Learning How To Build Distributed, Scalable Applications With COM+

COM+ Team
Microsoft Corporation

Agenda

- Goals
- Scenario recap and requirements (continued)
- Using COM+ to meet application requirements (continued)
- Even easier - coming soon (part 2)
- Conclusion
Goals

Show by example that COM+ makes it:

- As easy to develop Server Components as it is to develop Client Components
- As easy to deliver enterprise apps as it is to deliver workgroup apps!

Technology Scenario

COM+
MTS
COM

COM+ Services:
- Events
- Security
- Load Balancing
- Queued Components
- In Memory Database

COM+ The Caring, Sharing Environment for your components

MTS Services:
- Transactions
- Resource Pooling
- Security
- Administration

The Model
Tools Support
Multi-Language Discovery (QI)
Application Scenario

- Online service like MSN or AOL
- User login causes two databases to be updated
  - Customer Billing Log
  - Customer login status
- Total connect time is tracked
- User is notified of various activities while they are logged in (i.e., chat rooms)

Application Flow

- Web clients
- Web infrastructure
- Business applications
- Security billing
- Real-time Profile-directed information
- Content
- Billing
- Existing billing system
- Customer online system
- Other systems

Web clients → Web infrastructure → Business applications
Tackling The Application Requirements - Part 2

- Ultimate customer response and availability
- Always accept legitimate Logon
- No freebies! Always bill for entire time online
- Use existing billing system
- Use existing dev resources
- Other...

- Keep billing, Customer Online and other systems consistent
- Real-time two way info on who is on
- Real-time info on what is going on
- Flexible administration
- Maximize use of platform, minimize development time

Contradictory Requirements!

- Always accept logon:
  - Implies that we have to complete the customer login even if the billing system is unavailable

- Always bill for entire time online:
  - Implies that we must not accept logons when billing system is not available
**Tolerance**

- **Issues:**
  - Availability
  - Resilience
  - Failure Handling

**COM+ solution: Queued Components**

- **Features:** delivery of creation, method invocation, and life cycle management over MSMQ queues with transactions
Queued Components Example

- Code and attribute walkthrough
  - Login example would not allow a person to logon if the billing database was not available
  - It is not necessary for the billing DB to be available right now as long as we are guaranteed that the login information will be recorded

- Ref: AS10

Tackling The Application Requirements - Part 2

- Ultimate customer response and availability
- Always accept legitimate logon
- No freebies! Always bill for entire time online
- Use existing billing system
- Use existing dev resources
- Other...

- Keep billing, Customer Online and other systems consistent
- Real-time two way info on who is on
- Real-time info on what is going on
- Flexible administration
- Maximize use of platform, minimize development time
Responding To Environment

- **Issues**
  - You Never Told Me That!
  - Changing Environment
  - Object Model? What Object Model?

**Solution:** COM+ Events

- **Features:** sources consume event sinks, system provided event class, managed binding and delivery options
Events Example

❖ Events Walkthrough
  • When the user logs in the customer status database is updated
  • In order to notify the user on things that they are interested you can publish the interesting events and have clients
    • Poll the status database on a regular basis and then push information to them; this is not very efficient doing a lot of unnecessary work; or you can...
    • Build specific interfaces, protocols, and infrastructure for notification
  • Code walkthrough

❖ Ref: AS07

---

Tackling The Application Requirements - Part 2

❖ Ultimate customer response and availability
  ✔ Always accept legitimate logon
  ✔ No freebies! Always bill for entire time online
  ✔ Use existing billing system
  ✔ Use existing dev resources
  ✔ Other...

❖ Keep billing, Customer Online and other systems consistent
  ✔ Real-time two way info on who is on
  ✔ Real-time info on what is going on
  ✔ Flexible administration
  ✔ Maximize use of platform, minimize development time
Even More Scalability And Availability

- **Issues:**
  - We’ll never go above 600 clients! Honest!
  - Of course my application scales
    - What hardware are you running?

