside sales provider, limiting yourself to your existing dialogs
- You aren't going to do it with your data warehouse
- It takes forethought, but it can be done
 - Software architecture
 - Organize for unknown requirements
 - Deployment architecture and plan
 - Scalable, reliable, and low-cost

cloud computing vs. "the cloud"

- "The cloud" is the collection of cloud computing services available on the internet
- Cloud computing is a design and implementation architecture
 - Allows us to safely invest in our applications
 - Typically built using a Service Oriented Architecture
 - Currently SOAP and TCP/IP prevails
- Your private cloud is the collection of services provided and maintained by your organization
 - Some are private
 - Some are exposed outside the organization

cloud computing defined

- You may not know it but you have been using the cloud
 - Bing, Google Maps, Salesforce.com, and many others
- Not just web sites
 - e.g., when Hertz or Marriott want to show you a map they don't send you to Google
 - True for airlines, travel sites, industry sites, and many others
- EDI will eventually go the way of the dodo bird

maymainframemadness 2012

Copyright © 2012 CA. All rights reserved.



cloud computing defined

- Much more than merely putting up a web site
- Driven by major internet players
 - A fabric of interconnected callable services
 - Driven by users and consumers, not by the accounting department
- Interoperable and heterogeneous
 - Any platform, operating system, or database can play as long as they follow the rules
- Documents rule (levels the playing field)
 - Relational client/server model not effective
 - XML allows complex requests and responses

maymainframemadness 2012

Copyright © 2012 CA. All rights reserved.



cloud computing documents rule

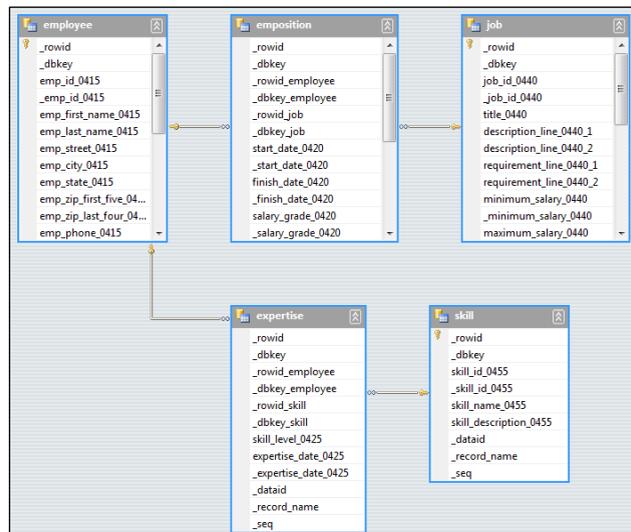
- Relational client/server model no longer effective
- Documents rule
- XML allows complex requests and responses
 - SOAP is the protocol of choice
 - SOA is the prevalent design path
- Levels the playing field for CA IDMS

maymainframemadness 2012

Copyright © 2012 CA. All rights reserved.



but first, a word about... XML documents



maymainframemadness 2012

Copyright © 2012 CA. All rights reserved.



but first, a word about...

XML documents

```
<XMLQuery_EmployeeInfo>
  <employee_record_name="EMPLOYEE" _seq="1">
    <_rowid>19201793</_rowid>
    <_dbkey>75007:1</_dbkey>
    <emp_id_0415>23</emp_id_0415>
    <_emp_id_0415 xml:space="preserve"></_emp_id_0415>
    <emp_first_name_0415>Katherine</emp_first_name_0415>
    <emp_last_name_0415>O'Hearn</emp_last_name_0415>
    ...
    <_dataid>3C294217</_dataid>
  </employee>
```

maymainframemadness 2012

Copyright © 2012 CA. All rights reserved.



but first, a word about...

XML documents

```
<emposition_record_name="EMPOSITION" _seq="73">
  <_rowid>19203330</_rowid>
  <_dbkey>75013:2</_dbkey>
  <_rowid_employee>19201793</_rowid_employee>
  <_dbkey_employee>75007:1</_dbkey_employee>
  <_rowid_job>19242759</_rowid_job>
  <_dbkey_job>75167:7</_dbkey_job>
  <start_date_0420>1980-12-21T00:00:00-07:00</start_date_0420>
  <_start_date_0420 />
  <salary_grade_0420>21</salary_grade_0420>
  . .
</emposition>
```

maymainframemadness 2012

Copyright © 2012 CA. All rights reserved.



cloud computing requirements

- Multi-platform and heterogeneous
 - XML is the enabler
- Web service consumers don't want to know
 - You must not expose platform-specific nuances
- Document oriented
 - Get it, use it, change it, and store it
 - XML documents are a universal language
- Autonomous transactions
 - You can't serve up relational cursors to your clients in cloud computing
 - Every call either did something or it did nothing

maymainframemadness 2012

Copyright © 2012 CA. All rights reserved.



cloud computing requirements

- Scalability
 - Requires linear scalability by adding machines
 - Google seems to do this better than everyone
- Reliability
 - Dropping a server for any reason should not disrupt a user's session
- World-wide performance
 - If the users are world-wide, the server base should be too

maymainframemadness 2012

Copyright © 2012 CA. All rights reserved.



cloud computing requirements

“You wouldn’t build an IDMS application without forethought. Cloud computing is no different.

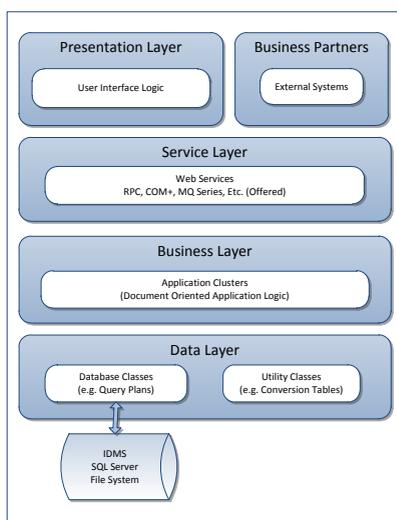
A well thought out architecture saves time.”

maymainframemadness 2012

Copyright © 2012 CA. All rights reserved.



cloud computing software architecture



A layered approach

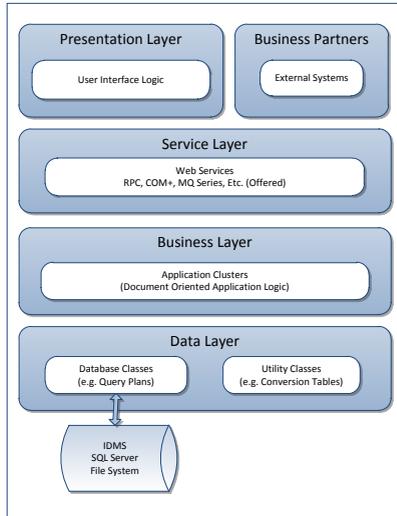
- Calls flow from the top down
- Rarely skips a level
- Calls NEVER flow up
- Layers are used by referencing DLLs
- Service layers are SOAP web service

maymainframemadness 2012

Copyright © 2012 CA. All rights reserved.



cloud computing software architecture



Data Layer

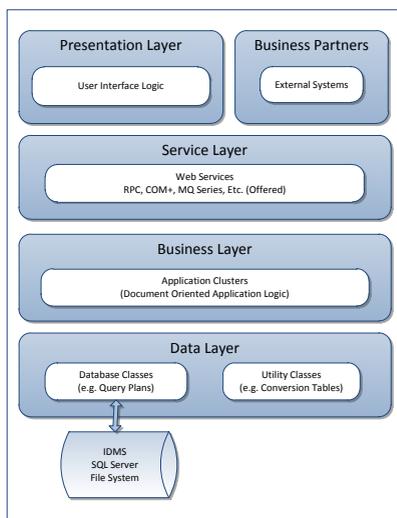
- Contains all code that directly accesses data
- Can be CA IDMS, relational, files, or calls to outside services like Google Maps
- No business rules here. Merely provides a consistent way to access data by the business layer
- Allows relocation of data without changing higher levels
- CA IDMS data can be supplied by SQL Option or ObjEx XMLQuery

maymainframemadness 2012

Copyright © 2012 CA. All rights reserved.



cloud computing software architecture



Business Layer

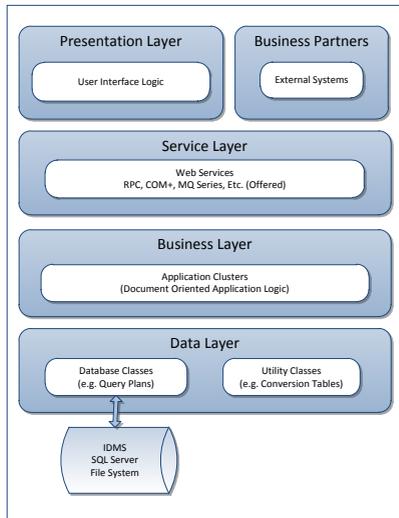
- Supplies ready-to-use documents
 - Get it, use it, change it, and save it
- Applies all edits and performs transformations
- Some object orientation
 - Could include an ER-like collection of classes
- Document examples
 - Purchase Orders, Sales Orders, others relevant to your organization
 - Reports

maymainframemadness 2012

Copyright © 2012 CA. All rights reserved.