- **Solution:** COM+ Dynamic Load Balancing
  - **Features:** system provided response time load balancing engine, activation server machine selection
Load Balancing Example

- Load Balancing Walkthrough
  - As the online service increases in popularity they will need to run the application on multiple servers
  - Load Balancing will distribute the work of the application over the available servers
  - In COM+ Preview, balancing algorithm is based on tracker statistics
- LB set up and attribute walkthrough
- Ref: AS08

Tackling The Application Requirements - Part 2

- Ultimate customer response and availability
- Always accept legitimate logon
- No freebies! Always bill for entire time online
- Use existing billing system
- Use existing dev resources
- Other...

- Keep billing, Customer Online and other systems consistent
- Real-time two way info on who is on
- Real-time info on what is going on
- Flexible administration
- Maximize use of platform, minimize development time
Ultimate Performance

- **Issue**
  - No matter how well I do everything the Database is still the bottleneck!

**COM+ Solution:** IMDB provides memory speed access to data, transient state management
**IMDB - How It Works**

- **HTTP**
- **DCOM**
- **Client Component Process**
- **IMDB Proxy**
- **IMDB Core**
- **IMDB Node**
- **Tables in Shared Memory**
- **R/W**
- **R/O**

A “typical” IMDB Node

**IMDB Example**

- **IMDB Walkthrough**
- **Selecting data for use with IMDB**
- **IMDB code and administration walkthrough**
- **Ref: AS11**
Tackling The Application Requirements - Part 2

- Ultimate customer response and availability
- Always accept legitimate logon
- No freebies! Always bill for entire time online
- Use existing billing system
- Use existing dev resources
- Other...

- Keep billing, Customer Online and other systems consistent
- Real-time two way info on who is on
- Real-time info on what is going on
- Flexible administration
- Maximize use of platform, minimize development time

Ultimate Scalability

- Issue:
  - You are doing Pooling
  - Why can’t I?

[Diagram showing COM+ Server Process and Object instances with Application Client and $$$$]
Ultimate Scalability

- **COM+ Solution:**
- **Object Pooling:** ultimate efficiency for components whose “in Method time” is less than their “construction Time”

Ultimate Scalability

- **Object Pooling Walkthrough**
- **Ref:** AS14
Tackling The Application Requirements - Part 2

- Ultimate customer response and availability
- Always accept legitimate logon
- No freebies! Always bill for entire time online
- Use existing billing system
- Use existing dev resources
- Other...
- Keep billing, Customer Online and other systems consistent
- Real-time two way info on who is on
- Real-time info on what is going on
- Flexible administration
- Maximize use of platform, minimize development time

Dealing With Legacy Apps

- COM+ Solution: compensating Resource Manager
- Features: easy template for wrapping an application or a resource so that it can participate in transactions with other resources
How Are We Doing?

- Ultimate customer response and availability
- Always accept legitimate logon
- No freebies! Always bill for entire time online
- Use existing billing system
- Use existing dev resources
- Other...
- Keep billing, Customer Online and other systems consistent
- Real-time two way info on who is on
- Real-time info on what is going on
- Flexible administration
- Maximize use of platform, minimize development time

Even Easier!

- Coming soon to a language near you: Attribute Based Programming
  - In Part 1 “even easier” we showed integration of the IDE using Attributes and administration
  - The logical next step is direct Attribute Based Programming (ABP)
  - ABP demo
Conclusion

COM+ makes it:

- As easy to develop Server Components as it is to develop Client Components
- As easy to deliver enterprise apps as it is to deliver workgroup apps!

Call To Action

- Architecting, designing, developing:
  - Assume COM+ level of service and use Windows NT 5.0 Beta 2 now
- Deploying and in production:
  - Use MTS now - your components will be uplifted when COM+ is installed
- Everything discussed will be a standard feature of Windows NT® 5.0
Resources

- http://www.microsoft.com/com
- http://www.microsoft.com/msdn
- PDC CD:
  - COM+ SDK
  - COM+ Wizard preview
- Hands-on theatre
  - All the code from the COM+ sessions
  - See Visual C++® sessions for more on ABP!

Where do you want to go today?