cloud computing software architecture



Service Layer

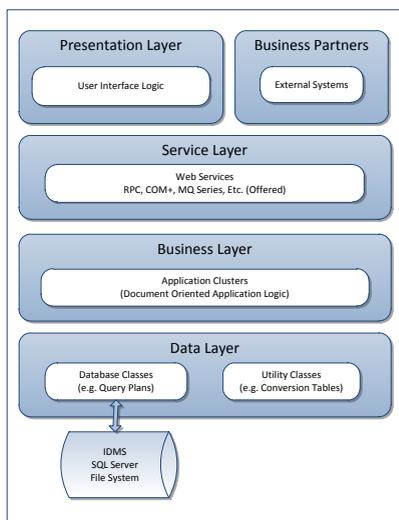
- Semantically, it simply provides RPC methods that access the Business Layer
- Both private use and for business partners
- Implementation is the key to:
 - Scalability
 - Reliability
 - Performance
- Uses local, system, and global caches
- We'll talk more about caching later on in physical deployment

maymainframemadness 2012

Copyright © 2012 CA. All rights reserved.



cloud computing software architecture



Presentation Layer

- The big unknown
 - This layer changes frequently...iPad anyone?
 - Can be windowed, cell phones, PDAs, tablets, mainframe with green screens
- Can be your applications or from your vendors, customers, or even consumers
- Once again, XML rules
- Outside callers always call the Service Layer
- In-house apps may optionally use the Business Layer DLLs directly

maymainframemadness 2012

Copyright © 2012 CA. All rights reserved.



cloud computing

nothing's ever perfect – there are challenges

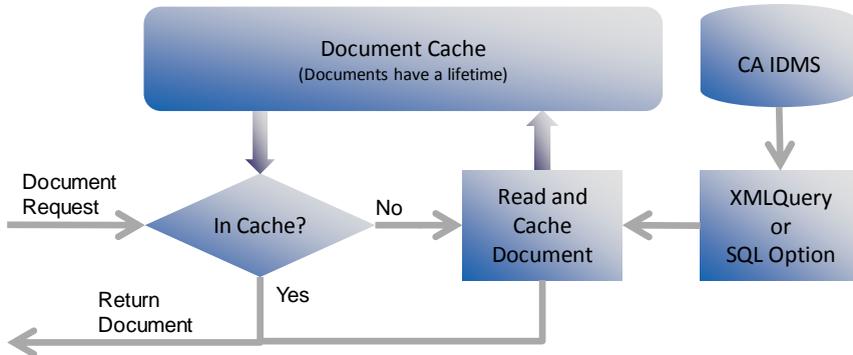
- Rich user interfaces demand data
 - This is why screen scrapers fail in volume situations
- Semi-static data
 - Edit and code tables
 - Why make thousands of calls to CA IDMS?
- Re-fetch
 - Web users tend to surf backward and forward
- Let's talk about data caching
 - Local, machine, cluster

cloud computing

why use a data cache?

- Because the world does not want this: 
 - The world wants split-second response times
- And you do not want more hardware
 - Caching reduces the demand on your data store

cache usage scenarios demand driven (pull)

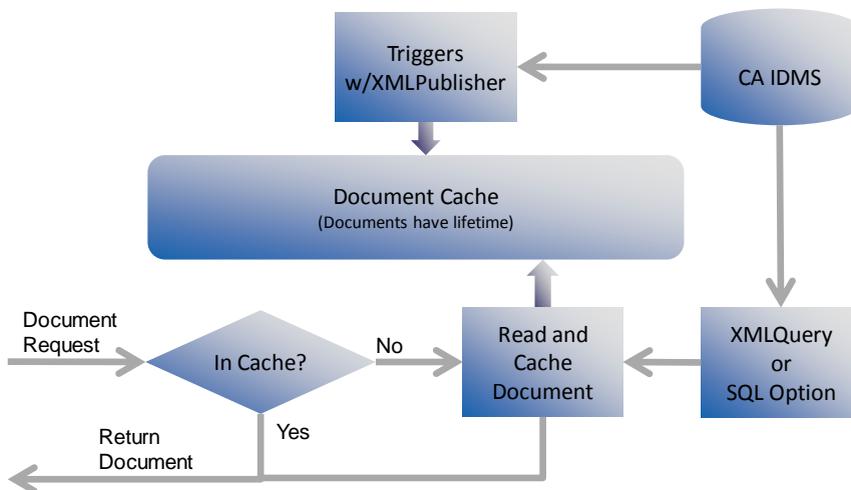


maymainframemadness 2012

Copyright © 2012 CA. All rights reserved.



cache usage scenarios event driven (push and pull)



maymainframemadness 2012

Copyright © 2012 CA. All rights reserved.



cache usage what you might put in

- Data Layer
 - XML documents composed with CA IDMS data
 - Edit and code tables
 - Recent activity from outside web services
- Business Layer
 - Organized documents for reports and forms
 - Information dashboard
- Service Layer
 - User session data
 - Dynamic web page content
 - XML documents transformed
 - Grid
 - Chart
 - Report

maymainframemadness 2012

Copyright © 2012 CA. All rights reserved.



cache usage options

- In-process
 - .NET, Java and C++ all sport in-process classes in their respective foundation class libraries
- Machine-level
 - Products like nCache facilitate caching at the machine level
 - Not reliable as the cache disappears when the machine fails
- Cluster caching
 - Stripes cached data across several machines
 - A little like RAID
 - Scalable and reliable

maymainframemadness 2012

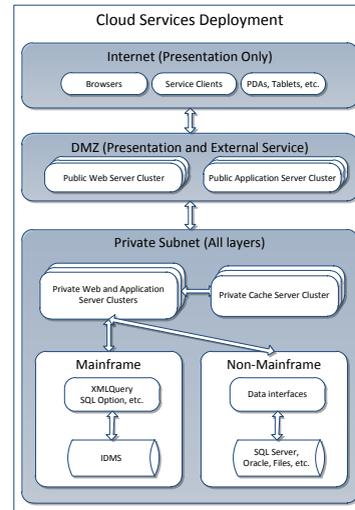
Copyright © 2012 CA. All rights reserved.



cloud computing deployment

The key to scalability and reliability

- Internet
 - Presentation only
- DMZ (Presentation and External Service)
 - Public web and application server
 - Presentation and Service Layers
- Private Subnet
 - Private network
 - All layers



maymainframemadness 2012

Copyright © 2012 CA. All rights reserved.



closing

- Cloud computing is a major evolution
 - Document oriented, scalable and reliable
 - Based on architecture and XML
- Cloud computing can be used for both public and private applications
- CA IDMS is a natural for Cloud Computing
 - Use an architecture for success
 - Get it, use it, change it, save it and leverage your current investments
- Contact information
 - Email: tom.hebert@obj-ex.com
 - Phone: (480) 788-6601

maymainframemadness 2012

Copyright © 2012 CA. All rights reserved.



thank you

maymainframemadness 2012



legal notice

© Copyright CA 2012. All rights reserved. All trademarks, trade names, service marks and logos referenced herein belong to their respective companies. No unauthorized use, copying or distribution permitted.

THIS PRESENTATION IS FOR YOUR INFORMATIONAL PURPOSES ONLY. CA assumes no responsibility for the accuracy or completeness of the information. TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENT "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. In no event will CA be liable for any loss or damage, direct or indirect, in connection with this presentation, including, without limitation, lost profits, lost investment, business interruption, goodwill, or lost data, even if CA is expressly advised of the possibility of such damages.

Certain information in this presentation may outline CA's general product direction. This presentation shall not serve to (i) affect the rights and/or obligations of CA or its licensees under any existing or future written license agreement or services agreement relating to any CA software product; or (ii) amend any product documentation or specifications for any CA software product. The development, release and timing of any features or functionality described in this presentation remain at CA's sole discretion.

Notwithstanding anything in this presentation to the contrary, upon the general availability of any future CA product release referenced in this presentation, CA may make such release available (i) for sale to new licensees of such product; and (ii) in the form of a regularly scheduled major product release. Such releases may be made available to current licensees of such product who are current subscribers to CA maintenance and support on a when and if-available basis.

maymainframemadness 2012

Copyright © 2012 CA. All rights reserved.

